Computational Method of Optimization

Convex optimization problems - Lecture VI

Convex optimization problems

- Agenda
- (Generalized) Linear-Fractional Program
- quadratic optimization
- geometric programming

Source: Convex Optimization — Boyd & Vandenberghe

Linear-fractional program

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & Gx \preceq h \\ & Ax = b \end{array}$$

linear-fractional program

$$f_0(x) = \frac{c^T x + d}{e^T x + f}, \qquad \mathbf{dom}\, f_0(x) = \{x \mid e^T x + f > 0\}$$

- a quasiconvex optimization problem; can be solved by bisection
- also equivalent to the LP (variables y, z)

$$\begin{array}{ll} \text{minimize} & c^T y + dz \\ \text{subject to} & Gy \preceq hz \\ & Ay = bz \\ & e^T y + fz = 1 \\ & z \geq 0 \end{array}$$

It's a generalized linear fractional problem. You minimize a linear fractional function here. You have to specify the domain to linear inequalities and linear equality constraints. Now that's a quasi-convex optimization problem.

You can solve it by bisection. $f_0(x)$ <t. So you're really asking is the optimal value of this problem less than equal or t. Can it even be achieved, the objective value t?

$c^T x + d \leq t(e^T x + f)$
Now that's a linear inequality, and therefore you'd call it an LP feasibility problem. To check if the optimal value – if there's an x that satisfies all this and has $f_0(x)$ <t, you would solve an LP feasibility problem.

And then you could bisect on t and solve this. So that's one way to solve this. It turns out though, some quasi-convex problems can actually be transformed to convex ones

Linear-fractional program

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & Gx \preceq h \\ & Ax = b \end{array}$$

linear-fractional program

$$f_0(x) = \frac{c^T x + d}{e^T x + f}, \qquad \textbf{dom}\, f_0(x) = \{x \mid e^T x + f > 0\}$$

- a quasiconvex optimization problem; can be solved by bisection
- also equivalent to the LP (variables y, z)

$$\begin{array}{ll} \text{minimize} & c^T y + dz \\ \text{subject to} & Gy \preceq hz \\ & Ay = bz \\ & e^T y + fz = 1 \\ & z \geq 0 \end{array}$$

Let me just start with the original problem up here, and I introduce a new variable called z scalar.

minimize $\frac{c^T x + dz}{e^T x + fz}$

s.t. $Gx \leq hz$

$Ax = bz$

// $z=1$

$e^T x + fz = 1$

$z > 0$

You make the problem homogeneous by introducing a new variable.

it’s completely homogeneous. In other words, scaling x and z up here by a positive number has no effect whatsoever on the inequalities, has no effect on the objective. That means in fact I can normalize – the current normalization just equals one, but we can normalize it any way we want and simply divide out later.

Linear-fractional program

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & Gx \preceq h \\ & Ax = b \end{array}$$

linear-fractional program

$$f_0(x) = \frac{c^T x + d}{e^T x + f}, \qquad \textbf{dom}\, f_0(x) = \{x \mid e^T x + f > 0\}$$

- a quasiconvex optimization problem; can be solved by bisection
- also equivalent to the LP (variables y, z)

$$\begin{array}{ll} \text{minimize} & c^T y + dz \\ \text{subject to} & Gy \preceq hz \\ & Ay = bz \\ & e^T y + fz = 1 \\ & z \geq 0 \end{array}$$

minimize $\frac{c^T x + dz}{e^T x + fz}$

s.t. $Gx \le hz$

$Ax = bz$

// $z=1$

$e^T x + fz = 1$

$z > 0$

This is now an LP here. And in fact, if you replace this with this, you get all sorts of information out of this. For example, if you were to solve this and z comes out positive, then by simply renormalizing here, you get this solution.

If z turns out to be zero, that's like this one case where this thing is unbounded below. By the way, for most quasi-convex problems this trick won't work, but for some it does.

