DEFINING PROPERTIES FOR DECOMPOSITION IN NONLINEAR PROGRAMMING

Technical Report No. UM-MEAM 97-04

Terrance C. Wagner Panos Y. Papalambros

1 INTRODUCTION

Five defining properties of an NLP problem determine which decomposition method is appropriate. The report reviews and compares prevalent methods in the context of these properties to develop a set of relationships between a given problem structure and a coordination strategy. Two of the most important properties are *linking variables* and *linking functions* which are most easily defined by the *functional dependence table (FDT)* discussed in Section 2. The other three determine convergence properties of the methods. The defining properties, the coordination strategy, and the merits and demerits of each method are itemized: prevalent hierarchical methods are presented in Section 3; prevalent non-hierarchical methods are presented in Section.4. Features are compared and summarized in Section 5. Inherent in each of the methods is an authority model which is also discussed. The report emphasizes defining properties of the NLP and features of the coordination strategy and hence, the reader is referred to the original sources for more explicit details of the algorithms in each coordination strategy.

2 DEFINING PROPERTIES

The most important defining properties are *linking variables* and *linking functions*. Linking variables are loosely defined as variables that, when held fixed, effect independent optimization problems; similarly, linking functions (usually constraints) are loosely defined as functions that, when deleted (relaxed), effect independent optimization problems. Rigorous mathematical definition of these properties is given in Chapter 3 of Wagner (1993).

A useful artifice for characterizing defining properties is a table of Booleans called a *functional dependence table* (FDT). Using the integers '1' and '0' for true and false, respectively, rows are labeled with function names, columns are labeled with variable names, and the element in the *ith* row and *jth* column is non-zero if the *ith* function depends on the *jth* variable.

The functional dependence table for the example NLP given as Equation (1) (from Kirsch [1981]) is shown in Table 1. Each term in the objective is represented separately in the FDT.

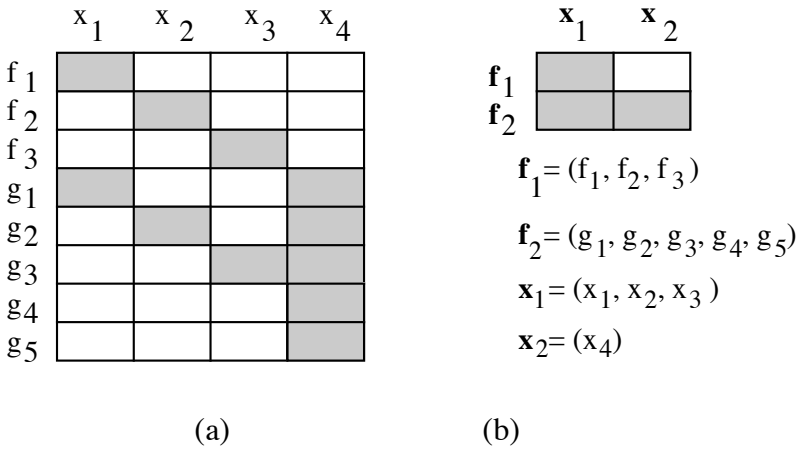
$$
\begin{aligned}
&\min \quad f(\mathbf{x}) = 400\, x_1 + 20\, x_2 + 130\, x_3^2 \\
\text{subject to:} \quad &g_1(\mathbf{x}) = 190\, x_1^2 - 43.6 + 14.9\, x_4 - 1.44\, x_4^2 \le 0 \\
&g_2(\mathbf{x}) = 38\, x_2^2 - 183.3 + 36\, x_4 - 2.67\, x_4^2 \le 0 \qquad (1) \\
&g_3(\mathbf{x}) = 650\, x_3^2 - 244 + 45.9\, x_4 - 3.29\, x_4^2 \le 0 \\
&g_4(\mathbf{x}) = 3.5 - x_4 \le 0 \\
&g_5(\mathbf{x}) = x_4 - 6.5 \le 0
\end{aligned}
$$

The FDT's of many NLP problems are presented in this report, and shading facilitates an easy visual comparison of the structure. Figure 1 (a) shows Table 1 using shading for the

Boolean value 'true'. If variables and functions are partitioned into vectors with disjoint index sets, compact forms like Figure 1 (b) can be constructed. In the compact form, a shaded block implies that a *k*-vector of scalar-valued functions *depends* on an *j*-vector of variables. Every function in the *k*-vector need not depend on every variable in the *j*-vector, but every function in the *k*-vector depends on at least one variable in the *j*-vector and every variable in the *j*-vector appears in at least one function in the *k*-vector. For example, in Figure 1(b), the functions f_1 and f_2, do not depend on x_3, but the function partition, $\mathbf{f}_1 = (f_1, f_2, f_3)$, depends on the variable partition, $(\mathbf{x}_1) = (x_1, x_2, x_3)$. Formally, the compact form can be constructed if and only if the columns in the *k x j* sub-table of the original dependence table can be arranged such that at least a diagonal of non-zero entries exists.

Table 1.
Functional Dependence Table for (1)

	x_1	x_2	x_3	x_4
f_1	1	0	0	0
f_2	0	1	0	0
f_3	0	0	1	0
g_1	1	0	0	1
g_2	0	1	0	1
g_3	0	0	1	1
g_4	0	0	0	1
g_5	0	0	0	1



(a) (b)

Figure 1. (a) FDT using shading for Booleans (b) FDT by vector partitions.

Additional defining properties are *additive separability*, *linearity*, and *convexity*. Additive separability exists when a function (constraint or objective) is a sum of terms, each term dependent on a subset of the problem variables, the subsets being disjoint. For example, the objective in Equation (1) is additively separable. Linear properties allow use of more classical techniques, as in Lasdon [1971]; convex properties often guarantee convergence; monotonic properties facilitate analytic solution in the subproblems. Linearity is easily identified; techniques for identifying and exploiting monotonicity are well documented (Papalambros and Wilde, [1988]). The defining properties are summarized in Table 2.

Table 2
Defining Properties of NLP Problems Utilized in Decomposition

1. Linking variables
2. Linking constraints
3. Additive separability
4. Linearity/Convexity
5. Monotonicity

Most decomposition methods exploit at least one of the defining properties to construct a coordination strategy. Certain variables are identified as coordinating variables, the remainder as local variables. A master problem is formulated in terms of coordinating variables; subproblems solve for optimal local variables treating the coordinating variables as parameters. This generic strategy applies to hierarchic and non-hierarchic methods and is summarized in Algorithm 0.

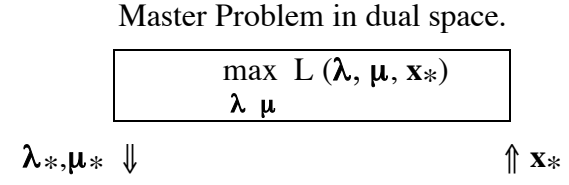
Algorithm 0: Generic Coordination Strategy for NLP Decomposition Methods.
1. Initialization.
2. Solve master problem to obtain optimal coordinating variables.
3. Solve subproblem(s) to obtain optimal local variables.
4. Optimality or Convergence Test.
5. If Step 4 fails make modifications based on Step 4 information and return to Step 2; otherwise stop. Decomposed solution solves original problem.

The selection of coordinating variables depends on the properties of the optimization problem. Generally, *feasible decomposition* methods choose linking variables as coordinating variables; *dual decomposition* methods choose dual variables associated with the linking constraints as coordinating variables. Separability in the objective function or constraints often effects a separable Lagrangian which can be decomposed. Specific instances of linearity, convexity, and monotonicity enhance convergence properties of the coordination strategies.

3 HIERARCHICAL DECOMPOSITION METHODS

Dual Decomposition Methods

The Lagrange formulation transforms a constrained problem into an unconstrained minimization problem. The Karush-Kuhn-Tucker conditions cast the NLP problem as a set of nonlinear zero-valued equalities. The regularity assumption guarantees existence of the Lagrange multipliers, $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$, for equalities and inequalities respectively. For differentiable functions, the stationarity conditions imply that an interior optimum, ($\mathbf{x}_*$, $\boldsymbol{\lambda}_*$, $\boldsymbol{\mu}_*$), is a saddle point of the Lagrange function, L. Formulation of the dual problem leads to the min-max iterative strategy that serves as the generic algorithm for dual decomposition methods given as Algorithm 1. Figure 2 illustrates the strategy. Necessary conditions can be exploited to derive update formulas for the dual variables.



$$\min_{\mathbf{x}} \; L(\boldsymbol{\lambda}_*, \boldsymbol{\mu}_*, \mathbf{x})$$

Subproblem in primal space.

Figure 2 Two level structure of dual methods.

Algorithm 1: Coordination Strategy for Dual Decomposition Methods.

1. Initialize k = 0, $(\mathbf{x}^k, \boldsymbol{\lambda}^k, \boldsymbol{\mu}^k)$.

2. Holding $(\boldsymbol{\lambda}^k, \boldsymbol{\mu}^k)$ constant, obtain $\mathbf{x}_*^k$ by solving

$$\min_{\mathbf{x}} \; L(\boldsymbol{\lambda}^k, \boldsymbol{\mu}^k, \mathbf{x})$$

3. Holding $\mathbf{x}_*^k$ constant, obtain $(\boldsymbol{\lambda}_*^k , \boldsymbol{\mu}_*^k)$ by solving

$$\max_{\boldsymbol{\lambda}\,\boldsymbol{\mu}} \; L(\boldsymbol{\lambda}, \boldsymbol{\mu}, \mathbf{x}_*^k)$$

4. If converged stop; otherwise increment k, set $(\mathbf{x}^k, \boldsymbol{\lambda}^k, \boldsymbol{\mu}^k) = (\mathbf{x}_*^{k-1} , \boldsymbol{\lambda}_*^{k-1} , \boldsymbol{\mu}_*^{k-1})$ and go to 2. By formulating the dual, the original NLP problem is partitioned into a two level problem where the optimal dual variables are sought in the master problem and the optimal primal variables are sought in the subproblem. If the objective function and the constraints are sums, the vector $\mathbf{x}$ can often be partitioned into p vectors, $\mathbf{x}_i$; i = 1,..., p where the indices of $\mathbf{x}_i$ and $\mathbf{x}_j$ form disjoint sets for i≠j. The Lagrangian can then be partitioned into p functions,

$$L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = L_1(\mathbf{x}_1, \boldsymbol{\lambda}, \boldsymbol{\mu}) + L_2(\mathbf{x}_2, \boldsymbol{\lambda}, \boldsymbol{\mu}) + ... + L_p(\mathbf{x}_p, \boldsymbol{\lambda}, \boldsymbol{\mu}). \qquad (2)$$

resulting in p independent subproblems.

Three prevalent dual methods by Dantzig and Wolfe [1960], Takahashi [1964], and Lasdon [1968], have the common defining property of linking functions. To facilitate a concise description of each method, an FDT schematic of the original problem is presented followed by a table with the format of Table 3. The original problem is stated in the upper left entry; properties and reformulations are listed in the upper right entry; the master problem is shown in the lower left; the subproblem(s) in the lower right. Variable value passing is shown in the middle column.

The classical dual decomposition method proposed by Dantzig and Wolfe [1960] is applicable to any linear program, but is very effective in solving LP problems with constraint sets having the FDT shown in Figure 3. The primal form of such problems, given in Table 3, is a set of *p* linear programs linked by the constraints, $\mathbf{A}_0\mathbf{x} = \mathbf{b}_0$. The master problem is formulated in terms of the extreme points of the feasible domain. Assuming the *p* polytopes, $S_i = \{\mathbf{x}_i: \mathbf{A}_i\mathbf{x}_i = \mathbf{b}_i; \mathbf{x}_i \geq 0\}$ are bounded, any point, $\mathbf{x}_i$ can be represented as a weighted sum of its t_i extreme points, $(\mathbf{x}_{e_{i1}},..., \mathbf{x}_{e_{it_i}})$. The original problem can be rewritten as the master and subproblems in Table 3.

Recall[1] that for linear programs the basic feasible solution that minimizes the objective is the optimal solution and that a basic feasible solution is an extreme point. Starting with an initial basic feasible solution, the simplex method iteratively solves the linear program by updating the basic feasible solution based on the non-basic variable that most reduces the objective function. The measure of that reduction is called the relative cost coefficient.

The Dantzig-Wolfe method implements the simplex method in a two step fashion. Given a basic feasible solution, $\mathbf{x}_b$, the master problem determines the optimal weighting coefficients, α, which in turn allow computation of the simplex multipliers, λ, associated with the basic feasible solution. The multipliers associated with the linking constraints, λ_o, are a sub-vector of λ, and their values are passed to the subproblems as parameters. The minimum cost coefficient associated with non-basic variables in each of the subproblems is an explicit function of the *ith* subproblem solution, $\mathbf{x}_{i*}$ and the multipliers, λ. The subproblem with the minimum cost coefficient that most reduces the objective contains the non-basic variable to update the basic feasible solution. The size

[1] See Bazaara et al. [1990] or Luenbuerger [1984] for a thorough review of linear programming solution methods.

of the subproblems effects reduced storage and computation in determining the relative cost coefficients. It constitutes the numerical advantage of the method. The algorithm is summarized in Algorithm 2.

Table 3

Dantzig-Wolfe Decomposition

Dantzig-Wolfe		
Original Problem		Properties/Transformations
min $\mathbf{c}^T\mathbf{x}$ $\mathbf{x}$ s.to: $\mathbf{A}_o\mathbf{x} = \mathbf{b}_o$ $\mathbf{A}_i\mathbf{x}_i = \mathbf{b}_i$ i = 1, ., p. $\mathbf{x} \geq \mathbf{0}$		Linear in $\mathbf{x}$ Linking functions: $\mathbf{A}_o\mathbf{x} = \mathbf{b}_o$ Extreme points transform: $\mathbf{x}_i = \sum_{j=1}^{t_i} \alpha_{ij}\mathbf{x}_{e_{ij}}$ i = 1,..., p Cost of the *jth* extreme point: (scalar): $p_{ij} = \mathbf{c}^T\mathbf{x}_{e_{ij}}$ $\mathbf{p} = (p_{11},..., p_{1t_1}, ..., p_{p1},..., p_{pt_p})$ Activity vector: $\mathbf{q}_{ij} = \mathbf{A}_o\mathbf{x}_{e_{ij}}$ Define: $\boldsymbol{\alpha} = (\alpha_{11},..., \alpha_{1t_1}, ..., \alpha_{p1},..., \alpha_{pt_p})^T$ $\mathbf{p} = (p_{11},..., p_{1t_1}, ..., p_{p1},..., p_{pt_p})^T$ $\mathbf{q} = (\mathbf{q}_{11},..., \mathbf{q}_{1t_1}, ..., \mathbf{q}_{p1},..., \mathbf{q}_{pt_p})$ Basic solution: $\mathbf{x}_b$ Basis matrix: $\mathbf{B}$
Master Problem		Subproblem
min $\mathbf{p}^T\boldsymbol{\alpha}$ $\boldsymbol{\alpha}$ s.to: $\mathbf{q}\boldsymbol{\alpha} = \mathbf{b}_o$ $\sum_{j=1}^{t_i} \alpha_{ij} = 1$: i = 1,..., p $\boldsymbol{\alpha} \geq \mathbf{0}$ $\boldsymbol{\lambda}^T = \mathbf{c}_b^T\ \mathbf{B}^{-1}$ $\boldsymbol{\lambda}_o$ = first *m* rows of $\boldsymbol{\lambda}$	$\boldsymbol{\lambda}_o \Rightarrow$ $\Leftarrow$ new basis $\mathbf{x}_b$	min $(\mathbf{c}_i^T - \boldsymbol{\lambda}_o^T\mathbf{A}_{oi})\mathbf{x}_i$ $\mathbf{x}_i$ s.to: $\mathbf{A}_i\mathbf{x}_i = \mathbf{b}_i$ $\mathbf{x}_i \geq \mathbf{0}$

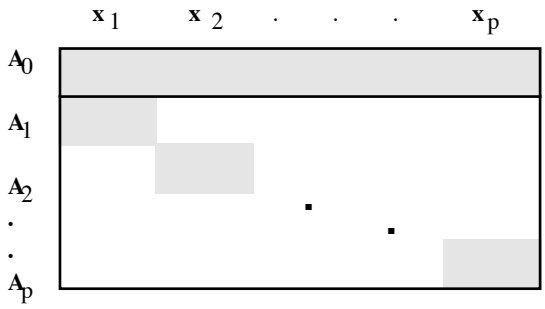


Figure 3. FDT schematic of LP constraints in D-W Decomposition.

Algorithm 2: Coordination Strategy for Dantzig-Wolfe Decomposition.

1. (Initialization) Select an initial basis feasible solution for the master problem.
2. (Master Problem) Solve the master problem to obtain the multipliers, $\boldsymbol{\lambda}$.
3. (Subproblem) With $\boldsymbol{\lambda}$ fixed, obtain the optimal solution $\mathbf{x}_{j*}$ for each subproblem.
   Compute the minimum cost coefficient associated with each subproblem.
4. (Optimality Test) If all cost coefficients are non-negative, stop. The current basic feasible solution is optimal.
   Otherwise, use the solution of the subproblem with the minimum cost coefficient to update the basic feasible solution.
5. Return to 2 with the new basic feasible solution.

The economic interpretation of the decomposition method, considers the LP problem as a model of a multidivisional firm minimizing cost with constraints on shared resources. If management regulates prices that the divisions must pay for common resources, the division that can best reduce cost at those prices is incorporated into the master plan. Initialization of the basic feasible solution is interpreted as management formulating a master plan; computation of the simplex multipliers is interpreted as management setting prices. In Step 3, each division reports its potential cost improvement based on the prices, the activity with the greatest improvement is determined in Step 4, and the master plan is updated in Step 5.

Takahashi [1964] proposed a dual method for convex programs which have the FDT schematic of Figure 4 and linear constraints of the form in Table 4. The method makes no assumptions about separability of the objective function. The constraints, $\mathbf{h}_c$, are coupling constraints in the sense that the problem would be easier to solve without them. For example, the remaining constraints, $\mathbf{h}_s$, may possess the structure shown in Figure 4. Formulating the dual of the coupling constraints, $\mathbf{h}_c$, yields the master problem and subproblem given in Table 4. Since $\boldsymbol{\lambda}_c$ is unconstrained, any direction for $\boldsymbol{\lambda}$ in the master problem is feasible. If the objective function is convex, it can be shown that the dual function, $h(\boldsymbol{\lambda})$, is differentiable (Geoffrion [1971]). Because the dual is linear in $\boldsymbol{\lambda}$, the gradient of the dual,

$$\nabla_{\lambda_c} h(\lambda) = \mathbf{h}_c(\mathbf{x}), \qquad (3)$$

is a feasible direction for improving the objective of the master problem. Takahashi suggests Algorithm 3, which uses a short step, α, in the direction of the gradient. A line search could also be used in Step 4.

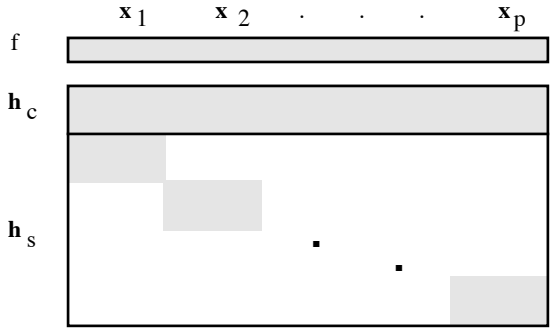


Figure 4. FDT schematic for Takahashi's Dual Decompostion Method.

Table 4.

Takashi's Decomposition.

Takahashi		
Original Problem		Properties/Transformations
min f(**x**) **x** s. to: $\mathbf{h}_c(\mathbf{x}) = \mathbf{0}$ $\mathbf{h}_s(\mathbf{x}) = \mathbf{0}$ $\mathbf{x} \in S_b$ (simple bounds)		Convex f(**x**) Linear constraints Linking functions: $\mathbf{h}_c(\mathbf{x})$
Master Problem		Subproblem
max $h(\lambda_c) = f(\mathbf{x}) + \lambda_c^T \mathbf{h}_c(\mathbf{x})$ λ_c	$\lambda_c \Rightarrow$ $\Leftarrow \mathbf{x}$	min $f(\mathbf{x}) + \lambda_c^T \mathbf{h}_c(\mathbf{x})$ **x** s. to: $\mathbf{h}_s(\mathbf{x}) = \mathbf{0}$ $\mathbf{x} \in S_b$ (simple bounds)

Algorithm 3: Coordination Strategy for Takahashi's Dual Decomposition.

1. Initialize k = 0, and the multipliers, $\lambda_c = \lambda^k$.
2. (Subproblem) Solve the subproblem to obtain $\mathbf{x}_*^k$.
3. (Optimality Test) If $\mathbf{h}_c(\mathbf{x}_*^k) = 0$ (or $\|\mathbf{h}_c(\mathbf{x}_*^k)\| \le \varepsilon$), then $\mathbf{x}_*^k$ is optimal. Stop. Otherwise go to 4.
4. (Master Problem) Let $\lambda^{k+1} = \lambda^k + \alpha\, \mathbf{h}_c(\mathbf{x}_*^k)$ where $\alpha > 0$. Return to 2.

The NLP problem with the FDT in Figure 5 has the separability properties summarized in Table 5. The objective function *and* the constraints are additively separable; Lasdon proposed Algorithm 4 below for such problems. The formulation given in Table 5 shows only inequalities but the method can be applied to probems with equalities. The Lagrange function is additively separable with respect to the variables $\mathbf{x}_i$ and linked by the multipliers, $\boldsymbol{\mu}$. The coordination strategy in Algorithm 4 solves for $\boldsymbol{\mu}$ in the master problem and for $\mathbf{x}_i$ in each of the subproblems.

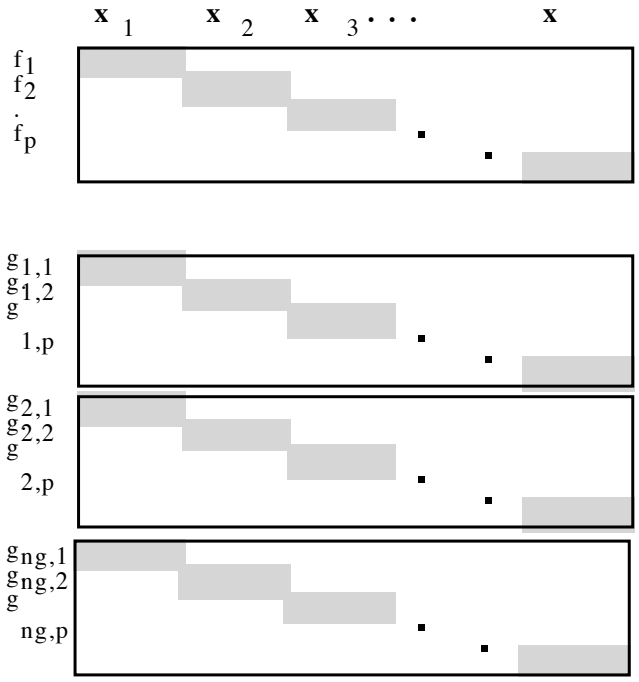


Figure 5. FDT schematic for Lasdon's Dual Decomposition Method.

Algorithm 4: Coordination Strategy for Lasdon's Dual Decomposition.

