The Matlab function "fmincon" can be used to solve for the critical slip surface as a generic optimization problem. fmincon takes as an argument an objective function, which itself must take as an argument a solution vector $x$. For this case, $x$ would be as follows:

$$x = \begin{bmatrix} x_0 \\ y_0 \\ x_1 \\ y_1 \\ \cdots \\ x_n \\ y_n \end{bmatrix}$$

where $n$ is the number of slices. This would be known based on the algorithm for slicing.

The Kinematic Admissibility module checks for 6 constraints on a slip surface, which can be translated to mathematical linear and non-linear constraints as follows. The index $i$ in the following equations represents the index of the solution vector.

1. Increasing $x$-ordinates

$$x(i) - x(i+2) \leq 0$$

2. Edge vertices of slip surface lie on slope surface

$$y_0 = y_{sl} \text{ at } x_0$$

$$y_n = y_{sl} \text{ at } x_n$$

3. Non-edge vertices of slip surface are within slope

$$y_i < y_{sl} \text{ for } 0 < i < n$$

4. Slip surface does not intersect slope surface except at edges. I believe this is covered already by the above constraints.

5. Slip surface is concave up

$$\frac{x(i+3) - x(i+1)}{x(i+2) - x(i)} - \frac{x(i+5) - x(i+3)}{x(i+4) - x(i+2)} \leq 0$$

6. No angles less than 110 degrees

$$\arccos\left(\frac{a+b-c}{2*a*b}\right) - 1.9199 \leq 0$$

where

$$a = (x(i+2) - x(i))^2 + (x(i+3) - x(i+1))^2$$
$$b = (x(i+4) - x(i+2))^2 + (x(i+5) - x(i+3))^2$$
$$c = (x(i+4) - x(i))^2 + (x(i+5) - x(i+1))^2$$

Additional constraints that may need to be introduced include constraints to ensure even-width slices (if desired) and constraints to ensure that the $x_0$, $x_n$, and all $y$ values are within the user-supplied bounds (input to SSP).

Some other notes and concerns:

- An initial guess must be supplied. We need an algorithm for determining this initial guess based on the slope geometry (initial guess could be a circular slip surface)
- Objective function and function for non-linear constraints each must only accept 1 argument (the solution vector $x$). However, they will also need input parameters to SSP. Thus, input parameters must be global variables or the objective function should be partially evaluated such that only the slip surface coordinate variables remain (need to look into how this can be done in Matlab)
- Would be interesting if we could find out why no one approaches slope stability this way, opting for genetic algorithms or other search methods instead.