

## Homework 8

1. *Conformal mapping via convex optimization.* Suppose that  $\Omega$  is a closed bounded region in  $\mathbf{C}$  with no holes (*i.e.*, it is simply connected). The Riemann mapping theorem states that there exists a conformal mapping  $\varphi$  from  $\Omega$  onto  $D = \{z \in \mathbf{C} \mid |z| \leq 1\}$ , the unit disk in the complex plane. (This means that  $\varphi$  is an analytic function, and maps  $\Omega$  one-to-one onto  $D$ .)

One proof of the Riemann mapping theorem is based on an infinite dimensional optimization problem. We choose a point  $a \in \mathbf{int} \Omega$  (the interior of  $\Omega$ ). Among all analytic functions that map  $\partial\Omega$  (the boundary of  $\Omega$ ) into  $D$ , we choose one that maximizes the magnitude of the derivative at  $a$ . Amazingly, it can be shown that this function is a conformal mapping of  $\Omega$  onto  $D$ .

We can use this theorem to construct an approximate conformal mapping, by sampling the boundary of  $\Omega$ , and by restricting the optimization to a finite-dimensional subspace of analytic functions. Let  $b_1, \dots, b_N$  be a set of points in  $\partial\Omega$  (meant to be a sampling of the boundary). We will search only over polynomials of degree up to  $n$ ,

$$\hat{\varphi}(z) = \alpha_1 z^n + \alpha_2 z^{n-1} + \dots + \alpha_n z + \alpha_{n+1},$$

where  $\alpha_1, \dots, \alpha_{n+1} \in \mathbf{C}$ . With these approximations, we obtain the problem

$$\begin{array}{ll} \text{maximize} & |\hat{\varphi}'(a)| \\ \text{subject to} & |\hat{\varphi}(b_i)| \leq 1, \quad i = 1, \dots, N, \end{array}$$

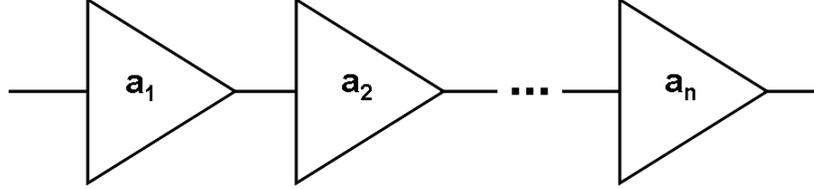
with variables  $\alpha_1, \dots, \alpha_{n+1} \in \mathbf{C}$ . The problem data are  $b_1, \dots, b_N \in \partial\Omega$  and  $a \in \mathbf{int} \Omega$ .

- (a) Explain how to solve the problem above via convex or quasiconvex optimization.
- (b) Carry out your method on the problem instance given in `conf_map_data.m`. This file defines the boundary points  $b_i$  and plots them. It also contains code that will plot  $\hat{\varphi}(b_i)$ , the boundary of the mapped region, once you provide the values of  $\alpha_j$ ; these points should be very close to the boundary of the unit disk. (Please turn in this plot, and give us the values of  $\alpha_j$  that you find.) The function `polyval` may be helpful.

*Remarks.*

- We've been a little informal in our mathematics here, but it won't matter.
- You do not need to know any complex analysis to solve this problem; we've told you everything you need to know.
- A basic result from complex analysis tells us that  $\hat{\varphi}$  is one-to-one if and only if the image of the boundary does not 'loop over' itself. (We mention this just for fun; we're not asking you to verify that the  $\hat{\varphi}$  you find is one-to-one.)

2. *Optimal amplifier gains.* We consider a system of  $n$  amplifiers connected (for simplicity) in a chain, as shown below. The variables that we will optimize over are the gains  $a_1, \dots, a_n > 0$  of the amplifiers. The first specification is that the overall gain of the system, *i.e.*, the product  $a_1 \cdots a_n$ , is equal to  $A$ , which is given.



We are concerned about two effects: noise generated by the amplifiers, and amplifier overload. These effects are modeled as follows.

We first describe how the noise depends on the amplifier gains. Let  $N_i$  denote the noise level (RMS, or root-mean-square) at the output of the  $i$ th amplifier. These are given recursively as

$$N_0 = 0, \quad N_i = a_i \left( N_{i-1}^2 + \alpha_i^2 \right)^{1/2}, \quad i = 1, \dots, n$$

where  $\alpha_i > 0$  (which is given) is the ('input-referred') RMS noise level of the  $i$ th amplifier. The *output noise level*  $N_{\text{out}}$  of the system is given by  $N_{\text{out}} = N_n$ , *i.e.*, the noise level of the last amplifier. Evidently  $N_{\text{out}}$  depends on the gains  $a_1, \dots, a_n$ .