1. Initialize $k = 0$ and $\boldsymbol{\mu}^k \geq 0$.
2. (Subproblem) Solve the subproblems with $\boldsymbol{\mu} = \boldsymbol{\mu}^k$ to obtain a solution $\mathbf{x}_*^k$.
3. (Master Problem) Solve the master problem using the gradient of the dual as a search direction and a line search in α.
4. (Convergence Test) If $\| \boldsymbol{\mu}^k - \boldsymbol{\mu}^{k-1} \| < \varepsilon$ stop; otherwise $\boldsymbol{\mu}^{k+1} = \boldsymbol{\mu}^k + \alpha \mathbf{s}^k$, increment k and return to 2.

Table 5.

Lasdon's Decomposition.

Lasdon		
Original Problem		Properties/Transformations
$\min_{\mathbf{x}_1 \ldots \mathbf{x}_p} \sum_{i=1}^{p} f_i(\mathbf{x}_i)$ s. to: $g_j(\mathbf{x}) = \sum_{i=1}^{p} g_{ji}(\mathbf{x}_i) \le \mathbf{0}$; $j = 1,\ldots, n_g$		$f(\mathbf{x})$ separable wrt $\mathbf{x}_i$ $g_j(\mathbf{x})$ separable wrt $\mathbf{x}_i$; $j = 1,\ldots, n_g$
Master Problem		Subproblem
$\max_{\alpha} h(\boldsymbol{\mu},\alpha) = \sum_{i=1}^{p} f_i(\mathbf{x}_i) + (\boldsymbol{\mu} + \alpha \mathbf{s})^T \mathbf{g}(\mathbf{x})$ s. to: $\alpha \ge 0$ Search direction, $\mathbf{s}$: $s_j = g_j(\mathbf{x})$ if $\mu_j > 0$ $= \max(0, g_j(\mathbf{x}))$ if $\mu_j = 0$ $j = 1,\ldots,n_g$	$\boldsymbol{\mu} \Rightarrow$ $\Leftarrow \mathbf{x}$	$\min_{\mathbf{x}_i} f_i(\mathbf{x}_i) + \sum_{j=1}^{n_g} \mu_j g_{ji}(\mathbf{x}_i)$ $i = 1,\ldots,p$

Lasdon suggests a gradient-based algorithm with line search for solving the master problem in the dual space. In the case of equality constraints, variable metric or conjugate gradient methods are suggested. Since the Lagrangian is nonlinear in $\mathbf{x}$, the subproblem solutions implicitly depend on $\boldsymbol{\mu}$. Note, $\mathbf{x}_*(\boldsymbol{\mu})$ may be non-smooth with respect to $\boldsymbol{\mu}$. In such regions, the derivative of the dual,

$$\nabla_{\boldsymbol{\mu}} h(\boldsymbol{\mu}) = \mathbf{g}(\mathbf{x}(\boldsymbol{\mu})) \qquad (4)$$

will not be continuous; only one-sided derivatives will exist. Lasdon suggests a piece-wise linear approximation to the dual to surmount this difficulty. Convergence is guaranteed as long as $h(\boldsymbol{\mu})$ remains differentiable.

Dual decomposition methods exploit separability in the Lagrangian effected when dual variables are held fixed. The design interpretation of the master problem is that of a

coordinator setting priorities (the multipliers) to which the subproblems respond. Structure can be imposed at the expense of dimensionality to allow use of a dual method as discussed next.

Goal Coordination

Constraint sets linked as shown in Figure 6, can be decoupled by introducing additional primal variables and equalities. Decomposition is obtained at the expense of added dimensionality. For example, if the constraint vectors $\mathbf{g}_1$ and $\mathbf{g}_2$ are coupled through a variable x, they can be decoupled by adding a variable, y, an equality $h(x,y) = x - y = 0$, and respective multiplier λ. A dual formulation controls λ in the master problem. For n_y variables, separability is imposed by introducing an n_y-vector of variables, $\mathbf{y}$, and n_y equality constraints. Under such circumstances the multipliers associated with the equalities appear in the master problem. Dual methods of this type are called *goal coordination* methods because the master problem is coordinating the goal of meeting the equality constraints.

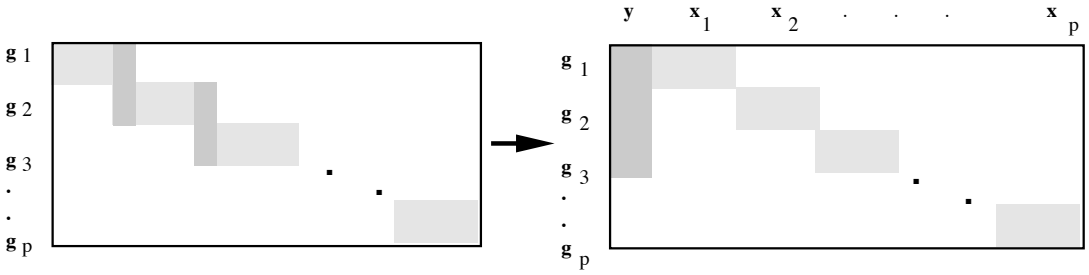


Figure 6. FDT Schematic of effect of introduction of **y** on constraint set for Goal Coordination Methods.

Wismer and Chattergy [1978] proposed this approach even when the coupling is high and use the KKT conditions to derive a gradient algorithm for the master problem. They note, however, that the variable bookkeeping is arduous. For a completely coupled constraint set, the approach implies the introduction of $(p-1)n$ new variables and equalities, which for $p \geq 2$, effects a master problem larger than the original. The merit

of imposed structure at the expense of added dimensionality determines the cost-to-benefit ratio of the method. Diaz and Belding [1991] formulate a master problem using goal programming instead of the dual to accommodate problems with such structure.

Feasible Decomposition Methods

The fundamental characteristic of a feasible method is that the *original* design variables are partitioned into two sets: *global* variables **y** and *local* variables **x**. The values of the vector **y** vary in the master problem, the optimal values $\mathbf{y}_*$ are treated as parameters in the subproblem; the values of the vector **x** vary in the subproblem and the optimal values $\mathbf{x}_*$ are returned to the master problem. The simplest two-level structure is illustrated in Figure 7. The methods are sometimes called *model coordination methods* because model variables are coordinated in both the master problem and the subproblems. The methods are attractive in design problems because even if convergence of the coordination is not obtained, the intermediate solutions are feasible and usually represent an improvement in the objective function. The design interpretation of the master problem is that of a coordinator who has authority over variables common to design groups which act independently.

The subproblem solution, $\mathbf{x}_*$, typically depends on **y**, and this must be accounted for in the master problem for the method to have convergence properties. Rarely, is an explicit solution obtained, so an approximation of the dependence $\mathbf{x}_*(\mathbf{y})$ is usually required. Numerical continuation methods can trace out $\mathbf{x}_*(\mathbf{y})$ when the dimension of **y** is small, (three or less) and a linear approximation can be constructed using sensitivity derivatives when the dimension of **y** is larger. Allgower and Georg [1990] present a comprehensive review of continuation methods, and Beltracchi [1988] gives a comprehensive review of the computational methods for sensitivity derivatives. These

coordination strategies are sometimes called projection methods because the subproblem solution is 'projected' onto the **y** space (Geoffrion [1971]).

Master problem in coordinating variables **y**.

$$\min_{\mathbf{y} \in R^{ny}} f(\mathbf{y},\mathbf{x}_*)$$
$$\mathbf{h}(\mathbf{y}) = \mathbf{0}$$
$$\mathbf{g}(\mathbf{y}) \le \mathbf{0}$$

$\mathbf{y}_* \Downarrow$ $\Uparrow \mathbf{x}_*$

$$\min_{\mathbf{x} \in R^{n}} f(\mathbf{y}_*,\mathbf{x})$$
$$\mathbf{h}(\mathbf{y},\mathbf{x}) = \mathbf{0}$$
$$\mathbf{g}(\mathbf{y},\mathbf{x}) \le \mathbf{0}$$

Subproblem in local variables **x**.

Figure 7. Two level structure of feasible methods.

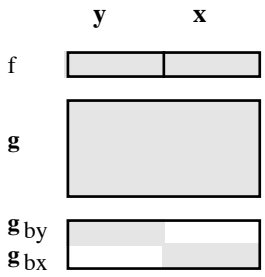


Figure 8. FDT schematic for Johnson's Two-Stage Decomposition.

Johnson [1984a] proposed a straightforward decomposition method given as Algorithm 5 for problems with the structure shown in Figure 8.

Algorithm 5:Coordination Strategy for Johnson's Two-Stage Decomposition.

1. (Subproblem) Solve the subproblem of Figure 7 to obtain the single-valued vector, $\mathbf{x}_* = \mathbf{x}_*(\mathbf{y})$.
2. (Master Problem) Solve the master problem of Figure 7, either analytically, or using a conventional numerical scheme, using $\mathbf{x}_* = \mathbf{x}_*(\mathbf{y})$ to obtain $\mathbf{y}_*$.

Assuming a relationship, $\mathbf{x}_* = \mathbf{x}_*(\mathbf{y})$, can be determined in the subproblem, the master problem seeks $\mathbf{y}_*$ to minimize the objective function, $f(\mathbf{y},\mathbf{x}_*(\mathbf{y}))$. Johnson's exposition assumes the partition is chosen such that optimality conditions in the subproblem can yield an *explicit single-valued* relationship, $\mathbf{x}_* = \mathbf{x}_*(\mathbf{y})$. If an analytical solution is determined also in the master problem, a global optimum is obtained and the method should converge in one overall iteration. Finding a partition that guarantees an explicit single-valued function a priori is not trivial. In some cases, even if $\mathbf{x}_* = \mathbf{x}_* (\mathbf{y})$ exists, nonlinearities may make the master problem of lower dimension but more difficult to solve than the original problem of higher dimension. While not addressed by Johnson, the requirement for an explicit relationship may be circumvented if the optimality conditions of the subproblem are used to derive an *approximation* of $\mathbf{x}_*(\mathbf{y})$. For example, sensitivity derivatives in the subproblem could provide a linear approximation that could be used in the master problem.

Separable Methods

NLP problems with the primal form given in Equation (5) can utilize several feasible decomposition methods.

$$
\begin{aligned}
&\min_{\mathbf{y}\,\mathbf{x}_1 \ldots\, \mathbf{x}_p} \sum_{i=1}^{p} f_i(\mathbf{y}, \mathbf{x}_i) + f_o(\mathbf{y}) \\
\text{subject to}\quad & \mathbf{h}(\mathbf{y}) = \mathbf{0} \\
& \mathbf{g}(\mathbf{y}) \le \mathbf{0} \\
& \mathbf{h}_i(\mathbf{y}, \mathbf{x}_i) = \mathbf{0} \quad \text{for } i = 1, \ldots, p \\
& \mathbf{g}_i(\mathbf{y}, \mathbf{x}_i) \le \mathbf{0} \quad \text{for } i = 1, \ldots, p \\
& \mathbf{y} \in S_b{}^{n_y} \\
& \mathbf{x}_i \in S_b{}^{n_i}
\end{aligned}
\tag{5}
$$

$S_b{}^n$ represents simple bound or side constraints in R^n. Figure 9 illustrates the FDT of Equation (5). The objective function is a sum, and the vectors $\mathbf{g}_i$ and $\mathbf{h}_i$ are independent of $\mathbf{x}_j$, for i, j = 1, .., p and i≠j. The Lagrangian is then additively separable with respect to $(\mathbf{x}_i, \boldsymbol{\lambda}_i\, \boldsymbol{\mu}_i)$ for i = 1, .., p,

$$L(\mathbf{y}, \mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = L_y(\mathbf{y}, \boldsymbol{\lambda}_y, \boldsymbol{\mu}_y) + L_1(\mathbf{y}, \mathbf{x}_1, \boldsymbol{\lambda}_1, \boldsymbol{\mu}_1) + \ ... + L_p(\mathbf{y}, \mathbf{x}_p, \boldsymbol{\lambda}_p, \boldsymbol{\mu}_p). \quad (6)$$

If $\mathbf{y}$ is temporarily treated as a parameter, the p terms of the Lagrangian can be minimized independently.

