1-1-1991

Mixed-integer linear programming reformuilations [sic] for some nonlinear discrete design optimization problems

Ignacio E. Grossmann
Carnegie Mellon University

V. T. Voudouris

Omar Ghattas

Carnegie Mellon University.Engineering Design Research Center.

Mixed-Integer Linear Programming Reformullations for Some Nonlinear Discrete Design Optimization Problems

I.E. Grossmann, V.T. Voudouris, O. Ghattas

EDRC 06-102-91

Mixed-Integer Linear Programming Reformulations for Some Nonlinear Discrete Design Optimization Problems

LE. Grossmann[1] , V/I\ Voudouris[1] and Ch Ghattas[1,2]

Abstract

This paper deals with a special class of nonlinear discrete design optimization problems which involve nonlinear separable objective functions and bilinear constraints. These constraints involve products of design and state variables in which the former are restricted to take discrete values Two special cases are identified for which advantage can be taken of the discrete nature of the design variables to reformulate these problems as MILP models which can be solved to global optimality. The computational expense can be reduced with SOS 1 sets and a simple solution strategy that is proposed. The application of the MILP reformulations is applied to multiproduct batch plant problems in chemical engineering and

[1] **Engineering Design Research Center, Carnegie Mellon University, Pittsburgh, PA 15213.**

[2] **The authors gratefully acknowledge financial support from the Engineering Design Research Center.**

to structural design problems in civil engineering. Numerical results and comparisons with other methods are also presented.

1. INTRODUCTION

Many problems in engineering design give rise to nonconvex nonlinear programming (NLP) problems (e.g. see Floudas and Pardalos, 1990). Furthermore, quite often due to manufacturing constraints, design variables are restricted to take discrete values for selecting standard sizes which gives rise to mixed-integer nonlinear programs (MINLP) (e.g. see Papalambros and Wilde, 1988; Grossmann, 1990). These problems in many cases have a continuous relaxation that corresponds to a nonconvex NLP. Due to the difficulty in solving these problems, many design models reported in the literature have assumed continuous sizes, and used ad-hoc rounding procedures. It is the purpose of this paper, to show that important classes of discrete design optimization problems that involve separable objective functions and bilinearities in the constraints, can in fact be reformulated as mixed-integer linear programs (MILP), and therefore solved rigorously to global optimality.

This paper will be organized as follows. In Section 2 we will present basic NLP and MINLP formulations that arise in discrete design optimization problems. In Section 3 we will consider two special cases of bilinear constraints that arise in many design applications. We will show that advantage can be taken of the discrete nature of the design variables in order to reformulate these problems as MILP models. Section 4 will compare the proposed formulations with other linearization schemes and briefly discuss computational aspects. Sections 5, 6 and 7 will present the application of the reformulations to multiproduct batch plant problems in chemical engineering and to structural design problems in civil engineering. These problems have traditionally been formulated as continuous optimization problems, and thereby neglected the fact that in most practical applications only standard sizes are available. Finally, Section 8 will present some numerical results.

2. BASIC FORMULATIONS

Consider the following MINLP problem with separable objective function and with linear and nonlinear constraints:

$$\begin{array}{c} \textbf{min C} = \underset{i=1}{\overset{n}{X}} {}^{f}\mathbf{i}({}^{x}\mathbf{i}) \\ \text{s.t.} \quad \mathbf{Az.b} \\ g(x,z)\leq 0 \\ \mathbf{x}\in \mathbf{X_D} \ , \ zeZ \end{array} \qquad \textbf{(P1)}$$

where $firR^1$ —> R^1 and $g{:}R^{n+m}$ —»RQ are continuous functions, $XoCR^{n}{}_{+}$ is a set of discrete values, ZcR^m is a set of continuous values, and $A€R^{rxm}$, $b€R^r$. In the context of a design optimization problem C is a cost function, x is the vector of discrete design variables which are restricted to choices of standard sizes in the set XD, and z is the vector of continuous state variables.

Since the MINLP problem (PI) is in general difficult to solve, it is common to consider the continuous relaxation of problem (PI) which leads to the NLP problem:

$$\begin{array}{c} \min C = \underset{i=l}{\overset{n}{£}}\ fi(xi) \\ \text{s.t.}\quad Az < b \\ g(x,z) \lesssim 0 \\ x€X\ ,\ zeZ \end{array} \qquad \text{(P2)}$$

where X is the convex hull of X_D.

The common approach is then to find an optimal solution to problem (P2) with a standard NLP solver (e.g. MINOS or SQP algorithm) and round up the variables XJ, i=l...n, to the next highest discrete value in XD- Clearly the difficulty is that this might lead to a suboptimal solution, or to a solution that is infeasible. Furthermore, to complicate matters, the functions fj(x), i=l..n, and g(x,z) are often nonconvex which can give rise to several local optima in problem (P2).

Another way to circumvent the problem of getting non-discrete design variables is to reformulate problem (P2) as an MINLP problem with 0-1 variables. That is, let DS(i)={dii,di2>—diN(i)} be the set of discrete values for each design variable xj. Furthermore, let yi_s, s=l,...N(i), be 0-1 variables defined as follows:

$$y_{is} = \begin{cases} 1 & \text{if } x_i = d_{is} \\ 0 & \text{otherwise} \end{cases} \qquad (1)$$

Then, since each variable xj can be expressed as:

$$x_i = \sum_{s=1}^{N(i)} d_{is} y_{is} \qquad (2)$$

$$\sum_{s=1}^{N(i)} yis = i \qquad (3)$$

problem (P2) can be reformulated as the MINLP model:

$$\min C = \underset{i=1}{\overset{n}{2}}{}_{rf} fi(x_i)$$

$$\text{s.t.} \quad Az \le b \qquad (P3)$$

$$g(x,z) \le 0$$

$$Xi = \underset{s=l}{\overset{N(i)}{X}} {}^{d}isYis \qquad i=1,\ldots n$$

$$\underset{s=l}{\overset{N(i)}{\underset{JLJ}{T}}} \underset{yis}{v} - \underset{A}{1} \qquad \underset{i}{i\text{-}l} \ \underset{i,..n}{n}$$

$$XGX \ , \ z \in Z \ , \ y_{is} = \{0,1\}$$

Problems (PI) and (P3) can be solved in principle with a branch and bound method (Gupta, 1980), Generalized Benders Decomposition (Geoffrion, 1972) or with the Outer-Approximation method (Duran and Grossmann, 1986). However, the inherent difficulty is that due to possible nonconvexities in the nonlinear functions, these algorithms may not converge to the global optimum. The next section will show, however, that for special cases of the nonlinear constraints g(x,z) that involve bilinearities, problem (P3) can be reformulated as an MILP problem and solved to global optimality.

3. MILP REFORMULATIONS FOR SPECIAL CASES

Consider the two following particular cases for the nonlinear constraints g(x,z):

a) Case 1: $g_{ij} = \alpha_{ij} x_i v_j - \beta_{ij} \leq 0$, $j \in J(i)$, $i = 1..n$ (4)
where v is a subvector of $z^T = [u,v]^T$ and $\alpha_{ij} \neq 0$, $\beta_{ij} \neq 0$.

b) Case 2: $g_{ij} = \alpha_{ij} x_i v_j - \beta_{ij} w_j \leq 0$, $j \in J(i)$, $i = 1..n$ (5)
where v and w are subvectors of $z^T = [u,v,w]^T$ and $\alpha_{ij} \neq 0$, $\beta_{ij} \neq 0$.

For simplicity we consider here the case of inequalities, although (4) and (5) could also involve equality constraints. Case 1 is clearly a particular case of Case 2, but as will be shown below it leads to a simpler reformulation which is worth considering. Also, as will be shown later in the paper, Cases 1 and 2 arise in multiproduct batch design problems, while Case 2 arises in structural design problems.

For the MILP reformulation consider first the objective function C in (P3). By introducing the binary variables y_{is} as in (1) subject to the constraints in (3), then by defining

$$c_{is} = f_i(d_{is}) \qquad (6)$$

it is clear that C can be expressed by the linear combination

$$C = \sum_{i=l}^{n} \sum_{s=l}^{NO)} c_{is}\ y_{is} \qquad (7)$$

Consider now Case 1. From (4) it follows that for xi > 0,[3]

$$(XjjVjA \quad jeJ(i) \ , \ i=l..n \qquad (8)$$

In order to remove the nonlinearity in the right-hand side of (8), the inverse of the design variable x$ will be represented by a linear combination of inverse values of the discrete sizes; that is,

$$\frac{1}{x_i} = \sum_{s=l}^{N(i)} \frac{m}{d_{is}} \qquad i=l,..n \qquad (9)$$

Then by substituting (9) into (8) yields the linear inequalities,

$$\alpha_{ij}\ v_j \le \sum_{s=l}^{N(i)} \frac{\beta_{ij}}{d_{is}}\ y_{is} \quad j \in J(i) \qquad i=l,..n \qquad (10)$$

Hence, from (7), (10), (3) and by expressing the linear constraints Az ≤ b as [Ai, Aj | U 1 <L ? L problem (P3) can be
LvJ Lb2J
reformulated for Case 1 as the binary MILP problem:

[3] If xi > 0 does not hold, Case 2 applies.

$$\min C = J) \sum_{i=l}^{n} \sum_{s=l}^{N(i)} c_{is}\, y_{is}$$

$$\text{s.t.} \quad [AiAdi:\wedge\wedge^{1}! \qquad \text{(Rl)}$$

$$«iJ^{v}J''^{v} \sum_{s=l}^{N(i)} Ŧ^{l}_{flis}\, B\, y« \wedge 0 \quad Je\ J(i),\ i=L.n$$

$$\sum_{S=l}^{N(i)} y_{is} = 1 \qquad i=1,..n$$

$$U \in U, V\ G\ V \qquad y_{is} = \{0,1\}$$

where UxV = Z and Ci_s is given by (6). Note that the interesting feature in this formulation is that it has fewer variables and fewer constraints than the MINLP model (P3).

Consider now Case 2. Substituting (2) into (5) leads to the bilinear constraints

$$^{a}ij\ 2_{s=l}^{N(i)}\ ^{d}isyisVj - pijWj \leq 0 \qquad jeJ(i)\ ,\ i=l,..n \qquad (11)$$

In order to remove the bilinear terms yisv.j, define the continuous variables $v;j_s$ such that

$$Vj = X_{s=l}^{N(i)}\ ^{v}ijs \qquad J^{eJ}(0\ .\ ^{i=1,,,n} \qquad (12)$$

$$v_j^L\, y_{is} \leq v_{ijs} \leq VjU\, y_{is} \qquad je\ J(i),\ s=l,\ N(i),\ i=l..n \quad (13)$$

where Vj^L, Vj^U are valid lower and upper bounds. Then, the constraints in (11) can be replaced by the linear inequalities

$$\text{otij} \; \overset{N(i)}{\underset{s=l}{X}} \; {}^{d}\text{is}^{v}\text{ijs} - \text{PijWj} \leq 0 \qquad \text{je } J(i) \text{ , i=l,..n} \qquad (14)$$

A proof for the equivalence of the constraints in (14), (12) and (13) with the inequalities in (5) for discrete values in x* is given in the Appendix. Hence, from (7), (12)-(14), (3) and by expressing the linear inequalities $Az \leq b$ as

$$[AiA_2A_3] \begin{bmatrix} U \\ v \\ w \end{bmatrix} \leq \begin{bmatrix} bi \\ b_2 \\ b_3 \end{bmatrix} \qquad (15)$$

problem (P3) can be reformulated for Case 2 as the binary MILP problem:

$$\min \; C = \overset{n}{\underset{i=l}{\text{£}}} \; \overset{N(i)}{\underset{s=l}{\text{£}}} \; c_{is} \; y_{is}$$

(R2)

$$\text{S.t.} \quad [A1A2A3] \begin{bmatrix} u \\ v \\ w \end{bmatrix} \leq \begin{bmatrix} bi \\ b_2 \\ b_3 \end{bmatrix}$$

$$\alpha_{ij} \sum_{s=l}^{N(i)} d_{is} v_{ijs} - \beta_{ij} w_j \leq 0 \qquad j \in J(i) \text{ , i=1,..n}$$

$$\text{'}\wedge i^L \; \text{Yis} \; \wedge \; \text{Viic} < \text{Vi}^U \; \text{Vi}_s \qquad \begin{matrix} j \in J(i) \text{ , s=1..N(i) ,} \\ \text{i=1..n} \end{matrix}$$

$$v_j = \sum_{s=1}^{N(i)} v_{ijs} \qquad j \in J(i)\ ,\ i=1..n$$

$$\sum_{s=1}^{N(i)} y_{is} = 1 \qquad i=1,..n$$

$$u \in U\ ,\ v \in V\ ,\ w \in W \qquad y_{is} = \{0,1\}$$

where UxVxW = Z and c_{is} is given by (6). Note that in this case the number of variables and constraints is larger than in the MINLP model (P3).

4. REMARKS

The proposed linearization of the bilinear constraints in (11) with (12)-(14) can also be applied to the case when (11) is given by equality constraints. This follows trivially from the proof in the Appendix. Also, given the reformulation of the objective function in (6) and (7), no assumption is required on the form of the cost function $f_i(x_i)$. In fact this function can be discontinuous, which is not an uncommon occurrence in practice. The linearizations in (12)-(14), however, are not unique. Other alternatives include the linearizations by Glover (1975) and by Torres (1991). As will be shown below the former requires a larger number of constraints and may yield a weaker LP relaxation. The latter, which is only applicable to inequalities, requires fewer constraints, but may also yield a weaker LP relaxation.

The bilinear constraints in (11) can be linearized with the following formulation proposed by Glover (1975):

$$v_j^L y_{is} \le v_{ijs} \le v_j^U y_{is}$$
$$v_{ijs} \ge v_j - v_j^U(1- y_{is})\; j\in J(i),\;\; s=l,\; N(i),\;\; i=1..n$$
$$v_{ijs} \le v_j - v_j^L(1- y_{is}) \qquad (16)$$

which requires $4 \overset{n}{\underset{i=l}{\wedge}}|J(i)||N(i)|$ inequalities. In contrast the proposed linearization scheme in (12) and (13) only requires $2\overset{n}{\underset{i=l}{XI}}{}^{J}G)\,11^{N}®I^{+}\;\overset{n}{\underset{i=l}{XI}}{}^{J}\wedge I$ constraints. Furthermore, while a point (vjj$_s$, Vj, yi$_s$) satisfying (12) and (13) satisfies the inequalities in (16), the converse may not be true. For instance, assume a non-integer point yj$_s$ such that vy$_s$ = Vj^{u}yi$_s$. Using (3) it follows from (16) that

$$vj' + Cvf-vjOyi, \le Vj \le vf \qquad (17)$$

while (12) yields Vj = Vj^u. Thus, the inequalities in (16) may produce a weaker LP relaxation.

For the case when the bilinear constraints in (11) are only inequalities, Torres (1991) has shown that it is sufficient to consider the following constraints from (16):

$$Vj^L\, y_{is} \wedge v_{ijs}$$
$$v_{ijs} \ge VJ - {}_{Vj}U(l- y_{is}) \quad je\; J(i),\;\; s=l,\; N(i),\;\; i=1..n \qquad (18)$$

which requires $\overset{n}{\underset{i=l}{\wedge}} |\, J(i) \,|$ fewer constraints than the proposed linearization in (12) and (13). However, the above inequalities also can produce a weaker LP relaxation. For instance, setting Vijs = Vj^is for a non-integer point *y\s* yields,

$$v_j \leq v_j^U - (v_j^U - v_j^L)\, y_{is} \tag{19}$$

while (12) yields Vj = Vj^L. In summary, the proposed linearizations in (12) and (13) are tighter since they exploit the convexity condition in (3), while the ones by Glover (1975) and Torres (1991) do not.

As for the computational requirements, the reformulations (Rl) and (R2) correspond to MILP problems that can be solved to global optimality with branch and bound methods such as the ones implemented in SCICONIC, MPSX, ZOOM and LINDO. Furthermore, the constraints in (3) correspond to special ordered sets of type 1 (SOS1; e.g. see SCICONIC, 1990) whose structure can be exploited to reduce the number of nodes that must be examined in the branch and bound enumeration. While problem (Rl) is somewhat smaller in size than the nonlinear model in (P3), problem (R2) is potentially much larger and has the additional difficulty that lower and upper bounds Vj^L, Vj^U,must be supplied which can have a great impact in the integrality gap of the LP relaxation.

The issue of size in problem (R2) can be addressed in several ways. One is to generate cutting planes in the MILP in order to strengthen the LP relaxation (e.g. see Van Roy and Wolsey, 1987). The other one is to apply Benders decomposition so as to greatly reduce the size of the LP subproblems (see Sahinidis, 1990). In our experience, however, we have found that the greatest source of computational difficulty in the proposed MILP models (particularly in (R2)) lies in their tendency to predict small sizes in the relaxed LP creating many infeasible nodes in the branch and bound tree. To circumvent this problem we have devised a simple but rigorous solution strategy for fixing subsets of 0-1 variables in (Rl) and (R2) that consists of the following steps:

Stepl. (Optional). Obtain an upper bound. The relaxed LP is solved to compute sizes xj^R from (9) or (2). The MILP model is solved with binary variables yi_s fixed to zero for which $dj_s <$

^{x}i .

Step 2. Predict valid lower bounds $x*^L$ for each size xj. In simple models these can be obtained analytically. In more complex models these can be obtained by maximizing (9) or minimizing (2)) with the relaxed LP model. Note that if step 1 is used, those Xi^R with zero value yield valid lower bounds.

Step 3. Obtain the global optimum by solving the MELP model with binary variables y_{is} fixed to zero for which $di_s < xj^L$, and with the upper bound obtained in step 1.

This procedure can obviously be made more effective if the MILP problems in steps 1 and 3 are solved with SOS1 sets.

5 • APPLICATION TO SIMPLE BATCH PROCESS DESIGN

In order to illustrate the application of the reformulation (Rl), consider the problem of sizing multiproduct batch plants with one unit per stage (see Fig. 1) and operating with single product campaigns (see Grossmann and Sargent, 1978). In these plants it is assumed that each product requires all the processing stages in the same sequence.

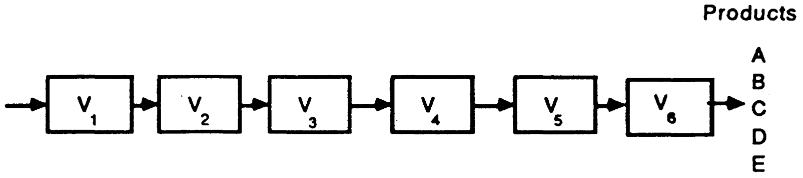

Fig. 1. Multiproduct batch plant with one unit per stage

The following parameters and variables are required to formulate the design optimization problem:

Parameters:

N	the number of products j=1..N
M	the number of stages i=1..M
T_{Lj}	the cycle time of product j
Q_j	demand of product j
S_{ij}	size factor for product j in stage i
H	horizon time
γ_i, δ_i	cost coefficients for unit in stage i with $0 < \gamma_i < 1$

Variables:

V_i	size of unit in stage i
B_j	batch size of product j

If the sizes V_i are assumed to be continuous, the problem of finding an optimal design as given by the NLP:

$$\text{min} \quad C = \overset{M}{\underset{i=l}{X}}\ 8j\ \mathit{Wf}$$

$$\text{(Bl)}$$

$$\text{s.t.} \quad Vj \geq Sij\ Bj \qquad j=l,..N, \quad i=l,..M$$

$$N \quad n$$

$$j\text{-}i \quad {}^{B}J$$

$$Bj,\ Vi > 0$$

In problem (Bl) the objective function represents the investment cost that is to be minimized. The first set of constraints simply states that at each stage i, the size Vi must be sufficiently large for all products j. Finally, the second constraint states that the total time for production, as given by the number of batches (Qj/Bj) by the corresponding cycle time (TLJ), should not exceed the allotted time H.

Problem (Bl) corresponds to an NLP that involves nonconvexities in the objective function and in the second constraint. However, as shown by Grossmann and Sargent (1978) this problem can be transformed into a geometric programming problem with posynomial terms.

Assume, however, that the sizes Vi are only available in discrete values DV(i) = {Vn,Vi2...ViN(i)}. Rather than formulating (Bl) as an MINLP according to problem (P3), we will show that it can be reformulated as an MILP. First, note that the first set of constraints in (Bl), which contains the design variable Vi, is linear while the second constraint is nonlinear. In order to remove the nonlinearity in the latter, let

$$\text{Bj} = \underset{\text{J}}{\wedge} \qquad \text{j=l..N} \tag{20}$$

with which problem (Bl) reduces to

$$\min \quad C = \sum_{i=1}^{M} \delta_i V_i^{\gamma_i}$$

(B2)

$$\text{s.t.} \quad \text{-Vibj+Sij} \leq 0 \qquad \text{j=l..N, i=l..M}$$

$$\sum_{j=1}^{N} Q_j \, T_{Lj} \, b_j \leq H$$

$$\text{Vi>0 , bj>0}$$

Note that the first set of constraints are now of the same form as equation (4) for Case 1. If we introduce the 0-1 variables yi_s as in (3) to select the discrete size Vi_s, then by letting

$$\frac{1}{V_i} = \sum_{s=1}^{N(i)} \frac{y_{is}}{\text{V}_{\text{»}}{}^{\text{s}}} \tag{21}$$

as in (9), following a similar treatment for the derivation of problem (Rl), the reformulated MBLP corresponds to:

$$\min \quad C = \sum_{i=1}^{M} \sum_{s=1}^{N(i)} {}^{c}\text{is} \ \wedge\text{s}$$

(RB)

$$\text{s.t.}\quad b_j \geq \sum_{s=1}^{N(i)} S_{ij} \frac{y_{is}}{V_{is}} \qquad j=1..N \;\; i=1..M$$

$$\sum_{s=1}^{N(i)} y_{is} = 1 \qquad i=1..M$$

$$\sum_{j=1}^{N} Q_j T_{Lj} b_j \leq H$$

$$b_j \geq 0, \qquad y_{is} = \{0,1\}$$

where C_{is} = *bi(V[$_s$)* The important feature of this model is that it readily allows the treatment of discrete sizes which have been commonly treated as continuous variables in previous work.

6. APPLICATION TO STRUCTURAL DESIGN

In order to illustrate the reformulation (R2), consider the least weight design of a truss consisting of a number of specified bars with fixed nodal locations (see Fig. 2) that is subject to a number of different loading conditions, and for which constraints on stresses, nodal displacements and bar elongations are specified (see Bremicker et al., 1990; Haftka et al., 1990; Ghattas and Grossmann, 1991).

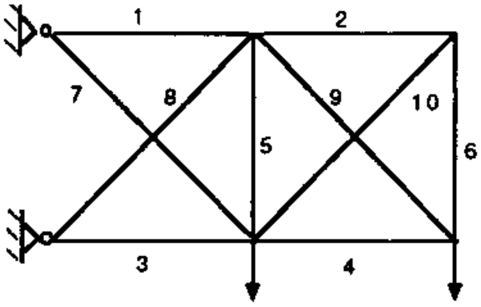

Fig. 2. Truss with 10 bars and 2 loads.

In order to find a design with minimum weight the following parameters and variables must be considered:

Parameters:

M	number of bars $i=1,M$
N	number of degrees of freedom[4] $j=1,N$
L	number of loading conditions $\ell=1,L$
ℓ_i	length of bar i
E_i	modulus of elasticity of bar i
ρ_i	density of bar i
$P_{\ell j}$	jth component of load at condition ℓ
b_{ji}	direction cosine relating the force in bar i with the degree of freedom j
σ_i^L, σ_i^U	maximum stresses in compression and tension in bar i
v_i^L,v_i^U	limits on elongations in bar i
d_j^L,d_j^U	limits of displacements for degree of freedom j

Variables:

A_i	cross sectional area of bar i
$s_{i\ell}$	force in bar i for condition ℓ
$\sigma_{i\ell}$	stress in bar i for condition ℓ
$v_{i\ell}$	elongation of bar i for condition ℓ
$d_{j\ell}$	displacement at degree of freedom j for condition ℓ

[4] Number of degrees of freedom is given by $N = N_N e - b$, where N_N is the number of nodes, e the degrees of freedom at each node, and b is the number of support fixities.

If the cross sectional areas are assumed to be continuous, the optimal design for a truss with minimum weight is given by the NLP:

$$\min \quad C = \underset{i=1}{\overset{M}{\Upsilon,}} \; P_i \; k \; A_i \qquad \text{(ST)}$$

S.t.

(a) Equilibrium equations

$$\sum_{i=1}^{M} b_{ji}\, s_{i\ell} = p_{\ell j} \qquad j=l,N \quad I=1,L$$

(b) Compatibility equations

$$\sum_{j=1}^{N} b_{ji}\, d_{j\ell} = v_{i\ell} \qquad i=l_f M \quad A=1,L$$

(c) Hooke's law

$$\frac{E_i}{1}\text{-}\wedge\text{7-}A_i \; v_{iA} = s_{iA} \qquad i=1\wedge M\_\, i\_ {,} IL$$

(d) Stress equations

$$\text{1Jl} \quad \frac{E_i}{ij} \quad \bullet * \qquad 1=1,M \quad 4=1,L$$

(e) Bounds

$$G_I^{L} \leq an < G_i^{U} \qquad i=l,M \quad 4=1,L$$

$$v\wedge\wedge v_j \wedge V_i^{11} \qquad i=l,M \quad H=1,L$$

$$d_j^{L} \leq d_{jA} \leq d_j^{u} \qquad j=l,N \quad 4=1,L$$

$$A_i \geq 0 \qquad i«=l,M$$

$$s_{i4} \; e \; R^{1} \qquad i=l,M, \quad H=1,L$$

Note that except for Hooke's law equations in (c), which involve bilinear terms of the form in equation (5), the above model is linear in the objective function and in the remaining constraints.

