

SANDIA REPORT

SAND99-2095

Unlimited Release

Printed December 1999

The Pantex Process Model: Formulations of the Evaluation Planning Module

Dean A. Jones, Craig R. Lawton, George F. List, and Mark A. Turnquist

Prepared by

Sandia National Laboratories

Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under Contract DE-AC04-94AL85000.

Approved for public release; further dissemination unlimited.



Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States
Department of Energy by Sandia Corporation.

NOTICE: This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government, nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, make any warranty, express or implied, or assume any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represent that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government, any agency thereof, or any of their contractors or subcontractors. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government, any agency thereof, or any of their contractors.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Prices available from (703) 605-6000
Web site: <http://www.ntis.gov/ordering.htm>

Available to the public from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Rd
Springfield, VA 22161



SAND99-2095
Unlimited Release
Printed December 1999

The Pantex Process Model: Formulations of the Evaluation Planning Module

Dean A. Jones
Craig R. Lawton
Critical Infrastructure Surety Department
Sandia National Laboratories
P.O. Box 5800
Albuquerque, NM 87185-0451

George F. List
Rensselaer Polytechnic Institute
Troy, NY 12180

Mark A. Turnquist
Cornell University
Ithaca, NY 14853

Abstract

This paper describes formulations of the Evaluation Planning Module that have been developed since its inception. This module is one of the core algorithms in the Pantex Process Model, a computerized model to support production planning in a complex manufacturing system at the Pantex Plant, a US Department of Energy facility. The model reflects the interactions of scheduling constraints, material flow constraints, and the availability of required technicians and facilities.

(blank page)

Contents

INTRODUCTION AND PROBLEM SETTING.....	1
PROBLEM DEFINITION AND SOLUTION STRATEGIES	12
PROBLEM FORMULATION.....	13
FIRST AND SECOND FORMULATIONS	14
THIRD FORMULATION.....	15
Formulation Overview	15
Converting Time Period Decisions into Monthly Decisions.....	22
FOURTH FORMULATION.....	24
Problem Size Reduction	27
More Ties Between Task Start Times and Task Activity.....	28
FUTURE EFFORTS	28
REFERENCES.....	30
APPENDIX A	31

Figures

Figure 1: Example Gantt chart.....	10
Figure 2: Parent and daughter jobs	14

Figure 3: ν-variable model solution process	15
Figure 4: Task separations.....	17
Figure 5: Dispersion of task activity.....	20
Figure 6: Iterative solution procedure	21
Figure 7: Task timings and time periods	25
Figure 8: Determining g_{jt}.....	26

Tables

Table 1: Selected EPM Jobs and Tasks.....	9
Table 2: Resource availability and demand by time period.....	11

The Pantex Process Model: Formulations of the Evaluation Planning Module

Introduction and Problem Setting

Sandia National Laboratories has developed and implemented a decision support tool called the *Pantex Process Model*, or PPM (Kjeldgaard, et al, 1998). The PPM is a computerized model that supports planning and scheduling activities at Pantex, a US Department of Energy production plant in Amarillo, Texas. Pantex is responsible for three major DOE programs – nuclear weapons disposal, stockpile evaluation, and stockpile maintenance – using shared facilities, technicians, and equipment. The PPM employs modern management science techniques to optimize production planning and scheduling in the complicated production system at Pantex.

This report describes the evolution of the Evaluation Planning Module (EPM), one of the core algorithms in the PPM. Various formulations of the EPM have been developed since the module's inception. Descriptions of these formulations, especially the “ v -variable” and “ s -variable” realizations, are followed by an example application. The report closes with thoughts about further avenues to explore.

In stockpile evaluation, weapons are brought to Pantex from the active arsenal. There they are partially disassembled, tested, re-assembled, and returned to the active stockpile. A number of these weapon evaluations occur in each year. They are few compared with the number of weapons that are disassembled, but the resources consumed, facilities and technicians, is substantial.

Overall, these evaluations fit a “job shop” paradigm rather than a “flow shop.” Each evaluation is like a miniature project with tasks that form a tree-like network. Some tasks have earliest allowable start times (e.g., weapon arrival) while others have latest possible finish times (e.g., availability of a weapon for a particular test). Some tasks require weeks while others are only a few hours. The task sequence in the parent job results in returning the weapon to the active stockpile. At some junctures during this task sequence, “daughter” jobs are spawned. These daughter jobs have their own, independent task sequences and requirements.

Resource assignments are a major issue, especially since many tasks have long durations and require scarce resources. Facilities and technicians are the primary focus, although tools and fixtures are also key. Each task needs a particular combination of facility type and technician certification. Hierarchical substitution exists within the facilities, which means facilities of greater capability can be used in less demanding ways. Scarcity plays a role in determining which facilities should be assigned to what tasks. Technicians hold multiple certifications. And here, too, scarcity plays a role. Those certifications less

prevalent in the labor pool must be assigned carefully to ensure that all tasks are completed.

The problem setting can be made clearer by providing an illustrative setting derived from those that typically arise. It is based on a 62/87 data set used for model development that includes 67 jobs and 130 tasks. The tasks involved require 11 certifications and 6 facility types.

Table 1: Selected EPM Jobs and Tasks

Jobs Description	Task Description	Dur	EAST	LAFT	Crew Size	Certification	Facility Type
06200105-068	62 SFT D&I MECH	80			2	62-00/D & I	(62,87)Bay w/ task
06200105-068	Vacuum Chamber	4			2	ALL/VACUUM	Vacuum Chamber
06200105-068	X-Ray	3			2	ALL/LINAC	X-Ray (LINAC)
06200105-068	62 SFT D&I Bore Down	40			2	62-00/BORE DO	(62,87)Bay w/ task
06200105-068	62 SFT D&I CELL	40		9/11/98	2	62-00/D & I	Cell
06200121-049	62 JTA-3 ASSY	560		2/10/98	2	62-00/N E L A	(62,87)NELA Bay
06200121-050	62 JTA-3 ASSY	560		9/30/98	2	62-00/N E L A	(62,87)NELA Bay
06200200-256	Vacuum Chamber	4			2	ALL/VACUUM	Vacuum Chamber
06200200-256	X-Ray	3			2	ALL/LINAC	X-Ray (LINAC)
06200200-256	62 SLT D&I Bore Down	40			2	62-00/BORE DO	(62,87)Bay w/ task
06200200-256	62 SLT D&I CELL	40		9/28/98	2	62-00/D & I	Cell
06200200-257	Vacuum Chamber	4			2	ALL/VACUUM	Vacuum Chamber
06200200-257	X-Ray	3			2	ALL/LINAC	X-Ray (LINAC)
06200200-257	62 SLT D&I Bore Down	40			2	62-00/BORE DO	(62,87)Bay w/ task
06200200-257	62 SLT D&I CELL	40		9/30/98	2	62-00/D & I	Cell
06200200-258	Vacuum Chamber	4			2	ALL/VACUUM	Vacuum Chamber
06200200-258	X-Ray	3			2	ALL/LINAC	X-Ray (LINAC)
06200200-258	62 SLT D&I Bore Down	40		9/18/98	2	62-00/BORE DO	(62,87)Bay w/ task
06200200-258	62 SLT D&I CELL	40			2	62-00/D & I	Cell
06200200-259	Vacuum Chamber	4	9/12/98		2	ALL/VACUUM	Vacuum Chamber
06200200-259	X-Ray	3			2	ALL/LINAC	X-Ray (LINAC)
06200200-259	62 SLT D&I Bore Down	40			2	62-00/BORE DO	(62,87)Bay w/ task
06200200-259	62 SLT D&I CELL	40			2	62-00/D & I	Cell
06200200-260	Vacuum Chamber	4			2	ALL/VACUUM	Vacuum Chamber
06200200-260	X-Ray	3			2	ALL/LINAC	X-Ray (LINAC)
06200200-260	62 SLT D&I Bore Down	40			2	62-00/BORE DO	(62,87)Bay w/ task
06200200-260	62 SLT D&I CELL	40			2	62-00/D & I	Cell
06200213-033	62 TB ASSY	40		11/12/97	1	62-00/TESTBED	(62,87)NELA Bay

Table 1 describes nine of the 67 jobs in the data set and presents the job numbers and the corresponding tasks complete with resource/labor requirements and feasible task timings.

