



Resonant Ultrasound Spectroscopy

BY EN YUNG HOANG



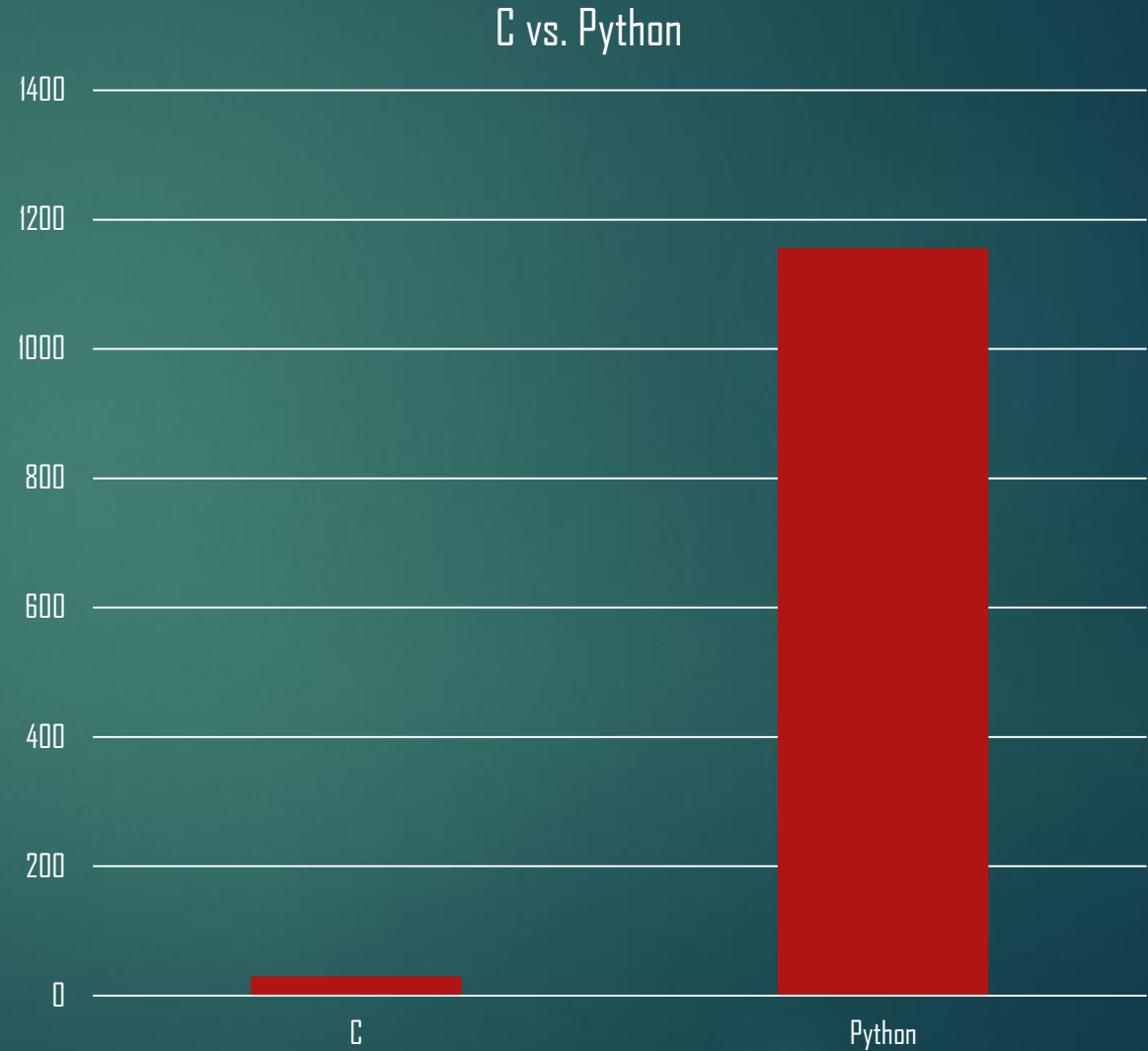
About this project

The Problem

- ▶ Original C implementation transformed into Python
 - ▶ Speed reduced significantly
- ▶ Solution:
 - ▶ Wrap C functions within Python functions

C vs. Python Implementations

- ▶ Running at 100 iterations each
- ▶ Results:
 - ▶ C: **29.545 seconds**
 - ▶ Python: **1156.073 seconds**



Cython

- ▶ Is an optimization static compiler
- ▶ Uses a Pyrex file

Alternatives to Cython

- ▶ SWIG
- ▶ Numba
- ▶ SIP

- ▶ Ctypes
- ▶ Weave

Why I chose Cython

- ▶ Close integration with external C libraries
- ▶ Interchangeability between C and Python within one file
 - ▶ Pyrex file (.pyx)
- ▶ Executable file
- ▶ Portability

