Proceedings of the 1998 Winter Simulation Conference
D.J. Medeiros, E.F. Watson, J.S. Carson and M.S. Manivannan, eds.

HEURISTIC OPTIMIZATION USING COMPUTER SIMULATION: A STUDY OF STAFFING LEVELS IN A PHARMACEUTICAL MANUFACTURING LABORATORY

Tom Brady, Ph.D.

Purdue University North Central
Westville, IN 46391, U.S.A.

Bernard McGarvey, Ph.D.

Eli Lilly & Company
Clinton Laboratories
Clinton, IN 47842-0099, U.S.A.

ABSTRACT

This paper summarizes the results of the integration of four heuristic optimization techniques and a computer simulation model. In this project, our goal is to optimize the operating performance of a pharmaceutical manufacturing laboratory in which a small set of operators service a larger set of testing machines. The combinatorial complexity of the operator/test machine set along with the inherent non-linearity, variability, and stochastic nature of pharmaceutical manufacturing make this problem very difficult, if not impossible to solve using traditional Operations Research tools. Thus, a very detailed computer simulation model of the laboratory was constructed to answer questions related to the capacity and cycle time capability of the laboratory. While this simulation model provided detail about the dynamics of the operation and functioned as a convenient "what-if" evaluator of proposed operational changes, it did little in terms of telling us what assignment of operators to test machines is best or what the potential range of improvement could be.

The key concept in this paper is the use of a computer simulation model to generate Output Responses for optimization techniques of Simulated Annealing, Tabu Search, Genetic Programming, and a novel frequency-based heuristic approach. Overall, we have shown that the integration of these optimization techniques into the computer simulation experimentation process can provide significant performance improvements over the results obtained by stand alone computer simulation modeling practice. More importantly, these improvements can be obtained in significantly short periods of time with less human intuition and intervention required.

1 SUMMARY OF RESULTS

Table 1 presents the results obtained. By integrating heuristic optimization techniques with the computer simulation model, we show that the **performance of the laboratory can be improved by nearly 16 percent** in the extreme case, with improvements in the range of 10 percent very likely.

Other conclusions reached by this analysis include 1) There are many "good" solutions to this problem that look very different in structure but produce the same magnitude of result, 2) The WIP terms in the Output Response dominate the other terms and have significant variation amongst solutions, and 3) If the laboratory is overstaffed, i.e. each Test covered by seven Operators, the variability of the results is significantly decreased.

Table 1: Summary of Results

Scenario	Output Response	Percent Improvement
Baseline Case	33.668	-
Estimate of Optimal Solution	27.984	16.9
Frequency-Based Heuristic	28.359	15.8
Simulated Annealing	30.552	9.3
Genetic Algorithm	30.873	8.3
Tabu Search	31.01	7.7

2 DESCRIPTION OF LABORATORY OPERATION

An assay, or demand on the laboratory enters according to three different input strategies. Two distinct Product types exist and are further categorized as eight different forms. Each assay generates a requirement for a number of Tests to be performed. There can be a maximum of 24 Tests per assay, all of which can be performed in parallel. The number of Tests required by an assay is a function of the Product type. The assay is not considered complete until all Tests have been performed.

There is a queue at each Test where a batching strategy is employed. Once a predetermined number of assays are queued up, the Test begins. Each Test consists of a number of Tasks, generally between 10 and 15. These Tasks must be carried out serially. Each Task may have a resource requirement of an Operator and/or a miscellaneous piece of equipment. Operators are constrained as to which Tests they can perform due to factors such as training, years of experience, etc. In addition to performing Tests, Operators are also responsible for performing miscellaneous tasks such as safety meetings, glassware cleaning, etc. These tasks are queued and done prior to normal tasks. The time required to perform a Task is dependent on the batch size. While most Tasks are required for each Test, some need only be performed on a frequency basis. For example, equipment calibration might be performed once a week rather than each time the Test is performed. The time to perform the Task includes common cause variability. All of the Tests have verification Task requirements which must be performed by a qualified Operator who has not performed any prior Tasks for the particular Test. This is a significant constraint when dealing with reduced Operator levels and assignments. Each Task has a failure probability. If a Task fails, a given number of previous Tasks must be repeated.

