UNLV | UNIVERSITY LIBRARIES

# University of Nevada, Las Vegas (UNL)
# Interlibrary Loan Article/Chapter Scanning Slip

**ILLiad TN: 128076**

**ILL Number: 79276365**

**Location:** UNLV Book Stacks NOT CHCKD OUT

**Call #:** **TS155.6 .I565 1993**

**Journal Title:** Intelligent manufacturing systems ; preprints of the IFAC workshop, Dearborn, Michigan, USA, 1-2 October 1992 /

**Volume:**
**Issue:**

**Month/Year:** October 1992
**Pages:** 235-239

**Article Title:** Chierotti, M. and Rozenblit, J.W.; Simulation Modeling in Factory Layout Optimization
**Article Author:**

**ISXN:** 9780080420370 ;

**Patron:** Valenzuela, Michael L

**Ariel:** 150.135.238.50

**Odyssey:** 150.135.238.6

**Email:** askill@u.library.arizona.edu

**Fax:** (520) 621-4619

**Borrower: AZU**
UNIVERSITY OF ARIZONA
LIBRARY
INTERLIBRARY LOAN
1510 EAST UNIVERSITY
TUCSON AZ 85721-0055

**Cost:**
**MaxCost:** $50IFM

**Resend Requests (Borrowing Library Only):**

Resend requests should be emailed to illiad@unlv.edu. Please include the ILL Number and pages needed.

6/22/2011 4:32 PM

# SIMULATION MODELING IN FACTORY LAYOUT OPTIMIZATION

M.Chierotti and J.W. Rozenblit

Department of Electrical and Computer Engineering, The University of Arizona, Tucson, Arizona, U.S.A.

Abstract This paper presents a new factory layout methodology and its integration in a simulation-based environment for FMS design. Once machines and material handling devices have been selected, a layout synthesis tool generates a workscene model used to plan robot trajectories and to investigate material handling control strategies by means of discrete event simulation. Layouts are represented as orthogonal structures of loosely packed arrangements of rectangles. They must satisfy a set of qualitative constraints and minimize material handling costs - a function of distances among components. Simulation results are used to modify the input data to the layout optimization tool, and the layout design process is iterated until a satisfactory solution is reached.

Keywords artificial intelligence, automation, computer-aided design, computer simulation, manufacturing processes, modeling, simulation

## INTRODUCTION

The physical layout of machines and material handling devices greatly influences various aspects of a production system such as the ease of installation and maintenance, safety, and dynamic performance. Due to the complexity of production systems, simulation models are needed to verify and optimize the impact of the physical layout on their dynamic behavior.

In this paper, we briefly summarize a layout optimization technique (Chierotti, 1991) capable of synthesizing device arrangements which satisfy installation and maintenance constraints and minimize material handling costs. Generated layouts are the basis for motion planning and synthesis of the system's control strategies.

## LAYOUT OPTIMIZATION

A solution of the layout design problem must satisfy a given set of qualitative requirements and minimize the material flow cost. The complexity and the success of the solution process is greatly influenced by the structure of the design space and by the type and number of operators used to manipulate it. In particular, the structure of this space should reflect the qualitative and quantitative design aspects perceived by a human designer.

In addition, a manufacturing plant is a system that evolves over time. For this reason, both the requirements and the performance indexes subdivide into two categories: static and dynamic. While static entities (e.g., adjacency between objects, fixed equipment costs, or length of a path) are evaluated independently of time, dynamic entities (e.g., time to move material from location to location, or work in progress) need to be monitored during operations. In the layout problem, static and dynamic entities are intrinsically connected. The production plan requires the most convenient arrangement of equipment in order to satisfy static criteria. The equipment layout affects the system dynamic behavior by determining timing and trajectories for material handling devices. An ideal approach to the problem would be to optimize the layout while dynamically simulating the production plan in order to synthesize the best overall solution. If, during the search process, dynamic entities are approximated by static ones, the need for simulation is greatly reduced. In fact, a search process limited to static evaluations could considerably narrow the set of promising solutions and facilitate a final refinement based on the actual simulation runs.