Generalized linear-fractional program

$$f_0(x) = \max_{i=1,\ldots,r} \frac{c_i^T x + d_i}{e_i^T x + f_i}, \qquad \mathbf{dom}\, f_0(x) = \{x \mid e_i^T x + f_i > 0,\ i = 1,\ldots,r\}$$

a quasiconvex optimization problem; can be solved by bisection

example: Von Neumann model of a growing economy

$$\begin{array}{ll} \text{maximize (over } x,\ x^+) & \min_{i=1,\ldots,n} x_i^+/x_i \\ \text{subject to} & x^+ \succeq 0, \quad Bx^+ \preceq Ax \end{array}$$

- $x, x^+ \in \mathbf{R}^n$: activity levels of n sectors, in current and next period
- $(Ax)_i$, $(Bx^+)_i$: produced, resp. consumed, amounts of good i
- x_i^+/x_i: growth rate of sector i

allocate activity to maximize growth rate of slowest growing sector

Generalized linear fractional problem. You minimize the maximum of a bunch of linear fractional functions. And that's on the polyhedron where all the linear – well, it's at the intersection of the domains of the linear fractional functions. That's quasi-convex. There's no way to convert that to an LP in general, so that you're going to solve by bisection.

Here's an example. It's Von Neumann growth Model. We're going to optimize over x and x^+ the following. So, x be a set of activity levels. we have to have x^+ here too, or something like that x is a bunch of activity levels at one period, and x^+ are the activity levels in an economy at the next period, so that's what these are. Then we have matrices Ax this gives you the amount produced by an activity level x.

Generalized linear-fractional program

$$f_0(x) = \max_{i=1,\ldots,r} \frac{c_i^T x + d_i}{e_i^T x + f_i}, \qquad \mathbf{dom}\, f_0(x) = \{x \mid e_i^T x + f_i > 0,\ i = 1, \ldots, r\}$$

a quasiconvex optimization problem; can be solved by bisection

example: Von Neumann model of a growing economy

$$\begin{array}{ll} \text{maximize (over } x,\ x^+) & \min_{i=1,\ldots,n} x_i^+/x_i \\ \text{subject to} & x^+ \succeq 0, \quad Bx^+ \preceq Ax \end{array}$$

- $x, x^+ \in \mathbf{R}^n$: activity levels of n sectors, in current and next period
- $(Ax)_i$, $(Bx^+)_i$: produced, resp. consumed, amounts of good i
- x_i^+/x_i: growth rate of sector i

allocate activity to maximize growth rate of slowest growing sector

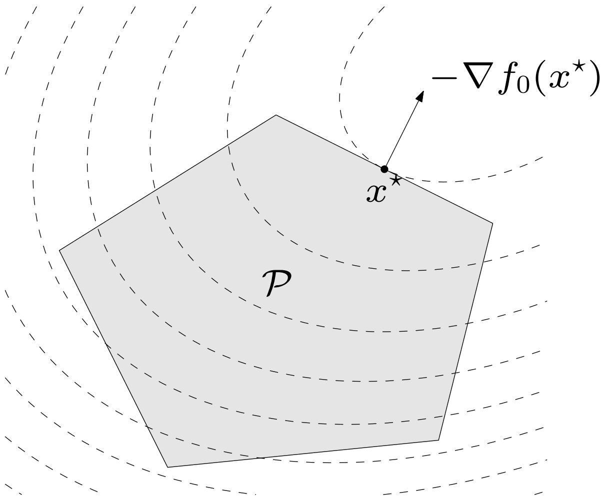
For example, in our m, if the n activity levels produce say m different classes of goods. So if you have an activity level x, Ax is a vector telling you how much goods – the vector of goods produced by that activity level here. That's Ax. Now when you operate the economy at the next step, that's x^+ – are the activity levels. Bx^+ – that's a vector telling you how much is consumed. And then you simply say that the amount consumed at the next stage here, it uses some – it has to use the goods produced in the first stage, so you'd have an inequality that looks like that.

Maximize over these the minimum activity growth level. That's x^+ over x. x^+ is implicit here. To maximize this is the same as minimizing the negative of this, which is minimizing the max of x over x^+ , each of those, so that's the same thing. That's a quasi-convex optimization problem, and you can solve it directly, so that's one.

