OLD DOMINION UNIVERSITY

IDEA FUSION

Research and Education in Accelerator Physics at Old Dominion University

Balša Terzić, PhD

Department of Physics, Old Dominion University

Center for Accelerator Science, Old Dominion University

James Madison University, Department of Physics, April 2, 2015

Outline

- Accelerator Physics Education at Old Dominion University
  - Center for Accelerator Science (CAS)
  - Research Experience for Undergraduates (REU) at ODU

- Accelerator Physics Research at Old Dominion University
  - Research directions within CAS
  - Computational accelerator physics
  - Outline a few projects
    - Optimization using Genetic Algorithms
    - Computation on Graphical Processing Units (GPUs)

- Summary

Accelerator Physics Education at ODU

- Interdisciplinary Center for Accelerator Science (CAS) at ODU (http://www.odu.edu/cas)
  - Founded in 2008 as an umbrella center for interdisciplinary approach to solving accelerator physics problems (note “Science” not only “Physics”)
  - Members from several departments
    - Physics (6 professors), computer science (2), engineering…
  - Capitalizes on the proximity of Jefferson Lab
    - 3 Jefferson Lab Professors (Jefferson Lab staff; spend 30% of time at CAS)
    - Accelerator physics students carry out their research at Jefferson Lab
  - Current numbers:
    - 13 graduate students
    - 2 postdocs
    - Graduated 7 PhDs in physics and 3 PhDs in engineering

Accelerator Physics Education at ODU

- Research Experience for Undergraduates (REU) at ODU (https://www.jlab.org/accel/reu/)
  - Highly selective, NSF-sponsored, 10-week paid summer program
    - Starts at the end of May, ends at the end of July
    - Students are housed on Jefferson Lab's campus
    - Weekly lectures on various research topics
    - Guidance provided on scientific writing and presentation
    - Many, free "mandatory fun" events are planned
  - Students are supervised by ODU professors or Jefferson Lab staff
  - At the end, students write a research paper and present posters
    - They are often chosen to present their work at national conferences
  - Current numbers (since 2008):
    - 53 students (15 female); 32 did research in accelerator physics

Accelerator Physics Education at ODU

- Summer Undergraduate Laboratory Internship (SULI) at Jefferson Lab (http://education.jlab.org/suli/)
  - Highly selective, DoE-sponsored, 10-week paid summer program
    - Starts at the end of May, ends at the end of July
    - Students are housed on Jefferson Lab's campus
    - Weekly lectures on various research topics
    - Guidance provided on scientific writing and presentation
    - Many, free "mandatory fun" events are planned
  - Virtually identical to REU, but only Jefferson Lab staff and users can serve as student mentors
    - Not a problem: virtually all of ODU physics professors are either Jefferson Lab staff or users (all of CAS members)
  - At the end, students write a research paper and present posters
    - They are often chosen to present their work at national conferences

Accelerator Physics Research at ODU

- Detailed description of accelerator research projects: http://www.odu.edu/cas
- Superconducting radio-frequency (SRF) accelerating structures (Professor Jean Delayen, CAS Director)
- Novel materials for future superconducting cavities (Professor Alex Gurevich)
- Plasma processing of superconducting structures (Professors Vušković and Popović)
- Accelerator design: Energy-recovering linacs, electron-ion colliders, light sources, energy-recovering linacs (Professors Krafft, Satogata,…)
- And others…

My Accelerator Physics Research at ODU

- New computational tools:
  - New methods
  - New computational hardware

- New methods:
  Multidimensional, nonlinear optimization using genetic algorithms (GA)
  - Brief motivation and background
  - Applications in accelerator physics

- New computational hardware:
  Parallel computation on Graphical Processing Units (GPUs)
  - Brief motivation and background
  - Applications in accelerator physics

Why Computations?

- Any scientific field can benefit from computations
  - *Experimental sciences*: data processing, model validation
  - *Theoretical sciences*: simulate physical processes, model validation
  - *Discovery science*: e.g. Lorenz's (re-)discovery of chaos in 1970's
- New computer architectures resolve old computational bottlenecks
  - Present state-of-the-art unfathomable even 5-10 years ago:
    - Codes now can utilize on the order of *millions of processors*
    - Particle simulations: *1 simulation particle = 1 electron in a bunch*
  - Relax approximations/simplifications → closer to the physics problem
  - *What once was computationally prohibitive it is now possible*
- Accelerator physics critically relies on computations for
  - *Validate new concepts*: no study without it is taken seriously
  - *Performance optimization*

Computations in Accelerator Science

- Prodigious increase in computational power
  → Relaxing simplifying approximations (i.e., 1D → 2D → 3D)
    → More trustworthy computer simulations
- Cannot be a "one-trick pony"
  - State-of-the-art computations require *all of these*:
    - Fundamental understanding of underlying physics
    - Utilization of (new) advanced mathematical techniques
    - Computational expertise (including newest computational platforms)
- Computations in accelerator science *must be interdisciplinary*
  - Utilize field experts in physics, computer science, math, engineering…
  - Center for Accelerator Science (CAS) proposal (2007), 1st paragraph:

    "We propose an interdisciplinary research and teaching center for accelerator science and technology. It would be unique in Virginia and one of only a handful of such programs in the country. Since accelerator science is inherently interdisciplinary, the center would be a source of innovation in pure and applied science, which is likely to engender spin-off industry and add to the university's capacity for generating patents."

GA Optimization: Motivation

- Multidimensional non-linear optimization becomes more challenging/ impossible as the dimensionality of the problem increases
  - Traditional, gradient-based methods (Newton, conjugate-gradient, steepest descent, etc…) are *not globally convergent*:
    - May get stuck in a local minimum and never come out
    - Final solution depends on the initial guess
    - Generally *not robust* in the non-linear regime
    - Direct multi-objective optimization not possible

- This demonstrates a clear need for *globally-convergent, robust, multidimensional, multi-objective, non-linear optimization* methods
  - Genetic Algorithm (GA) fills this need
    - Trade-off: not as efficient as traditional methods

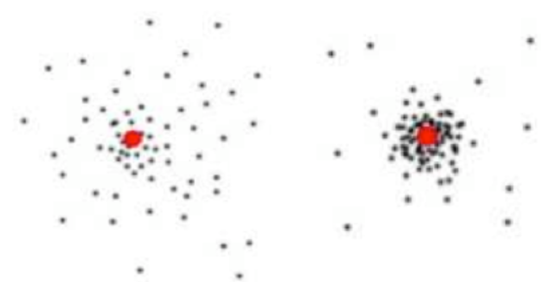
GA Optimization: Background

- GA uses principles of natural selection to solve an optimization problem

Evolution	Multidimensional optimization
Gene	Variable
Individual	Point in search space
Population	Set of points in search space
Mutation	Changing variables
Swap	Exchange of values of the same variable between two points in search space
Recombination (partial swap)	Change of values of the same variable between two points toward each other
Fitness	Value of the objective function