### Design Space Definition

A production plan is defined by a sequence of technological operations (assembly and/or machining) to be carried out by a set of devices. Material flow among machines is assured by material handling devices and production buffers. The layout of these devices must satisfy a set of qualitative and quantitative criteria. Qualitative (or symbolic) criteria can be interpreted as topological relationships. For example, two devices must be adjacent, or a device must be next to a given wall. Quantitative criteria are expressed by numeric entities, such as distance from a wall or length of a path from one device to another.

Layouts satisfying the same set of qualitative criteria constitute a class of solutions. Solutions inside an equivalence class have in common the same topological relationships among objects and satisfy the same qualitative criteria. For example, three devices placed in a row constitute a class of equivalent solutions containing an infinite set of possible layouts, each of them with different device coordinates, but all satisfying the same spatial relationships.

In other words, topological relationships are an equivalence relation used to subdivide the continuous design space into a finite number of partitions. These partitions are therefore exhaustively enumerable. The design process proceeds at two levels: the search for the "best" class of solutions and the search for the best solution inside a class. The application of qualitative criteria to an equivalence class generates a performance figure valid for all the elements of the class. Quantitative criteria allow us to search for the best element inside a class and to determine the related performance figure. The combined class performance index and best element performance index give the global performance index of a solution.

In order to describe classes of solutions, a layout description must capture both the qualitative and quantitative aspects. The adopted model is based on the following assumptions:

* The layout problem is restricted to a bidimensional space. Machines, material handling devices and architectural components appear as bidimensional and non-overlapping objects on a floorplan.

* The floorplan is a rectangle with sides parallel to the Cartesian coordinate axes.

* Each object is a rectangle with sides parallel to the coordinate axes. The position of a rectangle is defined by the coordinates of its center.

* Each rectangle has sides of size $dx$ and $dy$ and is surrounded by a service area to ensure minimum clearance between objects. The surrounding service area has dimensions $clx$ and $cly$. Therefore the actual dimensions of objects are $dx+2clx$ and $dy+2cly$.

* An object can be rotated by 90 degrees around its center.

* Topological relationships among objects have orthogonal structure and are subdivided into left/right relationships in the $X$ direction and above/below relationships in the $Y$ direction (Flemming, 1986, 1989). In this way, a two color graph represents the layout topology. Each object corresponds to a node, while edges are triplets of the type (<color> <obj_i> <obj_j>), where <color> can be LEFT or BELOW.

Layout Procedure

The design algorithm handles the symbolic and numeric aspects of the design space description. The solution strategy adopted here is the generative approach, i.e., solutions are built by placing one object at the time on the floorplan. These partially completed layouts are the states on which an AI production system operates.

A generic state in the search process corresponds to a partially completed layout and is characterized by a list of preplaced objects, a list of placed objects, a list of free objects, a topology description involving preplaced objects and objects placed so far, and the layout cost (a measure of the layout compliance to the imposed qualitative and quantitative evaluation criteria).

In the initial state the workscene is empty or contains only preplaced objects. In the goal state all the objects are present in the workscene, and the global layout cost is minimized.

A set of operators is needed in order to move from state to state in the search state. From a given state with some objects already placed, an object is extracted from the list of free objects and placed in the workscene. The possible positions and orientations of the new object generate a set of child states, all with the same placed objects, but with different topology. This expansion process generates a tree of states. A successful search leads to a set of leaf nodes, where all the objects have been placed. The search process is exhaustive. In fact, changing the order of objects insertion does not preclude the possibility of generating a feasible leaf node, although it modifies the sequence of intermediate states needed to reach it.

The state expansion process is composed of two major phases: state generation and state validation/optimization. In the state generation phase, all possible locations for an object insertion are considered. This is a purely symbolic process that manipulates the syntax of the topology description. Syntactically valid layouts must undergo a state validation/optimization process. The first step, qualitative state validation and evaluation, gives a first symbolic interpretation of the layout. The state is validated against qualitative constraints.

