Bridge 3: Optimizing* Design

*An introduction to the branch of Applied Math known as Optimization

Outline

bridge3(W,nos)

-last week's code

-optimization routine (calls work function)

-plot bridge with optimal thicknesses

[val,grad] = work(a, A, L, W)

bridge3(W,nos)

- use last week's code to build adjacency matrix and coordinate system ✓
- then we want to choose fiber thickness such that work is minimized when we apply a load (optimization routine)
- plot the bridge with optimal thicknesses (a-values) returned by fmincon, with work as title. (see btent code for plotting help)

[bridge3]

Unpacking Optimization

This is our objective function.

We solve for a set of a-values (fiber areas)….

$$\min_{a} \underbrace{x^T f}$$

That minimize the amount of work done.

Subject to a set of constraints:

$$A^T K(a) A x = f$$

must obey our equilibrium equation

$$L^T a = V$$

total thickness must meet volume constraint- amount of fiber material we have

$$alo \leq a(j) \leq ahi$$

thickness of each must stay within a given range

fmincon

- The good news: MATLAB has a built-in optimization package with solvers that take care of the minimizing.
- *Your* work is in handing fmincon the right objects.

Constraints

[x, fval] =

fmincon(function,initval,A,B,Aeq,Beq,LB,UB,nonlcon, options)

(objective) function we are minimizing

initial values of function

Linear *in*equalities A*a <= B

Linear *equalities* Aeq*a = Beq

Lower and Upper bounds on a-values

nonlinear constraints

set to handle gradients

[optimal_a, work] =

fmincon(@(a) work(a,A,L,w),a0,[],[],L,V,LB,UB,[], options)

fmincon

[x, fval] =

Constraints

fmincon(function,initval,A,B,Aeq,Beq,LB,UB,nonlcon, options)

(objective) function we are minimizing

initial values of function

Linear *in*equalities A*a <= B

Linear *equalities* Aeq*a = Beq

Lower and Upper bounds on a-values

nonlinear constraints

set to handle gradients

[optimal_a, work] =

fmincon(@(a) work(a,A,L,w),a0,[],[],L,V,LB,UB,[], options)

What are our initial a-values (what did we use in the last project?) all a-vals =1. So ,as an example, a0=ones(numfibers,1)

You computed matrix L in your adjacency automation. What is v? We base the amt of material available on the initial case.

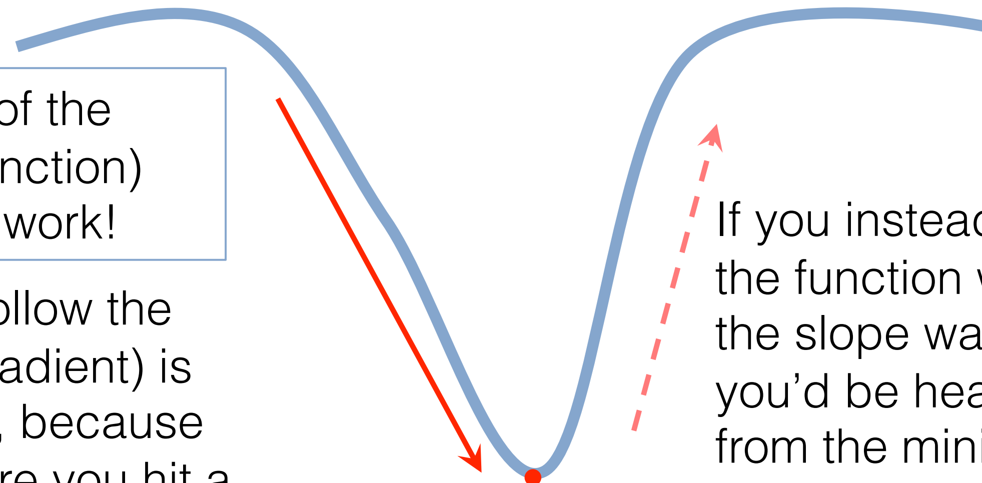
LB=alo= 0.1, UB=ahi=10, but we need them in vector form (for *each* a-val) …we are solving a linear system

options = optimset ('display','iter', 'gradobj','on');

[workfun]

Gradients for optimization

the objective function: work as a function of (a)



We're looking for a minimum (of the objective function, the work function) because we want to minimize work!

To get to the minimum, we follow the function where the slope (gradient) is *negative*. This makes sense, because the function decreases before you hit a minimum.

If you instead followed the function where the slope was positive, you'd be heading *away* from the minimum!

At this *minimum,* the *gradient* is 0.

We are lucky because the gradient of the work has been computed for us in the notes!!

$$\nabla \mathrm{work}(a) = -\begin{pmatrix} e_1^2(a)/L_1 & e_2^2(a)/L_2 & \cdots & e_m^2(a)/L_m \end{pmatrix}^T$$

How can we write this grad more simply using just e and L?

```
function [val,grad] = work(a,A,L,W)
```

- compute work as you would have last week

  … (matrix multiplication operations)

  …

  val = x'*f; (this is your work value)

- Now the gradient: $= -(Ax(a))^T \dfrac{\partial K(a)}{\partial a_1} Ax(a)$

What are these "e"s representing?

$$= -(e_1\ e_2\ \cdots\ e_m) \begin{pmatrix} 1/L_1 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & 0 \end{pmatrix} \begin{pmatrix} e_1 \\ e_2 \\ \vdots \\ e_m \end{pmatrix}$$

$$= -e_1^2(a)/L_1.$$

$$\nabla \text{work}(a) = -\left(e_1^2(a)/L_1 \quad e_2^2(a)/L_2 \quad \cdots \quad e_m^2(a)/L_m\right)^T$$

If you like optimization or find it intuitive, consider taking CAAM 378 (Intro to Optimization), or CAAM 571 (with me!) next semester

(-: