Discussion Paper No. 19

GENERALIZATION AND SYMMETRIZATION OF DUALITY IN GEOMETRIC PROGRAMMING

By

Elmor L. Peterson

October 16, 1972

GENERALIZATION AND SYMMETRIZATION OF DUALITY IN GEOMETRIC PROGRAMMING*

Elmor L. Peterson**

Abstract: The first completely symmetric formulation of duality for general convex programming with explicit constraints is achieved by further extending the author's recent symmetric formulation of duality for generalized unconstrained geometric programming. This extended formulation of duality is completely symmetric only in the convex case but is actually studied for the most part without convexity hypotheses. This study includes a domonstration of the equivalence of the extended formulation and the (seemingly) more special formulation that results from deleting all explicit constraints. This equivalence provides an efficient mechanism for generalizing to the extended formulation many important theorems that were previously established in the context of the more special formulation.

TABLE OF CONTENTS

---

* Research partially supported by a Summer Fellowship Grant and Urban Systems Engineering Center Grant Number 731, both from Northwestern University.

** Department of Industrial Engineering/Management Sciences and Department of Mathematics, Northwestern University, Evanston, Illinois 60201

1. Introduction. To illuminate and extend Zener's work [27,28] on optimal engineering design, Duffin [3] formulated a new duality theory for differentiable convex programming problems; and he applied it to the special but important class of "posynomial" minimization problems. The resulting specialized theory was then strengthened by Duffin [4]; and that strengthened theory was eventually extended by Duffin and Peterson [5] to the much larger class of posynomial-constrained posynomial minimization problems with which the subject of "geometric programming" [9,29] has subsequently been mainly concerned.

More recently, Peterson [13] generalized and completely symmetrized Duffin's original duality theory [3]; and he established new theorems that provide both a theoretical basis for certain important parametric analyses and an economic interpretation of the geometric dual problem. Some of those theorems were also used to help strengthen the original duality theory without imposing the special posynomial structure; and that strengthened duality theory was then related to both Fenchel's duality theory [11] and Rockafellar's duality theory [23,24,25].

This paper generalizes all previous formulations of duality in geometric programming by introducing explicit constraints in such a way that the duality symmetry achieved in [13] is not destroyed. (To date this is the only way to obtain a completely symmetric formulation of duality in general convex programming with explicit constraints.) The resulting formulation of duality actually turns out to be equivalent to the (seemingly) more special formulation in which all explicit constraints are deleted; and the demonstration of this fact provides an efficient mechanism through which the theory established in [13] can be applied to generalized geometric programming problems with explicit constraints.

This paper is actually rather self-contained in that the theory to be developed here depends only on the rather elementary properties of the "conjugate transformation" and the "geometric inequality" that are reviewed in section 3.

The theory that also depends on the results established in [13] will be included in a future paper.

2. Geometric Programming Families. Classical optimization theory and ordinary mathematical programming are concerned with the minimization (or maximization) of an arbitrary real-valued function G over some given subset S of the functions nonempty domain C. For pedagogical simplicity we shall restrict our attention to the finite-dimensional case in which C is itself a subset of N-dimensional Euclidean space E_N.

In the geometric programming format to be studied here both the given function G with domain C and the given subset S of C are required to have very specific (though very general) mathematical properties. In particular, G must be the sum of (at least) one arbitrary function g_0 with (possibly) other functions g_j^+ of a certain type; and C must be the cartesian product of (at least) the domain C_0 of g_0 with (possibly) other function domains C_i and (possibly) the domains C_j^+ of the functions g_j^+ (if any such function domains C_i or functions g_j^+ are present). Moreover, S must be determined not only in terms of the (possible) presence of arbitrary constraint functions g_i with the domains C_i, but also in terms of a certain cone X in the cartesian product of (at least) the Euclidean space E_{n_0} containing C_0 with (possibly) the Euclidean spaces E_{n_i} containing C_i and (possibly) the Euclidean spaces E_{n_j+1} containing C_j^+ (if any such domains C_i or C_j^+ are present). Both the arbitrary constraint functions g_i and the special functions g_j^+ along with their respective domains C_i and C_j^+ are not present in the geometric programming format studied in [13].

For both practical and theoretical reasons a given geometric programming problem should not be studied entirely in isolation but should also be embedded in a certain parameterized family A of closely related geometric programming problems $A(u,\mu)$. The given problem then appears in the parameterized family A as problem $A(0,0)$ and should be studied in relation to all other problems $A(u,\mu)$, with special attention given to those problems $A(u,\mu)$ that are close to $A(0,0)$ in the sense that (the norm of) the vector (u,μ) is small.

To give a precise definition of the most general geometric programming family A to be considered here it is convenient for bookkeeping purposes to introduce two nonintersecting (possibly empty) positive-integer index sets I and J with finite cardinality $o(I)$ and $o(J)$ respectively. In terms of these index sets I and J we introduce the following notation and hypotheses:

(Ia) For each $k \in \{0\} \cup I \cup J$ suppose that $g_k : C_k$ is a function g_k with a nonempty domain $C_k \subseteq E_{n_k}$, and for each $j \in J$ let D_j be a nonempty subset of E_{n_j} whose precise nature will be specified at the beginning of section 4.

(IIa) For each $k \in \{0\} \cup I \cup J$ let u^k be an independent vector parameter in E_{n_k}, and let μ be an independent vector parameter with components μ_i for each $i \in I$.

(IIIa) Denote the cartesian product of the vector parameters u^i, $i \in I$ by the symbol u^I, and denote the cartesian product of the vector parameters u^j, $j \in J$ by the symbol u^J. Then the cartesian product (u^0,u^I,u^J) of the vector parameters u^0, u^I, and u^J is an independent vector parameter u in E_n, where

$$n \triangleq n_0 + \sum_I n_i + \sum_J n_j .$$

(IVa) For each $k \in \{0\} \cup I \cup J$ let x^k be an independent vector variable in E_{n_k}, and let $\varkappa$ be an independent vector variable with components $\varkappa_j$ for each $j \in J$.

(Va) Denote the cartesian product of the vector variables x^i, $i \in I$ by the symbol x^I, and denote the cartesian product of the vector variables x^j, $j \in J$ by the symbol x^J. Then the cartesian product (x^0,x^I,x^J) of the vector variables x^0, x^I, and x^J is an independent vector variable x in E_n.

(VIa) Assume that X is a cone in E_n.

Now, consider the following (parameterized) <u>geometric programming family</u> A of geometric programmimg problems $A(u,\mu)$.

PROBLEM $A(u,\mu)$. <u>Consider the objective function</u> $G(\cdot+u,\varkappa):C(u)$ <u>whose domain</u>

$$C(u) \triangleq \{(x,\varkappa) \mid x^k+u^k \in C_k,\ k\in\{0\}\cup I,\ \underline{\text{and}}\ (x^j+u^j,\varkappa_j)\in C_j^+,\ j\in J\},$$

<u>and whose functional value</u>

$$G(x+u,\varkappa) \triangleq g_0(x^0+u^0) + \sum_J g_j^+(x^j+u^j,\varkappa_j),$$

<u>where</u>

$$C_j^+ \triangleq \{(c^j,\varkappa_j) \mid \underline{\text{either}}\ \varkappa_j=0\ \underline{\text{and}}\ \sup_{d^j\in D_j} \langle c^j,d^j\rangle < +\infty,\ \underline{\text{or}}\ \varkappa_j>0\ \underline{\text{and}}\ c^j\in\varkappa_j C_j\}$$

<u>and</u>

$$g_j^+(c^j,\varkappa_j) \triangleq \begin{cases} \sup_{d^j\in D_j} \langle c^j,d^j\rangle & \underline{\text{if}}\ \varkappa_j=0\ \underline{\text{and}}\ \sup_{d^j\in D_j}\langle c^j,d^j\rangle<+\infty \\ \varkappa_j g_j(c^j/\varkappa_j) & \underline{\text{if}}\ \varkappa_j>0\ \underline{\text{and}}\ c^j\in\varkappa_j C_j. \end{cases}$$