The second step is the quantitative state validation. Some insertions, although syntactically correct, may violate geometric constraints and thus must be discarded. For this reason, once a new class of layouts (i.e., a new topology) has been generated, it must not be empty, that is, there must beat least one set of values for all object coordinates that makes it feasible. If this solution exists, it represents the starting point of the search for the best solution inside this particular class. This is done in the third step, the quantitative state optimization. This optimization process involves the modification of object coordinates in order to minimize the layout cost with respect to the quantitative evaluation criteria, while satisfying the topological constraints of the given class of solutions.

A search strategy guides the tree expansion process in order to gain the most efficient path to the best goal state. The adopted search strategy, based on the branch and bound algorithm, ensures solution optimality with respect to the cost function defined below.

Cost Function Definition

The PI_tot performance index, or cost, associated with a generic state is the sum of a qualitative performance index PI_qual and a quantitative performance index PI_quant.

Whereas the PI_qual value is constant for all solutions inside a class, the PI_quant value varies inside a class and reaches the minimum for the best solution inside this class. Therefore, when states in the search process are ranked by cost, the minimum PI_quant value is used.

To compute the qualitative performance index PI_qual, criteria (or rules) are applied to the current state and a corresponding score is recorded. For each applicable rule, if the imposed condition is satisfied, then no penalty is recorded, otherwise a positive penalty score applies. The penalty score is proportional to the importance of the rule. The sum of all rule scores generates the PI_qual.

The quantitative performance index PI_quant reflects both dynamic and static measurable numeric quantities. Dynamic quantities involve an estimate of the cost of moving material. Static quantities are distances between devices. The total cost for material handling is computed as:

$$PI\_mat = \sum_{ij} w_{ij} * d_{ij} \tag{1}$$

where d_ij is the distance between the generic devices i and j, and w_ij is the cost per unit distance. Considering now static quantities, the performance index is given by

$$PI\_mat = \sum_{ij} wr_{ij} * | R_{ij} - d_{ij} | \tag{2}$$

where wr_ij > 0 if devices i and j must be kept to a desired distance, while wr_ij = 0, otherwise. R_ij is the desired (or optimal) distance between devices and d_ij is the actual distance between them.

## SIMULATION LAYER

The factory layout methodology is being augmented with a simulation modeling layer for further layout optimization and performance evaluation (Chierotti, Rozenblit and Jacak, 1991).

There are important reasons to verify the layout performance through simulation. First, material handing costs and computed distances might not reflect the actual scenario. In particular, using robots and AGV's, the number and cost of their moves while empty could be large and should be taken into account. On the other hand, these empty moves would be determined by the dynamic behavior of the system under certain experimental conditions. Second, intelligent control strategies for robots could base their decisions on the actual layout parameters in order to optimize their dynamic behavior and could therefore significantly modify the predicted layout performance.

Simulation of design models is carried out in the DEVS-Scheme environment (Zeigler 1990). DEVS-Scheme is an object-oriented simulation shell for modeling and design that facilitates construction of families of models specified in a discrete event formalism.

Simulation results are used to evaluate the dynamic behavior and eventually to modify the input data to the layout optimization tool. The layout design process is iterated until a satisfactory solution is reached.

## CASE STUDY

This case study illustrates an application of the proposed methodology in the context of an integrated framework for simulation based design of flexible manufacturing systems, as proposed by Chierotti, Rozenblit and Jacak (1991). The example used was to generate an optimal layout for the electric motor assembly workcell described in Chierotti (1991). The chosen plant configuration consists of a single workcell with four input feeders, one output feeder and three machines. Each input feeder supplies a part to a machine, while the output feeder receives the final assembly, as shown Table 1. The workcell is serviced by a single robot that moves parts among feeders and machines. External storage areas are connected to the workcell feeders through conveyors.