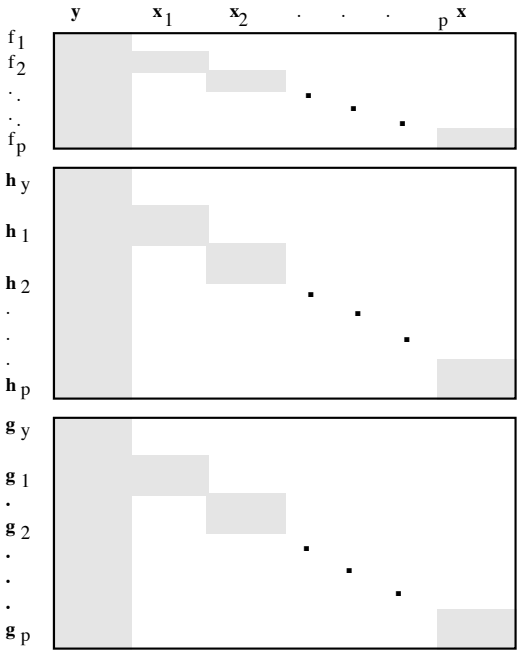


Figure 9. FDT shematic of NLP for a separable feasible decomposition method.

Kirsch [1981] suggests Algorithm 6 for the master-subproblem pair given in Table 6.

Algorithm 6: Coordination Strategy for Kirsch's Decomposition.

1. Initialize k = 0 and initialize the global variables $\mathbf{y} = \mathbf{y}^k$.
2. Holding $\mathbf{y}^k$ constant, solve the p independent subproblems of Table 6 with respect to $\mathbf{x}_i$ to obtain $\mathbf{x}_{*i}$.
3. Modify the value of $\mathbf{y}^k$ to $\mathbf{y}^{k+1}$ to reduce the objective in the feasible domain by a sufficient amount, δ,

$$\sum_{i=1}^{p} f_i(\mathbf{y}^{k+1}, \mathbf{x}_{*i}) - \sum_{i=1}^{p} f_i(\mathbf{y}^k, \mathbf{x}_{*i}) \le \delta.$$

4. If convergence criterion are met, for example, $\| \mathbf{y}^k - \mathbf{y}^{k+1} \| \le \varepsilon$, stop; otherwise increment k and return to 2.

Table 6

Kirsch's Decomposition.

Kirsch		
Orignal Problem		Properties / Transformations
$\min_{\mathbf{y}\,\mathbf{x}} f_O(\mathbf{y}) + \sum_{i=1}^{p} f_i(\mathbf{y}, \mathbf{x}_i)$ s. to: $\mathbf{h}(\mathbf{y}) = \mathbf{0}$ $\mathbf{h}_i(\mathbf{y},\mathbf{x}_i) = \mathbf{0}$ i = 1,..., p $\mathbf{g}(\mathbf{y}) \le \mathbf{0}$ $\mathbf{g}_i(\mathbf{y},\mathbf{x}_i) \le \mathbf{0}$ i = 1,..., p		Lagrangian separable when **y** is fixed.
Master Problem		Subproblem
$\min_{\mathbf{y}} f_O(\mathbf{y}) + \sum_{i=1}^{p} f_i(\mathbf{y}, \mathbf{x}_i(\mathbf{y}))$ s.to: $\mathbf{h}(\mathbf{y}) = \mathbf{0}$ $\mathbf{g}(\mathbf{y}) \le \mathbf{0}$	$\mathbf{y} \Rightarrow$ $\Leftarrow \mathbf{x}$	$\min_{\mathbf{x}_i} f_i(\mathbf{y}, \mathbf{x}_i)$ s.to: $\mathbf{h}_i(\mathbf{y}, \mathbf{x}_i) = \mathbf{0}$ $\mathbf{g}_i(\mathbf{y}, \mathbf{x}_i) \le \mathbf{0}$. i = 1,..., p

Step 3 is the essence of a robust algorithm. Kirsh offers no formal algorithm for step 3 but presents several examples which use analytical solutions $\mathbf{x}_*(\mathbf{y})$ to determine directions in **y.** Algorithm 6 serves as a generic algorithm to describe feasible decomposition methods which utilize separability in the objective and constraints. If explicit subproblem solutions, $\mathbf{x}_{*i}(\mathbf{y})$, exist and are used in Step 3, Johnson's multistage decomposition method [1984b] can be interpreted as an implementation of Kirsch's method. Also, the algorithms of Rosen [1963], Benders [1962], and Azarm and Li [1988] can be viewed as special forms of this generic algorithm. These are described next.

The specific form of the original problem for Rosen's method is given in Table 7. The feature of this structure is that for fixed **y**, the problem decomposes into *p* independent linear programs in $\mathbf{x}_i$ given as the subproblems in Table 7. By solving the linear programs independently and constructing a piece-wise linear approximation of $\mathbf{x}_i$ (**y**) in the subproblem, a master problem is constructed strictly as function of **y**. Solution of the linear subprograms with the simplex method yields optimal basis matrices and simplex multipliers that are used to set up a master problem in **y**. The algorithm converges for the case when the cost vector **c** and the constraint matrix **A** are independent

of **y**, and the constraints **b**(**y**) are convex. Algorithm 7 summarizes Rosen's method.

Table 7

Rosen's Decomposition

Rosen		
Original Problem		Properties/Transformations
$\min_{\mathbf{y}\,\mathbf{x}} \sum_{i=1}^{p} \mathbf{c}_i(\mathbf{y})^T \mathbf{x}_i + c_o(\mathbf{y})$ s. to: $\mathbf{b}_i(\mathbf{y}) - \mathbf{A}_i(\mathbf{y})\ \mathbf{x}_i \le \mathbf{0}$ i = 1, ..., p. $\mathbf{y} \in S$ feasible domain)		Linear in $\mathbf{x}_i$; i = 1, ..., p. Define basis, **B**, non-basis, **D**. $\mathbf{A}_i = \begin{bmatrix} \mathbf{B}_i \\ \mathbf{D}_i \end{bmatrix}$ $\mathbf{b}_i = \begin{bmatrix} \mathbf{b}_{Bi}(\mathbf{y}) \\ \mathbf{b}_{Di}(\mathbf{y}) \end{bmatrix}$ $\mathbf{Q}_i = \mathbf{D}_i\ \mathbf{B}_i^{-1}$. Linear approximation of constraints in **y**.
Master Problem		Subproblem
$\min_{\mathbf{y}} \varphi(\mathbf{y}) = \sum_{i=1}^{p} \boldsymbol{\lambda}*_i{}^T \mathbf{b}_{Bi}(\mathbf{y}) + c_o(\mathbf{y})$ $[\mathbf{Q}_i\ \nabla_{\mathbf{y}}\, \mathbf{b}_{Bi}(\mathbf{y}) - \nabla_{\mathbf{y}}\, \mathbf{b}_{Di}(\mathbf{y})]\,(\mathbf{y} - \mathbf{y}^k)$ $\ge \mathbf{b}_{Di}(\mathbf{y}^k) - \mathbf{Q}_i\, \mathbf{b}_{Bi}(\mathbf{y}^k)$	$\mathbf{y} \Rightarrow$ $\Leftarrow$ $\boldsymbol{\lambda}*_i$	$\min_{\mathbf{x}} \mathbf{c}_i^T \mathbf{x}_i$ s. to: $\mathbf{b}_i(\mathbf{y}) - \mathbf{A}_i(\mathbf{y})\ \mathbf{x}_i \le \mathbf{0}$. LP solution: $\mathbf{x}*_i = \mathbf{B}_i^{-1}\ \mathbf{b}_{Bi}(\mathbf{y})$ $\boldsymbol{\lambda}*_i = \mathbf{c}_i^T \mathbf{B}_i^{-1}$

Algorithm 7**:** Coordination Strategy for Rosen's Decomposition.

1. Initialize k=0; choose a feasible vector $\mathbf{y}^k = \mathbf{y}^o \in S$.
2. (Subproblem) Solve the linear subproblems with $\mathbf{y} = \mathbf{y}^k$, obtaining optimal solutions $\mathbf{x}*_i$, basis matrices $\mathbf{B}_i$ and simplex variables $\boldsymbol{\lambda}*_i$
3. (Master Problem) Solve the master problem to obtain $\mathbf{y}_*$.
4. (Optimality Test) If $\varphi(\mathbf{y}_*)$ converged,
   and all simplex multipliers are non-negative the solution is optimal. Stop.
   Otherwise the *ith* linear subproblem containing the least simplex multiplier contains the non-basic variable to be used in updating the basis matrix. Update and return to 3.
5. (Feasibility Test) If $\varphi(\mathbf{y}_*) < \varphi(\mathbf{y}^k)$ check feasibility.
   (a) If no constraints are violated, set k = k+1, return to 2 with $\mathbf{y}^k = \mathbf{y}_*$
   (b) If some constraints are violated, perform line search into feasible region and update
   $\mathbf{y}_*$ based on line search.
   (c) If line search is unsuccessful, the violated constraint identifies non-basic variable for
   basis update. Using this new basis return to step 3.

Table 8

Benders' Decomposition

Benders		
Original Problem		Properties/Transformations
$\min_{\mathbf{y}\,\mathbf{x}} \mathbf{c}^T\mathbf{x} + f(\mathbf{y})$ s. to: $\mathbf{b} - \mathbf{A}\mathbf{x} - \mathbf{F}(\mathbf{y}) \le 0$ $\mathbf{x} \ge 0$ $\mathbf{y} \in S$.		Linear in $\mathbf{x}$ Formulate dual in $\mathbf{x}$ and solve using a relaxed constraint set.
Master Problem		Subproblem
$\min_{\mathbf{y}\,v_o} v_o + f(\mathbf{y})$ s. to: $v_o \ge (\mathbf{b} - \mathbf{F}(\mathbf{y}))^T \boldsymbol{\mu}_i^p \quad i \in I_p'$ (for some extreme points) $0 \ge (\mathbf{b} - \mathbf{F}(\mathbf{y}))^T \boldsymbol{\mu}_i^r \quad 0 \quad i \in I_r'$ (for some extreme rays)	$\mathbf{y}, v_o \Rightarrow$ $\Leftarrow \boldsymbol{\mu} I_r'$ $, I_p'$	$v(\mathbf{y}) = \max_{\boldsymbol{\mu}} (\mathbf{b} - \mathbf{F}(\mathbf{y}))^T \boldsymbol{\mu}$ s. to: $\mathbf{A}^T \boldsymbol{\mu} \le \mathbf{c}$ $\boldsymbol{\mu} \ge 0$

Benders [1962] proposed Algorithm 8 for problems of the original form in Table 8. The matrix $\mathbf{A}$ is $m \times n$, $\mathbf{x}$ and $\mathbf{c}$ are n-vectors, $\mathbf{b}$ an m-vector, $\mathbf{y}$ a n_y-vector, f a scalar valued function of $\mathbf{y}$, $\mathbf{F}$ is an m-vector of scalar functions dependent only on $\mathbf{y}$, and S is an arbitrary subset of R^{n_y}. No assumptions are made about linearity with respect to $\mathbf{y}$, nor continuity on S. The set S may also be discrete valued. The algorithm exploits the $\mathbf{y}$-dependence of the constraint set to formulate a subproblem relating $\mathbf{y}$ and $\mathbf{x}$. The linearity with respect to $\mathbf{x}$ in the subproblem is exploited to formulate a dual subproblem in terms of extreme points. A relaxation strategy is used to reduce the dimensionality of the dual formulation.

Algorithm 8**:** Coordination Strategy for Benders' Decomposition.

