AT&T BELL LABORATORIES TECHNICAL JOURNAL
Vol. 63, No. 4, April 1984
Printed in U.S.A.

A Nonlinear Zero-One Combinatorial, Goal-Programming Model and Constructive Algorithm for Solving Multiobjective Assignment Problems

By K. R. LIPSKE* and J. H. FLETCHER*

(Manuscript received July 18, 1983)

This paper describes a technique for assigning orders against specific reels of inventoried cable to ensure a high level of customer service, efficient labor utilization, highly usable cable lengths in inventory, and minimal scrap. Some orders are interrelated and thus linked, requiring that the assignment of each order be contingent upon the assignment of all linked orders. Due to this contingency, penalties associated with certain assignments, and selection allowances, the problem is quite complex. Our method of solution is formulated as a preemptive, multipriority, zero-one goal programming model used in a constructive initial placement algorithm. The model is primarily linear, but it also contains some nonlinear terms. The goal programming model and initial constructive algorithm run weekly in a production environment that handles up to 100 orders and 150 reels per problem.

I. THE PROBLEM

In 1981 Western Electric began offering 26 types of PULP Cable (Paper Insulated Telephone Exchange Cable) on a short-interval stock basis. Instead of making the cable to order as before, Western Electric would fill customer orders from reels of cable manufactured and inventoried at the stocking locations for weekly shipment. A method was required for selecting the most suitable reel of cable for filling each order while ensuring a high level of customer service, efficient

* AT&T Technologies, Inc.

labor utilization, very usable cable lengths in inventory, and minimal scrap.

Due to the manner in which this cable is installed at the job site, three specific selection requirements exist. First, only one reel of cable may be used to fill each order because the orders are for specific lengths. Henceforth, because of this length requirement, we will refer to the orders as subitems. Second, shipping less than the amount ordered is forbidden and only 10 feet of tolerance is allowed in the amount that can be shipped over the quantity ordered. This corporate shipping policy fortunately provides leeway in the selection process. Finally, in most instances, there exist several subitems for the same cable type, which must be installed at the same time. Subitems of this nature will be referred to as linked subitems in this paper. To this end, it is imperative that if any linked subitem is assigned from inventoried cable, then all subitems to which it is linked must also be assigned to some cable. Also, all linked subitems must be filled from the same stocking location. To accommodate the reader, each subitem/reel of cable possibility will be termed a real variable.

Inventoried cable is cable that has been completely tested and found to meet predetermined electrical requirements. Each time a piece of cable is cut, the newly cut ends must be tested. A 3-foot section of cable is lost in testing each end. This lost footage, henceforth referred to as the cutting allowance, must be considered in the selection process.

Because of the subitem linkage, corporate shipping policy tolerance, and cutting allowance considerations, the method of solution required to solve this assignment problem is quite complex. Since each subitem must be completely filled from a single reel of cable, or left unassigned, a method was sought to provide optimal or near-optimal results in zero-one form, where a value of 1 indicates that the subitem is assigned to the reel of cable, and 0 indicates that the subitem is not assigned to the reel of cable. The problem then is to minimize:

$$Z = \sum_{\substack{\text{ALL CONSTRAINTS} \\ k}} \quad \sum_{\substack{\text{ALL PRIORITY LEVELS} \\ l}} P_{kl}[g(d_k^-, d_k^+)]$$

subject to:

$$U \sum (c_{ij}x_{ij} + d_k^- - d_k^+) - W \sum (c_{ij}x_{ij} + d_k^- - d_k^+) + Y \sum\sum (c_{ij}x_{ij} + d_k^- - d_k^+) < 0,$$

where

P_{kl} = preemptive priority l of constraint k

$g(d_k^-, d_k^+)$ = mathematical operation on slack variables (d_k^-, d_k^+) to be minimized

c_{ij} = various constraint coefficients on real variables X_{ij}

$X_{ij} = 1$ if subitem$_j$ is assigned to reel$_i$, 0 otherwise
$U, W, Y = 0, 1$ based on whether that term is needed in the constraint type (see Appendix A for specific formulation of constraints).

