Eng. Opt., 1996, Vol. 27, pp. 109–137
Reprints available directly from the publisher
Photocopying permitted by license only

ACTIVITY ANALYSIS: SIMPLIFYING OPTIMAL DESIGN PROBLEMS THROUGH QUALITATIVE PARTITIONING†

BRIAN C. WILLIAMS[1] and JONATHAN CAGAN[2]

[1]*Computer Sciences Division, NASA Ames Research Center, Moffett Field, CA 94035, U.S.A*
[2]*Department of Mechanical Engineering, Carnegie Mellon University, Pittsburgh PA 15213, U.S.A*

(Received 1 May 1995; in final form 9 January 1996)

Activity analysis is introduced as a means to strategically cut away subspaces of a design problem that can quickly be ruled out as suboptimal. This results in focused regions of the space in which additional symbolic or numerical analysis can take place. Activity analysis is derived from a qualitative abstraction of the Karush-Kuhn-Tucker conditions of optimality, used to partition an optimization problem into regions which are nonstationary and qualitatively stationary (*qstationary*). Activity analysis draws from the fields of gradient-based optimization, conflict-based approaches of combinatorial satisfying search, and monotonicity analysis.

Keywords: design; symbolic optimization; Karush-Kuhn-Tucker conditions; qualitative Karush-Kuhn-Tucker conditions

1. INTRODUCTION

Engineering design problems, due to their complexity, are often best solved through numerical optimization. For nonlinear problems, numerical codes often require user guidance to quickly move towards

†A previous version of this paper was published in *DE-Vol. 83, 1995 Design Engineering Technical Conferences* (Design Theory and Methodology Conference), ASME, **2**: 455–463, 1995; ASME grants permission for *Engineering Optimization* to reprint that portion of this paper.

the global optima. However, it is difficult for an engineer to extract from the numerical solution insights about the subspace where the optimum lies. The design space may be vast; however, the global optimum, or potential local optima, can often be determined to lie within distinct regions of the space. Insight into where those regions lie not only reduces the (numerical) optimization problem considerably, but allows the designer to investigate the influences and sensitivities of different variables and constraints to the objective function through a detailed analysis within the reduced region. These investigations provide insight into more fundamental changes to a design such as modifying the device structure or topology.

New-generation CAD tools provide the capability to symbolically model and analyze design problems within the CAD environment. We propose a technique called *activity analysis* for reasoning about symbolic design models that provides the ability to computationally partition the design space into regions where the optima provably cannot lie, and smallest regions in which a point is possibly optimal. The design system then focuses on those regions where the solution may lie. The power of activity analysis to eliminate large suboptimal subspaces is derived from *Qualitative KKT* (*QKKT*), an abstraction in *qualitative vector algebra* of the foundational Karush-Kuhn-Tucker (KKT) conditions of optimization theory. As KKT provides the foundation for continuous nonlinear numerical optimization algorithms, so too can QKKT provide a foundation for combinatorial algorithms for nonlinear symbolic problems (in the spirit in which traditional combinatorial approaches such as the Simplex Method tackle linear problems.) Power is derived by merging these novel combinatorial algorithms with numerical methods. Activity analysis is one such algorithm that partitions a design problem into smaller problems that can subsequently be pursued by KKT-based numerical algorithms or other symbolic methods.

Activity analysis is striking in the way it merges together three styles of search that are traditionally viewed as quite disparate: First is the rich suite of more tactical, numeric methods (e.g. Vanderplaats[16], Papalambros and Wilde[14]) used in continuous optimizing search to climb locally but monotonically towards the optimum. The second is the more strategic, conflict-based approaches used in combinatorial, satisficing search to eliminate finite, inconsistent subspaces (e.g. de Kleer and Williams[7]). Activity analysis draws from the power of both perspectives, strategically cutting away subspaces that it can quickly rule out as suboptimal.

The third style of search is the symbolic optimization approach of monotonicity analysis, derived from the Monotonicity Principles (Wilde[17], Papalambros and Wilde[13], Papalambros[12]) which examines the well-boundedness of an optimization problem, thereby indirectly partitioning the design space into regions which satisfy the first-order necessary conditions of optimality. Papalambros and Wilde provided a mathematical method to systematize the engineering insight of optimal problem solving. Choy and Agogino[5], Michelena and Agogino[11], Rao and Papalambros[15], Azarm and Papalambros[2] and Hansen, *et al.*[8] automated this insight with rule-based and graph-based programs.

Inspired by these three approaches, activity analysis provides a formal theory, and a simple algorithm, for optimally partitioning a design space, that is sound and complete. Activity analysis uses a qualitative version of the Karush-Kuhn-Tucker conditions of optimality that embodies the principles of monotonicity analysis. All points in a region identified by activity analysis as non optimal violate KKT; any points in a region identified by the technique as qualitatively optimal satisfy QKKT. Activity analysis generates these regions from an explicit statement of QKKT through a *prime assignment algorithm* analogous to the prime implicant algorithms used in satisficing search. Finally, activity analysis may use any available numerical methods to explore the remaining subspaces, thus complementing traditional methods.

2. MOTIVATING EXAMPLE: HYDRAULIC CYLINDER

To demonstrate the task consider the design of a hydraulic cylinder (Fig. 1) as introduced by Wilde[17] to demonstrate the related technique of monotonicity analysis. Given as variables the inside diameter (*i*), the wall thickness (*t*), the hoop stress (*s*), the force (*f*), and the pressure (*p*), the problem is to parametrically minimize the diameter of the cylinder, modeled as inside diameter plus twice the wall thickness, subject to an upper bound on the hoop stress (*S*), a lower bound on

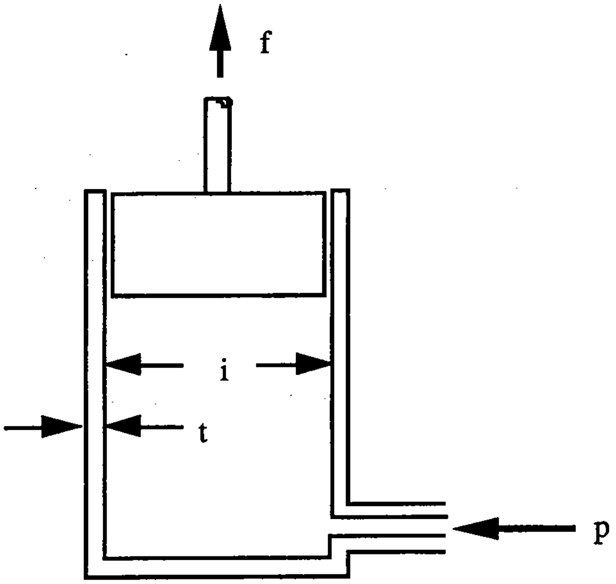

FIGURE 1 Hydraulic cylinder.

the force (F), an upper bound on the input pressure (P), and a lower bound on the wall thickness (T), and physical relations modeling stress and pressure; positivity conditions are assumed (*i.e.* $i, s, t, p, f > 0$). The optimization problem is modeled as:

Minimize: $i + 2t$

subject to:

$$s - \frac{pi}{2t} = 0 \quad (h_1 = 0)$$

$$f - \frac{\pi i^2}{4} p = 0 \quad (h_2 = 0)$$

$$F - f \leqslant 0 \quad (g_1 \leqslant 0)$$

$$T - t \leqslant 0 \quad (g_2 \leqslant 0)$$

$$p - P \leqslant 0 \quad (g_3 \leqslant 0)$$

$$s - S \leqslant 0 \quad (g_4 \leqslant 0)$$

in which design variables are in lowercase, fixed parameters in uppercase, and equality and inequality constraints are labeled h_i and g_i, respectively.

The optimization problem has three degrees-of-freedom (DOF)-five variables and two equality constraints-resulting in a bounded three-dimensional space. Figure 2 illustrates the characteristics of this space in terms of s, t, and p.

Activity analysis identifies that the stationary points, and thus the optima, must lie in two subspaces. The new problem formulation finds the optima of the two spaces and combines the results as follows (where "arg min" returns a set of optima):

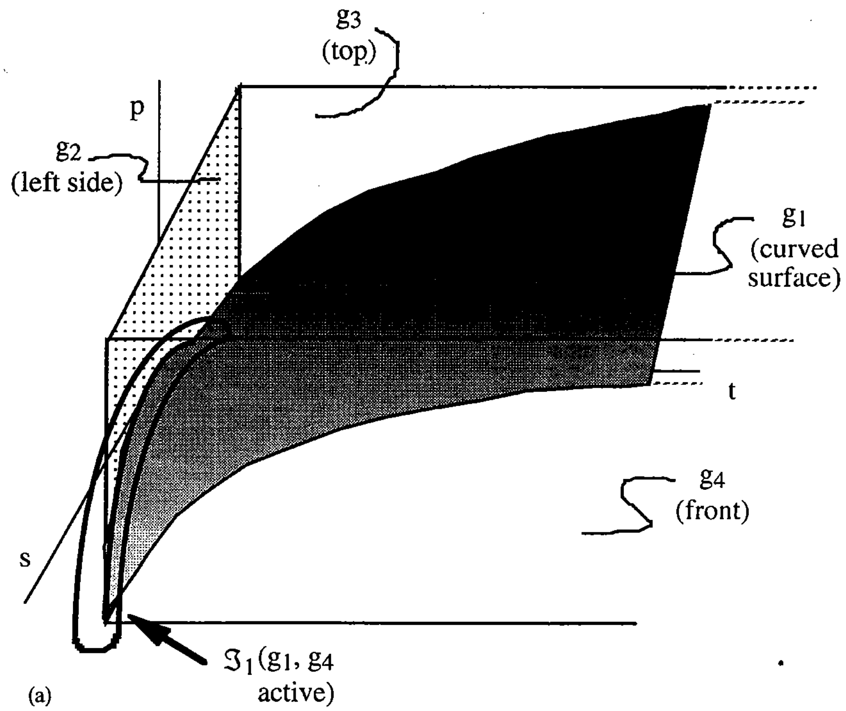

FIGURE 2 Characteristic hydraulic cylinder space and reduced spaces $\Im_1$ (2a) and $\Im_2$ (2b).

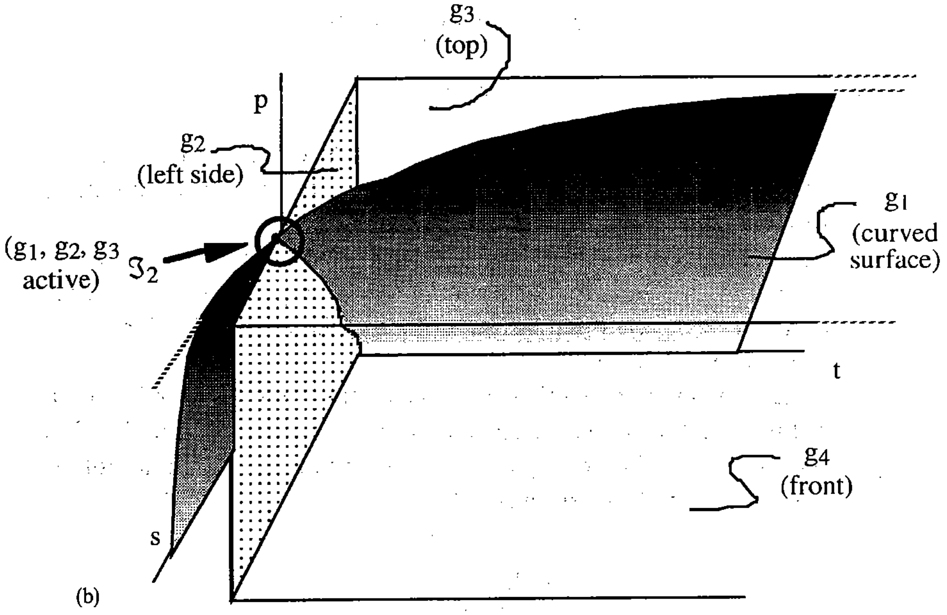

FIGURE 2 *Continued*

Given: vector $\mathbf{x} = (istpf)^T$,

1. Let $\mathbf{Y} = \arg\min_{\mathbf{x}}(i + 2t)$
   Subject to:

$$(h_1 = 0)\ (h_2 = 0)\ (g_1 = 0)\ (g_2 \leqslant 0)\ (g_3 \leqslant 0)\ (g_4 = 0).$$

2. Let $\mathbf{Z} = \arg\min_{\mathbf{x}}(i + 2t)$
   subject to:

$$(h_1 = 0)\ (h_2 = 0)\ (g_1 = 0)\ (g_2 = 0)\ (g_3 = 0)\ (g_4 \leqslant 0).$$

3. Return $\arg\min_{\mathbf{x}}(i + 2t)$, subject to:

$$\mathbf{x} \in \mathbf{Y} \cup \mathbf{Z}.$$

Figure 2 indicates the reduced subspaces, one showing $\Im_1$ in which g_1 and g_4 become strict equalities resulting in optimum $\mathbf{Y}$ (2*a*), and the other one showing $\Im_2$ in which g_1, g_2, g_3, become strict equalities resulting in optimum **Z** (2*b*). Search within these subspaces reduces from 3 DOF to search within a line (1 DOF) and search at a point (0 DOF). Visually it is striking to note the reduction in the size of the space in which search must take place.

The remainder of this paper reviews the definition of a stationary point and the Karush-Kuhn-Tucker (KKT) conditions of optimality prior to deriving a qualitative form of KKT which is called the Qualitative Karush-Kuhn-Tucker conditions (QKKT). From QKKT, activity analysis is derived and applied to both the hydraulic cylinder and a torsion beam problem. The characteristics of activity analysis are compared to that of monotonicity analysis before concluding.

3. STATIONARY POINTS AND KARUSH-KUHN-TUCKER

For a point $\mathbf{x}^*$ to be an optimum it is necessary that the point be *stationary,* that is any "down hill" direction is blocked by the constraints. Activity analysis exploits this fact to eliminate sets of points that can quickly be proven to be *nonstationary,* using a condition we call *Qualitative Karush-Kuhn-Tucker* (QKKT). This section reviews the optimization problem, the concept of a stationary point, and the traditional algebraic (Karush-Kuhn-Tucker) condition for testing stationary points. Activity analysis applies to the pervasive family of linear and non-linear, constrained optimization problems $\mathrm{OP} = (\mathbf{x}, f, \mathbf{g}, \mathbf{h})$:

$$\begin{aligned} &\text{Find } \mathbf{x}^* = \arg\min \quad f(\mathbf{x}) \\ &\text{subject to:} \qquad \mathbf{g}(\mathbf{x}) \leqslant \mathbf{0} \\ &\qquad\qquad\qquad\quad \mathbf{h}(\mathbf{x}) = \mathbf{0} \end{aligned}$$

where column vectors are denoted in bold (e.g. $\mathbf{x}, \mathbf{x}^*, \mathbf{g}(\mathbf{x})$ and $\mathbf{h}(\mathbf{x})$), $f(\mathbf{x})$ is the objective function, $\mathbf{g}(\mathbf{x})$ is a vector of inequality constraints and $\mathbf{h}(\mathbf{x})$ is a vector of equality constraints. The problem formulation assumes negative null form. A point $\mathbf{x} \in \mathscr{R}^n$ is *feasible* if it satisfies the constraints, and *feasible space* $\Im \subseteq \mathscr{R}^n$ denotes all feasible points (represented $\Im = (\mathbf{g}, \mathbf{h})$). A *feasible direction* $\vec{s}$ from a feasible point is one through which a non-zero distance can be moved before hitting a constraint boundary. $f(\mathbf{x})$ is *decreasing* at $\mathbf{x}$ in direction $\vec{s}$ if $\nabla f(\mathbf{x}) \cdot \vec{s} < 0$.

Finally, a point is *stationary* (denoted $\mathbf{x}^*$) if any direction that decreases the objective is infeasible. Not all stationary points are local minima; a stationary point satisfies only the first-order necessary conditions of optimality (not sufficiency), and thus local maxima and saddle points are also stationary. The Karush-Kuhn-Tucker (KKT) conditions (Karush[9], Kuhn and Tucker[10]) provide a set of vector equations that are satisfied for a feasible point $\mathbf{x}^*$ exactly when that point is stationary:

$$\nabla f(\mathbf{x}^*) + \lambda^T \nabla \mathbf{h}(\mathbf{x}^*) + \mu^T \nabla \mathbf{g}(\mathbf{x}^*) = \mathbf{0}^T \quad \text{(KKT1)}$$

subject to:

$$\mu^T \mathbf{g}(\mathbf{x}^*) = \mathbf{0}^T, \quad \text{(KKT2)}$$

$$\mu \geqslant \mathbf{0}. \quad \text{(KKT3)}$$

μ^T transposes column vector μ to a row. Gradients ∇f, $\nabla \mathbf{g}$ and $\nabla \mathbf{h}$ denote Jacobian matrices. ∇f is a row vector $(\partial f/\partial x_1 \cdots \partial f/\partial x_n)$. $\nabla \mathbf{g}$ and $\nabla \mathbf{h}$ are matrices $(\partial g_i/\partial x_j)$ and $(\partial h_i/\partial x_j)$, respectively, where (a_{ij}) denotes a matrix whose element in the ith row and jth column is a_{ij}, for all i and j. For example KKT1 and KKT2 are equivalencies between row vectors, and KKT3 is a relation between column vectors. Note that a stationary point that is on a boundary is sometimes referred to as a "constrained stationary point" or a "KKT point"; the use here of the term "stationary" is for any point that satisfies the KKT conditions and thus encompasses the use of these terms.

Rewriting KKT1 as:

$$-\nabla f(\mathbf{x}^*) = \lambda^T \nabla \mathbf{h}(\mathbf{x}^*) + \mu^T \nabla \mathbf{g}(\mathbf{x}^*),$$

the $-\nabla f$ term denotes directions of decreasing objective from $\mathbf{x}^*$, the term $(\lambda^T \nabla \mathbf{h}(\mathbf{x}^*) + \mu^T \nabla \mathbf{g}(\mathbf{x}^*))$ denotes infeasible directions from $\mathbf{x}^*$, and the equality says the decreasing objective directions are all infeasible; hence, $\mathbf{x}^*$ is stationary. More specifically, $\vec{s}$ decreases the objective if it has a component in the $-\nabla f$ direction $(\vec{s} \cdot \nabla f < 0)$. A direction is infeasible with respect to inequality constraint $g_i(\mathbf{x}^*)$ if $\mathbf{x}^*$ lies on the constraint boundary $(g_i(\mathbf{x}^*) = 0)$ and it has a component in the $+\nabla g_i(\mathbf{x}^*)$ direction. A direction is infeasible with respect to equality constraint $h_j(\mathbf{x}^*)$ if it has a component in either the $-\nabla h_j(\mathbf{x}^*)$ or $+\nabla h_j(\mathbf{x}^*)$ direction. Most importantly, if $\mathbf{x}^*$ lies on multiple constraint boundaries, then an infeasible direction has a component which is a linear, weighted combination of the above gradients for these constraints. The weights are μ and λ (called Lagrange multipliers), and the combination is $\mu^T\nabla\mathbf{g}+\lambda^T\nabla\mathbf{h}$ subject to KKT2 and KKT3. Hence all decreasing directions are infeasible when $-\nabla f$ equals one of these linear combinations (KKT1). Figure 3 shows an example of $\nabla\mathbf{g}$ gradient vectors ($\nabla\mathbf{g}_1$ and $\nabla\mathbf{g}_2$), and the combined weighted vector, $\vec{\mathbf{w}}$, which exactly cancels ∇f.

A key property of KKT is that it identifies *active* inequality constraints. Intuitively, a constraint (g_i) is active at a point $\mathbf{x}$ when $\mathbf{x}$ is *on* the constraint boundary and the direction of decreasing objective, ∇f, is pointing into the boundary. When this is true μ_i is positive. The basis of the present approach is to conclude, by looking at signs of μ, that the stationary points lie at the intersection of the constraint boundaries; one or more constraints have been identified as active, hence the name *activity analysis.*

4. QUALITATIVE KKT CONDITIONS

Qualitative KKT (*QKKT*) is an abstraction of KKT that is a necessary, but insufficient, condition for a point being stationary. It is the means by which activity analysis quickly rules out suboptimal subspaces. Qualitative properties used by QKKT to test a point $\mathbf{x}$ include whether each constraint is active at $\mathbf{x}$, and the quadrant of the coordinate axes each gradient ∇f, $\nabla\mathbf{g}$ and $\nabla\mathbf{h}$ lies within. In two dimensions the space is divided into four regions or quadrants; in n-dimensions the term quadrant is used to indicate the analogous n-dimensional regions. These properties can be extracted quickly and hold uniformly for large subsets of the feasible space, and parameterized families of optimization problems. QKKT and manipulations by activity analysis rely on a matrix version of SR1-a hybrid algebra combining signs and reals. This algebra behaves as one expects given a familiarity with (scalar) sign algebra and traditional matrix algebra (see Williams[19]). Essentially, the sign of a quantity is determined when possible. The

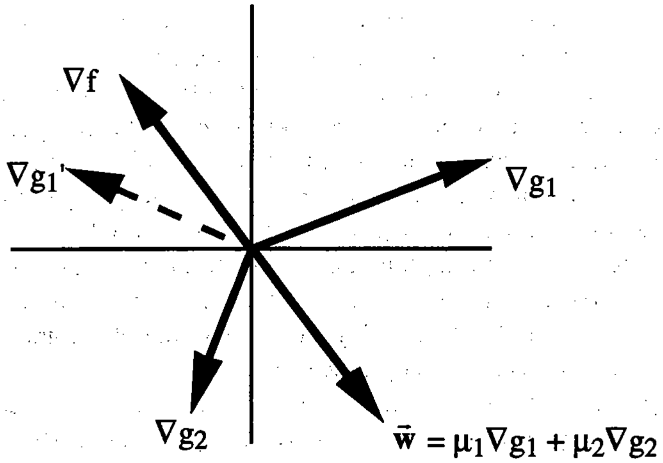

FIGURE 3 Example gradient vector diagram for KKT.

result is one of four values, namely $\hat{+}$ if the quantity is positive, $\hat{-}$ if the quantity is negative, 0 if the quantity is zero, and $\hat{?}$ if the quantity is unknown. If the sign of a quantity is desired, it is specified by placing that quantity within square brackets (*e.g.* $[x]$ for $x > 0$ is $\hat{+}$, while $[x - y]$ is $\hat{?}$ unless additional information is known).

Derived from KKT, QKKT states that a feasible point $\mathbf{x}^*$ is stationary only if:

$$[\nabla f(\mathbf{x}^*)] + [\lambda]^T[\nabla \mathbf{h}(\mathbf{x}^*)] + [\mu]^T[\nabla \mathbf{g}(\mathbf{x}^*)] \mathrel{\hat{=}} 0^T, \quad \text{(QKKT1)}$$

subject to

$$[\mu]^T[\mathbf{g}(\mathbf{x}^*)] = 0^T, \text{ and} \quad \text{(QKKT2)}$$

$$[\mu_i] \neq \hat{-}, \quad \text{(QKKT3)}$$

where $[\mathbf{v}]$, called a *sign vector*, denotes the signs of the elements of $\mathbf{v}$, such that $[v_i] \in \{\hat{-}, 0, \hat{+}\}$. Recall the KKT said that to be stationary there must exist a weighted sum ($\vec{\mathbf{w}}$) of $\nabla \mathbf{g}$ and $\nabla \mathbf{h}$ that exactly cancels ∇f (note $\vec{\mathbf{w}}$ is a row vector). QKKT says a point is *nonstationary* unless there exists a $\vec{\mathbf{w}}$ that lies in the quadrant diagonal from that which contains ∇f. For example, in Figure 3, ∇f lies in the upper left quadrant; thus a $\vec{\mathbf{w}}$ must exist that lies in the lower right. The sign vector $[\mathbf{v}]$ denotes the quadrant containing a vector $\mathbf{v}$, and each component $[v_i]$ describes where $\mathbf{v}$ lies relative to the $v_i = 0$ plane. For example $[\vec{\mathbf{w}}] = (\hat{+}\ \hat{-})$ indicates that $\vec{\mathbf{w}}$ is in the lower right quadrant. Using this algebraic representation, the condition on ∇f and $\vec{\mathbf{w}}$ lying in diagonal quadrants is expressed by $-[\nabla f] = [\vec{\mathbf{w}}]$.

It can be noted from KKT that the quadrant $\vec{\mathbf{w}}$ lies within is $[\vec{\mathbf{w}}] = [\mu^T\ \nabla \mathbf{g} + \lambda^T \nabla \mathbf{h}]$. Using only knowledge of the quadrant each constraint's gradient lies within and whether each constraint is active (indicated by the signs of the Lagrange multipliers $[\mu]$ and $[\lambda]$), it follows from the properties of sign algebra that the quadrants $\vec{\mathbf{w}}$ may lie within are a subspace of those described by $[\mu]^T[\nabla \mathbf{g}] + [\lambda]^T[\nabla \mathbf{h}]$. Thus, $-[\nabla f] = [\vec{\mathbf{w}}] \subseteq [\mu]^T[\nabla \mathbf{g}] + [\lambda]^T[\nabla \mathbf{h}]$ (*i.e.* QKKT1). For example, in Figure 3 since $\nabla g_1 (= (\hat{+}\ \hat{+}))$ lies in the upper right and $\nabla g_2\ (= (\hat{-}\ \hat{-}))$ lies in the lower left, it is possible for a $\vec{\mathbf{w}}$ to lie in the lower right; thus, any $\mathbf{x}$ satisfying these conditions may be stationary. But suppose ∇g_1 is replaced with $\nabla g'_1$, which lies in the upper left for points in some feasible subspace $\mathfrak{I}_1$. Then $\vec{\mathbf{w}}$ may lie in the upper or lower left, but not the lower right; thus, all points in $\mathfrak{I}_1$ must be nonstationary. That is, evaluating $-[\nabla f]$ and $[\mu]^T[\nabla \mathbf{g}]$ using ∇g_1 and ∇g_2 for $\nabla \mathbf{g}$ satisfies the subset relation of QKKT1:

$$(\hat{+}\quad \hat{-}) \subseteq (\hat{?}\quad \hat{?}) = (\hat{+}\quad \hat{+}) \begin{pmatrix} \hat{+} & \hat{+} \\ \hat{-} & \hat{-} \end{pmatrix}$$

However, evaluating these expressions using $\nabla g'_1$ and ∇g_2 for $\nabla \mathbf{g}$ doesn't satisfy the subset relations:

$$(\hat{+}\quad \hat{-}) \not\subseteq (\hat{-}\quad \hat{?}) = (\hat{+}\quad \hat{+}) \begin{pmatrix} \hat{-} & \hat{+} \\ \hat{-} & \hat{-} \end{pmatrix}$$

Thus points characterized by ∇g_1 and ∇g_2 may be stationary while those characterized by $\nabla g'_1$ and ∇g_2 are nonstationary. It is this second type of conclusion, made from only qualitative properties, that activity analysis uses to eliminate feasible subspaces of nonstationary points.

