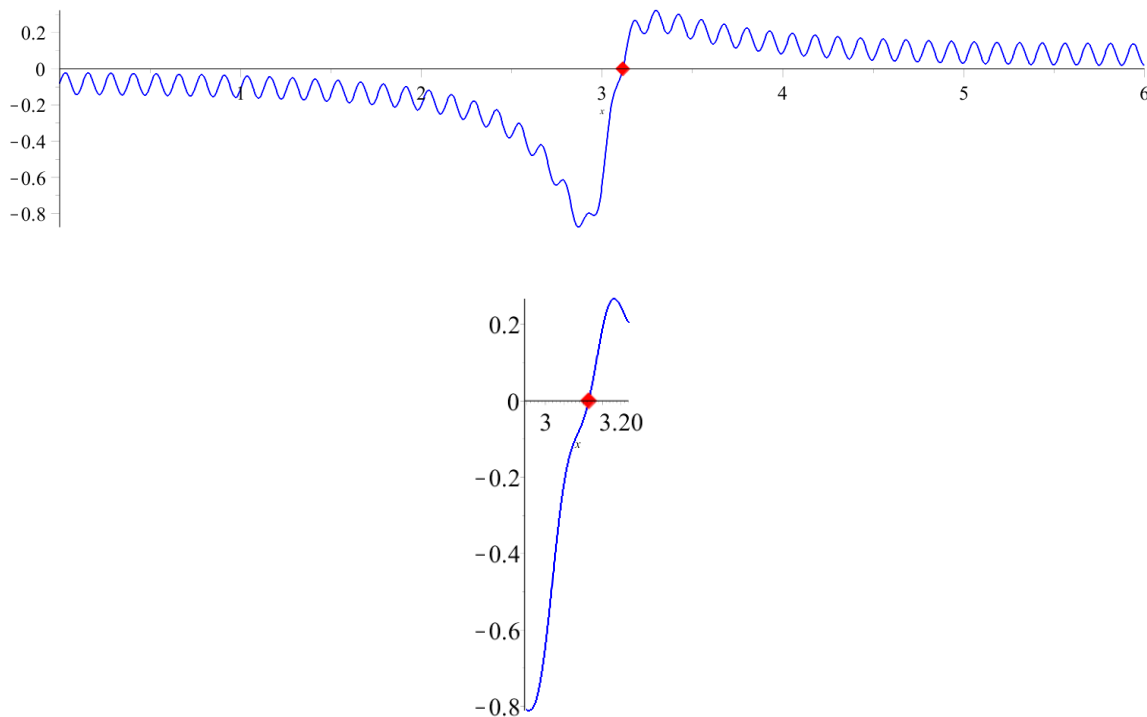The 2nd Function in the test script is not so nice:

$$f(x) = 0.01x - 0.03 + \frac{6(x - 3.1)}{40\,(x - 3)^2 + 1} + 0.06\sin(50x)$$

As you can, away from the root, the graph is not too smooth (Top Figure). However, if we get a bracket near the root, then things improve dramatically.

On the next page is output from my hybrid solver where I chose to print the "action" at each iteration. The initial bracket is $[0, 6]$, and the secant methods fails 3 times in the first 10 iterations and several of bisections are made, eventually cutting the bracket down to $[3.09, 3.18]$. Then the secant method quickly pinpoints the root.

One thing may be apparent as you read the output: *The bracket is not being updated during the Secant iterations.* This is because we want to give Secant a little "room to work." However, we are potentially giving up new information on the actual bracket. One way we can give Secant room to work and still use all the new information is to keep a "main bracket" $[a, b]$, and a temporary one $[c, d]$. Initial we would have $[c, d] = [a, b]$. After each successful secant attempt, readjust the temporary bracket. If the Secant attempt fails by falling outside the main bracket, we can replace the main bracket with the smaller temporary one!

1

```
>> testScriptForHybrid

Iter 1:  Secant Iteration Success!
    x=4.90427955, |f(x)|=1.03e-01,   Bracket: [0.000000,6.000000]
-----------------------------------------------------------------
Iter=2: x=6.234553760397,   Secant Iteration Failure!
  Iter 2 Bisecting!   New Bracket [3.000000,6.000000]
-----------------------------------------------------------------
  Iter 3 Bisecting!   New Bracket [3.000000,4.500000]
-----------------------------------------------------------------
Iter 4:  Secant Iteration Success!
    x=4.38874769, |f(x)|=8.54e-02,   Bracket: [3.000000,4.500000]
-----------------------------------------------------------------
Iter=5: x=4.668799674973,   Secant Iteration Failure!
  Iter 5 Bisecting!   New Bracket [3.000000,3.750000]
-----------------------------------------------------------------
  Iter 6 Bisecting!   New Bracket [3.000000,3.375000]
-----------------------------------------------------------------
Iter 7:  Secant Iteration Success!
    x=3.28401367, |f(x)|=3.09e-01,   Bracket: [3.000000,3.375000]
-----------------------------------------------------------------
Iter=8: x=3.557462041528,   Secant Iteration Failure!
  Iter 8 Bisecting!   New Bracket [3.000000,3.187500]
-----------------------------------------------------------------
  Iter 9 Bisecting!   New Bracket [3.093750,3.187500]
-----------------------------------------------------------------
Iter 10:  Secant Iteration Success!
    x=3.11282393, |f(x)|=7.36e-03,   Bracket: [3.093750,3.187500]
-----------------------------------------------------------------
Iter 11:  Secant Iteration Success!
    x=3.11484280, |f(x)|=1.07e-03,   Bracket: [3.093750,3.187500]
-----------------------------------------------------------------
Iter 12:  Secant Iteration Success!
    x=3.11458576, |f(x)|=2.28e-05,   Bracket: [3.093750,3.187500]
-----------------------------------------------------------------
Iter 13:  Secant Iteration Success!
    x=3.11459111, |f(x)|=6.74e-08,   Bracket: [3.093750,3.187500]
-----------------------------------------------------------------
Iter 14:  Secant Iteration Success!
    x=3.11459113, |f(x)|=4.26e-12,   Bracket: [3.093750,3.187500]
-----------------------------------------------------------------
Iter 15:  Secant Iteration Success!
    x=3.11459113, |f(x)|=1.87e-16,   Bracket: [3.093750,3.187500]
-----------------------------------------------------------------

Terminating because |x0-x1| is small..

Approx Root:  x=3.11459113,   Backward Error: |f(x)|=1.87e-16
```