Journal of Productivity Analysis, 11, 149–168 (1998)

Using Simulation and Data Envelopment Analysis to Compare Assembly Line Balancing Solutions

PATRICK R. MCMULLEN patmc@maine.maine.edu
University of Maine, Maine Business School, Orono, ME 04469-5723

GREGORY V. FRAZIER frazier@uta.edu
The University of Texas at Arlington, Information Systems and Management Sciences Dept., College of Business Administration, Arlington, TX 76019-0437

Abstract

This paper presents a technique for comparing the results of different assembly line balancing strategies by using Data Envelopment Analysis (DEA). Initially, several heuristics—which can be thought of as assembly line balancing strategies—were used to solve seven line-balancing problems. The resulting line balance solutions provided two pieces of information that were of particular interest: the number of workers needed and the amount of equipment needed. These two items were considered inputs for DEA. The different line balance solutions were then used as layouts for simulated production runs. From the simulation experiments, several output performance measures were obtained which were of particular interest and were used as outputs for DEA. The analysis shows that DEA is effective in suggesting which line balancing heuristics are most promising.

Keywords: Assembly line balancing, Data envelopment analysis, Heuristics, Paralleling

1. Introduction

Assembly line balancing has been a topic of research for several decades now. Generally, line balancing is the process of organizing the tasks required to produce a product into subgroups and assigning each subgroup to its own work cell. Most commonly, previous line balancing approaches have exploited one of two objectives. One objective (generally referred to as the Type I problem) is to minimize the amount of workers required on the assembly line, given a specified cycle time. The other objective (generally referred to as the Type II problem) is to minimize the cycle time, given a specified number of workers. With either type, *cycle time* is the average amount of time that elapses between two consecutive units being completed on the assembly line.

Another issue in line balancing problems is the number of workers allowed to perform a particular task. Most frequently, the problem addressed allows each task to be performed by only one worker. The alternative strategy of allowing multiple workers to each perform the same set of tasks is referred to as paralleling, or the use of parallel work-stations in a work cell. Paralleling is necessary in the case where the duration of any individual task is greater than the desired cycle time. With high-volume production, it is not unusual for at least one task to have a task time that exceeds the desired cycle time. Relatively little research has been published on line balancing with parallel workers.

The number of different products that are assembled on the line can further categorize line-balancing problems. The large majority of research has addressed the single product line-balancing problem. When multiple products have been addressed, it has usually been assumed that they would be produced in batches, with the line being re-balanced for each product type. The line balancing problem for multiple products and no re-balancing allowed is significantly more difficult to solve. Due to the widespread adoption of Just-in-Time (JIT) production, assembly and production lines with multiple products sequenced in mixed-model fashion have become much more common.

Yet another factor that distinguishes different line balancing problems is the uncertainty of task times. The large majority of research addresses line balancing with deterministic task times. The more difficult problem is when some or all of the task times are stochastic. The research presented here addresses the Type I assembly line balancing problem with parallel processing, stochastic task times, and both single and mixed-model production.

In much of the literature addressing the Type I line balancing problem, the desirability of a solution rests solely on the number of workers required to attain the specified cycle time. While using as few workers as possible is a major concern of the line balancing problem, it should not be the only performance measure that is examined, especially with today's time-based competition. Evaluating the production performance of a layout attained by assembly line balancing is not a common practice in related research–although the current availability of simulation software makes this type of evaluation practical. Analyzing layouts in this way allows many performance measures to be considered, such as average work-in-process (WIP) inventory level, average amount of time a unit spends in-process (flow-time), number of units produced (throughput), average unit labor cost, system utilization, and the actual cycle times attained.

In a JIT setting, one of the most important performance measures is the ability of the production line to adhere to the schedule, especially when task times are somewhat uncertain. The average cycle time actually achieved, compared with the desired cycle time, is an effective way to measure a production line's ability to adhere to a schedule. Through using simulation to analyze the production performance of different line balancing solutions, this cycle time performance of the line balances can be compared and is used as a DEA output measure in this research. Another important attribute of an assembly line layout is its ability to move units through each individual work station in a timely fashion. This attribute is analogous to on-time completion and is used as another DEA output measure.

The cost of implementing a particular line balance solution is another important performance characteristic. Two parts of this cost are most important, labor cost and equipment cost. Since different line balances will require varying numbers of workers and pieces of equipment (when paralleling is allowed), this information must be evaluated along with output performance. Therefore, the number of workers and amount of equipment required will be used as input performance measures in comparing line balancing approaches.

Data Envelopment Analysis (DEA) is a useful tool for comparing several alternatives when multiple performance measures are important, particularly when both input and output performance measures exist (Boussafiane, et al. (1991), Charnes, et al. (1978), Doyle

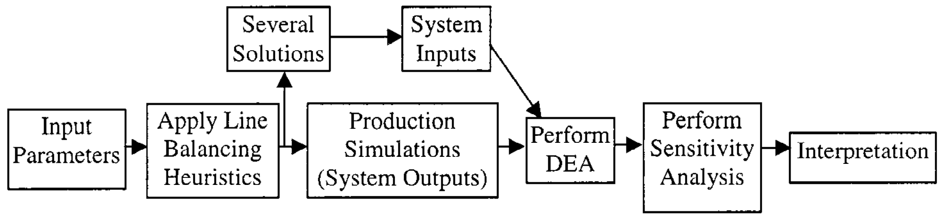

Figure 1. Overview of research.

and Green (1991)). In the problem addressed here, DEA is used to compare different line balancing heuristics using two output performance measures (cycle time performance and percentage of on-time completions within cells) and two input performance measures (number of workers and amount of equipment).

To overview this research, assembly line balancing problems are solved by using twenty-three alternative strategies, or heuristics. These twenty-three heuristics can be thought of decision-making units (DMU's) for DEA. Once a problem is solved with each of these twenty-three heuristics, the number of workers (crew size) and equipment requirement are recorded as inputs and the resulting layouts are used for simulated production runs. From these simulations, cycle time performance and percentage of on-time completions through each work center are recorded as the outputs for the layout associated with each line balancing rule. To measure cycle time performance, a ratio of desired cycle time to average cycle time actually achieved in the simulations is computed. This cycle time ratio is referred to as CTR. To measure percentage of on-time completions (POT), the number of parts produced within the desired cycle time in each work cell is divided by the total number of parts produced in each cell and an average is computed for all cells. Given these inputs and outputs, DEA is then used to determine the relative efficiency of the twenty-three different heuristics.

The primary purpose of this paper is to demonstrate how simulation and Data Envelopment Analysis can be used together to evaluate different line balancing approaches, particularly in a JIT environment. The following sections present the research methodology, discuss the experimental results, and offer conclusions.

2. Methodology

Figure 1 provides an overview of the research methodology used. The different line balancing heuristics, simulation design, and details of the Data Envelopment Analysis are each discussed in the following subsections.