3 PROBLEM DEFINITION

The formal problem statement is: ***Determine the values for a matrix that assigns eight(8) Operators to forty four(44) Tests in a manner that minimizes an Output Response composed of terms that represent Work In Process, Operator Efficiency, and Operator Balance.***

The Output Response is to be minimized and consists of the summation of four appropriately scaled terms. Term *one* is the average daily Work In Process level for products of Type A. Term *two* is the average daily Work In Process level for products of Type B. Term *three* is the sum of the idle time percent for all eight operators. This term represents a measure of the efficiency of the Operators. Term *four* represents the density of the assignment matrix and is the sum of all non zero entries. This term represents a measure of the balance of assignments.

Although the computer simulation model of the laboratory produces an extraordinary amount of useful, detailed information, the Output Response used to drive the optimization procedure to a solution is simply an aggregation of the four terms deemed most important by decision makers. The aggregation of these terms into a single value presents the likely possibility that many distinctly different solutions may produce an Output Response of roughly the same magnitude. Thus, the solution procedure is developed with this perspective in mind.

The Decision Variables for this problem consist of the individual cells of the assignment matrix of Tests and Operators. Thus we have 44 times 8 or 352 decision variables. Each decision variable can take on one of three values; 0,1, or 2. Zero(0) indicates the operator cannot perform the test. One(1) indicates that the operator should perform the test. Two(2) indicates that the operator could perform the test. These possibilities can be thought of as a priority scheme which is used by the computer simulation model as the primary decision logic in selecting operators to perform tests when multiple choices exist.

4 SOLUTION METHODOLOGY

The procedure used to solve this problem involves the integration of heuristic optimization techniques and a computer simulation model in a hierarchical fashion, with the heuristic using the simulation model to simply 'calculate' the value of the Output Response for a given instance of decision variables. Selecting a given decision variable instance, referred to as the "Pick Solution" step is a key element of the optimization heuristic that clearly has an effect on both the quality and efficiency of the solution obtained. The "Pick Solution" process used to select a decision variable instance for each iteration is a single step procedure, i.e., only one decision variable is altered per iteration of the heuristic. Selection of the decision variable instance includes the random selection of a Test (1-44), the random selection of the number of Operators(1-8) allowed to perform the Test, and the random selection of individual Operators(1-8) and priority values(1-2).

To estimate a Bound on the solution, we simply enable any Operator to perform any Test at the highest priority level. Using appropriate common random number schemes and replication methods, we obtain an average Output Response value of 27.984 with a standard deviation of .1711.

In addition to estimating the bound on the solution, it is essential to develop a Baseline estimate for the laboratory operation in its current state so that we can make comparisons as to the relative goodness of any alternative solution we obtain. The values of the decision variable set that are used in the "current" operation are considered as the baseline case. Using the same replication scheme as in the Bound case, we obtain an average Output Response of 33.668 with a standard deviation of .4999.

5 SIMULATED ANNEALING

In this section, we describe the use of a Simulated Annealing algorithm as the heuristic. Appropriate tuning of the algorithm for this problem was performed, followed by a 5000 iteration experiment. The "answer" is the minimum value of 29.918 occurring in iteration 3737. Proper replication of the decision variable instances from iteration 3737 produced an average Response of 30.552 with a standard deviation of .317.

Figure 1 shows a histogram of the Output Response values from the Simulated Annealing algorithm in which the distribution is skewed toward the right of the mean. Comparing this distribution to the Baseline solution of 33.668, we conclude that the Baseline solution is clearly quite good. Figure 2 shows the value of the Output Response for the first 2500 iterations of this experiment. There are many sharp spikes and relatively few regions in which the Output Response is steady for any length of time. This behavior of the Output Response indicates that

