Project 2. *Optimize a statically determinate 2D truss for uncertain loads.*

CEE 201L. Uncertainty, Design, and Optimization
Department of Civil & Environmental Engineering
Duke University
Henri P. Gavin and Shane McCormick

due: Parts 1 and 2: Wednesday, April 5, 2024

Introduction

In this project you will design a 2D truss to safely support uncertain loads. This will involve the use of 2D computational structural analysis to relate external loads to truss bar forces. In Part 1, you will apply this analysis within an optimization framework to iteratively and automatically converge upon an optimal design. In Part 2, you will use statistical simulation to assess the reliability of the optimized design to uncertainties in the external loads and in the as-built configuration. These two parts are broken down into seven tasks:

Part 1

Task 1. Sketch the design of a statically determinate truss.

Task 2. Parameterize the design in terms of the some of the node coordinates, and the bar cross sections.

Task 3. Derive equilibrium equations and use numerical optimization in order to minimize the weight of the truss such that the bars do not buckle or yield.

Part 2

Task 4. Quantify the sensitivity of re-optimized costs to constraint relaxation.

Task 5. Quantify the sensitivity of the cost and constraints to small perturbations in some of the design variables.

Task 6. Quantify the propagation of uncertainties on the optimized system’s safety.
Design Requirements

The truss shall transfer specified external loads to fixed reaction points. The location, magnitude, and direction of the external loads and the location of the reaction points are shown in Figure 1. The bars shall be made of steel: mass density $\rho = 7.86 \times 10^{-9}$ T/mm$^3$, elastic modulus $E = 200,000$ N/mm$^2$, and yield stress $S_y = 250$ N/mm$^2$. Units of N, mm, s, and T are consistent. Each truss bar, $b$, is to be a circular tube (outer diameter $D_{o,b}$ and inner diameter $D_{i,b} = 0.9D_{o,b}$). The design shall specify:

- the coordinates $(x_n, y_n)$ of the $N$ nodes in the truss;
- the outer diameters ($D_{o,b}$) of the $B$ bars in the truss.

The mass of material used (in Tons) shall be minimized such that the following criteria are satisfied:

- Safety against yielding shall be satisfied in each bar with a safety factor of 1.2 on the loads and a safety factor of 0.9 on the yield strength of the bars;
- Safety against buckling shall be satisfied in each bar with safety factor of 1.2 on the loads and a safety factor of 0.8 on the buckling strength of the bars; and
- Each bar shall be no longer than 6 m.

Figure 1. 2D Configuration of the reaction nodes and load nodes. (to scale) You will design a statically determinate truss to meet the requirements listed below. The vertical ($-y$, “downward”) forces have a value of $P_y = 100$ kN. The horizontal ($-x$, “to the left”) forces have a value of $P_x = 40$ kN. Nodes 1, 2, 3 and 4 are pinned. The goal of the design is to connect nodes 5 and 6 at least two of nodes 1, 2, 3 and 4.
Task 1: Sketch design alternatives.

Provide one sketch of a possible truss for each member of the group. All design alternatives must:

- inter-connect the nodes 5 and 6 to at least two of nodes 1, 2, 3, and 4 with a network of bars in which the bars could be no longer than the allowable length;
- have no un-stable nodes. In 2D, a stable node connects two or more truss elements that are not co-linear. In other words, at least two bars must connect to each node and the bars may not all lie along the same line. Almost all stable 2D trusses are a tessellations of triangles.

Sketch the configurations to be considered with the \((x, y)\) coordinate system provided in Figure 1. \textit{Use a straight edge!}

Consider one design alternative from each group member and, as a group, make a preliminary assessment of each design in terms of:

- the two “must-have’s” itemized above,
- complexity and aesthetics, and
- any other attributes you may wish to consider.

Score each alternative, rank the set of alternatives, discuss the ranking, and select one alternative for design optimization. Your group need not select the alternative with the highest ranking.