KKT has provided the foundation for a large body of work on gradient-based numerical analysis. Likewise, QKKT provides the foundation for a set of symbolic, combinatorial techniques. One such technique (activity analysis) is presented which manipulates QKKT directly. As will be discussed in Section 8, monotonicity analysis is a different technique whose principles follow as a consequence of QKKT.

5. INSTANTIATING QKKT

Instantiating QKKT1 on optimization problem OP $\equiv (\mathbf{x}, f, \mathbf{g}, \mathbf{h})$ involves three steps:

1. Compute Jacobians ∇f, $\nabla \mathbf{g}$ and $\nabla \mathbf{h}$ by symbolic differentiation.
2. Compute the signs of the Jacobians. For each element,

   (a) replace real operators with sign operators, using properties:

   - $[a+b] \subseteq [a]+[b]$,
   - $[ab]=[a][b]$,
   - $[a/b]=[a]/[b]$ and
   - $[-a]=-[a]$.

   (b) Substitute for sign variables [a] using positivity conditions ($[a]=\hat{+}$).

   (c) Perform sign arithmetic (e.g. $[5] \Rightarrow \hat{+}, [\hat{-}]+[\hat{-}] \Rightarrow \hat{-}$).
3. Expand QKKT1 by expanding matrix sums and products.

Returning to the hydraulic cylinder problem, recall that $\mathbf{x}$ is the vector $(i\,t\,f\,s\,p)^T$, the objective $f(\mathbf{x})$ is $i+2t$, and the constraint vectors are:

$$h=\begin{pmatrix} s-\frac{pi}{2t} & f-\frac{\pi i^2}{4}p \end{pmatrix}^T$$

$$g=(F-f \quad T-t \quad p-P \quad s-S)^T$$

The following shows $\nabla \mathbf{h}$ after steps 2a (top) and 2b (bottom):

$$[\nabla \mathbf{h}]=\begin{pmatrix} \frac{-[p]}{[2][t]} & \frac{[p][i]}{[2][t]^2} & 0 & [1] & \frac{-[i]}{[2][t]} \\ \frac{-[\pi][i]}{[2]}[p] & 0 & [1] & 0 & \frac{-[\pi][i]^2}{[4]} \end{pmatrix}$$

$$=\begin{pmatrix} \hat{-} & \hat{+} & 0 & \hat{+} & \hat{-} \\ \hat{-} & 0 & \hat{+} & 0 & \hat{-} \end{pmatrix}$$

Repeating for $[\nabla f]$ and $[\nabla \mathbf{g}]$ and inserting into QKKT1:

$$\mathbf{0}^T \subseteq \begin{pmatrix} \hat{+} \\ \hat{+} \\ 0 \\ 0 \\ 0 \end{pmatrix}^T + [\lambda]^T \begin{pmatrix} \hat{-} & \hat{+} & 0 & \hat{+} & \hat{-} \\ \hat{-} & 0 & \hat{+} & 0 & \hat{-} \end{pmatrix} + [\mu]^T \begin{pmatrix} 0 & 0 & \hat{-} & 0 & 0 \\ 0 & \hat{-} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \hat{+} \\ 0 & 0 & 0 & \hat{+} & 0 \end{pmatrix}$$

Expanding matrix operations for step 3 results in equations QKKT1 (1)–(5):

$$0 \subseteq (\hat{+}) - [\lambda_1] - [\lambda_2] \tag{1}$$

$$0 \subseteq (\hat{+}) - [\mu_2] + [\lambda_1] \tag{2}$$

$$0 \subseteq -[\mu_1] + [\lambda_2] \tag{3}$$

$$0 \subseteq [\mu_4] + [\lambda_1] \tag{4}$$

$$0 \subseteq [\mu_3] - [\lambda_1] - [\lambda_2] \tag{5}$$

For ease of reading we write terms $\hat{+}\ [x_i]$ as $[x_i]$, $\hat{-}\,[x_i]$, as $-[x_i]$, and eliminated terms $0[x_i]$, where x_i represents λ_i or μ_i. Note that the computation of sign matrices in step 2 is extremely simple, but surprisingly adequate for many problems. The symbolic algebra system Minima (Williams[19]) can provide a reasonably general tool for deducing the signs of sensitivities (*e.g.* $[\partial f(\mathbf{x})/\partial \mathbf{x}_i]$) subject to $\mathbf{x}$ satisfying the equality and inequality constraints. Thus far a condition has been developed that is easily evaluable yet sufficient for testing the suboptimality of infinite subspaces. Next, a technique called activity analysis is examined that uses QKKT to strategically focus the search for optima.

6. ACTIVITY ANALYSIS AND PRIME ASSIGNMENTS

Activity analysis reduces an optimization problem to a set of simpler subproblems by cutting out feasible subspaces that are suboptimal. These subspaces contain all and only those points that can be proved nonstationary according to QKKT. Recall that a point is stationary if and only if it satisfies KKT. In analogy we say a point is *qstationary* if and only if it satisfies QKKT. The output of activity analysis is a concise description of the remainder, a minimal covering composed of maximum qstationary subspaces, called a *minimal qstationary covering*. It is a set of feasible subspaces (and corresponding optimization problems), at least one of which is guaranteed to contain the true optimum. Thus three key features of the descriptions generated by activity analysis are parsimony, correctness and maximization of the "filtering" achievable using QKKT. This section states and demonstrates the activity analysis problem and an algorithmic instantiation. The core is a mapping between *minimal qstationary subspaces and prime assignments,* and a general prime assignment engine for systems of linear sign equations.

To start, a point is said to be *qnonstationary* if it follows from QKKT that it is nonstationary; otherwise, it is *qstationary.* A feasible subspace is *qstationary* if all its points are qstationary, and *qnonstationary* if all its points are qnonstationary. Activity analysis maximizes its use of QKKT while preserving correctness by eliminating exactly the qnonstationary subspaces from its description of the feasible space. This description is built from a set Σ whose elements result from strengthening one or more of the inequality constraints $g_i \leqslant 0$ to strict equalities $g_i = 0$; that is, Σ is the powerset of constraint boundary intersections. The description (called a *minimal qstationary covering*), covers the qstationary points by collecting all qstationary subspaces that are maximal under superset. These cover every qstationary subspace. The activity analysis problem is then: *given optimization problem* $\mathrm{OP} = (\mathbf{x}, f, \mathbf{g}, \mathbf{h})$ *and instantiation of* QKKT ($=$ QKKT(OP)), *construct the minimal qstationary covering C.*

Mapping QKKT(OP) to C relies on two observations: First, from QKKT2 ($\equiv [\mu_i(\mathbf{x})][g_i(\mathbf{x})] = 0$) it follows that $[\mu_i(\mathbf{x})] = \hat{+} \rightarrow g_i(\mathbf{x}) = 0$. That is, any point where $[\mu_i] = \hat{+}$ must be on the $g_i = 0$ constraint boundary. Thus, when activity analysis shows that a subspace of qstationary points makes $[\mu_i] = \hat{+}$ for one or more g_i's, it concludes that these points lie along the intersection of the g_i boundaries. Second, a particular set of variable assignments for QKKT1, called *prime (implicating) assignments*, directly maps to the minimal qstationary covering by applying the first observation. The key here is that achieving parsimony, maximum filtering and correctness reduces to generating complete prime assignments.