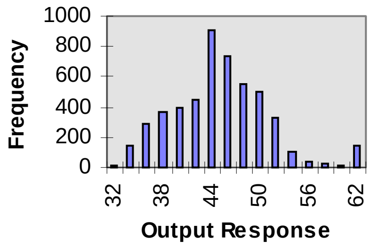

Figure 1: Histogram of Output Response Values

the true solution for this problem probably consists of finding the optimal Operator coverage for a few very sensitive Tests. In other words, the solution values for the non-critical Tests are probably inconsequential and can be set to literally any values. This supposition is based upon the observance of very sharp spikes at both good and bad solutions. This behavior, coupled with the single step "Pick a Solution" candidate selection process indicates that the Output Response is very sensitive to changes in the individual Test assignments. The critical question then becomes one of trying to find which Tests are the most sensitive. Figure 3 shows the average Output Response delta values by Test. This was calculated by measuring the change in the Output Response from iteration to iteration and categorizing the results by the Test that was altered for the particular iteration. Although no correlation can be attributed to the Test by itself, it can be seen that on average altering the solution status of certain Tests seems to improve the Output Response while others had the opposite effect. Of the 44 Tests, 20 produced adverse effects on the Output Response, 22 produced positive effects, and 2 produced negligible effects.

6 GENETIC ALGORITHM SOLUTION

In this section, we describe the use of a genetic algorithm as the heuristic. The structure of the problem, with a homogenous set of decision variables and range of values lends itself quite nicely to a genetic algorithm solution.

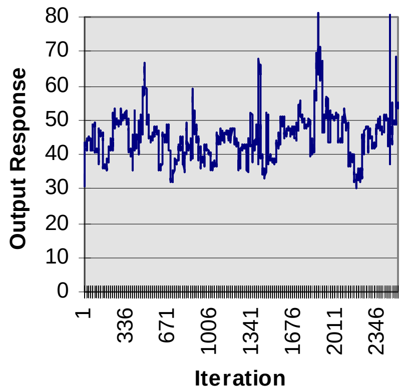

Figure 2: Plot of Output Response Values, Iterations 1-2500

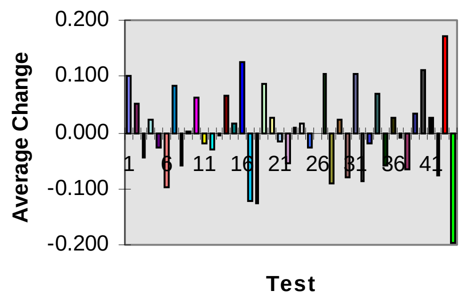

Figure 3: Average Change in Output Response by Test

Representation schemes are defined as character strings that identify decision variables and their corresponding range of values. In this case, the representation consists of the 44 by 8 matrix, with each position in the string representing the assignments of a single operator to the 44 Tests. Thus, each position in the string represents a column of the matrix. Using this representation, the Length of the character string, denoted by L, is eight.

In this problem, we use a subset of decision variable instances or solutions from the optimization procedure as the initial random population. Instead of evaluating this population as a whole and using the fitness of each instance as a reducing mechanism, we simply set a threshold value and save any solution that meets or exceeds this criteria. This will greatly reduce the size of the initial population and also filter out solutions which are deemed as undesirable.

Based upon prior experimentation, we select a value of 36.00 as a threshold for saving solutions that will form the initial population. Using this value, we end up with an initial sample population size of 95 from a 1000 iteration experiment. The average Output Response value for this sample population is 35.01. Compared to the mean value of the 1000 trials, we can see that this population of solutions improves the average fitness of the population from 43.00 to 35.00. Using this sample population, we can now apply a crossover mechanism that will allow us to create new population members with characteristics of the latter population.

Table 2 illustrates the process of generating offspring using the solution sets from iterations 776 and 992 and applying the crossover operation using a randomly selected crossover point of 3. For example, Offspring 2 combines the column assignments from Operators 1, 2, and 3 of Solution 992 with the column assignments of Operators 4, 5, 6, 7, and 8 from Solution 776.

Table 2: Crossover Operation

Solution	Fragment Operators	Remainder Operators
776	1,2,3	4,5,6,7,8
992	1,2,3	4,5,6,7,8
Offspring 1	1,2,3(776)	4,5,6,7,8(992)
Offspring 2	1,2,3(992)	4,5,6,7,8(776)

After performing the crossover operation, we obtain two offspring solution sets. Using these as input to the computer simulation model produced the results shown in Table 3.