- Mutation
  - Similar to random walk
    Given by a pdf $P_m(\eta_{mut})$

η_{mut}=1



η_{mut}=10

- Recombination
  - Given by a pdf $P_r(\eta_{rec})$

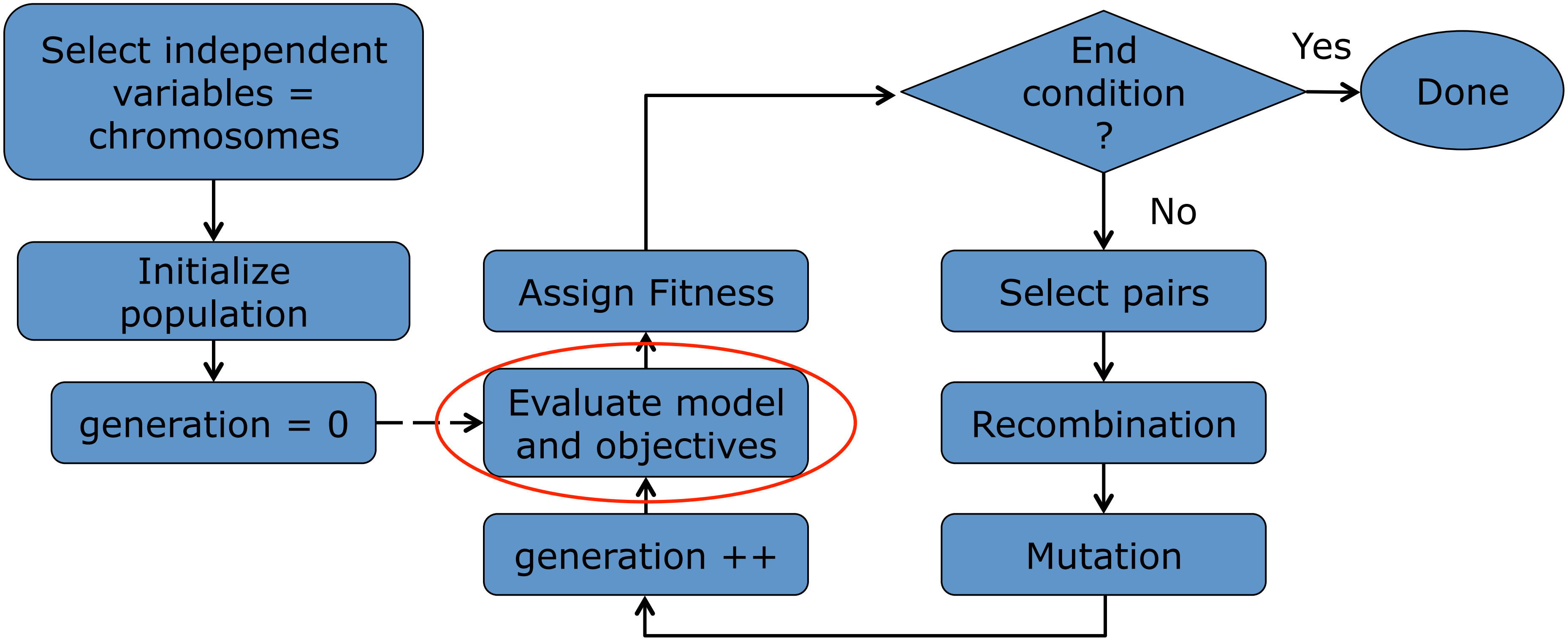
η_{rec}=1

η_{rec}=10

[Hofler, Terzić, Kramer, Zvezdin, Morozov, Roblin, Lin & Jarvis 2013, *Phys. Rev. ST AB* 16, 010101]

Students' names underlined

GA Optimization: Background



Minimize $f_m(\mathbf{x}),\quad m = 1,2,\ldots,M;$ ← objectives

Subject to $g_j(\mathbf{x}) \geq 0,\quad j = 1,2,\ldots,J;$ ← inequality constraints

$h_k(\mathbf{x}) = 0,\quad k = 1,2,\ldots,K;$ ← equality constraints

$x_i^{(L)} \leq x_i \leq x_i^{(U)},\quad i = 1,2,\ldots,n;$ ← decision variable constraints

GA Optimization: Applications

- We applied GA optimization to many problems in accelerator physics:

Single objective

  - Beam diagnostics (wire scanner fits)
    [REU Projects: Henderson 2013, Gabriele 2014]
  - Optimizing particle collider working point for luminosity

Multiple objectives

  - Maximizing dynamic aperture in a particle collider ring
  - Decoupling of the beam optics in the injector
  - Optimizing dynamic aperture and chromaticity in a collider ring
  - RF gun optimization for injector brightness
    [Hofler, Terzić, Kramer, Zvezdin, Morozov, Roblin, Lin & Jarvis 2013, *Phys. Rev. ST AB* 16, 010101]
  - Optimizing laser frequency modulation function in Thomson scattering
    [Terzić, Deitrick, Hofler & Krafft 2014, *Phys. Rev. Lett.*, 112, 074801]
  - Optimizing cavity heat load and trip rates in CEBAF linacs at Jefferson Lab
    [Terzić, Hofler, Reeves, Khan, Krafft, Benesch, Freyberger & Ranjan 2014, *Phys. Rev. ST AB* 17, 101003]

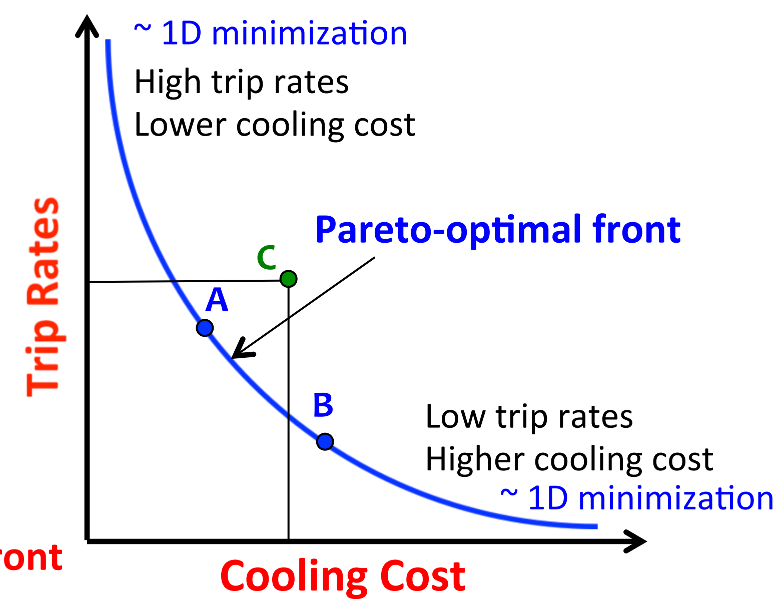
- Real applications for real machines
  – CEBAF and the proposed Medium-energy Electron Ion Collider (MEIC) at Jefferson Lab, *but not limited to these*

GA Application: Optimizing Cavity Heat Load and Trip Rates in the CEBAF Linacs

- What is the optimal configuration of cavity gradients needed to *maximize the science and minimize the cost of operation* (electricity bill)?
  - Monthly electricity bill for JLab is measured in millions of dollars – a large part of it is CEBAF cryogenics
    Even modest improvements in cooling may translate into millions in savings
  - Cooling (cavity heat load) and interrupted operation time (trip rates) are *competing objectives* – multi-objective optimization problem
- The goal here:
  - Provide a set of feasible solutions showing the *trade-offs between competing objectives*
- Asymptotic behavior provided by 1D minimization using Lagrange multipliers