Define a *(partial) assignment* to $[\mathbf{x}]$ as a set A which assigns each $[x_i]$ at most one value, $A \subseteq \{[x_i] = s | x_i \in \mathbf{x}, s \in \{\hat{-}, 0, \hat{+}\}\}$. We are interested in the *consistent assignments to* QKKT1, where the $[\mathbf{x}]$ to be assigned is a vector of Lagrange multipliers $([\mu]^T[\lambda]^T)^T$. Additionally, the consistent assignments must also satisfy the restriction of QKKT3 $([\mu] \neq \hat{-})$. Note that each consistent assignment C has a corresponding subset S of feasible space, produced by making active those constraints with $[\mu_i] = \hat{+}$ and adding them to the original constraint set. S has the property that every point in S satisfies assignment C. Since each C satisfies QKKT, any point in its corresponding S may be a stationary point. Note that a partial assignment assigns any of $\{\hat{-}, 0, \hat{+}\}$ to the Lagrange multipliers, while a *consistent assignment* assigns only those of $\{\hat{-}, 0, \hat{+}\}$ that satisfy the restriction that $[\mu_i] \neq \hat{-}$.

Next, an *implicating assignment* γ is a consistent assignment to QKKT1, such that whenever an extension to γ satisfies restriction QKKT3, it also is consistent with QKKT1. That is, given restriction QKKT3, assignment γ *implies* QKKT1. An implicating assignment has the important property that every point in its corresponding subspace S satisfies QKKT. Thus S is a qstationary subspace. Essentially this means that an implicating assignment is one such that QKKT1 is satisfied; it is partial in that additional consistent assignments can also be made.

Finally, a *prime assignment* P is an implicating assignment no proper subset of which is also an implicating assignment. Thus P's corresponding S is a maximal qstationary subspace. Conversely, every maximal qstationary subspace is the corresponding subspace of some prime assignment. Thus the set of subspaces corresponding to all prime assignments is a minimal qstationary covering. The prime assignments are thus the smallest assignments that still consistently imply QKKT1.

To produce all primes for QKKT1, the prime assignment engine first computes the primes P_i of each scalar equation in QKKT1, then combines them using minimal set covering. Pulling this all together, the activity analysis algorithm is:

Given problem $\mathrm{OP} = (\mathbf{x}, f, \mathbf{g}, \mathbf{h})$:

1. Instantiate QKKT1 producing QKKT1(OP).
2. Compute prime assignments, P_i, of each $\mathrm{QKKT1}_i(\mathrm{OP}) \in \mathrm{QKKT1}$ (OP).
3. Compute minimal set covering of P_i producing P_{total}. Delete inconsistent assignments.
4. Extract minimal sets of $[\mu_i] = \hat{+}$ assignments from P_{total} producing U'.
5. Remove supersets from U' producing. U.
6. Map each element of U to a maximal qstationary subspace by applying $[\mu_i(\mathbf{x})] = \hat{+} \rightarrow g_i(\mathbf{x}) = 0$, producing a covering, C.
7. Presume that $n+1$ equations overdetermine the solution, removing those subspaces.
8. Formulate and return a new optimization problem from C.

Various algorithms to instantiate activity analysis are feasible. One described in Williams and Cagan[20] is presented here. Step one was demonstrated in the previous section. For steps two and three it is noted that QKKT1 takes the form $\mathbf{0} \subseteq [\mathbf{B}] + [\mathbf{A}][\mathbf{x}]$, with $[\mathbf{A}]$ and $[\mathbf{B}]$ being sign constant matrices, $[\mathbf{x}]$ an n vector, $[\mathbf{A}]$ an n by m matrix and $[\mathbf{B}]$ an m vector. In particular, $\mathbf{x}^T$ is $(\mu^T \lambda^T)^T$, $[\mathbf{B}] = [\nabla f]$, and $[\mathbf{A}]$ is the matrix $(\nabla \mathbf{g}\, \nabla \mathbf{h})$. For the hydraulic cylinder (Section 5), QKKT1 has 5 equations, with $\mathbf{x} \equiv (\mu_1 \mu_2 \mu_3 \mu_4 \lambda_1 \lambda_2)^T$. It is known that $[\mu_i] \neq \hat{-}$ from QKKT3.

For step 2, the prime assignments of each QKKT1 equation are constructed from three sets of scalar assignments, consistent with non-negativity of μ: those restricting one of the equation's terms ($[a_{ij}][x_j]$) to be positive (P_i), those making a term zero (Z_i), and those making a term negative (N_i), respectively. For each QKKT1 equation, there are two possibilities:

1. $b_i = \hat{+}$, where $[b_i] = [\partial f / \partial x_i]$ (the analogous argument holds for $b_i = \hat{-}$). For QKKT1 to hold for that equation, $\Sigma_{j=1}^{m} [a_{ij}][x_j] = \hat{?}$, since $0 \subseteq (\hat{+}) + (\hat{?})$. This holds exactly when at least one of the

$[a_{ij}][x_j]$ terms is negative (since $0 \subseteq (\hat{+}) + (\hat{-}) = \hat{?}$). For example, in the cylinder QKKT Eq.(2), $\lambda_1 = \hat{-}$ guarantees that the equation is satisfied. The only other assignment that guarantees this is $\mu_2 = \hat{+}$. Thus the prime assignments for Eq. (2) are $\{\lambda_1 = \hat{-}\}$ and $\{\mu_2 = \hat{+}\}$.

2. $b_i = 0$. Here $\Sigma_{j=1}^{m}[a_{ij}][x_j] = 0$ or $\hat{?}$ for QKKT1 to hold. For the first case, all terms must be 0. For the second case, at least one term must be positive and one negative. For example, $[\partial f(\mathbf{x})/\partial \mathbf{x}_i] = 0$ in cylinder QKKT1(3): $0 \subseteq -[\mu_1] + [\lambda_2]$. Thus, the prime assignments are $\{\lambda_2 = 0, \mu_1 = 0\}$ and $\{\lambda_2 = \hat{+}, \mu_1 = \hat{+}\}$. Note that $\{\lambda_2 = \hat{-}, \mu_1 = \hat{-}\}$ is not acceptable, since by restriction $[\mu_i] \neq \hat{-}$.

Recall for the cylinder that instantiating QKKT1 produced the following equations:

$$0 \subseteq (\hat{+}) - [\lambda_1] - [\lambda_2] \tag{1}$$

$$0 \subseteq (\hat{+}) - [\mu_2] + [\lambda_1] \tag{2}$$

$$0 \subseteq -[\mu_1] + [\lambda_2] \tag{3}$$

$$0 \subseteq \quad [\mu_4] + [\lambda_1] \tag{4}$$

$$0 \subseteq [\mu_3] - [\lambda_1] - [\lambda_2] \tag{5}$$

Constructing the prime assignments for these equation uses:

	$\mathbf{N}_i$	$\mathbf{Z}_i$	$\mathbf{P}_i$
1	$[\lambda_1] = \hat{+}, [\lambda_2] = \hat{+}$	$[\lambda_1] = 0, [\lambda_2] = 0$	$[\lambda_1] \hat{-}, [\lambda_2] = \hat{-}$
2	$[\lambda_1] = \hat{-}, [\mu_2] = \hat{+}$	$[\lambda_1] = 0, [\mu_2] = 0$	$[\lambda_1] = \hat{+}$
3	$[\lambda_2] = \hat{-}, [\mu_1] = \hat{+},$	$[\lambda_2] = 0, [\mu_1] = 0$	$[\lambda_2] = \hat{+}$
4	$[\lambda_1] = \hat{-},$	$[\lambda_1] = 0, [\mu_4] = 0$	$[\lambda_1] = \hat{+}, [\mu_4] = \hat{+}$
5	$[\lambda_1] = \hat{+}, [\lambda_2] = \hat{+},$	$[\lambda_1] = 0, [\lambda_2] = 0,$ $[\mu_3] = 0$	$[\lambda_1] = \hat{-}, [\lambda_2] = \hat{-},$ $[\mu_3] = \hat{+}$

The prime (implicating) assignments for the table of cylinder equations QKKT1(1)–(5) are:

$$\{[\lambda_1] = \hat{+}\}, \{[\lambda_2] = \hat{+}\} \qquad P(1)$$

$$\{[\lambda_1] = \hat{-}\}, \{[\mu_2] = \hat{+}\} \qquad P(2)$$

$$\{[\lambda_2] = 0, [\mu_1] = 0\}, \{[\lambda_2] = \hat{+}, [\mu_1] = \hat{+}\} \qquad P(3)$$

$$\{[\lambda_1] = 0, [\mu_4] = 0\}, \{[\lambda_1] = \hat{-}, [\mu_4] = \hat{+}\} \qquad P(4)$$

$$\begin{gathered}\{[\lambda_1] = 0, [\lambda_2] = 0, [\mu_3] = 0\}, \\ \{[\lambda_1] = \hat{+}, [\lambda_2] = \hat{-}\}, \{[\lambda_1] = \hat{+}, [\mu_3] = \hat{+}\}, \\ \{[\lambda_1] = \hat{-}, [\lambda_2] = \hat{+}\}, \{[\lambda_2] = \hat{+}, [\mu_3] = \hat{+}\} \qquad P(5)\end{gathered}$$