Quadratic program (QP)

$$\begin{array}{ll} \text{minimize} & (1/2)x^T P x + q^T x + r \\ \text{subject to} & Gx \preceq h \\ & Ax = b \end{array}$$

- $P \in \mathbf{S}^n_+$, so objective is convex quadratic
- minimize a convex quadratic function over a polyhedron



In a quadratic program, you have a convex quadratic objective. So, P is positive semi-definite, and you minimize this over a polyhedron, so a set of linear inequalities and a set of linear equalities. By the way, if you just said P equals zero, you recover an LP, so this is a strict extension of LP.

And the picture now is this. Here's polyhedron which is the feasible set shaded, and objective function now instead of being affine or linear is quadratic and convex quadratic. So, the sublevel sets are ellipsoids, and so this is showing you the sublevel sets of a quadratic function. The gradient points down the slope, and is the normal to the tangent hyperplane at the level surface. So, for this problem instance, that's the solution there.

Examples

least-squares

$$\text{minimize} \quad \|Ax - b\|_2^2$$

- analytical solution $x^\star = A^\dagger b$ ($A^\dagger$ is pseudo-inverse)
- can add linear constraints, *e.g.*, $l \preceq x \preceq u$

linear program with random cost

$$\begin{array}{ll} \text{minimize} & \bar{c}^T x + \gamma x^T \Sigma x = \mathbf{E}\, c^T x + \gamma\, \mathbf{var}(c^T x) \\ \text{subject to} & Gx \preceq h, \quad Ax = b \end{array}$$

- c is random vector with mean $\bar{c}$ and covariance Σ
- hence, $c^T x$ is random variable with mean $\bar{c}^T x$ and variance $x^T \Sigma x$
- $\gamma > 0$ is risk aversion parameter; controls the trade-off between expected cost and variance (risk)

let's look at some examples. The most basic one would be just least squares. There's no constraints. The solution is just given by by this pseudo-inverse or Moore-Penrose inverse. That's just A dagger B.
So that's a quadratic program. For example a least squares problem with bounds on the x's, that can be solved basically as fast and as reliably as just a least squares problem with a very small modest factor, totally reliable.

Examples

least-squares

$$\text{minimize} \quad \|Ax - b\|_2^2$$

- analytical solution $x^\star = A^\dagger b$ ($A^\dagger$ is pseudo-inverse)
- can add linear constraints, *e.g.*, $l \preceq x \preceq u$

linear program with random cost

$$\begin{array}{ll} \text{minimize} & \bar{c}^T x + \gamma x^T \Sigma x = \mathbf{E}\, c^T x + \gamma\, \mathbf{var}(c^T x) \\ \text{subject to} & Gx \preceq h, \quad Ax = b \end{array}$$

- c is random vector with mean $\bar{c}$ and covariance Σ
- hence, $c^T x$ is random variable with mean $\bar{c}^T x$ and variance $x^T \Sigma x$
- $\gamma > 0$ is risk aversion parameter; controls the trade-off between expected cost and variance (risk)

Let's do a specific example-diet problem. So you are – x represents the amounts of a bunch of foodstuffs you're going to buy. And there are constraints on it. There are things that have to balance out exactly, and there's inequalities. Of course, this can include lower bounds and upper bounds on for example nutrient levels.

So our constraints become a polyhedron in x here. So it's not as simple as saying just minimize cTx which would give you the cheapest diet that would meet the requirements because you don't know c. c for example has a mean, $\bar{c}$, but it also has a covariance sigma. There are many things you could do. In practice, you would simply ignore the variation and just work with the expected values.

Quadratically constrained quadratic program (QCQP)

$$\begin{array}{ll} \text{minimize} & (1/2)x^T P_0 x + q_0^T x + r_0 \\ \text{subject to} & (1/2)x^T P_i x + q_i^T x + r_i \le 0, \quad i = 1, \dots, m \\ & Ax = b \end{array}$$

- $P_i \in \mathbf{S}^n_+$; objective and constraints are convex quadratic
- if $P_1, \dots, P_m \in \mathbf{S}^n_{++}$, feasible region is intersection of m ellipsoids and an affine set

Now another generalization is a QCQP, that's a quadratically constrained quadratic program. So here the constraints can also be convex quadratics. Now here, if these (constraints)are zero, you recover a QP. And of course, if all the Ps are zero, you recover a general LP.

