1994

Reduction of acquisition time through optimal component presentation at an assembly workstation

David O. Hunt
Carnegie Mellon University

Robert H. Sturges

Carnegie Mellon University.Engineering Design Research Center.

Reduction of Acquisition Time Through Optimal Component Presentation at an Assembly Workstation

D. Hunt, R. Sturges

EDRC 24-109-94

Reduction of Acquisition Time Through Optimal Component Presentation at ah Assembly Workstation

David O. Hunt, Graduate Research Assistant
Robert H. Sturges, Associate Professor

Department of Mechanical Engineering and the
Engineering Design Research Center
Carnegie MeHon University
Pittsburgh, PA 15213

ABSTRACT

An effective presentation of components at the workstation can have a significant impact in reducing assembly time. Our goal is to develop optimal presentation plans based on design for assembly theory. The assembly factors recognized as relevant to both parts acquisition and assembly workstation layout are recognition, orientation, weight, and handling distance. This study considers a single manual operator at an assembly station, with the components in rectangular bins of differing sizes and aspect ratios. Ninety degree rotations of the bins are allowed for minimizing potential handling distance. The assembly task is modeled with multiple assembly points representing the final location of the components. Components may or may not be preoriented in the bins, with preorientation removing the recognition and orientation time penalties. The problem formulation employs Mixed Integer Non-Linear Programming (M1NLP), and numerical evidence suggests an NP-hard problem. Heuristic methods reduce computational effort to practical levels for realistic assembly tasks. Our results show that numerical optimization of assembly workstation layout can reduce the expected level of difficulty over random or manual workstation design methods.

1. INTRODUCTION

Assembly tasks occupy approximately 50 percent of total production time [Owen 1985] and labor costs are between twenty and thirty percent of total production costs [Whitney/Nevins 1989] [Owen 1985]. Reductions of assembly times will therefore have potentially significant effects in reducing assembly costs.

Given that a large fraction of product assembly time is taken in acquisition of the components being assembled [Sturges 1989a], a reduction in acquisition difficulty should lead to a significant reduction in assembly time, with a commensurate rise in efficiency. This type of analysis targets high-volume operations where even a saving of pennies per assembly translates to significant returns in reduced production costs. Less quantifiable benefits may also accrue due to decreased error rates.

This paper focuses on the layout of component bins at an assembly workstation, and offers a mathematical optimization technique, combined with some heuristics, which can reduce the time spent in component acquisition during the assembly process. Since the characteristics of parts have a substantial effect on acquisition difficulty, results from assembly workstation analysis give another source of quantitative information for improved product design [Sturges/Hunt 1994].

1.1 Assembly Workstations

Manual assembly of products often occurs at single-operator workstations, and the design of easy-to-assemble products has become a priority for manufacturers in recent years. The Design for Assembly (DfA) discipline which has grown from these concerns has as its primary goals the reduction of assembly time and errors. Many advances have been made in the area of product design; for example, Boothroyd/Dewhurst [1983], among others, have proposed design guidelines for reduction of time and errors. Efforts by Whitney/Nevins [1989] and Jayarman [1985] have focused on work cell layout

1.2 Sources of Assembly Difficulty

Overall assembly time is known to depend on several factors which desoribe task elements during acquisition of the parts (free motions and grasping) and part insertion 0ine motions and fitting). On average, acquisition accounts for about one third of the total time [Sturges 1989a], and this time is affected by the location and orientation of the components internal to the product.

The most significant factors relevant to assembly difficulty are feature recognition, orientation, weight, and handling distance. Evaluations of assembly difficulty based on measures of both effectors and tasks have been developed which are independent of assembly sequence [Sturges 1989b]. A measure of assembly difficulty and actual elapsed time due to these factors is given by the Index of Difficulty (ID) based on Fitt's Law [Fitts 1954]. An ID is defined as the base-2 log of the range divided by the resolution, and has the units of bits. For example, the ID arising from handling distance is found by taking the distance to be traveled (the range) and dividing by the needed accuracy (the resolution), as shown in Figure 1. The average time needed to perform a task is found to be linearly proportional to the ID over a wide range of tasks and effector types, although the constant of proportionality varies with the effector. For example, the manual assembler performance constant for handling distance varies between 90 and 110 msec per bit [Fitts 1954].

Feature recognition IP is defined as the difficulty for an assembler (human or robotic) to fecognize a feature of a component so that it may correctly be oriented for assembly [Kilani/Sturges 1992]. The feature recognition ID is calculated by taking the base-2 log of the largest dimension of the object divided by the size in the same direction of the feature being recognized. For example, the head of a screw could be recognized by the head, where the range would be the diameter of the head and the resolution would be the difference between the head and shank radii. Preorientation of components removes the necessity for such feature recognition [Khwaja/Radhakrishnan 1990].

Orientation ID represents the time necessary to actually rotate the component to the proper orientation for assembly. Preorientation of components can remove the necessity for orientation as well as for feature recognition time penalties, and thus result in significant time savings. Orientation difficulty can be calculated from orientation entropy [Sanderson 1984]: if the resolution of each rotation axis is held to 7.5 bits, or 2 degrees out of 360, predicted task times correlate well with empirical data [Boothroyd/Dewhurst 1983]. Axes of symmetry reduce the rotation necessary to correctly orient the part for assembly, thus reducing orientation times. Similar guidelines for automatic preorientation and feeding of small components to an assembly effector have also been developed based on empirical studied [Boothroyd/Dewhurst 1981].

Taken together, the times associated with part rotation due to random placement of parts in a bin accounts for about half the total acquisition time in manual assembly. The cost associated with this time serves as a strong quantitative motivation for preserving part orientation between manufacture anil assembly.

Weight ID is observed for components with a mass greater than about ten percent of the effector mass, and it increases with both the mass and the distance traversed [Wong/Sturges 1992]. This observation contrasts with earlier empirical results which only take the part weight into account [Boothroyd/Dewhurst 1983] and considers only human assemblers [Sturges, et al.1986].

This paper assumes that the actions of orientation and traversing the distance from the bin to the point of assembly are concurrent if a preoriented component is being rotated. Motions involving randomly-oriented components, however, require that the assembler first recognize the current orientation before the component can be correctly oriented. Thus the orientation and feature recognition times are not presumed to be concurrent with the traversal of the workspace for randomly-oriented components, and are added in as a penalty.

As mentioned above, the handling distance ID increases as the logarithm of the distance traveled [Sturges 1989a}. For handling distances greater than "arm's reach" a fixed time can be added to the task time for "stand and sit" motions. Predictions of assembly time based on Indices of Difficulty for ail of the above factors have been shown to correspond well with empirical results [Sturges/Wright 1989]. Other factors associated with part acquisition include smallness of the components, whether the component is hot, delicate, etc. These factors are not included in tills analysis because they affect assembly time regardless of the positions and orientations of the components in their bins or in the product.

2. REDUCTION OF ACQUISITION TIME

Assembly acquisition time can be divided into phases during which different actions occur. Three of these phases are the feature recognition phase, the orientation phase, and the handling distance phase. Examination of these phases and what they represent in product and process design may identify significant time sayings in the factory.

Examples of efficient assembly layouts show components presented in drawers, pallets, or other such holders. Also, components should not normally be allowed to lie free in the assembly area; e.g., screws are placed in small drawers or recesses. Rectangular component bins of varying dimensions are considered in this study, with only right-angle rotations of these bins permitted. While it is possible that non-right angle rotations might lower the handling distance, such freedom would greatly increase the dimensionality of the problem. Examining the effects of moving assembly point positions relative to the bins supplying the components is simplified by modeling an assembly task with discrete points [Drezner/Nof 1984].

