Interfaces
Vol. 42, No. 5, September–October 2012, pp. 478–491
ISSN 0092-2102 (print) | ISSN 1526-551X (online)
http://dx.doi.org/10.1287/inte.1120.0650

Ford Motor Company Implements Integrated Planning and Scheduling in a Complex Automotive Manufacturing Environment

Ada Y. Barlatt
Department of Management Sciences, University of Waterloo, Waterloo N2L 3G1, Ontario, Canada, abarlatt@uwaterloo.ca

Amy Cohn
Industrial and Operations Engineering Department, University of Michigan, Ann Arbor, Michigan 48109, amycohn@umich.edu

Oleg Gusikhin, Yakov Fradkin
Research and Advanced Engineering, Ford Motor Company, Dearborn, Michigan 48121, {ogusikhi@ford.com, yfradkin@ford.com}

Rich Davidson
Information Technology, Ford Motor Company, Allen Park, Michigan 48101, rdavidso@ford.com

John Batey
Stamping Business Unit, Ford Motor Company, Woodhaven, Michigan 48183, jbatey1@ford.com

Ford Motor Company has designed, developed, and implemented a collective system of decision support tools, supply chain visualization methods, and optimization techniques to aid its planning and scheduling in a complex automotive manufacturing environment. Although originally developed particularly for stamping plants, the system can also be applied in other manufacturing environments where setup times play a dominant role. By enabling substantial improvements in workplace planning and production scheduling, the implementation of this system has resulted in significant reductions in premium freight charges, overtime wages, and inventory costs.

Key words: cyclic schedule; manufacturing; heuristic; integer programming; workforce planning.

Automotive stamping plants produce vehicle body parts, such as hoods and door panels. Stamping is one of the most complex operations in the automotive supply chain, supplying hundreds of part types to dozens of assembly plants and service facilities. In this paper, we present decision support tools and optimization techniques to solve stamping planning (SP), workforce planning, and production scheduling problems.

The motivation for this research stems from Ford's need to improve the efficiency of its stamping operations. Ford's stamping business unit had recognized this need for a long time; over the years, it made several attempts (at considerable cost to the company) to address the need. Those attempts failed because they did not capture the complexity of the operational requirements. SP scheduling is difficult because of (1) the general challenges found in most machine scheduling problems and (2) stamping-specific operational requirements, as we describe in the following sections.

Stamping Plant Operations

The automotive stamping process begins by inserting a large roll of sheet steel into a blanking press. This press cuts the sheet steel into pieces, called blanks, that are slightly larger than the final parts. The blanks are then passed to a stamping pressline that contains matching upper and lower dies. As a blank moves through the stamping pressline, the dies shape the blank into a three-dimensional part. These parts may then move onto subassembly and (or) assembly workstations within the stamping plant, or be shipped directly to other manufacturing facilities for use in the final assembly.

This research focuses on creating effective cyclic workforce plans for the pressline stage of production. Our goal is to develop a two-week schedule, to be repeated over an extended period. The pressline stage of production is the bottleneck operation because it has the most binding capacity constraints. Labor costs are the dominant costs in this system. Thus, our primary objective in creating the production plan is to minimize the labor costs, subject to satisfying part demand downstream.

Solution Approach

To address these challenges, we built a decision support tool, the just-in-time execution and distribution information system (JEDI); Gusikhin and Klampfl (2012) and Gusikhin and Rossi (2004) provide more details of its structure. JEDI acquires supply chain and plant floor data and integrates these data into the planning and scheduling information for visualization, decision support, and optimization (Gusikhin and Rossi 2005). In its initial implementation, JEDI focused on collecting stamping plant information, presenting this information in an intuitive way, and providing decision support capabilities to enable what-if analyses for interactive production and distribution scheduling. Even with only these fundamental capabilities, JEDI enabled schedulers to make better decisions, leading to reductions in overtime and premium freight during its first full pilot study. Furthermore, the underlying JEDI structure enabled and supported its subsequent enhancement with a series of novel models and algorithms that collectively form the stamping scheduling optimizer (SSO). SSO generates initial high-quality production plans via JEDI and provides them to schedulers, who can make additional modifications and conduct what-if analyses using the interactive JEDI tools.

At its core, SSO uses composite variables (CVs). CVs encapsulate multiple discrete decisions simultaneously within a single variable, enabling certain problem constraints to be captured within the variable definition. For example, we use CVs to represent feasible sequences of events that can be scheduled within a given shift. This enables many of the complex rules governing feasible changeovers to be embedded within the variables. We refer to these CVs as shift schedules, because each CV represents the complete activities for a feasible shift within the stamping facility.

CV modeling successfully improved realism and tractability for many real-world applications with complex operational constraints in transportation and logistics; Appelgren (1969), Armacost et al. (2002), Barnhart et al. (2009, 2002), Caraffa et al. (2001), Cohn and Barnhart (2006), Cohn et al. (2007), and Crainic and Rousseau (1987) provide examples. This research, including our previous work, Barlatt et al. (2008 and 2010), is among the first research projects to use CVs in production planning.

Results

In 2004, the research team successfully transferred the JEDI system to Ford's information technology (IT) group for production implementation and full deployment across Ford's North American stamping facilities. Based on the success of this deployment, many other Ford manufacturing locations worldwide are currently evaluating JEDI.

The manufacturing operations of automotive suppliers are often complex and highly unstable; automotive suppliers operate under conditions of extreme competitive pressure. In this environment, operational efficiency is essential to being successful. JEDI has helped to address these challenges. It has led to reductions in scheduled weekend overtime (e.g., an average reduction of 30 percent within the first year) and helped schedulers best utilize available capacity (e.g., a reduction of 40 percent in excess transportation costs within the first year, saving more than $1 million in some plants).

Stamping-Specific Operational Requirements

Each stamping pressline is assigned a specific set of part types to produce. The key challenge in scheduling production for a given pressline is the changeover between dies (i.e., the setup time incurred when moving from the production of one part type to another). A changeover consists of both an external preparation and an internal dieset to change the pressline from the production of part type A to that of type B.

First, the dies for part type B must be prepared. This external preparation can take place while part type A

is being produced, but it cannot begin before production of part type A starts. Second, once the external preparation (which can take several hours) has been completed and after the production of part type A has ended, the internal dieset to the dies for part type B may take place. This can take from as little as a few minutes to as much as several hours, depending on the technology used; during this period, no parts can be produced on the pressline. This changeover process uses the concept of single-minute exchange of die (SMED) (Shingō 1996), which strives to reduce the internal changeover time (i.e., the time required to stop production of one part type and start production of another part type) by, for example, converting internal setup operations to external ones. External preparation can be done without stopping the line, whereas internal diesets require stopping it. Although the internal dieset setup time may become practically negligible, the external preparation time continues to be a major complicating factor in production planning.