<u>Using the feasible solution set</u>

$$S(u,\mu) \triangleq \{(x,\varkappa)\in C(u) \mid x\in X,\ \underline{\text{and}}\ g_i(x^i+u^i)+\mu_i \le 0,\ i\in I\},$$

<u>calculate both the problem infimum</u>

$$\varphi(u,\mu) \triangleq \inf_{(x,\varkappa)\in S(u,\mu)} G(x+u,\varkappa)$$

<u>and the optimal solution set</u>

$$S^*(u,\mu) \triangleq \{(x,\varkappa)\in S(u,\mu) \mid G(x+u,\varkappa)=\varphi(u,\mu)\}.$$

For a given vector (u,μ) problem $A(u,\mu)$ is either "inconsistent" or "consistent", depending on whether the feasible solution set $S(u,\mu)$ is empty or nonempty. It is generally useful to interpret $A(u,\mu)$ as a perturbed version of $A(0,0)$, so the set

$$U \triangleq \{(u,\mu) \mid S(u,\mu) \text{ is nonempty}\}$$

is termed the feasible perturbation set. Note that there are no perturbation parameters associated with the variables κ_j; the reason is that such perturbations would clearly influence only the optimal value of κ_j and hence would essentially be superfluous.

The feasible perturbation set U generally consists of infinitely many vectors (u,μ) and is taken to be the domain of the infimum function φ. Thus the range of $\varphi:U$ may contain the point $-\infty$; but if $\varphi(u,\mu) = -\infty$ for some $(u,\mu) \in U$, the corresponding optimal solution set $S^*(u,\mu)$ is clearly empty.

It is important to make a sharp distinction between the cone condition $x \in X$ and the constraints $g_i(x^i + u^i) + \mu_i \leq 0$, $i \in I$, both of which restrict the vector variable (x,κ). In many cases the cone X is polyhedral (and hence finitely generated); and in most examples of practical significance X is actually a subspace (and hence has a finite basis). Consequently, the cone condition $x \in X$ can frequently be automatically satisfied and therefore explicitly eliminated by a linear transformation of the vector variable x through the introduction of generating vectors or basis vectors for X; but the (generally nonlinear) constraints $g_i(x^i + u^i) + \mu_i \leq 0$, $i \in I$ usually can not be explicitly eliminated by even a nonlinear transformation. Nevertheless, even when it is possible to do so, we do not explicitly eliminate the cone condition $x \in X$, because such a linear transformation would clearly introduce a common vector variable into the arguments of g_0, g_i, and g_j^+. Such a common vector variable would only tend to camouflage one of the extremely useful characteristics possessed by the geometric programming point of view - its (partial) separability.

A given mathematical programming problem can generally be put into the form of the geometric programming problem A(0,0) in many different ways by suitably choosing the functions $g_k:C_k$, $k \in \{0\} \cup I \cup J$ and the cone X. Actually, one very important aspect of applied geometric programming is the choosing of the functions $g_k:C_k$, $k \in \{0\} \cup I \cup J$ and the cone X so that a given inseparable programming problem is formulated as an equivalent geometric programming problem with as much separability as possible.

The key to such a formulation is usually the introduction of an appropriate nontrivial cone X to handle the linearities that are present in a given programming problem. Such linearities frequently appear as linear equations or linear inequalities, but they can also appear in rather subtle guises as matrices associated with nonlinearities.

In "signomial" programming [5,9,6,7,8] (formerly called "generalized polynomial" programming) the linearities appear in the rather subtle guise of "exponent matrices" for (nonlinear) generalized polynomials; and in quadratic programming (with quadratic constraints) [19,20,21,22] the linearities appear not only as coefficient vectors for linear functions but also in the more subtle guise of coefficient matrices for (nonlinear) quadratic functions. In both cases the inseparable programming problems can be formulated as equivalent geometric programming problems in which even the functions $g_k:C_k$, $k \in \{0\} \cup I \cup J$ are separable; in fact, each such function $g_k:C_k$ is completely separable in that it can be written as a sum of terms, each of which is itself a function of only a single scalar variable.

The perturbation parameters μ_i, $i \in I$ perturb the constraint upper bounds 0, and the perturbation parameters u^k, $k \in \{0\} \cup I \cup J$ translate and hence perturb the sets C_k, $k \in \{0\} \cup I \cup J$. These perturbations of the sets C_k actually amount to perturbations of (parameters that specify) the objective and constraint functions, by virtue of the invariant nature of the cone X. For example, in signomial programming they perturb (the logarithm of the absolute value of) the signomial coefficients - a type of perturbation that is generally of great interest in each of the many contexts in which signomial programming arises.

More detailed surveys and treatments of several important classes of programming problems that can be effectively formulated and studied within the present geometric programming format are given in [14] and the references cited therein. Decomposition principles that exploit the separability induced by geometric programming are given in [15,16] and the references cited therein. The fundamental relations between geometric programming and "ordinary programming" are given in [17]; and the fundamental relations between geometric programming and Rockafellar's "generalized programming" are given in [18] and the references cited therein.

To introduce the extremely important concept of duality into the present geometric programming format, we need the (more fundamental) "conjugate transformation" and the resulting "geometric inequality" that are described in the following section.

3. The Conjugate Transformation and the Geometric Inequality.

The conjugate transformation evolved from the classical "Legendre transformation" but was first studied in great detail only rather recently by Fenchel [10,11]. For a very thorough and modern treatment of both transformations see Rockafellar's recent book [25]. We now briefly review only those of their properties that are needed here. All such properties are quite plausible when viewed geometrically in the context of two and three dimensions.

The conjugate transformation maps functions into functions in such a way that the "conjugate transform" h:D of a given function g:C has functional values

$$h(y) \triangleq \sup_{x \in C} \{ \langle y,x \rangle - g(x) \}.$$

Of course, the domain D of h is defined to be the set of all those vectors y for which this supremum is finite, and the conjugate transform h:D exists only when D is not empty.

Geometrical insight into the conjugate transformation can be obtained by considering the "subgradient" set for g at x, namely,

$$\partial g(x) \triangleq \{y \in E_\eta \mid g(x) + \langle y, x' - x \rangle \leq g(x') \text{ for each } x' \in C\}.$$

Subgradients are related to, but considerably different from, the more familiar gradient. The gradient provides a "tangent hyperplane" while a subgradient provides a "supporting hyperplane" (in that the defining inequality obviously states that the hyperplane with equation $g' = g(x) + \langle y, x' - x \rangle$ intersects the "graph" of g at the point (x,g(x)) and lies entirely "on or below" it). It is, of course, clear that a subgradient may exist and not be unique even when the gradient does not exist. On the other hand it is also clear that a subgradient may not exist even when the gradient exists. There is, however, an important class of functions whose gradients are also subgradients - the class of convex functions. In fact, the notions of gradient and subgradient coincide for the class of differentiable convex functions defined on open sets, a class that arises in many geometric programming applications.

To relate the conjugate transform to subgradients, observe that if $y \in \partial g(x)$ then

$$\langle y, x' \rangle - g(x') \leq \langle y, x \rangle - g(x) \text{ for each } x' \in C,$$

which in turn clearly implies that $y \in D$ and that

$$h(y) = -\{g(x) + \langle y, -x \rangle\}.$$

Hence, h(y) is simply the negative of the intercept of the corresponding supporting hyperplane with the g'-axis. Consequently, the conjugate transform h always exists when g has at least one subgradient y, a condition that is known to be fulfilled when g is convex. Actually, the conjugate transform h restricted (in the set-theoretic sense) to the domain $\bigcup_{x \in C} \partial g(x)$ is termed the "Legendre transform" of g and has been a major tool in the study [1] of classical mechanics, thermodynamics, and differential equations. Generally, the domain D of the conjugate transform h consists of both $\bigcup_{x \in C} \partial g(x)$ and some of its limit points.

Each function g and its conjugate transform h give rise to an important inequality

$$\langle x,y \rangle \leq g(x) + h(y)$$

that is clearly valid for every point $x \in C$ and every point $y \in D$ (as can be seen from the defining equation for h(y)). Moreover, we have just shown that equality holds if

$$y \in \partial g(x),$$

a condition that actually characterizes equality by virtue of another elementary computation. This "conjugate inequality" (including the characterization of equality) has been used to establish many of the important classical inequalities [12] and has been of fundamental importance in the study [3,13] of geometric programming duality when explicit constraints are not present.

To study geometric programming duality when explicit constraints are present, the conjugate inequality has been extended in a very special way by Peterson [14]. The resulting "geometric inequality" (which removes certain undesireable restrictions required in the forerunners of Duffin and Peterson [5], and Duffin, Peterson, and Zener [9, Chapter VII]) can be derived directly from the conjugate inequality by introducing a scalar variable $\lambda \geq 0$. To do so, first suppose that $\lambda > 0$ and that y/λ is in D so that y/λ can be substituted for y in the preceding conjugate inequality. Then, multiply the resulting inequality by λ to establish the nontrivial part of the geometric inequality

$$\langle x,y\rangle \le \lambda g(x) + h^{+}(y,\lambda) \quad \text{for } x \in C \text{ and } (y,\lambda) \in D^{+},$$

where

$$D^{+} \triangleq \{(y,\lambda) \in E_{\eta+1} \mid \text{either } \lambda = 0 \text{ and } \sup_{c \in C} \langle c,y\rangle < +\infty, \text{ or } \lambda > 0 \text{ and } y \in \lambda D\}$$

and

$$h^{+}(y,\lambda) \triangleq \begin{cases} \sup_{c \in C} \langle c,y\rangle & \text{if } \lambda = 0 \text{ and } \sup_{c \in C} \langle c,y\rangle < +\infty \\ \lambda h(y/\lambda) & \text{if } \lambda > 0 \text{ and } y \in \lambda D. \end{cases}$$

Of course, the trivial part of this geometric inequality is an immediate consequence of the definition of $h^{+}(y,\lambda)$ for $\lambda = 0$. Moreover, it is clear from the equality characterization of the conjugate inequality that equality holds if and only if

$$\text{either } \lambda = 0 \text{ and } \langle x,y\rangle = \sup_{c \in C} \langle c,y\rangle, \text{ or } \lambda > 0 \text{ and } y \in \lambda \partial g(x).$$

Of course, another geometric inequality can be derived from the same conjugate inequality simply by introducing another scalar variable $\kappa \ge 0$ and substituting x/κ for x in the conjugate inequality. The details of that inequality are left to the reader.

When it exists, the conjugate transform is known to be both convex and "closed", in that its "epigraph" (which consists of all those points in $E_{\eta+1}$ that are "on or above" its graph) is both convex and (topologically) closed. In fact, Fenchel [10,11,25] has shown that the conjugate transformation provides a one-to-one mapping of the family of all closed convex functions onto itself in symmetric fashion (i.e. the mapping is its own inverse). Two such functions are said to be "conjugate functions" when they are the conjugate transform of one another.

If g is convex and closed, the preceding symmetry clearly implies that the condition $x \in \partial h(y)$ can replace the condition $y \in \partial g(x)$ in the characterization of equality for the conjugate inequality; in which case the relations $x \in \partial h(y)$ and $y \in \partial g(x)$ are equivalent and hence "solve" one another. Likewise, the condition $x \in \partial h(y/\lambda)$ can replace the condition $y \in \lambda\partial g(x)$ in the characterization of equality for the geometric inequality; in which case the relations $x \in \partial h(y/\lambda)$ and $y \in \lambda\partial g(x)$ solve one another when $\lambda > 0$.

In our study of geometric programming we must deal with both the arbitrary cone X and its "dual"

$$Y \triangleq \{y \in E_\eta \mid 0 \leq \langle x,y \rangle \text{ for each } x \in X\},$$

which is clearly itself a cone. Now, it is obvious that the conjugate transform of the zero function with domain X is just the zero function with domain - Y. Consequently, the theory of the conjugate transformation implies that Y is always convex and closed (a fact that can of course be established by more elementary considerations). Furthermore, if the cone X is also convex and closed, the symmetry of the conjugate transformation readily implies that the dual of Y is just X. If, in particular, X is a (vector) subspace of E_η, this symmetry readily implies the better-known symmetry between orthogonal complementary subspaces X and Y.

This completes our prerequisites for introducing the extremely important concept of duality into the present geometric programming format.

4. Geometric Programming Dual Families. Closely related to the given geometric programming family A is its geometric programming dual family B. To obtain B from A, we need the following additional notation and hypotheses:

(Ib) For each $k \in \{0\} \cup I \cup J$ suppose that the function $g_k:C_k$ has a conjugate transform $h_k:D_k$. Then $h_k:D_k$ is a closed convex function with a nonempty domain $D_k \subseteq E_{n_k}$. Moreover, for each $j \in J$ let the function domain D_j determine the non-empty subset D_j of E_{n_j} hypothesized in (Ia).

(IIb) For each $k \in \{0\} \cup I \cup J$ let v^k be an independent vector parameter in E_{n_k}, and let ν be an independent vector parameter with components ν_j for each $j \in J$.

(IIIb) Denote the cartesian product of the vector parameters v^i, $i \in I$ by the symbol v^I, and denote the cartesian product of the vector parameters v^j, $j \in J$ by the symbol v^J. Then the cartesian product (v^0, v^I, v^J) of the vector parameters v^0, v^I and v^J is an independent vector parameter v in E_n, where

$$n \triangleq n_0 + \sum_I n_i + \sum_J n_j .$$

(IVb) For each $k \in \{0\} \cup I \cup J$ let y^k be an independent vector variable in E_{n_k}, and let λ be an independent vector variable with components λ_i for each $i \in I$.

(Vb) Denote the cartesian product of the vector variables y^i, $i \in I$ by the symbol y^I, and denote the cartesian product of the vector variables y^j, $j \in J$ by the symbol y^J. Then the cartesian product (y^0, y^I, y^J) of the vector variables y^0, y^I and y^J is an independent vector variable y in E_n.

(VIb) Let Y be the dual of the cone X in E_n. Then Y is a closed convex cone in E_n.

Now, consider the following (parameterized) geometric programming family B of geometric programming problems $B(v, \nu)$.

PROBLEM $B(v,\nu)$. <u>Consider the objective function</u> $H(\cdot + v, \times): D(v)$ <u>whose domain</u>

$$D(v) \triangleq \{(y,\lambda) \mid y^k + v^k \in D_k,\ k \in \{0\} \cup J,\ \underline{\text{and}}\ (y^i + v^i, \lambda_i) \in D_i^+,\ i \in I\},$$

<u>and whose functional value</u>

$$H(y+v,\lambda) \triangleq h_0(y^0 + v^0) + \sum_I h_i^+(y^i + v^i, \lambda_i).$$

<u>where</u>

$$D_i^+ \triangleq \{(d^i,\lambda_i) \mid \underline{\text{either}}\ \lambda_i = 0\ \underline{\text{and}}\ \sup_{c^i \in C_i} \langle c^i, d^i \rangle < +\infty,\ \underline{\text{or}}\ \lambda_i > 0\ \underline{\text{and}}\ d^i \in \lambda_i D_i\}$$

<u>and</u>

$$h_i^+(d^i,\lambda_i) \triangleq \begin{cases} \sup_{c^i \in C_i} \langle c^i, d^i \rangle & \underline{\text{if}}\ \lambda_i = 0\ \underline{\text{and}}\ \sup_{c^i \in C_i} \langle c^i, d^i \rangle < +\infty \\ \lambda_i h_i(d^i/\lambda_i) & \underline{\text{if}}\ \lambda_i > 0\ \underline{\text{and}}\ d^i \in \lambda_i D_i. \end{cases}$$