It should be noted that the above model is commonly formulated by eliminating the variables v^ and d^ b y substituting the compatibility equations into Hooke's law equations, and the result in the equilibrium equations. Although this reduces the number of variables and constraints, it actually increases the number of bilinear terms.

Assume now that the cross section areas Ai are specified with discrete values DA(i)={An, Ai_S2,..AjN(i)}. To reformulate problem (ST) as an MILP, we introduce the 0-1 variables yi_s as in (3) to select the discrete sizes Ai_s. By then letting

$$A_i = \sum_{s=l}^{N(i)} A_{is}yis \qquad (22)$$

as in (2), substituting into Hooke's law equations yields

$$\frac{E;}{*i} \sum_{s=l} A_{is}\, v_{1\ell}\, y_{is} = s_{i4} \qquad i=l,M \quad 1=1,L$$

By defining the variables VJ& as in (12) to (14), and by setting

$$c_{is} = pi<tiA_{is} \quad . \qquad (23)$$

then by analogy to problem (R2) the MILP reformulation of (ST) yields:

$$\min \quad C = \hat{}\sum_{i=l}^{M} \sum_{s=l}^{N(i)} c_{is}\, y_{is}$$

(RST)

$$\text{s.t.} \quad \underset{i=1}{\overset{M}{X}} {}^{b}Ji\, {}^{s}ii = PH \qquad j=l..N, \;\; 1=1,L$$

$$\mathop{X}\limits_{j=l}^{N} {}^{b}Ji \ \% = \mathop{X}\limits_{s=l}^{N(i)} {}^{v}Us \qquad i=l..M,\ 1=1,L$$

$$\frac{E_i}{\ell_i} \sum_{s=l}^{N(i)} A_{is}\, v_{i\ell\, s} = s_{i\ell} \qquad i=1..M,\ \ell=1,L$$

$$v_{i\ell}{}^{L}\ yis \wedge v_{a}, \leq v_{i4}u\ y_{is} \qquad \begin{array}{c} s=1,\ N(i),\ \ell=1,L \\ \underset{x\sim\sim}{i}\underset{x}{=1}\, {}_{j}M \end{array}$$

$$\sum_{i=1}^{N(i)} y_{is} = 1 \qquad i=1,M$$

$$\sigma_{i\ell} = \frac{E_i}{\ell_i} \sum_{s=1}^{N(i)} v_{i\ell s} \qquad i=l..M,\ 1=1,L$$

$$ai^{L}\leq Oi|[\leq Oi^{U} \qquad i=l..M,\ \ 1=1,L$$

$$dj^{L}\leq djj|\leq dj^{U} \qquad j=l..N,\ \ 1=1,L$$

$$\textit{sue}\ R^{1} \quad i=l,M\ ,\ 1=1,L$$

$$v_{iils} \text{€}\ R^{1}\ \ s=l,N(i),\ \ Jl=l,L\ \ i=l,M$$

$$y_{is} = \{0,l\}\ \ s=l,N(i),\ \ i=l,M$$

where the variable v^ has been substituted from the equation

$$ViJ = \mathop{X}\limits_{s=l}^{N(i)} {}^{v}iis \qquad (24)$$

and cj_s is a parameter given by (23). The importance of model (RST) is that it allows the rigorous removal of bars with which one can optimize the topology, as well as the sizes of the bars. Further discussion on this model can be found in Ghattas and Grossmann (1991).

7. APPLICATION TO COMPLEX BATCH PROCESS DESIGN

As a third application we will consider the optimal design of multipurpose batch plants with multiple production routes, a problem that has been considered recently by Faqir and Karimi (1990).

As opposed to the problem considered in Section 5, in this problem we are given a number of stages with a number of potential units of identical type. Also, not all products require all production stages and therefore potential production routes are specified for each product as shown in Fig. 3. Faqir and Karimi (1989) formulated this design problem as an MINLP problem which involves bilinearities in the constraints which cannot be transformed into a geometric programming problem.

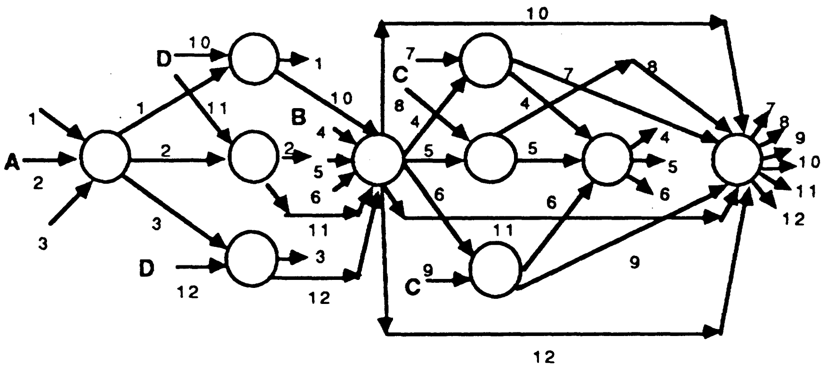

Fig.3. Complex multiproduct batch plant

The following parameters and variables are involved in this formulation.

Parameters

N the number of products j=1,..N

M the total number of potential units

NR the number of production routes r=1,..NR

Er index set for units i involved in route r

Rj index set for routes r that produce product j

TLT cycle time of route r

Qj demand of product j

Sri size factor for route r in unit i

Vi_s discrete size s for unit i, i=l, N(i) (Vii = 0)

H horizon time

Yi, 5i cost coefficients for unit i with 0 < Yi < 1

~~Variables~~

Vi size of unit i

B_r batch size in route r

q_r amount produced in route r

0_r production time spent with route r

yi_s 0-1 variable to denote selection of size s in unit i

The MINLP model for the optimal design problem is then given by

$$\underset{i=l\quad s=l}{\overset{M\quad N(i)}{\min C = n^{8}\, i^{v}\, i^{T\,l}}} \qquad \text{(MB)}$$

S.t.