For example, job 06200105-068 consists of five tasks. The first has a duration (DUR) of 80 hours, requires 2 people holding certification 62-00/D&I, and a (62,87) Bay with task exhaust. The second task, called "vacuum chamber," has a duration of 4 hours, requires 2 people with an ALL/VACUUM certification, and a vacuum chamber (unique facility type). The fifth and final task has a latest allowable finish time (LAFT) of 9/11/98.

A review of the nine jobs in Table 1 shows that the shortest task is 3 hours long and the longest is 560 hours. ALL/VACUUM and ALL/LINAC are certifications frequently required, as well as 62-00/D&I and 62-00/BORE DO. The facility types frequently required are (62,87)Bay w/task, Cell, Vacuum Chamber, and X-Ray (Linac).

Some jobs also have tasks with earliest allowable start times (EAST). This is true for the first task in job 06200200-259. An EAST is frequently associated with the expected arrival date for a weapon. In the 62/87 data set 12 tasks have EASTs and 59 have LAFTs.

Precedence relationships produce additional, implied EASTs and LAFTs. These can be computed through a pre-processing step prior to execution of the planning algorithm. Tasks that have no EAST or LAFT implied or otherwise, are assigned an EAST equal to the start date and a LAFT equal to the end date.

Figure 1: Example Gantt Chart

#	Jobs and Tasks	Q3 '97		Q4 '97		Q1 '98			
		Sep '97	Oct '97	Nov '97	Dec '97	Jan '98	Feb '98	Mar '98	Apr '98
1	06200105-068 [700646]		█						
2	62 SFT D&I MECH		█						
3	Vacuum Chamber		◆						
4	X-Ray		◆						
5	62 SFT D&I Bore Down		█						
6	62 SFT D&I CELL		█						
7	06200121-049 [320]		█	█	█				
8	62 JTA-3 ASSY		█	█	█				
9	06200121-050 [321]								
10	62 JTA-3 ASSY								
11	06200200-256 [190974]		█						
12	Vacuum Chamber		◆						
13	X-Ray		◆						
14	62 SLT D&I Bore Down		█						
15	62 SLT D&I CELL		█						
16	06200200-257 [218340]		█						
17	Vacuum Chamber		◆						
18	X-Ray		◆						
19	62 SLT D&I Bore Down		█						
20	62 SLT D&I CELL		█						
21	06200200-258 [256059]		█						
22	Vacuum Chamber		◆						

The solution algorithms produce outputs like those depicted in Figure 1 and Table 2. Figure 1 displays a Gantt chart showing the timing of individual tasks, and the implications for overall jobs. Table 2 shows resource availability, demand (usage) and shortage across a 12-month timeframe. In Figure 1, for example, job 06200106-068 spans the month of October '97, and comprises 5 tasks. The first task, Task 62 SFT D&I MECH starts at the beginning of October, and task 62 SFT D&I CELL, the last task, ends at the end of October. By comparison, job 06200121-049 comprises one task, 62 JTA-3 ASSY, which spans more than three months (560 hours based on Table 1). It begins in October and finishes in January.

Table 2 shows that the (62,87) NELA Bay is fairly heavily used. Its availability averages about 330 hours per month (varying up and down based on the number of Pantex workdays available) and its utilization or demand peaks at 215.5 hours in the first month. Months 3, 9 and 10 have the next highest use levels (176 hours) followed by months 2 and 8. In months 5, 6, 7, and 12 there is no demand. The other facilities see lighter usage, and no shortages exist in any month for any facility.

Table 2: Resource availability and demand (in hours) by time period

(62,87)Bay w/ task	Availability	704	576	704	672	640	704	672	640	704	704	672	672
(62,87)Bay w/ task	Demand	320	0	0	0	0	0	0	0	0	0	0	0
(62,87)Bay w/ task	Shortage	0	0	0	0	0	0	0	0	0	0	0	0
(62,87)NELA Bay	Availability	352	288	352	336	320	352	336	320	352	352	336	336
(62,87)NELA Bay	Demand	215.5	144	176	64.5	0	0	0	127.5	176	176	80.5	0
(62,87)NELA Bay	Shortage	0	0	0	0	0	0	0	0	0	0	0	0
Cell	Availability	563.2	460.8	563.2	537.6	512	563.2	537.6	512	563.2	563.2	537.6	537.6
Cell	Demand	240	0	0	0	0	0	0	0	0	0	0	0
Cell	Shortage	0	0	0	0	0	0	0	0	0	0	0	0
Cell with task and 220	Availability	0	0	0	0	0	0	0	0	0	0	0	0
Cell with task and 220	Demand	0	0	0	0	0	0	0	0	0	0	0	0
Cell with task and 220	Shortage	0	0	0	0	0	0	0	0	0	0	0	0
Cell with task, H2O, 220	Availability	0	0	0	0	0	0	0	0	0	0	0	0
Cell with task, H2O, 220	Demand	0	0	0	0	0	0	0	0	0	0	0	0
Cell with task, H2O, 220	Shortage	0	0	0	0	0	0	0	0	0	0	0	0
Vacuum Chamber	Availability	176	144	176	168	160	176	168	160	176	176	168	168
Vacuum Chamber	Demand	24	0	0	0	0	0	0	0	0	0	0	0
Vacuum Chamber	Shortage	0	0	0	0	0	0	0	0	0	0	0	0
X-Ray (LINAC)	Availability	176	144	176	168	160	176	168	160	176	176	168	168
X-Ray (LINAC)	Demand	18	0	0	0	0	0	0	0	0	0	0	0
X-Ray (LINAC)	Shortage	0	0	0	0	0	0	0	0	0	0	0	0

It is important to note that this is a plan and not a schedule. Although Figure 1 may look like a schedule, it is really a plan. The precise times at which tasks are to start are not specified (e.g., to the nearest hour) nor are the exact facilities or technician assignments specified. These are decisions that would be made to convert the plan into a schedule. Schedule development also requires input from the factory floor, nominally on a real-time basis, so that decisions can be tied to the current status of all workstations when

updating the schedule. These data are not currently part of the inputs to the EPM. Hence Figure 1 is providing guidance to the production manager about how to organize the tasks, but it is not specifying precisely how those tasks should be scheduled.

Problem Definition and Solution Strategies

Minimize: the resource costs associated with scheduling all evaluation tasks

Subject to these constraints:

- 1) *Scheduling tasks within the limits of their earliest allowable start times and latest allowable finish times.*
- 2) *Ensuring that task precedence is satisfied.*
- 3) *Meeting all needs for appropriate facility and technician-certification resources.*

The resource costs are derived from two sources: 1) penalties for using scarce resources in less-than-highest-and-best-use ways (e.g., technicians with scarce certifications being assigned to use less scarce certifications, or a cell being used as a bay) and 2) penalties for having to use resources that do not actually exist (e.g., time periods when the demand for certain resources exceeds the supply available even under an optimal strategy).

The main choice variables are the starting times for the tasks. Setting values for these variables produces resource demands and results in the resource costs reflected in the objective function. Hence, the optimal solution involves a set of task start times which minimize the total penalty cost while satisfying constraints #1-#3.

A classic approach toward formulating such a “job shop” problem is to treat the task start times as Boolean variables. For example, as $x_{jt} \in \{0,1\}$ indicates whether task j begins in time period t or not. The time periods are short relative to the task durations so that every task is effectively an integer number of time steps in length. This produces a problem formulation which is mixed integer and linear, but it has a large number of Boolean variables. For example, if there are 1000 tasks (typical size for the EPM) and time is divided into half-days (which could be argued is required since the span of task durations ranges from 3-4 hours to 560), and the average span of time over which a task can start is 20 days, then 40,000 Boolean variables would be involved. (If tasks were actually being scheduled to the nearest hour the time periods would be even finer, which would make the problem 8-24 times larger.)

