Ross Miller

11/15/17

(a)

G = [g1, g2, g3]

with ϵ [-1, 1]

x1 = g1() =

x2 = g2() =

x3 = g3() =

Given g1, g2 , and g3 we can see that these functions are all continuous for all ϵ [-1, 1]. Also note that g1, g2 and g3 have continuous partial derivatives for ϵ [-1, 1]. Notice that:

=

=

=

since G( ϵ [-1, 1] for each g, this implies the existence of a fixed point.

Furthermore we can show that this fixed point is unique

So all partial derivatives are bounded by so a unique fixed point does exist.

Using the Gauss-Seidel method of iteration with a tolerance of E=1e-3 and an initial guess of , we get the following iterations:

k| x1| x2| x3| norm(xn-xn-1)

-----------------------------------------------------

01 0.347 -0.004 -0.524 0.653

02 0.500 0.000 -0.524 0.153

03 0.500 0.000 -0.524 0.000

(b)

G = [g1, g2, g3]

with ϵ [0, 1.5]

x1 = g1() =

x2 = g2() =

x3 = g3() =

Given g1, g2 , and g3 we can see that these functions are all continuous for all ϵ [0, 1.5] (their polynomials). Also note that g1, g2 and g3 have continuous partial derivatives for ϵ [0, 1.5]. Notice that:

=

=

=

since G( is not in the range [0, 1.5] for each g, there is no guarantee that a fixed point exist however we will continue as normal.

The smallest K value such that K/3 is the bounds of all the partial derivatives is 2.9, since 2.9 > 1 we could not guarantee uniqueness of a fixed point even if we were sure that one existed. But we’ll run our iterative procedure on this system anyways and see what happens.

Using the Gauss-Seidel method of iteration with a tolerance of E=1e-3 and an initial guess of , we get the following iterations:

k| x1| x2| x3| norm(xn-xn-1)

-----------------------------------------------------

01 1.181 1.115 0.892 0.181

02 1.206 1.098 0.901 0.025

03 1.199 1.100 0.900 0.002

04 1.200 1.100 0.900 0.001

(c)

G = [g1, g2, g3]

with

x1 = g1() =

x2 = g2() =

x3 = g3() =

Given g1, g2 , and g3 we can see that these functions are all continuous for all Also note that g1, g2 and g3 have continuous partial derivatives for Notice that:

=

=

=

since G( is not in the range for each g, there is no guarantee that a fixed point exist however we will continue as normal, and try to show that if a fixed point does exist if it will be normal or not

The condition for uniqueness fails when finding the partial derivative of x3 of g2, so we cannot say for sure whether the fixed point is unique or not, however we will proceed using an initial guess of , and a tolerance of E=1e-3 we get the following iterations:

k| x1| x2| x3| norm(x(n)-x(n-1))

-----------------------------------------------------

01 0.000 0.000 1.000 1.000

02 0.000 1.450 1.196 1.450

03 0.000 1.722 1.279 0.272

04 0.000 1.837 1.319 0.115

05 0.000 1.892 1.339 0.055

06 0.000 1.919 1.349 0.027

07 0.000 1.932 1.354 0.014

08 0.000 1.940 1.357 0.007

09 0.000 1.943 1.358 0.004

10 0.000 1.945 1.359 0.002

11 0.000 1.946 1.359 0.001

The problem of the two populations:

After doing some analysis messing and getting iterations that diverged I solved the problem algebraically and came up with a stable population #’s of 8000 and 4000 for populations 1 and 2 respectively. However I had difficulty getting my original fixed point functions to converge even with initial guesses that were extremely close. Eventually I came up with a set of functions that converged nicely these being:

and .

With an initial guess of 1000 for each population and a tolerance of E = 1e-3 the following iterations were constructed using Gauss-Seidell

k| x1| x2| norm(x(n)-x(n-1))

-----------------------------------------------------

01 9500.000 2875.000 8500.000

02 8562.500 3578.125 703.125

03 8210.938 3841.797 263.672

04 8079.102 3940.674 98.877

05 8029.663 3977.753 37.079

06 8011.124 3991.657 13.905

07 8004.171 3996.871 5.214

08 8001.564 3998.827 1.955

09 8000.587 3999.560 0.733

10 8000.220 3999.835 0.275

11 8000.082 3999.938 0.103

12 8000.031 3999.977 0.039

13 8000.012 3999.991 0.015

14 8000.004 3999.997 0.005

15 8000.002 3999.999 0.002

16 8000.001 4000.000 0.001