Development of GPU-based Monte Carlo Simulation Packages for Radiotherapy

Xun Jia
xunjia@ucsd.edu

5/23/11

UC San Diego
Radiation Oncology

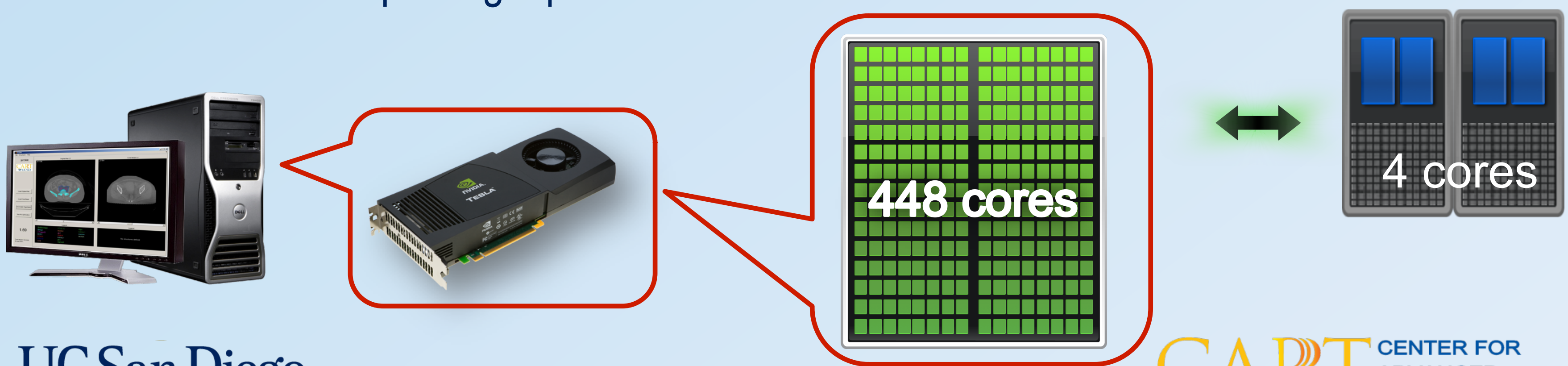
CART UCSD
CENTER FOR ADVANCED RADIOTHERAPY TECHNOLOGIES

Outline

- Introduction to GPU
- gDPM
  - Motivations
  - Approaches
  - Results
- gCTD/gMCDRR
  - Motivations
  - Approaches
  - Results
- Conclusions

Monte Carlo on GPU

- Speed up MC simulations for radiotherapy on GPU
- Graphics Processing Unit
  - Turn your PC into a supercomputer
    - Tesla C2050
    - 448 processors
    - 575 MHz clock speed
    - 3 GB memory
    - >1Tflops single precision



UC San Diego
RADIATION ONCOLOGY

CART UCSD
CENTER FOR ADVANCED RADIOTHERAPY TECHNOLOGIES

GPU details

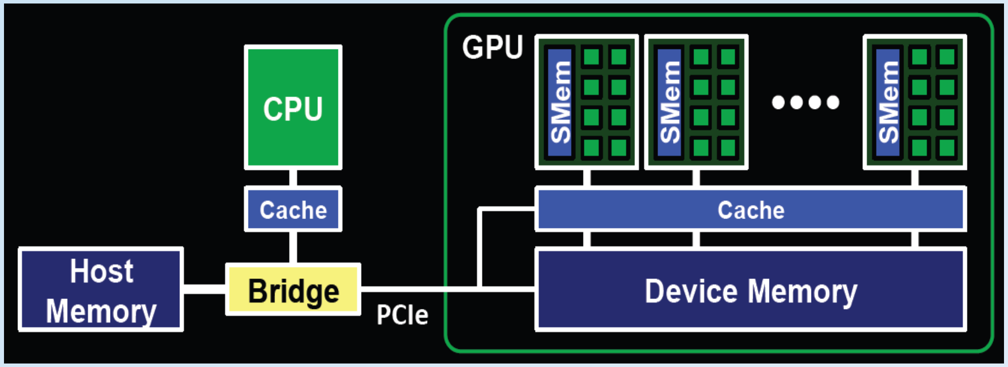
NVIDIA

GPU card	GeForce 9500 GT	GeForce GTX 480	Tesla C1060	Tesla S1070	Tesla C2050
Price ($)	~50	~400	~1,400	~8,000	~2,500
Memory (GB)	1	1.5	4	16	3
Computing power (Gflops)	134	1344	936	4147	1288
# of Processors	32	240	240	960	448