2.1. Line Balancing Heuristics

Table 1 provides a brief description of each of the twenty-three heuristics used in this research. The heuristic used to solve several of the line balancing problems for this research is a modification of Gaither's Incremental Utilization Heuristic (1996). This heuristic permits paralleling of workers within work centers. Permitting paralleling to occur enables the manager to place multiple workers in work centers, and also enables the manager to establish a cycle time that is less than the duration of the longest task (Pinto (1975), Pinto, Dannenbring, and Khumawala (1975, 1981)). One efficiency-oriented feature of this heuristic is that it will not permit the addition of a task and its respective workers to a particular work center unless the utilization of that work center increases as a result of adding the task (and an additional worker if necessary). The Modified Incremental Utilization Heuristic heuristics are labeled 1–8 and 11–17 in Table 1. These different heuristics basically function as task selection rules—strategies for selecting tasks to enter the work centers. For detailed descriptions of these heuristics, refer to McMullen (1995), McMullen and Frazier (1997), and Baybars (1986).

Six of the heuristics used in this research are Simulated-Annealing (SA) based. Simulated Annealing is a heuristic technique used to solve combinatorial optimization problems given some type of objective function. The first SA based heuristic (label 18) uses a SA search objective that seeks to minimize the design cost (the sum of labor and equipment requirement). Heuristic 19 seeks to minimize the smoothness index across work centers—the amount of inconsistency in terms of work load across the work centers (Moodie and Young (1965)). Heuristic 20 seeks to minimize the probability of late completion of activities across work centers. Heuristics 21 through 23 seek to minimize different combinations of design cost and lateness cost. For Heuristic 21, design cost and lateness are each weighted so that they make an equal contribution to a composite objective function. For Heuristic 22, lateness is weighted so that it makes three times the contribution to the objective function as design cost. Heuristic 23 is weighted so that design cost makes three times the contribution as lateness. For more information on Simulated Annealing and these particular heuristics, refer to McMullen and Frazier (1998).

Heuristics 9 and 10 utilize neither the Incremental Utilization Heuristic nor the Simulated Annealing strategy. Heuristic 9 places all tasks into a single "mega" work center and will always result in all of the equipment being replicated many times. For example, in the Appendix it can be seen that for the 11-task problem Heuristic 9 required 9 workers, each performing all tasks in parallel, in order to achieve the desired cycle time. This resulted in 99 pieces of equipment required. Heuristic 10, on the other hand, places each task into its own individual work center. Additional workers are added to a work center with Heuristic 10 only if a task time is greater than the cycle time. For example, as can be seen in the Appendix, Heuristic 10 required 14 workers for the 11-task problem. The three additional workers were necessary because in three work cells the task time exceeded the desired cycle time, and each additional worker works in parallel with the original worker in a cell. Heuristic 10, therefore, will always result in the same number of workers and machines.

Since the primary purpose of this paper is to demonstrate how simulation and DEA

Table 1. Description of heuristics used.

Label	Heuristic	Incremental Utilization
1	Select task providing maximum incremental utilization	Used
2	Select task randomly (Arcus (1966))	Used
3	Select task with longest duration	Used
4	Select task with shortest duration	Used
5	Select task providing minimum incremental utilization	Used
6	Select task providing minimum probability for lateness within work center	
7	Select task with best composite of 5 and 6	Used
8	Select task according to lexicographic attributes	Used
9	Single Mega work-center	Not Used
10	Individual work center for each task	Not Used
11	Select task having fewest followers	Used
12	Select task having fewest immediate followers	Used
13	Select task which is first to become available	Used
14	Select task which is last to become available	Used
15	Select task having most followers	Used
16	Select task having most immediate followers	Used
17	Select task with highest Ranked Positional Weight (Helgeson and Birnie, 1961)	Used
18	Simulated Annealing: Minimize Design Cost	Not Used
19	Simulated Annealing: Minimize Smoothness Index	Not Used
20	Simulated Annealing: Minimize Overall System Lateness	Not Used
21	Simulated Annealing: Minimize Composite Function 1	Not Used
22	Simulated Annealing: Minimize Composite Function 2	Not Used
23	Simulated Annealing: Minimize Composite Function 3	Not Used

can be used together to evaluate different line balancing strategies, details of the Modified Incremental Utilization (MIU) heuristic are not presented here but can be found in McMullen (1995) and McMullen and Frazier (1997). The general approach of using simulation in conjunction with DEA is the focus of this paper and can be applied using *any* set of line balancing heuristics.

Some general assumptions of assembly line balancing problems are listed below.

- A task will not be assigned to a work center until all of its predecessors have been assigned.
- A task is assigned to exactly one work center.
- All workers on the assembly line possess the same level of skill.
- The durations of all tasks are independent of each other.
- Changeover times between different products are negligible.

Table 2. Description of seven different line balancing problems.

Tasks	Different Products	Product-Mix Weights	Simulated Buildup Time (minutes)
11	1	$w_1 = 1$	500
21	1	$w_1 = 1$	1,000
25	1	$w_1 = 1$	1,000
29	2	$w_1 = 2$, $w_2 = 1$	3,000
40	3	$w_1 = 3$, $w_2 = 2$, $w_3 = 1$	3,000
45	2	$w_1 = 2$, $w_2 = 1$	3,000
74	4	$w_1 = 4$, $w_2 = 2$, $w_3 = 1$, $w_4 = 1$	3,000

2.2. Simulation Design

As previously mentioned, the twenty-three different heuristics are the decision making units (DMU's) for the Data Envelopment Analysis. Seven example line balancing problems were used for this study, each consisting of a number of products, a number of tasks, a precedence network, a desired cycle time, expected task durations, and a distribution for each task duration. Each of the seven problems was solved with each of the twenty-three heuristics, resulting in twenty-three layouts per problem. For each layout, the number of workers and amount of equipment required were recorded. Table 2 provides information about each of the seven problems used.

For each of the seven problems, the desired cycle time was specified to be 10 minutes between completed units. The durations for the tasks were created via a random number generator. For each task, there was a 75% probability the task duration would be uniformly distributed between 2 and 10 minutes, and a 25% probability the task duration would be uniformly distributed between 10 and 15 minutes. The precedence diagram for the: 11-task problem is from Mariotti (1970); the 21-task problem is from Tonge (1965); the 29-task problem is from Buxey (1974); for the 45-task problem is from Kilbridge and Wester (1961). The 25-, 40-, and 74-task problems were arbitrarily generated for this research.

It should be noted that four of the seven problems described in Table 2 have multiple products made simultaneously during the production run. These multiple-product problems assume mixed-model sequencing, which reflects a common situation in the design of assembly lines in JIT systems. The simulated production runs for these problems implement mixed-model sequencing. The product-mix weights are simply a representation of the relative number of products to be made for each of these mixed-model problem types. For example, the 29-task problem will have two units of product one made for each unit of product two. For more information on mixed-model systems and their sequencing, refer to Ding and Cheng (1993).

