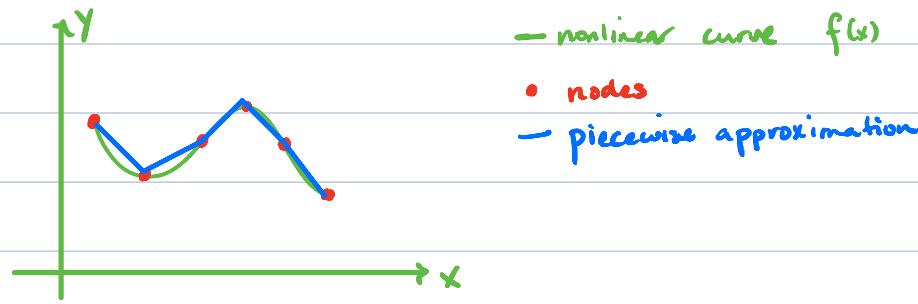
In these notes, we'll talk about a powerful technique for approximating nonlinear optimization problems as MILPs.
The key concept here is one you're already familiar with — piecewise linearization.



If we think about minimizing the green curve, we can see that an algorithm could get stuck at a local minimum.

If we think about minimizing the blue curve, we need to first decide which "piece" is "active" (this is an integer type decision) and then minimize over just that piece (this is a linear programming problem).

We'll never get stuck at a local min, but in general, our answer won't be exact because of the approximation.
We can reduce the error by adding more & more pieces.
The trade-off is that this adds more & more integer variables — so the problem will take longer to solve.

Suppose the nodes along the x-axis are

$$x_1, x_2, \dots, x_n$$

and the y-values of the function at these nodes are

$$y_1, y_2, \dots, y_n$$

The x_i's and y_i's are just data in the problem.
Introduce a binary decision variable $b_i \in \{0,1\}$, $i=1,..,n-1$

$$b_1, b_2, \dots, b_{n-1}$$

These binaries will tell us what interval is active. There are n nodes & $n-1$ intervals — so only $n-1$ binaries.
Only one interval can be active, thus,

$$\sum_{i=1}^{n-1} b_i = 1$$

We will also introduce a weighting for each node.

$$w_1, w_2, \dots, w_n$$

If the first interval is active, we still need to decide where to be between the 1^{st} + 2^{nd} nodes. $w_1 = 1$ means we are choosing x at node 1. $w_2 = 1$ means that we are choosing x at node 2. $w_1 = w_2 = 1/2$ means we are choosing x half-way between nodes 1 + 2.

$$0 \le w_i \le 1$$

$$\sum_{i=1}^{n} w_i = 1$$

The only weights that can be non-zero are the ones on an active interval.

$$\left.\begin{aligned} w_1 &\le b_1 \\ w_2 &\le b_1 + b_2 \\ w_3 &\le b_2 + b_3 \\ &\vdots \\ w_n &\le b_{n-1} \end{aligned}\right\}$$

These are called "special ordered set" (SOS) constraints. For simplicity, we may write $w \in SOS$. Some solvers have a special way of handling these.

Given these weights,

$$x = w_1 x_1 + w_2 x_2 + \dots + w_n x_n$$

$$y = w_1 y_1 + w_2 y_2 + \dots + w_n y_n$$

Thus, the problem

$$\min f(x)$$

becomes

$$\min \sum_{i=1}^{n} w_i y_i \quad \text{subj. to} \quad w \in SOS, \ \sum_{i=1}^{n} w_i = 1.$$

or more explicitly,

$$\begin{aligned}
\min \quad & y \\
\text{s.t.} \quad & y = w_1 y_1 + w_2 y_2 + \dots + w_n y_n \\
& 1 = b_1 + b_2 + \dots + b_{n-1} \\
& 1 = w_1 + w_2 + \dots + w_n \\
& w_1 \leq b_1 \\
& w_2 \leq b_1 + b_2 \\
& \vdots \\
& w_n \leq b_{n-1} \\
& b_i \in \{0,1\} \\
& w_i \in [0,1]
\end{aligned}$$

A single variable, unconstrained, nonlinear optimization problem became a 2n variable, constrained, mixed integer linear program. There are trade-offs!

The nonlinear function could also be in the constraints.
Using the same constructions as above,

$f(x) \le 0$ becomes

$$
\begin{aligned}
& y \le 0 \\
& y = w_1 y_1 + w_2 y_2 + \cdots + w_n y_n \\
& 1 = b_1 + b_2 + \cdots + b_{n-1} \\
& 1 = w_1 + w_2 + \cdots + w_n \\
& w_1 \le b_1 \\
& w_2 \le b_1 + b_2 \\
& \vdots \\
& w_n \le b_{n-1} \\
& b_i \in \{0,1\} \\
& w_i \in [0,1]
\end{aligned}
$$