Now we describe the amplifier overload limits.  $S_i$  will denote the (RMS) signal level at the output of the  $i$ th amplifier. These signal levels are related by

$$S_0 = S_{\text{in}}, \quad S_i = a_i S_{i-1}, \quad i = 1, \dots, n,$$

where  $S_{\text{in}} > 0$  is the *input signal level*. Each amplifier has a maximum allowable output level  $M_i > 0$  (which is given). (If this level is exceeded the amplifier will distort the signal.) Thus we have the constraints  $S_i \leq M_i$ , for  $i = 1, \dots, n$ . (A more accurate model would include the noise as well as the signal in the overload condition, but since the signal levels are usually much larger than the noise levels, we can reasonably neglect it here.)

The *maximum output signal level*  $S_{\text{max}}$  is defined as the maximum value of  $S_n$ , over all input signal levels  $S_{\text{in}}$  that respect the the overload constraints  $S_i \leq M_i$ . Of course  $S_{\text{max}} \leq M_n$ , but it can be smaller, depending on the gains  $a_1, \dots, a_n$ .

The *dynamic range*  $D$  of the system is defined as  $D = S_{\text{max}}/N_{\text{out}}$ . Evidently it is a (rather complicated) function of the amplifier gains  $a_1, \dots, a_n$ .

The goal is to choose the gains  $a_i$  to maximize the dynamic range  $D$ , subject to the constraint  $\prod_i a_i = A$ , and upper bounds on the amplifier gains, *i.e.*,  $a_i \leq A_i$  (which are given).

Explain how to solve this problem as a convex (or quasiconvex) optimization problem. If you introduce new variables, or transform the variables, explain. Clearly give the objective and inequality constraint functions, explaining why they are convex if it is not obvious. If your problem involves equality constraints, give them explicitly.

3. *Power flow optimization with ‘ $N - 1$ ’ reliability constraint.* We model a network of power lines as a graph with  $n$  nodes and  $m$  edges. The power flow along line  $j$  is denoted  $p_j$ , which can be positive, which means power flows along the line in the direction of the edge, or negative, which means power flows along the line in the direction opposite the edge. (In other words, edge orientation is only used to determine the direction in which power flow is considered positive.) Each edge can support power flow in either direction, up to a given maximum capacity  $P_j^{\max}$ , *i.e.*, we have  $|p_j| \leq P_j^{\max}$ .

Generators are attached to the first  $k$  nodes. Generator  $i$  provides power  $g_i$  to the network. These must satisfy  $0 \leq g_i \leq G_i^{\max}$ , where  $G_i^{\max}$  is a given maximum power available from generator  $i$ . The power generation costs are  $c_i > 0$ , which are given; the total cost of power generation is  $c^T g$ .

Electrical loads are connected to the nodes  $k + 1, \dots, n$ . We let  $d_i \geq 0$  denote the demand at node  $k + i$ , for  $i = 1, \dots, n - k$ . We will consider these loads as given. In this simple model we will neglect all power losses on lines or at nodes. Therefore, power must balance at each node: the total power flowing into the node must equal the sum of the power flowing out of the node. This power balance constraint can be expressed as

$$Ap = \begin{bmatrix} -g \\ d \end{bmatrix},$$

where  $A \in \mathbf{R}^{n \times m}$  is the node-incidence matrix of the graph, defined by

$$A_{ij} = \begin{cases} +1 & \text{edge } j \text{ enters node } i, \\ -1 & \text{edge } j \text{ leaves node } i, \\ 0 & \text{otherwise.} \end{cases}$$

In the basic power flow optimization problem, we choose the generator powers  $g$  and the line flow powers  $p$  to minimize the total power generation cost, subject to the constraints listed above. The (given) problem data are the incidence matrix  $A$ , line capacities  $P^{\max}$ , demands  $d$ , maximum generator powers  $G^{\max}$ , and generator costs  $c$ .

In this problem we will add a basic (and widely used) reliability constraint, commonly called an ‘ $N - 1$  constraint’. ( $N$  is not a parameter in the problem; ‘ $N - 1$ ’ just means ‘all-but-one’.) This states that the system can still operate even if any one power line goes out, by re-routing the line powers. The case when line  $j$  goes out is called ‘failure contingency  $j$ ’; this corresponds to replacing  $P_j^{\max}$  with 0. The requirement is that there must exist a contingency power flow vector  $p^{(j)}$  that satisfies all the constraints above, with  $p_j^{(j)} = 0$ , using the same given generator powers. (This corresponds to the idea that power flows can be re-routed quickly, but generator power can only be changed more slowly.) The ‘ $N - 1$  reliability constraint’ requires that for each line, there is a contingency power flow vector. The ‘ $N - 1$  reliability constraint’ is (implicitly) a constraint on the generator powers.

The questions below concern the specific instance of this problem with data given in `rel_pwr_flow_data.m`. (Executing this file will also generate a figure showing the network you are optimizing.) Especially for part (b) below, you must explain exactly how you set up the problem as a convex optimization problem.