The twenty-three solutions for each of the seven problems were used as layouts for simulated production runs. SLAMSYSTEM v4.6 with FORTRAN v5.1 user-written inserts was used on a Pentium-75 personal computer to model these 161 different layouts. For each layout, the simulation was run until steady-state conditions were attained (refer to Table 2). Statistics were then reset, and each simulation run was continued long enough so that reasonable estimates of means of all outputs could be obtained. For each of the 161

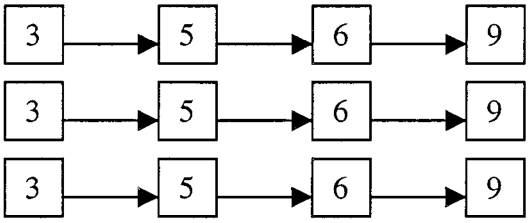

Figure 2. Example of paralleling and equipment required in work center 2.

different layouts, 25 simulation run replications were made. This resulted in a database of 4,025 observations.

2.3. Data Envelopment Analysis

As mentioned earlier, the inputs for the DEA were number of workers required and the amount of equipment required. The number of workers was provided directly from the line balance layouts. For simplicity, it was assumed that each task required a different piece of equipment (or machine). The amount of equipment required in each work center then was equal to the number of workers required (the amount of paralleling) multiplied by the number of tasks assigned to that work center. Figure 2 provides a graphic example. Each worker in work center 2 would perform tasks 3, 5, 6, and 9 in parallel. This results in a total of 12 pieces of equipment (3 workers * 4 tasks/worker).

A tradeoff exists between the number of workers required and the amount of equipment required. Typically, fewer workers are required as more paralleling is used. The extreme case would have every worker performing every task in parallel, in one large work center (Heuristic 9). In this way, the desired cycle time can be met with the minimum number of workers. However, this solution would require the maximum amount of equipment. This scenario illustrates the need to consider both the number of employees and the amount of equipment, as well as the actual cycle time achieved, when evaluating line balancing approaches.

Data Envelopment Analysis involves solving a set of mathematical programming models to determine a “technical efficiency” of a decision making unit when one or more outputs and inputs are involved (Charnes, Cooper, Lewin and Seiford (1994)).

The DEA envelopment surface used for this research is variable returns to scale (VRS). The reason for this is because the authors could not assume that a constant returns to scale (CRS) is appropriate. In other words, if the inputs of crew and equipment were doubled, it cannot be assumed that the outputs of percentage of on-time completions (POT) and cycle time ratio (CTR) would also double for the associated production layout. As a result, a VRS was deemed most appropriate for this research (Banker, Charnes and Cooper (1984)).

For this research, both input-oriented models and output-oriented models were used for each of the seven problems. The reason for this is so that inefficient DMUs could be further examined to determine what could be done to make them efficient. In other words, inefficient units are classified as inefficient due to “too much input,” “not enough output,” or some combination of both.

The DEA models used in this research each have two outputs (CTR and POT) and two inputs (crew requirement and equipment requirement). Because each model has both two inputs and two outputs, it is quite possible to find various DMUs to be relatively efficient when various weights take on negligible (or zero) values. This possibility does not reflect a realistic situation (Wong and Beasley (1990)). For example, a certain DMU could be found efficient when the weight for the equipment requirement receives a zero weight. This of course implies that the only resource needed is labor, and not equipment. Another example might find that a DMU is efficient when the weight pertaining to POT is zero. This implies that the output measure of on-time completion is ignored for this particular DMU. To prevent these occurrences and to reflect as realistic a situation as possible, weight restrictions are incorporated into these DEA models.

Consider the following variables

$$u_{1i} = \text{POT weight for DMU } i \tag{1}$$

$$u_{2i} = \text{CTR weight for DMU } i \tag{2}$$

$$v_{1i} = \text{Crew requirement for DMU } i \tag{3}$$

$$v_{2i} = \text{Equipment requirement for DMU } i \tag{4}$$

The constant α is chosen by the modeler to represent the minimum contribution each output is permitted to make to the virtual output value. Since there are only two outputs used, two important things are noteworthy. First, $1 - \alpha$ is then the maximum contribution each output is permitted to make to the virtual output value. Secondly, it is also important to note that specifying the upper limit of one of the virtual outputs implies specification of the lower limit of the other virtual output, and vice versa. Mathematically, this is expressed as follows:

$$\alpha \leq \frac{POT_i u_{1i}}{POT_i u_{1i} + CTR_i u_{2i}} \leq 1 - \alpha \tag{5}$$

$$\alpha \leq \frac{CTR_i u_{2i}}{POT_i u_{1i} + CTR_i u_{2i}} \leq 1 - \alpha \tag{6}$$

For this research, a value of .35 was used for α. This value was arbitrarily chosen to allow one measure to reflect at most nearly twice the importance of the other measure.

Imposing weight restrictions on the outputs of POT and CTR is fairly straightforward due to the fact that their measures are between 0 and 1 (with slight, rare exceptions), which is simple to address. The same scenario does not apply when considering the necessary weight restrictions for the inputs. The inputs of crew size (CREW) and equipment requirement (EQUIP) can take on a variety of values, which makes reasonable application of weight restrictions more difficult than for the outputs. As a result, the following weight restriction was imposed on the inputs:

$$.25 \leq \frac{v_{1i}}{v_{2i}} \leq 4.0 \tag{7}$$

In other words, neither input is permitted to take on a weight more than four times that of the other input. This four-to-one ratio also was arbitrarily selected for this research.

The restrictions on the weights then define "assurance regions" (Thompson et al. (1990)) which prevent "undesirable" heuristics from being classified as DEA-efficient.

3. Analysis and Results

While the input measures of crew requirement and equipment requirement are easy to accept as important performance measures, the authors would like to comment on the selection of cycle time ratio (CTR) and on-time completions (POT) as performance measures. CTR provides a macro-type measure of the schedule-meeting performance of the overall production line, when stochastic task times are considered. It was also found to be highly correlated ($r = -.966$ and $r = -.965$, respectively) with WIP inventory levels and average part flow time, two other common production line performance measures. POT provides a more micro-type measure of within-cell, schedule-meeting performance. Since the correlation between CTR and POT was not deemed to be unreasonably high ($r = .617$), both output performance measures were included in the analyses.

Separate input-oriented and output-oriented DEA's were performed for each of the seven problems to determine the efficiency of each of the twenty-three heuristics. Table 3 provides a summary of the DEA results. There are two lines to interpret in the table for each heuristic—the first line is the DEA-efficiency for the input-oriented model, while the second (italicized) line is the DEA-efficiency for the output-oriented model. A detailed explanation of the input-oriented models is presented first, followed by a detailed explanation of the output-oriented models.