II. HEURISTIC APPROACHES

Zero-one assignment problems traditionally are solved using several techniques, including exhaustive search, branch-and-bound enumeration, and integer linear programming adaptations.[1] These techniques become unusable, however, as the problem grows to large sizes. Since our problems sometimes consist of 1,000 objectives and 1,500 decision variables (for 50 subitems and 150 reels of cable), none of the traditionally small-to-medium solution techniques proved suitable when investigated.

Liggett[1] describes two types of heuristic approaches, which will lead to good if not optimal results with small-to-medium size problems. The first is the iterative improvement approach, which consists of obtaining an initial suboptimal solution by any of several methods and improving this solution in incremental steps. This method is characterized by its quick generation of an initial suboptimal solution with a subsequent step-by-step improvement, which is generally quite time-consuming. This approach has been used with a simplex goal programming algorithm with subsequent heuristic improvements.[2] We attempted to implement this technique; however, due to the many conflicting objectives in our problem, most variables possessed fractional values as a result of the simplex starting solution. Adjusting this solution to zero-one form then tended to defeat the purpose of starting with a simplex-generated solution.

The second method mentioned by Liggett[1] is constructive initial placement. Here a solution is built from scratch by adding variables to the solution set in successive steps that continually improve the results. Because zero-one form can be assumed from the start and each step improves the solution, this approach was more attractive for solving our assignment problem. Parker[3] interestingly compares various approaches of both techniques and concludes that the constructive approach of Graves and Whinston[4] is superior to the iterative improvement method in the solution to the component placement problem, which is somewhat similar but smaller than our problem. Graves and Whinston use conditional probability in their algorithm as a general-purpose enumerative technique to select the next assignment to be added to the partially constructed solution. In lieu of conditional probability, we use a look-ahead feature before generating the model equations to accord high priority to certain assignments. We also restrict certain order-to-reel assignments that could exclude other assignments that appear preferable. Our specific problem is depicted by the model generated (see the appendix formulation). The algorithm, however, is of a deterministic general-purpose nature and is much less application-oriented.

III. METHOD OF SOLUTION

Our approach to the problem was to develop an algorithm that would produce desirable solutions proceeding under the direction of a model containing the problem formulation. The technique is based on a preemptive multipriority zero-one goal programming model formulation, which is used in a constructive initial placement algorithm.

Goal programming is a multicriteria, problem-solving technique that utilizes mathematical equations and an associated priority scheme to govern its solution. Slack variables (see d_k^-, d_k^+ terms in the equations of the appendix) associated with each equation are ranked and weighted in an achievement function to govern the achievement of the desired results. It is absolutely necessary that higher-ranked objectives are satisfied before lower-ranked objectives, and that the optimality of the overall solution is maintained with higher-ranked objectives as lower-ranked objectives are solved.

The algorithm that we developed to solve this assignment problem does this and with much more flexibility than could be realized using the simplex algorithm. Our algorithm can either maximize and/or minimize variables and, in fact, with this model formulation, makes considerable use of both. Since the algorithm is void of simplex restrictions, it also can perform nonlinear optimization. In this application, we minimize the reciprocal of certain slack variables, implying that if the slack variable cannot be driven to 0, then it should be maximized.

Because we chose the constructive initial placement method, it is imperative that at each iteration, the real variable be chosen that will improve the solution most. To provide this, our algorithm uses the multiphase method of goal programming with entering variable tie logic. With tie logic, if several real variables improve the solution equally at a given priority level, then the improvement of each at the next lower priority level is used to decide which variable to bring into solution. Without tie logic in the algorithm, constructive initial placement would not provide satisfactory results.