The third step, constructing the composite primes for QKKT1, is based on a standard algorithm for minimal set covering (*e.g.* Corman, *et al.* [16]). The result is a consistent set of prime assignments across all QKKT1(i). More specifically, minimal set covering is defined as follows: Given a set of sets S, a covering of S is a set c which contains at least one element of each $s \in S$. A minimal set covering is a covering no subset of which is also a covering. A minimal set covering algorithm returns all minimal coverings given S. Roughly speaking, a minimal set covering algorithm constructs all smallest sets of assignments that select at least one prime assignment for each QKKT1(i). Sets are thrown away that are inconsistent (*i.e.* assign conflicing signs) or are supersets of other sets. Minimal set covering algorithms are worst-case exponential and the minimal set covering problem is NP-complete. For the cylinder, the minimal covering of $P(1)$–(5) produces just two prime assignments:

$$\begin{gathered}\{\{[\lambda_1] = \hat{-}, [\lambda_2] = \hat{+}, [\mu_1] = \hat{+}, [\mu_4] = \hat{+}\}, \\ \{[\lambda_1] = 0, [\lambda_2] = \hat{+}, [\mu_1] = \hat{+}, [\mu_2] = \hat{+}, [\mu_3] = \hat{+}, [\mu_4] = 0\}\}.\end{gathered}$$

The fourth step, extracting the minimal sets of $[\mu_i] = \hat{+}$ assignments, results in $\{[\mu_1] = \hat{+}, [\mu_4] = \hat{+}\}$ and $\{[\mu_1] = \hat{+}, [\mu_2] = \hat{+}, [\mu_3]$

$= \hat{+}\}$. The fifth step removes supersets from these sets. There are none here; however, this step is necessary since the minimal set covering produced in step 3 includes signs on the equality Lagrange multipliers (λ^T). In activity analysis all equality constraints are accounted for; thus different sub-region coverings differentiate only between active inequalities.

The sixth step uses the rule

$$[\mu_i] = \hat{+} \rightarrow g_i(x) = 0$$

to map these sets to the equivalent minimal qstationary covering. The sets indicate that g_1 and g_4, or g_1, g_2, and g_3 must be active. The resulting cover is:

$$\mathfrak{I}_1 \equiv (\{g_2, g_3,\}, \{h_1, h_2, g_1, g_4,\}) \text{ and}$$

$$\mathfrak{I}_2 \equiv (\{g_4\}, \{h_1, h_2, g_1, g_2, g_3\}),$$

where $\mathfrak{I} = (\mathbf{g}, \mathbf{h})$ is a space defined by inequality $\mathbf{g}$ and equality $\mathbf{h}$ constraints. $\mathfrak{I}_1$ and $\mathfrak{I}_2$ denote the line and point highlighted in the introduction to the cylinder example. The seventh step is to check for overdetermined spaces, of which there are none here. The final step, formulating a new optimization problem, produces:

Given: $S \equiv \{\mathbf{x}^* | \mathbf{x}^* = arg\ \min_{\mathbf{x} \in \mathfrak{I}} f(\mathbf{x}), \text{ and } \mathfrak{I} \in \{\mathfrak{I}_1, \mathfrak{I}_2\}\}$,

Find: $\min_{\mathbf{x} \in s} f(\mathbf{x})$.

The first part finds the minimum of each subspace in the covering. The second part selects from these the global minimum. As was stated earlier in section 2, the optimization problem for the hydraulic cylinder becomes:

1. Let $\mathbf{Y} = \arg \min_{\mathbf{x}}(i + 2t)$

   Subject to:

$$(h_1 = 0)\ (h_2 = 0)\ (g_1 = 0)\ (g_2 \leqslant 0)\ (g_3 \leqslant 0)\ (g_4 = 0).$$

2. Let $\mathbf{Z} = \arg\min_{\mathbf{x}}(i + 2t)$

   subject to:

$$(h_1 = 0)\ (h_2 = 0)\ (g_1 = 0)\ (g_2 = 0)\ (g_3 = 0)\ (g_4 \leqslant 0).$$

3. Return $\arg\min_{\mathbf{x}}(i + 2\mathrm{t})$, subject to:

$$\mathbf{x} \in \mathbf{Y} \cup \mathbf{Z}.$$

Again, the complexity of this problem is significantly reduced from that of the original problem. Each subproblem describes the region by identifying active constraints; this lends opportunity to gain insight into the optimum through further analysis.

7. EXAMPLE: TORSION ROD

Consider the design of a cylindrical rod under torsion load with the goal of minimizing weight (Figure 4). The variables are the radius (r), the angle of deformation (ϕ) and the maximum shear stress (τ), while the input parameters are the applied torque (T), length (L), maximum angle of deflection ($\phi_{\max}$), minimum radius ($r_{\min}$), minimum strength-to-weight ratio ($SW_{\min}$) and material density (ρ), yield stress (τ_y) and shear modulus of elasticity (G).

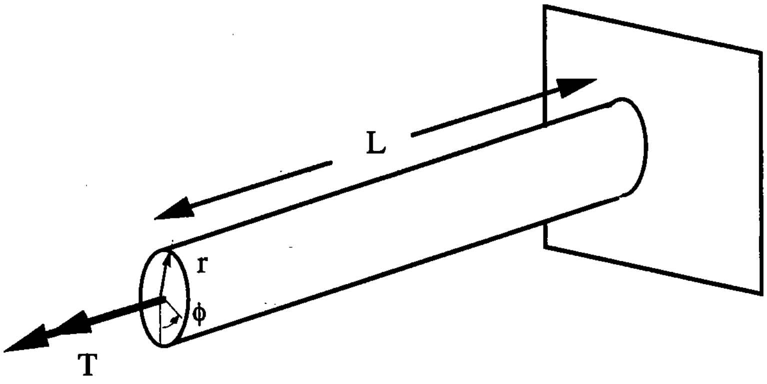

FIGURE 4 Torsion Rod.

The optimization problem is modeled as:

$$\text{min:} \qquad \rho L \pi r^2$$

$$\text{subject to:} \qquad \tau - \frac{G\phi r}{L} = 0 \qquad (h_1 = 0)$$

$$\phi - \frac{2TL}{\pi G r^4} = 0 \qquad (h_2 = 0)$$

$$\tau - \tau_y \leqslant 0 \qquad (g_i \leqslant 0)$$

$$r_{\min} - r \leqslant 0 \qquad (g_2 \leqslant 0)$$

$$\phi - \phi_{\max} \leqslant 0 \qquad (g_3 \leqslant 0)$$

$$SW_{\min} - \frac{\tau}{\rho L \pi r^2} \leqslant 0 \quad (g_4 \leqslant 0)$$

where $\mathbf{x} = (r, \phi, \tau)^T$, and positivity is assumed.

Following again the steps of activity analysis:

$$\nabla f = (2\rho L \pi r \quad 0 \;\; 0)$$

$$\nabla \mathbf{h} = \begin{pmatrix} -\dfrac{G\phi}{L} & -\dfrac{Gr}{L} & 1 \\ \dfrac{8TL}{\pi G r^5} & 1 & 0 \end{pmatrix}$$

$$\nabla \mathbf{g} = \begin{pmatrix} 0 & 0 & 1 \\ -1 & 0 & 0 \\ 0 & 1 & 0 \\ \dfrac{2\tau}{\rho L \pi r^3} & 0 & -\dfrac{1}{\rho L \pi r^2} \end{pmatrix}$$

Taking the signs of the Jacobians:

$$[\nabla f] = (\hat{+} \quad 0 \quad 0)$$

$$[\nabla \mathbf{h}] = \begin{pmatrix} \hat{-} & \hat{-} & \hat{+} \\ \hat{+} & \hat{+} & 0 \end{pmatrix}$$

$$[\nabla \mathbf{g}] = \begin{pmatrix} 0 & 0 & \hat{+} \\ \hat{-} & 0 & 0 \\ 0 & \hat{+} & 0 \\ \hat{+} & 0 & \hat{-} \end{pmatrix}$$