Here, if P is positive definite, this represents an ellipsoid, so here the feasible set is an intersection of ellipsoids and degenerate ellipsoids.
Now, a degenerate ellipsoid is one that can have a flat face and be infinite. So a half space is degenerate ellipsoid, for example.

But for example, if P is positive semi-definite but only has Rank 3, this is sort of a degenerate ellipsoid. It means that in N minus three dimensions, it's sort of infinite. In three dimensions, it's got curvature.

For example, it would be a cylinder in R^3 for example. A cylinder in R^3 would be a degenerate ellipsoid. That would be degenerated by a Rank 2 – a 3x3 matrix P that's Rank 2, and this inequality would then be not a cylinder, but it would be an ellipsoid like this, and then it would have an infinite extent in another direction. So that's it.
It's a convex problem and can do this.

Second-order cone programming (SOCP)

$$\begin{array}{ll} \text{minimize} & f^T x \\ \text{subject to} & \|A_i x + b_i\|_2 \le c_i^T x + d_i, \quad i = 1, \ldots, m \\ & Fx = g \end{array}$$

$(A_i \in \mathbf{R}^{n_i \times n}, \; F \in \mathbf{R}^{p \times n})$

- inequalities are called second-order cone (SOC) constraints:

$$(A_i x + b_i, c_i^T x + d_i) \in \text{second-order cone in } \mathbf{R}^{n_i+1}$$

- for $n_i = 0$, reduces to an LP; if $c_i = 0$, reduces to a QCQP
- more general than QCQP and LP

A second order cone program – this is SOCP. You minimize the linear function subject to the constraints. It's a norm of an affine thing is less than or equal to an affine thing.
Here, it is not norm squared. If this was norm squared, this problem would be a QCQP. It is not norm squared. This sort of the square root of a convex quadratic.

it's called second-order cone programming because each constraint is nothing but this. You have an affine mapping that maps x into $A_i x+b$, that's a vector, then a scalar – $c^T x+d_i$ – that should be in the unit second-order cone, the Lorentz-cone in R^{n+1}
you can write as an SOCP in the standard form
$\|A_i x+b\|-c^T x-d_i \le 0$

Robust linear programming

the parameters in optimization problems are often uncertain, *e.g.*, in an LP

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & a_i^T x \leq b_i, \quad i = 1, \ldots, m, \end{array}$$

there can be uncertainty in c, a_i, b_i

two common approaches to handling uncertainty (in a_i, for simplicity)

- deterministic model: constraints must hold for all $a_i \in \mathcal{E}_i$

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & a_i^T x \leq b_i \text{ for all } a_i \in \mathcal{E}_i, \quad i = 1, \ldots, m, \end{array}$$

- stochastic model: a_i is random variable; constraints must hold with probability η

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & \mathbf{prob}(a_i^T x \leq b_i) \geq \eta, \quad i = 1, \ldots, m \end{array}$$

Robust LP looks like this. It says minimize $c^T x$ subject to $a^T x$ is less than b_i. That's for all a in this ellipsoid. This constraint here a semi-infinite Constraint. Because you see this represents a single linear inequality for each element of an ellipsoid, and there's an infinite number points in an ellipsoid

a.

Deterministic approach via SOCP

- choose an ellipsoid as $\mathcal{E}_i$:

$$\mathcal{E}_i = \{\bar{a}_i + P_i u \mid \|u\|_2 \leq 1\} \qquad (\bar{a}_i \in \mathbf{R}^n, \quad P_i \in \mathbf{R}^{n \times n})$$

center is $\bar{a}_i$, semi-axes determined by singular values/vectors of P_i

- robust LP

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & a_i^T x \leq b_i \quad \forall a_i \in \mathcal{E}_i, \quad i = 1, \ldots, m \end{array}$$

is equivalent to the SOCP

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & \bar{a}_i^T x + \|P_i^T x\|_2 \leq b_i, \quad i = 1, \ldots, m \end{array}$$

(follows from $\sup_{\|u\|_2 \leq 1} (\bar{a}_i + P_i u)^T x = \bar{a}_i^T x + \|P_i^T x\|_2$)

So that's why this is a semi-infinite thing. When is it true that $a^T x$ is less than b_i. Let's fix x. You fix b. a varies over this ellipsoid. You check whether $\bar{a}_i^T x + P_i^T x$ – whether that's less than b_i for all $\|u\|_2$ less than one. I'll write it as u^T times P_i^T x like that. Now if I want to maximize this over $\|u\|_2$ less than one, that's a constant. This is an inner product of u with a vector. And so obviously by the Cauchy-Schwarz inequality, the largest that number could be is the norm of P_i^T x .