3.1 Contingencies

Two conditions exist that involve assigning a set of related variables at the same time. Reel handling can be minimized, in part, by assigning several subitems to a single reel of cable when all or nearly all of the cable on that reel would be used. We adopted a straightforward technique described by Petersen[5] to solve this type of contingency. Let variables X_{i1}, X_{i2}, and X_{i3} represent three different subitems that in combination would use nearly all the cable on reel$_i$. The variable X_{i4} is generated to represent the assignment of all these variables on that reel. A mutual exclusivity constraint (described in the appendix, Section A.3, item a) ensures that X_{i4} will not be allowed in the solution with X_{i1}, X_{i2}, and X_{i3}. If the combination is selected, then $X_{i4} = 1$ and X_{i1}, X_{i2}, and $X_{i3} = 0$. If the combination is not selected, $X_{i4} = 0$ and X_{i1}, X_{i2}, and $X_{i3} = 0$ or 1.

The second condition, the subitem linkage contingency, is formulated in the model as an absolute set of constraint equalities. A set of constraints for three linked subitems would be:

$$\sum_{p \in R} X_{p1} - \sum_{p \in R} X_{p2} = 0 \tag{1}$$

$$\sum_{p \in R} X_{p2} - \sum_{p \in R} X_{p3} = 0, \tag{2}$$

where X_{p1}, X_{p2}, X_{p3} represent subitems 1, 2, and 3, respectively, assigned to any reel p where p is an element of the reels$_R$ of plant P.

These equality constraints must be satisfied. If any of the three subitems were assigned, then the other two subitems also must be assigned to a reel of cable at the same plant. Bringing an X_{p1} into solution, for instance, would produce an infeasible condition, unless a corresponding X_{p2} and a corresponding X_{p3} were also brought into solution. If this could not be done because of other constraints, then the assignment of subitems 1, 2, or 3 would not be allowed.

3.2 The algorithm

The algorithm incorporates two sets of real variables: (1) those in solution, designated the IS set; and (2) those in the NS set, which are out of solution but candidates for inclusion in the IS set. Using the constructive initial placement method, all variables are initially placed in the NS set and the initial value of the achievement function of the model is calculated. To determine which variable is brought into solution, the net effect on the achievement function is calculated individually for each member of the NS set and assigned to that member. This effect is determined by summing, for each member, the contribution of every slack variable of each objective that would be affected if the member were brought into solution. The NS set is then sorted by these net effects. Next, members are selected individually starting from the top of this set for "fit" within the model formulation. If the variable fits, it is removed from the NS set, placed in the IS set, a new achievement function value is calculated, and the NS set reprioritized. If a member cannot be feasibly fit, it is dismissed from the NS set and the next member is tried. This process continues until the NS set is exhausted. The feasibility of the problem is then verified by testing the equality constraints. If any equality condition is not met, all real variables with a nonzero coefficient in the violated equality constraint are removed from both the IS set and the NS set to rectify the infeasible constraint violation. The NS set is reconstructed by placing into it all variables that are not in solution or that have not been eliminated because of this or any previously rectified equality constraint violation. The new achievement function value is then calculated and all variables in the NS set are reprioritized. The entire procedure from variable fitting through ensuring equality constraint feasibility is repeated until no equality violation is found.

3.3 The model

The model for this assignment application is formulated from input of a stock cable inventory and customer orders. For each product type we consider a separate problem. There are 14 possible objective types and 12 preemptive priorities. (See the appendix for the mathematical formulation.)

1. Subitem contingency (Priority = absolute)—To guarantee timely shipments, all linked subitems must be assigned at the same time at the same stocking location.

2. Reel capacity (Priority = absolute)—The reel capacity objective type ensures that the total of subitem lengths assigned to a real plus all cutting allowances cannot exceed the length of cable on that reel.

3. Combinations (Priority = 5)—Since a reel of cable is eliminated from inventory and reel handling is greatly minimized, this objective type, which encourages a good fit of multiple subitems to a single reel of cable, is accorded high priority. To further minimize reel handling, the model encourages combinations involving the largest number of subitems rather than smaller combinations.

4. Perfect one-to-one matches (Priority = 4)—This objective type encourages perfect and close (up to 20 feet short of cable length) individual subitem length to cable length matches. The benefits are those of eliminating reels of cable and of minimizing reel handling.