Problems of this size cannot be solved exactly in a reasonable amount of time. An approach other than “direct attack” is needed to obtain good answers quickly. Heuristics are most often employed. One option is to create a linear relaxation of the discrete problem and create a process whereby the upper and lower bounds on the true value of

the objective function are brought to convergence over a sequence of iterations. Another is to keep the discrete choice variables and use a search technique (e.g., genetic algorithm) to find progressively better solutions.

Neither of these approaches has been employed in the EPM. Instead, alternate formulations of the problem have been developed that contain no discrete variables. In the one case, time is left continuous, and the task start times are used as choice variables. A non-linear statement of the problem results. In the second, continuous variables are used to indicate the progress toward completion made on a given task during a given time period.

These two formulations produce related but different models. In the first, s_j is the start time for task j and takes on a specific value ranging from the earliest allowable start time (input or computed based on precedence) to the latest allowable start time (given by the latest allowable finish time minus the duration). Constraints like $s_{j+1} - s_j \geq d_j$, where d_j is the duration of task j , ensure precedence requirements are met. In the second, variable v_{jt} indicates the progress toward completion achieved on task j during time period t and constraints like $\sum v_{jt} = 1$ ensures that task j is completed. This paper describes these two formulations used in the EPM, the differences between them, and the implications those differences have in terms of solution methodologies and results interpretation.

Problem Formulation

Figure 2 shows a typical evaluation job. Tasks $P-1$ through $P-6$ represent the *parent* job, while tasks $D-1$ through $D-4$ represent a *daughter* job. Daughter jobs begin only after their enabling task is completed in the parent job. In this case, Task $D-1$ can commence only when $P-3$ is complete. Each task has a duration, depicted by the width of the boxes, which can be as short as an hour or as long as several months. In addition, each task can have an *earliest allowable start time* (EAST) and a *latest allowable finish time* (LAFT). The first task in a parent job often has an EAST that is tied to the arrival of the weapon. The task for the test itself often has a LAFT because the test has to be conducted on or before a certain date to avoid tying up external resources (e.g., off-site engineers). Each task needs a specific facility type (e.g., a Task Bay with 220) and a qualified crew (e.g., 2–3 people holding a specific certification). The result is a large scale mixed-integer optimization problem that is difficult to solve and does not fit a standard job-shop scheduling framework.

The EPM determines when these evaluation tasks should be scheduled during a given analysis time frame. Inputs for the EPM result in two sets of data. The first set pertains to the jobs: the tasks involved, their precedence relationships, and

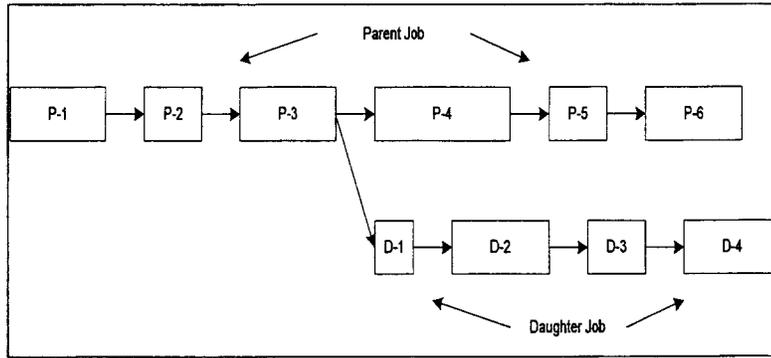


Figure 2: Parent and daughter jobs

EASTs and LAFTs. The second set pertains to facilities and personnel resources . Facilities resource means the type, daily availability, and alternate configurations to which a facility can be adapted. Personnel resource means daily availability and current certifications.

The output is correspondingly of two types. The first pertains to the tasks: their start times and level of activity by time period. This information can be reviewed in both Gantt chart and tabular form. The second type of output corresponds to the resources. For the facilities, this includes the number of hours of use, by configuration option and time period. For the personnel, it includes hours of use, by certification and time period.

Typically, an EPM problem spans a year and involves upwards of 500 jobs and 1000 tasks. Each job has from one to six tasks. About 28 facility types are involved along with 300 technicians, each of whom holds 2–3 of the 80 possible certifications.

First and Second Formulations

The first formulation of the EPM used LINGO to implement a mixed integer linear representation of the problem (see Appendix A). This model's critical component was a set of binary choice variables that indicated whether task j was to commence in time period t or not. The formulation was satisfactory for small problems, but for actual data sets it required long solution times or never reached a feasible solution.

The second formulation was based on a recursive heuristic created by Bell and Han (1991). In it, the makespan of a resource constrained large-scale project scheduling problem is minimized by finding the most judicious set of precedence arcs to add between tasks competing for the same resource. Low priority tasks are deferred. For example, if n tasks want to use resource k in time period t but only $m < n$ tasks can be scheduled given the amount of resource k available, then the procedure seeks to defer a task, say task s , until after one of the higher priority tasks has ended. It does this by adding a new precedence arc between the end of the higher priority task, say task r , and the beginning of task s . Relative priorities dictate the choice of tasks r and s . If deferring

task s does not bring demand in line with supply, then more tasks are deferred. This means more precedence arcs are added between new tasks r and s . The procedure works well on small problems, but has difficulty with realistic ones. It becomes slow, the rules for selecting tasks r and s become surprisingly complex and cumbersome, and the solutions have an unknown optimality. The procedure is also ill suited to address issues that have become important recently, such as facility substitutions when resources are in short supply and the selection of specific technicians for each task. Moreover, since the procedure always pushes tasks off into the future, resource surpluses are created in early time periods as the combination of deferred tasks unfolds and affects other tasks. Put another way, the heuristic has difficulty looking both backwards and forwards in time when seeking options for scheduling tasks.

Third Formulation

The third formulation of the EPM is the one currently implemented in the PPM (v. 3). It has become known as the **v -variable formulation** because of its use of the variables v_{jt} . v_{jt} indicates the level of activity for task j in time period t . These variables completely replace the Boolean variables in the MIP formulation (see Appendix A) that specify whether task j begins in time period t or not. It also addresses the issues that could not be accommodated in the Bell and Han-related formulation: facility substitutions and the assignment of specific technicians to each task.

Formulation Overview

As Figure 3 depicts, this formulation of the problem is coupled to a process in which the MIP problem is transformed into a linear representation, an optimal solution to that linear programming problem is obtained, and that solution is then

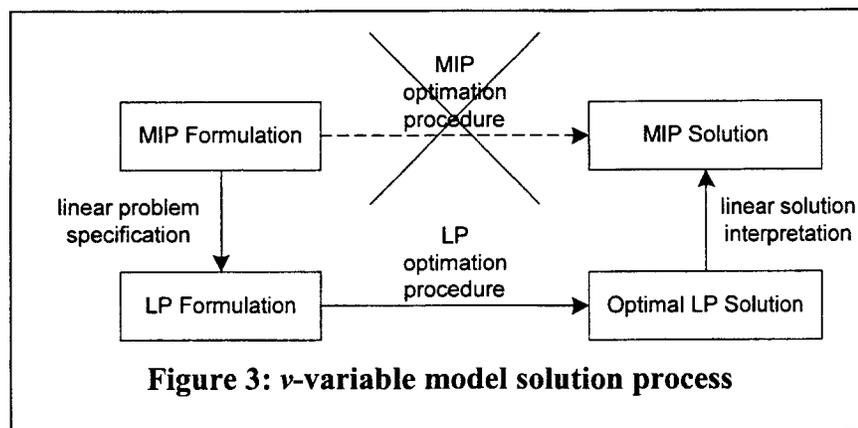


Figure 3: v -variable model solution process

interpreted back into the context of an MIP solution. During the linear program (LP) optimization step, an iterative procedure forces the v -variable solution to converge to an MIP-like solution. At each iteration, a non-linear, multi-objective programming problem is solved, using CPLEX as the solution engine.