CUDA Programming

- Compute Unified Device Architecture
  - Enable us to program GPU via standard programming languages such as C
- An essential conflict between GPU architecture and MC simulation
  - Single Instruction Multiple Data (SIMD)
  - Branching problem in MC simulation
- Optimize memory usage



gDPM project

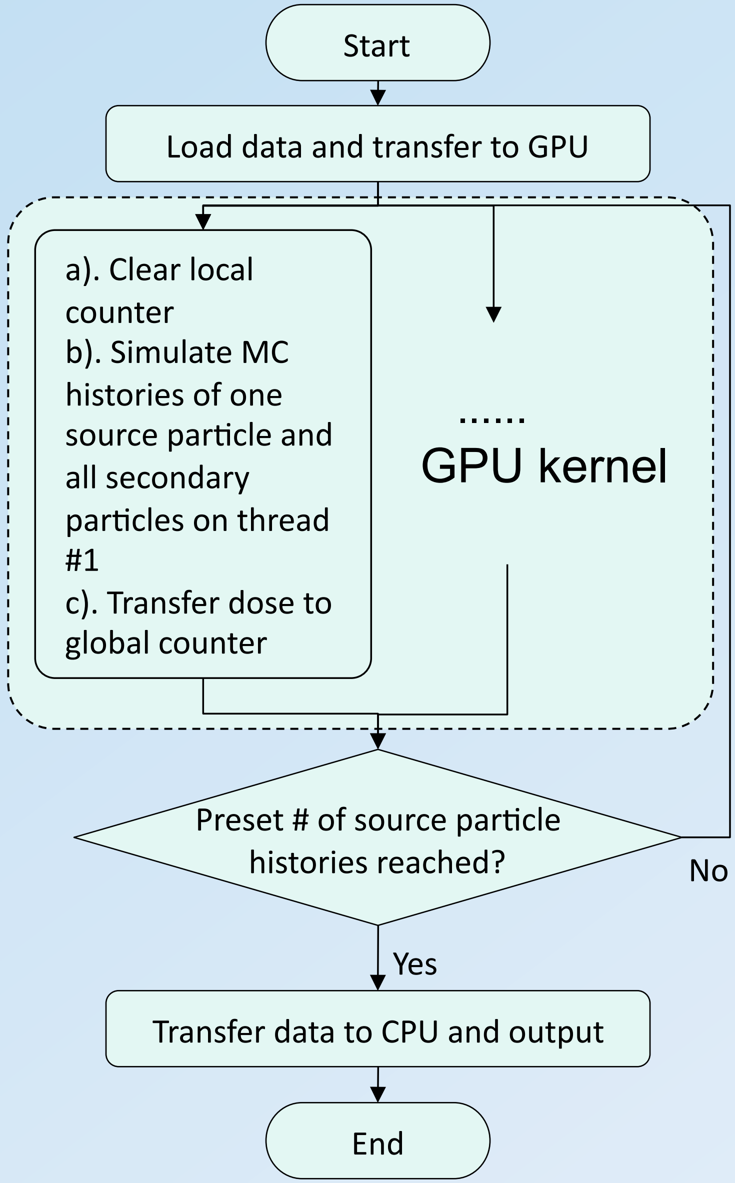
- Speed up full MC MV dose calculation using GPU
- Dose Planning Method

  --- Sempau *et al*, *Phys. Med. Biol*., 45, 2263(2000)

  - Designed for radiotherapy simulation
  - Fast compared to other general purposed MC packages
  - Relatively simple simulation process --- easy to program
- Key idea
  - Same physics as in DPM
  - Maintain computation accuracy
  - Obtain speed up by optimizing code for GPU architecture
- Approaches
  - First rewrite DPM in C
  - Write CUDA code on GPU