Using the feasible solution set

$$T(v,\nu) \triangleq \{(y,\lambda) \in D(v) \mid y \in Y,\ \underline{\text{and}}\ h_j(y^j + v^j) + \nu_j \le 0,\ j \in J\},$$

calculate both the problem infimum

$$\psi(v,\nu) \triangleq \inf_{(y,\lambda) \in T(v,\nu)} H(y+v,\lambda)$$

and the optimal solution set

$$T^*(v,\nu) \triangleq \{(y,\lambda) \in T(v,\nu) \mid H(y+v,\lambda) = \psi(v,\nu)\}.$$

Families A and B are clearly of the same type, so the observations made about A are equally valid for B. Notice how B is obtained from A: the functions $g_k:C_k$, $k \in \{0\} \cup I \cup J$ are replaced by their respective conjugate transforms $h_k:D_k$, $k \in \{0\} \cup I \cup J$; the cone X is replaced by its dual Y in E_n; and the roles played by the two index sets I and J are interchanged. Consequently, if the given functions $g_k:C_k$, $k \in \{0\} \cup I \cup J$ and the given cone X are convex and closed, the symmetry in these three operations implies that the family obtained by applying the same transformation to B is again A. Because of this symmetry in the closed convex case, A and B are termed geometric dual families.

To compute the geometric dual B of a given family A, only the corresponding conjugate transforms $h_k:D_k$, $k \in \{0\} \cup I \cup J$ and the corresponding dual cone Y need to be computed. In all the important cases known to the author the given functions $g_k:C_k$, $k \in \{0\} \cup I \cup J$ are so separable and elementary that the former computations are rather easy exercises in the differential calculus. In all such cases the given cone X is polyhedral, and hence the latter computation can be performed with a certain linear algebraic algorithm devised by Uzawa [26]. Actually, in all such cases except one the latter computation involves only the well-known linear algebra of orthogonal complementary subspaces. For detailed descriptions of such cases see [14] and the references cited therein.

Each of the geometric dual families A and B contains a problem of special interest, namely, the unperturbed problems A(0,0) and B(0,0). Due to the apparent symmetry between them in the closed convex case, A(0,0) and B(0,0) are termed <u>geometric dual problems</u>. To avoid confusion, it is important to bear in mind that problems $A(u,\mu)$ and $B(v,\nu)$ are termed geometric dual problems only when (u,μ) and (v,ν) are zero.

Unlike the usual min-max formulations of duality in mathematical programming, both problem A(0,0) and its geometric dual problem B(0,0) are minimization problems. The relative simplicity of this min-min formulation will become clear in succeeding sections, but the reader who is accustomed to the usual min-max formulation must bear in mind that a given duality theorem will generally have slightly different statements depending on the formulation in use. In particular, a theorem that asserts the equality of the min and max in the usual formulation will assert that the sum of the mins is zero (i.e. $\varphi(0,0)+\psi(0,0)=0$) in the present formulation.

In the closed convex case the symmetry between the geometric dual families A and B induces a symmetry on the theory that relates A to B, in that each mathematical statement about A and B automatically produces an equally valid "dual statement" about B and A. To be concise, our attention will be focused on the family A, and each dual statement will be left to the reader's imagination.

In the next section we temporarily focus our attention on the unperturbed problem A(0,0) and its geometric dual problem B(0,0).

<u>5. Duality Gaps and the Extremality Conditions</u>. The theory to be established in this section brings to light some of the most important properties of the geo-

metric dual problems A(0,0) and B(0,0). This theory is a direct consequence of the conjugate and geometric inequalities. In fact, we need only make repeated use of the following fundamental lemma that results from those inequalities.

Lemma 5a. If $(x,\varkappa)$ is in the domain

$$C(0) \triangleq \{(x,\varkappa) \mid x^k \in C_k,\ k \in \{0\} \cup I, \text{ and } (x^j,\varkappa_j) \in C_j^+,\ j \in J\}$$

of the objective function $G(\cdot,\varkappa)$ for problem A(0,0), and if (y,λ) is in the domain

$$D(0) \triangleq \{(y,\lambda) \mid y^k \in D_k,\ k \in \{0\} \cup J, \text{ and } (y^i,\lambda_i) \in D_i^+,\ i \in I\}$$

of the objective function $H(\cdot,\varkappa)$ for problem B(0,0), then

$$\langle x,y\rangle \le G(x,\varkappa) + \sum_I \lambda_i g_i(x^i) + H(y,\lambda) + \sum_J \varkappa_j h_j(y^j),$$

with equality holding if and only if

$$y^0 \in \partial g_0(x^0),$$

$$\text{either } \lambda_i = 0 \text{ and } \langle x^i,y^i\rangle = \sup_{c^i \in C_i} \langle c^i,y^i\rangle, \text{ or } \lambda_i > 0 \text{ and } y^i \in \lambda_i \partial g_i(x^i), \quad i \in I,$$

$$\text{either } \varkappa_j = 0 \text{ and } \langle x^j,y^j\rangle = \sup_{d^j \in D_j} \langle x^j,d^j\rangle, \text{ or } \varkappa_j > 0 \text{ and } y^j \in \partial g_j(x^j/\varkappa_j), \quad j \in J.$$

Moreover, if **y** also satisfies the constraints

$$h_j(y^j) \le 0, \qquad j \in J,$$

of problem B(0,0), then

$$G(x,\varkappa) + \sum_I \lambda_i g_i(x^i) + H(y,\lambda) + \sum_J \varkappa_j h_j(y^j) \le G(x,\varkappa) + \sum_I \lambda_i g_i(x^i) + H(y,\lambda),$$

with equality holding if and only if

$$\varkappa_j h_j(y^j) = 0 \qquad j \in J.$$

<u>Furthermore</u>, <u>if</u> x <u>also satisfies the constraints</u>

$$g_i(x^i) \le 0, \qquad i \in I,$$

<u>of problem</u> A(0,0), <u>then</u>

$$G(x,\varkappa) + \sum_I \lambda_i g_i(x^i) + H(y,\lambda) \le G(x,\varkappa) + H(y,\lambda),$$

<u>with equality holding if and only if</u>

$$\lambda_i g_i(x^i) = 0 \qquad i \in I.$$

Proof. From the conjugate inequality we know that

$$< x^0, y^0 > \le g_0(x^0) + h_0(y^0),$$

with equality holding if and only if

$$y^0 \in \partial g_0(x^0).$$

From the geometric inequality we know that

$$< x^i, y^i > \le \lambda_i g_i(x^i) + h_i^+(y^i, \lambda_i),$$

with equality holding if and only if

$$\text{either } \lambda_i = 0 \text{ and } < x^i, y^i > = \sup_{c^i \in C_i} < c^i, y^i >, \text{ or } \lambda_i > 0 \text{ and } y^i \in \lambda_i \partial g_i(x^i), \qquad i \in I.$$

From the geometric inequality we also know that

$$< x^j, y^j > \le g_j^+(x^j, \varkappa_j) + \varkappa_j h_j(y^j),$$

with equality holding if and only if

$$\text{either } \varkappa_j = 0 \text{ and } < x^j, y^j > = \sup_{d^j \in D_j} < x^j, d^j >, \text{ or } \varkappa_j > 0 \text{ and } y^j \in \partial g_j(x^j/\varkappa_j), \qquad j \in J.$$

Adding all $1+o(I)+o(J)$ of these inequalities and taking account of the defining equations for x, y, G, and H proves the first assertion of Lemma 5a. The second assertion is an immediate consequence of the fact that $\kappa_j \geq 0$ when $(x^j, \kappa_j) \in C_j^+$, $j \in J$. Similarly, the third assertion is an immediate consequence of the fact that $\lambda_i \geq 0$ when $(y^i, \lambda_i) \in D_i^+$, $i \in I$. This completes our proof of Lemma 5a.

We now begin with the most basic and easily proved duality theorem, which leads directly to both the duality gap concept and the extremality conditions.