The formulation involves the following framework:

- J tasks, indexed $j = 1, 2, \dots, J$ (with task J being a termination task)
- A precedence relationships, indexed $a = 1, 2, \dots, A$
- K facility types, indexed by either $k = 1, 2, \dots, K$, or $i = 1, 2, \dots, K$
- E technicians, indexed by $e = 1, 2, \dots, E$
- C certifications, indexed by $c = 1, 2, \dots, C$
- T periods, indexed by $t = 1, 2, \dots, T$.

It employs seven sets of choice variables:

v_{jt}	=	percent of activity for task j occurring in period t
σ_j	=	amount of dispersion in the activity pattern for task j
x_{ikt}	=	number of hours that type i facilities are used as type k in period t
y_{ect}	=	number of hours that technician e uses certification c in period t
f_{kt}	=	number of hours of shortage for type k facilities in period t
q_{ct}	=	number of hours of shortage for certification c in period t
Q_a	=	number of hours of precedence violation for precedence arc a

and the following inputs:

h_j	=	number of technicians (crew size) required for task j
M_{it}	=	hours of availability for facility type i in period t
H_{et}	=	hours of availability for technician e in period t
P_a	=	set of all predecessor tasks in precedence relationship a
S_a	=	set of all successor tasks in precedence relationship a
J_k	=	set of tasks which require facility type k
F_k	=	set of facility types which can be used as a facility type k
O_i	=	set of facility types for which facility type i can be configured
L_c	=	set of tasks requiring certification c
E_c	=	set of technicians which have certification c
C_e	=	set of certifications held by technician e

The first constraint guarantees that each task is accomplished:

$$\sum_t v_{jt} = 1 \quad \forall j \quad (1)$$

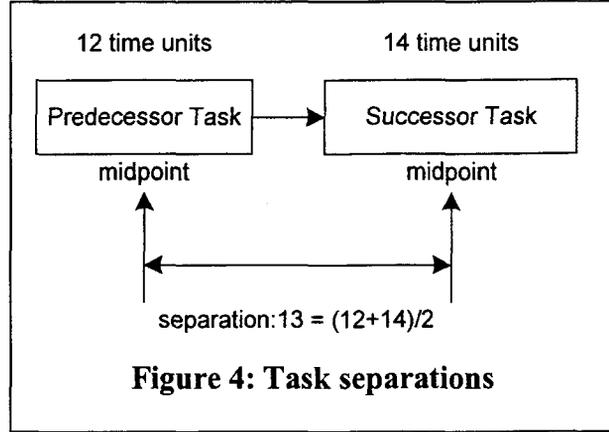
The second ensures that the precedence relationships between tasks (including parent-daughter relationships) are enforced based on the midpoints of the tasks:

$$\sum_t t v_{jt} - \sum_t t v_{it} \geq \frac{1}{2}(d_i + d_j) + Q_a \quad \forall a \in A, i \in P_a, j \in S_a \quad (2)$$

To show how this constraint affects tasks i and j , start with the terms on the left-hand side. The first term computes the midpoint for task j . (This is equivalent to computing an expected value for the midpoint.) The second term computes the midpoint for task i . The

right-hand side then specifies the minimum separation between these two tasks. Specifically, the midpoints for tasks i and j must be separated by half the duration of task j (midpoint to completion) plus half the duration of task i (from its beginning to its midpoint). Graphically, this relationship is illustrated in Figure 4.

If the predecessor task has a duration of 12 time units and the successor task 14, then the separation between their midpoints must be 13 time units ($\frac{1}{2}(12+14)$). The last term, Q_a , allows this separation requirement to be violated if an infeasible solution would otherwise result. (This is to accommodate a situation where the user has specified conflicting EAST and LAFT values.)



The third constraint prescribes the demand-supply relationships for facilities:

$$\sum_{j \in J_k} d_j v_{jt} = \sum_{i \in I_k} x_{ikt} + f_{kt} \quad \forall k, t \quad (3)$$

The term on the left-hand side computes the total facility-hours of configuration type k needed in time period t . (Tasks that belong to set J_k need a facility of configuration type k .) On the right hand side, the first term computes the number of type k facility-hours supplied by facility types $i \in I_k$. Any facility that is a member of set I_k can be configured as a type k facility. x_{ikt} is the number of facility hours of configuration type k being provided during time period t by facilities of type i . The second term, f_{kt} , represents the shortage in facility hours of configuration type k needed to meet the demand in time period t if the actual facilities are not sufficient.

A bit more discussion about f_{kt} is useful. This variable ensures that the model can always find a feasible solution. In the absence of f_{kt} , if facility resources were insufficient to meet demand, the model would conclude that there was “no feasible solution.” With f_{kt} included, it can always be used to close the gap between supply and demand. Moreover, when f_{kt} is non-zero, there is feedback about mismatches between resource demand and supply. Of course, the use of nonexistent resources is penalized more heavily than almost any other action the model can take. So shortages are not created artificially. A significant incentive exists to leave the shortage variables at zero if at all possible.

It is also important to reinforce the difference between facility configuration and type. A facility’s *type* is how it is classified (e.g., as a Cell); its *configuration* is how it is being used (e.g., as a Task Bay with 220). Most *types* of facilities have subordinate (lesser) *configurations*. When configuration demand exceeds supply, such facilities can be used

in these lesser configurations. For example: a cell can be used as a bay with 220 and H₂O, a bay with 220, a bay with H₂O, or a bay with neither 220 nor H₂O. The cost for such use is the wasted functional capability. The benefit is that the facility is not idle. Such use also provides guidance for future capital investments. Insofar as constraint (3) is concerned, the term on the left-hand side computes the demand for facility hours of configuration k while the terms on the right determine how this demand will be met, either by using real facilities ($\sum x_{ikt}$) or allowing a shortage (i.e., $f_{kt} > 0$).

A fourth constraint ensures that the facility hours supplied by the real facilities are not greater than the total number of facility hours available:

$$\sum_{k \in O_i} x_{ikt} \leq M_{it} \quad \forall i, t \quad (4)$$

For certification hours and people, similar relationships are needed. Constraint (5) relates the demand for certification hours (across tasks $j, j \in L_c$) to the hours that can be supplied by technicians having the desired certification ($e \in E_c$):

$$\sum_{j \in L_c} h_j d_j v_{jt} = \sum_{e \in E_c} y_{ect} + q_{ct} \quad \forall c, t \quad (5)$$

Again, the gap between demand and supply can be closed using the shortage variable q_{ct} as it was for f_{kt} which was previously discussed.

Constraint (6) ensures that the total hours supplied by technician e in time period t do not exceed the hours available, H_{et} :

$$\sum_{c \in C_e} y_{ect} \leq H_{et} \quad \forall e, t \quad (6)$$

Finally, constraint (7) ensures that no one technician supplies more certification hours than the total operation-hours of activities that use certification c :

$$y_{ect} \leq \sum_{j \in L_c} v_{jt} \quad \forall e, c, t \quad (7)$$

Four objectives direct the search for an “optimal” solution. The first seeks to minimize the makespan for all jobs. Task J , the “terminal task,” cannot be completed until the tasks for all other jobs are finished:

$$\text{minimize} \quad z_1 = \alpha \sum_t t v_{Jt} \quad (8)$$

The second objective penalizes precedence relationship violations:

$$\text{minimize} \quad z_2 = \gamma \sum_{a \in A} Q_a \quad (9)$$

γ is always set to a large value, so that precedence requirements are violated only if the model must do so to comply with user-specified EASTs and LAFTs.

The third objective penalizes various types of resource use decisions:

$$\text{minimize} \quad z_3 = \sum_{i,k,t} \zeta_{ikt} x_{ikt} + \sum_{e,c,t} \xi_{ect} e_{ect} y_{ect} + \sum_{k,t} \delta_{kt} f_{kt} + \sum_{c,t} \varphi_{ct} q_{ct} \quad (10)$$

The first two terms penalize the use of real resources in less-than-ideal ways. For facilities (x_{ikt}) this involves the use of a facility, type i , in a lesser configuration, k . For technicians (y_{ect}), this is a user specified “cost” of using certification c for technician e in time period t .