For each problem associated with the input-oriented models, the efficiency is provided. An efficiency of 1.0000 means that the heuristic of interest is efficient—the DMU exists on the efficiency frontier. For example, Heuristic 1 for the 40-task problem yields a DEA-efficient result. When the efficiency is less than 1.0000, the heuristic is considered inefficient—the DMU does not exist on the efficiency frontier. If weight restrictions were not used, or if they were non-binding, the efficiency value would describe the proportional reduction of inputs necessary to make the heuristic of interest DEA-efficient. For example, Heuristic 8 for the 45-task problem is an example of a heuristic that is not DEA efficient. If none of the weight restrictions were binding, then the inputs for this heuristic must be reduced to 85.28% of their current levels for it to become DEA-efficient. However, in the large majority of cases in this research a weight restriction was binding. Therefore, since Allen et al. (1997) concluded that under such binding weight restrictions this common interpretation of efficiency values may be invalid, caution must be exercised in interpreting efficiency values for the inefficient DMUs in Table 3.

The last column of the table is reserved for a subjective remark regarding the general performance of each heuristic. These remarks are basically self-explanatory with the possible exception of "excessive," which simply means there is a clear pattern that the heuristic of interest requires an excessive amount of inputs—crew and equipment. These remarks are subjective of course, but the authors believe they are reasonable.

Table 3 also presents the DEA efficiency summary for the 23 heuristics for each of the seven problems when the output-oriented models are examined. These efficiency measures are italicized. An efficiency measure of 1.0000 means that a certain heuristic was found to be DEA-efficient (Heuristic 1 for 40-task problem, for example). Any efficiency measure greater than 1.0000 describes a situation when the heuristic of interest is found to be DEA-inefficient. If weight restrictions were not used, or if they were non-binding, the efficiency value would reflect the enhancement of outputs necessary for this heuristic to become DEA-efficient. Heuristic 8 for the 45-task problem, for example, is found to be inefficient. If none of the weight restrictions were binding, and the outputs were enhanced to 8.16% above their current levels, the heuristic would then become DEA-efficient. However, as noted earlier, a weight restriction was binding in the large majority of cases, so caution must be exercised in interpreting efficiency values for the inefficient DMUs.

The last column of the table is reserved for a subjective remark regarding the general output-oriented performance of each heuristic. As is the case for the input-oriented models, all but one remark is basically self-explanatory—"deficient." In this context, "deficient" simply reflects a situation where a heuristic performs poorly with respect to the output measures of POT and CTR. The subjectivity of these assessments should also be considered.

3.1. Sensitivity Analysis

The outputs of POT and CTR are attained via a simulation model, which means they are stochastic. This means that the decision-maker must consider the uncertainty associated with these outputs when determining which heuristic to consider for adoption. A sensitivity analysis was conducted to determine which of the DEA-efficient heuristics are the most robust—least sensitive to unfavorable changes in the DEA models. The more robust a DEA-efficient heuristic is, the more confidence the decision-maker can have in its successful implementation. The first part of this sensitivity analysis was to determine which of the DEA-efficient heuristics were found to be "extreme efficient." A DEA-efficient heuristic (heuristic j) was found to be extreme efficient if and only if:

$$h_j(w) > h_k(w) \quad \text{for all } k \neq j$$

where:

$$h_j(w) = \frac{POT_j u_1 + CTR_j u_2}{CREW_j v_1 EQUIP_j v_2}, \; j = 1, \ldots n \text{ heuristics.}$$

where u_1, u_2, v_1, and v_2 are the multipliers associated with the heuristic being tested for extreme-efficiency (Thompson et al. (1994), Thrall (1996)).

The second part of the sensitivity analysis was to determine which of the DEA-efficient heuristics were robust enough to withstand unfavorable changes in the DEA models. In this context, an unfavorable change means a decrease of x percent in output for all DEA-efficient heuristics and a simultaneous increase of x percent in output for all DEA-inefficient heuristics. In other words, a change generally means that DEA-efficient heuristics are made to appear less desirable while DEA-inefficient heuristics are made to appear more favorable.

Table 3. DEA efficiency summary individual heuristics.