QKKT1 becomes:

$$0 \subseteq (\hat{+}) - [\lambda_1] + [\lambda_2] - [\mu_2] + [\mu_4] \tag{1}$$

$$0 \subseteq -[\lambda_1] + [\lambda_2] + [\mu_3] \tag{2}$$

$$0 \subseteq [\lambda_1] + [\mu_1] - [\mu_4] \tag{3}$$

The terms used to form the prime assignments are:

i	$\mathbf{N}_i$	$\mathbf{Z}_i$	$\mathbf{P}_i$
1	$[\lambda_1] = \hat{+}, [\lambda_2] = \hat{-}$ $[\mu_2] = \hat{+}$	$[\lambda_1] = 0, [\lambda_2] = 0,$ $[\mu_2] = 0, [\mu_4] = 0$	$[\lambda_1] = \hat{-}, [\lambda_2] = \hat{+},$ $[\mu_4] = \hat{+}$
2	$[\lambda_1] = \hat{+}, [\lambda_2] = \hat{-}$	$[\lambda_1] = 0, [\lambda_2] = 0,$ $[\mu_3] = 0$	$[\lambda_1] = \hat{-}, [\lambda_2] = \hat{+},$ $[\mu_3] = \hat{+}$
3	$[\lambda_1] = \hat{-}, [\mu_4] = \hat{+}$	$[\lambda_1] = 0, [\mu_1] = 0,$ $[\mu_4] = 0$	$[\lambda_1] = \hat{+}, [\mu_1] = \hat{+}$

The prime assignments are:

$$\{\{[\lambda_1] = \hat{+}\}, \{[\lambda_2] = \hat{-}\}, \{[\mu_2] = \hat{+}\}\} \qquad P(1)$$

$$\{\{[\lambda_1] = 0, [\lambda_2] = 0, [\mu_3] = 0\}, \{[\lambda_1] = \hat{-}, [\lambda_2] = \hat{-}\}\},$$

$$\{[\lambda_2] = \hat{+}, [\lambda_1] = \hat{+}\}, \{[\mu_3] = \hat{+}, [\lambda_1] = \hat{+}\}\}, \qquad P(2)$$

$$\{[\mu_3] = \hat{+}, [\lambda_2] = \hat{-}\}\}$$

$$\{\{[\lambda_1] = 0, [\mu_1] = 0, [\mu_4] = 0\}, \{[\lambda_1] = \hat{+}, [\mu_4] = \hat{+}\}\}, \qquad P(3)$$

$$\{[\mu_1] = \hat{+}, [\lambda_1] = \hat{-}\}, \{[\mu_1] = \hat{+}, [\mu_4] = \hat{+}\}\}.$$

The minimal set covering of $P(1)$–$P(3)$ results in seven prime assignments:

$$\{\{[\lambda_1] = \hat{+}, [\lambda_2] = \hat{+}, [\mu_4] = \hat{+}\},$$

$$\{[\lambda_1] = \hat{-}, [\lambda_2] = \hat{-}, [\mu_1] = \hat{+}\},$$

$$\{[\lambda_1] = 0, [\lambda_2] = \hat{-}, [\mu_1] = 0, [\mu_3] = \hat{+}, [\mu_4] = 0\},$$

$$\{[\lambda_2] = \hat{-}, [\mu_1] = \hat{+}, [\mu_3] = \hat{+}, [\mu_4] = \hat{+}\},$$

$$\{[\lambda_1] = 0, [\lambda_2] = 0, [\mu_1] = 0\}, \{[\mu_2] = \hat{+}, [\mu_3] = 0, [\mu_4] = 0\},$$

$$\{[\lambda_1] = 0, [\lambda_2] = 0, [\mu_1] = \hat{+}, [\mu_2] = \hat{+}, [\mu_3] = 0, [\mu_4] = \hat{+}\},$$

$$\{[\lambda_1] = \hat{+}, [\mu_2] = \hat{+}, [\mu_3] = \hat{+}, [\mu_4] = \hat{+}\}\}.$$

Extracting minimal sets $[\mu_i] = \hat{+}$, removing supersets, and mapping to a maximal qstationary subspace (minimal qstationary covering) gives:

$$\mathfrak{I}_1 \equiv (\{g_1, g_2, g_3\}\{h_1, h_2, g_4\}),$$

$$\mathfrak{I}_2 \equiv (\{g_2, g_3, g_4\}\{h_1, h_2, g_1\}),$$

$$\mathfrak{I}_3 \equiv (\{g_1, g_2, g_4\}\{h_1, h_2, g_3\}),$$

$$\mathfrak{I}_4 \equiv (\{g_1, g_3, g_4\}\{h_1, h_2, g_2\}),$$

where again $(\mathbf{g}, \mathbf{h})$ is a space defined by inequalities $\mathbf{g}$ and equalities $\mathbf{h}$. Step 7 removes overdetermined sets, of which there are none here.

Finally, the reduced optimization of the resulting sets $(\mathfrak{I}_1, \mathfrak{I}_2, \mathfrak{I}_3, \mathfrak{I}_4)$ becomes:

Given: vector $\mathbf{x} = (r\ \phi\ \tau)^T$,

1. Let $\mathbf{W} = \arg\min_{\mathrm{x}}(\rho L \pi r^2)$,

   Subject to:

$$(h_1 = 0)\ (h_2 = 0)\ (g_1 \leqslant 0)\ (g_2 \leqslant 0)\ (g_3 \leqslant 0)\ (g_4 = 0).$$

2. Let $\mathbf{X} = \arg\min_{\mathrm{x}}(\rho L \pi r^2)$,

   subject to:

$$(h_1 = 0)\ (h_2 = 0)\ (g_1 = 0)\ (g_2 \leqslant 0)\ (g_3 \leqslant 0)\ (g_4 \leqslant 0).$$

3. Let $\mathbf{Y} = \arg\min_{\mathrm{x}}(\rho L \pi r^2)$,

   Subject to;

$$(h_1 = 0)\ (h_2 = 0)\ (g_1 = 0)\ (g_2 \leqslant 0)\ (g_3 = 0)\ (g_4 \leqslant 0).$$

4. Let $\mathbf{Z} = \arg\min_{\mathrm{x}}(\rho L \pi r^2)$,

   subject to:

$$(h_1 = 0)\ (h_2 = 0)\ (g_1 \leqslant 0)\ (g_2 = 0)\ (g_3 \leqslant 0)\ (g_4 \leqslant 0).$$

5. Return $\arg\min_{\mathrm{x}}(\rho L \pi r^2)$

   subject to:

$$\mathbf{x} \in \mathbf{W} \cup \mathbf{X} \cup \mathbf{Y} \cup \mathbf{Z}.$$

The original problem has 1 DOF (3 variables, 2 equality constraints) while the final problem reduces to four constraint-bound problems requiring no numerical optimization.

8. RELATION TO MONOTONICITY ANALYSIS

Monotonicity analysis provides the inspiration for activity analysis. Based on monotonic arguments alone, the technique began as a set of principles and methods used by modelers to identify ill-posed optimization problems. The technique was later extended to partially solve optimization problems, identifying the global solution in constraint bound monotonic problems. Monotonicity analysis (Wilde[17], Papalambros and Wilde[13], Papalambros[12]) is based on two rules which test the boundedness of a problem formulation:

Rule 1: If the objective function is monotonic with respect to a variable, then there exists at least one active constraint which bounds the variable in the direction opposite of the objective function.

Rule 2: If a variable is not contained in the objective function then it must be either bounded from both above and below by active constraints, or not actively bounded at all (*i.e* in the latter case, any constraints monotonic with respect to that variable must be inactive or irrelevant).

Both of these rules can be derived from QKKT. A third rule, introduced by Wilde[18], called the Maximum Activity Principle, guarantees that a solution will not be overconstrained:

The number of non-redundant active constraints cannot exceed the total number of variables.

The Maximum Activity Principle has a similar effect as step 7 in the activity analysis algorithm. In monotonicity analysis active constraints are counted, while in activity analysis resulting equalities are counted, even if they are irrelevant.

Applying monotonicity analysis (exhaustively applying the rules) produces multiple sets of constraints such that all of the constraints in at least one set must be active to provide a well bounded problem (this is a necessary but not sufficient condition). As with activity analysis, the sets provide a reduced optimization problem which, if constraint bound, may describe the global optimum, and otherwise requires less numerical optimization.

Various levels of implementation of monotonicity analysis have been described in Choy and Agogino[5], Michelena and Agogino[11], Rao and Papalambros[15], Azarm and Papalambros[2], and Hansen, *et al.* [8]. Of particular relevance, Choy and Agogino[5] and Agogino and Almgren[1] propose the use of monotonicity as the basis of an intelligent optimal reasoning system. Cagan and Agogino[3, 4] use monotonicity analysis within a framework to optimally expand the design space to search for improved designs and induce optimal trends during the expansion process.

The problem activity analysis addresses is similar in spirit to that of monotonicity analysis; nevertheless, the approach differs. To analyze their relationship the relation between the underlying principles, the output produced by both techniques, and the algorithms used to produce those outputs are discussed. First, activity analysis operates directly on an abstraction (QKKT) of the Karush-Kuhn-Tucker (KKT) conditions of optimization theory. QKKT is sufficiently powerful that given only knowledge of monotonicities, the conclusions about optimality made from QKKT and KKT are equivalent. In addition, the two monotonicity principles can be derived from QKKT.