Cython's Performance

Before:

ncalls	tottime	percall	cumtime	percall	filename:lineno(function)
214350	0.263	0.000	0.263	0.000	rustools.py:807(volintegral)

After:

ncalls	tottime	percall	cumtime	percall	filename:lineno(function)
214350	0.057	0.000	0.060	0.000	rustools.pyx:808(volintegral)

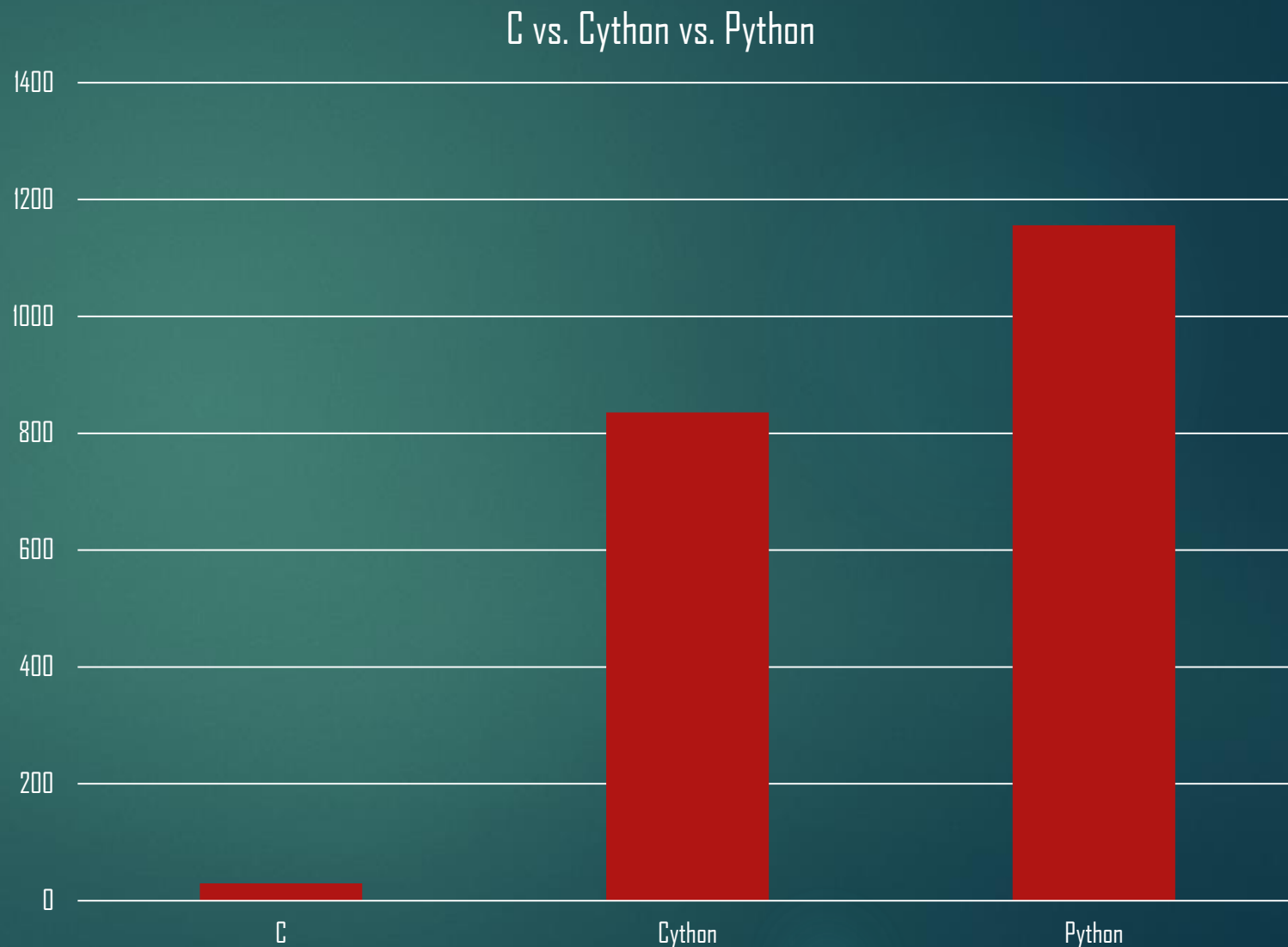
C vs. Cython vs. Python

► Each running at 100 iterations each:

► C: **29.545 seconds**

► Cython (Wrapped 'volintegral' – a frequently called function):
835.518 seconds

► Python: **1156.073 seconds**





Demo

Plans for semester 2

- ▶ Finish wrapping the code
 - ▶ Upload to GitHub
- ▶ Further developments of the RUS code



Any Questions?