Two key operational policies govern changeovers at Ford. First, the internal dieset to A should be fully contained within a single shift. Second, the subsequent external preparation time for B should also be fully contained in a single shift, because a single group of workers maintains full responsibility over each activity. Figure 1 illustrates proper and improper external preparation and internal dieset scheduling across two shifts, n and $n+1$. The Δ represents an internal dieset, a letter denotes production of a particular part type, and the dots represent an external preparation.

We note that the workforce costs dominate all other costs in the facility. The pressline uses two types of workers: direct laborers are responsible for operating the presslines, and indirect labor crews are responsible for the external preparation and conducting the internal diesets. Each part type requires a specific number of direct laborers to be present during production. One indirect labor crew is required during external preparation and internal dieset. Laborers must be hired for the entire planning horizon within a given shift type (i.e., first, second, or third shift of the day). Therefore, the daily direct labor staffing level for a given shift type is the maximum direct labor requirement across all such shifts in the planning horizon, and the daily indirect labor staffing level for a given shift type is the maximum indirect labor requirement across all such shifts in the planning horizon.

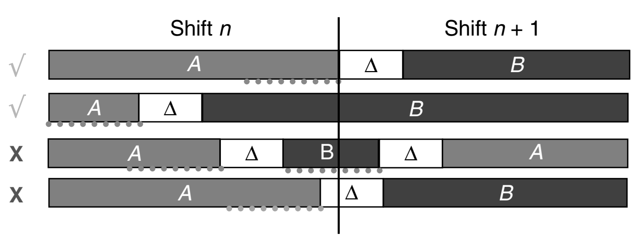

Figure 1: The external preparation denoted with the dotted lines and the internal die set changes (Δ) should be fully contained within a shift.

Stamping Press Schedules

The sequence of the part types and the time during which each part type is produced defines a pressline schedule. Each schedule has a corresponding workforce allocation.

Table 1 illustrates a two-day sample pressline schedule. This table includes the shift number and type, the tasks completed in the shift, the number of direct laborers required (denoted by D Req.), the number of indirect labor crews (denoted by I Req.), the number of direct laborers scheduled in the shift (denoted by D Sched.), and the number of indirect crews (denoted by I Sched.). In the shift tasks column, the letter indicates the part type; "idle" indicates that the pressline is set up to produce a part type, but the workers are currently idle; Δ_{AB} indicates the time to complete the internal dieset from part type A to part type B.

In this example, part type A requires three direct laborers for production, B requires four, and C requires one. Each changeover requires one indirect labor crew. The planning horizon begins by producing part type C for eight hours in shift 1, followed by a completely idle shift; however, the pressline remains set up for part type C. The third shift produces C for one hour, then is set up to produce A. A is produced for all of shift 4. In shift 5, the pressline is set up to produce B; the remainder of the shift is used for production. B is also produced for the first half hour of shift 6, after which the pressline is set up to produce C, which takes up the remainder of the shift.

Shift	Shift type	Shift tasks	D Req.	D Sched.	I Req.	I Sched.
1	First	C	1	$\max(1,3)=3$	0	$\max(0,0)=0$
2	Second	C-idle	0	$\max(0,4)=4$	0	$\max(0,1)=1$
3	Third	C, Δ_{CA}, A	3	$\max(3,4)=4$	1	$\max(1,1)=1$
4	First	A	3	$\max(1,3)=3$	0	$\max(0,0)=0$
5	Second	Δ_{AB}, B	4	$\max(0,4)=4$	1	$\max(0,1)=1$
6	Third	B, Δ_{BC}, C	4	$\max(3,4)=4$	1	$\max(1,1)=1$

Table 1: The number of direct laborers required during a shift is the maximum number of laborers for each part produced during the shift. One indirect labor crew is required for each changeover. The number of laborers scheduled for each shift type is the maximum number of laborers required across all shifts for that shift type.

Let $\{d'_1\, d'_2\, d'_3 \ldots \mid i'_1\, i'_2\, i'_3 \ldots\}$ represent the labor requirements in the production schedule. Then the labor requirements for the production schedule for this example are $\{1\,0\,3\,3\,4\,4 \mid 0\,0\,1\,0\,1\,1\}$.

Using this information, we can determine the workforce allocation. Recall that the number of laborers for each shift type is the maximum across all shifts in that shift type. Let $[d_1\, d_2\, d_3 \mid i_1\, i_2\, i_3]$ represent a workforce allocation with d_1 direct laborers (i_1 indirect labor crew) in the first shift type, d_2 direct laborers (i_2 indirect labor crew) in the second shift type, and d_3 direct laborers (i_3 indirect labor crews) in the third shift type. Thus, the workforce allocation for this example is $[3\,4\,4 \mid 0\,1\,1]$.

The objective of the pressline workforce planning problem is to minimize the cost of labor subject to the following constraints.

1. Three eight-hour shifts operate each day.
2. All daily demands must be met on time.
3. Production of a given part type can only take place when the pressline is set for that part.
4. Changeover operating policy constraints are enforced. The internal dieset for a given part type cannot occur until the external preparation has been completed. External preparation for a pressline cannot begin until the prior internal dieset on the pressline has been completed. Production for a pressline cannot occur while an internal dieset is taking place on the pressline. Each internal dieset must be conducted completely within a single shift. Each external preparation must be conducted completely within a single shift.
5. Labor requirements are enforced. Production cannot occur unless the correct number of direct laborers are present at the pressline. Internal diesets and external preparation cannot occur unless the correct number of indirect labor crews are present at the pressline.
6. Laborers must be hired for the entire planning horizon within a given shift type.
7. Laborers are shared among the machines in the pressline zone, which is a department within the stamping plant that shares a crew of direct and indirect laborers among several presslines.

Assumptions

The assumptions for the stamping workforce planning problem are as follows:

- Adequate raw materials are always available. Because the pressline is the bottleneck operation, blanks are always available to feed the pressline.
- Adequate storage space is always available. We also assume no capacity limitations for storing completed output from the presslines.
- Initial inventories can be decision variables or input parameters. We do not require inventory levels to be specified as inputs. We allow the user the option of specifying the inventory values as inputs or as variables, allowing the model to determine the inventory levels.
- The problem is static, deterministic, and repeating. The goal is to develop a two-week schedule, to be repeated over an extended period, composed of at most three shifts per weekday. Although demand

may vary from day to day over the two-week horizon, we assume that the inventory levels at the start and end of the planning horizon are the same.