The last two terms penalize use of the shortage variables. Nominally, δ_{kt} and φ_{ct} are large, so that real resources are used first. These coefficients also take on small values during the iterations so resource demand information can be captured.

The model’s fourth objective aims to minimize the dispersion in task activity. To achieve the functional equivalent of an integer solution this objective forces the v_{jt} ’s to be contiguous. It also ensures that task activity runs at the maximum rate possible until the task is finished. A second order nonlinear equation is used to compute the amount of dispersion in the v_{jt} ’s:

$$\text{minimize} \quad z_4 = \sum_j \sigma_j^2 \quad (11)$$

where:

$$\sigma_j^2 = \sum_t v_{jt} (t - \bar{t})^2 \quad (12)$$

and

$$\bar{t} = \sum_{\tau} \tau v_{j\tau} \quad (13)$$

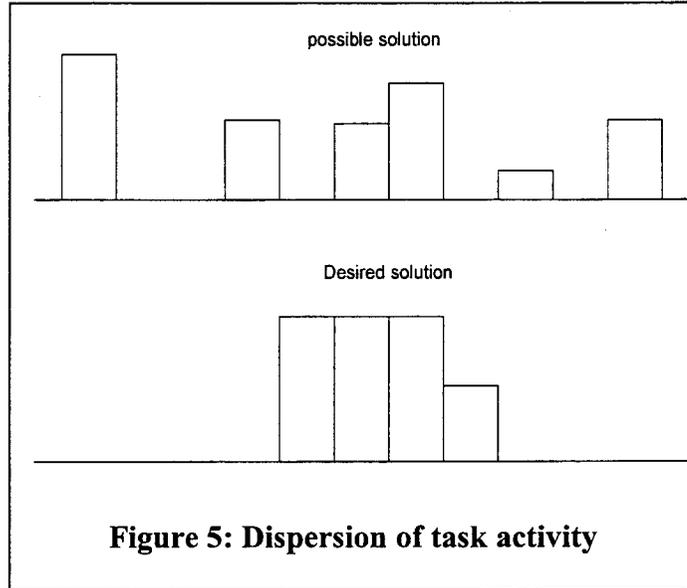
Figure 5 illustrates the intent and impact of this objective.

If little or no priority is placed on z_4 , answers like that depicted in Figure 5 can result. Task activity can vary from one time period to the next, and there can be time periods when little or no activity occurs. No constraint in the model forces the v_{jt} ’s to be contiguous, nor is there a requirement that the v_{jt} ’s take on maximal values so that the task is completed in accordance with its duration.

However, if high priority is placed on z_4 , the model compacts the task activity. Answers like the “desired solution” are produced. Task activity becomes contiguous, and the v_{jt} 's take on maximal values until the task is complete. This minimizes σ_j^2 .

Conversely, if the v_{jt} 's are dispersed and/or some v_{jt} 's take on values less than the maximum possible, then a larger-than-minimal value of σ_j^2 will result.

It is useful to note that σ_j^2 cannot be zero unless the task is shorter than a single time period. What this implies is that minimizing (11) produces the functional equivalent of an integer solution to the original problem.



Thus our goal is to find a set of task timings (v_{jt} 's) which minimize (11) while obtaining the smallest possible value for the other objectives. This is done by combining the four objectives with a set of weights θ_n that indicate the priority or importance being associated with objective n :

$$Z = \sum_{n=1}^4 \theta_n z_n \quad (14)$$

The weights and the underlying objective-specific cost coefficients are then adjusted to achieve the goal.

In summary, the overall problem involves *minimizing* (14) *subject to* (1)–(7), recognizing that (8)–(13) support (14) and that (11) must be minimized to obtain the functional equivalent of an integer solution to the problem.

Accomplishing this goal is not as simple as it might seem. A number of strategies have been explored, ranging from gradient-based search to intuitively based cost coefficient adjustments. The best approach identified so far makes use of an iterative process in which a resource leveling step aimed at objectives z_1 through z_3 is followed by a task dispersion minimization step focused on z_4 . Note that z_4 is approximated by using partial derivatives on equation (11) to determine a set of appropriate penalty weights. In between, there are updates to the cost coefficients and the task median target values. This approach is depicted in Figure 5. The resource-leveling step always comes first, and the task dispersion step is always last so that an integer solution is obtained. Six iterations are typically required to obtain a reasonable solution. Since an LP is solved during steps “a” and “b” of each iteration, cost coefficients are needed that reflect the current importance

of pursuing z_4 . Therefore a surrogate for z_4 is needed that involves cost coefficients focused on the v_{jt} variables:

$$\text{minimize} \quad \sum_{j,t} \beta_{jt} v_{jt} \quad (15)$$

At each step, an LP is solved with updated values for the cost coefficients. While the values for α (0.1) and γ (10.0) do not change, values for the rest of the coefficients do change.

For the initial iteration of step “a” the cost coefficients are set as follows:

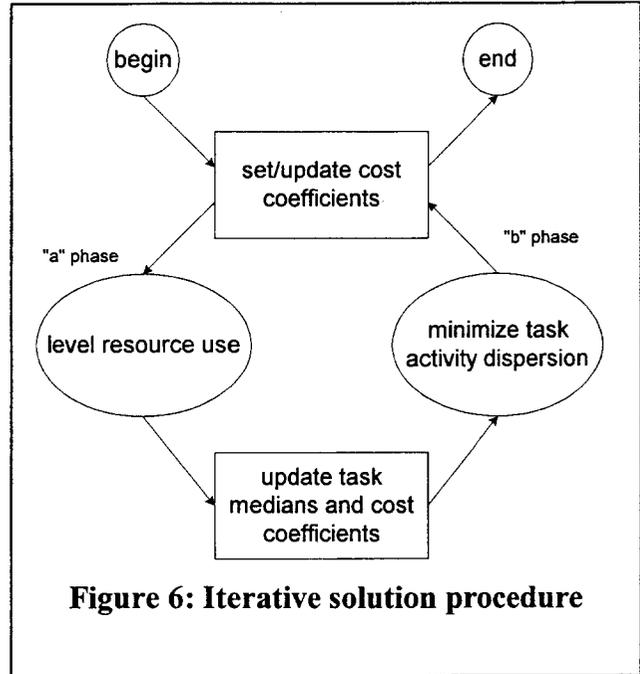
$$\begin{aligned} \zeta_{ikt} &= 0.01 + 0.01 * \pi_{ik} \\ \xi_{ect} &= 0.01 \\ \delta_{kt} &= 5.0 \\ \varphi_{ct} &= 5.0 \end{aligned}$$

where π_{ikt} is the cost of using facility type i in configuration k . The cost coefficients for the v_{jt} 's reflect the fact that earlier task activity is better:

$$\beta_{jt} = 0.01 * t$$

This produces a solution where the task timings fit within the resources available and use of the pseudo resources is minimized. As Figure 6 suggests, the task activities may not be contiguous and the v_{jt} 's may not be at the maximum values possible. This allows the task to be completed as quickly as the task duration would imply. In fact, for tasks that compete for the same resources and have identical EASTs and LAFs it is possible that the model may interlace the task activities, producing task medians that are nearly identical. Hence the motivation for step “b.”

During step “b” the focus is on minimizing the dispersion of task activity, pursuant to minimizing z_4 and obtaining an integer solution to the problem. To achieve this goal, the task medians from the “a” solution are treated as target values and the partial derivatives for equation (11) are evaluated to provide suitable cost coefficients for the v_{jt} 's. Moreover, δ_{kt} and φ_{ct} are set to minimal levels (and ζ_{ikt} and ξ_{ect} to high values), so that the shortage variables are used to record the resource demands associated with the resulting task timings.