Second, the result of activity analysis is formulated precisely in terms of minimal qstationary coverings that insure correctness, maximize the focus achievable using monotonicity information, and guarantee that the solution is parsimonious. In particular, the covering is guaranteed to contain exactly those points satisfying QKKT. Thus, the solutions are guaranteed to be correct (*i.e* they contain all stationary points), yet maximize focus by removing all points that can be proven nonstationary. Finally, the fact that the qstationary subspaces in the covering are maximal guarantees the simplicity of the resulting description.

Although there is no formal statement characterizing the results of monotonicity analysis, the result it produces and that of activity analysis are similar. There are, however, important conceptual distinctions. The purpose of activity analysis is to construct a concise description of those feasible points satisfying QKKT. The philosophy of activity analysis is that the decisions about how to explore these subspaces are best based on the rich set of methods provided by traditional numerical optimization or other symbolic techniques. On the other hand, the goal of monotonicity analysis is to perform a case analysis on an optimization problem which reduces the problem to a set of cases each of which is as restricted as possible. Monotonicity analysis generates a decision tree whose conditionals are properties of known design parameters and whose leaves are optimization problems.

The cases generated using the monotonicity principles roughly correspond to the qstationary subspaces of activity analysis. However, monotonicity analysis further divides these cases into subcases by case splitting on whether or not each of the remaining inequalities is active or inactive. This case splitting is repeated until no inequalities remain, subject to the Maximum Activity Principle, or the solution is uniquely determined.

For example, returning to the hydraulic cylinder problem, a monotonicity analysis of the same problem reveals four cases in which the optima lie, namely:

case 1: h_2, g_1, g_2, g_3 active, where h_1 must also be satisfied;
case 2: h_1, h_2, g_1, g_4 active;
case 3: h_1, h_2, g_1, g_2, g_4 active;
case 4: h_1, h_2, g_1, g_3, g_4 active.

Observing activity of inequality constraints only, note that cases 1 and 2 are identical to the two sets of constraints generated by activity analysis. Figure 5 shows a tree of constraint activity for this problem. The active constraints for a parent vertex is a subset of those for each child. The bold line indicates where the tree ends for activity analysis, while the tree for monotonicity analysis also includes both the vertices below the line.

This approach of monotonicity analysis to case splitting may be reasonable when the cases are being evaluated by an engineer, since each case is simpler to understand and the analysis can result in a

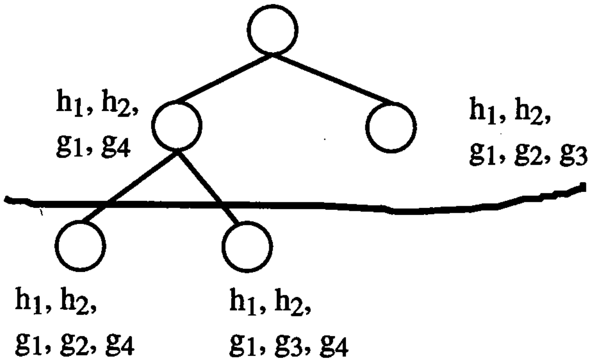

FIGURE 5 Tree of increasing constraint activity for hydraulic cylinder. Bold line separates activity analysis solution while monotonicity analysis also includes nodes below line.

parametric design chart. However, the danger is that the search space becomes fragmented, imposing artificial constraints that the optimization method is forced to respect. Thus, activity analysis defers these decisions to other methods; with respect to automated analysis, activity analysis takes the perspective that the way to explore and decompose the qstationary subspaces is best decided by the numerical or symbolic nonlinear optimization method chosen to explore each subspace.

Finally, in terms of algorithms, the activity analysis algorithm is simple. It reformulates the problem to one of generating prime assignments and introduces a complete prime assignment engine. This guarantees that the three properties of the output (*i.e* parsimony, maximum focus and correctness) are achieved.

9. CONCLUDING REMARKS

Activity analysis provides the following contributions: It formalizes the strategic way in which a modeler focuses optimization as the process of generating minimal qstationary coverings. It introduces QKKT as a powerful condition for quickly eliminating large, suboptimal subspaces. It exploits this condition through a novel problem reformulation based on the prime, implicating assignments of linear sign equations. The activity analysis algorithm classifies the design space into qstationary and qnonstationary subspaces, providing a reduced space in which further analysis can more efficiently be performed. Activity analysis has been implemented in *C* running on a Silicon Graphics Indigo and has been applied to a variety of engineering optimization problems. Activity analysis can be extended to provide explainable optimizers, ones that use QKKT to provide commonsense explanations about optimality. Possible directions also include an extension to activity analysis for cases where monotonicities are only partially known.

References

[1] Agogino, A. M. and Almgren, A. S. (1987) Techniques for integrating qualitative reasoning and symbolic computation in engineering optimization. *Engineering Optimization*, **12**, 117–135.

[2] Azarm, S. and Papalambros, P. (1984) An automated procedure for local monotonicity analysis. *Transactions of the ASME, Journal of Mechanisms, Transmissions, and Automation in Design*, **106**, 82–89.

[3] Cagan, J. and Agogino, A. M. (1987) Innovative design of mechanical structures from first principles. *AI EDAM: Artificial Intelligence in Engineering, Design, Analysis and Manufacturing*, **1** (3), 169–189.

[4] Cagan, J. and Agogino, A. M. (1991) Inducing constraint activity in innovative design. AI EDAM: *Artificial Intelligence in Engineering Design, Analysis, and Manufacturing*, **5** (1), 47–61.

[5] Choy, J. K. and Agogino, A. M. (1986) SYMON: Automated SYmbolic MONotoniciy analysis system for qualitative design optimization. In *Proceedings of ASME 1986 International Computers in Engineering Conference, Chicago*, 305–310.

[6] Corman, T., Leiserson, C. and Rivest, R. (1990) *Introduction to Algorithms.* MIT Press, Cambridge, MA, 974–978.

[7] de Kleer, J. and Williams, B. C. (1987) Diagnosing multiple faults. *Artificial Intelligence*, **32**, 97–130.

[8] Hansen, P., Jaumard, B. and Lu, S. H. (1989) An automated procedure for globally optimal design. *Transactions of the ASME, Journal of Mechanisms, Transmissions, and Automation in Design*, **111** 361–367.

[9] Karush, W. (1939) *Minima of Functions of Several variables with Inequalities as Side Conditions.* MS Thesis, Department of Mathematics, University of Chicago, Chicago, IL.

[10] Kuhn, H. W. and Tucker, A. W. (1951) Nonlinear programming. In Neyman, J., (ed.) *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability.* Berkeley, CA: University of California Press.

[11] Michelena, N. and Agogino, A. M. (1993) Monotonic influence diagrams: foundations and application to optimal design. *Engineering Optimization*, **21**,79–97.

[12] Papalambros, P. (1982) Monotonicity in goal and geometric programming. *Transactions of the ASME, Journal of Mechanical Design*, **104**, 108–113.

[13] Papalambros, P. and Wilde, D. J. (1979) Global non-iterative design optimization using monotonicity analysis. *Transactions of the ASME, Journal of Mechanical Design*, **101** (4), 645–649.

[14] Papalambros, P. and Wilde, D. J. (1988) *Principles of Optimal Design.* Cambridge University Press, New York.

[15] Rao, J. R. J. and Papalambros, P. (1987) Implementation of semi-heuristic reasoning for bounded analysis of design optimization models. In *Advances in Design Automation-1987, Proceedings of the ASME Design Automation Conference, 59–65.*

[16] Vanderplaats, G. N. (1984) *Numerical Optimization Techniques for Engineering Design with Applications.* McGraw-Hill, New York.

[17] Wilde, D. J. (1975) Monotonicity and dominance in optimal hydraulic cylinder design. *Transactions of the ASME, Journal of Engineering for Industry*, **94** (4), 1390–1394.

[18] Wilde, D. J. (1986) A maximal activity principle for eliminating overconstrainted optimization cases. *Transactions of the ASME, Journal of Mechanisms, Transmissions and Automation in Design*, **108**, 312–314.

[19] Williams, B. C. (1991) A theory of interactions: unifying qualitative and quantitative algebraic reasoning. *Artificial Intelligence Special Volume on Qualitative Reasoning About Physical Systems II*, **51**, 39–94.

[20] Williams, B. C. and Cagan, J. (1994) Activity analysis: The qualitative analysis of stationary points for optimal reasoning. In proceedings: *AAAI-94, 12th National Conference on Artificial Intelligence*, Seattle, WA, July, **2**, 1217–1223.