(a) Volume requirements for units

$$Vi \geq S_{ri} B_r \qquad iEE,, \quad r=l,NR$$

(b) Production in each route

$$q_r = \%\underset{\cdot\, l_r}{\wedge} \qquad i{*}l,NR$$

(c) Demand constraint

$$j=1,N$$

$$reRj$$

(d) Definition of sizes

$$V_i = \sum_{s=l}^{N(i)} V_{is}\, y_{is} \,; \qquad \sum_{s=l}^{N(i)} y_{is} = 1, \;\; i=1,M$$

(e) Horizon constraints

$$h(e_1,e_2,...e_r,..)\leq H$$

$$Vi\geq 0 \quad i=l,M \quad y_{is} = \{O,l\} \quad s=l,N(i) \quad i=l,N$$

$$B_r,q_r,0_r\geq O \qquad r=l,NR$$

where h(0i, G2..6NR) are linear functions that define the allocation of times for production (see Faqir and Karimi, 1990).

As can be seen, problem (MB) is an MINLP that involves nonconvexities in the objective function and in the constraints (b). In order to reformulate (MB) as an MILP, substitute B_r from constraints (b) into the constraints in (a) which then leads to

$$- Vi9_r + SriTuq, \leq 0 \quad ieEr, \;\; r=l,NR \tag{25}$$

which has exactly the same form as equation (5) for Case 2.

By setting the cost coefficients $Ci_s = 8i\, V\wedge^1$, and by applying equations (12) and (13) as in model (R2) for the variables Grfs, the resulting MILP model is

$$\min \quad C = \overset{M}{\underset{i=l}{2}} \; \overset{N(i)}{\underset{s=l}{]T}} \; c_{is}\; yis$$

(RMB)

$$\text{s.t.}\quad \sum_{s=l}^{N(i)} X\ {}^{V}\text{«}\ {}^{0}\mathbf{ris} * S_{ri}\ T\text{^}\ q_r \qquad \text{ie E,,}\quad r=l,NR$$

$$\theta_r = \sum_{s=l}^{N(i)} {}^{e}\text{"}s \qquad \text{ieEr,}\quad r=l,NR$$

$$\theta_{ris} - 6_r^U\ y_{is}\text{^}\ 0 \qquad s=l,\ N(i),\ \text{i€Er} \qquad r=l,NR$$

$$\sum_{s=1}^{N(i)} y_{is} = 1 \qquad i=l,M$$

$$\sum_{reRj} q_r = Q_j \qquad j=1,N$$

$$h(0i,\ 02,..e_r) \leq H$$

$$6_r,\ q_r \geq 0 \qquad r=l,NR$$

$$6_{ris}\ \text{^}0 \qquad s=l,N(i),\quad \text{ie}E_r,\quad r=l,NR$$

$$y_{is} = \{0,l\} \qquad s=l,N(i),\quad i=l,M$$

where a simple choice of the upper bound 9_r^U is H, the total horizon time.

8. NUMERICAL RESULTS

Batch process design

Consider the problem by Voudouris and Grossmann (1991) of a multiproduct batch plant which produces five different products A,B,C,D and E. The plant consists of six stages involving one piece of equipment (see Fig. 1). The demands for the five products are: 250,000 tons per year for A, 150,000 tons per year for B, 180,000 tons per year for C, 160,000 tons per year for D and 120,000 tons per year for E. Size factors and cycle times are given for each product and the specified time horizon is 6000 hours. The equipment at any stage are available in 5 discrete sizes; namely 3000,3750,4500,5860 and 7325 liters. In this case the equipment cannot be removed from the process train and so the value of 0 liters is not allowed.

The problem can be formulated as the NLP (B1) where the equipment volumes are relaxed to be continuous variables. The optimal design found by this formulation has an investment cost of $231,489.6, and the corresponding optimal volumes are [6017.6, 3483.6, 3960.9, 4823.4, 4646.5, 3885.5] liters. Rounding up to the next available size gives [7325, 3750, 4500, 5860, 5860, 4500] liters and a capital investment of $255,886.2. In this case there is no need to check for the feasibility of the proposed solution since rounding up gives always a feasible design. The CPU time required with GAMS/MINOS 5.2 (Brooke et al, 1988; Murtagh and Saunders, 1985) was 2.3 seconds in a VAX-6420. The NLP involved 11 continuous variables and 31 constraints.

The MINLP formulation of the problem was convexified with exponential transformations and solved using the Outer Approximation algorithm implemented in DICOPT (Kocis and Grossmann, 1989). It involved 30 0-1 variables, 11 continuous variables and 43 constraints. For the NLP's the solver used was MINOS 5.2 whereas for the MILP's SCICONIC (1990) was used. The problem converged in 19.9 seconds in a VAX-6420: 2.3 seconds or 12% of the total time were required for the NLP subproblems and 17.6 seconds or 88% of the total time were required for the MILP master problems. The optimal solution obtained was [5860 , 3750 , 3750 , 5860 , 4500 , 4500] liters for the equipment in each stage and the capital investment required was $238,650.24.

By formulating the problem as the MILP in problem (RB), 30 0-1 variables, 5 continuous variables and 37 constraints were involved. Using SCICONIC on a VAX-6420, 3.1 seconds were required to solve the MILP problem to optimality; using ZOOM (Marsten, 1986) in the same computer required 23.2 seconds. The globally optimal solution found using the MILP formulation had a capital investment of $238,650.24 with equipment sizes [5860 , 3750 , 3750 , 5860, 4500 , 4500] liters. This solution is the same as the one obtained by the MINLP formulation. Note that the solution found by the NLP formulation followed by the rounding of the equipment sizes does not yield an optimal design as its capital investment is 7.2% higher than the global optimum solution. It is also worth noting that the CPU time required by the MILP formulation when solved with SCICONIC is only slighly higher than the CPU time required by the continuous NLP model when solved with MINOS.

Structural Design

Consider the 5 bar fan stress shown in Fig. 4 which is subject to one load of 100,000 lbs (Ghattas and Grossmann, 1991). The modulus of elasticity is $1x1O^7$ psi, the density is 0.1 lb/in^3 for each bar and the maximum stress is 20,000psi in compression or tension. Also, for each bar 6 discrete sizes are assumed, [0,2,4,6,8,10] in^2, where the zero value allows for the possible removal of the bars. Also constraints were specified on maximum displacement and elongation; these were not binding at the solution. The problem was first formulated as the NLP in (ST) which led to a minimum weight of 146.25 lb with cross-section areas of A=[6.988,0,0,0,6.525] in^2. By rounding up the sizes to [8,0,0,0,8] and resolving the NLP a feasible solution was obtained with a weight of 172.97 lb. The total CPU-time required with GAMS/MINOS was 1.94 sec on VAX-6420 (1.43 sec first NLP, 0.51 sec second NLP). The NLP involved 23 continuous variables and 18 constraints.