5. Assignment of future subitems (Priority = absolute)—A lateness factor is calculated for every subitem. It is based on when the customer wants the cable and takes into account the amount of time required to process and ship the subitem from the time it is received. Subitems that need to be assigned in this week's processing cycle in order to arrive at the job site on time are considered current subitems. Future subitems need not be assigned yet. Hence, future subitems are not assigned unless their assignment eliminates short remnant pieces that had been generated by the assignment of current and late subitems. Subitem linkage rules also apply to future subitems.

6. Assignment of current and late subitems (Priority = 6)—To guarantee that customers are satisfied, the assignment of current and late subitems is encouraged by this objective type. These objectives are weighted by the lateness factor of the associated subitem and more consideration is given to subitems the later they become. They are also weighted by order length so that the large subitems can be placed on the large reels first, while the large reels of cable are still available.

7. Forced subitems (Priority = 6)—In emergency situations, subitems can be entered with a force code indicating that they should be assigned before all nonforced subitems. Even though the preemptive priority is the same, the weighting factor on this objective type is larger than for any nonforced subitem, hence they receive greater consideration.

8. Use of full reels (Priority = 7)—This objective type encourages using all or almost all the cable on a reel when customer orders are assigned.

9. Minimization of scrap (Priority = 8)—This objective type minimizes assignments that cause cable to be scrapped. If cable must be scrapped, it also minimizes the amount scrapped. It is the only nonlinear objective type in the formulation.

10. Remnants of just over the scrapping limit (Priority = 9)—Since few subitems are desired for an amount less than 50 feet over the scrapping limit, this objective type is formulated to prevent assignments from leaving less than this amount on any reel, thereby keeping carrying costs to a minimum.

11. Remnants of less than 500 feet (Priority = 10)—Since approximately 80 percent of the subitems are for 500 feet or more, assignments that leave remaining cable lengths less than 500 feet are not desirable. This objective type provides this.

12. Use of minimum number of reels and largest reel (Priority = 11, 12)—This objective type encourages using as few reels as possible (Priority 11). The use of the largest reel is also encouraged (Priority 12).

13. Vacation (Priority = 1, 2)—This objective type accommodates shutdowns of stocking location in a multilocation stocking environment. To avoid adversely affecting customer service due to a scheduled shutdown at a particular location, subitems will be assigned against inventories at locations that are not shut down because of vacation or other causes.

14. Transportation (Priority = 3)—Transportation charges differ depending on the distance from the stocking location to the customer job site. Overall transportation charges are minimized by this objective type.

IV. PROCESSING IMPROVEMENTS

The algorithm has undergone considerable enhancement since the initial form was coded. Because it began producing good results using the constructive initial placement technique alone, the need to perform subsequent time-consuming iterative improvements was eliminated.

After we obtained good results with the algorithm, the problem of computer core size and run time was looked at extensively. The amount of core required was quickly reduced by trimming what was unnecessary in Lee's goal programming simplex tableau.[6] Because the simplex tableau is not executed, the sparse simplex tableau was replaced by dense lists. This allowed us to handle a much larger number of variables and objectives.

Reprioritizing the members of the NS set requires the largest segment of run time. This area was improved with several enhancements. When reprioritizing each member of the NS set, we found that many members could not be feasibly fit, given the set of variables already fit. Since constructive initial placement is a "block building" approach (once a variable is fit, it will never leave the solution at any time before the feasibility of the equality constraints is tested), we enhanced the procedure by removing the members from the NS set during reprioritizing. This considerably cut the size of the NS set. For instance, after a subitem is assigned, all associations of that subitem with other reels can be dropped from the NS set. Another large reduction was realized when we decided to recalculate the effect on the achievement function only for members of the NS set that are affected by the real variable just fit.