The truss configuration is fully specified by its node location coordinates and its connectivity matrix. For your selected design:

- number the additional node numbers starting from node 7.
- number the additional bar numbers.
- write \((x, y)\) coordinates for the approximate location of all nodes.
- write the truss element connectivity for all bars.
Statically indeterminate, statically determinate, stable, and unstable 2D trusses

Any statically determinate 2D truss can be analyzed by deriving two equilibrium equations for each node. The truss bar forces and the reaction component forces are the unknown variables in these equations. If this system of $2N$ equilibrium equations has $2N$ unknowns and all $2N$ equations are linearly independent, then the system of equations has a unique solution and the truss is called \textit{statically determinate}. For a planar (2D) truss to be statically determinate, the number of bars ($B$) plus the number of reaction force components ($R$) must equal twice the number of nodes ($2N$). This is a \textit{necessary} condition for static determinacy, but it is not \textit{sufficient}. Trusses for which $B + R = 2N$ could be unstable. In an unstable configuration the set of $2N$ equilibrium equations are not linearly independent and a unique solution can not be determined from the set of equilibrium equations. In an unstable truss one or more nodes can move without stretching or compressing any bar. Matrix equations with two or more linearly dependent rows (or columns) are called \textit{singular}. The determinant of such matrices is exactly zero.

If your design has \textit{more} unknowns than equations ($B + R > 2N$) it is called \textit{statically indeterminate} (e.g. Fig 2(a)). Redesigning an indeterminate system to be statically determinate involves reducing the number of unknowns (bars or reactions) or increasing the number of equations (nodes). This can be done by removing bar(s) (or independent reaction force components). But be careful. Depending upon the bar(s) you choose to remove, your truss could become unstable! In general, it is safe to remove a bar attached to a two-force reaction (Fig 2(b)). (Doing so effectively creates a linear dependence between the two force components at that reaction point.) Removing an interior bar could lead to instability, even if $R + B = 2N$ (Fig 2(c), 2(f), and 2(g)) but might not (Fig 2(c), 2(d), and 2(e)). Trusses with more than three reaction force components and one or more 4-node panels could be stable, depending on the geometry. In the truss shown in Fig 2(d)), node $G$ and bars $BG$ and $GF$ may be removed without affecting the stability of the truss. So nodes $B$ and $F$ are stable even without bars $BG$ and $GF$ (Fig 2(e)). Note that if no external force is applied at node $G$ in truss (d) bars $BG$ and $GF$ are zero-force bars, and therefore may be removed.

The stability of trusses (c) and (e) depends on the location of the truss nodes. If the 4-node panel in truss (c) is rectangular, the truss is unstable (Fig 2(f)). In truss (e), if the nodes $A,D,$ and $F$ are co-linear, the truss is unstable (Fig 2(g)).

In the process of optimizing the node location coordinates in a truss, the geometry of trusses like (c), (d), or (e) could become unstable. In trusses with a nearly unstable geometry, bar forces are very large, and small changes in geometry incur very large changes in truss-bar forces. The optimization process will naturally tend to avoid unstable geometries, since large bar forces require larger cross sections.

In this project the stability of the truss is quantified by the \textit{rank} of the equilibrium matrix that you will derive in Task 2. The equilibrium matrix has $2N$ rows (equations) and $B + R$ columns (unknowns). Design a truss so that the equilibrium matrix is square ($2N = B + R$). The solution to this matrix equation exists only if each equilibrium equation is \textit{linearly independent} of all the other equilibrium equations. In other words, the rank of the equilibrium matrix must be equal to its dimension, (twice the number of nodes). If the
configuration of the truss results in a rank-deficient equilibrium matrix, a warning messages like:

```plaintext
*** UNSTABLE TRUSS *** rank(EqMat) = 11 ***
Warning! matrix singular to numerical precision
```

are displayed on the console window. These messages indicate that the truss configuration being analyzed is practically unstable, and the analysis of this truss should not be trusted.

The computed performance of an unstable truss is non-sensible; such non-sensible results might cause the optimization process to fail (but might not).

Figure 2. statically indeterminate, determinate, stable, and unstable trusses.
Task 2: Specify your set of design variables.

In this task you will parameterize the truss design in terms of a set of design variables $\mathbf{v}$ that specify the geometry of the truss (the location coordinates of some nodes) and the outer diameters of the truss bars.

Consider which nodes could be moved during design optimization — not the loaded nodes and not the reaction nodes. You may chose to keep one or more coordinates of the movable nodes fixed. Those fixed coordinates would not be design variables.

Consider how enforcing a particular kind of symmetry can reduce the number of design variables while still allowing a number of nodes locations to be optimized. To maintain a desired symmetry, the coordinates of one node may be used to determine the coordinates of other nodes. Doing so reduces the number of independent design variables.

