FACILITY LAYOUT USING SWARM INTELLIGENCE

C. T. Hardin

Department of Computer Science and
Computer Engineering
University of Louisville
Louisville, KY 40292
cthard01@louisville.edu

John S. Usher

Department of Industrial Engineering
University of Louisville
Louisville, KY 40292
john.usher@louisville.edu

ABSTRACT

Facility layout problems are intriguing combinatorial problems that involve determining the location and shape of various departments in a facility based upon inter-department volume and distance measures. We propose a method that divides a facility into a swarm of intelligent tiles and devises a set of simple rules for tile behavior. Using these simple rules, the tiles self-organize, and a solution to the layout problems evolves. This method provides improved results compared to CRAFT, which is one of the primary methods for facility layout in use today.

1. INTRODUCTION

The facility layout problem is one that involves the arrangement of departments within a facility to meet one or more objectives. McKendall, Noble, and Klein[1] citing Mecklenburgh [2] and Francis *et al*. [3] give a useful list of objectives that are typically considered in most layout problems. Objectives that are pertinent to this paper include:

- Minimize material handling cost, time and frequency of handling.
- Minimize capital and operating cost in equipment and plant.
- Increase effective and economical use of space.
- Facilitate the manufacturing process and flow of operation.
- Maintain flexibility of arrangement and operation.
- Provide for safe and efficient construction.

Because of the quantity and variety of objectives, a large number of algorithms are proposed in the literature; most of which involve the development of the "block layout" which is then evaluated with respect to the stated objectives. The block layout is the most common manner of representing the solution to the facility layout problem. It specifies the relative location and size of departments in the facility. Typically, it is represented in either a discrete (as tiles on a grid) or a continuous fashion (with a centroid, perimeter, area, length, and/or width values to specify exact placement within the layout).

The discrete facility layout problem was first modeled as a Quadratic Assignment Problem (QAP) by Koopmans [4],where all departments are assumed to be of equal sizes. A review of many QAP-based models and heuristics can be found in Kusiak and Heragu [5], and Meller and Gau [6], the latter of which states "...cases of the QAP can only be found for problems with less than 18 departments, and it is not possible to solve even small problems with a few unequal-area departments."

Very few models have been proposed for continuous representations of the facility layout problem. Montreuil [7] presented a mixed-integer programming model for the facility layout problem based on a continuous representation. A similar model was also proposed by Heragu and Kusiak [8].

Many of the proposed approaches found in the literature result in irregularly shaped departments (long and narrow, non-rectangular, or split, which is the worst case). CRAFT by Armour and Buffa [9], and MULTIPLE by Bozer *et al*. [10] are perhaps the most well-known approaches. These algorithms attempt to either (a) achieve an optimal adjacency based score, which is a subjectively weighted value of department closeness, or (b) minimize the material handling distance in the layout.

Because of the constrained multi-objective nature of real facility design problems, true "optimal" layouts, i.e. ones that meet all objectives and constraints are not practically achievable. The challenge in this area of research is to find fast, easy-to-use algorithms that yield "good" solutions, i.e. those that meet the stated objectives, but also have characteristics that are applicable to an actual operating facility.

2. PARTICLE SWARM

Particle Swarm Optimization is a population based optimization technique by Kennedy and Eberhart [11]. It exhibits some evolutionary attributes in that a series of particles, each representing a solution, are allowed to fly through a solution space. The particles cooperate to converge upon a solution.

In this paper, many solutions are considered infeasible due to the requirement that each department may not be split. Thus, we arrived at a slightly different paradigm.

Each department is represented by a number "tiles" representing that department's relative size in the overall facility. We then give each tile a simple rule and allow it to use these rules to self-organize with the remaining tiles into a feasible (contiguous) solution. Then each tile uses a second simple rule to transform one solution toward a potentially improved new solution. Thus, one solution evolves into another.