You can solve SOCPs now basically as fast as LPs, so almost as fast. Certainly, it's the same speed, same reliability, everything. You solve an LP, then you ask the person how well do you know the data. If they don't say perfectly then you should be solving SOCP.

Stochastic approach via SOCP

- assume a_i is Gaussian with mean $\bar{a}_i$, covariance Σ_i ($a_i \sim \mathcal{N}(\bar{a}_i, \Sigma_i)$)
- $a_i^T x$ is Gaussian r.v. with mean $\bar{a}_i^T x$, variance $x^T \Sigma_i x$; hence

$$\mathbf{prob}(a_i^T x \le b_i) = \Phi\left(\frac{b_i - \bar{a}_i^T x}{\|\Sigma_i^{1/2} x\|_2}\right)$$

where $\Phi(x) = (1/\sqrt{2\pi}) \int_{-\infty}^{x} e^{-t^2/2}\, dt$ is CDF of $\mathcal{N}(0,1)$

- robust LP

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & \mathbf{prob}(a_i^T x \le b_i) \ge \eta, \quad i = 1, \ldots, m, \end{array}$$

with $\eta \ge 1/2$, is equivalent to the SOCP

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & \bar{a}_i^T x + \Phi^{-1}(\eta) \|\Sigma_i^{1/2} x\|_2 \le b_i, \quad i = 1, \ldots, m \end{array}$$

So in the stochastic approach, we'll assume that the constraint vector is Gaussian with mean $\bar{a}_i$ and covariance Σ_i . Now when you form a_i^T, that's just a scalar Gaussian random variable. It depends on x of course, but we're fixing x. It's got mean $\bar{a}_i^T x$ and $x^T \Sigma_i x$; And so the probability that you meet your constraint is equal to the CDF. That's a normalized random variable. That's b_i minus a_i – this is the probability. That's the normalized random variable

By the way, these are called chance constraints, and problems that generally involve constraints that involve probabilities of something are – that's a whole field called stochastic programming.

We can write this out analytically as SOCP. It's an SOCP provided the inverse CDF of a Gaussian evaluated at η is positive.

Geometric programming

monomial function

$$f(x) = c x_1^{a_1} x_2^{a_2} \cdots x_n^{a_n}, \qquad \mathbf{dom}\, f = \mathbf{R}_{++}^n$$

with $c > 0$; exponent a_i can be any real number

posynomial function: sum of monomials

$$f(x) = \sum_{k=1}^{K} c_k x_1^{a_{1k}} x_2^{a_{2k}} \cdots x_n^{a_{nk}}, \qquad \mathbf{dom}\, f = \mathbf{R}_{++}^n$$

geometric program (GP)

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

with f_i posynomial, h_i monomial

Here's a GP, a geometric program. You minimize a posynomial subject to some posynomial inequalities less than one, and a bunch of monomials are equal to one.

So, what's a monomial. It's a positive coefficient times a product of variables, each one to a power, and the power can be anything.
It can be an integer, it can be irrational, and it can be negative. It's clear this comes up in a lot of engineering. People call this a scaling law or something like this. They came up with this thing called a posynomial, it's supposed to combine positive and polynomial. but the point is these aren't even polynomials because the exponents here can be like minus 0.1, and they can be 1.1,

For example, $3\sqrt{\frac{x_1}{x_2}}$ That's a monomial and here's a posynomial $2x_1x_2+\sqrt{x_2}$.