Theorem 5A. If (x,κ) and (y,λ) are feasible solutions to the geometric dual problems $A(0,0)$ and $B(0,0)$ respectively, then

$$0 \leq G(x,\kappa) + H(y,\lambda),$$

with equality holding if and only if

$$0 = \langle x, y \rangle,$$

$$y^0 \in \partial g_0(x^0),$$

either $\lambda_i = 0$ and $\langle x^i, y^i \rangle = \sup_{c^i \in C_i} \langle c^i, y^i \rangle$, or $\lambda_i > 0$ and $y^i \in \lambda_i \partial g_i(x^i)$, $\quad i \in I$,

either $\kappa_j = 0$ and $\langle x^j, y^j \rangle = \sup_{d^j \in D_j} \langle x^j, d^j \rangle$, or $\kappa_j > 0$ and $y^j \in \partial g_j(x^j/\kappa_j)$, $\quad j \in J$,

$$\lambda_i g_i(x^i) = 0, \ i \in I \quad \text{and} \quad \kappa_j h_j(y^j) = 0, \ j \in J.$$

Proof. The fact that x and y are in the cone X and its dual Y respectively combined with a sequential application of all three assertions of Lemma 5a shows that

$$0 \leq \langle x, y \rangle \leq G(x,\kappa) + \sum_I \lambda_i g_i(x^i) + H(y,\lambda) + \sum_J \kappa_j h_j(y^j) \leq G(x,\kappa) + \sum_I \lambda_i g_i(x^i) + H(y,\lambda)$$

$$\leq G(x,\kappa) + H(y,\lambda),$$

with equality holding in all four of these inequalities if and only if the equality conditions stated in the theorem are satisfied. q.e.d.

It is worth recalling that if X is in fact a subspace of E_n, then $Y = X^{\perp}$. In that event the equality condition $0 = \langle x,y \rangle$ is automatically satisfied by arbitrary feasible solutions (x,κ) and (y,λ), and hence it can be deleted from Theorem 5A and everywhere that Theorem 5A is used.

The basic inequality provided by Theorem 5A implies important properties of the dual infima $\varphi(0,0)$ and $\psi(0,0)$.

Corollary 5A1. If the geometric dual problems A(0,0) and B(0,0) are both consistent, then

(i) the infimum $\varphi(0,0)$ for problem A(0,0) is finite, and

$$0 \leq \varphi(0,0) + H(y,\lambda)$$

for each feasible solution (y,λ) to problem B(0,0),

(ii) the infimum $\psi(0,0)$ for problem B(0,0) is finite, and

$$0 \leq \varphi(0,0) + \psi(0,0).$$

The proof of this corollary is, of course, a trivial application of Theorem 5A.

Consistent geometric dual problems A(0,0) and B(0,0) for which $0 < \varphi(0,0) + \psi(0,0)$ are said to have a duality gap of $\varphi(0,0) + \psi(0,0)$. It is well-known that duality gaps do not occur in finite linear programming, but they do occasionally occur in infinite linear programming where this phenomenon was first encountered by Duffin [2]. Although duality gaps occur very frequently in the present (generally nonconvex) formulation of geometric programming, we shall eventually see that they can occur only very rarely in the convex case, in that they can then be excluded by very weak conditions on the geometric dual problems A(0,0) and B(0,0). Yet, they do occur in the convex case, even when I and J are both empty and $g_0 : C_0$ is closed; and examples (originally set forth by J.J. Stoer within the equivalent Fenchel formulation [11] of duality) are given in Appendix C of [13].

Geometric programming problems A(0,0) that are convex are usually much more amenable to study than those that are nonconvex, mainly because of the relative lack of duality gaps in the convex case. Duality gaps are undesirable from a theoretical point of view because we shall see that relatively little can be said about the corresponding geometric dual problems. They are also undesirable from a computational point of view because they usually destroy the possibility of using the inequality $0 \le G(x,\varkappa) + H(y,\lambda)$ to provide an algorithm stopping criterion.

Such a criterion results from specifying a positive tolerance ϵ so that the numerical algorithms being used to minimize both $G(x,\varkappa)$ and $H(y,\lambda)$ are terminated when they produce a pair of feasible solutions $(x^\dagger,\varkappa^\dagger)$ and $(y^\dagger,\lambda^\dagger)$ for which

$$G(x^\dagger,\varkappa^\dagger) + H(y^\dagger,\lambda^\dagger) \le 2\,\epsilon.$$

Because conclusion (i) to Corollary 5A1 and the defining property for $\varphi(0,0)$ show that

$$-H(y^\dagger,\lambda^\dagger) \le \varphi(0,0) \le G(x^\dagger,\varkappa^\dagger),$$

we conclude from the preceding tolerance inequality that

$$\left|\ \varphi(0,0) - \frac{G(x^\dagger,\varkappa^\dagger) - H(y^\dagger,\lambda^\dagger)}{2}\right| \le \epsilon.$$

Hence, $\varphi(0,0)$ can be approximated by $[G(x^\dagger,\varkappa^\dagger) - H(y^\dagger,\lambda^\dagger)]/2$ with an error no greater than $\pm\,\epsilon$; and, dually, $\psi(0,0)$ can be approximated by $[H(y^\dagger,\lambda^\dagger) - G(x^\dagger,\varkappa^\dagger)]/2$, also with an error no greater than $\pm\,\epsilon$.

Note, however, that the defining properties for $\varphi(0,0)$ and $\psi(0,0)$ imply that

$$\varphi(0,0) + \psi(0,0) \leq G(x,\varkappa) + H(y,\lambda)$$

for each pair of feasible solutions $(x,\varkappa)$ and (y,λ). Now, suppose that the geometric dual problems $A(0,0)$ and $B(0,0)$ have a duality gap, and let the positive tolerance ϵ be chosen so small that

$$2\epsilon < \varphi(0,0) + \psi(0,0).$$

(Of course, such a choice for ϵ is possible only if the dual problems $A(0,0)$ and $B(0,0)$ actually have a duality gap.) From the preceding two inequalities we easily infer that there are no feasible solutions $(x^{\dagger},\varkappa^{\dagger})$ and $(y^{\dagger},\lambda^{\dagger})$ for which

$$G(x^{\dagger},\varkappa^{\dagger}) + H(y^{\dagger},\lambda^{\dagger}) \leq 2\epsilon;$$

so the numerical algorithms being used to minimize both $G(x,\varkappa)$ and $H(y,\lambda)$ will never be terminated. Consequently, this algorithm stopping criterion may not be very useful for solving geometric dual problems $A(0,0)$ and $B(0,0)$ that have a duality gap, especially if the gap is rather large.

For those geometric dual problems $A(0,0)$ and $B(0,0)$ that do not have a duality gap, Theorem 5A provides a useful characterization of dual optimal solutions $(x^*,\varkappa^*)$ and (y^*,λ^*) in terms of the following <u>extremality conditions</u>:

(I) $\quad x \in X$ and $y \in Y$,

(II) $\quad g_i(x^i) \leq 0,\ i \in I$ and $h_j(y^j) \leq 0,\ j \in J$,

(III) $\quad 0 = \langle x,y \rangle$,

(IV) $\quad y^0 \in \partial g_0(x^0)$,

(V) $\quad$ either $\lambda_i = 0$ and $\langle x^i,y^i \rangle = \sup_{c^i \in C_i} \langle c^i,y^i \rangle$,

or $\lambda_i > 0$ and $y^i \in \lambda_i \partial g_i(x^i)$, $\quad i \in I$,

(VI) either $\varkappa_j = 0$ and $\langle x^j, y^j \rangle = \sup_{d^j \in D_j} \langle x^j, d^j \rangle$,

or $\varkappa_j > 0$ and $y^j \in \partial g_j(x^j/\varkappa_j)$, $j \in J$,

(VII) $\lambda_i g_i(x^i) = 0,\ i \in I,$ and $\varkappa_j h_j(y^j) = 0,\ j \in J,$

We formalize this characterization as the following corollary.

