Geometric Programming and its Application in Network Resource Allocation

Presented by: Bin Wang

Why this talk?

- Nonlinear and nonconvex problem, can be turned into nonlinear convex problem
  - Global optimal, zero dual gap,
  - Numerical efficiency: interior point (polynomial)
- Wide range of application
  - Circuit design
  - Signal processing and information theory
  - Network resource allocation
- Not just using software package and applying some algorithms, it involves skills in modeling and approximation

Outlines

- Basic Geometric Programming
- Extended Geometric Programming
- References

Outlines

- Basic Geometric Programming
  - What is GP?
  - Example (Power Control, rate allocation)
- Extended Geometric Programming
- References

What is GP?

Monomial is a function $f : \mathbf{R}^n_+ \to \mathbf{R}$:

$$f(x) = dx_1^{a^{(1)}} x_2^{a^{(2)}} \ldots x_n^{a^{(n)}}$$

Multiplicative constant $d \geq 0$

Exponential constants $a^{(j)} \in \mathbf{R}, j = 1, 2, \ldots, n$

Posynomial: A sum of monomials:

$$f(x) = \sum_{k=1}^{K} d_k x_1^{a_k^{(1)}} x_2^{a_k^{(2)}} \ldots x_n^{a_k^{(n)}}.$$

where $d_k \geq 0,\ k = 1, 2, \ldots, K$, and $a_k^{(j)} \in \mathbf{R},\ j = 1, 2, \ldots, n, k = 1, 2, \ldots, K$

Example: $\sqrt{2}x^{-0.5}y^{\pi}z$ is a monomial, $x - y$ is not a posynomial

What is GP?

- GP standard form in variables x:

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 1, \quad i = 1, 2, \ldots, m, \\ & h_l(x) = 1, \quad l = 1, 2, \ldots, M \end{array}$$

where $f_i, i = 0, 1, \ldots, m$ are posynomials and $h_l, l = 1, 2, \ldots, M$ are monomials

  - Objective must be the minimization of posynomial
  - Equality constraints can only have the form of a monomial equal one
  - inequality constraints can only have the form of a posynomial less than or equal to one
- Nonlinear and nonconvex

What is GP?

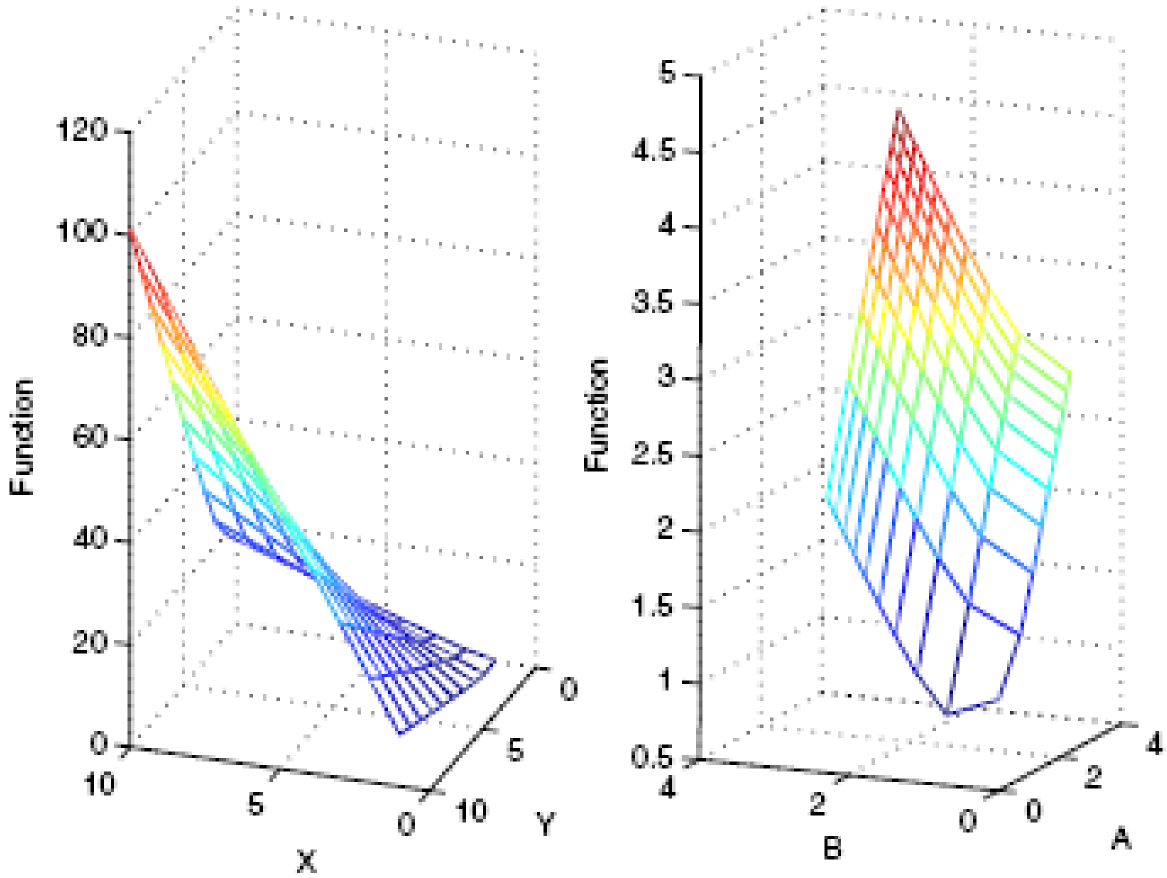
- The trick to solve GP problem is to use original variables' logarithms: $y_i = \log x_i$ (so $x_i = e^{y_i}$) , this results in the standard problem now becomes:

$$\begin{array}{ll} \text{minimize} & \log f_0(e^y) \\ \text{subject to} & \log f_i(e^y) \le 0, \quad i = 1, \ldots, m, \\ & \log g_i(e^y) = 0, \quad i = 1, \ldots, p, \end{array}$$

- GP convex form

$$\begin{array}{ll} \text{minimize} & p_0(y) = \log \sum_{k=1}^{K_0} \exp(a_{0k}^T y + b_{0k}) \\ \text{subject to} & p_i(y) = \log \sum_{k=1}^{K_i} \exp(a_{ik}^T y + b_{ik}) \le 0, \;\; i = 1, 2, \ldots, m, \\ & q_l(y) = a_l^T y + b_l = 0, \;\; l = 1, 2, \ldots, M \end{array}$$

Example


Function
120
100
80
60
40
20
0
10
5
0
X
0
5
10
Y
Function
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
4
2
0
B
0
2
4
A

GP example- power control

- CDMA based ad-hoc network, it has *n* transmitters, labeled 1*, . . . , n*, which transmit at (positive) power levels *P*1*, . . . , Pn*, which are the variables in our power control problem. We also have *n* receivers, labeled 1*, . . . , n*; The power received from transmitter *j*, at receiver *i*, is given by *GijPj .* The *signal power* at receiver *i* is *GiiPi* , and the *interference power* at receiver *i* is $\sum_{k \neq i} G_{ik} P_k$. The *noise power* at receiver *i* is given by σ_i . The *signal to interference and noise ratio* (SINR) of the *i*th receiver/transmitter pair is given by $S_i = \frac{G_{ii} P_i}{\sigma_i + \sum_{k \neq i} G_{ik} P_k}$, we require that

$$\frac{G_{ii} P_i}{\sigma_i + \sum_{k \neq i} G_{ik} P_k} \geq S^{\min}, \quad i = 1, \ldots, n.$$

We also impose limits on the transmitter powers: $P_i \leq P_i^{\max}$

GP application 1: power control

- The problem of minimizing the total transmitter power, subject to these constraints, can be expressed as

$$\begin{array}{ll} \text{minimize} & P_1 + \cdots + P_n \\ \text{subject to} & P_i^{\min} \leq P_i \leq P_i^{\max}, \quad i = 1, \ldots, n, \\ & G_{ii} P_i / (\sigma_i + \sum_{k \neq i} G_{ik} P_k) \geq S^{\min}, \quad i = 1, \ldots, n. \end{array}$$

- This is not a GP, but is easily cast as a GP, by taking the inverse of the SINR constraints:

$$\frac{\sigma_i + \sum_{k \neq i} G_{ik} P_k}{G_{ii} P_i} \leq 1/S^{\min}, \quad i = 1, \ldots, n.$$

GP application 2: Rate allocation

- Consider the same CDMA based ad-hoc network, system objective is to achieve the maximization of the proportional fairness rate allocation $\max \sum_{i=1}^{N} \log(R_i)$:

$$\max \sum_{i=1}^{N} \log(R_i)$$
$$\text{s.t. } \frac{W P_i G_{ii}}{R_i \left(\sum_{j \neq i} P_j G_{ji} + \eta \right)} \geq S_i,$$
$$0 \leq P_i \leq P_{\max}, \; i = 1, 2, \ldots, N.$$

-
- Since $\sum_{i=1}^{N} \log(R_i) \; = \log(\prod_{i=1}^{N} R_i) \sim \prod_{i=1}^{N} R_i$
- And maximize $\prod_{i=1}^{N} R_i$ is equal to minimize $\prod_{i=1}^{N} R_i^{-1}$

Outlines

- Basic Geometric Programming

- Extended Geometric Programming
  - Extended Geometric Programming
  - GP Approximation

- References

Extensions

- Simple transformation
  – the inverse of the SINR constraints
  – Maximized objectives to minimization problem
- Generalized GP that allows compositions of posynomials with other functions
  – Use techniques: introduce new variable
- GP formulations based on monomial approximations of nonlinear functions.
  – Objective or constraints can not be transformed into GP problems

Monomial Approximation

- Example: $$\min f_0(x)$$ $$\text{s.t. } f_1(x) \leq 1$$

$$f_1(x) = \frac{f(\mathbf{x})}{g(\mathbf{x})} = \frac{\sum_{j \neq i} G_{ji} x_j + \eta}{\sum_j G_{ji} x_j + \eta}$$

- If we can approximate the denominator $g(\mathbf{x})$ with a monomial $\tilde{g}(\mathbf{x})$, but leaving the numerator unchanged

- Let $g(\mathbf{x}) = \sum_i u_i(\mathbf{x})$, then we use $g(\mathbf{x}) \geq \tilde{g}(\mathbf{x}) = \prod_i \left(\frac{u_i(\mathbf{x})}{\alpha_i} \right)^{\alpha_i}$

  here $\alpha_i = u_i(\mathbf{x}_0)/g(\mathbf{x}_0)$ for any fixed positive $\mathbf{x}_0$, $\tilde{g}(\mathbf{x}_0) = g(\mathbf{x}_0)$

  $\tilde{g}(\mathbf{x}_0)$ is the best local monomial approximation to $g(\mathbf{x}_0)$ near $\mathbf{x}_0$

Monomial approximation

- Asking for a monomial approximation of *f (x)* near *x* corresponds to ask for an affine approximation of $F(y) = \log f(e^y)$ near $y = \log x$.
- Such an approximation is provided by the first order Taylor approximation of F:

$$F(z) \approx F(y) + \sum_{i=1}^{n} \frac{\partial F}{\partial y_i}(z_i - y_i) \qquad (*)$$

which is valid for $z \approx y$

$$\frac{\partial F}{\partial y_i} = \frac{1}{f(e^y)} \frac{\partial f}{\partial x_i} e^{y_i} = \frac{x_i}{f(x)} \frac{\partial f}{\partial x_i},$$

- Using this formula, and taking the exponential of (*), we have:

$$f(e^z) \approx f(x) \prod_{i=1}^{n} \exp\left(\frac{x_i}{f(x)} \frac{\partial f}{\partial x_i}(z_i - y_i) \right).$$

Defining $w_i = e^{z_i}$

$$f(w) \approx f(x) \prod_{i=1}^{n} \left(\frac{w_i}{x_i} \right)^{a_i}, \qquad a_i = \frac{x_i}{f(x)} \frac{\partial f}{\partial x_i}$$

Iterative Approximation

- Iterative approximation:

Step 0: Choose an initial feasible point $\mathrm{x}^{(0)}$ and set $k = 1$.
Step 1: Approximate $g_i(x)$ with $\tilde{g}_i(\mathrm{x})$ around the previous point $\mathrm{x}^{(k-1)}$
Step 2: Solve the approximated problem and obtain solution $\mathrm{x}^{(k)}$.
Step 3: Increase k by 1 and go back to Step 2 until the solution converges.

Monomial approximation

- Any heuristics?
- How about the performance?
  - With the guarantee of KKT condition, monomial approximation can get over 90% global optimal, with the left results to be local optimal.

References

- S.Boyd, L.Vandenberghe, Convex Optimization, Cambridge University Press (2004)
- M. Chiang, "Geometric Programming for Communication Systems," *Foundations and Trends in Communications and Information Theory,* vol. 2, no. 1, pp. 1-156, Aug. 2005.
- M. Chiang, C. W. Tan, D.Palomar, D. O'Neill, and D. Julian "Geometric programming for power control" *IEEE Trans. Wireless Communications*, 2006.
- R.Duffin, E.Peterson, C.Zener, Geometric Programming: Theory and Application, Wiley, New York (1967)