The production environment is not fully deterministic—fluctuations in daily demand and yield, machine failures, and other disruptions will occur. Thus, SSO is used to create an initial basic schedule, and the scheduler modifies this schedule within JEDI daily to recover from minor deviations resulting from daily variability.

Challenges

A stamping production plan has two components: the workforce allocation (i.e., the number of laborers of each type available during each shift type in the planning horizon) and the workforce utilization (i.e., the production schedule). Solving the workforce allocation and utilization problem simultaneously is challenging for three reasons. The first is the complexity of the rules regarding when and how part type changeovers can occur. The second relates to the rules requiring laborers for one shift to be hired for all shifts of that type across the planning horizon. The third is the sharing of laborers across multiple presslines. These challenges introduce tremendous computational complexity into traditional mathematical programming modeling and solution techniques. In the next section, we present our approaches to overcoming the challenges.

Solution Approach

As we quickly discovered when we began this research, using a traditional mixed-integer programming (MIP) approach for even a single pressline is intractable because of the inherent weakness of the linear program (LP) relaxation, which is common to many machine scheduling problems, and the complexities associated with modeling the changeover requirements.

Therefore, we instead developed SSO, which contains of a series of four phases solved in succession, with the solution from each phase providing a useful bound for the subsequent phases.

This approach builds upon two ideas developed in Barlatt et al. (2008): shift schedule variables and test and prune (T&P). We will first give some background information on these two ideas before explaining how they are applied within the four SSO phases.

Shift Schedule Variables

A shift schedule variable is a variable that represents an entire feasible shift of production. For example, we define one shift schedule to represent workers producing part type A for eight hours. Another shift schedule represents workers producing part type B for the first hour, changing over to part type A, and then producing A for the remainder of the shift. Note that this variable definition allows us to not require that batch sizes, number of changeovers, labor availability, or sequencing of part types be restricted or predefined to achieve tractability, as is often the case in most machine scheduling literature. In addition, we are able to allow sequence-dependent changeover times and demand-specific due dates, which are also enhancements over much of what is in the literature.

When using CVs, the goal is to embed complexity within the variable definition to simplify the objective function and (or) complicating constraints. In the stamping scheduling problem, the complexity is largely shift specific. For example, changeovers must be fully contained within a shift, and workforce calculations are also made at the shift level. Therefore, we define CVs for this problem to represent a feasible set of ordered tasks that can be completed in an individual shift.

By defining the variables in this way, many of the constraints are automatically enforced—a shift schedule is not defined if it does not satisfy the operational rules associated with changeovers. Additionally, each shift schedule has associated characteristics (e.g., tasks worked on, duration of tasks, number of changeovers) that can be used in constraints, and the objective function; see Figure 2 for some feasible examples.

Of course, the number of possible shift schedule variables is infinitely large. One way to overcome this challenge, similar to that seen in much of the CV literature, is to limit the number of candidate variables by discretizing time. For example, we might only consider production in half-hour increments and then enumerate the corresponding set of valid shift schedules. However, because of the combinatorial aspect of scheduling more than one job in a shift, this number might still be quite large. Thus, this discretization might result in the need for delayed column generation. Furthermore, this restriction of the

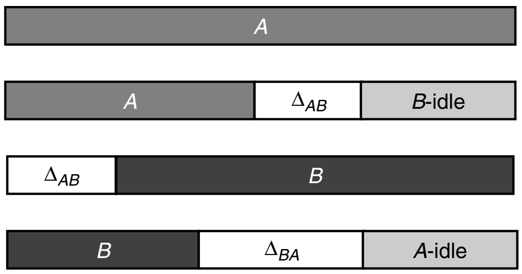

Figure 2: Shift schedule variables represent an entire feasible shift of production. A shift schedule is not defined if it does not satisfy the operational rules associated with changeovers. We define a small number of extreme shift schedules in the model. Convex combinations of extreme shift schedules represent every feasible shift schedule.

solution space can decrease the quality of the resulting schedule.

In Barlatt et al. (2008), we show that we only require a few specifically designed variables, which we refer to as extreme shift schedules, to capture the entire feasible region; we can easily capture the other operational requirements with linear constraints. The convex combination of extreme shift schedules defines the entire feasible region. Using the extreme shift schedules reduces the number of variables, allows the extreme shift schedules to be continuous rather than integer, and refines the granularity. By using these extreme shift schedules, we both decrease the number of variables (relative to this proposed, discretized-time approach) and increase the solution quality by implicitly incorporating all possible shift schedules.

For example, a shift schedule in which part type A is run for four hours, followed by four hours of idle time, can also be represented by taking the average of two shift schedules, one with eight hours of producing A and the other with eight hours of idle time (during which the pressline is set for part A). Any combination of production of A plus idle time can be found by taking a convex combination of these two extreme shift schedules. We can identify extreme shift schedules when more than one part type is also produced in a shift.

By using this approach, we achieve three benefits that are critical in achieving tractability.

1. A convex combination of extreme shift schedules is also feasible and captures all the complex changeover constraints. Thus, the number of variables is greatly reduced. Each shift has only two extreme shift schedules per part type (one with eight hours of production and one with eight hours of idle but set for production) and four extreme shift schedules for each ordered pair of part types. Thus, an instance with 10 part types would have only 380 extreme shift schedules. We contrast this with other CV models, which often have variables numbering in the millions or even billions (e.g., the airline crew pairing problem), as Crainic and Rousseau (1987) discuss.

2. The solution set is exhaustive. We do not need to discretize, which would decrease solution quality, to ensure tractability. We can represent the entire feasible region with a small set of variables in the model at the start, thus negating the need to generate any columns.

3. The extreme shift schedule variables become continuous, greatly reducing the number of integer variables in the model and the corresponding amount of branching.

This approach also has a significant implementation benefit—we can use a commercial integer programming solver (e.g., CPLEX) directly instead of coding the algorithm, as is required with column generation within a branch-and-bound framework. This makes the extreme schedule approach faster to implement and solve than a column generation approach.

The final implementation benefit of this approach is that we can further limit the extreme shift schedules included in the model if the user has specific requirements of the solution. For example, if the user does not want part type A to be followed by part type B within a shift, we can simply remove the extreme shift schedules that correspond to that situation.

Test and Prune

Despite the benefits of the using shift schedule variables to capture the changeover constraints, and extreme shift schedules to reduce the number of variables, significant fractionality still exists with regard to the remaining auxiliary variables in the model, leading to significantly long run times for several instances. As an alternative solution approach, we developed T&P, a new algorithm for solving resource allocation and utilization problems. To motivate this algorithm, consider the following two observations.

