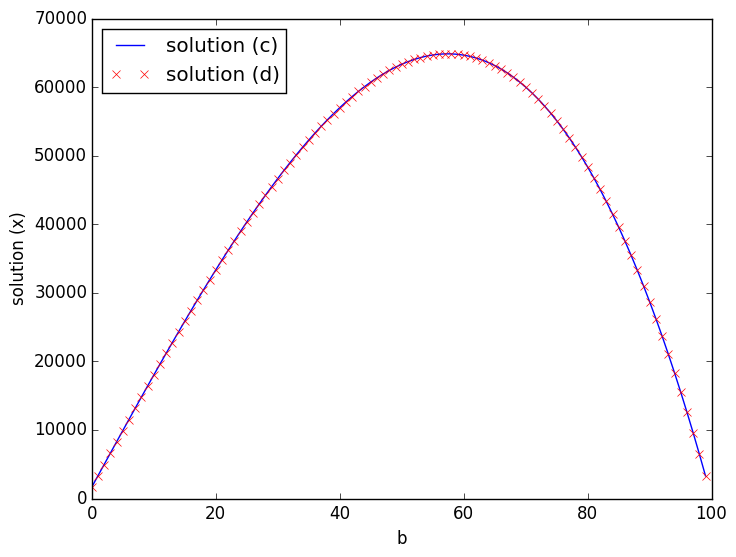
Problem 1

See P1.py. Plotting the solutions to (c) and (d) on the same plot yields the following.



Problem 2

The spectral radius is used as a condition for convergence. In some cases, it is used as a sufficient and necessary condition; in others, it is only a sufficient condition. The theorem given without proof states that:



…implying that:

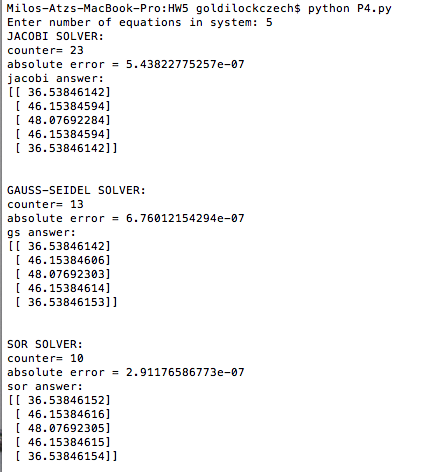


Because the rate of convergence is governed by the norm of **P**k+1, the second equation allows the relation of rate of convergence with the spectral radius of **P**k+1. In order for the convergence error to decrease as the number of iterations increases, this value must be less than one. Thus, the sufficient and necessary condition for convergence is:



Problem 4

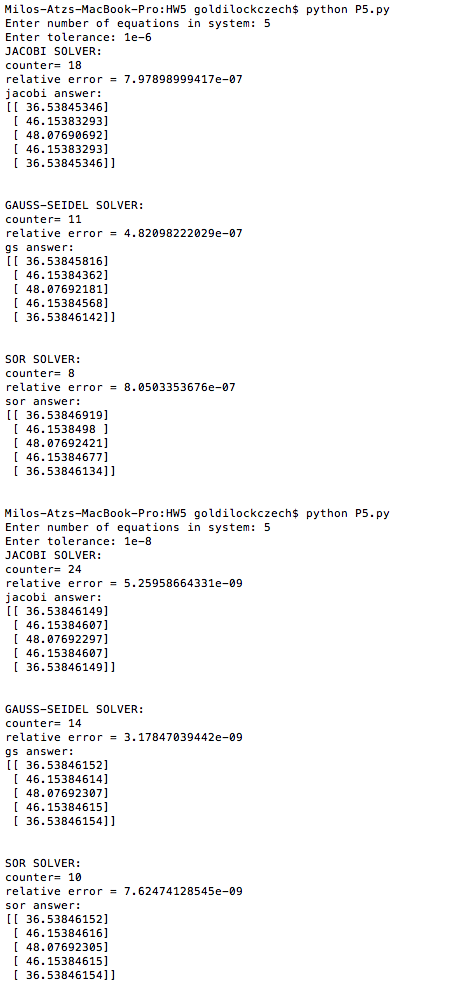
The python script “P4.py” contains all of the materials required to solve the given system of n equations. To run the script, simply navigate from the command line to the directory where the file is stored and run it from bash using the command “python P4.py”. The program prompts the user for the number of equations in the system (n) and then evaluates the equations using each of the three methods, printing the answer and the number of iterations required for convergence (indicated as “counter” in the output).

