EMA 601

Homework 6

You just took a job at General Dynamics - Convair. Your boss wants to have you work on a new team of engineers that will perform test-analysis correlation and finite element model updating on several new spacecraft projects that are being proposed by the Air Force. She wants you to become familiar with Matlab's capability to perform model updating using its constrained optimization function "fmincon". A simple spring-mass system, shown below, is assembled and tested by GD for use as a test bed for the group. You are given an analytical model of the system in the Matlab database on the class webpage, called "example6.mat". The database also contains the test frequencies and mode shapes from the vibration test. The files in the database are:

DOF = dof vector corresponding to vertical deflection of masses labeled below

Mm = physical mass matrix

Km = physical stiffness matrix

thetanom = nominal design variable values (k1 thru k10, m1 thru m6)

PHITest = experimental mode shapes

fTest = experimental modal frequencies in Hz.

All spring stiffnesses and mass values are assumed to be design variables that must be updated, 16 in all.

Your job is to update the model based on matching modal frequencies alone. Use the optimization techniques shown in class and the Matlab function "fmincon". You can modify the Matlab functions "xupdate" and "xmin" discussed in class and found on the class webpage to fit this problem. At each iteration in "xupdate", you should compute all eigenvectors and eigenvalues of the current model, compute the eigenvalue sensitivities, match the test and analysis modes using cross-orthogonality, form the test/analysis frequency error vector, call "xmin" which iteratively computes the objective function and its gradient, and calls "fmincon".

This problem is small enough that you can do everything in physical space instead of transforming to modal coordinates. The sensitivities of the mass matrix and stiffness matrix with respect to the design variables can be computed analytically because the system is so simple. For example, the sensitivity of the stiffness matrix with respect to the ith design variable is a 6x6 matrix. This can be stored in a three dimensional array in Matlab as dKi(:,:,i). The total sensitivity array dKi has the size 6x6x16. The first index correspond to rows of the sensitivity matrix, the second is columns of the matrix, while the last index indicates the appropriate design variable.

Note that you must use dimensionless design variables in your optimization.

Good Luck!