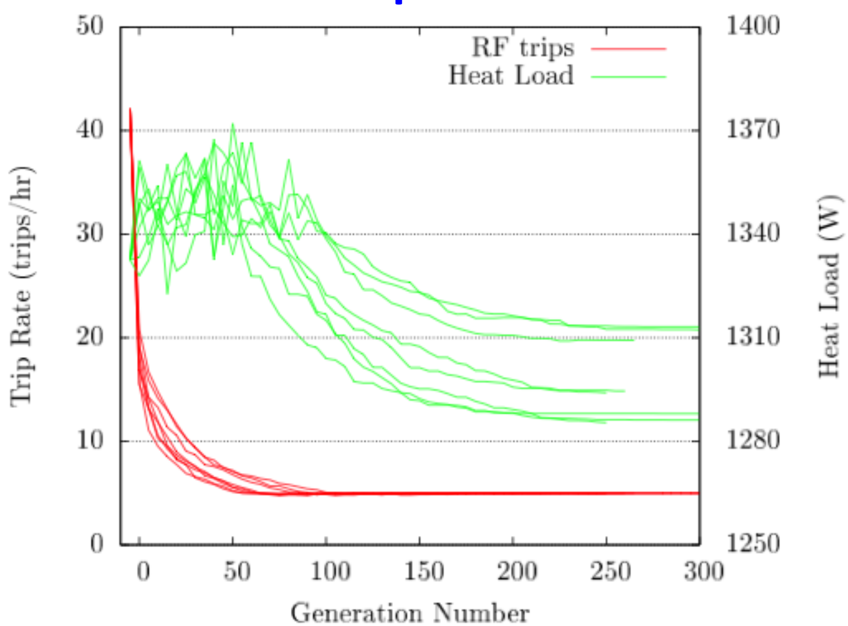
[Terzić, Hofler, Reeves, Khan, Krafft, Benesch, Freyberger & Ranjan 2014, *Phys. Rev. ST AB* 17, 101003]

A *dominates* C: C is not on the Pareto-optimal front



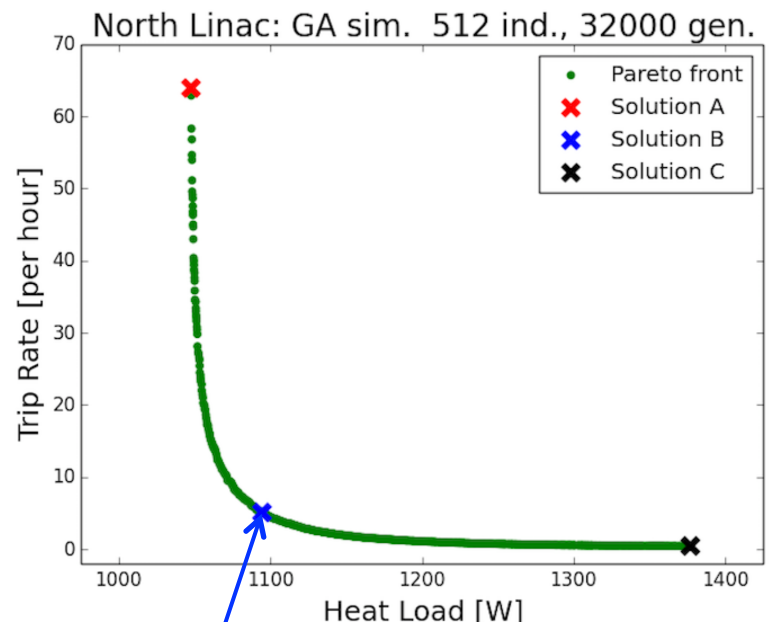
GA Application: Optimizing Cavity Heat Load and Trip Rates in the CEBAF Linacs

Previous optimization



Trip Rate = 5
Minimum heat load 1285 W

Our Study



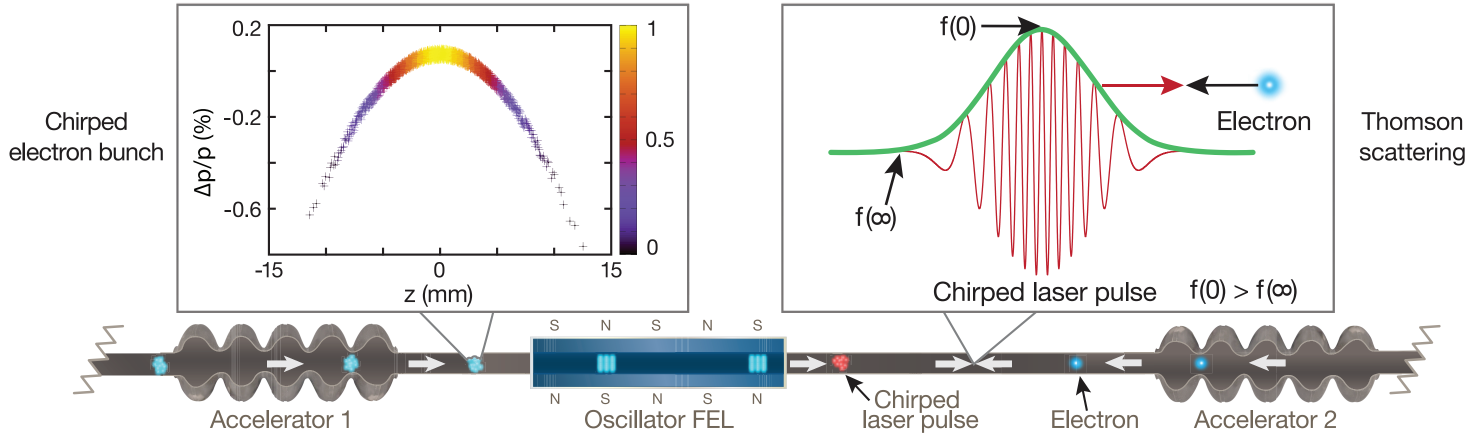
Trip Rate = 5
Minimum heat load 1094 W
(4% from the minimum of 1048 W @ A)

Reduced heat load by 15%

(Savings exceed my salary many times over!)

[Terzić, Hofler, Reeves, Khan, Krafft, Benesch, Freyberger & Ranjan 2014, *Phys. Rev. ST AB* 17, 101003]

GA Application: Narrow-Band Emission in Thomson Scattering

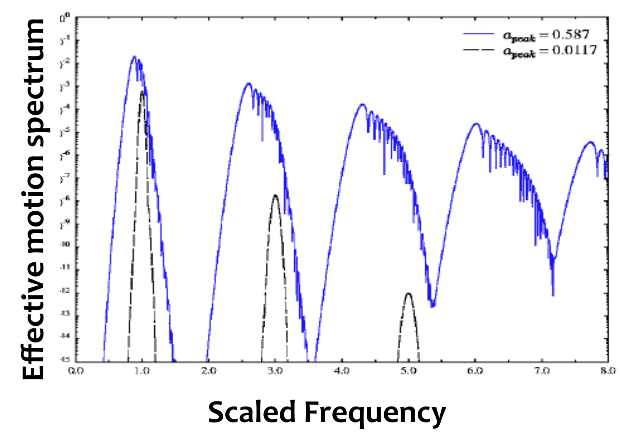


[Terzić, Deitrick, Hofler & Krafft 2014, *Phys. Rev. Lett.*, *Phys. Rev. Lett.*, 112, 074801, Fig. 1]

- Thomson scattering:
  - Classical regime: no electron recoil, no quantum effects
  - Factor of $4\gamma^2$ increase in energy

$$E_{\text{radiation}} = \gamma^2(1+\beta)^2 E_{\text{laser}} \approx 4\gamma^2 E_{\text{laser}}$$

  - Constant-frequency laser produces *broadened spectra* in high-field regime [Krafft 2004, *Phys. Rev. Lett.* 92, 204802]
- Can a judicious laser frequency modulation ("chirp") lead to narrowing of the spectra? We believed so.
- Enter GAs



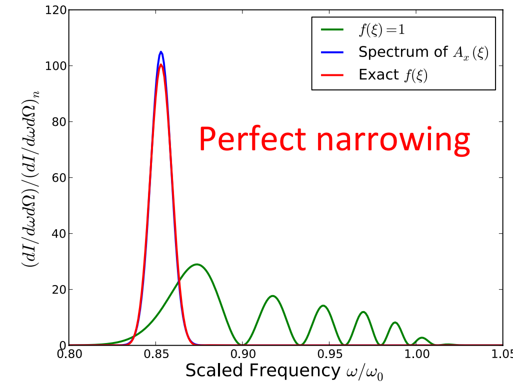
[Krafft 2004, Fig. 2]

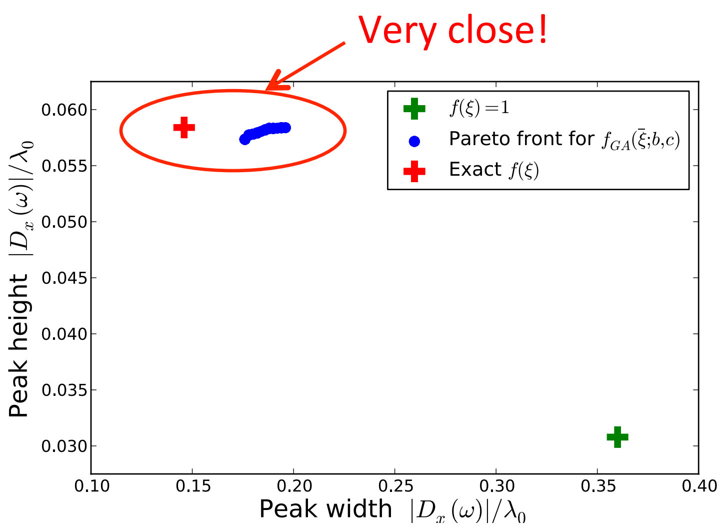
low field

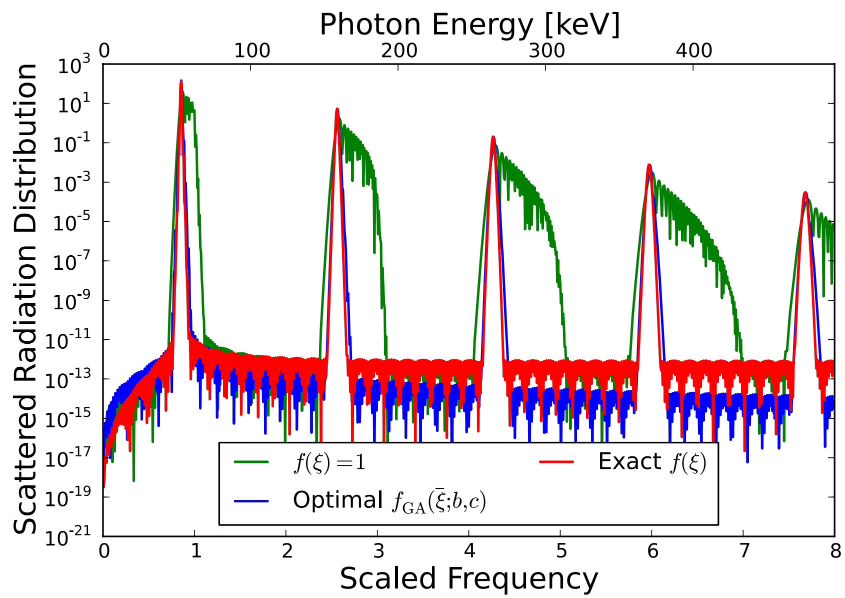
high field

GA Application: Narrow-Band Emission in Thomson Scattering

- After some heuristic investigation (*educated guessing!*), we settled on a two-parameter modulation function: $f_{GA}(\bar{\xi}; b, c) = c / \left[1 - (1-c)\exp\left(-b\bar{\xi}^2\right)\right]$
- We set up a GA optimization which
  - Maximize the height of the main peak
  - Minimize the width at 10^{-6}







GA optimization provided vital clues about the shape of the modulation function which was later found exactly and analytically

[Terzić, Deitrick, Hofler & Krafft 2014, *Phys. Rev. Lett.*, *Phys. Rev. Lett.*, 112, 074801]

GPU Computation

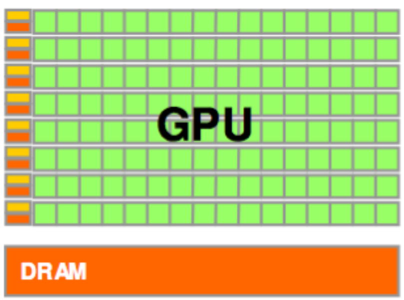
- *Why is it important?*
  - Making simulations much more efficient computationally (through GPUs) *enables studying previously inaccessible physics*
- *What are we doing that is new and different?*
  - Interdisciplinary approach – division of labor among experts in the field:
    - Physicists: physics, algorithm development, prototyping
    - Computer scientists: algorithm development and implementation, parallel programming
- *What are our goals?*
  - Develop GPU-parallelized *state-of-the-art accelerator physics codes*
  - Design methods useful *beyond the scope of accelerator physics*
  - Develop expertise useful on other problems and other architectures

