

Exponential transformation

From optimization

Author: Daniel Garcia (ChBE 345)

Stewards: Dajun Yue and Prof. Fengqi You

Date presented: May 25, 2014

This article concerns the exponential transformation method for globally solving posynomial (or general geometric/signomial) optimization problems with nonconvex objective functions or constraints. A discussion of the method's development and use will be presented.

Contents

- 1 Background
 - 1.1 Example
 - 1.2 Development
- 2 Formulation
 - 2.1 Feasibility Analysis
- 3 Applications
- 4 Examples
- 5 References

Background

Before discussing methods to solve posynomial optimization problems, a brief review of posynomials is of use. A posynomial, as defined by Duffin, Peterson, and Zener (1967) as a function of the form

$$f(x_1, x_2, \dots, x_n) = \sum_{k=1}^K c_k x_1^{a_{1k}} \dots x_n^{a_{nk}}$$

where the variables x_i and the coefficients c_k are positive, real numbers, and all of the exponents a_{ik} are real numbers. For example,

$$f(x_1, x_2) = 14x_1^{3.2}x_2^4 + x_1^2x_2^7$$

is a posynomial in two variables, and

$$f(x_1) = 1500x_1^{3/5}$$

is a posynomial with one variable.

The process of formulating a problem as a geometric program (named after the geometric-arithmetic mean inequality) is called *GP modeling*, and has been gaining traction as a means to solve a variety of problems. A geometric program may be represented as follows:

$$\begin{aligned} \min \quad & f_0(x) \\ \text{s.t.} \quad & f_i(x) \leq 1, i = 1, \dots, m \\ & h_i(x) = 1, i = 1, \dots, p \end{aligned}$$

Where $f_0, \dots, f_m$ are posynomials and $h_1, \dots, h_p$ are monomials, and the variables are all positive.

Note that this formulation is more restrictive than a typical nonlinear program (NLP). Specifically, the GP formulation is more constrained in the form of the objective function and the constraints. This represents a potential payoff - while it is more difficult to formulate a problem as a GP due to these constraints, if a problem can be formulated as a GP, highly efficient, global solving methods can be employed. In the NLP case, one may have to settle for a local solution, if a solution can be found at all. Of course, GP's have their own set of solving challenges.

For example, it is not difficult to imagine a posynomial that is nonconvex, such as the examples above. Unfortunately, this can cause some problems when attempting to find a globally optimal solution of a posynomial program, as it is known that only convex problems *guarantee* a globally optimal solution.

Posynomial or geometric programming has been applied to solve problems in varied fields, such as signal circuit design, engineering design, project management, and inventory management, just to name a few. Clearly, the solution of such problems are important to the chemical engineer, and being able to globally solve such problems will equip the engineer with a powerful tool to solve a myriad of problems.

Example

A variety of problems can be algebraically reformulated into GP's. As an example, consider the optimization of a box with height h , width w , and depth d . As constraints, the total area of the walls, $2(hw+hd)$, is limited to some A_{wall} , the floor area wd is limited to some A_{floor} , and there are upper and lower constraints on the ratios h/w and w/d . The volume of the box, hwd is to be optimized:

$$\begin{aligned} \max \quad & hwd \\ \text{s.t.} \quad & 2(hw + hd) \leq A_{wall} \\ & wd \leq A_{floor} \\ & \alpha \leq h/w \leq \beta \\ & \gamma \leq d/w \leq \delta \end{aligned}$$

While this problem is not in the standard GP form shown above, minor algebraic manipulations allow for the rearranging of the constraints and objective into a standard GP:

$$\begin{aligned} \min \quad & h^{-1}w^{-1}d^{-1} \\ \text{s.t.} \quad & (2/A_{wall})hw + (2/A_{wall})hd \leq 1 \\ & (1/A_{floor})wd \leq 1 \\ & \alpha h^{-1}w \leq 1 \\ & (1/\beta)hw^{-1} \leq 1 \end{aligned}$$

$$\begin{aligned}\gamma w d^{-1} &\leq 1 \\ (1/\delta) w^{-1} d &\leq 1\end{aligned}$$

Development

Many researchers attempted to solve such problems starting in the 1960's and 1970's. Methods used in the day aimed to find only locally optimal solutions, and employed methods such as successive approximation of posynomials (called "condensation"), so-called "psuedo-duality" methods which use a weaker form of duality, and adapted nonlinear programming methods. While locally optimal solutions are certainly better than no solution at all, the desire to find a globally optimal solution was strong enough to spur the development of other methods for posynomial programs in the 1990's. Such methods included global optimization algorithms based on exponential variable transformations of the original posynomial program, convex relaxation of the original problem, and branch-and-bound-type methods.

The exponential transformation method was introduced by Costas Maranas and Christodoulos Floudas in 1997, and, under certain circumstances, provides a method to globally optimize certain posynomial programming problems - a review of the basics of formulation and theory is presented in the next section.