1. Initialize the index sets of the extreme rays and extreme points to the empty set.
Solve the master problem with respect to $\mathbf{y}$ and v_o.
   (a) If it is infeasible, stop; original is infeasible.
   (b) If it is finite, it is optimal. Stop.
   (c) If it is unbounded, go to 3.
3. (Subproblem) Solve dual form of the subproblem wrt $\boldsymbol{\mu}$ ($\mathbf{y}$ is fixed) and obtain $v'(\mathbf{y})$.
   (a) If dual is infeasible stop; original is unbounded.
   (b) If dual is unbounded go to 6.
   (c) Go to 4.

4.(Subproblem Optimality Test)

(a)If $v_*(\mathbf{y}) = v_O$, Stop. The current vector, $(\mathbf{y}, \mathbf{x})$ solves original prolem.

(b) Otherwise $v_*(\mathbf{y}) < (\mathbf{b} - \mathbf{F}(\mathbf{y}))^T \boldsymbol{\mu}_*$ for some multipliers. Go to 5.

5. (Dual is bounded but Optimality Test failed)

(a) Update index set, I_p and constraint on v_O to $v_O \geq (\mathbf{b} - \mathbf{F}(\mathbf{y}))^T \boldsymbol{\mu}_*$ and return to

6.(Dual unbounded) The simplex method will locate some extreme ray and an extreme point such that the dual objective approaches $+\infty$ along a half-line of the two. Add the ray index to the master problem. In addition, if the constraint containing the extreme point is violated, add the point index to the master problem. Return to (2).

Table 9

Azarm and Li's Decomposition.

Azarm and Li		
Original Problem		Properties/Transformations
$\min_{\mathbf{y}\,\mathbf{x}} f_O(\mathbf{y}) + \sum_{i=1}^{p} f_i(\mathbf{y}, \mathbf{x}_i)$ s. to: $\mathbf{g}(\mathbf{y}) \leq \mathbf{0}$ $\mathbf{g}_i(\mathbf{y},\mathbf{x}_i) \leq \mathbf{0}$ i = 1, ..., p		Monotonic in $\mathbf{x}_i$
Master Problem		Subproblem
$\min_{\mathbf{y}} f_O(\mathbf{y}) + \sum_{i=1}^{p} f_i(\mathbf{y}, \mathbf{x}*_i(\mathbf{y}))$ s. to: $\mathbf{g}(\mathbf{y}) \leq \mathbf{0}$	$\mathbf{y} \Rightarrow$ $\Leftarrow \mathbf{x}*(\mathbf{y})$	$\min_{\mathbf{x}_i} f_i(\mathbf{y}, \mathbf{x}_i)$ s. to: $\mathbf{g}_i(\mathbf{y},\mathbf{x}_i) \leq \mathbf{0}$ i = 1, ..., p

If the NLP in Equation (5) has no equality constraints and is monotonic in $\mathbf{x}_i$, in *both* the objective function *and in every* constraint of the *ith* subproblem, Azarm and Li's Monotonicity Based Decomposition Method (MBDM) can be used. Table 9 gives the master problem and subproblems. Global monotonicity properties surface in the subproblem that can be exploited when $\mathbf{y}$ is fixed. Problems with only linear terms in $\mathbf{x}$ and cross terms $\mathbf{y}\,\mathbf{x}$ can readily exploit this method. The method also assumes the number of active constraints is exactly equal to the dimension of $\mathbf{x}_i$. This assumption may preclude use of the method on many problems. The partition allows elimination of known inactive constraints (in the subproblems) reducing the dimensionality of the master problem. In this context it may be viewed as a special model reduction technique. The strategy is summarized in Algorithm 9.

Algorithm 9: Coordination Strategy for Azarm and Li's Decomposition.

1. Initialize k = 0, $\mathbf{y}^k = \mathbf{y}^0$.
2. (Subproblem) Use monotonicity to solve subproblems and obtain $\mathbf{x}_{*i}(\mathbf{y})$.
3. (Master Problem) Solve the master problem with respect to **y** using a conventional method to obtain $\mathbf{y}_*$.
4. If f converges stop; otherwise, increment k, set $\mathbf{y}^k = \mathbf{y}_*$ return to 2.

The key difficulty with the method is that the monotonicity analysis usually yields more than one candidate active constraint in step 2. If, for the *ith* subproblem, there are K_i candidate active sets, and $\mathbf{q}_{ij}(\mathbf{y})$ is the solution vector resulting from the *jth* candidate active set for the *ith* subproblem, the subproblem solution $\mathbf{x}_{*i}(\mathbf{y})$ can be expressed as

$$\mathbf{x}_{*i}(\mathbf{y}) = \max \{\mathbf{q}_{ij}(\mathbf{y}); j = 1, ..., K_i\}. \qquad (6)$$

Evaluation of the max function in the subproblem, requires an estimate of **y**. Moves in **y** in the master problem may invalidate $\mathbf{x}_{*i}(\mathbf{y})$; evaluating the max function in the master problem may introduce discontinuities, a real difficulty for a conventional SQP algorithm.

Three strategies could preclude this. First, the master problem could be formulated as a mixed discrete problem where, the candidate solutions,

$$\mathbf{q}_{ij}(\mathbf{y}); j = 1, ..., K_i \qquad (7)$$

are treated as discrete variables. Second, the max function could be evaluated in the subproblem and appropriate move limits on **y** also be computed and returned to the master problem. Third, the candidate active constraints could be represented by a K-S function (Kreisselmeier-Steinhauser [1979]) which asymptotically envelopes the dominant constraint.

The MBDM is very similar to Johnson's Multi-Stage Decomposition. Johnson suggested solving the subproblems analytically and the master problem either analytically or using a conventional numerical method. If monotonicity analysis is the method for the analytical solution in the subproblem and an SQP method solves the master problem, the

MBDM can viewed as an implementation of Johnson's multistage method on problems with special structure.

Wismer and Chattergy [1978] proposed a method where the objective is separable with respect to $\mathbf{x}_i$ but the constraints are coupled. The original problem, the master problem, and the subproblems are given in Table 10.

Table 10

Wismer and Chattergy's Decomposition

Wismer and Chattergy		
Original Problem		Properties/Transformations
$\min_{\mathbf{x}_1 \ldots \mathbf{x}_p} \sum_{i=1}^{p} f_i(\mathbf{x}_i)$ s. to: $\mathbf{g}_i(\mathbf{x}) \le \mathbf{0}$ $\mathbf{x}_i \in R^{n_i}$ for i = 1,..., p.		Objective separable Highly coupled constraints Lagrangian minimization Introduce **y** $\mathbf{h}(\mathbf{x}, \mathbf{y}) = \mathbf{x} - \mathbf{y} = \mathbf{0}$ $\mathbf{g}_i(\mathbf{x}) = \mathbf{g}_i(\mathbf{x}_i, \mathbf{y}_1, .., \mathbf{y}_j, .., \mathbf{y}_p)$ $j \neq i$
Master Problem		Subproblem
$\min_{\mathbf{y}} \sum_{i=1}^{p} L_i(\mathbf{y}, \mathbf{x}*, \boldsymbol{\lambda}*, \boldsymbol{\mu}*)$	$\mathbf{y} \Rightarrow$ $\Leftarrow \mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}, \frac{\partial \mathbf{g}}{\partial \mathbf{y}}$	$\min_{\mathbf{x}_i \, \boldsymbol{\lambda}_i \, \boldsymbol{\mu}_i} L_i(\mathbf{y}*, \mathbf{x}_i, \boldsymbol{\lambda}_i, \boldsymbol{\mu}_i)$

Figure 10 shows the FDT for Wismer and Chattergy's method. The strategy begins by introducing a vector of new variables, **y**, and a vector of equality constraints, $\mathbf{h}(\mathbf{x}, \mathbf{y}) = \mathbf{x} - \mathbf{y} = \mathbf{0}$, both of dimension *n*. The vector **h** is partitioned into *p* vectors, $\mathbf{h}_i$, and **y** is partitioned into *p* vectors $\mathbf{y}_i$ each associated with vector $\mathbf{x}_i$. The vectors $\mathbf{h}_i$ and $\mathbf{y}_i$ are each of dimension n_i . Since $\mathbf{x}_j = \mathbf{y}_j$, for j = 1,..., *p*, the following transformation holds for the *ith* vector of inequality constraints,

$$\mathbf{g}_i(\mathbf{x}) = \mathbf{g}_i(\mathbf{x}_i, \mathbf{y}_1, .., \mathbf{y}_j, .., \mathbf{y}_{p;\, j\neq i}) = \mathbf{g}_i(\mathbf{x}_i, \mathbf{y}_{j;\, j=1, ..., p;\, j\neq i}). \qquad (8)$$

To formulate the Lagrangian, introduce *p* vectors of Lagrange multipliers $\boldsymbol{\lambda}_i$ associated with $\mathbf{h}_i$ and introduce *p* vectors of Lagrange multipliers $\boldsymbol{\mu}_i$ associated with $\mathbf{g}_i$. Defining Lagrange functions for the *ith* partition,

$$L_i(\mathbf{x}_i, \mathbf{y}, \boldsymbol{\lambda}_i, \boldsymbol{\mu}_i) = f_i(\mathbf{x}_i) + \boldsymbol{\lambda}_i^T \mathbf{h}_i(\mathbf{x}_i, \mathbf{y}_i) + \boldsymbol{\mu}_i^T \mathbf{g}_i(\mathbf{x}_i, \mathbf{y}_{j;\, j=1, .., p;\, j\neq i}) \qquad (9)$$

the Lagrangian for the original problem is the sum,

$$L(\mathbf{x}, \mathbf{y}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = \sum_{i=1}^{p} L_i(\mathbf{x}_i, \mathbf{y}, \boldsymbol{\lambda}_i, \boldsymbol{\mu}_i) \tag{10}$$

which effects p independent Lagrangians when $\mathbf{y}$ is fixed. A conventional SQP algorithm can usually solve this subproblem and provide $(\mathbf{x}_*, \boldsymbol{\lambda}_*, \boldsymbol{\mu}_*)$. The master problem is to minimize L with respect to $\mathbf{y}$. A gradient can be employed to reduce L on each iteration in the master problem,

$$\mathbf{y}_i^{k+1} = \mathbf{y}_i^{k} - \alpha \left[\sum_{j=1}^{p} \frac{\partial \mathbf{g}_j}{\partial \mathbf{y}_i}^T \boldsymbol{\mu}_j - \boldsymbol{\lambda}_i \right] \tag{11}$$

with $\alpha > 0$.

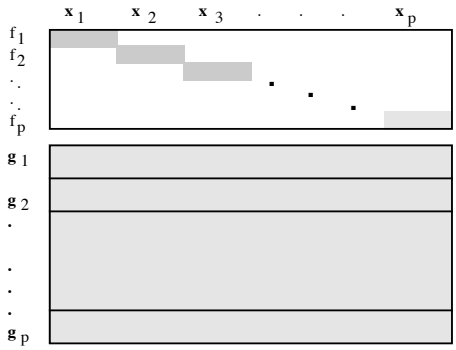


Figure 10. FDT schematic for Wismer and Chattergy's Decomposition Method.

Algorithm 10: Coordination Strategy for Wismer and Chattergy's Decomposition.

1. Initialize k= 0, feasible $\mathbf{x} = \mathbf{x}^k$, and $\mathbf{y}^k = \mathbf{x}^k$.
2. Holding $\mathbf{y}^k$ constant, solve subproblems using any method to obtain, $(\mathbf{x}_{*i}, \boldsymbol{\lambda}_{*i}, \boldsymbol{\mu}_{*i})$
3. Update $\mathbf{y}$ in master problem using Equation (11).
4. (Convergence test) If $\| f(\mathbf{y}_*) - f(\mathbf{y}^k) \| < \varepsilon$, stop;
   Otherwise increment k and go to (2).

Advantages to this method are that it can handle highly coupled constraint sets and the master problem converges if local solutions are feasible. When local feasible solutions cannot be found, the most recent feasible solution, while perhaps not optimal, has reduced the Lagrangian. A line search in Step 3, instead of a fixed α may also be used.

One disadvantage is that the introduction of **y** doubles the number of variables in the problem. Another drawback is that certain partitions of **g** may result in unbounded subproblems. Subproblems need to be checked for boundedness as part of the subproblem solution.

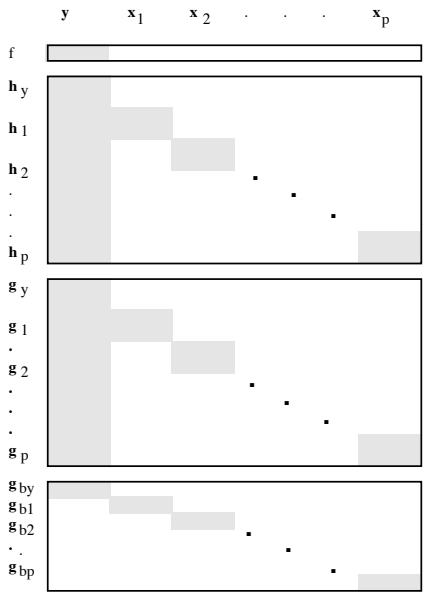


Figure 11. FDT schematic for Sobieski's Decomposition Method.

Sobieski [1982] proposed a method where the constraints can be partitioned with respect to $\mathbf{x_i}$ but the objective is not separable. The FDT structure is shown in Figure 11 and the problem form is given in Table 11. The objective function is only a function of global variables **y**. Note that partitioning is not driven by any structure in the objective function. Local variables, $\mathbf{x_i}$, appear only in the constraint set. The constraint set is partitioned by the incidence of the local variables. Such structure occurs when design objectives can be expressed as functions of many detailed variables, but all variables need not appear explicitly in the objective function. Problems where the objective is expressed in terms of behavior variables which in turn are functions of geometric variables also tend to have this structure.

Table 11

Sobieski's Decomposition

Sobieski		
Original Problem		Properties/Transformations
$\min_{\mathbf{y}\,\mathbf{x}_1 \ldots \mathbf{x}_p} f(\mathbf{y})$ s. to: $\mathbf{h}(\mathbf{y}) = \mathbf{0}$ $\mathbf{g}(\mathbf{y}) \le \mathbf{0}$ $\mathbf{g}_b(\mathbf{y}) \le \mathbf{0}$ $\mathbf{h}_i(\mathbf{y}, \mathbf{x}_i) = \mathbf{0}$ $\mathbf{g}_i(\mathbf{y}, \mathbf{x}_i) \le \mathbf{0}$ $\mathbf{g}_{bi}(\mathbf{x}_i) \le \mathbf{0}$ for i = 1, ..., p.		Non-separable opjective Separable constraints
Master Problem		Subproblem
$\min_{\mathbf{y}} f(\mathbf{y})$ s. to: $\mathbf{h}(\mathbf{y}) = \mathbf{0}$ $\mathbf{g}_b(\mathbf{y}) \le \mathbf{0}$ $\mathbf{g}_{bi}(\mathbf{x}_{*i}(\mathbf{y})) \le \mathbf{0}$ $G(\mathbf{y}) = \sum_{j=1}^{ng} \langle g_j(\mathbf{y}) \rangle^2 + \sum_{j=1}^{p} P_{*i}(\mathbf{y}) - P_t^k \le 0$.	$\mathbf{y} \Rightarrow$ $\Leftarrow$ $P_{*i}(\mathbf{y})$, $\mathbf{x}_{*i}(\mathbf{y})$	$\min_{\mathbf{x}_i} P_i(\mathbf{y}, \mathbf{x}_i) = \sum_{j=1}^{ngi} \langle g_{ij}(\mathbf{y},\mathbf{x}_i) \rangle^2$ s. to: $\mathbf{h}_i(\mathbf{y}, \mathbf{x}_i) = \mathbf{0}$ $\mathbf{g}_{bi}(\mathbf{x}_i) \le \mathbf{0}$ where $\langle g_{ij}(\mathbf{y},\mathbf{x}_i) \rangle = 0$ for $g_{ij}(\mathbf{y},\mathbf{x}_i) \le 0$ $= g_{ij}(\mathbf{y},\mathbf{x}_i)$ for $g_{ij}(\mathbf{y},\mathbf{x}_i) > 0$.

Since the objective function is independent of $\mathbf{x}$, it appears only in the master problem. In the spirit of penalty methods, an objective function is formed for each subproblem, where a measure of constraint violation is minimized. Assuming $\mathbf{g}_i(\mathbf{y}, \mathbf{x}_i)$ is a vector of scalar-valued functions defined on R^{ngi}, a penalty associated with the *ith* subproblem, $P_i(\mathbf{y}, \mathbf{x}_i)$, is minimized as shown in the subproblem of Table 11. If the sensitivity derivatives,

$$\nabla_{\mathbf{y}} P_i(\mathbf{y}, \mathbf{x}_*) \qquad (12)$$

$$\nabla_{\mathbf{y}} \mathbf{x}_*$$

are also computed when solving the subproblem, a linear approximation of the subproblem solution can be constructed for use in the master problem. The algorithm is summarized in Algorithm 11. Note the bounds on $\mathbf{x}_i$ are included in the master problem. No proof of convergence is given, but Sobieski suggests the addition of move limits on $\mathbf{y}$ to keep the linear approximations valid.

Algorithm 11: Coordination Strategy for Sobieski's Decomposition.

1. Initialize k = 0 and all variables $\mathbf{y} = \mathbf{y}^k$ and $\mathbf{x}_i = \mathbf{x}_i^k$ for i = 1, ..., p.
2. (Subproblem) For a given $\mathbf{y}^k$, solve the *p* independent subproblems with respect to $\mathbf{x}_i$ to obtain $\mathbf{x}_{*i}^k$.
   (a) Compute sensitivity derivatives of the subproblem solution with respect to **y**, $\nabla_{\mathbf{y}} \mathbf{x}_{*i}^k$, $\nabla_{\mathbf{y}} P_i(\mathbf{y}, \mathbf{x}_{*i}^k)$. (Finite difference is suggested.)
   (b) Using the sensitivity derivatives computed in (a), construct a linear representation of
   $P_i(\mathbf{y}^k, \mathbf{x}_{*i}^k)$ and $\mathbf{x}_{*i}^k$ about the point $\mathbf{y}^k$ yielding $\mathbf{x}_{*i}^k(\mathbf{y})$ and $P_i(\mathbf{y}, \mathbf{x}_{*i}^k(\mathbf{y}))$.
3. (Master Problem) Solve the master problem with respect to **y** and obtain solution $\mathbf{y}_*$
4. (Termination Test) If $\|f(\mathbf{y}_*) - f(\mathbf{y}^k)\| \le \varepsilon$ and $G(\mathbf{y}) \le \varepsilon_g$ stop; otherwise set $\mathbf{y}^{k+1} = \mathbf{y}_*$, $\mathbf{x}_i^{k+1} = \mathbf{x}_{*i}^k$ and k = k+1. Return to 2.

The method can be interpreted as Langrangian decomposition in the context of penalty methods. If all constraints except for simple bounds and equalities are represented with exterior penalty functions in a modified objective, the Lagrangian of the penalty formulation is additively separable with respect to $\mathbf{x}_i$. The master problem minimizes this Lagrangian with respect to **y** and the subproblems independently minimize each of the terms in the Lagrangian with respect to $\mathbf{x}_i$.

Sobieski et al. [1985] used the method solve a three member portal framework problem. The decomposed solution compared favorably to the solution obtained without decomposition. In the portal frame problem, the K-S cumulative constraint (Kresselmeier-Steinhauser, [1979]) was used instead of the penalty function given in Table 11.

Parksinson et al. [1987] evaluated Sobieski's decomposition method on several test problems from Stoecker [1981] but with a different approximation scheme. Using Design of Experiment techniques to determine 'test' points in the **y** space, they approximated the subproblem solution dependence on **y** with regression equations. The

objectives of problems solved with decomposition were within a few per cent of the non-decomposed solutions.

Table 12

Summary of Defining Properties for Prevalent Hierarchical Decomposition Methods

	Original Problem		Master Problem		Subproblem	Convergence conditions
Method	Objective	Constraint	Manipulation	Solution	Solution	
Dantizig-Wolfe (Dual)	Linear Program	Some highly coupling constraints	Formulatie dual wrt coupling constraints	Outer linearization Relaxation	Simplex	Yes
Takahashi (Dual)	Convex	Equalities; some highly coupling constraints	Formulate dual wrt coupling constraints	Feasible Directions	Conventional	Yes
Lasdon (Dual)	Additively Separable	Each constraint is additively separable	Formulate dual	Gradient method; or tangential approx.	Conventional	Yes
Wismer-Chattergy (Dual)	Additively separable	Highly coupled	For p partitions introduce k- vectors **y**,**h**. $k = n^p - n$.	Gradient for dual variables usingKKT conditions.	Conventional	Yes, if subproblems stay feasible;
Diaz-Belding (Dual-like)	Goal Programs	Linking variables **y**		Conventional NLP (GRG)	Conventional NLP (GRG); Sensitivity wrt to **y**	Not guaranteed
Johnson (Feasible)	No special properties	No special structure	Projection	Analytical or Numerical	Analytical; Suggests M.O.D.	Yes, if subproblem is single valued.
Kirsch (Feasible)	Additively Separable	Linking variables **y**	Projection	No suggestions		Not guaranteed
Rosen (Feasible)	Linear in **x**	Linear in **x**	Projection	Piecewise linear wrt **y**.	Simplex Method	Yes, if **A**,**c** are indepenndent of **y.**
Benders (Feasible)	Linear in **x**.	Linear in **x**	Projection	Outer linearization/ Relaxation	Dual or Primal Method	Yes, explicitly for assumed form.
Azarm-Li (Feasible)	Additively separable monotonic wrt **x**.	Linking variables **y**	Projection	SQP	Monotonicity Analysis	Yes, if subproblem are single valued functions. (single step)
Wismer-Chattergy (Feasible)	Additively Separable	Highly Coupled	Introduce n-vector **y**	Gradient Method (using KKT)	Conventional	Yes, if subproblems stay feasible
Sobieski (Feasible)	Not dependent on **x**	Linking variables **y**	Penalty formulation of linking constraints	Conventional NLP method	Conventional; Sensitvity analysis wrt **y.**	Not guaranteed but move limits can help.

The hierarchical methods reviewed in this section are summarized in Table 12. Dual methods have the common defining property of linking functions; feasible methods have the common defining property of linking variables. Additional properties of

additive separability, linearity, and monotonicity determine the suitability of a given method for a given problem. The underlying assumption to the use of these methods is the structure of the original problem, represented here by the structure in the FDT, is known.

4 NON-HIERARCHICAL METHODS

The methods reviewed thus far deal with *independent* subproblems generated when either linking functions or linking variables are accommodated in the master problem. However, problems may have both linking functions and linking variables as the FDT's of Figure 12 illustrate. Using the methods described earlier, a three-level method could be used to decompose the NLP. Use of a dual method would account for the linking functions, and the subproblem would be further decomposed using a feasible decomposition method; such an approach is termed *dual-feasible*. Here, linking in the subproblems is accounted for by using a feasible method to further decompose the problem. Such a strategy is essentially a recursive decomposition, and could be invoked to any desired level.

If the problem is highly coupled as in Figure 12 (b), such a method may not succeed. To facilitate independent subproblems in such circumstances, approximations are required to account for the interdependence of the subproblems. Ritter [1967], Wismer and Chattergy [1978], Sobieski [1988], Pan and Diaz [1990], Wu [1991], and Unger et al. [1992] have proposed various methods to handle such coupling. Such methods have acquired the name *non-hierarchical* decomposition because the *information coupling* is non-hierarchical. Note, however, that each method has a coordination strategy that is strictly hierarchical. The approximations are effectively 'cuts' that allow non-hierarchical information coupling to be handled hierarchically.

The discussion that follows considers linear programs first. Ritter [1967] proposed a method for NLP's with the structure in Figure 12 (a) which are linear in **x**, with linking variables **y**, and with linking constraints. The subproblem is the linear program in **x** with **y** fixed. Solution of the subproblem yields basis vectors $\mathbf{x}_{ib}$ and non-basis vectors $\mathbf{x}_{id}$. The master problem is formulated as an NLP problem in **y** and non-basis vector $\mathbf{x}_{id}$. The master problem is interpreted as a relaxation strategy because explicit elimination of $\mathbf{x}_{ib}$ from the master problem effects relaxation of the constraints on $\mathbf{x}_{ib}$. Under explicit conditions, convergence is guaranteed and computational experience with the method is given in Grigoriadis and Ritter[1968].

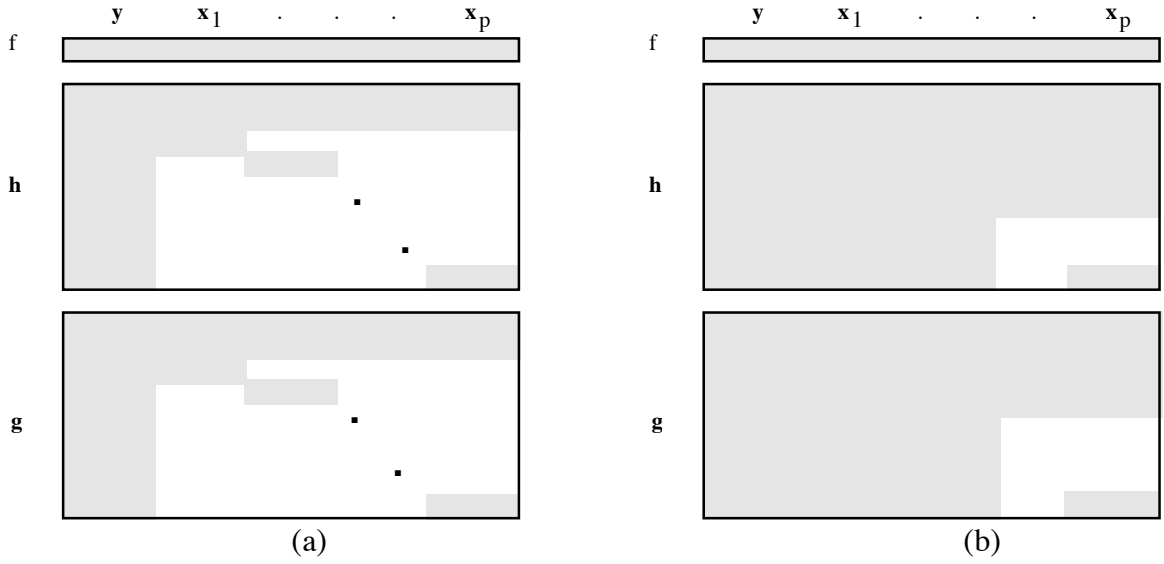


Figure 12. FDT for NLP with linking functions and linking variables. (a) low coupling. (b) high coupling.

Wismer and Chattergy [1978] suggested a relaxation strategy for decomposing highly coupled constraint sets in problems with an additively separable objective in $\mathbf{x}_i$. Introduction of an *n*-vector of variables **y** an *n*-vector of constraints,

$$\mathbf{h}(\mathbf{y}, \mathbf{x}) = \mathbf{y} - \mathbf{x} = \mathbf{0}, \tag{13}$$

and the KKT conditions yield an explicit relationship for the multipliers,

$$\boldsymbol{\lambda}_i = \sum_{j=1}^{p} \frac{\partial \mathbf{g}_j}{\partial \mathbf{y}_i}^T \boldsymbol{\mu}_j \qquad (14)$$

In each iteration of the coordination strategy, the master problem sets $\mathbf{y} = \mathbf{x}_*$ and computes $\boldsymbol{\lambda}$ from Equation (14). Clearly, the method requires the sensitivity derivatives $\partial \mathbf{g}/\partial \mathbf{y}$ in each subproblem. These derivatives are the mechanism which accounts for subproblem coupling.

Pan and Diaz [1990] presented an algorithm for decomposing a highly coupled NLP problem as follows. The variables are partitioned into *p* arbitrary partitions $\mathbf{x}_i$ and *p* subproblems are defined. The *ith* subproblem solves the entire NLP with respect to $\mathbf{x}_i$ holding all other variables fixed. Results using the first order sensitivity theorem (Fiacco [1983]) are employed to develop a scheme for sequencing the solution of the subproblems. The master problem examines the constraint activity in each subproblem and identifies a constraint active in *all* subproblems. The subproblem indices are ranked according to the value of the Lagrange multiplier associated with such a constraint. The sequence for the next iteration of subproblems is based on this ranking. The authors report two examples where the method was effective in solving a highly coupled NLP, but caution that experience is still limited. The method uses the Lagrange multiplier as a measure of a variable partition's effect on the entire problem and gives that method priority in the sequence of the subproblems.

Sobieski [1988, 1990] proposed a method for highly coupled problems which makes no assumptions about separability of the objective. Multi-disciplinary problems where the input-output relationships for function evaluations are defined a priori motivated the method. The method can take advantage of discipline-dependent optimization strategies (for example, optimality criteria in structural optimization). Treating input as the independent variables and output as the dependent variables for a given function, the implicit function theorem is used to derive the Global Sensitivity

Equations (GSE), (Sobieski, [1990]) which provide the derivative of the output with respect to the input at a constant function value. If a given output is required as input in another function, the derivatives facilitate decoupling of the two functions. The GSE approach decouples the constraint sets.

The constraint vector is partitioned into *p* subvectors based on physical subsytems and each sub-vector is represented with a single cumulative constraint. The variables are partitioned based on effectiveness coefficients (Hajela and Sobieski [1981]) which quantify the impact of the variable on reducing the objective and each cumulative constraint. Also, a *p*x*p* matrix of 'responsibility' coefficients **r** weight the effect of the *ith* variable partition on reducing the *jth* cumulative constraint violation. For the feasible regions, a *p*x*p* matrix of 'tradeoff' coefficients **t** weight the effect of the *ith* variable partition on increasing the *jth* cumulative constraint. The system NLP is approximated with the original objective, the *p* cumulative constraints, and the bound constraints. The *ith* subproblem minimizes the system objective with respect to $\mathbf{x}_i$ subject to a *p* vector of cumulative constraints modified with a *p*-vector of weights defined in terms of **r** and **t**. These are fixed in the subproblems. The 'coordination' problem minimizes the system objective with respect to the coefficients **r** and **t**. The coefficients **r** and **t** coordinate all subproblem coupling in the master problem. In this context, the method bears resemblance to a dual method.

Wu [1991] proposed a method for highly coupled problems which requires no optimization in the subproblems. The method begins with partitioning the original variables into coordinating variables **y** and local variables **x**. The local variables **x** are further partitioned into *p* partitions. Partitioning by discipline (aspect decomposition) or by component (object decomposition) is suggested but the partitions can also be arbitrary. Similarly, the constraints are partitioned into *p* sets and the *ith* set of constraints is represented in the master problem with a single cumulative constraint, Ω_i.

A *p*-vector of multipliers $\boldsymbol{\alpha}$ is introduced where the scalar α_i is associated with the *ith* partition of variables. The coordinating variables are ($\mathbf{y}$, $\boldsymbol{\alpha}$). The master problem minimizes the objective with respect to ($\mathbf{y}$, $\boldsymbol{\alpha}$) subject to the *p* cumulative constraints. A master problem iteration yields a vector of Lagrange multipliers, $\boldsymbol{\mu}$, associated with each cumulative constraint. The local variables are represented in the master problem through the vector $\boldsymbol{\alpha}$. At each iteration the subproblem simply computes the derivative,

$$\frac{df}{d\mathbf{x}_i} = \frac{\partial f}{\partial \mathbf{x}_i} + \frac{\partial \boldsymbol{\Omega}}{\partial \mathbf{x}_i}^T \boldsymbol{\mu} \tag{15}$$

and the local variables are represented in the master problem as

$$\mathbf{x}_i^{k+1} = \mathbf{x}_i^{k} - \alpha_i \frac{df}{d\mathbf{x}_i} \tag{16}$$

The method has several interesting features. It is a model reduction method; the cumulative constraints serve to reduce the dimensionality of the constraint set; the vector $\boldsymbol{\alpha}$ reduces the dimension of the local variable space. Second, no optimization is done in the subproblems; they only report gradient information. Third, the derivative in Equation (15) is a measure of the local variable's effect on the objective and the derivative's magnitude and sign is used to increase or decrease the value of the local variable. Fourth, the vector $\boldsymbol{\alpha}$ is analogous to a set of step-sizes in a line search algorithm. Fifth, the method needs no move limits. The method was evaluated with numerous small to medium size problems from the literature. The accuracy of the solutions depend primarily on the accuracy of the cumulative constraints. The method is derived on the assumption that there are no changes in constraint activity which may limit its generality.

Unger et al.[1992] proposed the use of *variable complexity* models to decompose the analysis sequence in a multi-disciplinary design problem. The problem of interest was the design of a transport wing for minimum weight which required both structural analysis and aerodynamic analysis. The aerodynamics are modeled by vortex-lattice theory and the structure is described at two levels of complexity. A simple algebraic

equation model of structural weight is first used in a numerical optimization procedure to obtain a design that approximately accounts for effects of wing geometry on structural weight. The design is then refined based on a more complex finite-element model for the wing structure. They demonstrate how variable level of detail in the iteration sequence leads to substantial savings in computational effort. They report a 75% reduction in CPU time with this approach. The method can be interpreted as a model reduction decomposition technique. A function of many variables is replaced with a function of few; the many finite element variables are reduced to a few geometric variables and the required function evaluation is replaced with a simpler evaluation: an algebraic expression.

5 SUMMARY

The report reviewed and compared coordination strategies for hierarchically and non-hierarchically coupled optimization problems. The coordination strategy is the fundamental element of a decomposition method and depends very strongly on the defining properties of the problem. Figure 13 summarizes the FDT structures for most of the methods reviewed. The figure illustrates the defining properties of linking functions, linking variables, and separability. Moving down the figure the problem coupling increases.

The coordination strategy of every method presented has an underlying generic strategy given in Algorithm 0. While coupling may be non-hierarchical, the coordination strategies are strictly hierarchical. A master problem coordinates the solution of the independent subproblems. The algorithms in the coordination strategy take explicit advantage of structure (found or imposed) in the primal problem statement. Dual methods accommodate linking functions by coordinating dual variables in the master

problem; feasible methods accommodate linking variables by coordinating them in the master problem; problems with linking functions and linking variables can be accommodated with a recursive decomposition or by incorporating a mechanism into the coordinating problem to account for coupling. That mechanism may be a Lagrange multiplier (Pan and Diaz [1990]), a sensitivity coefficient (Sobieski [1990]) or a simpler model (Unger et al. [1992]).