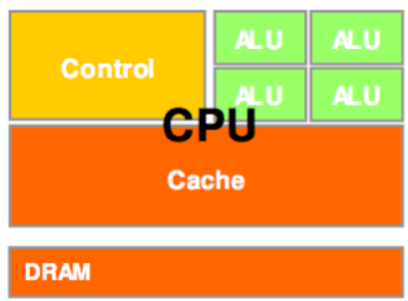
GPU Computation: Motivation

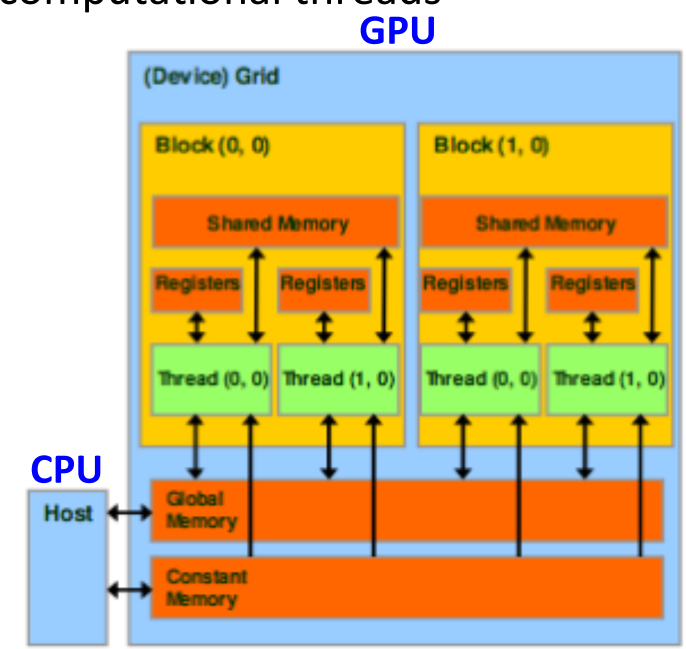
- There are many problems in accelerator physics that can *greatly benefit* from a speedup from a GPU-based computation
  - Particle tracking codes
  - Beam collision codes
  - Monte Carlo-based codes
- *Speedup*: ratio of execution times on a host CPU to that on a GPU
- Some have already been GPU-parallelized with impressive speedup of about 20 - 70 times
- In general, if a problem is inherently parallelizable, an implementation on GPUs can improve performance by 1-3 orders of magnitude
- This kind of speedup means:
  - Simulation time: several months or a year → about a day
  - *Opening the doors to studying previously inaccessible physics!*

GPU Computation: Background

- Parallel computation on GPUs
  - Ideally suited for algorithms with *high arithmetic operation/memory access ratio*
  - Same Instruction Multiple Data (SIMD)
  - *Several types of memories* with varying access times (global, shared, registers)
  - Uses extension to existing programming languages to handle new architecture
  - GPUs have many smaller cores (~400-2500) designed for parallel execution
  - *Avoid branching and communication* between computational threads







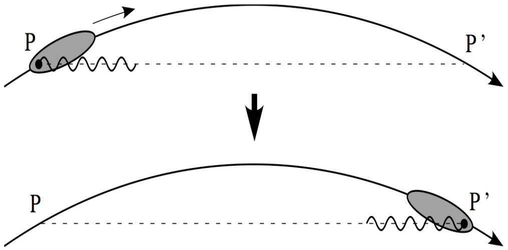
More space for ALU, less for cache and flow control

GPU: grid → blocks → threads

Example: Tesla M2090 GPU has 512 cores

GPU-Based Simulations of Electron Beams

- Physical Problem
  - When electron beams are bent, they radiate
  - The radiation "catches up" and adversely affects the beam (breaks it up and renders useless)
  - Numerical simulation difficult and computationally prohibitive because of integration over beam's history



- Importance
  - Dynamics of electrons in most electron machines
  - Better simulations lead to better understanding of electron beam's dynamics and mitigation of the unwanted effects
- Our contribution
  - Designed a new adaptive multidimensional integration algorithm optimized for GPUs [Arumugam, Godunov, Ranjan, Terzić & Zubair 2013a, b]
    - Useful beyond this project
  - Achieved over 3 orders of magnitude speedup over a serial code [Terzić, Arumugam, Godunov, Ranjan & Zubair 2015, Phys. Rev. ST AB, *in preparation*]

Summary

- Strong accelerator physics educational program at ODU
  - CAS, REU, SULI @ Jefferson Lab
  - Students involved in cutting-edge research
    - Publish, go to conferences in exotic locations, graduate, get good jobs!

- Computational Accelerator Physics Research at ODU
  - Interdisciplinary collaboration at CAS
  - High-performance computations
    - Using GAs to optimize performance and design of accelerators
    - Parallel computations on GPUs

- We are always on a lookout for hard-working, motivated students, so if you are interested, please get in touch!

More info: http://www.odu.edu/cas http://www.odu.edu/physics
http://www.odu.edu/~bterzic
bterzic@odu.edu

My Interdisciplinary Collaborators

Center for Accelerator Science (CAS) at Old Dominion University (ODU):

Professors:

Physics: Alexander Godunov
Computer Science: Mohammad Zubair, Desh Ranjan

PhD students:

Physics: Kirsten Deitrick
Computer Science: Kamesh Arumugam, Sabbir Khan, Mohamed Aturban

Undergraduate students:

Physics: Mark Stefani, Marvin Munoz

Jefferson Lab (Newport News):

Accelerator Division:

Geoff Krafft, Alicia Hofler, Vasiliy Morozov, Fanglei Lin, He Zhang, Yves Roblin, Jay Benesch, Arne Freyberger

Nuclear Theory Group:

Wally Melnitchouk

Undergraduate Summer Interns (REU and SULI programs) (7 since 2010)

Colin Jarvis, Matt Kramer, Anton Zolotor, Alyssa Henderson, Cody Reeves, Victoria Gabriele, Todd Hodges

Details at http://www.odu.edu/~bterzic

Refereed Publications:

- Terzić, <u>Deitrick</u>, Hofler & Krafft 2014, *Phys. Rev. Lett.*, 112, 074801
- Terzić, Hofler, <u>Reeves</u>, <u>Khan</u>, Krafft, Benesch, Freyberger & Ranjan 2013, *Phys. Rev. ST AB* 16, 010101
- Hofler, Terzić, <u>Kramer</u>, <u>Zvezdin</u>, Morozov, Roblin, Lin & <u>Jarvis</u> 2013, *Phys. Rev. ST AB* 16, 010101
- <u>Arumugam</u>, Godunov, Ranjan, Terzić & Zubair 2013a, *International Conference on Parallel Processing – 42nd Annual Conference* (refereed)
- <u>Arumugam</u>, Godunov, Ranjan, Terzić & Zubair 2013b, *20th Annual International Conference on High-Performance Computing* (refereed)
- Terzić & Bassi 2011, *Phys. Rev. ST AB* 14, 070701

Conference and Other Contributions:

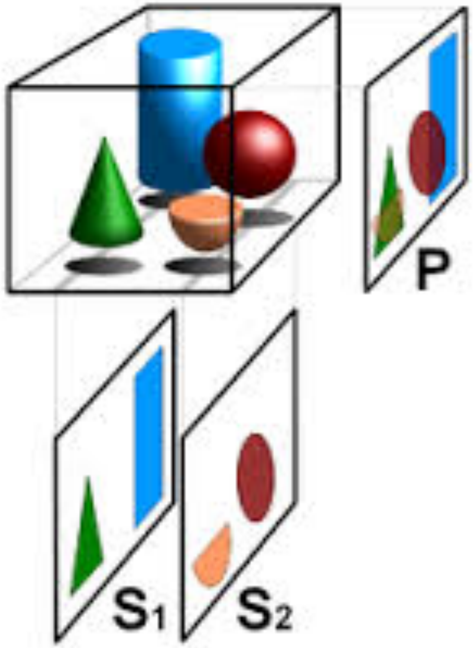
- <u>Arumugam</u>, Godunov, Ranjan, Terzić & Zubair 2013, GPU Tech conference
- <u>Henderson</u>, Terzić & Hofler 2013, REU (@ODU) project
- Roblin, Morozov, Terzić, <u>Aturban</u>, Ranjan & Zubair 2013, *International Particle Accelerator Conference (*MOPWO080)
- Terzić, <u>Kramer</u> & <u>Jarvis</u> 2011, Particle Accelerator Conference (*WEP167*)
- <u>Kramer</u>, <u>Jarvis</u> & Terzić 2010, JLab Tech Note JLAB-TN-10-034