Heuristic	11 Task	21 Task	25 Task	29 Task	40 Task	45 Task	74 Task	Remark
1	.8858	.9794	.9990	.9277	1.0000	1.0000	1.0000	Good
	1.1867	*1.0191*	*1.0517*	*1.1367*	*1.0000*	*1.0000*	*1.0000*	*Fair*
2	1.0000	.9304	.8821	.9655	.9931	.8401	.9083	Weak
	1.0000	*1.0934*	*1.2094*	*1.1005*	*1.0162*	*1.1221*	*1.1172*	*Deficient*
3	.9887	.9616	.9537	.9415	.9651	.8729	.9603	Weak
	1.0847	*1.0393*	*1.0562*	*1.1008*	*1.0490*	*1.1316*	*1.1379*	*Deficient*
4	1.0000	.9811	.8340	.8107	.8661	.9215	.8077	Fair
	1.0000	*1.0517*	*1.0702*	*1.1887*	*1.1292*	*1.0562*	*1.1571*	*Deficient*
5	.8488	.7800	.8444	.6782	.7058	.7326	.6926	Weak
	1.0534	*1.1290*	*1.0673*	*1.2905*	*1.2025*	*1.1589*	*1.2192*	*Deficient*
6	.8488	.7800	.8444	.7910	.8706	.6903	.6444	Weak
	1.0534	*1.1290*	*1.0673*	*1.1111*	*1.1047*	*1.1979*	*1.1819*	*Deficient*
7	.8488	.7800	.8444	.8317	.8706	.7790	.6493	Excessive
	1.0534	*1.1290*	*1.0673*	*1.1390*	*1.1047*	*1.1096*	*1.1188*	*Deficient*
8	.9828	.9728	.9410	.8237	.9074	.8528	.9235	Fair
	1.0160	*1.0231*	*1.0796*	*1.1129*	*1.1132*	*1.0816*	*1.1350*	*Deficient*
9	.4732	1.0000	.2815	.2463	.2006	.1826	.1754	Excessive
	1.0278	*1.0000*	*1.0426*	*1.0773*	*1.0519*	*1.0371*	*1.4343*	*Fair*
10	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	Strong
	1.0000	*1.0000*	*1.0000*	*1.0000*	*1.0000*	*1.0000*	*1.0000*	*Strong*
11	.7105	.9881	.9531	.8470	.9557	.8380	1.0000	Fair
	1.0198	*1.0095*	*1.0868*	*1.1528*	*1.0327*	*1.1553*	*1.0000*	*Fair*
12	.8642	.9417	.9289	.8457	.9361	.9124	.8918	Fair
	1.0198	*1.0571*	*1.0845*	*1.1872*	*1.0768*	*1.0881*	*1.0839*	*Fair*
13	1.0000	.9242	.9868	.9425	.9032	.9639	.8918	Fair
	1.0000	*1.0773*	*1.0255*	*1.0762*	*1.0760*	*1.0339*	*1.0839*	*Fair*
14	.8442	.9634	1.0000	.9580	.9488	.8827	.9484	Fair
	1.0197	*1.0359*	*1.0000*	*1.1879*	*1.0390*	*1.1583*	*1.0757*	*Fair*
15	1.0000	1.0000	.7193	.7976	.9611	.9651	.9170	Fair
	1.0000	*1.0000*	*1.0481*	*1.2739*	*1.0287*	*1.0240*	*1.1156*	*Fair*
16	.8642	.9548	1.0000	1.0000	.9163	.9013	1.0000	Good
	1.0197	*1.0320*	*1.0000*	*1.0000*	*1.0648*	*1.0823*	*1.0000*	*Good*
17	.9784	1.0000	.8850	1.0000	.9315	.9342	.9170	Fair
	1.1100	*1.0000*	*1.0524*	*1.0000*	*1.0522*	*1.0635*	*1.1156*	*Fair*
18	.9998	1.0000	.9785	1.0000	1.0000	.9518	1.0000	Strong
	1.0045	*1.0000*	*1.1004*	*1.0000*	*1.0000*	*1.0760*	*1.0000*	*Good*
19	.9995	.6543	.6635	.7239	.3655	.6494	.4427	Excessive
	1.0086	*1.0657*	*1.1263*	*1.1459*	*1.2133*	*1.1604*	*1.1946*	*Deficient*
20	1.0000	1.0000	1.0000	1.0000	.7323	.9287	1.0000	Strong
	1.0000	*1.0000*	*1.0000*	*1.0000*	*1.0053*	*1.0133*	*1.0000*	*Good*
21	.9730	.6255	.9890	.9632	.9820	1.0000	.9185	Fair
	1.0005	*1.0221*	*1.0015*	*1.0062*	*1.0135*	*1.0000*	*1.0255*	*Good*
22	.9730	.6249	.9670	.9632	1.0000	.9327	.9364	Fair
	1.0005	*1.0228*	*1.0039*	*1.0062*	*1.0000*	*1.0173*	*1.0204*	*Good*
23	1.0000	.7974	1.0000	.9632	.9764	.8940	.8583	Fair
	1.0000	*1.0067*	*1.0000*	*1.0062*	*1.0134*	*1.0354*	*1.0451*	*Good*

Table 4. Sensitivity analysis for line balancing problems.

Problem	Efficiency Set	Extreme Efficient Set	Efficient Set (1%)	Efficient Set (2.5%)	Efficient Set (5%)
11 Tasks	2, 4, 10, 13, 15, 20, 23	13, 15, 20, 23	4, 10, 13, 15, 18, 21, 22	4, 10, 13, 15, 18, 21, 22	4, 10, 13, 15, 18, 21, 22
21 Tasks	9, 10, 15, 17, 18, 20	9, 10, 15, 17, 18, 20	9, 10, 11, 15, 17, 18, 20, 23	1, 10, 11, 16, 18, 23	1, 10, 11, 16, 18, 23
25 Task	10, 14, 16, 20, 23	10, 14, 16, 20, 23	10, 14, 16, 20, 21	10, 13, 14, 16, 21	1, 10, 13, 21
29 Task	10, 16, 17, 18, 20	10, 16, 17, 18, 20	10, 16, 17, 18, 20, 21, 22, 23	10, 16, 17, 18, 20, 21, 22, 23	2, 10, 13, 16, 17, 18, 20, 21, 22, 23
40 Task	1, 10, 18, 22	1, 10, 18, 22	1, 2, 10, 18, 20, 21, 22, 23	1, 2, 10, 18, 20, 21, 23	1, 2, 10, 11, 15, 18, 20, 21, 23
45 Tasks	1, 10, 21	1, 10, 21	1, 10, 20, 21, 22	1, 10, 13, 15, 20, 22	1, 10, 13, 15, 20, 22
74 Tasks	1, 10, 11, 16, 18, 20	1, 18, 20	1, 10, 11, 16, 18, 20, 21, 22	1, 10, 11, 16, 18, 20, 21, 22	1, 10, 11, 14, 16, 18, 21, 22

For this research, three different levels of change were explored: 1%, 2.5%, and 5% (roughly 1σ, 3σ, and 6σ, respectively). It is important to note that this analysis does not contain any input changes because the inputs of crew and equipment needed are not stochastic. Table 4 shows the results of the sensitivity analyses in terms of extreme efficient points and robustness to change in output (Thompson et al, 1994).

Table 4 shows that almost all heuristics found to be DEA-efficient are also found to be extreme-efficient (corner points). The exceptions to this are heuristics 2, 4, and 10 for the 11-task problem and heuristics 10, 11, and 16 for the 74-task problem—these heuristics are initially found to be DEA-efficient, but not extreme efficient. Table 4 also shows that with "changes" to the original set of DEA-efficient heuristics, a certain amount of robustness exists. For the most part, heuristics that are part of the original set of DEA-efficient heuristics remain DEA-efficient despite having their outputs reduced while DEA-inefficient heuristics simultaneously have their outputs enhanced. As the degree of this "change" increases, more heuristics that were initially DEA-inefficient join the set of DEA-efficient heuristics, but fewer heuristics initially found to be DEA-efficient leave the set of DEA-efficient heuristics. This issue is elaborated upon further in the next section. Due to the general presence of robustness, finding DEA solutions which seek the Strong Complementary Slackness Condition (which generally results in robust solutions, (Thrall (1996), Gonzalez-Lima et al. (1996), and Thompson et al. (1996))) is not explored.

4. Discussion of Results

As previously mentioned, this research is dedicated to helping the decision-maker determine the heuristics best suited for their needs. The following paragraphs present arguments on which of the 23 heuristics are the best and worst performing in terms of DEA.

From inspection of the efficiency measures in Table 3 and the sensitivity analysis information in Table 4, two things become immediately clear. The first is that Heuristic 10 is the strongest performer in terms of DEA efficiency—it is found to be efficient for all seven problems, and demonstrates robustness to the degree that it remains DEA-efficient even with 5% changes for all problems. The second is that Heuristic 9 is a very poor performer. It was found to be efficient only for the 21-task problem, and lacks robustness for this problem as it departs the set of DEA-efficient heuristics with a 2.5% change of outputs. The reason for such differing results can be explained.