Corollary 5A2. *If the extremality conditions* (I-VII) *have at least one solution* $(x', \varkappa')$ *and* (y', λ'), *then*

(i) $(x', \varkappa') \in S^*(0,0)$ *and* $(y', \lambda') \in T^*(0,0)$,

(ii) $S^*(0,0) = \{(x,\varkappa) \mid (x,\varkappa)$ *and* (y',λ') *solve the extremality conditions* (I-VII)$\}$,

and

$T^*(0,0) = \{(y,\lambda) \mid (x',\varkappa')$ *and* (y,λ) *solve the extremality conditions* (I-VII)$\}$,

(iii) $0 = \varphi(0,0) + \psi(0,0)$.

On the other hand, if the geometric dual problems A(0,0) *and* B(0,0) *are both consistent and if* $0 = \varphi(0,0) + \psi(0,0)$, *then arbitrary vectors* $(x,\varkappa)$ *and* (y,λ) *are optimal solutions to problems* A(0,0) *and* B(0,0) *respectively if and only if* $(x,\varkappa)$ *and* (y,λ) *satisfy the extremality conditions* (I-VII).

The proof of this corollary is an immediate consequence of Theorem 5A and the conjugate transform relation $\partial g_k(x^k) \subseteq D_k$, $k \in \{0\} \cup I$ that was given in section 3.

The extremality conditions (I) are simply the "cone conditions" for problems A(0,0) and B(0,0); and the extremality conditions (II) are simply the "constraints" for problems A(0,0) and B(0,0). The extremality condition (III) is termed the *orthogonality condition*; the extremality conditions (IV-VI) are termed the *subgradient conditions*; and the extremality conditions (VII) are, of course, termed the *complementary slackness conditions*.

In the closed convex case the subgradient conditions (IV-VI) have several equivalent formulations that result from the symmetry of the conjugate transformation. In particular, the condition $y^0 \in \partial g_0(x^0)$ can be replaced by the equivalent condition $x^0 \in \partial h_0(y^0)$; the conditions $y^i \in \lambda_i \partial g_i(x^i)$, $i \in I$ can be replaced by the equivalent conditions $x^i \in \partial h_i(y^i/\lambda_i)$, $i \in I$; and the conditions $y^j \in \partial g_j(x^j/\varkappa_j)$, $j \in J$ can be replaced by the equivalent conditions $x^j \in \varkappa_j \partial h_j(y^j)$, $j \in J$.

In the closed convex case the second part of Corollary 5A2 and its (unstated) dual are very useful when $0 = \varphi(0,0) + \psi(0,0)$ and <u>both</u> $S^*(0,0)$ and $T^*(0,0)$ are known to be nonempty; because they then provide a method for calculating all optimal solutions from the knowledge of only a single optimal solution. For example, if $(x^*,\varkappa^*) \in S^*(0,0)$ is a known optimal solution to problem $A(0,0)$, then

$$T^*(0,0) = \{(y,\lambda) \mid (x^*,\varkappa^*) \text{ and } (y,\lambda) \text{ solve the extremality conditions (I-VII)}\},$$

and

$$S^*(0,0) = \{(x,\varkappa) \mid (x,\varkappa) \text{ and } (y^*,\lambda^*) \text{ solve the dual of the extremality conditions (I-VII)}\}$$

for each $(y^*,\lambda^*) \in T^*(0,0)$.

We have just seen that problem $A(0,0)$ need not always be solved directly. Under appropriate conditions it can actually be solved indirectly by solving either the extremality conditions (I-VII) or problem $B(0,0)$. In some cases it may be advantageous to solve the extremality conditions (I-VII), especially when they turn out to be linear. In other cases it may be advantageous to solve problem $B(0,0)$, especially when the index set I is nonempty while the index set J is empty; in which event problem $A(0,0)$ has constraints while problem $B(0,0)$ has no constraints. For a more thorough discussion of these aspects of geometric programming see [14] and the references cited therein.

In our fundamental Lemma 5a the reader probably noticed the appearance of the (ordinary) "Lagrangians" $G(x,\varkappa) + \sum_I \lambda_i g_i(x^i)$ and $H(y,\lambda) + \sum_J \varkappa_j h_j(y^j)$ for problems $A(0,0)$ and $B(0,0)$ respectively. Indeed, such Lagrangians play an important role in geometric programming, as shown in [17]. However, geometric programming is not nearly as dependent on the classical theory of such Lagrangians as is ordinary programming. In fact, the remaining section of this paper shows that a very large and important part of geometric programming has essentially nothing to do with the presence of explicit constraints (and hence such Lagrangians).

6. Equivalence of the Constrained and Unconstrained Formulations. The present constrained formulation of geometric programming can of course be specialized to its corresponding unconstrained formulation, simply by letting both index sets I and J be empty. In doing so we drop the (now unnecessary) subscript 0 from the symbol $g_0:C_0$ for the objective function and its domain; and we also replace all remaining geometric programming symbols with their script counterparts in order to avoid ambiguous notation when reversing this specialization.

We suppose then that $\mathcal{g}:\mathcal{C}$ is a function $\mathcal{g}$ with a nonempty domain $\mathcal{C} \subseteq E_n$, and we assume that $\mathcal{X}$ is a cone in E_n. The (parameterized) geometric programming family $\mathcal{A}$ being considered then consists of the following geometric programming problems $\mathcal{A}(u)$.

PROBLEM $\mathcal{A}(u)$. Using the feasible solution set

$$\mathcal{S}(u) \triangleq \mathcal{X} \cap (\mathcal{C} - u),$$

calculate both the problem infimum

$$\varphi(u) \triangleq \inf_{x \in \mathcal{S}(u)} \mathcal{g}(x+u)$$

and the optimal solution set

$$\mathcal{S}^*(u) \triangleq \{x \in \mathcal{S}(u) \mid \mathcal{g}(x+u) = \varphi(u)\}.$$

If the function $\mathcal{g}:\mathcal{C}$ has a conjugate transform $h:\mathcal{D}$, then the geometric dual $\mathcal{B}$

of the family $\mathcal{A}$ (as constructed in section 4) clearly exists and is of course defined in terms of $h:\mathcal{D}$ and the dual $\mathcal{Y}$ of the cone $\mathcal{X}$. In particular then, $\mathcal{B}$ clearly consists of the following geometric programming problems $\mathcal{B}(v)$.

PROBLEM $\mathcal{B}(v)$. <u>Using the feasible solution set</u>

$$\mathcal{T}(v) \triangleq \mathcal{Y} \cap (\mathcal{D} - v),$$

<u>calculate both the problem infimum</u>

$$\psi(v) \triangleq \inf_{y \in \mathcal{T}(v)} h(y+v)$$

<u>and the optimal solution set</u>

$$\mathcal{T}^*(v) \triangleq \{ y \in \mathcal{T}(v) \mid h(y+v) = \psi(v) \}.$$

In contrast with the constrained formulation of geometric programming given in sections 2 and 4 notice the innate simplicity of the preceding unconstrained formulation. That simplicity was a great aid in uncovering most of the theorems established in [13]. The somewhat surprising fact is that such theorems can actually be applied to the (seemingly) more general constrained formulation. The mechanism for doing so is the following specialization of the unconstrained formulation.