A further reduction in the execution time was realized in the handling of future subitems. Since a future subitem is only fit to prevent a remnant cable length less than 500 feet from being generated, the futures only need to be considered after all the current and late subitems have been considered. Also, only future subitems less than 500 feet need be considered since those over 500 feet will not improve the solution.

Run time for solving all 26 problems has varied based on the weekly mix, from 30 seconds to 30 minutes in the central processing unit (CPU) on an IBM 3033.

V. OPTIMALITY CONSIDERATIONS

This algorithm is constructive initial placement in the strict sense. Each iteration is performed without knowledge of its effect on other iterations. Also, once a variable is brought into solution, it will never be removed unless equality conditions are not met. Even though each iteration improves the result, the optimum may not, of course, be obtainable without a subsequent exchange of in-solution with out-of-solution real variables. A method of variable exchanging was not implemented because, with our subitem linkage restrictions, it would call for exchanges of the magnitude of 26/15 (26 variables in-solution exchanged for 15 variables out-of-solution) or more. An exchange algorithm would require a tremendous amount of time just to decide that a 26/15 exchange was necessary. Fortunately, due to our methods of eliminating certain assignments and identifying combinations of good fit prior to model generation, suboptimal solutions have been found empirically in only 2 percent of the solutions.

VI. EASE OF USE

This method of solution comprises the model and the algorithm. The model provides direction for the algorithm in its attempt to find an optimal solution.

If new objectives are added or existing ones are changed, the model can easily be modified and the algorithm need not be changed. The algorithm can operate on any objective with zero-one real variables in which either maximization or minimization of the slack or real variables is desired. The result of the exponentiation operation on variables can also be maximized or minimized. Variable interaction (i.e., $d_1^- d_2^- = 10$), however, is not provided for.

Since its initial implementation in August 1981, the model has undergone one revision. It was changed to handle two types of subitems: (1) the subitems as previously described, and (2) subitems with a shipping policy tolerance and cutting allowance different from those previously described. Some objective types in the model required modification to provide for this but the algorithm has not required modification to accommodate model changes.

It is also important that the algorithm was coded in such a way that our use of multiple preemptive priorities has only a minimal effect on computer processing time.

VII. CONCLUSION

Assignment problems of large size are difficult to solve. Hence, a great deal of attention among operations researchers has been devoted to the subject. To our knowledge, as yet no usable technique guarantees optimal solutions to large problems. Large assignment problems with the additional complexities of this particular assignment problem are even more difficult to solve. The nonlinear zero-one combinatorial goal programming model with constructive initial placement algorithm that we developed solves large and complex assignment problems satisfactorily. It is efficient, easy to modify, and produces good results.

REFERENCES

1. R. S. Liggett, "The Quadratic Assignment Problem: An Experimental Evaluation of Solution Strategies," Manage. Sci., *27*, No. 4 (April 1981), pp. 442–58.
2. J. P. Ignizio, *Goal Programming and Extensions*, Lexington, MA: D. C. Heath and Co., 1976, pp. 202–6.
3. C. S. Parker, "An Experimental Investigation of Some Heuristic Strategies for Component Placement," Oper. Res. Quart., *27*, No. 1, i (1976), pp. 71–81.
4. G. W. Graves and A. G. Whinston, "An Algorithm for the Quadratic Assignment Problem," Manage. Sci., *16*, No. 7 (March 1970), pp. 453–71.
5. C. C. Petersen, "A Capital Budgeting Heuristic Algorithm Using Exchange Operations," AIEE Trans., *6*, No. 2 (June 1974), pp. 143–50.
6. S. M. Lee, *Goal Programming for Decision Analysis*, Philadelphia, PA: Auerback Publishers Inc., 1972.

