PARTruss - Parallel Truss Solver

May 2017

Version 1.0

Document Revisions

Date	Version Number	Document Changes
05/03/2017	0.01	First Release
05/04/2017	1.0	Final Release

Table of Contents

1 Introduction

1.1 Scope and Purpose

PARTruss is a parallel truss solver using the direct stiffness method. The results are portable, machine parsable, and can be reused in PARTruss with no modification needed. PARTruss requires initial truss data to match the existing data spec, available online at cuTRUSS.tk or on the project GitHub.

This guide will cover basic usage of running stress calculations on a truss file, and the visualization of both the initial conditions and the results. This guide assumes a c++11 compliant compiler, as well as a CUDA compatible GPU and an installation of a NVCC.

1.2 Process Overview

The back end portion of the software handles calculation of stresses and forces throughout a truss system. This workflow in CLI only, and will return all output to an output file.

The front end portion of the software visualizes the truss, showing stresses on the individual elements. This visualization can be moved and rotated as needed to see pats of interest.

Typical Workflow:

1. Build backend executable
2. Run executable, passing a truss file
3. Open truss file on visualization site
4. View truss in browser

2 Processing a Truss File

This guide assumes a c++11 compliant compiler, as well as a CUDA compatible GPU and an installation of a NVCC. The truss file must follow the data specifications laid out on the project page. See the project page for more information.

2.1 Building the Executable

PARTruss allows for an easy and intuitive compilation on any compatible system.

2.1.1

1. git clone https://github.com/PARTruss/PARTruss.git
2. cd ./PARTruss/src/
3. make
4. The executable will be stored as "main", in the current directory

2.2 Processing A Truss File

2.2.1

1. Return to the "./PARTruss/src/" folder
2. ./main <inputFile> [-vvvvv] [-ccccc] -o <outFile> **OR**
   ./main -o - (to read from STDIN, write to STDOUT)
3. -v flags increase verbosity, including matrices at -vvv
4. -c flags increase level of commentary, human readable state information
5. Results will be returned in a "trussOut.json" in the same directory
6. If there is any additional console output, check the debug level defined in main.c and recompile as needed.

3 Visualizing a Truss File

PARTruss visualization can be run locally through the provided WebUI files, or using the online interface. Online will be the most recent release, and may provide additional features. The online version is available at partruss.github.io/PARTruss/src/webUI/

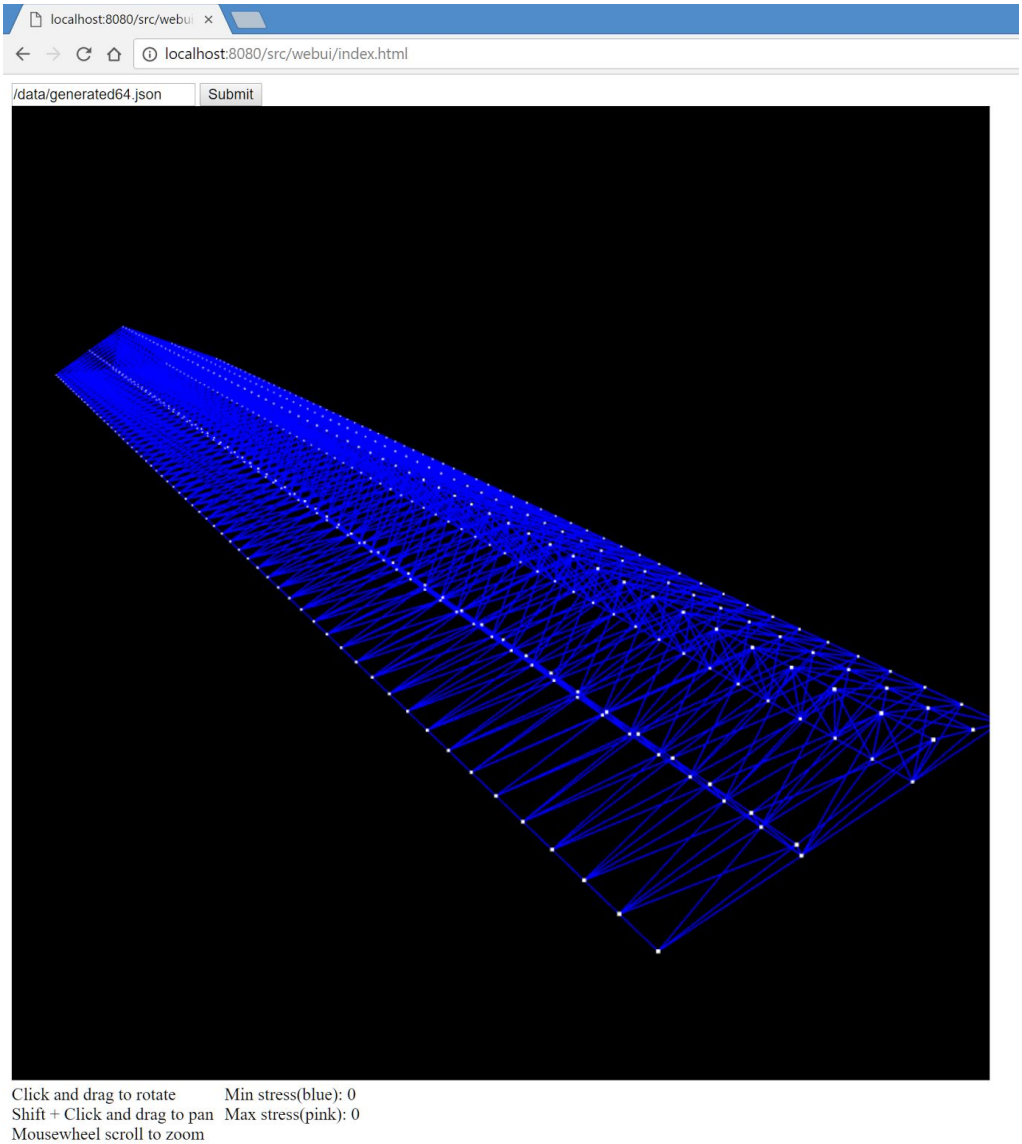
Locally generated files can only be viewed in the locally running webUI.

Please follow the directions provided by the web UI as these may change more frequently with added features and improved file handling.

3.1 Local Operation

3.1.1

1. The webUI is available in the directory PARTruss/src/webUI
2. Host the base repository directory on an http enabled webserver
3. Navigate to localhost/src/index.html
4. Javascript must be enabled in browser
5. Supply the path of the truss file to be generated, where the base directory is at the root of the repository



If you encounter issues not addressed by this user guide, please contact your account manager for additional support