For example, if nodes 11, 12, and 13 are at pre-specified locations, and if nodes 14, 15, and 16 are to have the same height, $y_{14} = y_{15} = y_{16} = h$, and if all elements are to have the same diameter $D_o$, then $y_{14}$, $y_{15}$, and $y_{16}$ would not be design variables but $h$ would be a design variable; the value of the $y$-coordinates $y_{14}$, $y_{15}$, and $y_{16}$ would be set to the value of $h$. Simplifying the optimization problem by reducing the list of design variables restricts the kinds of topologies the optimized design can achieve to supposedly desirable topologies and simplifies the optimization process.

List the set of design variables (node coordinates and bar diameter(s)) in the problem.

Task 3: Equilibrium Analysis

Derive an equilibrium analysis of your group’s truss as described in Example of an Optimization-Oriented Analysis for Truss Design and Safety Analysis. The names of the .m-files in section 5 of this document are hyperlinks to downloadable files. (You’re welcome.)

Copy truss_opt.m to ABC_truss_opt.m and copy truss_analysis.m to ABC_truss_analysis.m where ABC are initials of your group members.

Lines to be edited are marked with an asterisk (*).

ABC_truss_opt

Edit your ABC_truss_opt_2024.m as follows,

1. Edit the truss element connectivity matrix, TEC, starting with row 8 using the connectivity values listed in Task 1.

2. Edit the truss bar diameter vector, $D_0$, starting with bar 8 of your truss.

3. Edit the initial guess of design variables, $v_{\text{init}}$, with values of the design variables listed in Task 2.
4. Edit the lower and upper bounds of the design variables. Bounding the truss node coordinates into non-overlapping areas could prevent the geometry from becoming unstable (e.g., overlapping nodes).

5. Note that ...
   If AnP = 0 then the truss is analyzed without plotting
   If AnP = 1 then the truss is plotted but not analyzed.
   If AnP = 2 then the truss is analyzed and plotted

6. At first use AnP = 1 to be sure that the nodes and bars are as you intend them to be. Once the nodes and the bars are as you intend, then set AnP = 2 to run an analysis and plot the deformed system. Blue bars in are in tension; red bars are in compression. If the truss is unstable then you will see excessive displacements and an error message
   Warning: Matrix is singular, close to singular or badly scaled.
   Results may be inaccurate. RCOND = NaN.
   If that happens refer to Pointer (i) on page 13.

ABC_truss_analysis

   Edit your ABC_truss_analysis_2024.m as follows,

1. Separate the design variables param into design variables related to truss bar diameters, Do, and node coordinates. Assign the design variables associated with truss bar diameters to Do. Assign the remaining design variables to node coordinates as specified in Task 1.

2. Edit the node coordinate matrix XY with variables corresponding to your node coordinate locations.

Optimize the design objective while satisfying constraints

   Run ABC_truss_opt.m

   If the optimization does not converge to a feasible solution, edit the values in the optimization options vector. If optimized design variables are bounded by your lower or upper parameter bounds, you may expand the bound in that direction.

   The Lagrange multipliers are returned by SQPopt in the vector lambda of length $(m + 2n)$. The first $m$ Lagrange multipliers correspond to the $m$ constraint equations. The next $n$ Lagrange multipliers correspond to the $v_b \leq v$ constraints, and the last $n$ Lagrange multipliers correspond to the $v \leq v_{ub}$ constraints.
Pointers

(a) Figure 1 shows six nodes, nodes 1, 2, 3 and 4 are reaction nodes, nodes 5 and 6 carry external loads. The loaded nodes are not reaction nodes. The \((x, y)\) location coordinates of these eight nodes are not design variables. The location of these nodes may not be modified in the design optimization.

(b) The only reaction nodes available to you are nodes 1, 2, 3 and 4, as shown in Figure 1. These nodes are fixed, in the \(x\) and \(y\) directions. Nodes 5 and 6 are not reaction nodes, but their specified location may not be allowed to change during the design optimization.

(c) Bars connecting two fixed nodes can not stretch or compress and therefore do not carry tension or compression. There’s no need to connect two fixed nodes with a bar.

(d) Your truss must be stable. In an unstable truss, nodes can move without stretching or compressing bars. If your truss is unstable, your truss_analysis function gives an error Warning: Matrix is singular, close to singular or badly scaled. Results may be inaccurate. RCOND = NaN.

One way to fix this might be to re-arrange the bars so that all the panels are triangular. Or you may need to add bars. Alternatively, your truss configuration might be fine, but it might be too flexible to support the given loads. One way to fix this may be to increase the bar cross section area (or bar diameters) in your initial design. Or you may need to add bars.