Geometric program in convex form

change variables to $y_i = \log x_i$, and take logarithm of cost, constraints

- monomial $f(x) = cx_1^{a_1} \cdots x_n^{a_n}$ transforms to ($y_i = \log x_i$, so $x_i = e^{y_i}$)

$$\log f(e^{y_1}, \ldots, e^{y_n}) = a^T y + b \qquad (b = \log c)$$

- posynomial $f(x) = \sum_{k=1}^K c_k x_1^{a_{1k}} x_2^{a_{2k}} \cdots x_n^{a_{nk}}$ transforms to

$$\log f(e^{y_1}, \ldots, e^{y_n}) = \log \left(\sum_{k=1}^K e^{a_k^T y + b_k} \right) \qquad (b_k = \log c_k)$$

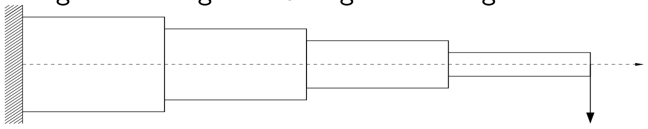
- geometric program transforms to convex problem

$$\begin{array}{ll} \text{minimize} & \log \left(\sum_{k=1}^K \exp(a_{0k}^T y + b_{0k}) \right) \\ \text{subject to} & \log \left(\sum_{k=1}^K \exp(a_{ik}^T y + b_{ik}) \right) \leq 0, \quad i = 1, \ldots, m \\ & Gy + d = 0 \end{array}$$

Let's take a monomial, and let's take the log of this monomial. Well, you get log c. That's b. Then you get plus, and then you get the exponents. You get $a_1$log x_1, but log x_1 is y_1, so when you take – when you change variables by this logarithmic transform, and you take the log of a monomial, you get an affine function.

Now take a posynomial. So here I take the log of a sum of these things, but that's the same as log sum x of an affine function. Because each of the x's I replace with e^{y1} like this. Then I take a sum of those things, and I take the log. That's log sum x. That's this thing. And if you look at this, this function is convex in yY because it's log sum x of an affine function of y, so it is convex.

Design of cantilever beam



- N segments with unit lengths, rectangular cross-sections of size $w_i \times h_i$
- given vertical force F applied at the right end

design problem

minimize	total weight
subject to	upper & lower bounds on w_i, h_i
	upper bound & lower bounds on aspect ratios h_i/w_i
	upper bound on stress in each segment
	upper bound on vertical deflection at the end of the beam

variables: w_i, h_i for $i = 1, \ldots, N$

The question is going be to get the optimal tapering of a beam. That's a cross-section. We're going design the shape of a beam, cantilever beam. So we'll minimize the total weight subject to these have a width and height, so there's a height.

So we've got lower bounds on the width and the height of the segments. We'll limit the aspect ratio so you don't get like an infinitely thin thing like this. We'll upper bound the stress, and we'll upper bound the vertical deflection at the end of the beam, where the maximum deflection occurs.

Objective and constraint functions

- total weight $w_1 h_1 + \cdots + w_N h_N$ is posynomial
- aspect ratio h_i/w_i and inverse aspect ratio w_i/h_i are monomials
- maximum stress in segment i is given by $6iF/(w_i h_i^2)$, a monomial
- the vertical deflection y_i and slope v_i of central axis at the right end of segment i are defined recursively as

$$
\begin{aligned}
v_i &= 12(i-1/2)\frac{F}{Ew_i h_i^3} + v_{i+1} \\
y_i &= 6(i-1/3)\frac{F}{Ew_i h_i^3} + v_{i+1} + y_{i+1}
\end{aligned}
$$

for $i = N, N-1, \ldots, 1$, with $v_{N+1} = y_{N+1} = 0$ (E is Young's modulus)

v_i and y_i are posynomial functions of w, h

This is quite a complicated problem. It's highly nonlinear and so on and so forth. So but let's say it's actually a GP. We look at the total weight. Now, these are variables. What kind of function of w and h is that? it's a posynomial because each of these is a monomial, and it's a sum with positively weighted. Now the aspect ratio and inverse aspect ratio are monomials, so if I set them less than a number, I could divide by it, and I'd get a monomial – well, they're posynomials.

Now the vertical deflection and slope are very – it's quite complicated to work out how much this thing – the tip deflects as a function of the width and height of each of these beam segments. But it's given by a recursion.