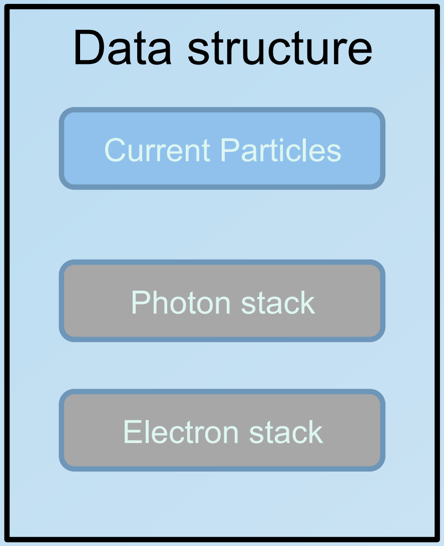
gDPM v1.0

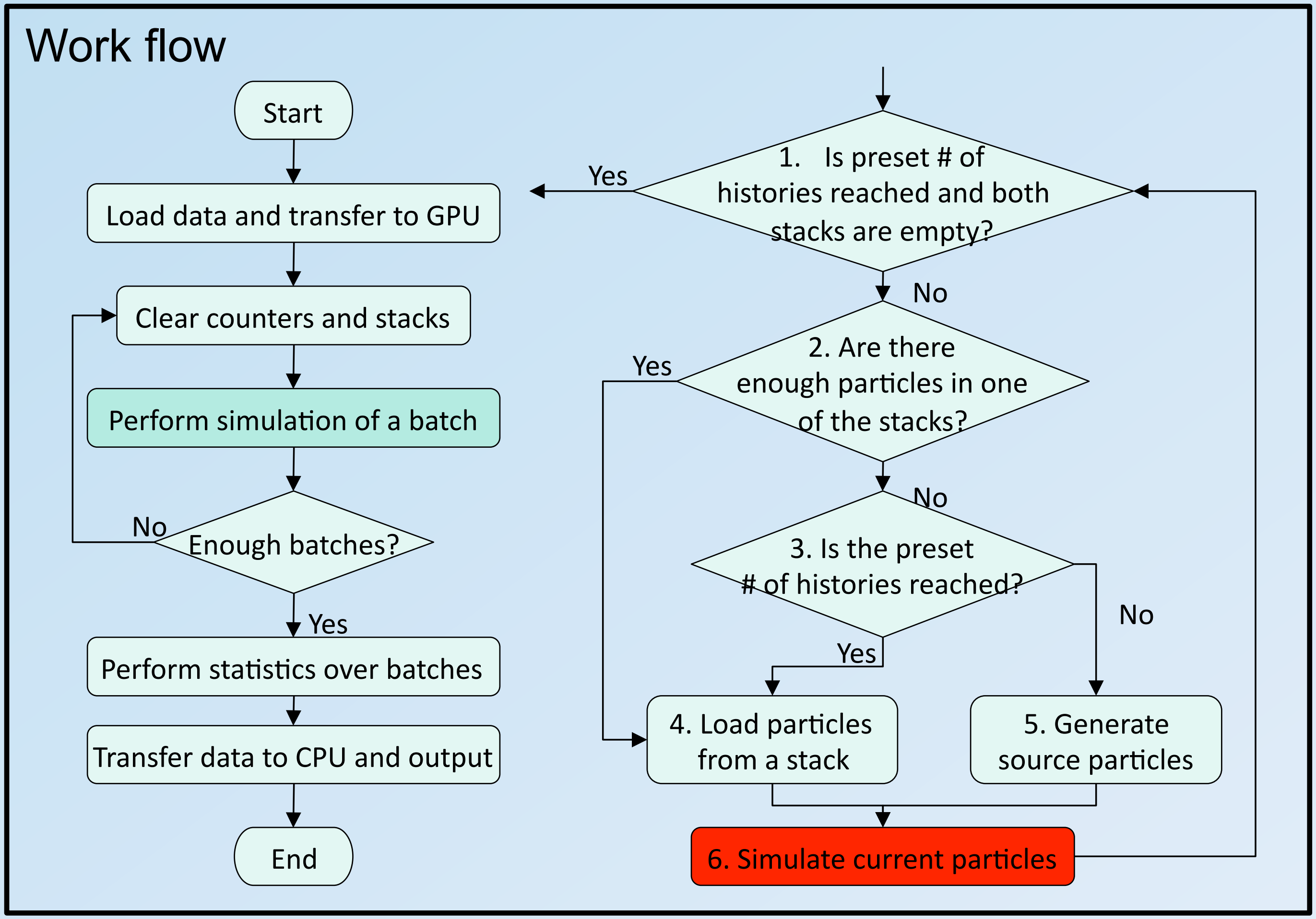
- Method
  - Treat each computational thread on a GPU as an independent computing unit
  - Multiple thread run simultaneously
- Implementation
  - Each thread keeps its own RND seed
  - Each thread tracks its own particles
  - Transfer dose deposition in all threads to a global counter at the end of GPU kernel
- Speed-up factors of about 5.0 ~ 6.6 times have been observed



gDPM v2.0

- Separate photon and electron transport





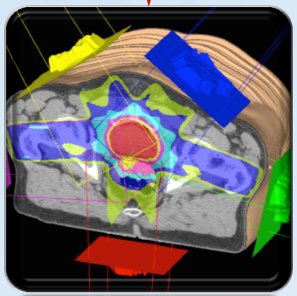
gDPM v2.0

- Improve random number generator efficiency
  - Use CURAND, a light-weight RND generator provided by NVIDIA
- Interpolation of cross section data
  - Linear interpolation is used in gDPM v2.0
    - No loss of accuracy is observed
    - GPU support hardware interpolation
- Optimize GPU memory access
  - Use shared memory

Other Components

- Load DICOM RT format to define patient anatomy (voxel materials and structure information)
- Enable gantry, couch, collimator rotations
- Flexible source function
  - User can supplement with their own realistic Linac source model or phase space file
- Enable simulating fluence map and MLC