Backup Slides

GPU-Based Particle Collider Simulations

- Physical Problem
  - Simulate long-term behavior of colliding beams in a collider
  - Colliding beams disturb each other slightly during each collision
  - One hour of collider operation is on the order of billion collisions!
  - New, efficient algorithms and architectures are needed
- Importance
  - Dynamics of electrons in most electron machines
  - Better simulations lead to better understanding of collider's long-term dynamics and mitigation of the unwanted (resonant) effects
- Our contribution
  - Designed a new GPU-optimized particle tracking algorithm
    [Arumugam, Godunov, Ranjan, Terzić & Zubair 2015, *in preparation*]
    - Useful beyond this project
  - Implementing a tracking + collision code
    [Terzić et al. 2015, Phys. Rev. ST AB, *in preparation*]

Tomography



- Physical Problem
  - Recover 2D/3D shape from a set of 1D projections
- Importance and Applications
  - Accelerator physics (beam diagnostics)
  - Plasma physics, medical physics, astrophysics
- What Needs to Be Done
  - Effects of noise in experiments (noise removal – wavelets)
  - Quantify the accuracy of reconstructed image vs. number of projections
- Student Skills Developed
  - Mathematical physics (integral equations, integration methods…)
  - Computational physics (all work is done on computers)

Wavelet Denoising and Compression

- When the signal is known, one can compute *Signal-to-Noise Ratio* (*SNR*):

$$SNR = \sqrt{N_{ppc}}$$

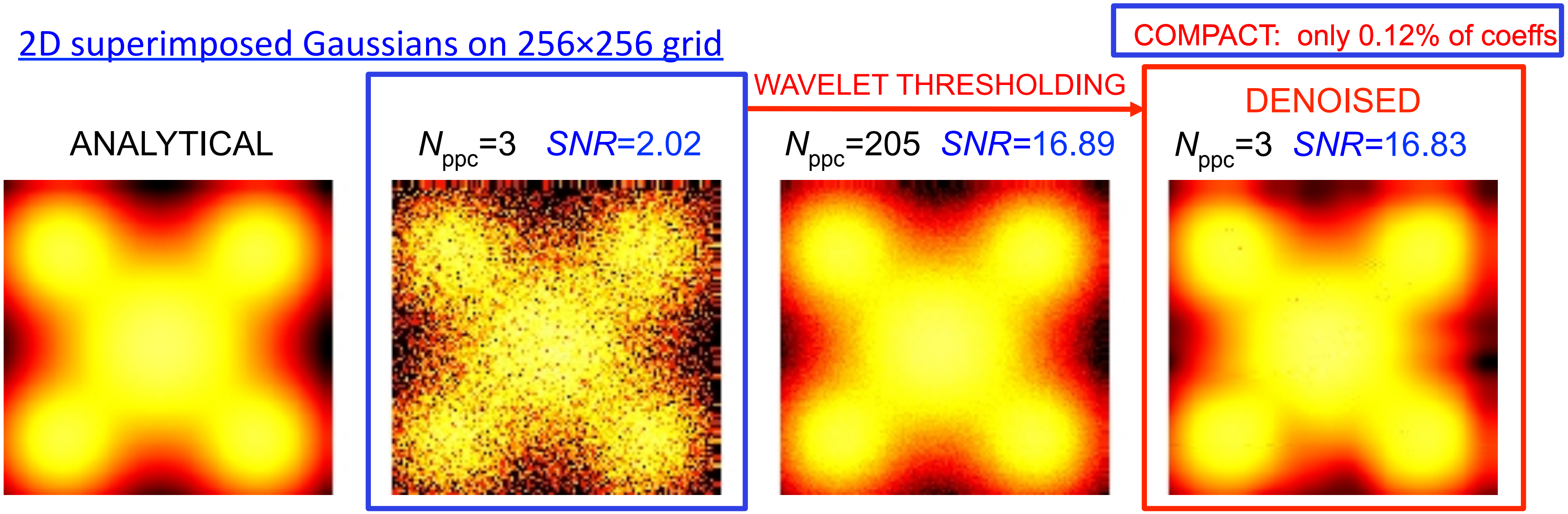
N_{ppc}: avg. # of particles per cell $\quad N_{ppc} = N/N_{cells}$

$$SNR = \sqrt{\frac{\sum_{i=1}^{N_{grid}} \bar{q}_i^2}{\sum_{i=1}^{N_{grid}} \left(q_i - \bar{q}_i\right)^2}}$$

$\bar{q}_i$ *exact*

q_i *grid*

2D superimposed Gaussians on 256×256 grid



Wavelet denoising yields a representation which is:

- Appreciably more accurate than non-denoised representation
- Sparse (if clever, we can translate this sparsity into computational efficiency)

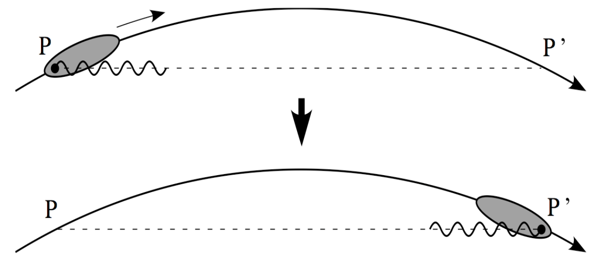
Parallel Computation on GPUs

- The largest resolution tested so far is 128x128
- 1 step of the simulation on a 128x128 grid and 32 GPUs: ~ 10 s
- Execution time *reduces* as the number of macropraticles grows

Number of Particles (N)	Grid Resolution	Sequential Time(sec.)	Single GPU Time (sec.)	Single GPU Speedup	32 GPUs Time (sec.)	32 GPUs Speedup
102400	32×32	145.52	1.48	98	1.29	113
	64×64	1736.24	16.78	104	1.13	1537
	128×128	27049.30	256.85	105	13.88	1950
1024000	32×32	121.41	1.30	93	1.23	99
	64×64	1140.15	11.12	103	1.75	652
	128×128	15153.60	144.03	105	11.78	1287
4096000	32×32	119.73	1.29	93	1.23	97
	64×64	939.96	9.19	102	1.74	540
	128×128	10654.00	101.37	105	9.33	1142