Thus, for step “b” the cost coefficients are updated as follows:

$$\begin{aligned}
 \zeta_{ikt} &= 10.0 \\
 \xi_{ect} &= 10.0 \\
 \delta_{kt} &= 0.001 \\
 \varphi_{ct} &= 0.001
 \end{aligned}$$

and the cost coefficients for the v_{jt} 's reflect the task medians from step “a.”

$$\beta_{jt} = (t - \bar{t}_j)^2 \quad (16)$$

where:

$$\bar{t}_j = \sum_{\tau} \tau v_{j\tau} \quad (17)$$

In the second iteration, the cost coefficients for step “a” are set as follows:

$$\begin{aligned}
 \zeta_{ikt} &= 0.01 + 0.01 * \pi_{ik} + 0.01 * f_{kt} \\
 \xi_{ect} &= 0.01 + 0.01 * q_{ct} \\
 \delta_{kt} &= 5.0 \\
 \varphi_{ct} &= 5.0 \\
 \beta_{jt} &= 0.01 * t
 \end{aligned}$$

where f_{kt} and q_{ct} are the shortage variable values found in step “b” from the first iteration. Thus a situation is created where early task activity is desirable (β_{jt}) and use of the shortage variables is discouraged (δ_{kt} and φ_{ct}) but use of resources during time periods of heavy demand is discouraged (ζ_{ikt} and ξ_{ect}). The net effect is to move task activity away from time periods when resource demand is high. In the previous example of two tasks that compete for the same resources and have identical EASTs and LAFTs, this will force one of the tasks to move earlier, and the other later. It does not matter which one moves which way; both solutions are optimal.

In step “b” of the second iteration, the new task medians resulting from iteration two, step “a” are used to compute new partial-derivative-based cost coefficients for the v_{jt} 's. A new set of resource demands is obtained. This process repeats until a suitable solution is obtained. Suitability implies that the task activities are contiguous and the resource demands fit within the limits of the true resources available or minimize the demand on fictitious resources. As mentioned earlier, six iterations are typically sufficient.

Converting Time Period Decisions into Monthly Decisions

An important requirement for the third formulation is that the time periods be of equal length. Four to six days is typical. However, the breakpoints between these time periods do not necessarily match monthly boundaries. Moreover, while monthly time periods are too coarse for scheduling EPM tasks, they are critical for synchronizing resource

utilization between the EPM and the Disposal Planning Module. Thus, a second stage problem is formulated and solved to obtain EPM resource use by month.

To describe the second stage problem formulation, the same time subscript, t , is used but with the recognition that the actual length of the periods in the first and second stages are different.

Let z_{jt} be the hours of activity for task j in month t based on the medians from the first stage. For purposes of the second stage problem, these are treated as known values. They are employed to determine facility and technician use (the x and y variables) on a monthly basis. The formulation needed to establish the facility and technician utilization for each month is as follows:

$$\min \phi \sum_k \sum_t f_{kt} + \theta \sum_c \sum_t q_{ct} + \sum_i \sum_k \sum_t \pi_{ik} x_{ikt} \quad (18)$$

subject to:

$$\sum_{i \in F_k} x_{ikt} + f_{kt} = \sum_{j \in J_k} z_{jt} \quad \forall k, t \quad (19)$$

$$\sum_{k \in O_i} x_{ikt} \leq M_{it} \quad \forall i, t \quad (20)$$

$$\sum_{e \in E_c} y_{ect} + q_{ct} = \sum_{j \in L_c} h_j z_{jt} \quad \forall c, t \quad (21)$$

$$\sum_{c \in C_e} y_{ect} \leq H_{et} \quad \forall e, t \quad (22)$$

$$y_{ect} \leq \sum_{j \in L_c} z_{jt} \quad \forall e, c, t \quad (23)$$

where:

z_{jt}	=	hours of task j performed during month t
x_{ikt}	=	hours of facility type i used as facility type k during month t
f_{kt}	=	excess hours of facility type k required in month t
M_{it}	=	hours of facility type i available in month t
h_j	=	number of technicians (crew size) required for task j
y_{ect}	=	hours of technician e allocated to using certification c in month t
q_{ct}	=	excess hours of certification c required in month t
H_{et}	=	available hours for technician e in month t
J_k	=	set of tasks which require facility type k
F_k	=	set of facility types which can be used as a facility type k
O_i	=	set of facility configurations (possible uses) for facility type i
L_c	=	set of tasks which require certification c
E_c	=	set of technicians which have certification c

C_e	=	set of certifications held by technician e
π_{ik}	=	penalty (per hour) for using a facility of type i as a type k facility
ϕ, θ	=	constants

The problem defined by (18)–(23) is quite similar to the v-variable problem, but the values of z_{jt} are now considered fixed and the principal outputs are the x_{ijt} and y_{ect} values.

With a little work, the second stage problem can be transformed into a set of separable network problems, one for facilities and one for technicians in each month. So far this feature has not been exploited because a specialized network solver is being used instead. Because the largest part of the input structure for CPLEX is still available from solving the first stage problem, that structure is used again for the second stage and for solving the second stage problem as a regular LP.

Fourth Formulation

The fourth formulation is also based on math programming and avoids the use of integer variables. It derives its **s-variable** designation from the use of continuous variables s_j that designate the start time for each task j .

From a conceptual standpoint, three main features distinguish this formulation from the third. First, its time periods can have variable lengths. This means that monthly (weekly or other) boundaries can be matched precisely. To do this a set of variables, h_t , is used to indicate when time periods begin and end (as in the number of hours since the beginning of the planning horizon). Second, the principal choice variables are the starting times s_j for each task j . Third, a set of supporting variables track the relationship between the h_t 's and s_j 's. The following variables are involved:

s_j	=	start time for task j
h_t	=	start time for time period t

plus two more related directly to the tasks:

g_{jt}	=	activity for task j in time period t
d_j	=	duration of task j

A set of time difference variables is defined using the following relationships:

$$s_j = h_t - a_{jt}^+ + a_{jt}^- \quad \forall \quad i, j, t \quad (24)$$

$$s_j + d_j = h_t - b_{jt}^+ + b_{jt}^- \quad \forall \quad i, j, t \quad (25)$$

where:

$$\begin{aligned}
 a_{jt}^+ &= \text{time from when task } j \text{ starts until } h_t \text{ (} s_j < h_t \text{)} \\
 a_{jt}^- &= \text{time from } h_t \text{ until task } j \text{ starts (} h_t < s_j \text{)} \\
 b_{jt}^+ &= \text{time from when task } j \text{ ends until } h_t \text{ (} s_j + d_j < h_t \text{)} \\
 b_{jt}^- &= \text{time from } h_t \text{ until task } j \text{ ends (} h_t < s_j + d_j \text{)}
 \end{aligned}$$

As Figure 7 shows, for a given h_t , if a_{jt}^+ is non-zero, then a_{jt}^- will be zero, and similarly, if b_{jt}^+ is non-zero, then b_{jt}^- will be zero. Three hypothetical cases are shown. In case (a), task j starts and ends before h_t so a_{jt}^+ and b_{jt}^+ are both non-zero. In case (b), the obverse is true: task j starts and ends after h_t so a_{jt}^- and b_{jt}^- are both non-zero. In case (c), the task starts before h_t and ends after, so a_{jt}^+ and b_{jt}^- are non-zero.

Each s_j is constrained to be between the earliest possible start time for task j , e_j (i.e., the EAST), and the latest possible start time, $f_j - d_j$ (i.e., the LAFT minus the task duration):

$$e_j \leq s_j \leq f_j - d_j \quad \forall j \quad (26)$$

The precedence relationships among the tasks are ensured:

$$s_k + d_k \leq s_j \quad \forall j, k \in P \quad (27)$$

where s_j is the successor task, s_k is the predecessor task, and P is the set of all precedence relationships.