The problem was also formulated as an MINLP which required 30 0-1 variables, 23 continuous variables and 28 constraints. With the outer-approximation method implemented in DICOPT (Kocis and Grossmann, 1989) the problem converged in 6.4 sec using MINOS for the NLP subproblems and ZOOM for the MELP master problems. Generalized Benders decomposition was also applied but it failed to converge to the optimum. Finally, a branch and bound method

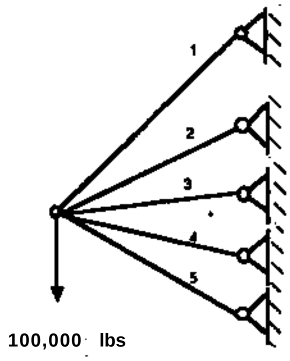

Fig. 4. Five-bar fan truss

was also used for the MINLP which was formulated as in (PI) requiring 5 discrete variables, 23 continuous variables and 18 constraints. The method required 11.5 sec enumerating 10 nodes which were solved with MINOS. By formulating the problem as an MILP in problem (RST) 30 0-1 variables, 43 continuous variables and 83 constraints were involved. The computer code SCICONIC solved the MILP problem to optimality in 3.5 sec while ZOOM required 9.3 sec. The optimal solution of the MILP led to a minimum weight of 172.18 lb with areas A = [8,0,0,2,6] in^2 which was the same solution that was found with DICOPT and with the branch and bound method. Note that in this case the rounded solution is [8,0,0,0,8] with weight 172.97 lb. This solution does not correspond to the optimum design although the difference in the weight is only 0.5% higher. It is also worth noting that the rounded solution yields a different topology than the global optimum.

As an additional example, consider the truss shown in Fig. 2. which consists of 10 bars and is subjected to two loading conditions of 100,000 lbs. The modulus of elasticity is lxlO^7 psi, the density is 0.1 lb/in^3 for each bar and the maximum stress is 25,000psi in compression or tension. For each bar 11 discrete sizes are assumed, [0,1,2,3,4,5,6,7,8,9,10] in^2, where as in the example above, the zero value allows for the possible removal of the bars. Also constraints were specified for the displacements and elongations. By formulating the problem as the MILP (RST), 110 0-1 variables, 148 continuous variables and 268 constraints

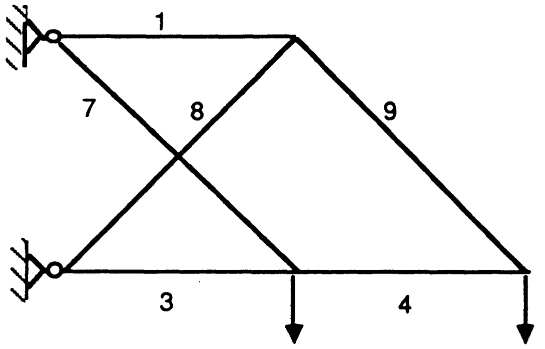

Fig. 5. Optimal topology for truss in Fig. 2.

were involved. The computer code SCICONIC solved the MILP problem to optimality in 602 sec on an HP-9000/720 (without SOS1 sets). On a VAX-6420 the problem could not be solved to optimality in 900 sec without the SOS1 sets; when they were included, the problem was solved to global optimality in 135 sec. As for the solution strategy described in the remarks section, the problem was solved also with SCICONIC on the VAX-6420 in 75.2 sec (step1: 6 sec LP, 7.5 sec MILP; step 2: 30.2 sec LP's; step3: 31.5 sec MILP with SOS1). The optimal solution of the MILP led to a minimum weight of 1636.4. lb with areas A = [8,0,8,4,0,0,6,6,6,0] in^2 which yields the configuration shown in Fig. 5. When this problem was solved as an MINLP in (P3), DICOPT (Kocis and Grossmann, 1989) failed to converge to the global optimum from several starting points due to the nonconvexities; Generalized Benders decomposition did not converge after 900 sec due to the large number of infeasible NLP subproblems. It should be noted, that when the problem was solved as a continuous NLP, the model required 48 continuous variables and 38 constraints. With zero initial guesses for all variables MINOS failed to converge; with nonzero guesses (all areas with size 5), the rounded solution was the same as the one of the MILP requiring a total of 6.1 sec.

Complex batch process design

Consider the design of a multipurpose batch plant with multiple production routes that is shown in Fig.3 (Faqir and Karimi, 1990). Assume that this plant must produce four products A, B, C and D satisfying the demands of 300000, 250000, 180000 and 200000 tons per year, respectively. The time horizon in which these demands have to be satisfied is 6000 hours. As seen in Fig. 3, there are a total of 10 potential equipment units available which are placed in 6 different processing stages. Each product must be manufactured with different processing stages which then gives rise to a total of 12 potential production routes. The equipment are available in 6 discrete sizes; namely 0, 500, 1000, 2000, 2500 and 3000 liters. Note that the nonexistence of an equipment is represented by a volume of 0 liters. There is a constant processing time characterizing every task of a product in a specific equipment group.

Faqir and Karimi (1990) have developed a special purpose strategy for solving the associated MINLP problem which requires a significant amount of preprocessing and user interaction. Formulating the problem as the MILP in (RMB), 50 0-1 variables , 225 continuous variables and 234 constraints are involved. The computer code SCICONIC was used in order to solve the problem through GAMS. The optimal solution obtained requires a capital investment of $124,500 which is the same solution as the one reported by Faqir and Karimi (1990). The CPU requirements on a VAX-6420 were 305.2 seconds. It has to be mentioned here that exploiting the fact that the logical constraints in (RMB) can be treated as special ordered sets of type 1 (SOS1), the CPU time was further reduced to 148.4 seconds.