APPENDIX

The Mathematical Formulation (Initial Implementation Formulation—August 1981)

Let:

R_i = Reel$_i$ of cable large enough to hold subitem$_j$

S_j = Subitem$_j$ that can be taken from reel$_i$ of cable

X_{ij} = 1 if subitem$_j$ assigned to reel$_i$, 0 otherwise

CA = 2 (Cutting Allowance)

SA_j = Amount ordered on subitem$_j$

RC_i = Amount of cable on reel$_i$

$LATENESS_j$ = Degree of lateness of subitem$_j$

SP = Shipping policy (assignment) tolerance

SL = Maximum amount allowed to scrap

$d^-, d^+ \geq 0$

A.1 Subitem contingency

$$\sum_{\substack{\text{ALL } R_i \\ \text{ON PLANT}_p \\ \text{OF LINKUP}_k}} X_{ij} = \sum_{\substack{\text{ALL REELS}_i \\ \text{ON PLANT}_p \\ \text{LARGE ENOUGH FOR} \\ \text{SUBITEM}_{j+1} \text{ OF} \\ \text{LINKUP}_k}} X_{ij+1}$$

$\cdot \forall$ LINKUP$_k$, SUBITEM$_j$ OF LINKUP$_k$, PLANT$_p$

where $1 \leq j <$ number of subitems in LINKUP$_k$,

where LINKUP$_k$ = subitem contingency of subitems involving SUBITEM$_j$.

A.2 Reel capacity

$$\sum_{\text{ALL } S_j} (SA_j + CA)X_{ij} + d_{2i}^- = RC_i + CA \ \forall \ \text{REEL}_i$$

A.3 Combinations

a. $$\sum_{\substack{\text{ALL SUBITEMS}_j\\ \text{IN GOOD MATCH}\\ \text{ON REEL}_i}} X_{ij} + (\text{NOSUBFIT}_i)X_g + d_{3i}^- = \text{NOSUBFIT}_i$$

$\forall$ REEL$_i$ where a MATCH$_g$ exists

b. $X_g + d_{3Bi}^- = 1\ \forall$ MATCH$_g$

$WEIGHT = 500 - \text{FITDIFF}_i + (\text{NOSUBFIT}_i 1000),$

where FITDIFF$_i$ = the footage short of an absolutely perfect match

NOSUBFIT$_i$ = the number of subitems in the fit on REEL$_i$

X_g is a variable generated to represent MATCH$_g$
Minimize at Priority 5: $(d_{3Bi}^-\ WEIGHT)\ \forall_i$.

A4. Perfect one-to-one matches

$X_{ij} + d_{4ij}^- = 1\ \forall$ REEL$_i$, SUBITEM$_j$ where a one-to-one match exists

$WEIGHT = (10{,}000{,}000 - (SA_j 1000) + (1000 - \text{REELRANK}_i),$

where REELRANK$_i$ = ranking of REEL$_i$ relative to other reels by amount of cable on reel

Minimize at Priority 4: $(d_{4ij}^-\ WEIGHT)\ \forall_{i,j}$.

A.5 Assignment of future subitems

$$\sum_{\substack{\text{ALL FUTURE}\\ S_j}} X_{ij} - \infty \sum_{\substack{\text{ALL CURR.}\\ \text{AND LATE}\\ S_j}} X_{ij} + d_{5i}^- = 0\ \forall\ \text{REEL}_i$$

A.6 Assignment of current and late subitems

$$\sum_{\text{ALL } R_i} X_{ij} + d_{6j}^- = 1\ \forall\ \text{SUBITEM}_j$$

$WEIGHT = [(\text{LATENESS}_j + 1)100{,}000] + SA_j$, if SUBITEM$_j$ not future; 0, if SUBITEM$_j$ is future

Minimize at Priority 6: $(d_{6j}^-\ WEIGHT)\ \forall_j$.

A.7 Forced subitems

$$\sum_{\substack{\text{ALL FORCED}\\ \text{SUBITEMS}_j}} \sum_{\text{ALL } R_i} X_{ij} + d_7^- = \infty$$

$WEIGHT = 1{,}000{,}000$

Minimize at Priority 6: $(d_7^- \ WEIGHT)$.

A.8 Use of full reels

$$\sum_{\text{ALL } S_j} (SA_j + CA + SP)X_{ij} + d_{8i}^- - d_{8i}^+ = RC_i + CA - 51 \ \forall \ \text{REEL}_i$$

Maximize at Priority 7: $d_{8i}^+ \ \forall_i$.