Formulation

Geometric programs with posynomials should be transformed to a convex problem for quick solving times. This can be achieved with an exponential transformation, as described and formulated in this section.

A posynomial program may be written as a generalized geometric problem (GGP) as so:

$$\begin{aligned}\min \quad & G_0(t) = G_0^+(t) - G_0^-(t) \\ \text{s.t.} \quad & G_j(t) = G_j^+(t) - G_j^-(t) \leq 0, j = 1, \dots, M \\ & t_i \geq 0, i = 1, \dots, N\end{aligned}$$

Where:

$$\begin{aligned}G_j^+(t) &= \sum_{k \in K_j^+} c_{jk} \prod_{i=1}^N t_i^{\alpha_{ijk}}, j = 0, \dots, M \\ G_j^-(t) &= \sum_{k \in K_j^-} c_{jk} \prod_{i=1}^N t_i^{\alpha_{ijk}}, j = 0, \dots, M\end{aligned}$$

where $t = (t_1, \dots, t_N)$ is the positive variable vector; $G_j^+, G_j^-, j = 0, \dots, M$ are positive posynomial functions in t ; α_{ijk} are arbitrary real constant exponents; and, finally, c_{jk} are strictly positive coefficients. Sets K_j^+, K_j^- keep track of how many positively or negatively signed monomials form posynomials G_j^+, G_j^- , respectively. This formulation is constructed by grouping together monomials with identical signs.

One can simply apply the transformation $t_i = e^{z_i}, i = 1, \dots, N$ to obtain the transformed optimization problem:

$$\begin{aligned} \min \quad & G_0(z) = G_0^+(z) - G_0^-(z) \\ \text{s.t.} \quad & G_j(z) = G_j^+(z) - G_j^-(z) \leq 0, j = 1, \dots, M \\ & z_i^L \leq z_i \leq z_i^U, i = 1, \dots, N \end{aligned}$$

Where:

$$\begin{aligned} G_j^+(z) &= \sum_{k \in K_j^+} c_{jk} e^{(\sum_{i=1}^N \alpha_{ijk} z_i)}, j = 0, \dots, M \\ G_j^-(z) &= \sum_{k \in K_j^-} c_{jk} e^{(\sum_{i=1}^N \alpha_{ijk} z_i)}, j = 0, \dots, M \end{aligned}$$

Note that this reformulation results in the difference between two convex functions. Thus, the reformulation has, in theory, "convexified" the original nonconvex problem. However, since $z_i^L = \ln t_i^L$, it is necessary that the lower bound t_i^L be strictly positive for this reformulation to exist. Maranas and Floudas help circumvent this issue by essentially pre-scaling the original variables to ensure that their lower bounds will be positive. Thus, make sure that for each t_i :

$$t_i' = t_i + \max(0, -t_i^L + \epsilon), \epsilon > 0.$$

With this pre-scaling in place, it is evident that the reformulated problem is a convex programming problem, and a global solution can be found. A full algorithm and theoretical proofs can be found in the 1997 Maranas and Floudas paper. The reader is directed to the paper for a more rigorous and comprehensive interrogation into the theory behind the problem; this wiki article is simply meant to introduce the curious reader to the topic at large.

Feasibility Analysis

Of course, it is important to determine that the solution to the exponentially transformed convex is feasible to the original geometric program.

It is immediately apparent that if the transformed problem is infeasible, then the original problem must also be infeasible, as the constraints are identical in both programs. Thus, infeasibility in this case means the original constraints are too "tight" or do not allow for a feasible solution anyway, regardless of the exponential transformation.

One common method to determine feasibility between the transformed problem and the original geometric program is to find a point that is, albeit, still infeasible to the original problem, but not far from feasibility. One way to do this might be to set up the following geometric program:

$$\begin{aligned} \min \quad & s \\ \text{s.t.} \quad & f_i(x) \leq s, i = 1, \dots, m \\ & g_i(x) = 1, i = 1, \dots, p \\ & s \geq 1 \end{aligned}$$

Thus, as s nears a value of 1, the original problem nears feasibility. For example, if the optimal $s = 1.1$, then the optimal x is, theoretically, only 10% infeasible for the original problem. Thus, the goal of method is to find a solution such that $s = 1$, and x is feasible to the original geometric program.

Applications

Exponential transformation to convexify an objective function or constraints can be used on any geometric program that meets the criteria discussed above. As noted above, geometric programs can arise in a number of different fields and applications.

Ron Dembo a geometric optimization problem concerning optimal reactor design. His system has eight nonconvex terms, and can be transformed with the exponential transformation method. Thus, chemical systems such as reactors, kinetics, and mass balance equations can be solved with exponential transformations: consider the following law of mass-action for any generic reaction:

$$\text{rate} = kC_A^{0.5}C_B^{0.44}$$

This equation is nonconvex as it stands, but makes a good candidate for exponential transformation.