Introducing an additional independent vector variable α with components α_i for each $i \in I$, we let the functional domain

$$\mathcal{C} \triangleq \{ (x^0, x^I, \alpha, x^J, \varkappa) \in E_{\eta} \mid x^0 \in C_0;\ x^i \in C_i,\ \alpha_i \in E_1, \text{ and }$$
$$g_i(x^i) + \alpha_i \le 0,\ i \in I;\ (x^j, \varkappa_j) \in C_j^+,\ j \in J \};$$

and we let the functional value

$$\mathcal{g}(x^0, x^I, \alpha, x^J, \varkappa) \triangleq g_0(x^0) + \sum_J g_j^+(x^j, \varkappa_j) \triangleq G(x, \varkappa).$$

We also let the cone

$$\mathcal{X} \triangleq \{(x^0, x^I, \alpha, x^J, \varkappa) \in E_{\eta} \mid (x^0, x^I, x^J) \in X;\ \alpha = 0;\ \varkappa \in E_{o(J)}\}.$$

Then, problem $\mathcal{A}(0)$ is clearly identical to problem A(0,0). In fact, it is easy to see that problem $\mathcal{A}(u)$ is identical to problem A(u,μ) when:

(1a) the parameters u^0, u^I, and u^J corresponding to x^0, x^I, and x^J respectively are identified with the parameters u^0, u^I, and u^J respectively,

(2a) the parameter u_i corresponding to α_i is identified with the parameter μ_i for each $i \in I$,

(3a) the parameter u_j corresponding to $\varkappa_j$ is set equal to zero for each $j \in J$. Of course, the parameter u_j corresponding to $\varkappa_j$ clearly influences only the optimal value of $\varkappa_j$, so setting u_j equal to zero deletes from the family $\mathcal{A}$ only problems $\mathcal{A}(u)$ that are essentially superfluous. Consequently, the family $\mathcal{A}$ is essentially identical to the family A.

The crucial question now is whether the family $\mathcal{B}$ is essentially identical to the family B. To obtain the answer, we need to compute both the conjugate transform $h:\mathcal{D}$ of the given function $\mathcal{g}:\mathcal{C}$ and the dual $\mathcal{Y}$ of the given cone $\mathcal{X}$.

To compute $h:\mathcal{D}$, first note that

$$h(y^0, y^I, \lambda, y^J, \beta) = \sup_{(x^0, x^I, \alpha, x^J, \varkappa) \in \mathcal{C}} \{\langle y^0, x^0\rangle + \sum_I \langle y^i, x^i\rangle + \sum_J \langle y^j, x^j\rangle + \sum_I \lambda_i \alpha_i + \sum_J \beta_j \varkappa_j - g_0(x^0) - \sum_J g_j^+(x^j, \varkappa_j)\},$$

which is clearly finite only if $\lambda_i \geq 0$, $i \in I$; in which case we readily see that this expression

$$= \sup_{x^0 \in C_0} [\langle y^0, x^0 \rangle - g_0(x^0)] + \sum_I \sup_{x^i \in C_i} [\langle y^i, x^i \rangle - \lambda_i g_i(x^i)] +$$

$$\sum_J \sup_{(x^j, \varkappa_j) \in C_j^+} [\langle y^j, x^j \rangle + \beta_j \varkappa_j - g_j^+(x^j, \varkappa_j)].$$

Consequently, $(y^0, y^I, \lambda, y^J, \beta) \in \mathcal{D}$ if and only if both $\lambda_i \geq 0, i \in I$ and each term on the right-hand side of the preceding equation is finite. Of course, the first term is finite if and only if $y^0 \in D_0$, in which case the first term is equal to $h_0(y^0)$. The finiteness of the remaining terms can be conveniently characterized with two lemmas.

The following lemma characterizes the finiteness of the terms involving the index set I.

Lemma 6a. Given that $\lambda_i \geq 0$, the $\sup_{x^i \in C_i} [\langle y^i, x^i \rangle - \lambda_i g_i(x^i)]$ is finite if and only if $(y^i, \lambda_i) \in D_i^+$, in which case

$$\sup_{x^i \in C_i} [\langle y^i, x^i \rangle - \lambda_i g_i(x^i)] = h_i^+(y^i, \lambda_i).$$

Proof. Simply observe that

$$\sup_{x^i \in C_i} [\langle y^i, x^i \rangle - \lambda_i g_i(x^i)] = \begin{cases} \sup_{x^i \in C_i} \langle y^i, x^i \rangle & \text{if } \lambda_i = 0 \\ \lambda_i h_i(y^i/\lambda_i) & \text{if } \lambda_i > 0 \text{ and } y^i \in \lambda_i D_i \\ +\infty & \text{if } \lambda_i > 0 \text{ and } y^i \notin \lambda_i D_i, \end{cases}$$

and then use the defining formula for $h_i^+ : D_i^+$ q.e.d.

The next lemma characterizes the finiteness of the terms involving the index set J.

Lemma 6b. The $\sup_{(x^j, \varkappa_j) \in C_j^+} [\langle y^j, x^j \rangle + \beta_j \varkappa_j - g_j^+(x^j, \varkappa_j)]$ is finite if and only if both $y^j \in D_j$ and $h_j(y^j) + \beta_j \leq 0$, in which case

$$\sup_{(x^j,\varkappa_j)\in C_j^+} [\langle y^j, x^j\rangle + \beta_j\varkappa_j - g_j^+(x^j,\varkappa_j)] = 0.$$

Proof. First, observe that

$$\sup_{(x^j,\varkappa_j)\in C_j^+} [\langle y^j, x^j\rangle + \beta_j\varkappa_j - g_j^+(x^j,\varkappa_j)]$$

$$= \sup_{\varkappa_j \geq 0} [\sup_{x^j} \{\langle y^j, x^j\rangle + \beta_j\varkappa_j - g_j^+(x^j,\varkappa_j) \mid (x^j,\varkappa_j)\in C_j^+\}]$$

$$= \sup_{\varkappa_j \geq 0} [\beta_j\varkappa_j + \sup_{x^j} \{\langle y^j, x^j\rangle - g_j^+(x^j,\varkappa_j) \mid (x^j,\varkappa_j)\in C_j^+\}]$$

$$= \sup_{\varkappa_j \geq 0} \left[\beta_j\varkappa_j + \begin{pmatrix} \sup_{x^j} \{\langle y^j, x^j\rangle - \sup_{d^j\in D_j} \langle x^j, d^j\rangle \mid \sup_{d^j\in D_j} \langle x^j, d^j\rangle < +\infty\} & \text{if } \varkappa_j = 0 \\ \\ \sup_{x^j} \{\langle y^j, x^j\rangle - \varkappa_j g_j(x^j/\varkappa_j) \mid x^j/\varkappa_j \in C_j\} & \text{if } \varkappa_j > 0 \end{pmatrix}\right]$$

$$= \sup_{\varkappa_j \geq 0} \left[\beta_j\varkappa_j + \begin{pmatrix} 0 & \text{if } \varkappa_j = 0 \text{ and } y^j \in \overline{D}_j \\ +\infty & \text{if } \varkappa_j = 0 \text{ and } y^j \notin \overline{D}_j \\ +\infty & \text{if } \varkappa_j > 0 \text{ and } y^j \notin D_j \\ \varkappa_j h_j(y^j) & \text{if } \varkappa_j > 0 \text{ and } y^j \in D_j \end{pmatrix}\right],$$

where the final step makes use of the fact that the zero function with domain $\overline{D}_j$ (the topological closure of D_j) is the conjugate transform of the conjugate transform of the zero function with domain D_j. Now, note that the last expression is finite only if $y^j \in D_j$, in which case the last expression clearly

$$= \sup_{\varkappa_j \geq 0} [\beta_j\varkappa_j + \varkappa_j h_j(y^j)].$$

But this expression is obviously finite if and only if $h_j(y^j) + \beta_j \leq 0$, in which case this expression is clearly zero. q.e.d.