9. CONCLUSIONS

This paper has considered a special class of nonlinear discrete design optimization problems which involve nonlinear separable objective functions in the design variables and bilinear constraints that are given by products of design and state variables, where the former are restricted to take discrete values. Two special cases for the constraints were identified for which it was shown that the discrete nature of the design variables can be exploited to reformulate these problems as MILP models. The solution of these models can be expedited through the use of SOS1 sets, and with a simple solution strategy that relies on deriving valid lower bounds on the sizes. The application of the MILP reformulations was applied to multiproduct batch plant problems in chemical engineering and to structural design problems in civil engineering. These represent novel design optimization models that can explicitly handle discrete sizes, and therefore avoid the common heuristic rounding procedures for discrete nonlinear programming models. Numerical results have been presented which show that the proposed models not only produce global optimum solutions, but are computationally competitive when compared to nonlinear formulations with continuous sizes.

REFERENCES

Bremicker, M., P.Y. Papalambros and H.T. Loh, "Solution of Mixed-Discrete Structural Optimization Problems with a New Sequential Algorithm", *Computers and Structures,* 37,451-461 (1990).

Brooke, A., D. Kendrick and A. Meeraus, "GAMS A User's Guide", Scientific Press, Palo Alto (1988).

Duran, M.A. and I.E. Grossmann, "An Outer-Approximation Algorithm for a Class of Mixed-Integer Nonlinear Programs", *Mathematical Programming,* 36, 307-339 (1986).

Faqir, N.M. and LA. Karimi, "Design of Multipurpose Batch Plants with Multiple Production Routes", Proceedings of Foundations of Computer Aided Process Design (Eds. Siirola et al.), Elsevier, Amsterdam, 451-468 (1990).

Floudas, C.A. and P.M. Pardalos, "A Collection of Test Problems for Constrained Global Optimization Problems", Lecture Notes in Computer Science (Eds. G.Goos and J.Hartmanis), Springer-Verlag, Berlin (1990).

Geoffrion, A.M., "Generalized Benders Decomposition", *Journal of Optimization Theory and Applications,* 10, 237-260 (1972).

Ghattas, O. and I.E. Grossmann, "MINLP and MILP Strategies for Discrete Sizing Structural Optimization Problems", to appear in Proceedings of ASCE 10th Conference on Electronic Computation, Indianapolis (1991).

Glover, F., "Improved Linear Integer Programming Formulations of Nonlinear Integer Problems", *Management Science,* 22,455-460 (1975).

Grossmann, I.E., "Mixed-Integer Nonlinear Programming Techniques for the Synthesis of Engineering Systems", *Research in Engineering Design* , 1, 205-228 (1990).

Grossmann, I.E. and R.W.H. Sargent, "Optimum Design of Multipurpose Chemical Plants", *Ind.Eng.Chem. Process Design and Dev.*, **18**, 343 (1978).

Gupta, O.K., "Branch and Bound Experiments in Nonlinear Integer Programming", Ph.D. Thesis, Purdue University (1980).

Haftka, R.T., Z. Gurdal and M.P. Kamat, "Elements of Structural Optimization", Kluwer Academic Publishers, Dordecht (1990).

Kocis, G.R. and I.E. Grossmann, "Computational Experience with DICOPT Solving MINLP Problems in Process Systems Engineering", *Computers and Chemical Engineering*, **13**, 307-315 (1989).

Marsten, R., "User's Manual for ZOOM/XMP", University of Arizona (1986).

Murtagh, B.A. and M.A. Saunders, "MINOS User's Guide", Systems Optimization Laboratory, Department of Operations Research, Stanford University (1985).

Papalambros, P.Y. and D.J. Wilde, "Principles of Optimal Design", Cambridge University Press, Cambridge (1988).

Sahinidis, N.V., "Mixed-Integer Optimization Techniques for Planning and Scheduling of Chemical Processes", Ph.D. thesis, Carnegie Mellon University, Pittsburgh (1990).

SCICONIC/VM User Guide (Version 2.11), Scicon Ltd., Milton Keynes (1990).

Torres, F.E., "Linearization of Mixed-Integer Products", *Mathematical Programming*, **49**, 427-428 (1991).
Van Roy, T.J. and L.A. Wolsey, "Solving Mixed Integer Programs by Automatic Reformulation", *Operations Research*, **35**, 45-57 (1987).

Voudouris, V.T. and I.E. Grossmann, "MILP Formulations for the Optimal Design of Batch Processes", submitted for publication (1991).

APPENDIX. On the equivalence for the linearization of case 2.

Proposition. The constraints,

$$\alpha_{ij} \sum_{s=l}^{N(i)} \mathrm{d}\text{isVijs - pijWj\_} \leq 0 \qquad \text{je J(i), i=l,..n} \qquad \text{(Al)}$$

$$v_j = \sum_{s=l}^{N(i)} {}^{v}\text{>js} \qquad \text{jeJ(i), i=l,..n} \qquad \text{(A2)}$$

$$\sum_{s=l}^{N(i)} y_b = 1 \qquad \text{(A3)}$$

$$Vj^{L}\, y_{is} \leq v_{ijs} \leq VJU_{yis} \quad j_e\, j(i),\ _{s=}i\ _f\ N(i),\ i=l.._n \qquad \text{(A4)}$$

are equivalent to the inequalities,

$$\alpha_{ij}x_iv_j - \beta\text{ijWj} \leq 0\ ,\ \text{jej(i),}\quad i=l.._n \qquad \text{(A5)}$$

for $$xj = \underset{s=l}{\overset{N(i)}{\text{LT}}} d_{is}y_{is},\quad \underset{s=l}{\overset{N(i)}{\text{£}}} \text{yis=l, i=l,-n;} \qquad \text{yis=O,l} \quad \text{(A6)}$$

Proof. Let y_{it}(i) =1 and y_{is}=0 for s * t(i), s=l,..N(i), i=l,...n. From (A6) it follows that xi = dj_t(i), with which (A5) becomes,

$$\text{otijdit(i)Vj - PijWj} \leq 0\ ,\ \text{j€j(i),} \qquad i=i.._n \qquad \text{(A7)}$$

Also, from (A4) it then follows that VJJ_S = 0, j€j(i), s = l,...N(i), i = l,...n, s * t(i). Hence, from (A2),

$$^{v}j = {}^{v}ijt(i) \qquad \text{J€J(i),} \qquad \text{i=l..n} \qquad \text{(A8)}$$

and $$\text{aijdi}_t\text{(i)Vij}_t\text{(i) - pyWj} \leq 0\ ,\ \text{j€j(i),} \qquad i=l.._n \qquad \text{(A9)}$$

Substituting (A8) into A(9) leads to

$$\text{aydit(i)Vj - pij WJ\_} \leq 0\ ,\ \text{j€j(i),} \qquad i=i.._n \qquad \text{(A 10)}$$

which is identical to (A7).