CS 179 Project Ideas

Proposal Guidelines

- 1-3 sentence summary of project
- team members (pair or solo)
- Hoping to do 3 week or 5 week?
- 1-3 paragraph explanation of project with background
- Why is this challenging? Has it been done before? What tricky things are you going to have to figure out? 1-2 paragraphs
- What are the deliverables? Goals? 1 paragraph
- Week by week timeline: What are you going to do each week?

Available CUDA libraries

- cuBLAS: dense linear algebra
- cuSPARSE: sparse linear algebra
- cuRAND: random numbers, good for MCMC simulations
- cuFFT: Fast Fourier Transform
- cuSOLVER: dense and sparse factorization and system solvers
- cuDNN: common operations for deep neural nets

Might be useful for project planning to to check out what they provide!

16 bit matrix multiplication

Some applications (such as deep neural nets) don't need 32 bits of precision.

16 bit advantages: speed up matrix multiply due to less IO, fit twice as much data in GPU memory

If you create a fast implementation, there's a good chance the deep learning community will use it a lot!

Cryptocurrency

Find some cryptocurrency with an proof of work algorithm that hasn’t been optimized to death, and then optimize it to death

An interesting read on the topic

Randomized Matrix Factorizations

Can quickly approximate SVD, QR decomposition, etc using randomized algorithms!

Good project for someone who has taken ACM 106a

- Method for least squares solutions, ultra-high dim'l spaces that represent highly constrained systems
- Can compare performance to cuSOLVER, for instance, for different size and different types of problems
- Good survey paper on the subject (PDF)

Branch and Bound Systems

Make global B&B solution-finding environment.

An N-dimensional box-like “volume” is tested for a criterion.

- If the box passes the test, it is put into a list of “solution boxes.”
- If the box does not pass, it is subdivided into children boxes which are then tested. Method finds all such boxes.
- The criterion/test will be run once per box, suitable to run on GPU
- Can include octree or K-D tree algorithms for representing surfaces and solids, for instance, http://www.nvidia.com/docs/IO/88889/laine2010i3d_paper.pdf or
- N-body gravitational systems that subdivide space with one body per box http://www.cs.nyu.edu/courses/spring12/CSCI-GA.3033-012/nbody-problem.pdf

Interval Analysis “Corner Form”

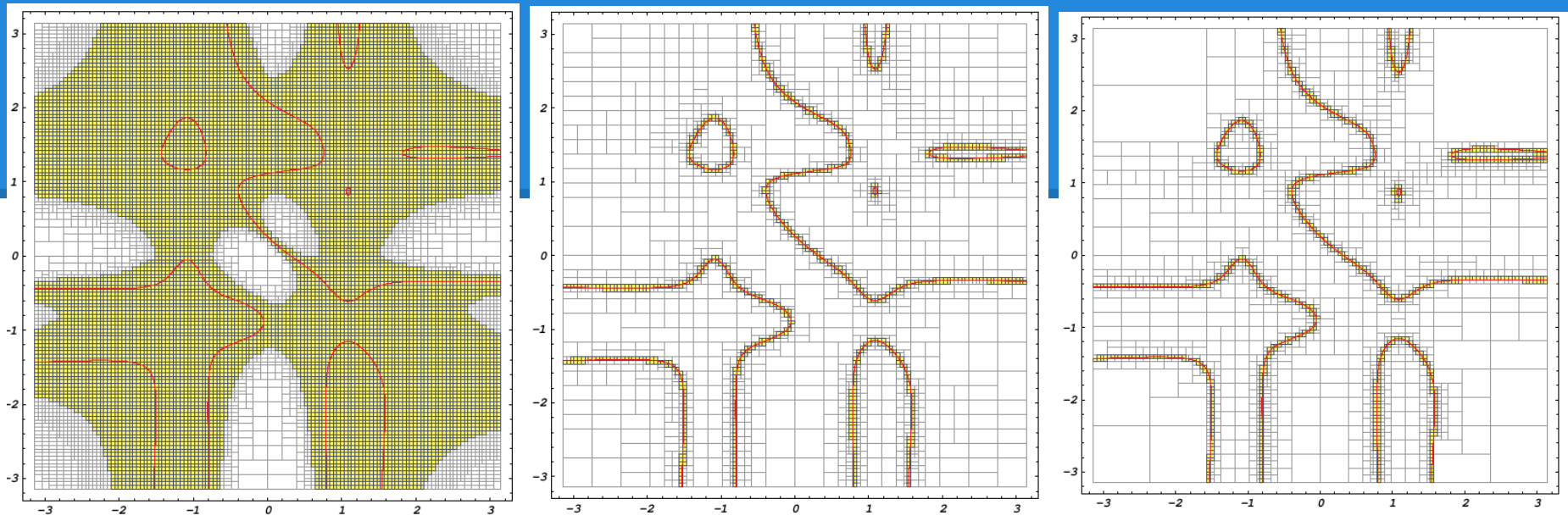
A method for global root finding -- an interval-based B&B test that guarantees that a “box” does not contain a root of

$$f(x,y,z,w) = 0$$

where f() is a polynomial in x, y, z, w (or more variables).

- Intervals of f, given input intervals, are computed by “inclusion function”
- The Corner Taylor Form (inclusion function) is more accurate than the Midpoint Taylor Form for large input regions, eliminating many boxes early in the process.

See http://thesis.library.caltech.edu/view/author/Gavriliu-M.html



Finding global roots of cos2x sin3y + sin3x cos2y - cos2x cos3y + sin3xsin2y = 0

1. (First turn function into polynomial, with error term). Sol'ns in yellow.
2. On left, "Natural Extension" inclusion function: too many potential solutions.
3. In the middle, "Centered Form" is much improved:
4. On right, "Corner Form" is even better. Note upper right corner. Large region excluded quickly, without need for further subdivision.

Hash Table (and/or malloc)

Implement a concurrent hash table that lives in global or shared memory.

Implement malloc for global or shared memory.

These will be tricky parallel programming problems!

Build an assembler

Reverse engineer Fermi or Kepler binaries and build an assembler.

Already done for Maxwell: https://github.com/NervanaSystems/maxas

“Speed dating”

Talk to another person about your ideas for 3 minutes.

Will cycle several times.

Goals: hear lots of ideas, connect people with similar ideas, offer suggestions to each other ideas