Table 3: Genetic Algorithm Solutions

Offspring	Output Response	Adjusted Output Response
1	50.544	35.641
2	148.600	30.873

The solutions generated by the offspring are much worse than those obtained in any previous experiment. Upon closer inspection of the detailed output from the simulation model, we find the reason for inferior solutions lies in constraint violations created from the crossover operations. Significant waiting times were present at select Tests, resulting in severe performance degradation in the Output Response. In the crossover operation, several Tests were covered by only one Operator, leading to the increased waiting time situations. As an operational constraint, each Test must be covered by at least 2 Operators. Manually adjusting the offspring by randomly increasing the number of Operators allowed to perform the constraining Tests produced the results shown in the Adjusted Output Response column with the replicated trial of Offspring 2 producing an Output Response significantly better than the Baseline case.

7 TABU SEARCH

In this section, we describe the results obtained through use of a Tabu Search algorithm as the heuristic. A Tabu period of 25 iterations was selected after considerable experimentation. Using appropriate common random number schemes and replication methods, we obtain an average Output Response value of 31.01.

8 A FREQUENCY BASED HEURISTIC

Taking advantage of the intuition gained by studying the results obtained using standard heuristic optimization techniques, a new heuristic was developed. The basis of this heuristic is that certain configurations of decision variables tend to cluster into “good” and “not good” solutions. This frequency-based heuristic involves the following steps:

1) Use a standard improving search or heuristic search-based methodology to generate a sample population of solutions. Using this sample population of solutions, determine a distribution of the Output Response values.

2) Using the distribution of the Output Response values, select a threshold value that partitions the solutions into ‘good’ and ‘not good’ categories.

3) Using the ‘good’ and ‘not good’ criteria, rerun the heuristic and generate a frequency value for each decision variable by this category scheme. Thus, for each decision variable, you will have the percent of time that the variable was in a solution set that produced a ‘good’ solution, the percent of time that the variable was in a solution set that produced a ‘not good’ solution, and the percent of time that the variable was not in the solution set.

4) Using the frequency values obtained in Step 4, subtract the ‘not good’ value from the ‘good’ value for each decision variable. This value represents the tendency for this decision variable to be in ‘good’ solutions. Thus, we should strive to select the minimum set of “high positive value” decision variables. In the context of this problem, we should hope for a decision variable matrix with a large density of positive numbers. From this, we could then pick a minimal cardinal set of assignment pairings.

5) Use a standard assignment type optimization formulation or greedy algorithm approach to pick a minimal covering set of non zero decision variables using the matrix of Step 4. The value in the cells can be thought of as the cost coefficient of the objective function. It is also necessary in this step to be cognizant of row and column constraints.

8.1 Partitioning the Solution Set

To effectively partition the solution set of Output Response values into "good" and "not good" solutions for Step 2 of the frequency-based heuristic, it is necessary to have some information about the distribution of Output Response values. To obtain this information, a trial of the optimization procedure consisting of 1000 iterations was conducted, producing an average Output response of 43.835 with a standard deviation of .2341. Figure 4 shows a frequency distribution of the Output Response values.

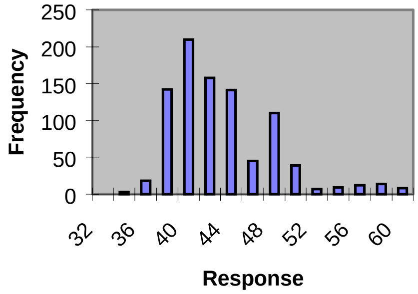

Figure 4: Histogram of Trial Output Response Values

In comparing this distribution to the Bound and Baseline scenarios, the difference between the mean Output Response values is slightly more than 10.0, or roughly 30 percent. Figure 4 shows what appear to be two fairly distinct "bad" zones of Output Response values, while the rest of the values appear to be distributed fairly evenly throughout the 38 to 44 range. It should also be noted that an extremely long tail exists to the right or "bad" side of the mean while little or no tail exists to the left or "good" side of the mean.

Upon closer inspection, the primary difference between the Baseline scenario and this trial lies in selection of the initial solution. The Baseline scenario uses the current assignment of operators to tests used in the lab; one that has been developed and improved over the years while this trial scenario generates an initial solution randomly. Also, 1000 iterations is hardly sufficient to allow convergence of the algorithm. From this, we can conclude that the assignment of operators to tests in the Baseline scenario is done very well, yet there is still plenty of opportunity for improvement when compared to the Bound scenario.