3. MODEL DESCRIPTION

The facility to be analyzed is divided into a grid of fixed length and width that represents the dimensions of the facility. Each grid element is represented by a tile, which is a member of the swarm. Each department in the facility is represented by a fixed number of tiles based upon its relative fractional area of the entire facility. If a department is required to be in a fixed position, those parameters can be specified.

The final input is the volume matrix $[v_{ij}]$, which represents the relative flow of materials from department *i* to department *j.*

Each tile has two possible rules that govern the direction of its movement:

1) A tile will move towards the centroid of its own department.
2) A tile will move toward the centroid of department with which it has volume relationship.

Rule 1 is applied to allow tiles to self organize into contiguous departments. Rule 2 is applied to allow a department to locate near departments with which a high traffic volume relationship exists.

To initialize the algorithm, tiles (except those with a fixed location requirement) are randomly placed in the floor plan grid. (Figure 1.)

At the beginning of each iteration, the location of the centroid of each department is calculated. A pair of departments (d_{from}, d_{to}) is selected using the "roulette wheel" method [12] with more weight being given to department pairs with higher volume relationships. All tiles belonging to department d_{from} are then allowed to move in the direction of the centroid of department of d_{to}. At the end of each iteration, all tiles are allowed to move toward their own department's centroid until contiguity is achieved. The score of the current solution is then logged. If this solution has the best score thus far, its solution space is also recorded. This process continues until the predetermined number of iterations is reached. (Figure 2.)

8	1	4	7	3	10	6	10	10	6
1	6	7	10	7	5	7	8	2	1
9	2	10	3	3	8	10	6	4	3
3	9	5	1	3	6	3	7	6	10
8	10	7	5	8	7	8	3	6	4
1	4	6	10	6	4	1	1	5	10
4	7	4	7	6	9	6	9	2	1
3	1	1	3	8	1	6	8	1	2
7	1	10	3	7	6	10	10	6	1
10	8	5	6	1	10	7	6	3	2

Figure 1. Example initial grid with 10 departments. Tile locations are randomized.

8	8	8	8	7	7	7	7	7	7
8	8	8	8	7	7	7	7	10	10
9	9	8	3	3	3	7	7	10	10
9	9	3	3	3	3	10	10	10	10
1	1	3	3	3	3	10	10	10	10
1	1	3	6	6	6	5	10	10	10
1	1	6	6	6	6	5	5	5	5
1	1	1	6	6	6	4	4	2	2
1	1	1	6	6	6	4	4	2	2
1	1	1	6	6	6	4	4	4	2

Figure 2. Example solution with all tiles forming contiguous departments.

4. EXPERIMENTS

For a metric to compare solutions of various algorithms, we chose the Volume-Distance Product (VDP)—a common measure of facility layout efficiency. The layout with the lowest VDP is deemed more efficient. VDP is defined as:

$$VDP = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} d_{ij} v_{ij}$$

where:

c_{ij} = cost of transportation between departments *i* and *j*.

d_{ij} = rectilinear distance between centroids of department *i* and *j*

v_{ij} = flow volume between departments *i* and *j*

Technically, the optimal solution, based upon VDP, would consist of a series of concentric rectangles. In such case, the centroids would all be co-located and the VDP would be zero. However, such a layout is not a practically feasible solution.

5. RESULTS

We used 2 datasets from DePuy and Usher [13], and compared the results of the Layout Swarm to CRAFT. Since CRAFT is a deterministic algorithm, its result is governed by its starting condition. In [13], CRAFT was run with 10 different starting conditions. For comparison, we ran the Layout Swarm using 10 different random starting conditions. (Being evolutionary, the Layout Swarm is not dependent upon its starting condition.) Results are presented in Table 1, below.

Table 1: Comparison of results of this paper with CRAFT

Dataset [13]	Best VDP Solution Found (lower value is better)	
	This paper	CRAFT
M11	1,378	1,368
M15	32,752	34,301

Dataset M11 was a relatively small problem consisting of 11 departments in a 6 x 6 grid. With this smaller problem, the results of CRAFT and this paper were essentially the same. The range of CRAFT's solutions was 1,368 to 1,646, and Layout Swarm had a solution range of 1,378 to 1,460.