2.1 Reduction of Handling Distances

Handling distances <»n be reduced in at least three ways. One way is to redesign the assembly workstation. Optimization of component bin placement around the assembly workspace has previously been examined with various objective functions, such as workstation area minimization [Yunis/Cavalier 1990], distance minimization [Drezner/Nof 1984], and an information-based predictive DfA theory [Hunt/Sturges 1993]. Minimizing the empty workspace through knowledge of the product size and ergonomics has also been extensively studied [McCormick/Sanders 1982] [Kvålseth 1983] [Clark/Corlett 1984] [Konz 1983].

Another way to reduce handling distances is to optimally locate the components in the product where the functionality of the product is insensitive to such location. Relocation may involve both planar and spatial layering of components to reduce product size [Sturges/Hunt 1994].

A third way is to redesign the components themselves to locate them closer to their source bins. Components that mate with others may have a range of mating options for which the component's function is not impaired.

Knowledge of preliminary product and workstation layouts may thus bring valuable information into the detailed component design stage [Sturges/Hunt 1994].

2*2 Reduction of Orientation Times

No orientation is required between grasping ahd use of the component in the assembly if a component has been preoriented for assembly by a bowl feeder or other device. Preorientation is effective for small components in mechanical assemblies [Boothroyd/Dewhurst 1991], and in the electronics industry [Kwaja/Radhakrishnan 1990]. Preorientation has not been significantly used in general for manual mechanical assembly, which suggests an area for further investigation.

A significant percentage of total orientation time is due to component asymmetries [Boothroyd/Dewhurst 1983]. Redesigning components to create rotational symmetries reduces the time required to orient the component for assembly when starting from a random orientation.

2.3 Reduction of Feature Recognition Times

An important factor in reducing feature recognition time is the degree to which components are subtly asymmetric. The worker must spend additional time determining which orientation is correct if the features of the component are not easily recognized. Reductions in feature recognition time are achieved by increasing the relative feature sizes on the component in accordance with Fitt's Law and [Boothroyd/Dewhurst 1981]. As this area of acquisition is not affected by the design of the workstation, it is outside the scope of this study.

3. PROBLEM FORMULATION

3.1 Variable Definitions

Our model of the assembly workstation uses continuous variables to represent the x and y coordinates of the bin centroids. The centroid of the bins is used because the average component position in the bin will be at or near the centroid. Right angle rotations of each bin are represented by two binary variables, rot(i) and norot(i), only one of which can be active at a time. These two binary variables multiply the appropriate x and y dimensions so that the correct physical dimensions are expressed by non-overlap constraint equations. These nan-overlap equations include additional binary variables that describe the spatial relationship between the bins. Each bin also has fixed attributes such as component mass, whether the parts in it are stored at random or pre-oriented, the frequency of use in the assembly, where in the assembly they are used, and the bin dimensions.

3.2 Assumptions

As mentioned above, examples of efficient assembly layout show components presented in drawers, pallets, or other such holders. Components are not normally allowed to lie free in the assembly area, and even screws are placed in small drawers or recesses. Rectangular component bins are considered here, with only right-angle rotations permitted. While it is possible that non-right angle rotations could improve the packing density and hence lower the handling distance, such detail would greatly increase the dimensionality of the problem.

An assembly task is modeled with discrete points since the handling distance required to move the incoming component is much greater than the tolerance of component placement. Point modeling of assembly processes has been effectively used by previous researchers [Drezner/Nof 1984]. Inclusion of multiple assembly points adds little or no additional effort to the formulation set-up.

Fitt's Law will govern the motions for most manual assembly workstations. The traversal time has been found to be proportional to the square root of the distance below a handling distance ID of 3 bits. Between 3 and 4 bits there is a transition from a ballistic motion to a visually-controlled motion, and Fitts Law dominates above 4 bits [Gan/Hoffman 1988]. Considering an ID of 4 bits with a free motion resolution of 3 mm [Sturges 1989a], the maximum distance at which ballistic motion will have any effect is 64 mm, which is less than all distances in the manual assembly tasks modeled here.

3.3 Non-Overlap Constraints

Each pair of bins i and j are prevented from overlapping each other by spatial constraint equations which are modeled with four binary variables, viz.: left(i,j), right (i,j), above(i.j), and below(ij). These four binary relations determine the spatial relation between any two bins i and j, an example of which is shown in Figure 2. While it is possible for bin i to be both above and to the right of bin j simultaneously, this level of refinement is unnecessary in preventing overlaps. These spatial binary relation variables ate constraineid with respect to each other such that only one can be active at a time. For example, a non-overlapping constraint is expressed by:

$$x(i)+hsx(i) \leq x(j)-hsx(j)+M \cdot (1-left(i,j)) \qquad (1)$$

where x(i) stands for the x-coordinate of the bin centroid, hsx(i) stands for half of the corresponding x direction bin size, and M is a length that is large compared to the scale of the workspace and bins to be laid out. The rotations have been left out for simplicity, and will be treated below.

Equation 1 shows that if the binary relation i LEFT OF j is 1, then the equation will actively constrain the two bins' relative positions. Hence, the x-coordinate of bin i's right side is constrained from being larger than the x-coordinate of bin j's left side. If the binary relation LEFT OF is 0, the right side of i can be to the right of the left side of j by the amount M. In addition to the non-overlapping spatial constraints, the bins are also prevented from overlapping the edges of the workspace surface (e.g. a work table).

In certain instances one component, such as a fastener, may be required at several assembly points. Our formulation accounts for this by making the bins for these components large enough to hold the total number of these components for the assembly. These bins are then linked by auxiliary equations to have the same location and rotation. Since the binary relations for these overlapped bins to other bins are constrained to be equal, minimal additions to computation time result.

3.4 Bin rotations

If a rectangular bin has an aspect ratio substantially larger than unity and has its long axis oriented towards the assembly point, as bin 1 shown in Figure 3, it is possible that the decrease in handling distance could offset the time penalty required to return the component to the original orientation. The orientation of the component will be known, so recognition of the current orientation is unnecessary and the rotation of the component can be considered to be concurrent with the traversal of the handling distance. A criterion combining handling distance IDs and rotation time penalties may be formulated to show whether or not such a rotation would be advantageous:

$$K\log_2\left(\frac{L+D}{3}\right) - MAX\left[K\log_2\left(\frac{S+D}{3}\right), T_{rot}\right] > 0 \qquad (2)$$

In this criterion, L and S are the larger and smaller half-sides of the bin (in mm), respectively, K is the effector-specific constant relating ID to time (in sec/bit), D is the distance from the edge of tile, bin nearer the assembly point to the assembly point, and Trot is the time in seconds necessary to rotate the component through 90 degrees (back to the proper orientation). The 3 mm value in the denominator is the free motion resolution, marking the transition between free and fine motions [Sturges 1989a].

For example, if we choose a rotation time of 1 second to correctly orient the component [Boothrayd/Dewhurst 1983], a K of 0.1 seconds/bit [Fitts 1954], and a best-case D of zero, Equation 2 reduces to:

$$\log_2\left(\frac{L}{3}\right).MAX[\log_2(|),10]>0 \qquad (3)$$

Recognizing that the first term in the MAX function must exceed 10 to be selected, we find S $\geq$ 3072 mm, which is an unrealistic dimension for a manual workstation. If S is not larger than 3 meters, the above criterion calls for L to be larger than 3 meters. Therefore, at a manual workstation the rotation of a preoriented bin will never reduce the handling distance sufficiently to offset the rotation time penalty. For robotic effectors the results of this analysis may be different, since 3 meters is possible, although unlikely, and K may be small.