Posynomial expressions also develop in the field of electrical engineering. In 2001, Hershenson et al formulated a geometric optimization problem for the optimal construction, layout, and connections within CMOS operational amplifiers. Key trade-offs they looked to optimize included power dissipation, unity-gain bandwidth, and open-loop gain. An example of a nonconvex posynomial constraint that arose in their analysis involves the voltage constraints upon a sample transistor in the network, for example:

$$\sqrt{\frac{I_7 L_7}{\mu_p C_{ox} / 2 W_7}} \leq V_{dd} - V_{out, max}$$

The square root of the variables (current, inductance) results in a nonconvex term that can be transformed with the exponential transformation. In the same vein, many other problems exist in electrical engineering that can also benefit from the exponential transformation.

Additional applications can be in the field of finance, supply chain, and project management. See the tutorial under "External Links and References" for an excellent resource recapping the theory of this article on geometric programming as well as a plethora of other applications and examples.

Examples under the External Links and References section include applications concerning:

- Power control in communications systems
- Optimal doping profile in semiconductor device engineering
- Floor planning for configuration of potentially furniture, process units, etc. on a floor
- Digital circuit gate sizing
- The optimal design of a mechanical truss system
- Wire segment sizing within an integrated circuit

An illustrative example involving digital circuit optimization via geometric programming is given by Boyd, Kim, Patil, and Horowitz in the references section.

Examples

Convexification of a few objective functions and constraints are provided in this section as illustrative examples. The resulting reformulations can be tested in GAMS using appropriate solvers to verify that there is indeed a globally optimal solution.

Example 1

Original problem:

$$\begin{aligned} \min \quad & f(t_1, t_2) = 14t_1^{3.2}t_2^4 + t_1^2t_2^7 \\ \text{s.t.} \quad & t_1 \geq 0 \\ & t_2 \geq 0 \end{aligned}$$

After exponential transformation, this problem becomes:

$$\begin{aligned} \min \quad & f(z_1, z_2) = 14e^{(3.2z_1+4z_2)} + e^{(2z_1+7z_2)} \\ \text{s.t.} \quad & e^{(z_1)} \geq 0 \\ & e^{(z_2)} \geq 0 \end{aligned}$$

Example 2

Original problem:

$$\min \quad f(t_1, t_2) = t_1^{3/5}t_2^2$$

$$s.t. \quad \frac{t_1}{t_2} \leq t_1^{0.7}$$

becomes:

$$\begin{aligned} \min \quad & f(z_1, z_2) = e^{(0.6z_1 + 2z_2)} \\ s.t. \quad & z_1 - z_2 \leq 0.7z_1 \end{aligned}$$

Note that in this case, the reformulated constraint is linear and, thus, convex.

There are a multitude of cases where this reformulation can be applied. As long as the coefficients of the original posynomial are strictly positive.

References

1. R.J. Duffin, E.L. Peterson, C. Zener, *Geometric Programming*, John Wiley and Sons, 1967.
2. T.F. Edgar, D.M. Himmelblau, L.S. Lasdon, *Optimization of Chemical Processes*, McGraw-Hill, 2001.
3. J. Nocedal, S.J. Wright, *Numerical Optimization*, Springer, 2006.
4. J.F. Tsai, M.H. Lin, *An Efficient Global Approach for Posynomial Geometric Programming Problems*, INFORMS Journal on Computing, 23 (3) pp. 483-492, 2011.
5. C.D. Maranas, C.A. Floudas, *Global Optimization in Generalized Geometric Programming*, Computers & Chemical Engineering, 21 (4) pp. 351-369, 1997.
6. R. Dembo, *A set of geometric programming test problems and their solutions*, Mathematical Programming, 10 (1) pp. 192-213, 1976.
7. M.D.M. Hershenson, S.P. Boyd, *Optimal Design of a CMOS Op-Amp via Geometric Programming*, IEEE Transactions of Computer-Aided Design of Integrated Circuits and Systems, 20 (1) pp.1-21, 2001.
8. B.M. Worrall, M.A. Hall, *The analysis of an inventory control model using psynomial geometric programming*, International Journal of Production Research 20 (5) pp. 657-667, 1982.
9. S. Boyd, S. J. Kim, L. Vandenberghe, and A. Hassibi, A Tutorial on Geometric Programming
10. S. Boyd, S. J. Kim, D. Patil, and M. Horowitz Digital Circuit Optimization via Geometric Programming

Retrieved from "https://optimization.mccormick.northwestern.edu/index.php?title=Exponential_transformation&oldid=1002"

-
- This page was last modified on 8 June 2014, at 21:36.
 - This page has been accessed 50,737 times.