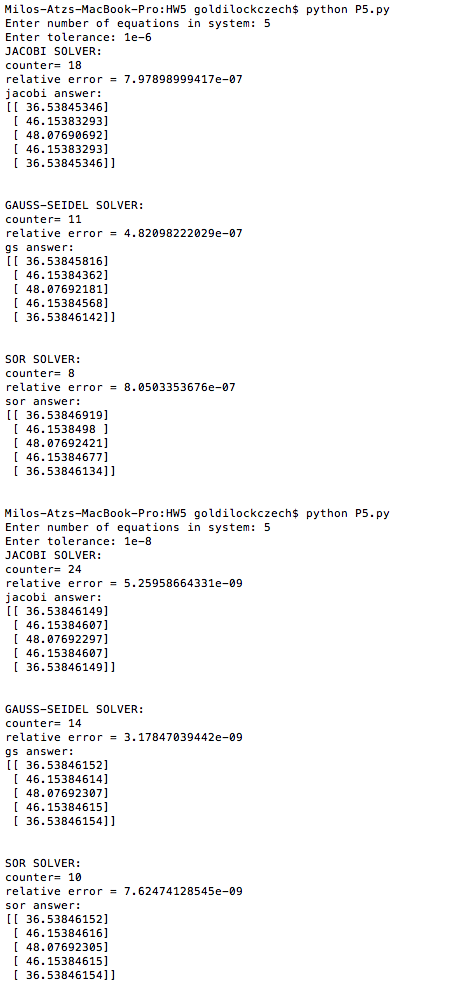


I was not sure what was meant by "absolute tolerance" - I couldn't find those words explicitly in the notes. I was torn between doing it this way and finding the actual absolute error (error = Ax-b, which should equal 0). I figured that for a larger system of equations, computing the actual error could become more expensive, whereas simply checking the convergence is much simpler. In addition, to properly compare this method with that in Problem 5, I figured this was more appropriate. Interestingly enough, this change does not affect the numerical answer nor the number of iterations...

Problem 5

a) To determine the number of iterations required for convergence using the relative error instead of the absolute error, we redefine the convergence criteria in the programs above. We can compare the “counter” output using this method to that from the previous question for both levels of relative tolerance.





The relative tolerance of 10-6 requires the least number of iterations. The absolute tolerance and relative tolerance of 10-8 require just about the same number of iterations. SOR requires the fewest number of iterations regardless of the convergence criterion. Reaching a tighter convergence tolerance requires more iterations. The absolute error at e=10-6 is less than the relative error at that tolerance for the Jacobi and SOR methods. Only the Gauss-Seidel method has lower relative error at that tolerance.

b) To determine *wopt* for the SOR method, we can calculate the spectral radius of the PSOR matrix for various values of *w*. The value that yields the lowest spectral radius is the optimum. The spectral radius is the maximum eigenvalue. To do this, I set up a vector of 101 omega values between 0 and 2. I then generate the PSOR matrix for each omega, and then find the spectral radius of that matrix. The value of omega that yields the lowest maximum eigenvalue (the smallest spectral radius) is the optimum value. This experiment found that omega = 1.0 is the optimum value. Two observations about this result: first, the left hand side of the graph is linear rather than quadratic, and second, this is not the value that gives the least number of iterations for the SOR solver. That value is somewhere in the range of 1.06 – 1.08, found via guess and check.