1. The number of possible workforce allocations is finite. Although an infinite number of distinct feasible sequences and durations of production runs exists,

the number of unique workforce allocations is discrete, finite, and often quite small. Many feasible production schedules will correspond to a common workforce allocation, which in turn determines the solution cost.

2. When the workforce allocation is fixed, the remaining problem is a feasibility problem. If we know the workforce allocation, then the problem that remains is finding a feasible production plan that does not exceed the allocated workforce. Computational experiments for the stamping case studies show that this feasibility problem is typically easy to solve.

The basic concept behind T&P is to iteratively search over all possible workforce allocations; in each iteration, we test the feasibility of a specific workforce allocation (e.g., can we meet the demand if we hire 10 workers for the first shift type, 8 for the second, and 6 for the third?). If a feasible production schedule exists, we can reduce the set of allocations that must be tested by disregarding any allocation with higher cost, because such allocations will all be suboptimal. If a feasible production schedule does not exist for a given workforce allocation, we can reduce the search space by removing any allocation with fewer resources in each shift type because such allocations will also be infeasible. Any allocation with fewer resources will also be infeasible. We found that T&P can yield provably optimal solutions much faster than branch and bound.

Stamping Scheduling Optimizer

Based on the background information above, we can describe the four phases in JEDI's SSO. In Phase 1, we find an upper bound on the overall problem, solving each pressline independently to find the minimum workforce allocation needed for that machine. Note that this neglects the possible synergies that can be found by sharing workers across presslines. In Phase 2, we improve upon this upper bound by solving the rotation model (see *Rotation Model* in the appendix), which loops through all potential starting points for each (cyclic) schedule to leverage synergies across presslines and reduce costs. In Phase 3, we establish a lower bound via the aggregate workforce model (see *Aggregate Workforce Model* in the appendix), which aggregates the demands for individual part types into general bounds on the amount of labor needed. In some cases, the optimal solution to the aggregate workforce model is also operationally feasible. In these cases, we are finished and the SSO returns the solution to the aggregate workforce model to JEDI. However, when the optimal solution to this model is not operationally feasible, we enter Phase 4, the multiple machine feasibility algorithm. This heuristic makes the operationally infeasible solution to the aggregate workforce model problem feasible by adding additional workers. Figure 3 provides an overview of the approach.

In the next sections, we will use a pressline zone example with two presslines, α and β, to illustrate our approach. Assume that pressline α produces part types A, B, C, and D, and pressline β produces part types E and F. Table 2 summarizes the key information about the machines and products. We assume a planning horizon of four days (12 shifts).

Phase 1: Set Upper Bound

Recall that we can use T&P and the single-machine model to obtain an optimal workforce allocation for each pressline, assuming no sharing of labor across presslines. After finding the optimal solution for each machine individually, we can quickly establish an upper bound on the pressline zone by simply summing together these workforce allocations.

Figure 4 depicts optimal schedules for machines α and β. The optimal workforce allocation found for machine α is $[6\,6\,3 \mid 0\,1\,0]$, and the optimal allocation for machine β is $[2\,3\,0 \mid 0\,1\,0]$. The upper bound established at the end of the first phase is therefore $[8\,9\,3 \mid 0\,2\,0]$.

Phase 2: Reduce Upper Bound

The second phase uses the single pressline solutions as input and looks to reduce the upper bound established in the previous phase by rotating the production schedules. Recall that the two-week planning horizon is cyclic; we can thus rotate each individual pressline's schedule to try to reduce the overall cost for the pressline zone. The rotation model is an MIP with variables z_{mn}, which equal 1 if shift n on machine m is rotated to be the first shift, and 0 otherwise.

Figure 5 illustrates this example for presslines α and β, showing the original schedules for presslines

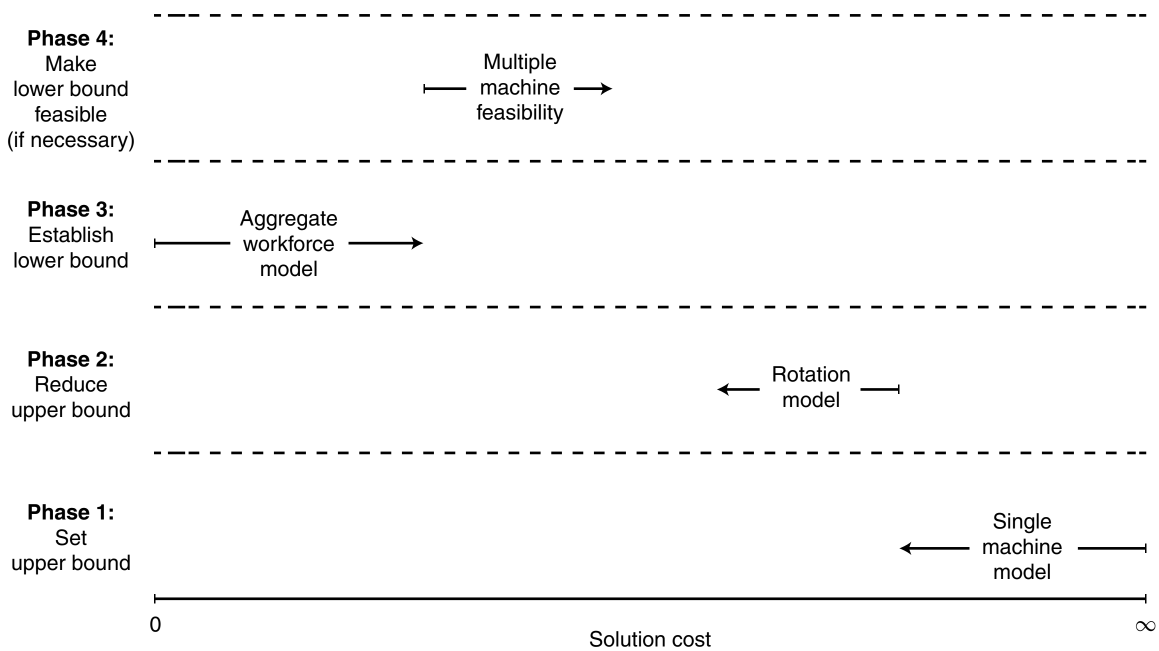

Figure 3: At each of the solution approach's four phases, we establish or improve bounds to determine a solution to the problem.

α and β and the rotated schedule (denoted as β'). The resulting pressline zone workforce allocation after using this rotation model is $[6\,8\,3\,|\,0\,2\,0]$, requiring two fewer direct laborers in the first shift type and one fewer in the second shift type compared to the result of Phase 1. To obtain this lower-cost solution, β was rotated, moving what had been shift 7 to shift 1.