(e) The more design variables there are, the more flexibility the optimization algorithm has to fine-tune the solution, and to converge to a more optimal solution (a lighter truss), but the longer the optimization process will take, and the more sensitive the optimization could be to changes in the initial guesses. Also, the more finely-tuned a system is optimized, the less robust it’s performance might be to changes in operating environments.

(f) Each bar may have its own outer diameter. Or you may choose to design the truss to be built from bars with just a few diameters. For example, you could specify one group of bars to have a diameter \(D_{o1}\), a second group of bars to have a diameter \(D_{o2}\) and a third group of bars to have a diameter \(D_{o3}\).

(g) If your truss has 4 nodes in addition to the nodes with prescribed coordinates (the reaction nodes and the loaded nodes), then the design variables will include the eight values of the \((x, y)\) location coordinates of these four nodes, and the remaining design variables will give the diameters of the bars. . . . Lines 24-33 of truss_analysis_2024.m would change to something like . . .

```
x07 = v(1); % x-coordinate of joint 7
y07 = v(2); % y-coordinate of joint 7
x08 = v(3); % x-coordinate of joint 8
y08 = v(4); % y-coordinate of joint 8
x09 = v(5); % x-coordinate of joint 9
y09 = v(6); % y-coordinate of joint 9
x10 = v(7); % x-coordinate of joint 10
y10 = v(8); % y-coordinate of joint 10
Do = v(9); % outer diameters of all bars
```
Or, if you wish the pair of nodes 7 and 9 and the pair of nodes 8 and 10 to be symmetric about the \( y \) axis you can represent these eight values in terms of four design variables.

\[
\begin{align*}
x_07 &= v(1); & \text{\% } x \text{-coordinate of node } 7 \\
y_07 &= v(2); & \text{\% } y \text{-coordinate of node } 7 \\
x_08 &= v(3); & \text{\% } x \text{-coordinate of node } 8 \\
y_08 &= v(4); & \text{\% } y \text{-coordinate of node } 8 \\
x_09 &= -x_07; & \text{\% node } 9 \\
y_09 &= y_07; \\
x_{10} &= -x_08; & \text{\% node } 10 \\
y_{10} &= y_08;
\end{align*}
\]

(h) Use different values for the lower and upper parameter bounds \( v_{lb} \) and \( v_{ub} \) for each parameter. To start with, don’t make the in-bound ranges very large. If an optimized parameter is bounded by its lower bound (\( v_{lb} \)) or its upper bound (\( v_{ub} \)), expand the range for those bounded values and re-optimize; you’ll probably get a more optimal (lighter) design.

(i) Once you see the plot of the truss generated by \texttt{truss.plot}, carefully scrutinize the figure to check that the truss as modeled in the code conforms to your expectations. If it does not, re-check the truss node location matrix \( XY \) and the truss element connectivity matrix \( TEC \).

(j) Adjust the initial values of the node location coordinates, to get the plotted figure to look reasonable. Starting the optimization from a weird-looking configuration is challenging for numerical optimization algorithms.

(k) Try relocating long bars in compression (i.e., long diagonal braces) so that they carry tension.

(l) Once you see that the figure generated by the \texttt{truss.plot.m} analysis looks right, you may set \( \text{AnP}=0; \) in your function \texttt{ABC_truss_opt.m} to get the optimization to run faster, or you may leave \( \text{AnP}=1; \) to experience how cool and informative it is to visualize the optimization process as your truss geometry changes from iteration to iteration.

(m) \textit{Do not ever} edit any of the optimization codes downloaded from the course webpages, e.g., \texttt{NMAopt.m, avg_cov_func.m, truss_plot.m}. They are already \textit{perfect}. 
Part 1 Project Report Contents

1. A (one page) introduction describing the goals of your project design, the conceptual approach you took to developing a parameterized design, the benefits of optimization-oriented design analysis, and the numerical methods taken to optimize the design ... as if you were describing your project to a BME sophomore.

2. Include the candidate sketches from Task 1, with a brief explanation of why the designer of each candidate thinks the design is good. Include the assessment table from Task 1 with an explanation of why the selected candidate design was chosen.

3. Include the figure showing the initial guess for your selected truss configuration, showing labels for node numbers.

4. List the design variables for your design and their optimal values. Include the figure of your optimized truss showing truss bar forces.