The above analysis ignores the effects of rotation of one bin on the surrounding bins. Consider the complete situation shown in Figure 3. While the rotation of bin 1 will never yield an improvement, the effects of bringing bin 2 in closer may sufficiently compensate for the time penalty to make this rotation worthwhile.

The criterion that considers another bin in the rotation of a preoriented component bin is given by:

$$F_1\left[\log_2\left(\frac{D+2L}{3}\right) - \frac{T_{rot}}{K}\right] + F_2\log_2\left(\frac{D+2L+A}{D+2S+A}\right) > 0 \qquad (4)$$

with S, L, K, D/^»icfTrot as define above, A is the half-side length of the other bin, and Fi and F2 are the components[1] frequency of use fronv bins 1 and 2, respectively. If we assume some best-case values for S and A, the 5 degrees of freedom of Equation 4 can be transformed into a function of the aspect ratio, the frequency ratio, and the distance from the assembly point, D.

For example, S and A are set to 25 mm to approximate the smaller dimension of a typical fastener storage drawer. Trot is known to be about 1 second. Creating the variables G and B to represent the ratio of Ft to F2 and the aspect ratio of bin 1 (L to S), respectively, and substituting into Equation 4 results in:

$$\left[\log^{\wedge} j\frac{25B}{3} - 10\right] + G\log g \pm^{\wedge}) \quad O \tag{5}$$

Figure 4 shows the "break-even" curves for three different values of D, with an aspect ratio range of one to five on the X-axis, and a frequency ratio of one to eight on the Y-axis. Even with a small D and a high aspect ratio, the frequency ratio required to make the rotation of bin 1 advantageous is large. With a D of 25 mm and an aspect ratio of 4 the required frequency ratio is 3.5, at 50 mm the minimum ratio is 3.9, and at 100 mm it becomes 4.4. Not only is this frequency ratio unlikely to occur in a realistic assembly, but if the arrangement of component bins has been optimized, the higher frequency bin is likely to have been placed closer to the clear workspace, rendering the analysis moot.

While the foregoing ignores the effects of rotation on additional neighboring bins, we may conclude that rotations of preoriented component bins are never advantageous. This heuristic has been borne out by not having observed rotations of bins containing preoriented components in over 200 component layout optimization tests of 25 different 15-bin layout problems for which the rotational freedoms have not been set [Hunt/Sturges 1993].

3.5 Objective Function

An objective function that predicts the assembly time for any preoriented bin i subject to the above variable definitions, assumptions, and other constraints may be given by:

$$Times = fi\text{-}MAX\left[|lK+mp-mi) \cdot \log_2\left|\frac{\Lambda\sqrt{dx^2+dy^2}}{V}\right| I. \ {*P*}^{rot} i\right] \tag{6}$$

where f is the frequency of use, K is the effector time constant, mp is ttie mass penalty constant [Sturges/Wong 1991], dx and dy are the x and y distances, in mm, from the bin centrokJ to the assembly pointtoeing considered, respectively. Again, the parameter 3 mm derives from the transition between free and fine motions, rp is the right angle rotation time penalty, and rot is a binary variable that is equal to unity if the bin has been rotated from its original orientation, and zero otherwise. For a bin wit) randomly oriented components, the objective function is given by the distance-based time, as randomly oriented components are insensitive to bin orientation.

By ignoring the constraint that the bins cannot overlap, while maintaining the constraint that they cannot intrude into the workspace, an infeaisible solution can be obtained that is lower than any possible feasible arrangement of bins, and hence lower than the global optimum. This value provides a lower bound on the objective function. Since this lower bound is infeasible, we know that

$$\text{Lower Bound} < \text{Global Optimum} \leq \text{Best Solution} \tag{7}$$

hence the solution is within a certain percentage of the global optimum. The example of Section 6 shows convergence of Equation 7 to within practical ranges of a few percent.

4. SOLUTION METHOD

4.1 Mixed-Integer Non-Linear Formulation

Since both the non-overlap constraints of Equation 1 and the objective function of Equation 6 have both binary and continuous variables, and the objective function also contains non-linear terms, a Mixed Integer Non-Linear Program (MINLP) method is used to find the minimum.

MINLP problems are generally solved by iterating between a Mixed-Integer Linear Program (MILP) formulation in which the non-linear functions are approximated by linear approximations, and a Non-Linear (NLP) formulation in which the binary variables are fixed [Viswanathan/Grossmann 1990]. The MILP solution phase gives the lower bound to the problem, as the linear approximations generally underestimate the non-iinear equation. The NLP solution phase represents the upper bound, and the goal is to decrease the difference between the two bounds to zero, at which point an optimum has been found. In MINLP problems the MILP phase of the solution process takes most of the computational effort [Quesada/Grossmann 1992], commonly dominating the NLP phase by over an order of magnitude. The problem presented in this paper is no exception.

A MILP algorithm solves a problem by first optimizing with the binary variables relaxed to continuous variables between 0 and 1. This relaxed MILP (RMILP) gives the absolute lower bound for the MILP problem. The solver then chooses binary variables whose values are near 0.5 and examines the effects on the objective function and constraint equations when the chosen variable is set to be zero or unity. The sequential examination of zero/unity choices is called branching, and as the variables are binary the solver is said to be searching a binary tree for solutions.

There are two basic solution methods possible: a depth-first search and breadth-first search. A depth-first search seeks out a feasible solution by following one vector of the binary tree down to a feasible solution consisting of a vector of binary variables. This feasible solution vector becomes an upper bound. The branching of binary variables can only increase the objective function (as variables move away from the lower bound relaxed solution). For any node of the search tree that exceeds the upper bound, all daughter nbdes of that node, including any feasible solutions, will also be above the upper bound, and the solver may safely remove branches whose partial solution objective function is higher than the current upper bound from further consideration. This technique, called branch and bound,, reduces the number of nodes that must be examined in order to obtain an optimal solution.

A breadth-first search seeks the binary variable which increases the partial solution objective function the least, no matter the level in the binary solution tree. A breadth-first search will often find the optimal solution as, its first feasible solution, but the working memory requirement is much greater than a depth-first search, as more nodes must be kept active at one time. The solution method of choice is usually a depth-first search, because of the larger memory requirements associated with a breadth-first search and the greater speed of a depth-first search.

The fundamental goal is to raise the lower bound, given by the partial solution in-progress, to meet the current upper bound, representing the best feasible solution found thus far. Once these two values meet, the optimum has been found.

Constraint equations link binary variables, thus reducing the available degrees of freeciom. Constraint equations also help reduce the search space, as branches of the binary tree being searched can be pruned as infeasible if they violate a constraint equation. Constraint equations can also be used to rule out previously discovered solutions through the use of integer cuts, and to incorporate heuristic relations between binaries [Raman/Grossmann 1992]. NLP formulations occur if either the constraint equations or the objective function are non-linear.

4.2 Convexity Considerations

Every optimization is a search through an n-dimensional search space (n being the number of parameters which are being optimized). A convex search space is defined as a region in which all points on a straight line between any two valid solutions are also valid solutions. Convexity is a guarantee of optimality [Kocis/Grossmann 1989]; however, while each iteration between the MILP and NLP phases may be at the global optimum for that particular phase, there is no guarantee that the end result is the global optimum, although past results have been encouraging [Quesada/Grossmann 1992]. MILP formulations are convex by definition, while NLP formulations may or may not be convex. Non-convex NLP formulations may converge to a local optimum instead of the global optimum.

While the constraint equations for bin layout are all convex, the objective function outlined in this paper is not, so by the above definition the NLP iterations may lead to local minima. Not finding the global optima for the NLP iterations could also result in not finding the global optimum for the full problem. However, since the objective function for each bin increases monotonically with the distance from the centroid of the bin to its assembly point, optimality of the NLP iterations is assured [Papalambros/Wilde 1988].

4.3 Software Implementation

The workstation optimizations were modeled with objective functions (Equation 6) based on the several ID's and performance constants noted above. GAMS, a multi-purpose interpreter for dynamic programming applications [Brooke et al. 1988], was the commercial optimization package used. Trial problems representing 40 component bins have been successfully solved with this software on a SUN 4, with a typical solution time of 5 to 10 minutes. Such practical efficiency is obtained, however, through several heuristics, which are discussed in the next sections.

5. DIFFICULTIES IN FINDING SOLUTIONS

5.1 Order of the problem

Given a set of n objects such as the bins in this layout problem, each of which has two-way links to all other members of the set, it is easy to show that the number of such links is

2

or O[n2] relations. An example could be a set of cities With the links being the distances between each city and all others in the set. Since the links between any bin and all the other bins are expressed in the form of binary variables, the possible combinations of binary variables is 2P[n2}. This order of complexity is indicative of an NP-hard problem. Thus, even small numbers of bins lead to large combinatorial problems. For example, a 5 bin case embodies $2^{1}0$, or 1024 possible combinations of binary variables. A more practical problem, such as 20 bins, embodies 2^{190}, or $1.6x10^{57}$ possible combinations.

5.2 Experimental Verification of Problem Order

Baseline cases of one to ten component bins were examined to confirm the theoretical numerical nature of the problem, and to test the feasibility of the solution method. Figure 5 shows the results of the baseline cases, all of which involved identical square bins. At 10 bins the solver indicated convergence difficulties: two cases exceeded the allowed number of MILP-NLP major iterations. It was found that both interchangeability and equivalent binary vectors were a major factor in the convergence difficulties.

The curve fit of a quadratic equation to the log of the times vs. the number of bins verifies the dependence of the solution time on the number of bins, and supports the assertion that the order of the problem is $2O[^{n2}I$. A problem is considered NP-hard if it is unlikely that the optimal solution can be found in polynomial time [Garey/Johnson 1979]. The workstation layout problem is polynomial in the log of the time and hence is NP-hard.

It was also found that the MILP optimizations required mom than 90 percent of the elapsed time with ten component bins, which is consistent with past findings [Quesada/Grossmann 1992].

5.3 Identical Bins

Assemblies may require components which are stored in identical bins. In some cases two components may be used with the same frequency; an example is a bolt and nut pair in separate drawers. If the layout is being optimized without masses, e.g. for a preliminary solution, any pair of identically-sized bins with the same number of components presents combinatorial difficulties. Since the objective function for the two bins is identical if the bin positions are switched, the solver must separately enumerate identical solutions. Figure 6 shows an example of a pair of interchangeable bins.

Two different solutions which possess identical objective functions result in unnecessary computation, as both solutions must be enumerated. During the MILP-NLP iteration process, each pair of interchangeable bins can potentially double the number of major iterations, significantly lengthening the solution time required.

Inclusion of component masses in the objective function for bins that are otherwise identical (size, number of components, and assembly point) can significantly reduce the computation effort by eliminating unnecessary solution enumeration. Since the distance ID includes a mass dependency, the solver will place the heavier bin closer to the assembly point.

5.3 Equivalent Binary Vectors

It is quite possible for two different vectors of binary values to produce the same bin layout configuration. Figure 6 also shows an example of bin i and bin j having the same geometric layout with two different binary relations between the two. This differs from the identical bins difficulty in that the bins are not switched, but two different binary variables relating the two bins result in the same objective function value. Since either binary variable returns the same objective function value, there is no difference in the two branches of the binary tree search, and the solver must proceed further into the search before ruling out one or the other solution vector. This can significantly add to the computational effort, as the strength of branch and bound searches lies in the ability to eliminate branches from consideration as early as possible.

6. HEURISTICS

As shown above, the computational difficulty for even a small numbers of bins can become intractable. Four heuristics were developed based on the difficulties noted above to reduce the problem scale to one that is solvable in a cost-effective time.

6.1 Problem Splitting

A common technique for simplifying large problems is to divide them into more tractable sub-problems. By pairing bins whose areas are similar and dividing the problem into left and right halves, with each half containing one bin from each pair, it is possible to significantly reduce computation time. Pairing bins is also observed in practice by manual assemblers who develop their own workstation layouts. Figure 7 shows an example of one 10-bin problem split into two 5-bin problems. The average case study solution time for the 10-bin problem is approximately 600 seconds, obtained from in Figure 5. Split into two 5-bin problems, the total average run time is approximately 40 seconds. The time savings grow as the number of bins increases.

6.2 Circle Model

For each pair of bins i and j a relation exists that defines, their spatial relation. Since four possible binary variables exist for every pair of bins, any binary variables that can be set before starting the optimization will enable faster convergence. A preprocessing heuristic was developed to preset as many binary variables as possible, thus removing them from consideration by the solver.

The heuristic involved circumscribing the rectangular bins (including the workspace) with circles and then solving for the optimum arrangement of circles using the same objective function as for the rectangular bins. Since circles have no orientation, ttie need for binaries was removed and the problem became a NLP, which solved very quickly. To obtain a large choice of preset binary relations muttipte configurations were created by varying the initial conditions of the heuristic randomly. A large choice is helpful because the problem is non-convex, and thus the global optimum is not guaranteed.

A total of 50 cases examining a 30 bin layout problem were analyzed. Figure 8 shows an example of a typical circle model analysis, with the rectangular bins they represent included for reference, and details how the circle model results were interpreted to preset binary relations. A sub-heuristic of the circle model enables NOT relations to be established even if two bins overlap in any direction by a small amount (in this case, ten percent of the total side lengths).

Use of the circle model heuristic will preset a minimum of half the binary relations. For example, a 20 bin problem would have 2^{190}, or approximately 1057 possible binary relations. If the minimum number of binary relations were preset with the results of the circle model, the number of binary relations that the solver may vary falls to 2^{95}, or approximately TO^{38}, resulting in a minimum reduction of the search space by approximately 29 orders of magnitude.

A rough correlation was found between the results of applying the circle model, and the actual optima found. Figure 9 shows the correlation for an example problem, discussed in Section 7. Partitions One and Two refer to the left and right sides of the workstation. For Partition One the best circle model out of thirteen runs corresponded with the best problem solution, while for Partition Two the correlation is not as strong. These thirteen cases for each partition are representative of the distribution of optima obtained from the circle model heuristic. The values have been normalized with respect to each partitions' known infeasible lower bound.

Circle model results are easily obtained, as the NLP models converged in within 30 seconds. We used the above correlation by solving a large number of circle models and then choosing the best ten percent of the circle models to preset binary relations and solve the complete MINLP formulation. Figure 10 shows the convergence of the optimization vs. the number of trials, and indicates that no significant improvements are likely after 10 MINLP solutions.

6.3 Penalties for Equivalent Binary Vectors

If two different binary relations between bins i and j yield the same configuration, the solver must continue further down the binary tree to determine which variable yields a better solution. The difficulty of two possible solutions resulting in the same objective function Is removed by establishing a weighting in the objective function which favors one binary relation over the other. Weighting of the branches is strictly a trial-and-error process, as the weight must be large enough to affect the branching process, yet small enough to not prevent the solver from attempting the other branch should the weighted branch not produce a feasible solution. A weight magnitude of one percent of the smallest possible feasible distance ID in the layout was found to be large enough to affect the branching, yet small enough to avoid searching alternate branches of the binary tree.

6.4 Spatial Logic

It is possible to represent spatial relation logic between sets of bins. For example, if bin i is found to be to the left of bin j, and bin j is found to be to the left of bin k, then bin i can be said to be to the left if bin k if the relation between i and k has not yet been found. Use of spatial logic has the advantage of trimming branches off the search tree. The disadvantage is that it increase the solution time, as more constraint equations are added. Examination of the permutations of i, j, and k shows that the number of constraint equations added to the problem is

$$O[n^3].$$

For n equal to 10, on the order of 1000 additional constraint equations are required to the represent the spatial logic heuristic. Whife theoretically sound, this heuristic was found to increase the solution time by approximately an order of magnitude for the example shown in the next section.

7. PRACTICAL WORKSTATION LAYOUT EXAMPLE

7.1 Problem Description

The Engineering Design Research Center at Carnegie Mellon University has developed a highly-portable, wearable computer called the Vu-Man (shown open to see the components in Figure 11) with which a user can manipulate blueprints that have been stored in memory. Many copies of this product had been assembled in its initial production. We measured the parameters of the assembly workstation layout used by an experienced Vu-Man assembler as a reference case, n required 30 bins of mixed size and shape.

For simplicity, this example considers a single assembly point. Although the Vu-Man is not smaller than the workspace by an order of magnitude, this was a preliminary demonstration of the method's abilities, and thus did not require a high order of accuracy. Had multiple assembly points been included, the time prediction would have been somewhat more accurate.

The assembler's layout included a product workspace that was 700 mm wide by 420 mm deep. These dimensions were subjectively determined by the assembler; however, workstation design guidelines in [McCormick/Sanders 1982] recommend reductions in workspace width and depth by 50 and 70 mm, respectively for this example. This implies that without application of human factors guidelines workers may choose a larger workspace than accepted practice indicates, leading to higher than necessary acquisition times.

Although not used by the manual assembler, the guidelines from McCormick/Sanders [1982] were included as a component of the optimization of an assembly workspace. Use of these guidelines had the net result of enabling the bins to move closer to the assembly point, reducing the handling distance difficulty. The human factors guidelines did not, however, provide any indication of bin layout.

The Vu-Man had no components that were heavy enough to warrant incorporation of ID's based on mass. Feature recognition ID's were not included because they are position-insensitive, thus an optimization based on handling distance and right-angle rotations alone is found in this case.

7.2 Solution Procedure

The component bins were assigned to one of two partitions by pairing bins of approximately equal area and then assigning one bin from each pair to each partition, as described in Section 6a. Manual assembly stations usually locate the workspace in the middle, with both sides used for component presentation. Frequency of component use was considered in assigning bins to partitions One or Two, as assemblers typically have a dominant hand and prefer to use that hand in acquisition and assembly operations. This led to significant differences in solution results between the partitions, as partition Two has most of the multiple-use components.

Over 100 circle model approximations were solved with randomly-determined initial points, and the best 15 were used to preset binaries for the MINLP. A typical circle model result for the Vu-Man is shown in Figure 8.

7.3 Quantitative Results

The numerical results for the Vu-Man assembly workstation design are shown in Table 1. The "Vu-Man Assembly Time" represents the assembly time predicted for the base case, a layout designed by an experienced engineer and actually used to assemble many units of the Vu-Man. The "Circle Model Time" represents the times predicted by the bin layout according to the circle model heuristic. The predicted times for the circle model layout were very close to the base case, since substantial spaces were left between bins. Using the binary relations extracted from the circle model heuristic, the MINLP solver was used to eliminate as much of the intervening space as possible and to vary bin rotations. Since most of the difficulty in handling occurs over relatively short distances, the predicted time is sensitive to bin locations near the assembly work area. The computation time for the MINLP analyses averaged 7 minutes on a Sun 4 UNIX workstation. The best solution layout is shown in Figure 12.

TABLE 1: INITIAL NUMERICAL RESULTS OF THE EDRC VU-MAN OPTIMIZATION

Partition	Vu-Man Acquisition Time (sec)	Circle Model Time(avg)	Optimized Time (lowest)	Reduction Factor
One	177.9	160.4	35.9	5:1
Two	423.2	414.7	87.3	5:1

The results of this example indicate a reduction of acquisition time of almost 5 to 1 over the layout used by an experienced assembler: Despite the experience of the assembler, the component bins were much farther away from the workspace than necessary, leading to high handling distance acquisition time reductions.

A more realistic appraisal of the method is obtained by comparing the results of our optimization to cases where the bins had, been tightly packed against the workspace to minimize the total workstation area. A packing optimization [Yunis/Cavalier 1990] formulation was used to minimize the area of both workstation partitions.

Table 2 shows the predicted acquisition times for tie area-based optimization and compares them wife our optimized times. While the times of the area-based optimization are much lower that the base case arrangement, the results are still significant. In this case, reductions of handling distance acquisition times of 5 percent are easily possible, with 10 percent reductions being within feasibility.

TABLE 2: EDRC VU-MAN OPTIMIZATION VS RANDOMLY PLACED AND PACKED BINS

Partition	Area-Packed Time (sec)	ID-Optimized time (lowest)	Percent Reduction
One	39.7	35.9	9.6%
Two	92.0	87.3	5.1%

7.4 Qualitative Results

No bin rotations occurred in this example despite the possibility of reducing the handling distance in some cases. The objective function allowed any bin to rotate if such a rotation could reduce the assembly time by increasing the bin density. The "no rotation" result supports the analysis in Section 3.4. Mechanical assembly systems characterized by different ID/time relations may result in solutions in which bin rotation occurs.

7.5 Optimality of Results

Given the setting of binary relations by the circle model, it is impossible to know whether or not the global optimum has been reached. Since an infeasibie lower bound on the objective function is easily obtained, an estimate of the optimality of the results is possible. Table 3 compares the best solution times with the known lower bounds, and shows that our results are less than five percent above the global optimum.

TABLE 3: OPT4MAUTY OF RESULTS (Times in Seconds)

Partition	Infeasible Lower Bound	Best Feasible Solution	Percentage Difference
One	34.4	35.9	4.4%.
Two	83.4	87.3	4.7%

8. CONCLUSION

Workstation assembly is commonplace in industry and accounts for significant production cost. Given data on component characteristics and appropriately selected bin sizes, one can now synthesize a near-optimal parts bins layout for a manual assembler. The effector model used here is valid for tasks which are "within normal reach* and for bin counts up to 40. The effects of vertical height differences, operator fatigue and "learning curve" variations need not be considered since the calculated bin layout will provide the minimum acquisition time independent of these factors. We predict the ability to reduce handling distance acquisition times for a manual assembler by approximately 5 to 10 percent, based on an application of Fitt's Law to our base case.

Since Fitt's Law has been shown to provide an accurate predictor for the acquisition times of assembly tasks [Sturges 1989a], our results should prove to be an accurate indicator of the possible reductions in assembly acquisition times in the factory.

9. FUTURE WORK

The current model considers only a horizontal planar workspace. Stacking several planes into a 2-1/2 D model should be sufficient to include all practical manual assembly applications.

