AMPL Problem for Homework 5

A few years ago, Elon Musk (the Space-X dude) proposed building a train system that would go from San Francisco to Los Angeles at supersonic speed via a subterranean tube. This train system is called the *Hyperloop*. See

`https://en.wikipedia.org/wiki/Hyperloop`

The hyperloop has inspired much discussion of the so-called *gravity train*, which would use gravity alone (or almost alone) to travel below ground in a frictionless vacuum tube from one location to another. See

`https://en.wikipedia.org/wiki/Gravity_train`

The gravity train is closely related to the famous *Brachistochrone curve*. See

`https://en.wikipedia.org/wiki/Brachistochrone_curve`

We'll use AMPL to solve a slightly simpified version of the problem. Let's consider making a frictionless ramp for a "puck" to slide along. We will use x for the horizontal axis with, as usual, x increasing to the right and we will use y for the vertical axis with the *unusual* convention that y increases as we move downward. The beginning of the ramp will be at the origin of our coordinate system, $(0,0)$, while of the end of the ramp will be a meters out and b meters down: (a,b). The numbers a and b will be given specific values shortly.

Let's let $y(x)$ denote the vertical depth of the ramp at horizontal position x. Without going into the details of the derivation, it can be shown that the time T it takes to reach point (a,b) is

$$T = \int_0^a \sqrt{\frac{1+\left(\frac{dy}{dx}\right)^2}{2gy(x)}}\, dx,$$

where $g = 9.8m/s^2$ is the acceleration due to gravity and, of course, we have the constraint that $y(0) = 0$ and $y(a) = b$. The points on the curve $y(x)$, $0 \le x \le a$, are our optimization "variables". There's an infinite number of them. But, we can make a discrete approximation with $n+1$ points:

$$x_j = \frac{j}{n}a, \qquad j = 0, 1, \ldots, n.$$

The point $x_0 = 0$ is our "initial" x position and the point $x_n = a$ is our "final" x position. Corresponding to each of the deterministically determined x-values, we've got the corresponding unknown y-values:

$$y_j, \qquad\qquad j = 0, 1, \ldots, n.$$

In this discrete representation of the problem, we can approximate T like this:

$$T \approx \sum_{j=0}^{n-1} \sqrt{\frac{1+\left(\frac{y_{j+1}-y_j}{x_{j+1}-x_j}\right)^2}{2gy_j}}\,(x_{j+1}-x_j),$$

Because y appears in a denominator in the formula for T, we need to set y_0 to a number slightly larger than 0 to avoid division by zero. In other words, our problem can be described mathematically like this:

$$
\begin{array}{ll}
\text{minimize} & \displaystyle\sum_{j=0}^{n-1} \sqrt{\frac{1 + \left(\frac{y_{j+1}-y_j}{x_{j+1}-x_j}\right)^2}{2gy_j}}\,(x_{j+1} - x_j) \\
\text{subject to} & y_0 = \varepsilon \\
 & y_n = b
\end{array}
$$

Write an AMPL model to solve this problem. In your AMPL model use these values for the parameters: $g = 9.8$, $a = 10$, $b = 4$, $\varepsilon = 10^{-12}$ and a discretization with $n = 600$. In addition to giving values for the parameters in the problem, AMPL also allows you to give *initial* values for the variables. If no values are given, AMPL assumes the initial values for the variables should all be zero. That turns out to be problematic for this optimization model. To avoid the trouble, initialize the y_j's like this

$$y_j = jb/n$$

Use the solver LOQO to solve the problem. Using any of your favorite plotting tools (PYTHON, MATLAB, EXCEL, ...), make an (x, y)-plot showing the optimal curve.