Open Source in Scientific Computation

Moorthy

Open Source in Scientific Computation

Created by Moorthy / @mskmoorthy and modified by Wesley Turner / @wd-turner

Learning Objectives

1. Become familiar with the breadth of Open Source Code used in Scientific Computation
2. Gain some practical experience with Open Source alternatives to closed source code
3. Learn how Scientific Computing principles are used to provide realism even in simple games

Reading List

http://doc.sagemath.org/html/en/a_tour_of_sage/

http://www.pymunk.org/en/latest/

- Read the first page and familiarize yourself with the API Reference.

Open Source in Scientific Computation has two axes

- Heavy Lifting Backend Computations
- Higher-level language Wrappers for rapid Deployment

Backend Computations

- Usually written in C, C++ and Fortran
- May use extensive well developed/researched library packages
- Will have experts in the specific discipline (physics, mathematics, climate modeling, circuit designers. financial experts)

- Will have support for Parallelism

Backend Computations (contd)

May test the algorithms in Matlab, Mathematica, Python, etc. before full fledged implementation

Arrays (multidimensional) are used extensively

Double precision computations are used extensively.

See:

- OpenSurgSim
- Eigen
- ITK

Applications of Scientific Computation

- Finite Element Modeling - Structural Safety
- Fluid Mechanics - Aerodynamics - Aircraft Design
- Circuit Simulation
- Genomics and Drug Simulation
- Image Processing Applications
- Airline Scheduling, Linear and Nonlinear Programming Application
- Astronomy and Space Computations

Applications of Scientific Computation (contd)

- Simulation of Materials
- Manufacturing Process
- Network Simulation
- Weapon Simulation/Modeling
- Solving NP-hard/complete Problems
- Protein Folding
- Physics Engines for Game Playing
- Chemistry

Commonly Used Tools

(Not All Open Source)

- Matlab (simulink), Maple, Sage (pure math), Mathematica
- CUDA, MPI libraries for parallel computation
- LINPACK, EISPACK Libraries to do matrix computations (netlib)
- glpk (for linear programming) and other mathematical programming packages
- ACM Collected Algorithms

Commonly Used Tools(contd)

- Finite Element Packages
- boinc for distributed/volunteer computation - factoring large numbers, Milkyway@Home
- Visualization Toolkit (VTK)
- Image Processing toolkit: Insight Toolkit (ITK), opencv
- Graphics rendering: openGL
- VLSI design : Spice

Python

- pypy, mplotlib, cpython, numpy, pygame, python notebook
- networkx - graph/network manipulations
- biopython - Biological libraries in Python

Other Language Support

- Julia - high performance numerical and scientific computing
- R - Statistical Computing Data Science (we have a module...)
- Java - Special Packages like Cytoscape

Warm Up Exercise 0

- Navigate to http://www.sagemath.org/
- Sagemath is an open source alternative to Maple and Mathematica
- Now go to SageMathCell
- Compute 450th Fibonacci Number

Warm Up Exercise 0 (contd)

Try a few more commands

Evaluate the 12th root of (3987^12+4365^12)

- Then look at: website https://n.pr/2TRyZjc
- Moorthy has sample output in https://github.com/rcos/Sci-Computing/blob/master/output-console.pdf

Now check out CoCalc for a business model

Warm Up Exercise 1

- Install pygame and go through the first page of the tutorial at https://nerdparadise.com/programming/pygame/part1
- At the end, you will have a simple "game"

Warm up exercise 2

Download angrybirds for python:

https://github.com/estevaofon/angry-birds-python

Install pygame and pymunk using pip (pip install pygame pymunk)

Run by going to the “src“ directory and running “python main.py“

- (Or “pythonw main.py“ on a Mac.)

Study the source code

Warm up exercise 2

- We may be nearing "End of Life" on this example:
  - You need Python v3.6 or above
  - You need pymunk 5.7.0 or below:
    - pip install 'pymunk<=5.7.0' --force-reinstall
  - You MAY need to disable sound on WSL
    - Edit main.py and disable the music
- I am willing to host a revised version if anyone gets ambitious

Warm up exercise 2

Play the game:

- Understand where the physics occurs
- How will the angry birds game play on the moon?
- How will the angry birds game play in a vacuum?
- Read the Wired article http://www.wired.com/2010/10/physics-of-angry-birds/

The End

by Moorthy