Lab: Computing Airy’s Ai function

Brian Sutton

Airy’s Ai function is the unique solution to the initial value problem

\[ Ai'' - x Ai = 0, \]
\[ Ai(0) = 0.3550280538878172, \]
\[ Ai'(0) = -0.2588194037928068. \]

See the attached figure. There is no simple formula for \( Ai(x) \). In this lab, you will approximate the function, focusing on the negative real axis.

One approach to computing the function is to use a differential equation solver. To do this, it is necessary to convert to a system of first order differential equations,

\[ \frac{d}{dx}[Ai] = Ai', \quad Ai(0) = 0.3550280538878172, \]
\[ \frac{d}{dx}[Ai'] = x Ai, \quad Ai'(0) = -0.2588194037928068. \]

Then a routine like `rungekutta442` can be used to solve the system of first order equations, as in the following code:

```matlab
function Y=airy442a(X)
% airy442a Airy’s Ai function on the negative real axis.
% airy442a(X) is Airy’s Ai function evaluated at the entries of X.
% X is expected to have entries between -20 and 0.
%
% This function uses Airy’s differential equation \( y'' - x*y = 0 \).
persistent Xtable Ytable
if isempty(Xtable)||isempty(Ytable)
diffeq=@(x,y) [y(2);x.*y(1)];
a=-20;
b=0;
ai0=0.3550280538878172; % = Ai(0)
```
aiprime0=-0.2588194037928068;  \quad \% = Ai'(0)
y0=[ai0;aiprime0];
n=2000;
[Xtable,Ytable]=rungekutta442(diffeq,b,a,y0,n);
\% Xtable is a row vector (1-by-n)
\% Ytable is 2-by-n. The first row approximates Ai, and the second row
\% approximates Ai'.
end

Y=interp1(Xtable,Ytable(1,:),X);

A couple of notes about the preceding code: A table of values is first produced, approximating the solution on a grid. Then interpolation is used to extend the approximation to the entire interval $[-20,0]$. The persistent keyword provides a huge speedup—the table only needs to be computed once; on each subsequent invocation, the original table is reused.

Part 1. Analyze the absolute error associated with airy442a on $[-20,0]$. You may compare with Matlab’s built-in airy function. (Note that airy returns complex values with very small imaginary part. You may use the real function to throw away the imaginary part, as in real(airy(x)).)

The Airy function can also be expressed as the product of two functions,

$$Ai(x) = M(-x) \cos \theta(-x). \quad (1)$$

($M$ and $\theta$ will be defined in a moment.) This representation has the advantage that neither $M(-x)$ nor $\theta(-x)$ oscillates. See the attached figures.

Equation (1) could form the basis for an alternative method for computing $Ai(x)$. In the following code, airyMtheta442 refers to a Matlab function that you will write for computing $M$ and $\theta$.

```matlab
function Y=airy442b(X)

\% airy442b Airy’s Ai function on the negative real axis.
\% airy442b(X) is Airy’s Ai function evaluated at the entries of X.
\% X is expected to have entries between -20 and 0.

\% This function is based on the equation Ai(x) = M(-x) * cos(theta(-x)).
\% (See Abramowitz and Stegun.) M and theta are computed by solving
\% differential equations.

[M,Theta]=airyMtheta442(-X);
Y=M.*cos(Theta);
```
Exactly what are $M$ and $\theta$? Both solve initial value problems. $M(x)$ solves the initial value problem

$$\frac{d^2 M}{dx^2} + xM - \pi^{-2}M^{-3} = 0,$$

$M(0) = 0.71005\,61077\,75634\,5\ldots$,  
$M'(0) = -0.25881\,94037\,92807\ldots$,

[Note: Typo corrected. The second term now correctly involves $M$ instead of $\frac{dM}{dx}$.] and $\theta(x)$ solves the initial value problem

$$M^2 \frac{d\theta}{dx} = -\frac{1}{\pi},$$

$\theta(0) = 1.04719\,75511\,96598\ldots$.

Part 2. Convert the previous two initial value problems into a single system of first order differential equations,

$$\frac{d}{dx}[M] = \cdots, \quad M(0) = \cdots,$$

$$\frac{d}{dx}[M'] = \cdots, \quad M'(0) = \cdots,$$

$$\frac{d}{dx}[\theta] = \cdots, \quad \theta(0) = \cdots.$$

Part 3. Using airy442a as a model, implement a function airyMtheta442 for computing $M$ and $\theta$. You should use your answer to Part 2. You may start with the following template.

```matlab
function [M,Theta] = airyMtheta442(X)
    persistent Xtable Mthetatable
    if isempty(Xtable)||isempty(Mthetatable)
        ...
        ...
        ...
    end

    M=interp1(Xtable,Mthetatable(1,:),X);
    Theta=interp1(Xtable,Mthetatable(3,:),X);
```

Part 4. Analyze the absolute error associated with airy442b. You may compare with Matlab’s airy function.