A linear isotropic model of human effector behavior was used for this study. Actual effectors, either human or robotic, may feature non-linear position and direction biases which may indicate a different optimal solution. For example, Oolan [1991] mapped the impedance of the human arm in a horizontal plane. Using the gradients of such a map to compute an effective handling distance could affect the placement of the bins by adding directional preferences in conformance with a valid two dimensional extension of Frtt's Law. It also remains to include the effects Of other ID'S that relate to windage (air drag), delicate parts, use of tools, and part flexibility, should these ID's prove to be distance-dependent. The Design for Assembly theory used here e not limited to human assemblers. Given the ID/time relations for robotic effectors, the present formulation could be used to synthesize an assembly workstation based on the characteristics of any assembly machine [Sturges 1989b].

REFERENCES

Amarger, R.J., Biegler, L.T., and Grossmann, I.E. 1992: "An Automated Modelling and Reformulation System for Design Optimization", Computers and Chemical Engineering, Vo(16, No. *7, pp.* 623-636.

Boothroyd, G., and Dewhurst, P., 1981: Automation Project, University of Massachusetts, Amherst, MA.

Boothroyd, G., and Dewhurst, P., 1983: Design for Assembly: A Designer's Handbook, University of Massachusetts, Amherst, MA.

Brooke, A., Kendrick, D., and Meeraus, A., 1988: -GAMS: A User's Guide, The Scientific Press., Redwood City, CA.

Clark, T.S., and Coriett, E.N., 1984: The Ergonomics of Workspaces and Machines: A Design Manual, Taylor & Francis, London and Philadelphia.

Dolan, J., 1991: An Investigation of Postural and Voluntary Human Arm Impedance Control, Ph.D. Dissertation, Carnegie Mellon University, Pittsburgh, PA.

Drezner, Z., and Nof. S.Y. 1984, "On Optimizing Bin Picking and Insertion Plans for Assembly Robots," HE Transactions, Vol. 16, No. 3,1984.

Fitts, P.M., 1954: "The information Capacity of the Human Motor System in Controlling the Amplitude of Movemenf Journal of Experimental Psychology Vol47No.6, 1954.

Garey, M.R., and Johnson, D.S., 1979: "Computers and Intractibility: A Guide to NP-Completeness," W.H. Freeman, San Francisco.

Hunt, D.O., and Sturges, R.H., 1993: "Application of an Information-Based Design for Assembly Theory to Assembly Workstation Design," *ASME Design Automation Conference,* Albuquerque, New Mexico, September 1993.

Jayarman, R., 1985: "GALOP/2D: A Graphical System for Work-Cell Layout Evaluation," Winter Annual Meeting of the ASME, Miami Beach, FL, Nov 1985.

Khwaja, J.A., and Radhakrishna, T, 1990: "A Design for Parts Storage/Feeding in PC Board Assembly", Journal of Manufacturing Systems, Vol.9, No. 2, pp 129-138.

Kilani, M.I., and Sturges, R.H., 1992: "Detection and Evaluation of Orientation Features for CAD Part Models", Journal of Engineering Design, Vol 2, No 3, Jan 1992.

Kocis, G.R., and Grossmann, I.E. 1989: "Computational Experience with DICPOT sdh/ing M4NLP Problems in Process Systems Engineering", Computers and Chemical Engineering, Vol 13, No. 3, pp. 307-315.

Konz, S.A., 1983: Work Design: Industrial Ergonomics, Grid Publishers, Columbus, Ohio.

Kvalseth, T.O., 1983: Ergonomics of Workstation Design, Butterworths, London.

McCormick, Ernest J., and Sanders, Mark S., 1982: Human Factors in Engineering and Design, Fifth Edition, McGraw-Hill Book Company, New York, NY. pp 321-327.

Montreuil, Benoit, and Nof, Shimon Y., 1988: "Approaches for Logical vs. Physical Design of Intelligent Production Facilities", Recent Developments in Production Research, Elsevier Science Publishers B.V., Amsterdam (printed in the Netherlands), pp. 252-360.

Papalambros, P.Y., and Wilde, D.J., 1988: Principles of Optimal Design: Modeling and Computation, Cambridge University Press.

Quesada, I., and Grossmann, I.E. 1992: "An LP/NLP-Based Branch and Bound Algorithm for Convex MINLP Optimization Problems",Computers and Chemical Engineering, Vol 16, No. 10-11, pp. 937-947.

Raman, R., and Grossmann, I.E., 1992: "An Automated Modelling and Reformulation System for Design Optimization", Computers and Chemical Engineering, Vol 16, No. 7, pp. 623-636.

Sanderson, A.E., 1984: "Parts Entropy Methods for Robotic Assembly System Design.", IEEE Int. Conference on Robotics, Atlanta, GA, 1984.

Sturges, R.H., 1989a: "A Quantification of Manual Dexterity: The Design for an Assembly Calculator," Journal of Robotics and Computer Integrated Manufacturing, Vol .6, No. 3, pp 237-252.

Sturges, R.H., 1989b: "A Quantification of Machine Dexterity Applied to an Assembly Task," Int. J. Robotics Res. Vol. 9, No. 3, pp 49-62.

Sturges, R.H., and Hunt, D.O., 1994: "Reduction of Acquisition Time Through New Design for Assembly Heuristics," Submitted to the *ASME Design Automation Conference,* September 1994.

Sturges, R.H., and Wright, P.K., 1989: "A Quantification of Dexterity," Journal of Robotics and Computer Integrated Manufacturing, Vol 6, No 3, pp 237-252.

Sturges, R.H., Dorman, J.G., and Brecker, J.N., 1986: Design for Producability. Westinghouse Electric Corp., Productivity and Quality Control Center

Viswanathan, J. and Grossmann, I.E., 1990: "A Combined Penalty Function and Outer-Approximation Method for MINLP Optimization", Computers and Chemical Engineering, Vol 14, No. 7, pp. 76$-782.

Whitney, D.E., Nevins, J.L. 1989: Concurrent Design of Products and Processes, McGraw Hill Publishing Co., New York, NY.

Wong, J. H.-W., and SWrges, R.H., 1991: "Design for Assembly Factors for Large and Heavy Parts" ASME Design Automation Conference, Phoenix, AZ Sept. ,1992

Yunis, N.A., and Cavalier, T.M., 1990: "On Locating Part Bins in a Constrained Laybut Area for an Automated Assembly Process," *Computers and Industrual Engineering,* Vol, 18, No2., pp. 111-118.

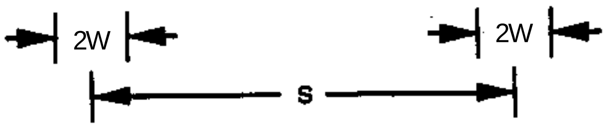


FIGURE 1: SCHEMATIC OF FITTS TAPPING TASK: SUBJECT MAKES A JOT BETWEEN EACH SET OF BARS ALTERNATELY AT MAXIMUM SPEED AND MINIMUM ERROR.

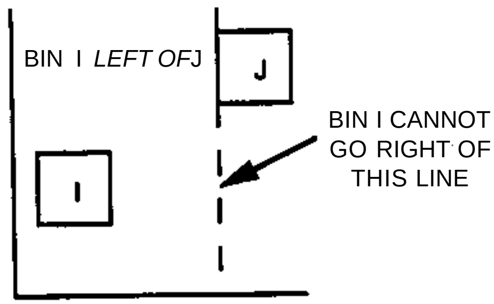


FIGURE 2: EXAMPLE OF A BINARY RELATION

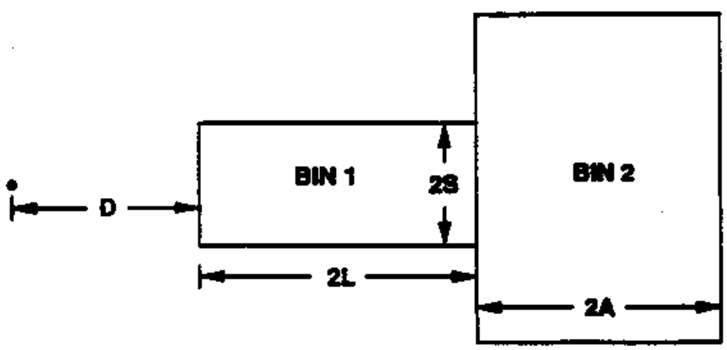


Figure 3: Bin **Arrangement** for Study of **Rotations va HandHng Distance**

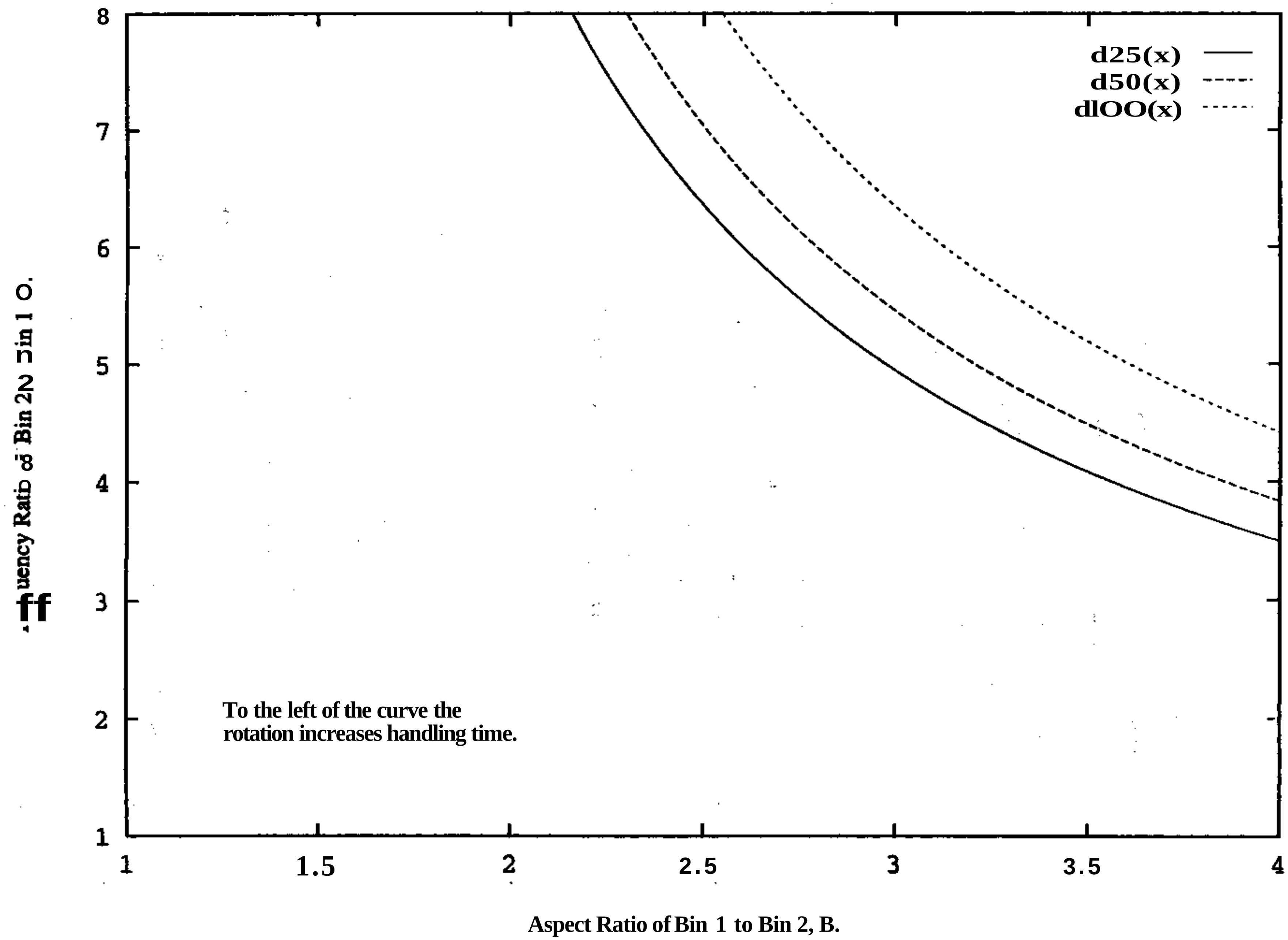


FIGURE 4: CURVES SHOWING IF A ROTATION OF BIN 1 IS ADVANTAGEOUS

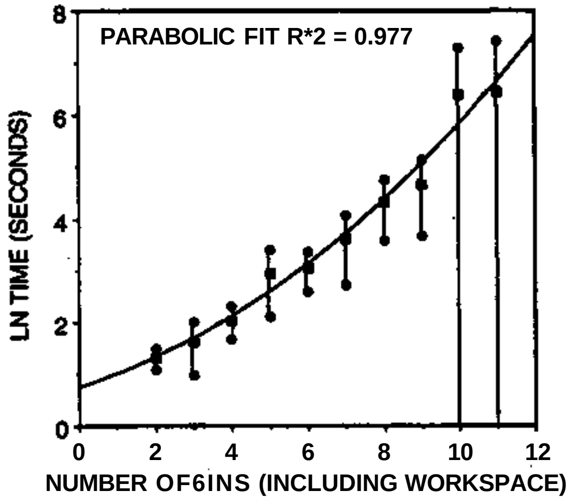


FIGURE 5: LN OF TIME VS NUMBER OF BINS (WITH ERROR BARS)

