Final Exam: Part I Take-Home ME 575

due 4/20/2016 11:59pm via Learning Suite

Timing: This take-home portion of the exam can be completed anytime during finals week. I have designed this to be completed with a 1.5 hours period, but you may take up to 2 hours if you need to. You must complete the exam in one consecutive period (you cannot stop and start again later). Please time yourself and don't go over 2 hours.

Materials: This portion of the exam is open-book, open-notes, open-homework. You may access any of the resources on the me575 website, but **not** the broader internet. You will need access to a computer with Matlab or Python setup to perform numerical optimization. To be clear, you will not need to write your own optimizers, but you will need to use existing commercial or open-source optimizers. You may download the starter files anytime before you begin, but don't look at them until you actually start.

Deliverables: Each problem asks for three things: 1) your answer, 2) a 1–2 sentence description of the how and why of your approach, 3) your code. Put your answer and brief remarks (1 and 2) directly in your code file as comments. You do **not** need to create a PDF write up. I must be able to run your code so don't include any random custom dependencies that I wouldn't have. I must also be able to read your code, so make sure it is reasonably clear.

Pacing: Each problem can be completed in around 15 lines of code. If you find yourself creating something much longer, you may want to rethink your approach (unless you are writing extra code for some testing). Don't spend too much time one problem. If you have a problem 80% right, don't spend too long trying to get that last 20% unless you've already completed your other problems.

Miscellaneous Be sure to read the instructions for each problem carefully. **Do not** turn this in late. Good luck!

1. The provided script "p1.m" or "p1.py" contains a function that takes in a vector x of length 5 and returns one scalar output h. Compute the gradient of h with respect to x using the complex step method about the point $x = [0.5, 0.1, 0.1, 1.0, 0.2]$. Report all the values of the gradient to 8 decimal places. Briefly describe the how/why of your approach and turn in your code.

2. The provided script "p2.m" or "p2.py" contains a function that takes in three inputs x_1, x_2, x_3, and returns one scalar output f. Your goal is to *maximize* f using all three x values as design variables. There is one nonlinear constraint:
$$\frac{x_1}{1000} + x_2 + 5x_3^2 \leq 9$$
There is a lower bound constraint: $(1000, 0, 0)$, and an upper bound constraint: $(6000, 1, 1)$. Report the optimal solution you find for x and f. Briefly describe the how/why of your approach and turn in your code.

3. The provided script "p3.m" or "p3.py" contains a function that takes in a vector x of length 3 and returns one scalar output f. Minimize f with respect to x. The only constraints are bound constraints (lower: $(-512, -512, -512)$, upper: $(512, 512, 512)$). Briefly describe the how/why of your approach and turn in your code.