Formulation as a GP

$$
\begin{array}{ll}
\text{minimize} & w_1 h_1 + \cdots + w_N h_N \\
\text{subject to} & w_{\max}^{-1} w_i \le 1, \quad w_{\min} w_i^{-1} \le 1, \quad i = 1, \ldots, N \\
& h_{\max}^{-1} h_i \le 1, \quad h_{\min} h_i^{-1} \le 1, \quad i = 1, \ldots, N \\
& S_{\max}^{-1} w_i^{-1} h_i \le 1, \quad S_{\min} w_i h_i^{-1} \le 1, \quad i = 1, \ldots, N \\
& 6iF\sigma_{\max}^{-1} w_i^{-1} h_i^{-2} \le 1, \quad i = 1, \ldots, N \\
& y_{\max}^{-1} y_1 \le 1
\end{array}
$$

note

- we write $w_{\min} \le w_i \le w_{\max}$ and $h_{\min} \le h_i \le h_{\max}$

$$
w_{\min}/w_i \le 1, \qquad w_i/w_{\max} \le 1, \qquad h_{\min}/h_i \le 1, \qquad h_i/h_{\max} \le 1
$$

- we write $S_{\min} \le h_i/w_i \le S_{\max}$ as

$$
S_{\min} w_i/h_i \le 1, \qquad h_i/(w_i S_{\max}) \le 1
$$

Minimizing spectral radius of nonnegative matrix

Perron-Frobenius eigenvalue $\lambda_{\text{pf}}(A)$

- exists for (elementwise) positive $A \in \mathbf{R}^{n\times n}$
- a real, positive eigenvalue of A, equal to spectral radius $\max_i |\lambda_i(A)|$
- determines asymptotic growth (decay) rate of A^k: $A^k \sim \lambda_{\text{pf}}^k$ as $k \to \infty$
- alternative characterization: $\lambda_{\text{pf}}(A) = \inf\{\lambda \mid Av \preceq \lambda v \text{ for some } v \succ 0\}$

minimizing spectral radius of matrix of posynomials

- minimize $\lambda_{\text{pf}}(A(x))$, where the elements $A(x)_{ij}$ are posynomials of x
- equivalent geometric program:

$$\begin{array}{ll} \text{minimize} & \lambda \\ \text{subject to} & \sum_{j=1}^n A(x)_{ij} v_j / (\lambda v_i) \leq 1, \quad i = 1, \ldots, n \end{array}$$

variables λ, v, x

Let's take a matrix that's element-wise positive, so a square matrix element-wise positive. It is called Perron-Frobenius theory. It says that the eigenvalue of A – A's not symmetric, so it can have complex eigenvalues. But it says that the complex value of largest magnitude – that it's spectral radius is that magnitude – is positive
A^k tells you how this dynamical system would propagate, and the asymptotic growth rate, which could be negative by the way, is simply completely determined by this number.

Another way to characterize this is that the Perron-Frobenius eigenvalue of a matrix is it's smallest **λ** for which Av is less than or equal to **λ**. For some positive v, It turns out you never get inequality here. It's always equality here, **λ**'s an eigenvalue like that.

Minimizing spectral radius of nonnegative matrix

Perron-Frobenius eigenvalue $\lambda_{\text{pf}}(A)$

- exists for (elementwise) positive $A \in \mathbf{R}^{n \times n}$
- a real, positive eigenvalue of A, equal to spectral radius $\max_i |\lambda_i(A)|$
- determines asymptotic growth (decay) rate of A^k: $A^k \sim \lambda_{\text{pf}}^k$ as $k \to \infty$
- alternative characterization: $\lambda_{\text{pf}}(A) = \inf\{\lambda \mid Av \preceq \lambda v \text{ for some } v \succ 0\}$

minimizing spectral radius of matrix of posynomials

- minimize $\lambda_{\text{pf}}(A(x))$, where the elements $A(x)_{ij}$ are posynomials of x
- equivalent geometric program:

$$\begin{array}{ll} \text{minimize} & \lambda \\ \text{subject to} & \sum_{j=1}^n A(x)_{ij} v_j / (\lambda v_i) \leq 1, \quad i = 1, \ldots, n \end{array}$$

variables λ, v, x

Minimize the Perron- Frobenius eigenvalue of A where the entries of A are themselves posynomials by making a problem that looks like this. You minimize **λ** subject to A(x) sum – and these are all monomials in the variables, which are x, v, and **λ**, and these are posynomial inequalities. So basically, log sum x comes out of this.