TABLE 1: Feeder Data

| Feeders | Role | Assigned Part |
|---|---|---|
| Feeder 0 | input | stator |
| Feeder 1 | input | rear plate |
| Feeder 2 | input | rotor |
| Feeder 3 | input | front plate |
| Feeder 4 | output | motor |

Each machine receives parts and produces a subassembly as shown in Table 2. In this table the list of parts in square brackets indicates that these parts have been connected into a subassembly and are now a single product.

TABLE 2: Machine Data

| Machines | Parts Needed | Subassemby Produced |
|---|---|---|
| Machine 0 | rear plate, stator | [rear plate, stator] |
| Machine 1 | [rear plate, stator], rotor | [rear plate, stator, rotor] |
| Machine 2 | [rear plate, stator, rotor], front plate | [rear plate, stator, rotor, front plate] |

The choice of machines and devices allows to determine dimensions and clearances of each object to be placed in the workscene. This information is presented in Table 3.

TABLE 3: Devices Dimensions

| Device | dx | dy | clx | cly |
|---|---|---|---|---|
| Feeder 0 | 1.00 | 1.00 | 0.25 | 0.25 |
| Feeder 1 | 1.00 | 1.00 | 0.25 | 0.25 |
| Feeder 2 | 1.00 | 1.00 | 0.25 | 0.25 |
| Feeder 3 | 1.00 | 1.00 | 0.25 | 0.25 |
| Feeder 4 | 1.00 | 1.00 | 0.25 | 0.25 |
| Machine 0 | 2.00 | 1.00 | 1.00 | 1.00 |
| Machine 1 | 3.00 | 1.00 | 0.50 | 0.50 |
| Machine 2 | 3.00 | 1.00 | 0.50 | 0.50 |

The objects are not allowed to rotate and none of them has been already preplaced in the workscene. The floor plan has dimensions of 15 and 10 length units respectively in the $X$ and $Y$ directions. The initial layout is therefore a state in which the lists of preplaced and placed objects are empty and no topological relations are specified.

Tables 1 and 2 allow to determine the material flow between objects, as shown in Table 4.

TABLE 4: Material Flow

| From Device | To Device |
|---|---|
| Feeder 0 | Machine 0 |
| Feeder 1 | Machine 0 |
| Feeder 2 | Machine 1 |
| Feeder 3 | Machine 2 |
| Machine 0 | Machine 1 |
| Machine 1 | Machine 2 |
| Machine 2 | Feeder 3 |

The desired layout must satisfy the following set of qualitative constraints:

* Input feeders must be at the four corners of the workcell, so that they can be accessible from two contiguous sides of the workcell contour. Feeders 0 and 1 must be at the top, on the left and on the right respectively, while feeders 2 and 3 must be at the bottom, on the left and on the right respectively.

* The output feeder must be accessible from the bottom.

* To simplify material flow, devices exchanging material must be adjacent.

* Machine 0 should have its left side free for inspection and maintenance.

Moving now to the quantitative aspect of the problem, material flow information of Table 4 allows us to specify the list of material handling costs w_ij. All costs w_ij per unit distance have the same value. No optimal distances between components are required. This completes the problem definition.

The synthesized layout is shown in Figure 1. In this figure, solid and dashed lines between object centers represent left/right and above/below relationships, respectively. Each object is shown with the corresponding center coordinates. As it is easily verifiable, all qualitative constraints are satisfied.

The result obtained allows us to build a discrete event simulation model by adding timing information for feeders, machines and material handling devices and to evaluate workcell dynamic performance indexes (Chierotti, Rozenblit and Jacak, 1991).