Dataset M15 was a more complex problem consisting of 15 departments in a 25 x 25 grid. In this case, the Layout Swarm reduced the traffic cost by 4.52% compared with CRAFT. We also note that CRAFT solutions had a range of 34,301 to 47,634, while the Layout Swarm had a range of 32,752 to 35,773.

6. SCOPE AND LIMITATIONS

With the current implementation, the solutions generated result in contiguous departments; however, the departments are not rectangular. This is consistent with other available algorithms and remains an open item for further research.

7. CONCLUSIONS

This paper presents a new conceptual approach to facility layout and a new application domain for swarm intelligence. The solutions generated result from emergent behavior in the absence of an explicit objective function. The objective is merely implied to each tile, but no attempt is made to calculate the value of the objective function during the iteration process.

The conceptual framework is extremely easy to explain to a practitioner in the area. Even though it is simple in nature, it is robust and adaptable to a wide variety of facility configurations that have been proven intractable. Unlike many deterministic algorithms, the evolutionary nature of this algorithm provides the investigator with a number of practical solutions in a very short period of time.

As Kennedy and Eberhart [11] so eloquently stated, this solution belongs to a class of algorithms that "allows wisdom to emerge rather than trying to impose it, (and) … seeks to make things simpler rather than more complex."

8. REFERENCES

[1] A. R. J. McKendall, J. S. Noble, and C. M. Klein, "Facility layout of irregular-shaped departments using a nested approach," *International Journal of Production Research*, vol. 37, no. 13, pp. 2895-2914, 1999.

[2] J. C. Mecklenburgh, *Process Plant Layout*. New York: Halstead Press, 1985.

[3] R. L. Francis, L. F. McGinnis, and J. A. White, *Facility Layout and Location: An Analytical Approach (2nd Ed)*. Englewood Cliffs, NJ: Prentice Hall, 1992, pp. 27-171.

[4] T. C. Koopmans and M. Beckmann, "Assignment Problems and the Location of Economic Activities," *Econometrica*, vol. 25, no. 1, pp. 53-76, Jan.1957.

[5] A. Kusiak and S. S. Heragu, "The facility layout problem," *European Journal of Operational Research*, vol. 29, no. 3, pp. 229-251, June1987.

[6] R. D. Meller and K.-Y. Gau, "The facility layout problem: recent and emerging trends and perspectives," *Journal of Manufacturing Systems*, vol. 15, no. 5, pp. 351-366, 1996.

[7] B. Montreuil, "A modeling framework for integrating layout design and flow network design," in *Proc.of the Material Handling Research Colloquium, Hebron, KY* 1990, pp. 43-58.

[8] S. S. Heragu and A. Kusiak, "Efficient models for the facility layout problem," *European Journal of Operational Research*, vol. 53, no. 1, pp. 1-13, July1991.

[9] G. C. Armour and E. S. Buffa, "Heuristic algorithm and simulation approach to relative location of facilities," *Management Science*, vol. 9, pp. 294-309, 1963.

[10] Y. A. Bozer, R. D. Meller, and S. J. Erlebacher, "Improvement-type layout algorithm for single and multiple-floor facilities," *Management Science*, vol. 40, no. 7, pp. 918-932, 1994.
[11] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc.IEEE International Conference on Neural Networks,* 4 ed 1995, pp. 1942-1948.
[12] *The Practical Handbook of Genetic Algorithms*, 2nd ed. Boca Raton: Chapman&Hall/CRC, 2001, pp. 177-178.
[13] G. W. DePuy, J. S. Usher, and T. J. Miles, "Facilities Layout Using CRAFT Meta-Heuristic," in *Proc.Industrial Engineering Research Conference* 2004.