Generally, both Heuristics 9 and 10 show reasonably good performance in terms of the output measures, with the exception of Heuristic 9's performance for the 74-task problem (see Appendix). The reason for the difference in efficiency between these two heuristics lies in their input requirements (see Table 3 and Appendix). Each heuristic sacrifices greatly in performance on one input resource in order to minimize use of the other resource. For each problem, Heuristic 9 requires the fewest workers but the most equipment of all the heuristics, and Heuristic 10 requires the least amount of equipment but the most workers. In the experiment, Heuristic 10 generally required roughly 50% more workers than most heuristics and roughly 50% less equipment. Heuristic 9 generally required slightly fewer workers, but excessively more equipment (several hundred percent more).

Because of equal crew and equipment requirements with Heuristic 10, the weight restrictions placed on the inputs will most likely never be binding, giving the heuristic relatively more freedom to have its input weights find values resulting in robust DEA-efficient solutions. For heuristic 9, the equipment requirement far exceeds the crew requirement. This will most likely cause the input weight restrictions to be binding, which limits the DEA model's ability to find weights resulting in DEA-efficient solutions. In short, Heuristic 9 is "input-excessive," while Heuristic 10 is not, despite both heuristics resulting in generally favorable levels of output.

The next group of heuristics addressed includes those that use simple task selection rules for assembly line balancing (2, 8, and 11–16). These heuristics work by placing tasks into work centers according to the rules described in Table 1. The attribute which makes these heuristics similar to each other is that these rules never consider the duration of the task at hand, but instead explore precedence relationships (e.g., most followers [Heuristic 15]) or randomly select tasks for membership in work centers (Heuristic 2). It is worth noting that these heuristics have been in the assembly line balancing literature since the 1960's. With the exception of Heuristic 16, which selects tasks based upon most immediate followers, these heuristics do not perform particularly well. These heuristics typically exhibit "fair" performance with regard to both the input and output-oriented DEA models. This is not terribly surprising due to the fact that since these heuristics either select tasks randomly or explore precedence relationships, no specific line balancing strategy exists. Rule 16 is the exception that generally demonstrates a good performance. It is found to be DEA-efficient for the 25, 29, and 74-task problems and shows robustness up to 2.5% changes for 25-task problem and, 5% for the 29-task problem.

The next group of heuristics discussed has the common attribute of exploring the duration of the task at hand to make task selection decisions (Heuristics 1, 3–7, and 17). In general, these heuristics were designed to require as few workers as possible. Furthermore, the production performance when these layouts are used is quite poor as exhibited by the number of heuristics in this category noted as “output-deficient” in Table 3. One possible exception is Heuristic 1. Table 3 suggests that Heuristic 1 requires a reasonable amount of inputs, and also shows a fair performance in terms of the production output measures. Also, Table 4 shows that Heuristic 1’s efficiency is robust, displaying robustness through 5% changes for these problems.

The last group of heuristics discussed includes the Simulated Annealing based techniques (Heuristics 18–23). These heuristics attained good achievement of their search objective functions for the line balancing problem. Because of this, specific goal-oriented results can be obtained, which is basically what occurred. Heuristic 18, which focuses only on design cost, provided favorable results in terms of the input-oriented DEA model, which shouldn’t be surprising, since the objective function is dedicated to minimizing the total cost of crewing and equipment. It also provided solutions that generated somewhat favorable performance with regard to the output-oriented DEA models. Heuristic 19 performed poorly. It was determined to be “input-excessive” as well as being “output-deficient. This heuristic utilizes an objective function dedicated to making the workload across all work centers equal. This strategy showed no benefits. Heuristic 20 utilizes an objective function dedicated to minimizing the probability of units being completed late. This strategy was designed to generate layouts that would perform well with respect to the output-oriented DEA models, which it did. The surprising result of this heuristic, however, was that it also showed favorable performance in terms of the input-oriented DEA models. The last three heuristics use composite functions with two components: design cost and probability of lateness (Heuristic 21 gives equal weight to each component, 22 give more weight to lateness, while 23 gives more weight to design cost). These are multiobjective heuristics. Generally, these three heuristics provided layouts that perform reasonably well on the DEA output-oriented models, but only fair on the DEA input-oriented models.

4.1. Summary of Discussion

Table 5 provides the decision-maker with some guidelines in selecting from these line balancing heuristics, based on this research. For each heuristic, two pieces of information are given. The first is the number of times a heuristic was found to be DEA efficient in this experiment (seven is the maximum, since seven problems were examined). The second is a brief listing of situations when a manager might consider using this heuristic.

From inspection of Tables 3 and 4, it becomes clear that several of these line balancing heuristics need not be considered for implementation due to their excessive resource consumption and/or their poor performance in terms of the output measures. These tables also provide information on which heuristics appear most promising. The authors’ short list of preferred heuristics includes Heuristics 1, 10, 16, 18, and 20. Exactly which one to

Table 5. Performance summary of each heuristic.

Heuristic	Times Efficient	When to Consider
1	3	When there are no delivery pressures/when resources are expensive
2	1	Never
3	0	Never
4	1	When resources are inexpensive/when there are no delivery pressures
5	0	Never
6	0	Never
7	0	Never
8	0	When there are no delivery pressures
9	1	Never
10	7	When there are: delivery pressures, expensive equip., inexpensive labor
11	1	When resources are inexpensive/when there are no delivery pressures
12	0	When resources are inexpensive/when there are no delivery pressures
13	1	When resources are inexpensive/when there are no delivery pressures
14	1	When resources are inexpensive/when there are no delivery pressures
15	2	When resources are inexpensive/when there are no delivery pressures
16	3	When resources are expensive/when there are delivery pressures
17	2	When resources are expensive/when there are no delivery pressures
18	4	When resources are expensive/when there are no delivery pressures
19	0	Never
20	5	When resources are expensive/when there are delivery pressures
21	1	When resources are inexpensive/when there are no delivery pressures
22	1	When resources are inexpensive/when there are no delivery pressures
23	2	When resources are inexpensive/when there are no delivery pressures

Note: in this table, *resources* implies both labor and equipment.

select would depend on which performance objectives are most important in a particular environment.

Heuristic 10 provided the best overall performance in terms of the DEA models. As previously noted, however, while this heuristic provides the minimum equipment requirement, it also requires more workers than the other heuristics (see Appendix). Because of these labor demands, the manager should be aware that when the specific assembly line application has an expensive hourly labor component, adoption of Heuristic 10 should probably be discouraged. Instead, one of the other heuristics showing promise should be considered—perhaps Heuristic 20, since it specifically addresses (and achieves) layouts which excel at requiring reasonable amounts of labor and equipment while simultaneously providing the layouts with the ability to show a strong performance in terms of output measures.

5. Conclusions

A technique has been presented that uses simulation and Data Envelopment Analysis to assist management in deciding which assembly line balancing heuristics would best suit their needs. For the heuristics that were not found to be DEA-efficient, the DEA models inform the decision-maker as to why a certain heuristic is not efficient and what changes could be made to either the inputs or outputs so that the heuristic could become DEA-efficient. For the heuristics that were found to be DEA-efficient, the sensitivity analyses provided some insight as to how "robust" these heuristics are to unfavorable changes. Generally, a reasonable amount of robustness was found for the better heuristics.