- (a) *Nominal optimization.* Find the optimal generator and line power flows for this problem instance (without the  $N - 1$  reliability constraint). Report the optimal cost and generator powers. (You do not have to give the power line flows.)
- (b) *Nominal optimization with  $N - 1$  reliability constraint.* Minimize the nominal cost, but you must choose generator powers that meet the  $N - 1$  reliability requirement as well. Report the optimal cost and generator powers. (You do not have to give the nominal power line flows, or any of the contingency flows.)
4. *Optimizing a portfolio of energy sources.* We have  $n$  different energy sources, such as coal-fired plants, several wind farms, and solar farms. Our job is to size each of these, *i.e.*, to choose its capacity. We will denote by  $c_i$  the capacity of plant  $i$ ; these must satisfy  $c_i^{\min} \leq c_i \leq c_i^{\max}$ , where  $c_i^{\min}$  and  $c_i^{\max}$  are given minimum and maximum values.

Each generation source has a cost to build and operate (including fuel, maintenance, government subsidies and taxes) over some time period. We lump these costs together, and assume that the cost is proportional to  $c_i$ , with (given) coefficient  $b_i$ . Thus, the total cost to build and operate the energy sources is  $b^T c$  (in, say, \$/hour).

Each generation source is characterized by an availability  $a_i$ , which is a random variable with values in  $[0, 1]$ . If source  $i$  has capacity  $c_i$ , then the power available from the plant is  $c_i a_i$ ; the total power available from the portfolio of energy sources is  $c^T a$ , which is a random variable. A coal fired plant has  $a_i = 1$  almost always, with  $a_i < 1$  when one of its units is down for maintenance. A wind farm, in contrast, is characterized by strong fluctuations in availability with  $a_i = 1$  meaning a strong wind is blowing, and  $a_i = 0$  meaning no wind is blowing. A solar farm has  $a_i = 1$  only during peak sun hours, with no cloud cover; at other times (such as night) we have  $a_i = 0$ .

Energy demand  $d \in \mathbf{R}_+$  is also modeled as a random variable. The components of  $a$  (the availabilities) and  $d$  (the demand) are *not* independent. Whenever the total power available falls short of the demand, the additional needed power is generated by (expensive) peaking power plants at a fixed positive price  $p$ . The average cost of energy produced by the peakers is

$$\mathbf{E} p(d - c^T a)_+,$$

where  $x_+ = \max\{0, x\}$ . This average cost has the same units as the cost  $b^T c$  to build and operate the plants.

The objective is to choose  $c$  to minimize the overall cost

$$C = b^T c + \mathbf{E} p(d - c^T a)_+.$$

**Sample average approximation.** To solve this problem, we will minimize a cost function based on a sample average of peaker cost,

$$C^{\text{sa}} = b^T c + \frac{1}{N} \sum_{j=1}^N p(d^{(j)} - c^T a^{(j)})_+$$

where  $(a^{(j)}, d^{(j)})$ ,  $j = 1, \dots, N$ , are (given) samples from the joint distribution of  $a$  and  $d$ . (These might be obtained from historical data, weather and demand forecasting, and so on.)

**Validation.** After finding an optimal value of  $c$ , based on the set of samples, you should double check or validate your choice of  $c$  by evaluating the overall cost on another set of (validation) samples,  $(\tilde{a}^{(j)}, \tilde{d}^{(j)})$ ,  $j = 1, \dots, N^{\text{val}}$ ,

$$C^{\text{val}} = b^T c + \frac{1}{N^{\text{val}}} \sum_{j=1}^{N^{\text{val}}} p(\tilde{d}^{(j)} - c^T \tilde{a}^{(j)})_+.$$

(These could be another set of historical data, held back for validation purposes.) If  $C^{\text{sa}} \approx C^{\text{val}}$ , our confidence that each of them is approximately the optimal value of  $C$  is increased.

Finally we get to the problem. Get the data in `energy_portfolio_data.m`, which includes the required problem data, and the samples, which are given as a  $1 \times N$  row vector  $\mathbf{d}$  for the scalars  $d^{(j)}$ , and an  $n \times N$  matrix  $\mathbf{A}$  for  $a^{(j)}$ . A second set of samples is given for validation, with the names `d_val` and `A_val`.

Carry out the optimization described above. Give the optimal cost obtained,  $C^{\text{sa}}$ , and compare to the cost evaluated using the validation data set,  $C^{\text{val}}$ .

Compare your solution with the following naive ('certainty-equivalent') approach: Replace  $a$  and  $d$  with their (sample) means, and then solve the resulting optimization problem. Give the optimal cost obtained,  $C^{\text{ce}}$  (using the average values of  $a$  and  $d$ ). Is this a lower bound on the optimal value of the original problem? Now evaluate the cost for these capacities on the validation set,  $C^{\text{ce, val}}$ . Make a brief statement.

MIT OpenCourseWare  
<http://ocw.mit.edu>

6.079 / 6.975 Introduction to Convex Optimization  
Fall 2009

For information about citing these materials or our Terms of Use, visit: <http://ocw.mit.edu/terms>.