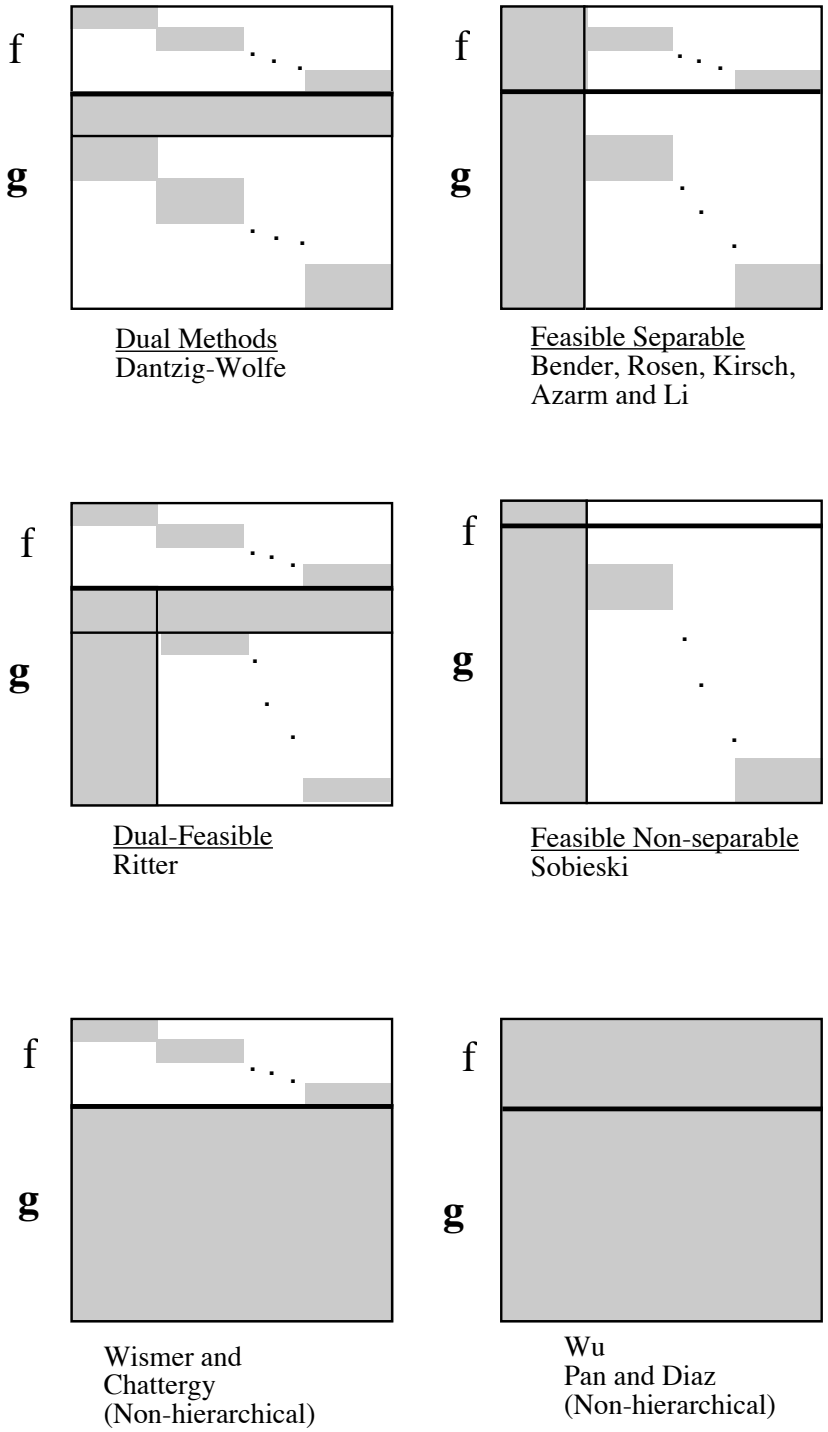


Figure 13. Summary of FDT structures for decomposition methods reviewed.

The design interpretation of the decomposition methods lies primarily in the authority model of the master problem. In dual methods, the master problem is setting priorities with the multipliers; in feasible methods the master problem is directly controlling those variables common to all subproblems. In problems with non-hierarchical coupling, the master problem coordinates independent subproblems through some measure of the subproblem effects on the system. The mechanisms vary, but several of the coordination strategies bear a strong resemblance to dual methods.

The assumption throughout this report is that the defining properties for decomposition were known. For the general NLP this is not easy to accomplish. An undirected graph representation of the FDT coupled with specific partitioning algorithms facilitates identification of defining properties in the general NLP. Wagner (1993) used a k-clique graph representation of the FDT and heuristics to find desired defining properties; Michelena and Papalambros (1995) used a hypergraph representation and spectral partitioning to find *optimal* defining properties. Krishnamachari and Papalambros (1996) used a bipartiite graph of the FDT and integer programming to find optimal defining properties.

REFERENCES

Azarm, S., and Li, W.C., 1988, "A Two-level Decomposition Method for Design Optimization", *Engineering Optimization*, Vol. 13, pp211-224.

Azarm, S., and Li, W.C., 1989, "Multi-level Design Optimization Using Global Monotonicity Analysis", *ASME Transactions, Journal of Mechanisms, Transmissions, and Automation in Design,* Vol. 111, pp259-263.

Benders, J.F., 1962, "Partitioning Procedures for Solving Mixed Variables Programming Problems", *Numerische Mathematik*, Vol. 4., pp. 238-252.

Bloebaum, C.L., 1991, "Formal and Heuristic System Decomposition Methods in Multidisciplinary Synthesis", *NASA Contractor Report 4413*.

Dantzig, G.B., and Wolfe, P., 1960, "Decomposition Principle for Linear Programming", *Operations Research*, Vol. 8, pp101-111.

Diaz, A.R., and Belding, B., 1991, "Modeling and Solution Strategies for Hierarchical Design Optimization", *Engineering Optimization*, Vol. 16, pp 247-273.

Geoffrion, A.M., 1971, "Large Scale Linear and Nonlinear Programming", *Optimization of Large Systems*, ed. D. A. Wismer, McGraw-Hill Book Company, New York.

Grigoriadis, M.D., and Ritter, K., 1968, "A Decomposition Method for Structured Linear and Nonlinear Programs", *Computer Sciences Technical Report !0*, Computer Sciences Department, University of Wisconsin.

Hajela, P., Sobieszczanski-Sobieski, J., 1981, "The Controlled Growth Method - A Tool for Structural Optimization", *Proceedings of the AIAA/ASME/ASCE/AHS 22nd Structures, Structural Dynamics, and Materials Conference*, Atlanta, GA.

Johnson, R. C., 1984, "A Method of Optimal Design Extended to Systems of Mechanical Elements Having Interrelated Boundary Effects", PhD Dissertation, Dept. of Mechanical Engineering, University of Rohester, Rochester, NY.

Johnson, R. C., Benson, R. C., 1984a, "A Basic Two-stage Decomposition Strategy for Design Optimization", *ASME Transactions, Journal of Mechanisms, Transmissions, and Automation in Design,* Vol. 106, pp380-386.

Johnson, R. C., Benson, R. C., 1984b, "A Multi-stage Decomposition Strategy for Design Optimization", *ASME Transactions, Journal of Mechanisms, Transmissions, and Automation in Design*, Vol. 106, pp. 387-393.

Krishnamachari, R., and Papalambros, P., "Optimal Hierarchical Decomposition Synthesis Using Integer Programming", *Advances in Design Automation,* 1996, ASME, New York.

Kirsch, U., 1981, *Optimum Structural Design*, McGraw-Hill Company, New York, NY.

Kreisselmeier, G., and Steinhauser, R., 1979, "Systematic Control Design by Optimizing a Vector Performance Index", *Proceedings of the IFAC Symposium on Computer Aided Design of Control Systems*, Zurich, Switzerland, pp. 113-117.

Lasdon, L.S., 1968, "Duality and Decomposition in Mathematical Programming", *IEEE Transactions of System Science and Cybernetics*, Vol. SSC-4, No.2, July.

Lasdon, L. S., 1970, *Optimization Theory for Large Systems,* The Macmillan Company, London.

Lootsma, F.A., and Ragsdell, K.M., 1988, "State-of-the-art in Parallel Non-linear Optimization", *Parallel Computing*, Vol. 6, pp. 133-155.

Lootsma, F.A., 1989, "Exploitation of Structure in Non-linear Optimization", *Parallel Computing*, Vol. 6, pp. 31-45.

Michelena, N., and Papalambros, P., "Optimal Model Based Decomposition of Powertrain System Design", ASME Journal of Mechanical Design, Vol. 117, No. 4, 1995, pp. 499-505.

Pan, J. and Diaz, A. R., 1990, "Some Results in Optimization of Non-hierarchic Systems", *ASME Journal of Mechanical Design*, Vol. 112, September pp 399 - 405.

Papalambros, P. Y. and Wilde, D.J., 1988, *Principles of Optimal Design: Modeling and Computation*, Cambridge University Press, New York, NY.

Parkinson, A., Balling, R., Wu, A., and Free, J., 1987, "A General Strategy for Decomposing Large Design Problems Based on Optimization and Statistical Test Plans", *Internation Conference on Engineering Design 87*, Boston.

Ritter, K., 1967, "A Decomposition Method for Linear Programming Problems with Coupling Constraints and Variables", Technical Report No.739, Mathematics Research Center, University of Wisconsin.

Rogers, J.L., 1989, "A Knowledge Based Tool for Multi-level Decomposition of a Complex Design Problem", *NASA TP 2903*, Langley Research Center, Hampton, VA.

Rosen, J.B., 1963, “Convex Partition Programming”, in *Recent Advances in Mathematical Programming*, pp 159-176, R. L. Graves and P.Wolfe, eds., McGraw-Hill, Inc. New York, NY.

Sobieszczanski-Sobieski, J., Barthelemy, J. F., and Riley, K. M., 1981 "Sensitivity of Optimum Solutions to Problem Parameters", *Proceedings of AIAA/ASME/ASCE/AHS 22nd Structures, Structural Dynamics and Materials Conference*, Atlanta, GA, April 6-8, AIAA Paper No. 81-0548.

Sobieszczanski-Sobieski, J., 1982, “A Linear Decomposition Method for Large Optimization Problems - Blueprint for Development”, *NASA TM83248*, Langley Research Center, Hampton, VA.

Sobieszczanski-Sobieski, J., James, B.B, and Dovi, A.R., 1985, “Structural Optimization by Multilevel Decomposition”, *AIAA Journal*, Vol. 23, November, pp. 1775-1782.

Sobieszczanski-Sobieski, J., and Haftka, R. T., 1987, " Multi-disciplinary and Multi-level Optimum Design", *Computer Aided Optimal Design: Structural and Mechanical Systems,* ed. C.A. Mota Soares, Springer-Verlag, Berlin.

Sobieszczanski-Sobieski, J., 1988, "Optimization by Decomposition: A Step from Hierarchic to Non-hierarchic Systems", *NASA T M 101494*, Langley Research Center, Hampton, VA.

Sobieszczanski-Sobieski, J., 1990, "Sensitivity of Complex, Internally Coupled Systems", *AIAA Journal*, Vol. 28., January, pp. 153-160.

Takahashi, I., 1964, “Variable Separation Principle for Mathematical Programming “, *Journal of the Operations Research Society of Japan,* Vol. 6, No. 1., pp 82-105.

Unger, E.R, Hutchison, M.G., Rais-Rohani, M., Haftka, R.T., and Grossman, B., 1992, “Variable Complexity Multi-disciplinary Design of a Transport Wing”, *Int'l Journal of Systems Automation: Research and Applications*, Vol. 2, pp. 87-113.

Wagner, T.C., *A General Decomposition Methodology for Optimal System Design*, Ph.D. Dissertation, Department of Mechanical Engineering and Applied Mechanics, University of Michigan, 1993.

Wagner, T.C., and Papalambros, P., 1993, "A General Framework for Decomposition Analysis in Optimal Design”, *ASME Advances in Design* Optimization, Vol. 2, 1993, pp. 315-325.

Watton, J.D., 1989, “Automatic Reformulation of Mechanical Design Constraints to Enhance Qualitative and Quantitative Design Decisions”, Ph.D. Dissertation, Department of Mechanical Engineering, Carnegie Mellon University, Pittsburgh, PA.

Wismer, D. A.(ed.), 1971, *Optimization Methods for Large Scale Systems,* McGraw-Hill Book Company, New York.

Wismer D.A, and Chattergy, R., 1978, *Introduction to Non-linear Optimization: A Problem Solving Approach*, North-Holland, New York, NY.

Wu, B.C., 1991, "Optimization Based Design of Non-hierarchical Systems", Ph.D. Dissertation, Dept. of Mechanical Engineering, University of Maryland, College Park, ML.