This methodology used two output performance measures, reflecting a production layout's ability to achieve the desired cycle time and its ability to complete parts on time within each cell. Labor and equipment requirements were considered as well. Weight restrictions were imposed on these inputs and outputs so that all of the performance measures were given reasonable consideration. For a particular line balancing application, the decision-maker would want to weight the inputs and outputs to best reflect the current situation—e.g., if equipment cost is quite high relative to labor cost, the weights should be allocated accordingly.

Line balancing is only one type of layout, or production design procedure. Other procedures are also concerned with resource requirements as well as production performance. These other procedures, such as cellular manufacturing design and laying out job shops, may be good candidates for evaluation of multiple solution approaches using DEA.

Acknowledgments

The authors are grateful for the constructive comments made by the Associate Editor and the anonymous referees. These comments have helped with the improvement of this paper.

Appendix. Data Used for DEA Models

	11-Task Problem				**21-Task Problem**			
Heuristic	POT (11)	CTR (11)	Crew (11)	Equip (11)	POT (21)	CTR (21)	Crew (21)	Equip (21)
1	0.5944	0.9752	10	31	0.7758	0.9371	17	43
2	0.9547	1.0012	10	24	0.6996	0.9331	16	52
3	0.8067	0.9627	9	28	0.7897	0.9329	17	46
4	0.7566	0.9449	9	27	0.6943	0.9291	16	46
5	0.8878	0.9808	10	35	0.7475	0.9461	16	71
6	0.8878	0.9808	10	35	0.7475	0.9461	16	71
7	0.8878	0.9808	10	35	0.7475	0.9461	16	71
8	0.9206	0.9999	10	25	0.8100	0.9535	16	51
9	0.9193	1.0001	9	99	1.0000	1.0001	16	336
10	0.9908	1.0000	14	14	0.9686	0.9995	26	26
11	0.9413	1.0000	14	34	0.8453	0.9717	18	45
12	0.9413	1.0000	10	34	0.8183	0.9427	18	47
13	0.8797	0.9885	10	23	0.8066	0.9332	17	52
14	0.9413	1.0002	10	34	0.7907	0.9369	17	46
15	0.8797	0.9885	10	23	0.9077	0.9845	18	48
16	0.9413	1.0002	10	34	0.9271	0.9635	17	59
17	0.7399	0.9708	10	25	0.9457	0.9985	18	53
18	0.9444	1.0000	10	24	0.7662	0.9372	17	40
19	0.9331	0.9999	10	24	0.8749	0.9700	16	115
20	0.9966	1.0000	11	28	0.9970	1.0005	19	108
21	0.9953	1.0000	11	28	0.9344	1.0001	18	127
22	0.9953	1.0000	11	28	0.9344	1.0001	18	127
23	0.9942	1.0012	11	25	0.9700	0.9997	19	91

	25-Task Problem				**29-Task Problem**			
Heuristic	POT (25)	CTR (25)	Crew (25)	Equip (25)	POT (29)	CTR (29)	Crew (29)	Equip (29)
1	0.6948	0.9307	20	64	0.6881	0.7952	25	78
2	0.5804	0.9349	20	83	0.6608	0.772	24	74
3	0.7946	0.9404	20	71	0.7121	0.8004	25	77
4	0.8585	0.9629	20	98	0.6183	0.8535	24	108
5	0.8356	0.9773	20	96	0.7997	0.5435	25	145
6	0.8356	0.9773	20	96	0.7740	0.8876	25	123
7	0.8356	0.9773	20	96	0.6716	0.8795	25	102
8	0.7757	0.9332	20	73	0.7684	0.8795	25	113
9	0.8757	1.0009	19	475	0.6910	0.9979	23	667
10	0.9388	0.9995	32	32	0.8853	0.9468	38	38
11	0.7069	0.9491	20	71	0.7230	0.8457	25	100
12	0.7257	0.9664	20	75	0.7219	0.8108	25	96
13	0.8181	0.9324	20	67	0.7779	0.8777	26	83
14	0.889	0.9531	21	65	0.6601	0.5834	24	73
15	0.8844	0.9884	20	134	0.6884	0.7571	27	98
16	0.7928	0.9453	20	64	0.7395	0.7876	26	64
17	0.8630	0.9768	20	90	0.6773	0.8643	25	66
18	0.6491	0.9300	19	71	0.6893	0.6484	25	62
19	0.7342	0.9677	19	141	0.7360	0.8779	24	142
20	0.9939	0.9999	23	96	0.9498	0.9999	28	162
21	0.9695	0.9999	22	89	0.8431	1.0007	27	133
22	0.9637	1.0004	22	90	0.8431	1.0007	27	133
23	0.9637	1.0004	21	88	0.8431	1.0007	27	133

40-Task Problem / **45-Task Problem**

Heuristic	POT (40)	CTR (40)	Crew (40)	Equip (40)	POT (45)	CTR (45)	Crew (45)	Equip (45)
1	0.7598	0.8061	36	83	0.7294	0.8081	35	79
2	0.6995	0.8209	34	92	0.6479	0.8667	34	131
3	0.7011	0.8045	34	98	0.7747	0.7603	35	118
4	0.6623	0.8297	33	129	0.7351	0.8257	35	104
5	0.6800	0.8753	35	193	0.6602	0.8832	34	179
6	0.7221	0.8289	34	126	0.6981	0.8325	34	185
7	0.7221	0.8289	34	126	0.7918	0.8748	35	177
8	0.6893	0.7900	34	112	0.7508	0.8873	35	147
9	0.7787	1.0000	32	1280	0.7252	1.0016	32	1440
10	0.8612	0.8212	49	49	0.8967	0.8861	53	53
11	0.7528	0.8183	35	100	0.7479	0.7759	36	119
12	0.6703	0.8139	34	105	0.7199	0.7856	35	100
13	0.7465	0.8315	36	113	0.7620	0.7860	37	85
14	0.8097	0.8639	36	119	0.7344	0.7052	35	108
15	0.7587	0.8307	35	103	0.7414	0.8395	37	90
16	0.7466	0.8337	35	114	0.7678	0.8092	36	109
17	0.7727	0.8281	35	112	0.7747	0.7081	37	88
18	0.6849	0.6340	35	79	0.6572	0.7968	33	98
19	0.6384	0.8852	32	507	0.6726	0.8831	33	226
20	0.9172	0.9992	38	259	0.8331	0.9804	41	168
21	0.9119	0.9712	38	146	0.9305	0.9509	40	161
22	0.9287	0.9972	38	150	0.9129	0.9173	39	157
23	0.9186	0.9724	38	149	0.8814	0.9100	38	157