Tying the task timings, s_j , to the task activities by time period, g_{jt} , is accomplished through the following constraint:

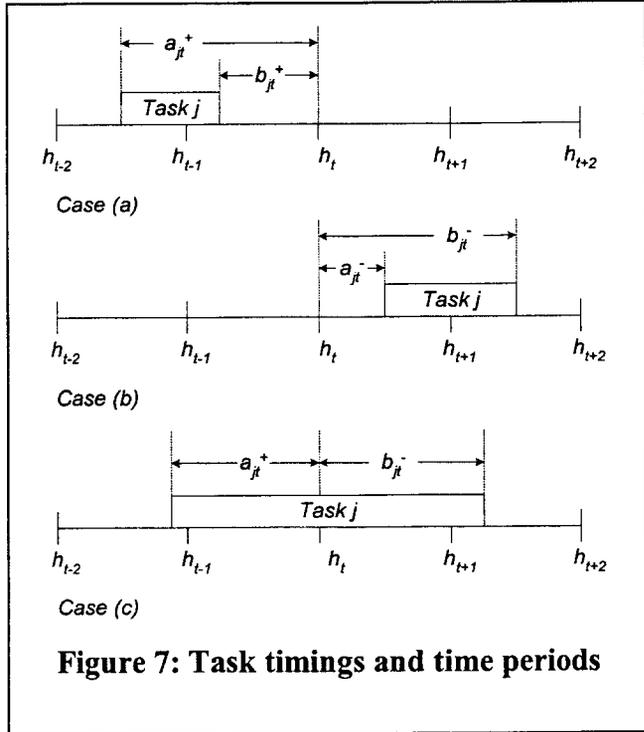


Figure 7: Task timings and time periods

$$g_{jt} = \min(d_j, a_{j,t+1}^+ - a_{j,t}^+, b_{jt}^- - b_{j,t+1}^-) \quad (28)$$

If task j begins and ends in time period t , then d_j is the upper bound on g_{jt} . That part of equation (28) is clear.

Figure 8 explains the two other terms. In the first case, task j begins before time period t and ends after time period t , so a_{jt}^+ and $a_{j,t+1}^+$ will both be positive and the value of $a_{j,t+1}^+ - a_{jt}^+$ will be the length of time period t

(i.e., $h_{t+1} - h_t$). In the second case, task j starts during time period t but continues on. Now a_{jt}^+ is zero, but $a_{j,t+1}^+$ is non-zero. Thus $a_{j,t+1}^+ - a_{jt}^+$ will equal $a_{j,t+1}^+$, which is the length of time that task j is active in time period t . In the third case, task j starts before time period t but then ends during the time period. In this instance, b_{jt}^- is non-zero, but $b_{j,t+1}^-$ is zero. Thus, $b_{jt}^- - b_{j,t+1}^-$ will equal b_{jt}^- , the length of time the task is active during time period t . In the last case, task j is active throughout time period t , so b_{jt}^- and $b_{j,t+1}^-$ are both non-zero and the difference, $b_{jt}^- - b_{j,t+1}^-$, is the duration of period t .

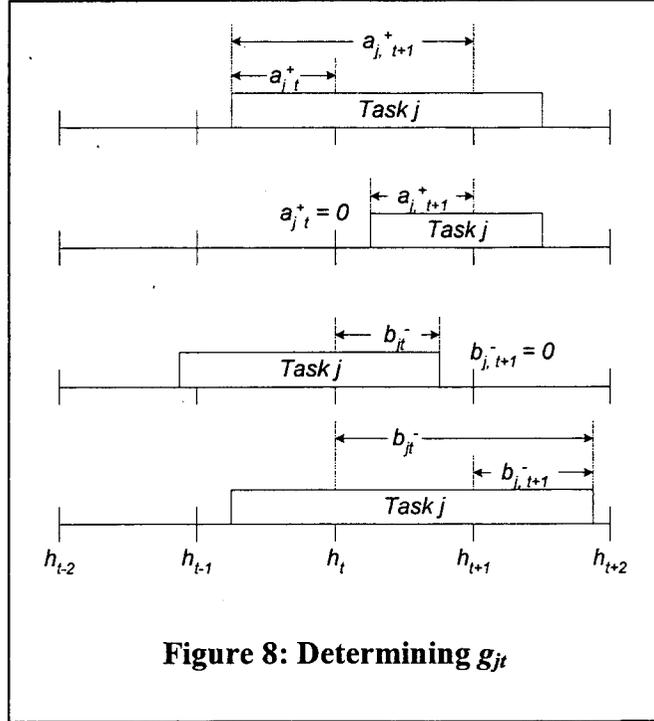


Figure 8: Determining g_{jt}

Also ensured is the g_{jt} 's sum to the duration of the task:

$$\sum_t g_{jt} = d_j \quad \forall j \quad (29)$$

Resource use is tied to task activity through constraints for facility and technician utilization. In the case of the facilities, there is:

$$\sum_{j \in J_k} g_{jt} = \sum_{i \in F_k} x_{ikt} + R_{kt} \quad \forall k, t \quad (30)$$

$$\sum_k x_{ikt} \leq M_{it} \quad \forall i, t \quad (31)$$

where:

k	=	facility configuration
i	=	facility type
x_{ijt}	=	time of facility type i used in configuration k in time period t
R_{kt}	=	demand shortage for facility configuration k in time period t
M_{it}	=	hours of facility type i available in time period t

For the technicians and certifications, there is a similar pair of constraints:

$$\sum_{j \in L_c} n_j g_{jt} = \sum_{e \in E_c} y_{ect} + Q_{ct} \quad \forall c, t \quad (32)$$

$$\sum_c y_{ect} \leq H_{et} \quad \forall e, t \quad (33)$$

where:

c	=	certification
e	=	technician
y_{ect}	=	time that technician e uses certification c in time period t
Q_{ct}	=	demand shortage for certification c in time period t
H_{et}	=	availability of technician e in time period t

Finally, the objective is to minimize:

$$z = \alpha S_J + \beta \sum_{j=1}^{J-1} s_j + \delta \sum_{j,t} (a_{jt}^+ + b_{jt}^+) + \varepsilon \sum_{j,t} (a_{jt}^- + b_{jt}^-) + \phi \sum_{i,k,t} x_{ikt} + \theta \sum_{k,t} R_{kt} + \Omega \sum_{c,t} Q_{ct} \quad (34)$$

The first term places emphasis on minimizing the makespan. The second encourages all tasks to begin as early as possible. The third and fourth terms aim to ensure that only the “+” or “-” values of the a_{jt} and b_{jt} are made non-zero in any solution. The last three terms impose penalties for 1) using facilities in subordinate configurations, 2) creating facility shortages, and 3) having certification shortages.

The model is then: *minimize* (34) subject to (24)–(33).

Problem Size Reduction

Variable substitutions can be used to reduce the size of the formulation. To expand upon (28), work occurs in conjunction with (24) and (25) to produce the following:

$$s_j - a_{jt}^- \leq h_t \quad \forall j, t \quad (35)$$

$$s_j - b_{jt}^- \leq h_t - d_j \quad \forall j, t \quad (36)$$

$$g_{jt} + a_{jt}^- - a_{j,t+1}^- \leq h_{t+1} - h_t \quad \forall j, t \quad (37)$$

$$g_{jt} - b_{jt}^- + b_{j,t+1}^- \leq 0 \quad \forall j, t \quad (38)$$

Equations (24), (25), and (28) are thereby replaced with equations (35)–(38), and the need for variables a_{jt}^+ and b_{jt}^+ is eliminated.

More Ties Between Task Start Times and Task Activity

In some situations, the fact that the a and b variables appear in equations (24) and (25) as well as (28) permits solutions where both the “+” or “-” values of the a_{jt} and b_{jt} variables are non-zero simultaneously, even though this is not true of an integral solution. This permits the s_j and g_{jt} variables to become disconnected. Equations (24) and (25) by themselves would not cause a difficulty, but equation (28), while quite necessary, confounds the situation.