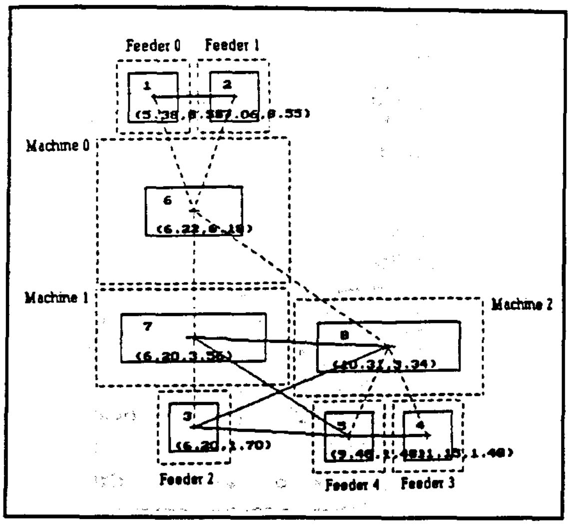


Fig 1. Sintesized Workcell Layout

All feeders have a capacity of two parts. Input feeders have a processing time of 5 time units, while the output feeder has processing time of 10. Processing time is 30 for Machine 0 and Machine 2 and 20 for Machine 1. To complete the system description, a model for the robot is introduced. When a part has to be handled, the robot moves from its current position to the device where it has to pick it up, then holding the part, moves to the second device and places it. When the robot is done, it stops near the destination device waiting for a new task. Robot trajectories are determined by a very simple bidimensional model. Trajectories are straight lines between points. The robot accelerates with constant acceleration, then moves with constant velocity, finally decelerates with constant deceleration until it stops.

Simulation Results

The production of a batch of ten motors has been simulated. The utilization profile of workcell machines is shown in Table 5, where: Sa signifies that machine is free, Sb signifies that machine is loading, Sc signifies that machine is busy processing an operation, and Sd signifies that machine has completed an operation.

TABLE 5: Utilization Statistics [%].

| Machine | $S_a$ | $S_b$ | $S_c$ | $S_d$ |
|---|---|---|---|---|
| $M_0$ | 40.22 | 13.34 | 14.83 | 31.59 |
| $M_1$ | 31.71 | 47.37 | 9.89 | 12.01 |
| $M_2$ | 60.57 | 17.50 | 14.83 | 7.08 |

The Robot was idle for 2.32% of the time, moved empty for 57.28%, and moved holding a part for 40.40%.

The results illustrate the importance of the simulation layer. The percentage of time the machines are free (i.e. in state Sa), along with a negligible amount of time the robot is idle, suggest that the robot is not able to service the workcell efficiently and constitutes a bottleneck. Its speed or control strategy could be improved.

Furthermore, the robot spends about the same amount of time moving empty and moving while holding a part. The simulation results allow us to modify the input data for the layout optimization tool, giving unit costs w_ij depending on the frequency of each move. This process can be iterated until the robot moves and the material handling costs w_ij are consistent.

## CONCLUSIONS

The integration of automated factory layout techniques in a comprehensive FMS design framework is well justified. Such a framework must also include product design tools, production planning, equipment and material handling selection procedures, and intelligent workcell control algorithms. Simulation is essential in verifying the operation of a proposed manufacturing design.

## REFERENCES

Chierotti, M. (1991). Factory Design Layout: A combined Artificial Intelligence and Simulation Approach, *MS Thesis*, 1991, Department of Electrical and Computer Engineering, University of Arizona.

Chierotti, M., Rozenblit, J.W., Jacak, W. (1991). A Framework for Simulation Design of Flexible Manufacturing Systems; *Proceedings of the 1991 Winter Simulation Conference*, December 8-11, 1991,Phoenix, Arizona.

Flemming, U. (1986). On the Representation and Generation of Loosely Packed Arrangements of Rectangles, *Environment and Planning B: Planning and Design*. 1986, Vol. 13, pp. 189-205.

Flemming, U. (1989) More On the Representation and Generation of Loosely Packed Arrangements of Rectangles. *Environment and Planning B: Planning and Design*. 1989, Vol. 16, pp. 327-359.

Zeigler, B. P. (1990). *Object-Oriented Simulation with Hierarchical, Modular Models*. Academic Press.