Phase 3: Establish Lower Bound

In this phase, we take a new perspective to establish a lower bound on the optimal workforce allocation.

Product name	Machine	Total demand (hrs)	Number of workers required
A	α	0.75	2
B	α	18.55	3
C	α	5.93	6
D	α	8.6	2
E	β	7.3	3
F	β	32.5	2

Table 2: The table summarizes key data for a fictional pressline zone to illustrate the solution approach. The fictional pressline zone has two presslines. Pressline α produces part types *A*, *B*, *C*, and *D*; pressline β produces part types *E* and *F*.

Rather than looking at each part type individually, we aggregate all part types that have a common labor requirement (i.e., that require the same number of direct laborers). Table 3 illustrates this aggregation for our simple example.

Within the aggregate workforce model, we use the shortest sequence-dependent setup time between aggregate part types to establish a lower bound. For example, on pressline α, we determine the internal dieset time when changing from the aggregate part type of all parts requiring two direct laborers to the aggregate part type associated with all part types requiring three direct laborers, by taking the minimum between Δ_{AB} and Δ_{DB}. We do the same with the external preparation time.

The aggregate workforce model has shift schedule variables that represent changing from one aggregate part type to another. We use these variables in an integer programming formulation. In the constraints, we ensure that the aggregate demand is met and that at least one changeover occurs for every part type that requires that number of workers (e.g., there must be at least two changeovers to the aggregate part type requiring two workers on pressline α). Because the

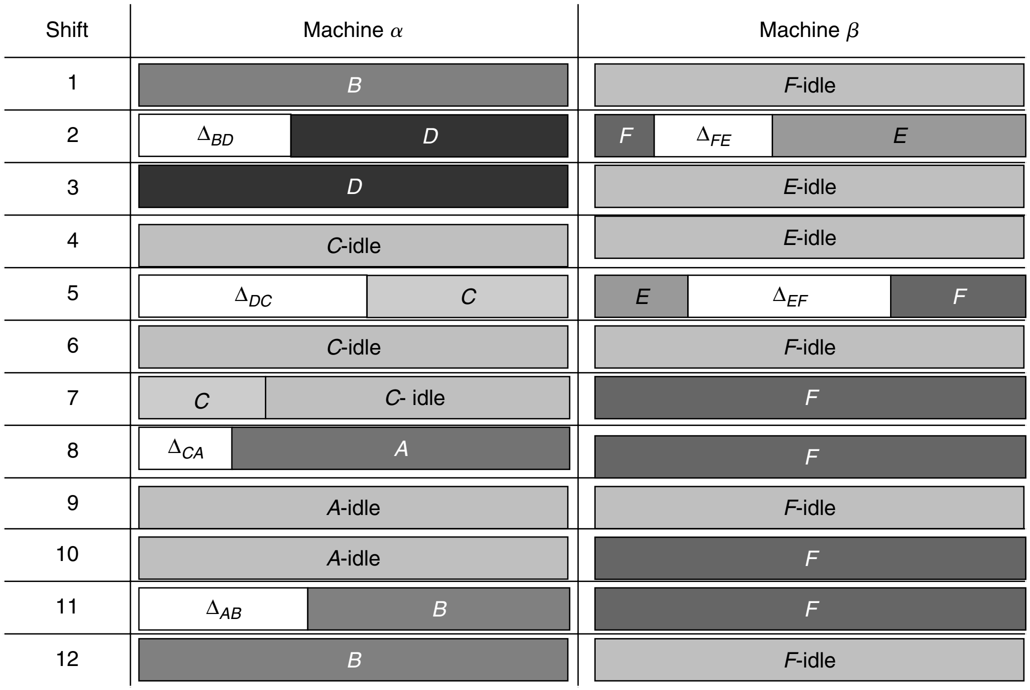

Figure 4: The figure illustrates the initial solution to Phase 1 of the approach for the pressline zone example.

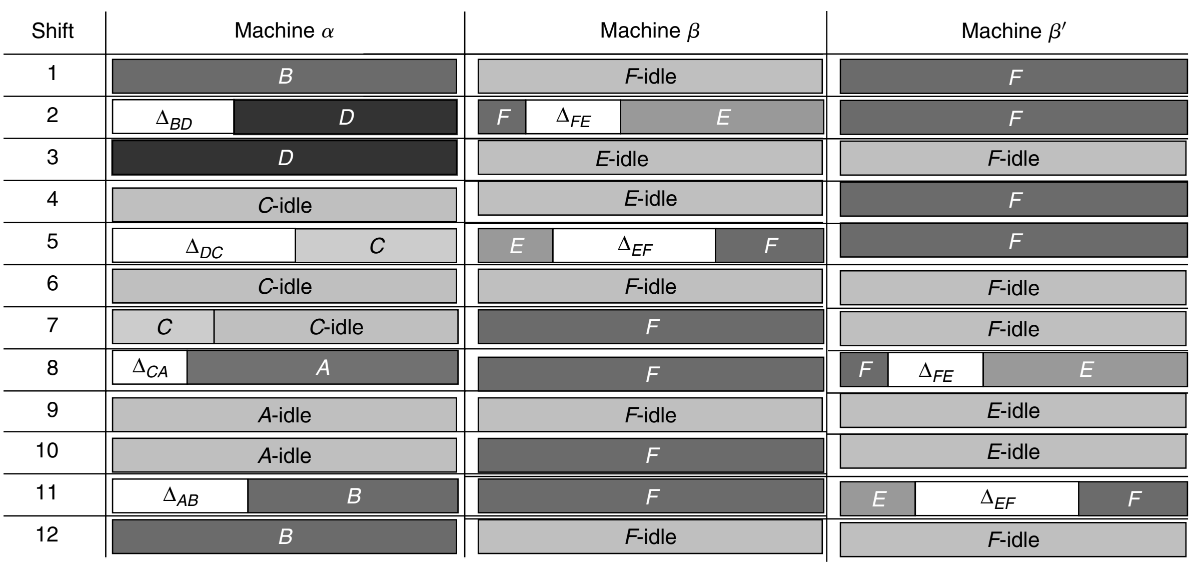

Figure 5: The figure illustrates the Phase 2 solution for the pressline zone example. This new rotated solution (denoted machines α and β') requires two fewer direct laborers in the first shift type and one fewer in the second shift type compared to the result of Phase 1 (denoted machines α and β).

Number of workers required (aggregate part type)	Machine	Total demand (hrs)	Part types (s)
2	α	$0.75 + 8.6 = 9.35$	*A*, *D*
3	α	18.55	*B*
6	α	5.93	*C*
2	β	7.3	*F*
3	β	32.5	*E*

Table 3: This table displays the aggregate part types for the pressline zone example. Part types are aggregated by the number of direct laborers required.