74-Task Problem

Heuristic	POT (74)	CTR (74)	Crew (74)	Equip (74)
1	0.7355	0.7215	64	128
2	0.7020	0.8152	61	245
3	0.7569	0.6411	63	302
4	0.6683	0.8263	60	309
5	0.7322	0.7931	60	398
6	0.6914	0.8467	60	460
7	0.7886	0.8700	61	476
8	0.6830	0.7980	60	236
9	0.6934	0.5415	32	2368
10	0.8954	0.7896	95	95
11	0.7426	0.7896	64	186
12	0.7438	0.8463	64	256
13	0.7438	0.8463	64	256
14	0.7749	0.8205	65	218
15	0.7302	0.8019	64	228
16	0.7678	0.7516	64	182
17	0.7302	0.8019	64	228
18	0.7355	0.7215	64	128
19	0.6930	0.8339	59	776
20	0.9149	0.9596	70	361
21	0.8332	0.9213	67	314
22	0.8570	0.9363	68	338
23	0.7871	0.9168	66	318

References

Allen, R., A. Athanassopoulos, R. G. Dyson, and E. Thanasoulis. (1997). "Weights Restrictions and Value Judgements in Data Envelopment Analysis: Evolution, Development and Future Directions." *Annals of Operations Research* 73, 13–34.

Arcus, A. L. (1966). "COMSOAL: A Computer Method of Sequencing Operations for Assembly Lines." *International Journal of Production Research* 4, 259–277.

Banker, R. D., A. Charnes, and W. W. Cooper. (1984). "Some Models for Estimating Technical and Scale Inefficiencies in Data Envelopment Analysis." *Management Science* 30, 1078–1092.

Baybars, I. (1986). "A Survey of Exact Algorithms for the Simple Assembly Line Balancing Problem." *Management Science* 32, 909–932.

Boussafiane, R. G., R. G. Dyson, and E. Thanassoulis. (1991). "Applied Data Envelopment Analysis." *European Journal of Operational Research* 52, 1–15.

Buxey, G. M. (1974). "Assembly Line Balancing with Multiple Stations." *Management Science* 20, 1010–1021.

Charnes, A., W. W. Cooper, and E. Rhodes. (1978). "Measuring the Efficiency of Decision Making Units." *European Journal of Operational Research* 2, 429–444.

Charnes, A., W. W. Cooper, A. Y. Lewin, and L. M. Seiford. (1994). *Data Envelopment Analysis: Theory, Methodology and Applications*. Boston, MA: Kluwer Academic Publishers.

Ding, F., and L. Cheng. (1993). "An Effective Mixed-Model Assembly Line Sequencing Heuristic for Just-in-Time Production Systems." *Journal of Operations Management* 11, 45–50.

Doyle, J. R., and R. H. Green. (1991). "Comparing Products Using Data Envelopment Analysis." *OMEGA: International Journal of Management Science* 19, 631–638.

Gaither, N. (1996). *Production and Operations Management*, Seventh Edition. Belmont, California: Wadsworth Publishing Co., pp. 297–299.

Gonzalez-Lima, M., R. A. Tapia, and R. M. Thrall. (1996). "On the Construction of Strong Complementary Slackness Solutions for DEA Linear Programming Problems Using a Primal-Dual Interior Point Method." *Annals of Operations Research* 66, 139–162.

Helgeson, W. B., and D. P. Birnie. (1961). "Assembly Line Balancing Using the Ranked Positional Weight Technique." *The Journal of Industrial Engineering* 12, 394–398.

Mariotti, J. (1970). "Four Approaches to Manual Assembly Line Balancing." *Industrial Engineering* 21, 35–40.

McMullen, P. R. (1995). "A Simulation Approach to Solving the Type I Assembly Line Balancing Problem for Mixed-Models with Stochastic Task Durations." Doctoral Dissertation: University of Oregon.

McMullen, P. R., and G. V. Frazier. (1997). "A Heuristic for Solving the Mixed-Model Line Balancing Problem With Stochastic Task Durations and Parallel Stations." *International Journal of Production Economics* 51(3), 177–190.

McMullen, P. R., and G. V. Frazier. (1998). "Using Simulated Annealing To Solve A Multiobjective Assembly Line Balancing Problem With Parallel Work Stations." *International Journal of Production Research* 31(10), 2717–2741.

Moodie, C. L., and H. H. Young. (1965). "A Heuristic Method of Assembly Line Balancing for Assumptions of Constant of or Variable Work Element Times." *The Journal of Industrial Engineering* 16, 23–29.

Pinto, P. A. (1975). "Assembly Line Balancing with Paralleling." Doctoral Dissertation: University of North Carolina at Chapel Hill.

Pinto, P. A., D. A. Dannenbring, and B. M. Khumawala. (1975). "A Branch and Bound Algorithm for Assembly Line Balancing with Paralleling." *International Journal of Production Research* 13, 183–196.

Pinto, P. A., D. A. Dannenbring, and B. M. Khumawala. (1981). "Branch and Bound and Heuristic Procedures for Assembly Line Balancing with Paralleling of Stations." *International Journal of Production Research* 19, 565–576.

Talbot, F. B., J. H. Patterson, and W. V. Gehrlein. (1986). "A Comparison of Heuristic Line Balancing Techniques." *Management Science* 32, 430–454.

Thompson, R. G., P. S. Dharmapala, J. Diaz, M. D. Gonzalez-Lima, and R. M. Thrall. (1996). "DEA Multiplier Analytic Center Sensitivity with an Illustrative Application to Independent Oil Companies." *Annals of Operations Research* 66, 163-177.

Thompson, R. G., P. S. Dharmapala, and Robert M. Thrall. (1994). "Sensitivity Analysis of Efficiency Measures with Applications Farming and Illinois Coal Mining." In *Data Envelopment Analysis: Theory, Methodology and Applications*. Boston: Kluwer Academic Publishers, pp. 393–422.

Thompson, R. G., L. N. Langemeier, C. Lee, E. Lee, and R. M. Thrall. (1990). "The Role of Multipler Bounds in Efficiency Analysis with Application to Kansas Farming." *Journal of Econometrics* 46, 93–108.

Thrall, Robert M. (1996). "Duality, Classification and Slacks in DEA." *Annals of Operations Research* 66, 109–137.

Tonge, F. M. (1961). "A Heuristic Program of Assembly Line Balancing." *Management Science* 7, 21–42.

Tonge, F. M. (1965). "Assembly Line Balancing Using Probabilistic Combinations of Heuristics." *Management Science* 11, 727–735.

Wong, Y.-H. B., and J. E. Beasley. (1990). "Restricting Weight Flexibility in Data Envelopment Analysis." *Journal of the Operational Research Society* 9, 829–835.