5. What is the optimal cost in units of metric Tons? (Remember to specify units.)

6. Which, if any, of the constraints are binding the optimum solution?
   \[ \text{Active Constraints} = \text{find}( g > 0 ) \]
   \[ \text{Active Constraint Values} = g( \text{find}( g > 0 )) \]
   What do each of these constraints represent, physically?

7. Why might the Lagrange multipliers for buckling be larger than, or smaller than the Lagrange multipliers for yielding? How might that guide a design revision for a lighter truss?

8. Why is the safety factor for buckling strength \( \phi_B \) smaller than it is for yielding strength \( \phi_Y \)?

9. Please share any additional pointers that would have been helpful for Part 1 of this project.

10. Submit your .m-files ABC_truss_analysis_2003.m and ABC_truss_opt_2003.m to gradescope. Do not attach any other code and do not include the code in your project report. Comment all your .m-files extensively. Comment most lines. Be fastidious about alignment and indenting. Use the downloadable code provided for this project as an example of clean and commented code.
Task 4: Quantify the sensitivity of re-optimized costs to constraint relaxation.

To investigate the effect of relaxing constraints, individually decrease the value of the load and increase the value of the material strength (the yield stress) by ten percent (in your .m-function ABC_truss_analysis.m). Re-run the optimization from Task 4, and tabulate the results of the optimization in terms of parameter values and the cost, using the optimal values from Task 4 as your initial guess. Complete the table below:

<table>
<thead>
<tr>
<th>$P_x$</th>
<th>$P_y$</th>
<th>$S_y$</th>
<th>cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>100</td>
<td>250</td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>100</td>
<td>250</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>90</td>
<td>250</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>100</td>
<td>275</td>
<td></td>
</tr>
</tbody>
</table>

1. If the point load values are decreased by 10 percent, how much steel can be saved through reduced section areas?
2. If the yield stress is increased by ten percent, how much steel can be saved through reduced section areas?
3. What makes the biggest difference in the optimized cost, reducing the loads by 10 percent, or increasing the yield strength of steel by 10 percent? Why do you think that is the case?

Task 5: Quantify the sensitivity of the cost and constraints to small perturbations in some of the design variables.

Using the optimal solution from Task 4, evaluate the cost and constraints for a set of perturbed design variables. Select three design variables that you think are most important to the design. Call them $v_i$, $v_j$, and $v_k$. These design variables could be a node location coordinate or a bar section diameter. If $p^*$ represents the vector of optimal design variables, evaluate the cost and constraints for the six sets of design variables in the table below; note the cost and max constraint for each set.

<table>
<thead>
<tr>
<th>parameter label</th>
<th>set 1</th>
<th>set 2</th>
<th>set 3</th>
<th>set 4</th>
<th>set 5</th>
<th>set 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>$v_i$</td>
<td>$0.95v_i^*$</td>
<td>1.05$v_i^*$</td>
<td>$v_i^*$</td>
<td>$v_i^*$</td>
<td>$v_i^*$</td>
<td>$v_i^*$</td>
</tr>
<tr>
<td>$v_j$</td>
<td>$v_j^*$</td>
<td>$v_j^*$</td>
<td>$0.95v_j^*$</td>
<td>1.05$v_j^*$</td>
<td>$v_j^*$</td>
<td>$v_j^*$</td>
</tr>
<tr>
<td>$v_k$</td>
<td>$v_k^*$</td>
<td>$v_k^*$</td>
<td>$v_k^*$</td>
<td>$v_k^*$</td>
<td>$0.95v_k^*$</td>
<td>1.05$v_k^*$</td>
</tr>
</tbody>
</table>

1. Which of these three design variables most significantly affects the cost function?
2. Which of these three design variables most significantly affects $\max(g)$?
Task 6: Quantify the propagation of uncertainties on the optimized system’s safety.

In realistic use case scenario, the product will not conform exactly to the design specification and the environment that the product is subject to may be more or less demanding than the designer anticipated. Because of this, the operating environment and the associated performance of the product (in terms of its ability to withstand its operating environment) should be viewed as random variables. In this project, we assume that all of the design variables must be positive, which is quite often a necessary assumption.

To determine the “probability of failure” of the design, we perform many analyses, with each analysis involving a random value for the operating environment (external loads) and material properties (yield strength). Because the loads and the material strength must be positive, the probability distributions of these quantities must conform to a particular class of probability distributions that are defined only for positive values. The log-normal probability distribution is one such distribution. Unlike the symmetric normal (Gaussian) probability distribution, the log-normal distribution function is not defined for negative values. (A negative value for strength would be senseless.) To generate a log-normal random value \( v_i \) with median \( v_{\text{med}} \) and coefficient of variation \( c_{P_i} \), use the downloadable .m-functions `logn_rnd.m` and `corr_logn_rnd.m`. If in your analysis code, you specified a load as a negative number, apply the negative sign in the code wherever you use the variable, and change the value of the number to be positive.