	Presslines α and β	Zone with five presslines
Total number of allocations	27,000	1,061,208
With Phases 1 and 2 upper bound	404	18,692

Table 5: This table compares the number of workforce allocations that must be considered by the T&P algorithm with and without Phases 1 and 2. The table shows that using the bounds from Phases 1 and 2 eliminates a significant number of possible solutions.

number of distinct direct labor requirements is quite small compared to the number of parts produced on the pressline, the number of aggregate products is also quite small, enabling us to concurrently consider all presslines within a zone.

Table 4 highlights the differences between the problems solved in Phases 1 and 3. First, Phase 1 provides an upper bound, whereas Phase 3 provides a lower bound. Second, Phase 1 finds the optimal workforce allocation for each pressline independently, without considering any other presslines in the zone. Phase 3 simultaneously considers all the presslines in the zone. Third, in Phase 1, the shift schedule variables represent actual part types; in Phase 3, they represent aggregate part types.

A disadvantage of Phase 3 is that the production schedule solution might not be operationally feasible. Therefore, we end this phase by checking if the production schedule that it produced is feasible for the original problem. If not, we move to Phase 4 to modify the solution, ensuring feasibility.

We note that the performance of Phase 3 can be greatly enhanced by using good upper bounds found in Phases 1 and 2. Table 5 illustrates the computational impact of using the Phases 1 and 2 upper bounds on Phase 3 performance.

	Phase 1	Phase 3
Provides:	An upper bound	A lower bound
Math model considers:	Each machine individually	All machines in the zone simultaneously
Shift schedules represent:	Actual products	Aggregate products
Operationally feasible solution:	Yes	Maybe

Table 4: This table compares the differences between the models solved in Phases 1 and 3.

Phase 4: Make Lower Bound Feasible

We use the multiple machine feasibility algorithm to convert an infeasible workforce aggregate production schedule to a feasible one. In this algorithm, we look at the labor requirements in the production schedule—the number of workers (both direct and indirect) assigned to each pressline during each shift in the planning horizon. We define the slack as a situation in which the workforce allocation provides more laborers than the production schedule requires during that shift. Let us assume that the lowest-cost allocation found during Phase 3 is [663 | 020]. Assume also that the labor requirements in the production schedule for α are {322060620330 | 010010010010} and the labor requirements in the production schedule for machine β are {220200003020 | 000000010010}. All shifts, except shifts 5, 7, and 9, have direct laborer slack. For example, the number of direct laborers required in shift 1 is 5 $(3+2)$; therefore, a slack of one direct laborer exists. Shift 2 and shift 5 are the only shifts with indirect laborer crew slack; these shifts only require one indirect labor crew, but two are available. We can then use the single-machine model to check if this allocation of workers to each shift (plus any slack in that shift) in the planning horizon is feasible. If we find a feasible production schedule, we update the slack based on the number of required workers in the production schedule and move to the next pressline; otherwise, we increment the number of workers allocated to that pressline. We keep adding workers until we can find a feasible production schedule or we reach the bound provided by Phase 2.

We summarize the steps of the algorithm as follows:

1. For the labor requirements for the ith machine in the zone, check if a schedule that can be made with the workforce allocated from the aggregate workforce model (plus any remaining slack) is feasible;
   (a) If feasible, update the slack and go to Step 3;
   (b) If infeasible, increase the workers available and go to Step 2.
2. Check to see if the current workforce allocation has lower cost than the upper bound provided in Phase 2;
   (a) If yes, go to Step 1;
   (b) If no, stop.
3. If $i =$ the number of presslines in the zone, stop; else, increment i and go to Step 1.

Implementation

Figure 6 presents an overview of how JEDI creates and displays productions plans.

When creating production plans, the scheduler interacts with two JEDI user interfaces: the production planning interface and the scheduling interface. The scheduler sets the optimization parameters (e.g., the presslines to consider) in the production planning interface. The information on the JEDI server and from the production planning interface is then passed to the central optimization server and the SSO begins. The result of the SSO optimization is then passed back to the scheduler's workstation and displayed via the scheduling interface.

The scheduling interface, which is interactive, allows the scheduler to view and edit the schedule. It provides a block diagram, similar to a Gantt chart, for the schedule representation. This chart displays the changeover start time, changeover duration, and production run time for each batch of parts. Once the SSO solution has populated the scheduling interface, the scheduler can analyze and manually adjust the schedule. Once this analysis (and modifications) is complete, the schedule is accepted and is passed back to the JEDI server for use in updating the MRP system and the requirements of the upstream supply chain.

It is important to note that the scheduler can set restrictions on the schedule before initiating the SSO in JEDI's scheduling interface. By limiting the extreme shift schedule variables used within the optimization, we enforce these restrictions within SSO (see the *Shift Schedule Variables* section).

Use Cases

JEDI can be used to create production plans for two use cases:

1. Pressline zone optimization, which is this paper's focus, is used for resource planning over long planning horizons (i.e., of at least four weeks). Pressline zone optimization is also necessary when new products are introduced and (or) customer demand requirements have changed substantially. To solve these problems, we use all four phases of the SSO.
2. Individual pressline optimization is used for operational decision analysis (e.g., within two weeks). It provides schedules for a pressline to get back on plan after a disruption or to make adjustments

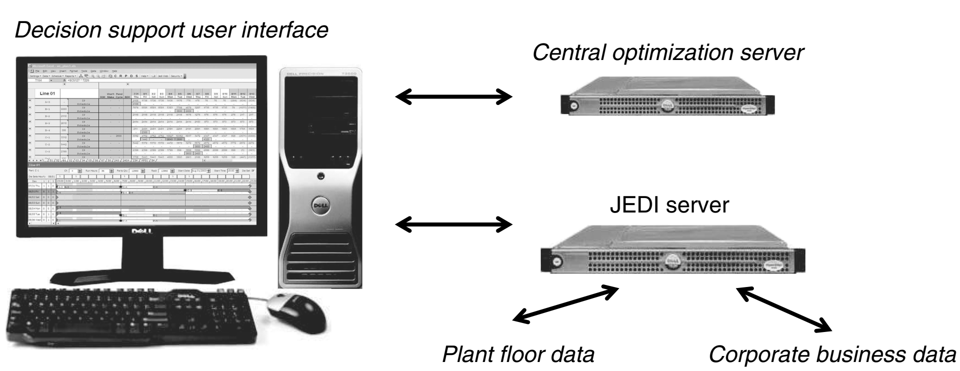

Figure 6: High-level JEDI architecture: JEDI has three main elements: (1) the JEDI server where the corporate business data (e.g., material requirements planning (MRP) data and plant floor data (e.g., changeover time and production rates)) are collected and consolidated, (2) decision support user interfaces accessed via the scheduler's workstation, and (3) the SSO. Gusikhin and Klampfl (2012) provide more details on the JEDI architecture.

because of minor changes in the customer demand requirements. Individual pressline optimization is particularly useful for presslines that have a large number of jobs with significantly different labor requirements. To solve these problems, we use only Phase 1 of the SSO.