We have now shown that the functional domain

$$\mathcal{D} = \{ (y^0, y^I, \lambda, y^J, \beta) \in E_n \mid y^0 \in D_0;\ (y^i, \lambda_i) \in D_i^+,\ i \in I;\ y^j \in D_j,\ \beta_j \in E_1, \text{ and } h_j(y^j) + \beta_j \le 0,\ j \in J\};$$

and we have also shown that the functional value

$$h(y^0, y^I, \lambda, y^J, \beta) = h_0(y^0) + \sum_I h_i^+(y^i, \lambda_i) \triangleq H(y, \lambda).$$

Moreover, elementary considerations show that the cone

$$\mathcal{Y} = \{(y^0, y^I, \lambda, y^J, \beta) \in E_n \mid (y^0, y^I, y^J) \in Y;\ \lambda \in E_{o(I)};\ \beta = 0\}.$$

Therefore, problem $\mathcal{B}(0)$ is clearly identical to problem B(0,0). In fact, it is easy to see that problem $\mathcal{B}(\upsilon)$ is identical to problem $B(v, \nu)$ when:

(1b) the parameters υ^0, υ^I, and υ^J corresponding to y^0, y^I, and y^J respectively are identified with the parameters v^0, v^I, and v^J respectively,

(2b) the parameter υ_j corresponding to β_j is identified with the parameter ν_j for each $j \in J$,

(3b) the parameter υ_i corresponding to λ_i is set equal to zero for each $i \in I$. Of course, the parameter υ_i corresponding to λ_i clearly influences only the optimal value of λ_i, so setting υ_i equal to zero deletes from the family $\mathcal{B}$ only problems $\mathcal{B}(\upsilon)$ that are essentially superfluous. Consequently, the family $\mathcal{B}$ is essentially identical to the family B.

Many theorems concerning the present constrained formulation of geometric programming can now be readily obtained from comparable theorems concerning the present unconstrained formulation. The mechanism for doing so is twofold: (1) using the definition of the function $\mathcal{g}:\mathcal{C}$ in terms of the functions $g_0:C_0$, $g_i:C_i$, and $g_j^+:C_j^+$, find hypotheses about $g_0:C_0$, $g_i:C_i$, and $g_j^+:C_j^+$ that imply the required hypotheses about $\mathcal{g}:\mathcal{C}$; and (2) using the definition of the cone $\mathcal{X}$ in terms of the cone X, find hypotheses about X that imply the required hypotheses about $\mathcal{X}$.

An inspection of the definitions of $\mathcal{g}:\mathcal{C}$ and $\mathcal{X}$ indicates that a knowledge of the algebraic and topological properties of cartesian products is one major prerequisite for exploiting this mechanism. The only other major prerequisite for doing so is of course a knowledge of the theorems concerning the present unconstrained formulation of geometric programming. Many such theorems have already been given in [13], but only for the case in which $\mathcal{g}:\mathcal{C}$ is a closed convex function and $\mathcal{X}$ is a subspace of E_n. (Actually, script notation is not used in [13], and A and B are designated by A_1 and B_1 respectively.)

The most powerful theorems definitely require at least some convexity hypothesis in one form or another on $\mathcal{g}:\mathcal{C}$. Yet, in view of some recent work of Duffin and Peterson [6,7,8] such convexity hypotheses are not nearly as restrictive as they may seem. Closedness hypotheses are even less restrictive in that the properties of a given convex programming problem with at least one nonclosed convex function can usually be obtained from the properties of the convex programming problem that results from replacing all nonclosed convex functions with their respective "closures".

On the other hand, the most powerful theorems definitely do not require that $\mathcal{X}$ be a subspace of E_n. In fact, the note added in proof at the end of [13] indicates that $\mathcal{X}$ need only be a closed convex cone in E_n, a rather unrestrictive hypothesis in view of the real-world applications known to the author.

A weakening of the hypotheses used in [13] will be included in a future paper. The resulting theorems will then be applied to the present constrained formulation of geometric programming.

An equally important source of theorems for the present constrained formulation of geometric programming is the work of Rockafellar [25]. The mechanism for exploiting that source of theorems is provided in [18].

REFERENCES

1. Courant, R., and D. Hilbert, Methods of Mathematical Physics, Volume I, Interscience, New York, 1953.

2. Duffin, R.J., Infinite programs, Linear Inequalities and Related Systems, Annals of Mathematical Studies, No. 38, H.W. Kuhn and A.W. Tucker, eds., Princeton University Press, Princeton, 1956, p. 157.

3. Duffin, R.J., Dual programs and minimum cost, SIAM Jour. Appl. Math., 10(1962), p. 119.

4. Duffin, R.J., Cost minimization problems treated by geometric means, ORSA Jour., 10(1962), p. 668.

5. Duffin, R. J., and E.L. Peterson, Duality theory for geometric programming, SIAM Jour. Appl. Math., 14(1966), p. 1307.

6. Duffin, R.J., and E.L. Peterson, Geometric programming with signomials, Jour. Opt. Th. Appls., to appear.

7. Duffin, R.J., and E.L. Peterson, Reversed geometric programs treated by harmonic means, Indiana Univ. Math. Jour., to appear.

8. Duffin, R.J., and E.L. Peterson, The proximity of (algebraic) geometric programming to linear programming, Math. Prog. Jour., to appear.

9. Duffin, R.J., E.L. Peterson, and C. Zener, Geometric Programming - Theory and Application, Wiley, New York, 1967.

10. Fenchel, W., On conjugate convex functions, Canadian Jour. Math., 1(1949), p. 73.

11. Fenchel, W., Convex Cones, Sets and Functions, mineographed lecture notes, Princeton University, Princeton, 1951.

12. Hardy, G.H., J.E. Littlewood, and G. Polya, Inequalities, Cambridge University Press, Cambridge, 1959.

13. Peterson, E.L., Symmetric duality for generalized unconstrained geometric programming, SIAM Jour. Appl. Math., 19(1970), p. 487.

14. Peterson, E.L., Geometric programming and some of its extensions, The Impact of Optimization Theory on Technological Design, M. Avriel, M. Rijckaert, and D. Wilde, eds., Prentice-Hall, New York, to appear.

15. Peterson, E.L., The decomposition of large (generalized) geometric programming problems by tearing, Proc. NATO Inst. Large Scale Systems, D.M. Himmelblau, ed., North Holland, Amsterdam, to appear.

16. Peterson, E.L., The decomposition of large (generalized) geometric programming problems by geometric Lagrangians, in preparation.

17. Peterson, E.L., Relations between geometric programming and ordinary programming, in preparation.

18. Peterson, E.L., Relations between geometric programming and Rockafellar's generalized programming, in preparation.

19. Peterson, E.L., and J.G. Ecker, Geometric programming: a unified duality theory for quadratically constrained quadratic programs and ℓ_p - constrained ℓ_p - approximation problems, Bull. Am. Math. Soc., 74(1968), p. 316.

20. Peterson, E.L., and J.G. Ecker, Geometric programming: duality in quadratic programming and ℓ_p - approximation I, Proc. Princeton Symp. Math. Prog., ed. H.W. Kuhn, Princeton University Press, Princeton, 1970, p. 445.

21. Peterson, E.L., and J.G. Ecker, Geometric programming: duality in quadratic programming and ℓ_p - approximation II (canonical programs), SIAM Jour. Appl. Math., 17(1969), p. 317.

22. Peterson, E.L., and J.G. Ecker, Geometric programming: duality in quadratic programming and ℓ_p - approximation III (degenerate programs), Jour. Math. Anal. Appls., 29(1970), p. 365.

23. Rockafellar, R.T., Duality and stability in extremum problems involving convex functions, Pacific Jour. Math., 21(1967), p. 167.

24. Rockafellar, R.T., Duality in nonlinear programming, Amer. Math. Soc. Lects. Appl. Math., Vol. 11, G.B. Dantzig and A. . Veinott, eds., Amer. Math. Soc., Providence, 1968, p. 401.

25. Rockafellar, R.T., Convex Analysis, Princeton University Press, Princeton, 1970.

26. Uzawa, H., A theorem on convex polyhedral cones, Studies in Linear and Nonlinear Programming, K.J. Arrow, L. Hurwicz, and H. Uzawa, eds., Stanford University Press, Stanford, 1958, p. 23.

27. Zener, C., A mathematical aid in optimizing engineering designs, Proc. Nat. Acad. Sci., 47(1961), p. 537.

28. Zener, C., A further mathematical aid in optimizing engineering designs, Proc. Nat. Acad. Sci., 48(1962), p. 518.

29. Zener, C., Engineering Design by Geometric Programming, Wiley, New York, 1971.