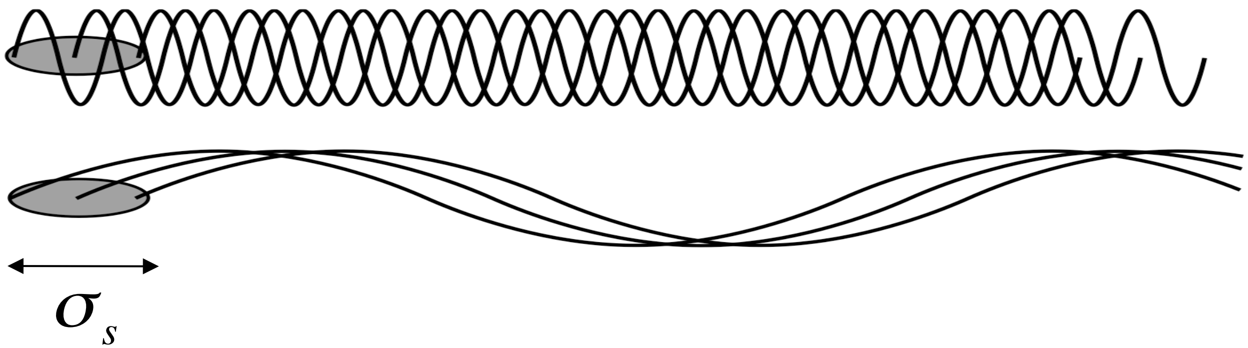
GPU Computation: CSR Simulations

- When an electron beam travels along a curved trajectory (bending magnet), it emits synchrotron radiation



Incoherent (ISR)
$\lambda < \sigma_s$

Coherent (CSR)
$\lambda > \sigma_s$



largely cancels out

has systematic effects

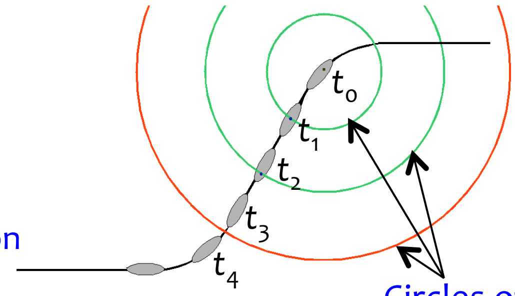
- CSR adversely impact beam quality:
  - Increased energy spread and emittance, longitudinal instability (microbunching)
- CSR effects are important for machines which bend electrons (FELs, light sources, ERLs, electron colliders, etc…)
  - JLab FEL, LCLS-II, NSLS-II, ALS, Fermi@ELETTRA…
- It is of vital importance to have a trustworthy code to simulate and mitigate the CSR effects

GPU Computation: CSR Simulations

- CSR simulations have proven to be extremely challenging
  - Computing *retarded potentials* requires integration over the retarded time *t'*:

$$t' = t - \frac{|\vec{r} - \vec{r}\,'|}{c} \qquad \begin{bmatrix} \phi(\vec{r},t) \\ \vec{A}(\vec{r},t) \end{bmatrix} = \int \begin{bmatrix} \rho(\vec{r}\,',t') \\ \vec{J}(\vec{r}\,',t') \end{bmatrix} \frac{d\vec{r}\,'}{|\vec{r} - \vec{r}\,'|}$$

Retarded time — Retarded potentials — Charge & current distribution



  - *Huge computational bottleneck!*
  - For a particle-in-cell (PIC) CSR code the computations scale as ~ N_{res}^2 (N_{res} is the grid resolution)
- Solution: Develop an efficient, parallel multidimensional integrator on GPUs
  - Integration over grid is ideally suited for GPU parallelization (SIMD)
  - Used NVIDIA CUDA framework (extension to C++)
  - *Deterministic*: based on integration rules like Gauss or Newton – not Monte Carlo
  - Useful beyond this project: *outperforms Monte Carlo in medium/high dimensions*

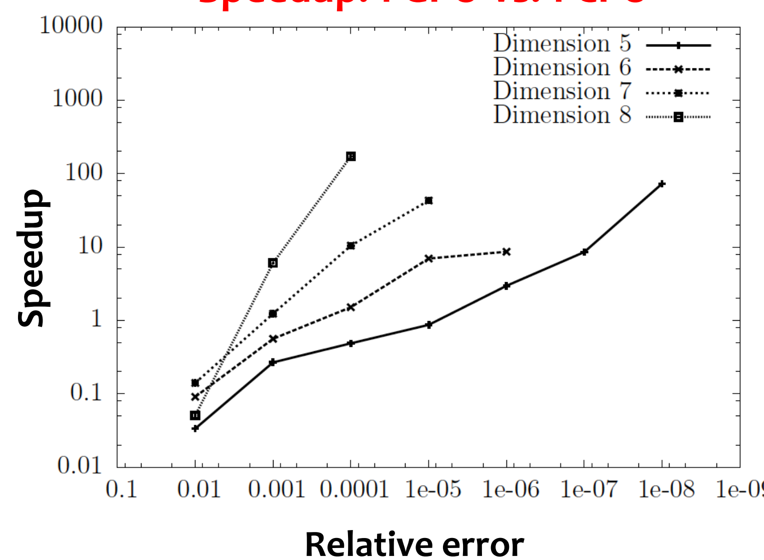
Adaptive Multidimensional Integration On a Single GPUs

- Direct parallelization of the serial methods *does not take advantage of GPU data parallelism* and *does not provide load balancing* → *inefficient code*
- *We developed a new two-phase parallel algorithm multidimensional integration on GPUs*
  - *Phase 1: Parallel identification of subintervals needing higher resolution*
  - *Phase 2: Parallel evaluation of identified sub-regions to prescribed accuracy*
- GPU-based implementation outperforms the best known sequential method (CUHRE) and achieve up to 10-100 times speedup on a single GPU

Benchmark functions in *n* dimensions
oscillatory, strongly peaked and of varying scales

1. $f_1(\mathbf{x}) = \left[\alpha + \cos^2\left(\sum_{i=1}^{n} x_i^2\right)\right]^{-2}$, where $\alpha = 0.1$
2. $f_2(\mathbf{x}) = \cos\left(\prod_{i=1}^{n} \cos\left(2^{2i} x_i\right)\right)$
3. $f_3(\mathbf{x}) = \sin\left(\prod_{i=1}^{n} i \arcsin(x_i^i)\right)$
4. $f_4(\mathbf{x}) = \sin\left(\prod_{i=1}^{n} \arcsin(x_i)\right)$
5. $f_5(\mathbf{x}) = \frac{1}{2\beta} \sum_{i=1}^{n} \cos(\alpha x_i)$, where $\alpha = 10.0$ and $\beta = -0.054402111088937$