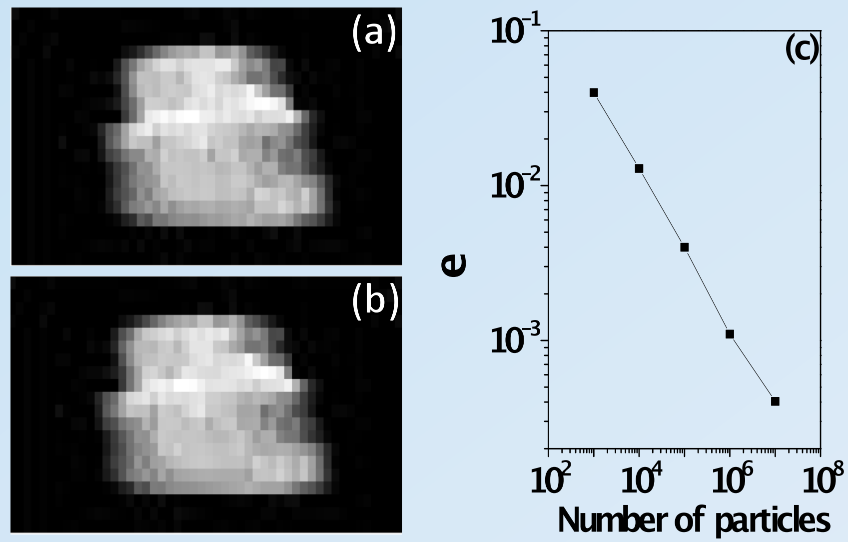
Dose caculation in realistic IMRT & VMAT treatment plans



Fluence map

- Fluence map: a set of beamlets I with associated weights p_I
- Metropolis sampling

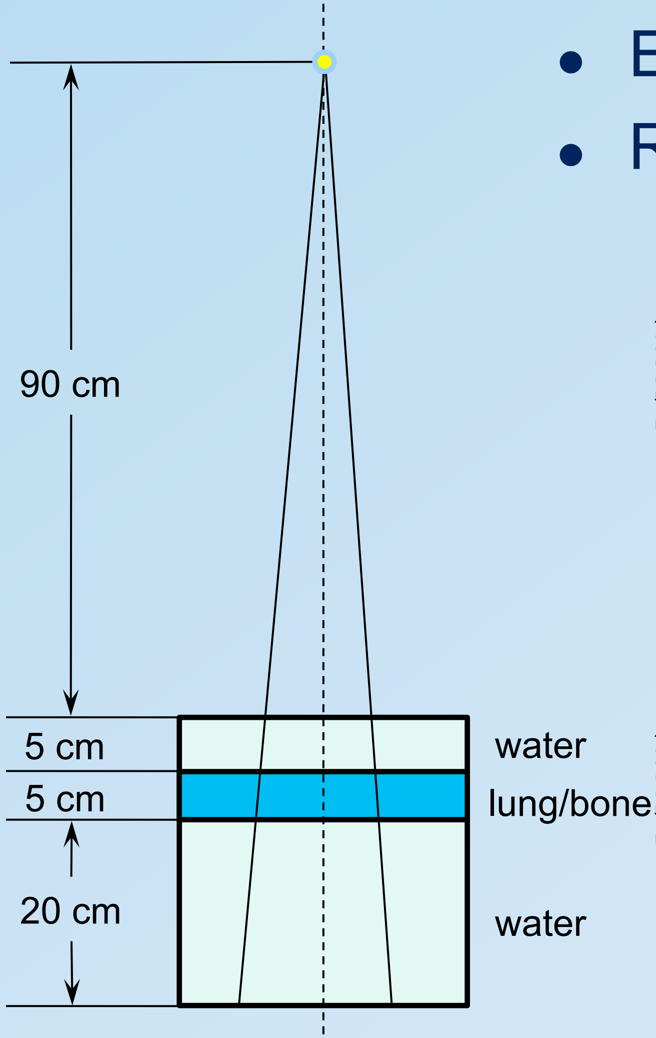
Start with a initial beamlet I_0
for $i = 1 \ldots N$
    Generate a trial beamlet J
    Generate a random number r
    if $r<p(J)/p(I_{i-1})$
        $I_i = J$
    else
        $I_i = I_{i-1}$
    endif
    Sample a particle inside the beamlet I_i uniformly
end