Furthermore, to model the randomness of manufacturing variability, a level of quality-control can be presumed so that components that are out of spec (i.e., truss bar cut too long, or too short) are discarded. To model this variability, the design variables \( (v_1, v_2, \cdots, v_n) \) can be perturbed by adding a small uniformly-distributed random variable.

To determine the probability of failure in a situation in which there is uncertainty both in the as-built condition of the product (random material properties and random design variables) and in the operating environment (random loads), evaluate the cost function and the constraints 1000 (or more) times, each time use a randomly perturbed set of design variables and randomly perturbed values for the material strength and the loads. Set the median parameter values to be the optimal value you found in Task 3, \( v_{\text{im}} = v_{\text{opt}} \), and the standard deviation being 1 percent of the mean value, \( c_{P_i} = 0.01 \). Set the median strength value to be the nominal strength of 250 N/mm², and the median external loads to be the nominal loads given in Figure 1. The standard deviation of the strength will be 10 percent of the mean, and the standard deviation of the load will be 30 percent of the mean. Furthermore, consider the components of force \( P_x \) and \( P_y \) to be correlated with a correlation coefficient of 0.3.

The system is considered “failed” if one or more safety constraints (yielding or buckling) is positive. For each evaluation of the cost and the constraints, count how many evaluations result in a constraint that is positive. The probability of failure is the number of evaluations that result in a constraint violation divided by the total number of evaluations. If you don’t get any constraint violations, then increase the total number of evaluations, and re-run the
Optimize a 2D truss.

Safety analysis.

The downloadable .m-functions `truss_safety_2024.m` carries out these safety analysis calculations. Copy `truss_safety_2024.m` to `ABC_truss_safety_2024.m` and `ABC_truss_safety_2024.m` by changing the values for `constants`, `opt_params`, and `NSC` and by editing lines marked with * for your particular design.

The downloadable .m-function `correlation_analysis.m` determines how constraint violations are correlated with errors in the design variables and uncertainty in the strength $S_y$ and loading $P_x$ and $P_y$.

The Monte Carlo analysis will take a minute or less to complete.

1. In computing the failure probability, why are all the safety factors set to 1.0?

2. What is the failure probability for your optimized design?

3. If your failure probability is non-zero, what are the failure mechanisms corresponding to the constraints in violation (the active constraints)?

4. Which (if any) design variable errors are most positively correlated with the active constraints?

5. Which (if any) design variable errors are most negatively correlated with the active constraints?

6. Which (if any) load or strength uncertainties are most positively correlated with the active constraints?

7. Which (if any) load or strength uncertainties are most negatively correlated with the active constraints?

8. Using your answers to questions 3-7, how could you modify your design optimization to most effectively reduce the failure probability?
Part 2 Project Report Contents

1. A (one page) introduction describing the importance of uncertainty analysis for systems with uncertain loads and behavior and how Monte Carlo simulation can help quantify uncertainties in system behavior. as if you were describing uncertainty propagation to a BME sophomore.

2. Include completed tables shown in Tasks 4 and 5.

3. Include and describe figures 30 and 31 generated by ABC_truss_safety

4. Answer the questions listed at the end of Tasks 4, 5, and 6.

5. Please share any additional pointers that would have been helpful for Part 2 of this project.

6. Submit your .m-files ABC_truss_analysis.m and ABC_truss_safety_2024.m to grade-scope. Do not attach any other code and do not include the code in your project report. % Comment all your .m-files extensively. Use the downloadable code provided for this project as an example of clean and commented code. Comment most lines. Be fastidious about alignment and indenting.

Grading

(45 points) Correctness of the formulation and implementation of your optimization-orientation analysis. The quality of your code in terms of formatting and commenting.

(5 points) The optimized cost of your truss, in comparison with other group projects.

(5 points) The failure probability of your truss, in comparison with other group projects.

(30 points) A complete report including the ten items from Part 1 and the six items from Part 2.

(15 points) Quality of the report: Hand-drawn sketches are drawn with a straight edge and neatly annotated. (typed, 11pt or 12 pt, 1.5 space, 1 inch margins, page numbers, figure numbers, figure captions)