1. BINS I AND J INTERCHANGEABLE
2. BIN I CAN BE EITHER BELOW OR LEFT OF BWJ WITHOUT CHANGING OBJECTIVE^ VALUE

FIGURE 6: EXAMPLES OF TWO TYPES OF COMPUTATIONAL DIFFICULTIES

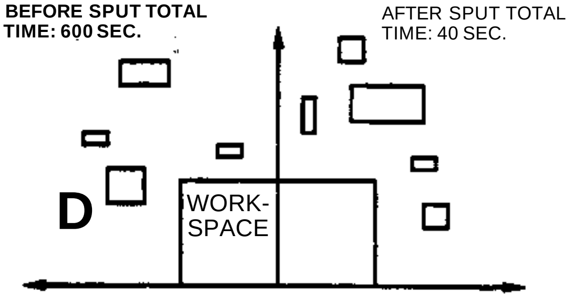


FIGURE 7: ONE 10-BIN PROBLEM SPLIT INTO TWO 5-BIN PROBLEMS

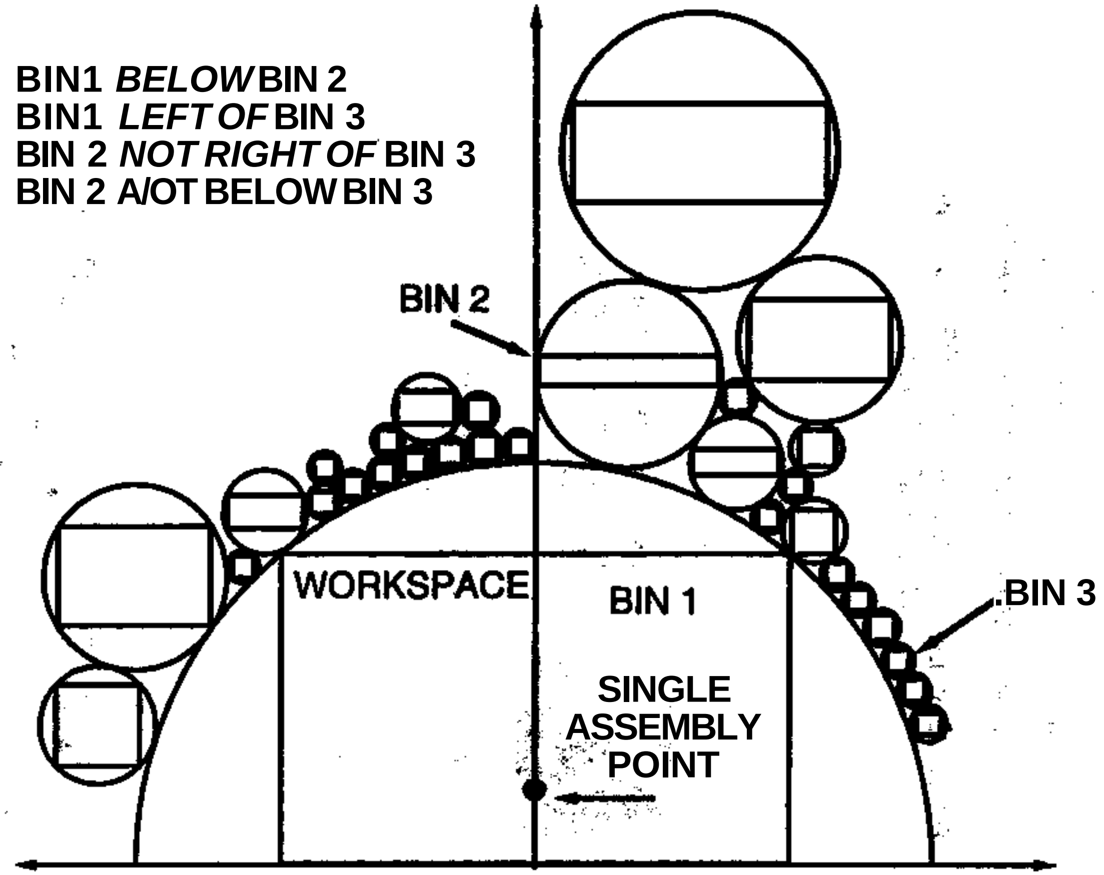


FIGURE 8: EXAMPLE CIRCLE MODEL RESULTS (VU-MAN) AND SAMPLE BINARY EXTRACTION

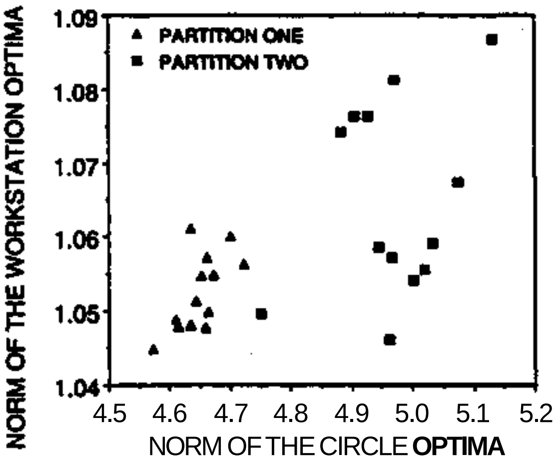


FIGURE 9: CORRELATION OF CIRCLE MODEL RESULTS TO OPTIMIZED WORKSTATIONS

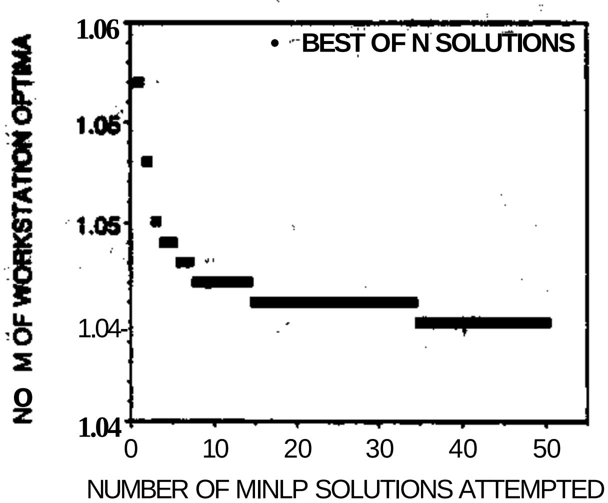
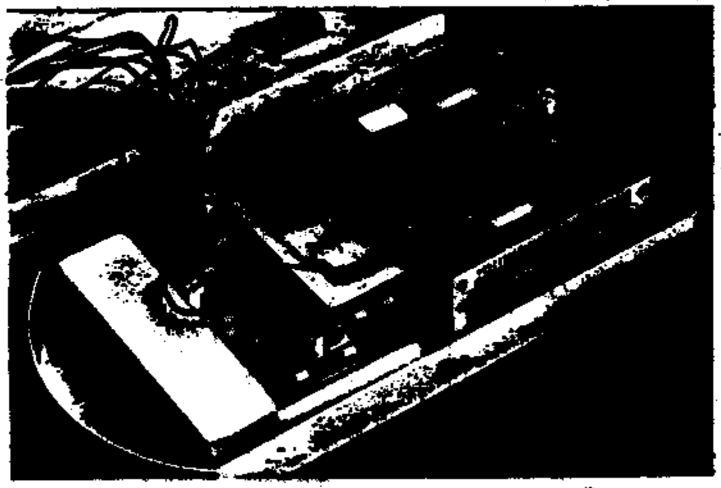


FIGURE 10: CONVERGENCE TOWARDS BEST ANSWER

(Photo by Bill Redic)



*I«R5URE 11: EDRC VU-MAN INTERIOR

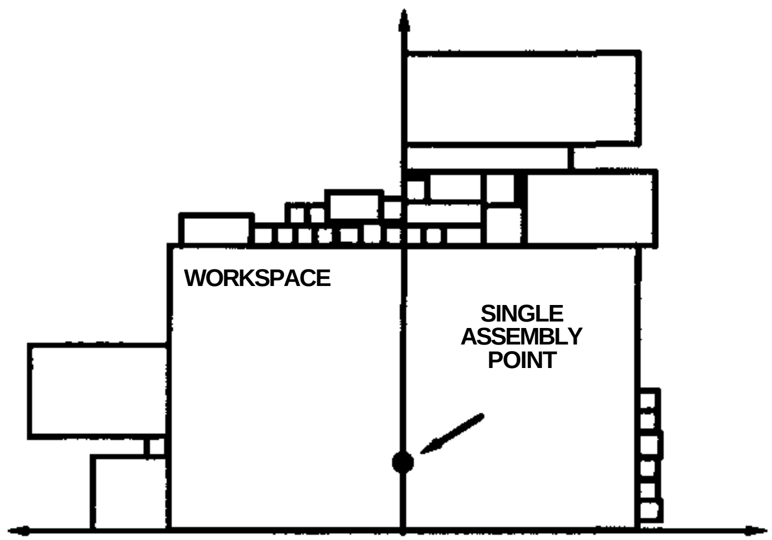


FIGURE 12: BEST SOLUTION OBTAINED