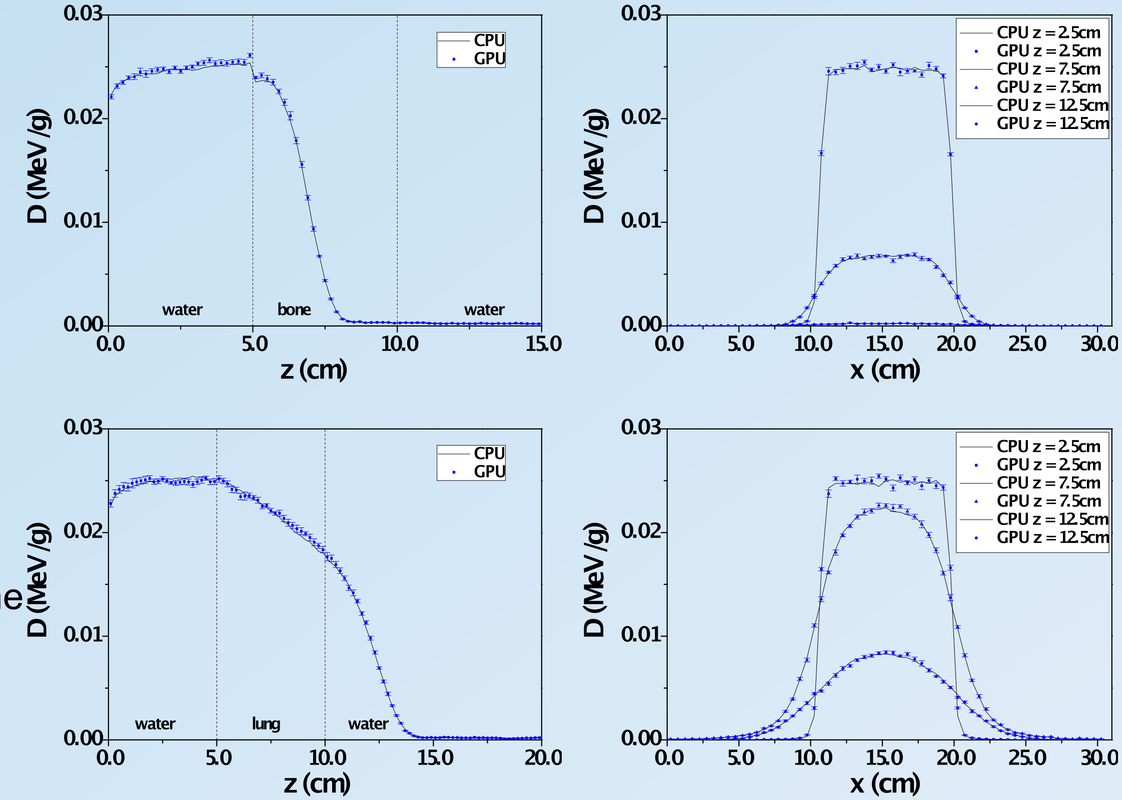
UC San Diego
RADIATION ONCOLOGY

CART UCSD
CENTER FOR ADVANCED RADIOTHERAPY TECHNOLOGIES

Electron Cases



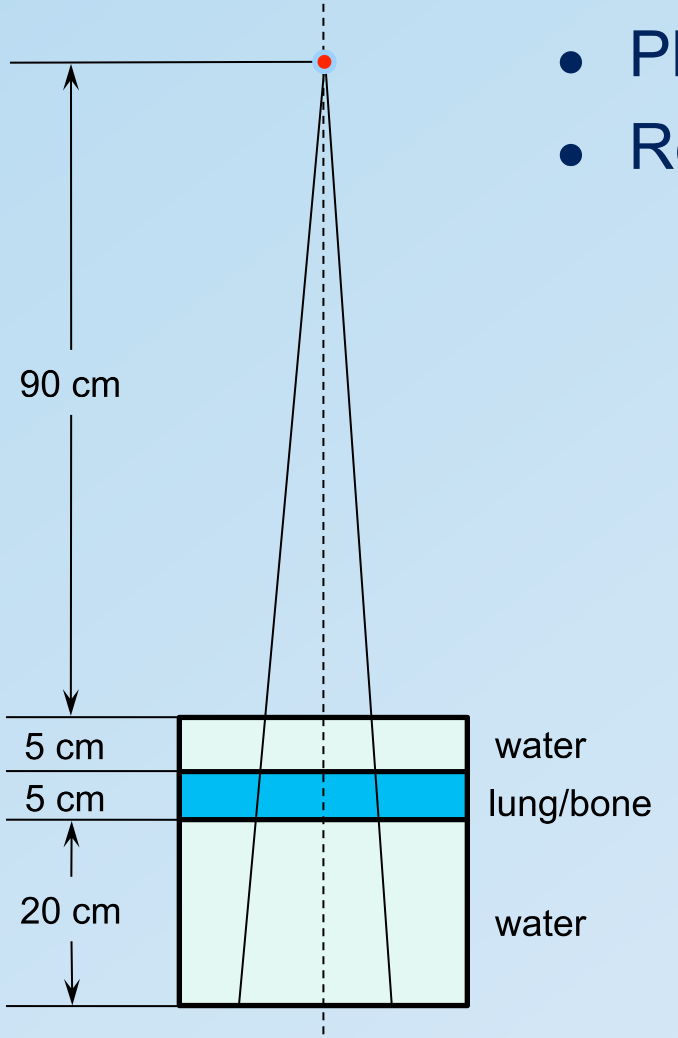
- Electron point source, 20 MeV
- Results