Speedup: 1 GPU Vs. 1 CPU



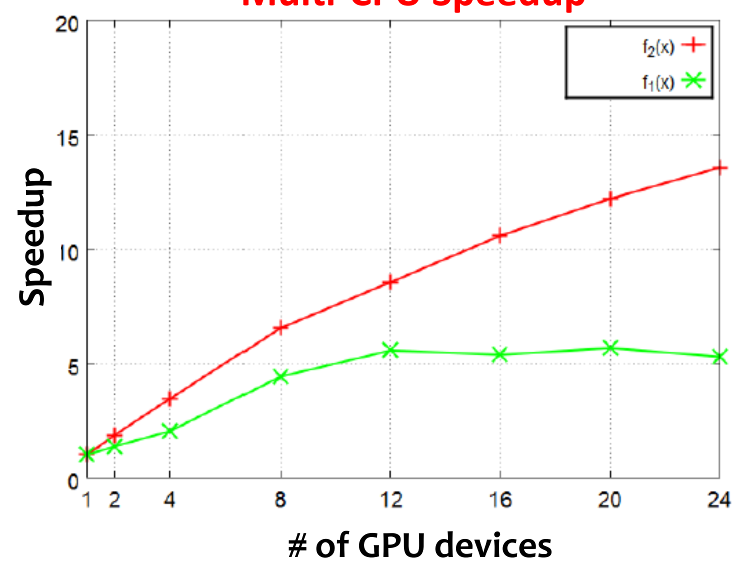
[Arumugam, Godunov, Ranjan, Terzić & Zubair 2013a]

Adaptive Multidimensional Integration On Multiple GPUs

- Next, we *optimized* our new GPU-based algorithm *for memory efficiency* and *scaled to multiple GPU devices*
- The algorithm has been implemented on a cluster of Intel® Xeon® CPU X5650 computes nodes with 4 Tesla M2090 GPU devices per node (512 cores per device)
- Memory optimization on a single GPU earned us another factor of 3.5 (speedup increased from 70 to 240)
- Scaling up to 24 GPU devices earned us another factor of 13.5 (speedup increased from 240 to 3250)

1 GPU: With Vs. Without Memory Optimization

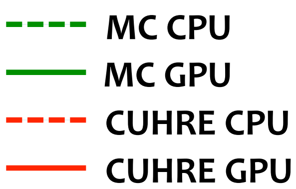
Multi-CPU Speedup



[Arumugam, Godunov, Ranjan, Terzić & Zubair 2013a, 2013b]

Monte Carlo Vs. Adaptive Multidimensional Integration With a Single GPU

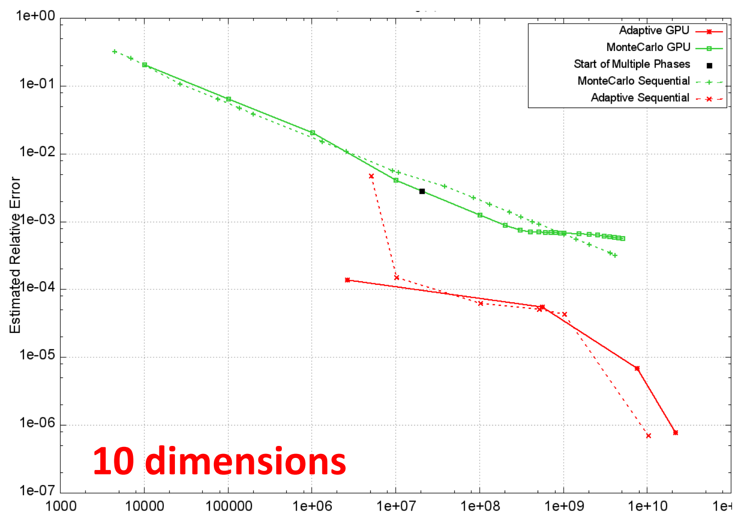
- Monte Carlo integration on GPU (VEGAS and BASES methods) has been published previously in *The European Physical Journal C* [Kanazaki 2011, 71:1559]
- We compare Monte Carlo Vs. our method on a set of *6 functions with exact solutions*
- *Preliminary results*: Even in higher dimensions our adaptive multidimensional integration method outperforms Monte Carlo method on a single GPU
- *Preliminary results*: Monte Carlo on GPU fails for large number of function evaluations
- *Possible ramifications*: Our new code can replace Monte Carlo in many physics application for improved performance



$$f_5(x) = \frac{1}{2\beta}\sum_{i=1}^{n}\cos(\alpha x_i)$$

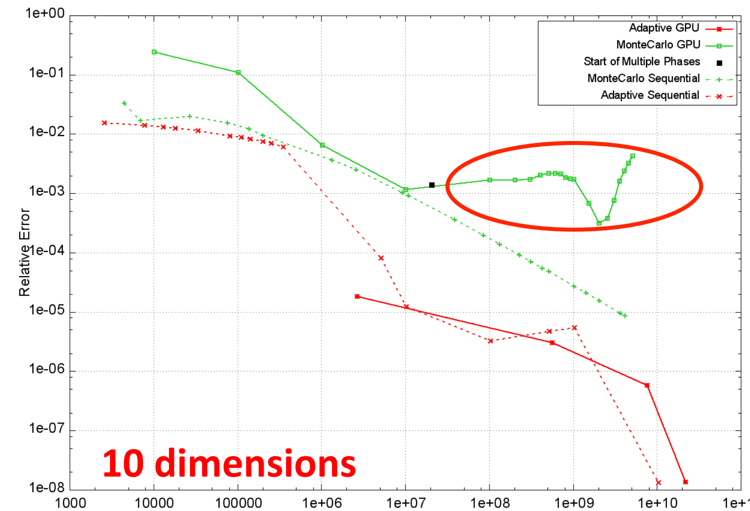
where α = 10.0 and β = -0.0544

Internal Relative Error



\# of function evaluations

True Relative Error



\# of function evaluations

[Arumugam, Godunov, Ranjan, Terzić & Zubair 2014]

GA Application: Optimizing Collider Working Point

- As particles circulate in collider rings, they oscillate around design orbits in both *x* and *y* transverse directions: *betatron oscillations*
- Collider luminosity is sensitive to beam-beam effect and betatron resonances of the two colliding beams
- Careful selection of a tune *working point* is essential for stable operation of a collider as well as for achieving high luminosity
- Simulate the proposed Medium-energy Electron-Ion Collider (MEIC) at JLab
- *Optimization problem*:
  - Independent variables: betatron tunes for the two beams $(\nu^1_x, \nu^1_y, \nu^2_x, \nu^2_y)$
  - Objective function: collider's luminosity $L(\nu^1_x, \nu^1_y, \nu^2_x, \nu^2_y)$
    (Evaluated via a simulation with *BeamBeam3D* parallel code on the JLab cluster)
  - Subject to constraints (e.g., confine tunes to particular regions)
- GA is the only non-linear optimization method that can work in a search space so violently fraught with resonances (*very* sharp peaks and valleys)

GA Application: Optimizing Collider Working Point

- Resonances occur when $m_x \nu_x + m_y \nu_y = n$
  m_x, m_y, and n are integers
- Green lines: *difference* resonances (stable)
- Black lines: *sum* resonances (unstable)
- Restrict search to a group of small regions along the diagonal devoid of black resonance lines. Restricts the search space by ~30 in 2D, ~1000 in 4D
- Found an excellent working point near half-integer resonance
  - e-beam: ν_x = 0.530, ν_y = 0.548
  - p-beam: ν_x = 0.501, ν_y = 0.527
- Luminosity about 33% above design value in only ~300 simulations (5 gen. of 64 individuals)
- Systematic scan with a modest 0.01 resolution: $100^4 = 10^8$ simulations!

  ➔ GA search orders of magnitude more efficient
- *This is just a proof of principle* – future realistic simulations will include other important effects: magnet errors, non-linear maps, IBS, cooling …