Tests of various schemes for specifying the cost coefficients have not shown that an optimal solution can always be obtained in a single iteration. Additional tests with a gradient-based search procedure based on Frank and Wolfe (1956) indicate that convergence may be slow. It is also possible that a multi-step iterative procedure like the one employed in the third formulation will produce results in an expedient manner. A focus on the s_j variables, followed by the g_{jt} 's may be beneficial.

Future Efforts

Many issues are still unresolved about the EPM. Little is known about the convergence properties of the iterative procedure currently being used with the ν -variable formulation. Also, a solution methodology for the s -variable formulation is needed. These should be major focal points of the ongoing research effort.

Granularity is also a significant issue. The first reason is that granularity affects the linkage between task timings and resource utilization. In the ν -variable formulation, for example, medians that are real valued numbers are needed in order to preserve task precedence and minimize slack. But the task median equation involves multiplying task period numbers (integers) by the task activity in a given time period. Thus, to obtain non-integer task medians, task activity must exist in at least two time periods. If the granularity is coarse relative to the size of the task (e.g., a 2-hour task with 6-day time periods), this produces a misalignment between task timing and resource use. In effect, the model produces task activity in time periods when the task is not really active. When

the granularity is smaller relative to the size of the task, the misalignment is less problematic. Fortunately, most of the tasks involved with EPM jobs are as long as, or longer than, a single time period.

The second reason granularity is important has to do with the way the EPM is used. Currently, it is employed for planning purposes, with a year being the analysis timeframe. But there is no reason why the EPM could not be used for scheduling as well, with modifications to the treatment of facilities. A greater understanding needs to be gained about the relationship between granularity and the analysis timeframe being considered.

Yet a third issue with granularity is the accuracy of the plan/schedule. Finer degrees of granularity improve the temporal precision of resource assignments. It has been noted that the EPM formulation more accurately reflects the true mapping of technicians to tasks when smaller time periods are used in the model. Thus there is real incentive to use as fine a granularity as possible.

Issues related to the gaps between tasks need to be explored as well. Two issues have surfaced so far. The first is that some tasks need to be tightly coupled, with little or no slack in between, as in cases where critical elements are exposed. Currently, the model does not impose a penalty for allowing slack between such tasks. A term is probably needed in the objective function that minimizes the elapsed time between the medians of specific tasks, such as the first and last task in such time-critical sequences. The second reason gaps are important has to do with accounting for the implicit storage capacity that must be present when slack is allowed. For example, if tasks i and j are sequential, but j does not start immediately after i , the weapon must be stored somewhere during the interim. Currently, there is no account of this storage requirement. Since the model thinks task i is done, and j has not started, that weapon is not currently demanding resources. But, in fact, since the weapon is waiting for task j to start, it must be stored, and resource capacity must be used to accomplish that storage. In some cases, this may not be critical; it may be possible to store the device without any critical resource being involved (e.g., bay or cell capacity). But in other instances, this may not be the case, and a demand for resource use would be overlooked.

A final thought is that transport times are currently not modeled explicitly. It is assumed that those times are included in the task timings. For planning purposes, this may be an allowable level of detail, but if the EPM is ever used for scheduling, then transport times may need to be addressed at a finer level of detail.

From an LDRD perspective, two challenges exist. The first is to benchmark the performance of both the v -variable and s -variable formulations. The second, through extensive experimentation, is to find ways of enabling the EPM to find the best possible answers in the shortest amount of time, given the limitations of a PC platform. Thus, emphasis should first be placed on finding iterative strategies that force the LP relaxations of the IP problem to converge toward IP solutions as quickly as possible. Second, effort should be put forth to ensure that the formulation is as compact as possible

(to save machine time), yet captures all the critical variables and impacts associated with the problem.

References

Kjeldgaard, E.A., D.A. Jones, G.F. List and M.A. Turnquist (1998). *Planning and Scheduling for Agile Manufacturers: The Pantex Process Model*, SAND98-0030. Sandia National Laboratories, Albuquerque, NM.

Bell, C. E. and J. Han (1991). "A new heuristic solution method in resource-constrained project scheduling," *Naval Research Logistics*, Vol. 38, pp. 315–331.

Frank, M. and P. Wolfe (1956). "An algorithm for quadratic programming," *Naval Research Logistics*, Vol. 3, pp. 95–110.

Appendix A

This appendix contains the mixed integer programming formulation of the Evaluation Planning Module (EPM):

$$\text{Minimize:} \quad \sum_{t=e_j}^{\tau_j} t v_{jt} \quad (\text{A-1})$$

subject to:

$$\sum_{t=e_j}^{\tau_j} v_{jt} = 1 \quad \forall j \quad (\text{A-2})$$

$$\sum_{t=e_l}^{\tau_l} t v_{lt} \leq \sum_{t=e_j}^{\tau_j} (t - d_j) v_{jt} \quad \forall j, l \in P_j \quad (\text{A-3})$$

$$\sum_j \sum_{i=t}^{t+d_j-1} g_{jk} v_{ji} \leq \Gamma_{kt} \quad \forall k, t \quad (\text{A-4})$$

where:

- $d_j =$ duration of task j
- $e_j =$ earliest time for completion of task j , based on earliest possible start time for the evaluation activity of which j is a part and the precedence relationships among the tasks
- $\tau_j =$ latest time for completion of task j , based on required due dates and precedence relationships among the tasks
- $g_{jk} =$ units of resource k required per period for task j
- $\Gamma_{kt} =$ units of resource k available in period t
- $P_j =$ set of all tasks which immediately precede task j
- $v_{jt} =$ 1 if task j ends in period t ; 0 otherwise
- $J =$ index for an "imaginary" task whose completion cannot occur until all other tasks have been completed. In network terminology, this is the "sink." (Task J has no duration.) This implies that the value of t for which $v_{jt} = 1$ is the period in which all evaluation tasks are complete. This represents the **makespan** for all activities, the time at which all activities are complete.

The objective function (A-1) minimizes the makespan of all tasks. Constraint (A-2) ensures that each task is scheduled to end in one (and only one) period. The limits on the summation, e_j and τ_j , are determined prior to the optimization, based on due dates and precedence relationships among the tasks. In the terminology of critical path method project scheduling, these values are the earliest allowable start time (EAST) and latest allowable finish time (LAFT), provided the end of the task is taken as the reference point. These limits on the scheduling of each task ensure that due dates are met.

Constraint (A-3) enforces the precedence relationships among tasks. If task j is scheduled to end in period t ($v_{jt} = 1$) after a duration of d_j periods, and task l is an immediate predecessor of task j , then task l must end in period $t - d_j$ or earlier.

Constraint (A-4) calculates the resource requirements in each period and ensures that that sum is less than or equal to the total resource units available. For each task j , g_{jk} represents the units of resource k required per period, and v_{jt} indicates when the task will end. Thus, task j is active in period t if $v_{jt} = 1$ sometime between t and $t + d_j - 1$, the limits on the sum. Implicitly, constraint (A-4) assumes that if a task ends in period t , it requires g_{jk} units of resource during that period and the task ends at the conclusion of the period. [This means the formulation assumes that the periods are short relative to the duration of the tasks or that the precision of the input data (i.e., the task duration) is not high enough to be concerned about precise ending points within periods.]

Distribution

John Hudson (1)
Mason & Hanger
Pantex Plant
P.O. Box 30020
Amarillo, TX 79177

Professor George List (5)
Civil and Environmental Engineering
Rensselaer
Troy, NY 12180-3590

Professor Mark Turnquist (5)
School of Civil & Environmental Engineering
Cornell University
Hollister Hall
Ithaca, NY 14853

MS 0718 E. Kjeldgaard, 6235 (5)
MS 0451 J. Nelson, 6235 (1)
MS 0451 D. Jones, 6235 (5)
MS 0451 C. Lawton, 6235 (5)
MS 0451 S. Wright, 6235 (1)
MS 1110 C. Phillips, 9211 (2)
MS 1110 W. Hart, 9211 (2)
MS 9018 Central Technical Files, 8940-2
MS0899 Technical Library, 4916
MS 0612 Review and Approval Desk, 4912
For DOE/OSTI