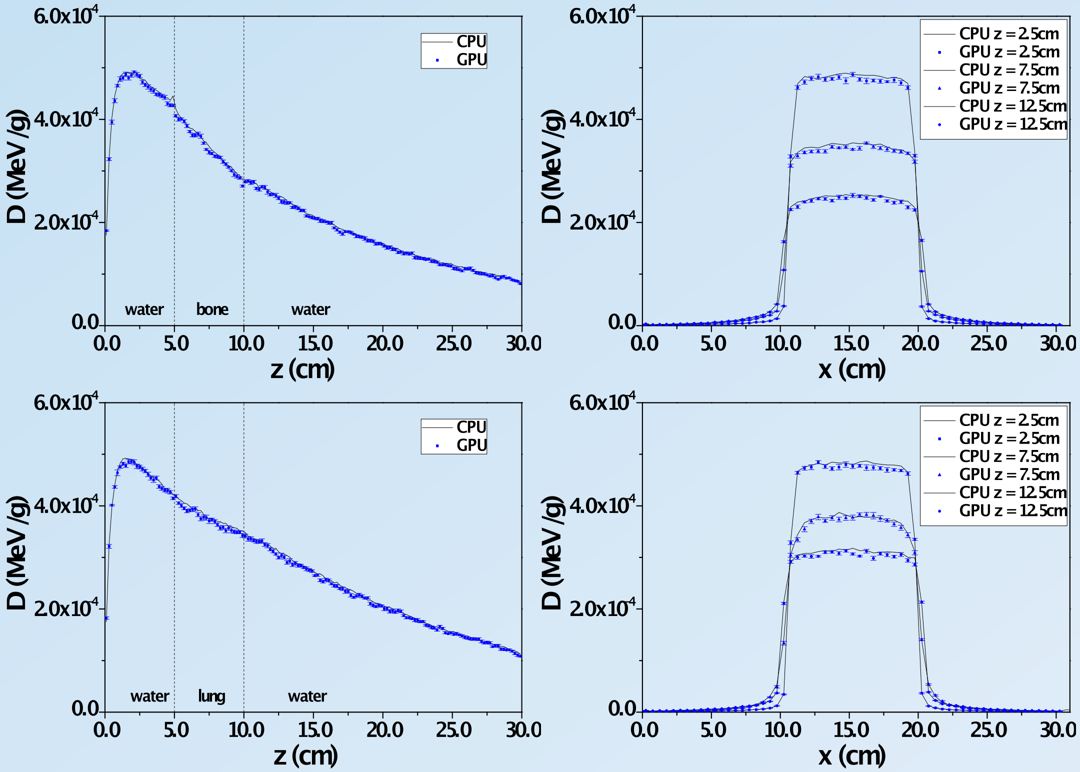
UC San Diego
RADIATION ONCOLOGY

CART UCSD CENTER FOR ADVANCED RADIOTHERAPY TECHNOLOGIES

Photon Cases



- Photon point source, 6 MV spectrum
- Results

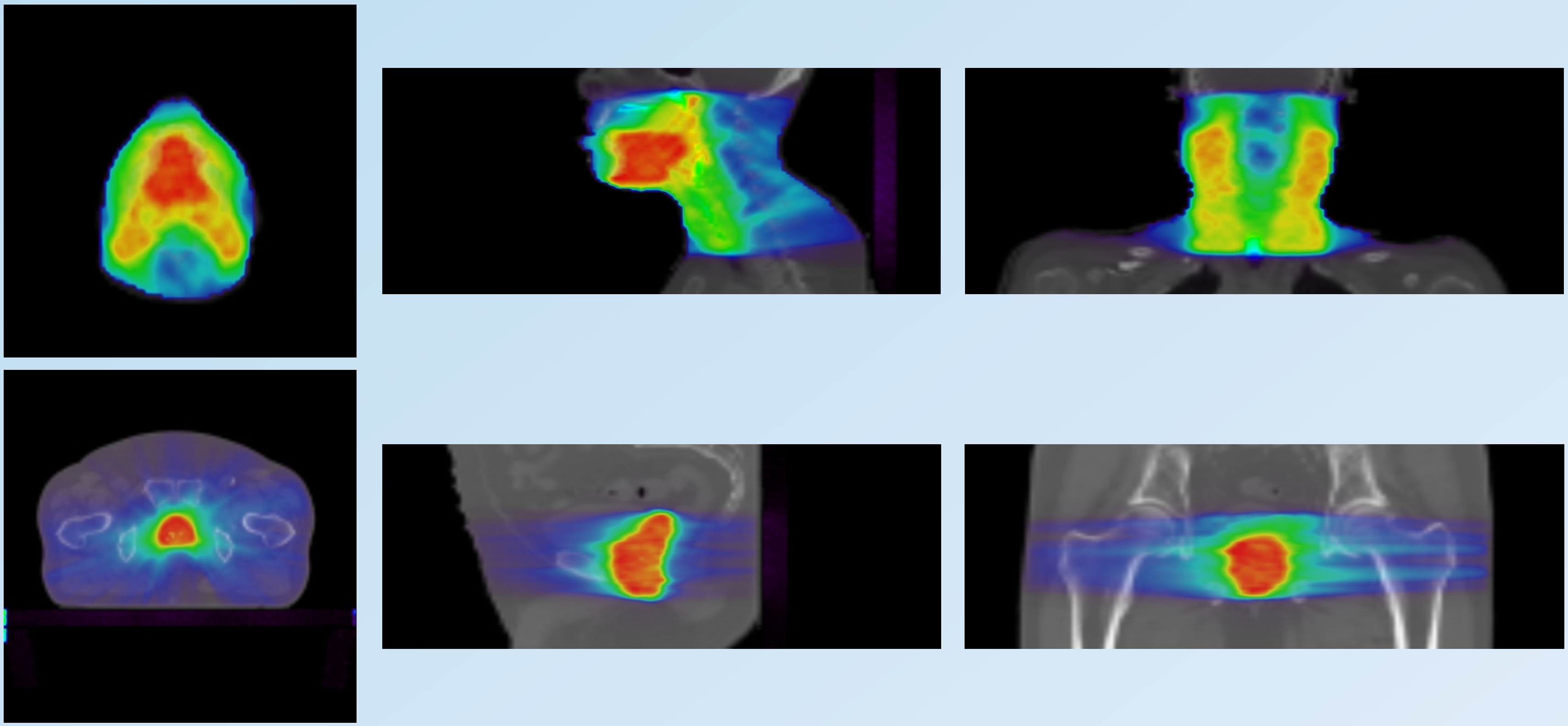


UC San Diego
RADIATION ONCOLOGY

CART UCSD CENTER FOR ADVANCED RADIOTHERAPY TECHNOLOGIES

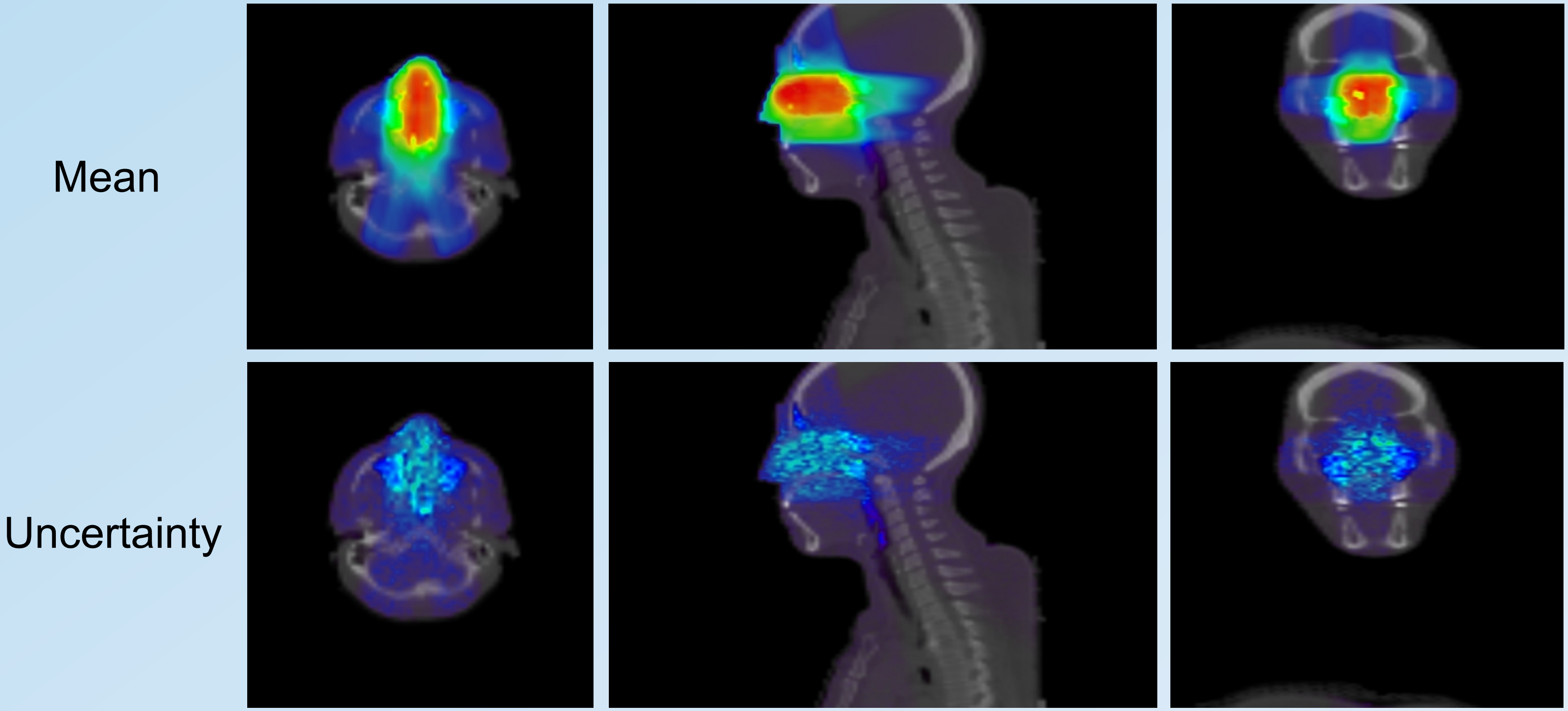
RapidArc Cases

- Photon point source, 6 MV spectrum
- 2 arcs



IMRT Case

- Photon point source, 6 MV spectrum
- 8 non-coplanar beams



Uncertainty is amplified by 50 times for clear visualization

Results

Average relative uncertainty $\langle \sigma_D/_D \rangle$ (computed in region where D > $0.5D_{max}$), Passing rate P_t.