A.9 Minimization of scrap

$$\sum_{\text{ALL } S_j} (SA_j + CA + SP)X_{ij} + d_{9i}^- - d_{9i}^+ = RC_i - SL + CA \ \forall \ \text{REEL}_i$$

Minimize at Priority 8: $(1/d_{9i}^+) \ \forall_i$.

A.10 Remnants of just over the scrapping limit

$$\sum_{\text{ALL } S_j} (SA_j + CA)X_{ij} + d_{10i}^- - d_{10i}^+ = RC_i - SL - 50 \ \forall \ \text{REEL}_i$$

Minimize at Priority 9: $d_{10i}^+ \ \forall_i$.

A.11 Remnants of less than 500 feet

$$\sum_{\text{ALL } S_j} (SA_j + CA)X_{ij} + d_{11i}^- - d_{11i}^+ = RC_i - 500 \ \forall \ \text{REEL}_i$$

Minimize at Priority 10: $d_{11i}^+ \ \forall_i$.

A.12 Use of minimum number of reels and largest reel

$$\sum_{\text{ALL } S_j} X_{ij} + d_{12i}^- - d_{12i}^+ = 1 \ \forall \ \text{REEL}_i$$

Maximize at Priority 11: $d_{12i}^- \ \forall_i$
Minimize at Priority 12: $(RCd_{12i}^-) \ \forall_i$.

A.13 Vacation

a. $$\sum_{\substack{\text{ALL CURRENT}\\ \text{AND LATE}\\ S_j}} \ \sum_{\substack{\text{ALL REELS}_i\\ \text{OF PLANTS}\\ \text{NOT ON}\\ \text{VACATION}\\ \text{DURING}\\ \text{PREPARATION}\\ \text{OF SUBITEM}_j}} X_{ij} + d_{13a}^- = \infty$$

Minimize at Priority 1: d_{13a}^-.

b. $$\sum_{\substack{\text{ALL CURRENT}\\ \text{AND LATE}\\ S_j}} \ \sum_{\substack{\text{ALL REELS}_i\\ \text{OF PLANTS}\\ \text{ON VACATION}\\ \text{ONE WEEK}\\ \text{DURING}\\ \text{PREPARATION}\\ \text{OF SUBITEM}_j}} X_{ij} + d_{13b}^- = \infty$$

Minimize at Priority 2: d_{13b}^-.

A.14 Transportation

$$\sum_{\text{ALL SUBITEMS}_j} \sum_{\text{ALL } R_i} \{[1{,}000{,}000(LATENESS_j + 1)]/(TRANSPRT_{ij}SA_j) + 1\}X_{ij} + d_{14}^- = \infty,$$

where $TRANSPRT$ = transportation rate from stocking location of REEL$_i$ to job site of SUBITEM$_j$

Minimize at Priority 3: d_{14}^-.

Note: For generated variables representing a combination of good match, $LATENESS_j$ is the largest degree of lateness of all orders represented by the generated variable.

AUTHORS

Julian H. Fletcher, B.S. (Industrial Engineering), 1970, Georgia Institute of Technology; AT&T Technologies, Inc., 1970—. Mr. Fletcher's assignments include design and implementation of aggregate plant loading and inventory management systems for products manufactured within the Cable and Wire Division of AT&T Technologies, Inc. He is in the Information Systems Development Organization at the Atlanta Works. Member, the Institute of Management Sciences (TIMS).

Kenneth R. Lipske, B.S. (Mathematics), 1970, University of Wisconsin, Eau Claire; AT&T Technologies, Inc., 1970—. Mr. Lipske's assignments entail optimizing the aggregate plant loading and inventory management functions for the various product lines manufactured by the Cable and Wire Division of AT&T Technologies, Inc. He is in the Information Systems Development Organization at the Atlanta Works. Member, The Institute of Management Sciences (TIMS).