Using this information, a more effective experiment for determining the partitioning value would be to run the heuristic using the Baseline scenario as the initial solution. Using this scenario, a trial of 1000 iterations yielded an Average Output Response of 43.062 with a standard deviation of .1507. Figure 5 shows a frequency distribution of the Output Response values.

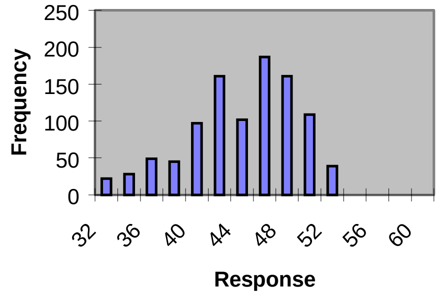

Figure 5: Histogram of Output Response Values Using Baseline Initial Solution

Figure 5 appears to be a reverse image of Figure 4 in which the tail of the distribution appears to the left of the mean rather than the right. While the mean value is very similar for both cases, the range and standard deviation of the values is much smaller when the Baseline scenario is used as the initial solution. Also, the best value obtained by the heuristic was nearly 10 percent better.

The results of this experiment show that the initial solution for the optimization heuristic is a very important factor in determining the quality of the solution obtained when a fixed number of iterations are available. While it is expected that the heuristic could converge to similar if not better solutions using a random initial solution, the use of a known, good initial solution can greatly improve the efficiency of the algorithm.

Continuing with the analysis in selecting the partitioning value, we need to make sure that a sufficient number of solutions will be deemed "good", i.e., providing a basis for determining what the critical variables and parameters are for good solutions. At the same time, we need to make sure this partitioning value is not too large so as to mask the discernment of "good" and "not good" solutions. Using Figure 5 as a guide, we choose the partitioning value to be 38.00. This should provide us with approximately 17 % of the solutions being considered "good". Using this value with a 5000 iteration experiment, we produce a matrix in which each cell defines the tendency for the particular operator and test assignment to occur in a "good" solution, with "good" solution defined as one in which the Output Response is below 38.00.

Using this matrix, we use a greedy procedure to cover each Test with a given number of Operators. We start by attempting to find a minimal assignment solution in which each Test may be covered by a maximum of three Operators and proceed up to a solution with seven Operators covering each Test. Eight Operator coverage is equivalent to the Bound case. Table 4 presents the results of these solutions.

Table 4: Greedy Algorithm Results

Number of Operators	Output Response
3	47.881
4	39.569
5	37.878
6	38.464
7	28.346

The solution was fairly indifferent for the 4, 5, and 6 Operator cases, but showed large sensitivity for the 3 and 7 Operator cases. While it may not be practical or cost effective to have each operator sufficiently capable of performing seven tests, it is selected nonetheless as optimal for the method. Selective pruning of this solution could be done prior to implementation. Replication of the seven operator case produced an average Output Response of 28.359 with a standard deviation of .280.

9 CONCLUSIONS

This report has demonstrated that the integration of optimization techniques and computer simulation models can result in double digit performance improvements for critical metrics. Suggestions for further study and analysis include:

- Refine the “Pick Solution” logic of the heuristic to include multiple Test selection.
- Look closely at the optimal solution(s) for feasibility/affordability.
- Validate the solution(s) on the real system in a controlled, experimental fashion.
- Incorporate the Test Priority scheme into the decision variables.

AUTHOR BIOGRAPHIES

TOM BRADY is an assistant professor of Industrial Engineering Technology at Purdue University North Central. He received a Ph.D. in industrial engineering from Purdue University. He is a member of INFORMS, IIE and The Society for Computer Simulation.

BERNARD McGARVEY is a Senior Engineering Consultant at Eli Lilly & Company, Clinton, Indiana. He received a Ph.D. in chemical engineering from University College Dublin. He is a member of the American Society of Quality and the Systems Dynamics Society. His areas of interest include manufacturing methods, stochastic modeling and safety.