Benefits

In the previous sections, we focused on the SSO's technical components and its implementation within JEDI. However, perhaps the synergies gained by the integration of SSO within the JEDI environment might be even more important. Production scheduling algorithms (including SSO) typically assume fixed and known input data; this assumption is usually necessary to ensure computational tractability. However, real manufacturing environments are dynamic and constantly changing. Given the high-quality initial production plans generated by SSO, the other capabilities of JEDI then allow the user great flexibility and power in adjusting the initial schedule in response to such changing circumstances as disruptions and variations in demand. Furthermore, once a change has been made, SSO provides a mechanism for quickly recovering from the current state back to the original production plan. It is also worth noting the various levels of planning for which the SSO within the JEDI system can be used. For example, long-term production and workforce plans can be created to determine the resources required for an extended period. In addition, more tactical plans can be developed based on the resources allocated in the long-term plan.

JEDI, a collective system of decision support tools, supply chain visualization methods, and optimization techniques has provided significant benefits to Ford, including substantial financial savings associated with reductions in premium freight, labor costs, and inventory. In addition, it has provided less tangible benefits associated with improved scheduling and manufacturing execution control. Although the specific financial benefits of JEDI depend on the complexity of the given plant environment, the savings that have typically been observed after the first year of a JEDI launch include a 30 percent average reduction of overtime labor costs and a reduction of premium transportation costs by 40 percent, thus saving more than $1 million per year at some plants. JEDI has been used successfully to manage complex automotive stamping operations at multiple locations for several years, facilitating sustained improvements in operating performance.

Finally, we conclude by noting that JEDI's benefits are not limited to stamping. The JEDI model, in combination with SSO, is built on general concepts of the automotive supply chain. Other operations in Ford's global network are exploring and piloting this system.

Appendix. Rotation Model

Sets

- **M:** the set of presslines.
- **N:** the set of shifts.
- **H:** the set of shift types.

Parameters

- α_{mn}: the number of direct laborers for each shift $n \in \mathbf{N}$ on pressline m.
- ω_{mn}: the number of indirect labor crews for each shift $n \in \mathbf{N}$ on pressline m.
- h_n: the shift type of shift $n \in \mathbf{N}$ (e.g., $h(1) = h(4) = h(7) = \text{first}$).
- c_h^d: the cost of direct laborers assigned to shift type h.
- c_h^i: the cost of indirect laborers assigned to shift type h.

Variables

- $q_{mn} = 1$ if the first α and ω for pressline m starts on shift n.
- y_h^d: the number of direct laborers assigned to shift type h.
- y_h^i: the number of indirect labor crews assigned to shift type h.

Objective

$$\min \sum_{h \in \mathbf{H}} y_h^d c_h^d + \sum_{h \in \mathbf{H}} y_h^i c_h^i. \tag{1}$$

Constraints

$$q_{00} + q_{01} + q_{02} = 1, \tag{2}$$

$$\sum_{n \in \mathbf{N}} q_{mn} = 1 \quad \forall m \in \mathbf{M}, \tag{3}$$

$$\sum_{m \in \mathbf{M},\, n' \in \mathbf{N}:\, n' \le n} \alpha_{m(n-n'+1)} q_{mn} + \sum_{m \in \mathbf{M},\, n' \in \mathbf{N}:\, n' > n} \alpha_{m(|\mathbf{N}|-n-n'-1)} q_{mn} \le y_h^d \quad \forall n \in \mathbf{N}, \tag{4}$$

$$\sum_{m \in \mathbf{M},\, n' \in \mathbf{N}:\, n' \le n} \omega_{m(n-n'+1)} q_{mn} + \sum_{m \in \mathbf{M},\, n' \in \mathbf{N}:\, n' > n} \omega_{m(|\mathbf{N}|-n-n'-1)} q_{mn} \le y_h^i \quad \forall n \in \mathbf{N}, \tag{5}$$

$$q_{mn} \in \{0, 1\}. \tag{6}$$

Expression 1, the objective, minimizes the workforce cost. Constraint 2 limits the rotation of the pressline to only the first three shifts; this reduces symmetry in the formulation. Constraint 3 assigns one shift to be the first shift for each pressline in the zone. Constraints 4 and 5 determine the number of direct laborers and indirect labor crews needed on each shift based on the number of shifts rotated.

Aggregate Workforce Model

Sets

- **M:** the set of presslines in the zone.
- $\mathbf{J}_m$: the set of aggregate labor shift schedules for pressline m; this is the number of direct laborers assigned to each shift during the day.
- $\mathbf{K}_m$: the number of possible direct laborers on pressline m (e.g., for pressline β in the example, this is $\{0, 2, 3\}$).
- **N:** the set of shifts.
- **H:** the set of shift types.

Parameters

- p_{mk}: the number of products that require k direct laborers on pressline m.
- d_{mk}: the total demand for k number of direct laborers (hours) on pressline m.
- a_h^i: the number of indirect labor crews assigned within the given workforce allocation.
- a_h^d: the number of direct laborers assigned within the given workforce allocation.
- $f_{mjk} = 1$ if k direct laborers are required to start the shift in aggregate labor shift schedule j on pressline m.
- $l_{mjk} = 1$ if k direct laborers are required to end the shift in aggregate labor shift schedule j on pressline m.
- $\delta_{mj} = 1$ if aggregate labor shift schedule j includes a changeover on pressline m.
- q_{mjk}: the duration of time assigned to k direct laborers in aggregate labor shift schedule j on pressline m.
- h_n: the shift type of shift n $\forall n \in N$ (e.g., $h(1) = h(4) = h(7) = \text{first}$).

Variables

- $0 \le z_{mnj} \le 1$ is the proportion of aggregate labor shift schedule j assigned to shift n on machine m.
- $v_{mnk}^f = 1$ if k direct laborers are assigned to start shift n on pressline m, 0 otherwise.
- $v_{mnk}^l = 1$ if k direct laborers are assigned to end shift n on pressline m, 0 otherwise.
- $\gamma_{mn} = 1$ if shift n has a changeover on pressline m, 0 otherwise.
- $u_{mnk} = 1$ if k direct laborers are used on pressline m during shift n, 0 otherwise.

Constraints

$$\sum_{j\in\mathbf{J_m}} z_{mnj} = 1 \quad \forall m \in \mathbf{M},\ n \in \mathbf{N}, \tag{7}$$

$$\sum_{n\in\mathbf{N}}\sum_{j\in\mathbf{J_m}} q_{mjk} z_{mnj} \ge d_{mk} \quad \forall m \in \mathbf{M},\ k \in \mathbf{K_m}, \tag{8}$$

$$\sum_{j\in\mathbf{J_m}} f_{mjk} z_{mnj} = v_{mnk}^f \quad \forall m \in \mathbf{M},\ n \in \mathbf{N},\ k \in \mathbf{K_m}, \tag{9}$$

$$\sum_{j\in\mathbf{J_m}} l_{mjk} z_{mnj} = v_{mnk}^l \quad \forall m \in \mathbf{M},\ n \in \mathbf{N},\ k \in \mathbf{K_m}, \tag{10}$$

$$\sum_{j\in\mathbf{J_m}} \delta_{mj} z_{njm} = \gamma_{mn} \quad \forall m \in \mathbf{M},\ n \in \mathbf{N}, \tag{11}$$

$$v_{mnk}^l = v_{m(n+1)k}^f \quad \forall m \in \mathbf{M},\ k \in \mathbf{K_m},\ n \in \{1 \ldots |\mathbf{N}|-1\}, \tag{12}$$

$$v_{m|\mathbf{N}|k}^l = v_{m1k}^f \quad \forall m \in \mathbf{M},\ k \in \mathbf{K_m}, \tag{13}$$

$$\sum_{m\in\mathbf{M}} \gamma_{mn} \le a_{h_n}^i \quad \forall n \in \mathbf{N}, \tag{14}$$

$$\sum_{m\in\mathbf{M}}\sum_{k\in\mathbf{K_m}} k u_{mnk} \le a_{h_n}^d \quad \forall n \in \mathbf{N}, \tag{15}$$

$$\sum_{n\in\mathbf{N}}\ \sum_{j\in\mathbf{J_m}:\, l_{mjk}=1 \text{ and } \delta_{mj}=1} z_{mnj} \ge p_{mk} \quad m \in \mathbf{M},\ k \in \mathbf{K_m}, \tag{16}$$

$$v_{mnk}^f,\ v_{mnk}^l,\ u_{mnk},\ \gamma_{mn} \in \{0, 1\}. \tag{17}$$

Constraint 7 assigns an aggregate labor shift schedule to each shift in the planning horizon. Constraint 8 enforces that each schedule must meet the demand. Constraints 9 and 10 determine how many direct laborers are assigned to each pressline to start and end its shift. Constraint 11 determines if a changeover is completed during each shift in the planning horizon on each pressline. Constraint 12 ensures that the number of direct laborers assigned to the end of shift n equals the number of direct laborers who begin shift $n + 1$; this ensures that the production schedule will flow properly. Constraint 13 ensures that the schedule is cyclic by stipulating that the number of direct laborers required at the end of the last shift is equal to the number of direct laborers required at the beginning of the first shift. Constraints 14 and 15 enforce the workforce allocation given by T&P; these constraints limit which aggregate labor shift schedules can be assigned to shifts based on the number of laborers required. Constraint 16 enforces the fact that there should be at least one changeover for each product that the aggregate product represents in the schedule. Note that this formulation has no objective because it is a feasibility problem within T&P.

Acknowledgments

We would like to acknowledge the contribution from many people at Ford Stamping, Material Planning and Logistics, and Information Technology (IT) for their continual support of this project. Specifically, we would like to acknowledge Edward Zvoch, James Higgins, Craig Morford, Mark Catri, and Robert Quick from Stamping, Kasey Kasemodel and Serguei Vassiliev from IT, and Giuseppe Rossi and Erica Klampfl from Ford Research and Advanced Engineering, for their invaluable contributions. We also gratefully acknowledge the work of Kamal Aman and Willie Cheong at the University of Waterloo in implementing the test version of the code. This work was supported by the Ford–University of Michigan Alliance, the National Science Foundation, and the Natural Sciences and Engineering Research Council of Canada.

References

Appelgren LH (1969) A column generation algorithm for a ship scheduling problem. *Transportation Sci.* 3(1):53–68.

Armacost A, Barnhart C, Ware K (2002) Composite variable formulations for express shipment service network design. *Transportation Sci.* 35(1):1–29.

Barlatt A, Cohn A, Guisikin O (2010) A hybridization of mathematical programming and dominance-driven enumeration for solving shift-selection and task-sequencing problems. *Comput. Oper. Res.* 37(7):1298–1307.

Barlatt A, Cohn A, Guisikin O, Fradkin Y, Morford C (2008) Using composite variable modeling to achieve realism and tractability in production planning: An example from automotive stamping. *IIE Trans.* 41(5):421–436.

Barnhart C, Farahat A, Lohatepanont M (2009) Airline fleet assignment with enhanced revenue modeling. *Oper. Res.* 57(1): 231–244.

Barnhart C, Kniker T, Lohatepanont M (2002) Itinerary-based airline fleet assignment. *Transportation Sci.* 36(2):199–217.

Caraffa V, Ianes S, Bagchi T, Sriskandarajah C (2001) Minimizing makespan in a blocking flowshop using genetic algorithms. *Internat. J. Production Econom.* 70(2):101–115.

Cohn A, Barnhart C (2006) Composite-variable modeling for service parts logistics. *Ann. Oper. Res.* 144(1):17–32.

Cohn A, Root S, Wang A, Mohr D (2007) Integration of the load matching and routing problem with equipment balancing for small package carriers. *Transportation Sci.* 41(2):238–252.

Crainic T, Rousseau K (1987) The column generation principle and the airline crew scheduling problem. *INFOR* 25(2):136–151.

Gusikhin O, Klampfl E (2012) JEDI: Just-in-time execution and distribution information support system for automotive stamping operations. Armbruster D, Kempf KG, eds. *Decision Policies for Production Networks* (Springer, London), 119–142.

Gusikhin O, Rossi G (2004) Improving automotive supplier operations through information logistics. Sandkuhl K, Smirnov A, Weber H, eds. *The Knowledge Gap in Enterprise Information Flow* (Jonkoping University, Jonkoping, Sweden), 81–90.

Gusikhin O, Rossi G (2005) Well-connected: MES data integration in the automotive supply chain. *APICS Performance Advantage* 15(2):32–35.

Shingō S (1996) *Quick Changeover for Operators: The SMED System* (Productivity Press, Portland, OR).