Weekly Team Meeting Individual Report

Eric Sirois

3-Point Bending Device

BME-4910

Meeting Date: 4/02/09

**Where I left off last week:** Last week I finished debugging the transmural strain calculation. I also removed recurring bugs in the triangular meshing program and developed a new subroutine that output the transmural strain and meshing results. Additionally, I developed a simulated transmural strain experiment and used the finished program to analyze the results. Finally, Xuan and I performed a transmural strain experiment on a porcine aorta specimen. The results of this experiment can be seen below in Figure 1.

![Figures](image.png)

*Figure 1 – (a) undeformed mesh of a porcine aorta and (b) E22 strain result after deformation.*

**Difficulties last week:** Last week marked significant progress toward the completion of the project. That being said, difficulties still persisted that need to be addressed moving forward. First, the triangular meshing program has a bug that still allows overlapping triangles to be built. Secondly, the point tracking for the transmural strain tissue experiment described above was done manually. This process will need to be automated and implemented seamlessly for desired device function. Lastly, we will have to develop a better and more consistent method for marking the tissue.
**Comments from last week’s team meeting:** No comments this week, but the client would like to see an experiment performed where the bath, linear actuator, and bending bar mount are all used together. The experiment will not include the camera or camera mount yet, and will thus not require data analysis. This test would just be to show the progress of whole device integration.

**Actions this week:** This week I have been working on development of controlling programs that will call the subroutines that I have already developed. My first challenge was to understand the appropriate method for integration of Matlab subroutines into LabView virtual instruments. Fortunately, LabView is capable of directly calling Matlab functions, and sending in arguments just as they would be used in a script. The only remaining task for me has thus been to alter the functions such that the only inputs or interactions with outside programs are through the input arguments. This applies most specifically to calls to Excel spreadsheets, which has been our preferred method of calling test data. This has turned out to be a more difficult task than anticipated, but is moving along steadily. One challenge has been that much of the flexural rigidity program was hard-wired for data sets of a given size, and it must be rewritten in several locations such that loops iterate through based on a measurement of the input data file size. This is not necessarily a difficult change, but has required a thorough evaluation of the code.

The implementation of the calling programs has been the main focus of this week. LabView Mathscript files were developed for both the flexural rigidity (Figure 2) and transmural strain (Figure 3) subroutines. The most challenging part of this process was identifying what exactly the function input and output arguments should be. The input arguments, in particular, had to be identified in great detail to facilitate integration with inputs from the CCD camera data collection, which has been developed by Xuan. Output plots were also developed for the deltaK vs. M/I and deltaK vs. E plots of the flexural rigidity program. Unfortunately, it appears that LabView cannot call the Matlab “plot” or “fill” built in functions. This means that graphical strain output will not be possible in real time. LabView does however have the capability of plotting points using a color gradient scale. My plan at this time is to output the location of the neutral axis along with the point distribution plot to the user in real time and facilitate the output of full strain data for post-processing using Matlab.
Figure 2 – Mathscript virtual instrument implementing Matlab user-defined functions used in the determination of flexural rigidity
**Figure 3** – Mathscript virtual instrument implementing Matlab user-defined functions used in the determination of transmural strain.

**Current Status**: I am continuing to implement the Matlab subroutines into LabView virtual instruments. I am hopeful that by the end of next week, I will be able to reproduce the pilot trial analysis using these programs. This will naturally lead to integration with data input directly from the CCD camera.