Source type	# of Histories	Case	$\langle \sigma_D/D \rangle$ CPU (%)	$\langle \sigma_D/D \rangle$ GPU (%)	P_t (%)
20MeV Electron	2.5×10^6	water-lung-water	0.99	0.98	99.9
20MeV Electron	2.5×10^6	water-bone-water	0.98	0.99	100.0
6MV Photon	2.5×10^8	water-lung-water	0.71	0.72	98.5
6MV Photon	2.5×10^8	water-bone-water	0.64	0.64	96.9
6MV Photon	2.5×10^8	VMAT HN patient	N/A	0.88	N/A
6MV Photon	2.5×10^8	VMAT Prostate patient	N/A	0.78	N/A
6MV Photon	2.5×10^8	IMRT HN patient	N/A	0.57	N/A

CPU: Intel Xeon processor with 2.27GHz

GPU: NVIDIA Tesla C2050

Results

Execution time T, and speed-up factor T_{CPU}/T_{GPU} for four different testing cases.

Source type	# of Histories	Case	T_{CPU} (sec)	T_{GPU} (sec)	T_{CPU}/T_{GPU}
20MeV Electron	2.5×10^6	water-lung-water	117.5	**2.05**	57.3
20MeV Electron	2.5×10^6	water-bone-water	127.0	**1.97**	64.5
6MV Photon	2.5×10^8	water-lung-water	1403.7	**18.6**	75.5
6MV Photon	2.5×10^8	water-bone-water	1741.0	**24.2**	71.9
6MV Photon	2.5×10^8	VMAT HN patient	N/A	**36.5**	N/A
6MV Photon	2.5×10^8	VMAT Prostate patient	N/A	**46.7**	N/A
6MV Photon	2.5×10^8	IMRT HN patient	N/A	**48.0**	N/A

CPU: Intel Xeon processor with 2.27GHz

GPU: NVIDIA Tesla C2050

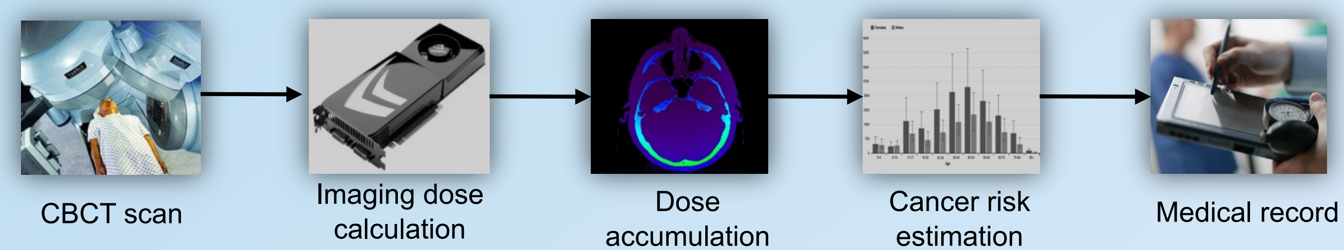
Results

- Multi-GPU implementation
  - Bash script to submit job to 4 GPUs (2 GTX590)
  - Summation and statistics are performed

Source type	# of Histories	Case	T_{GPU} (sec)	T_{4GPU} (sec)	T_{GPU}/T_{4GPU}
6MV Photon	4×10^9	water-lung-water	312.82	78.4	3.99
6MV Photon	4×10^9	water-bone-water	403.75	101.19	3.99

gCTD/gMCDRR project

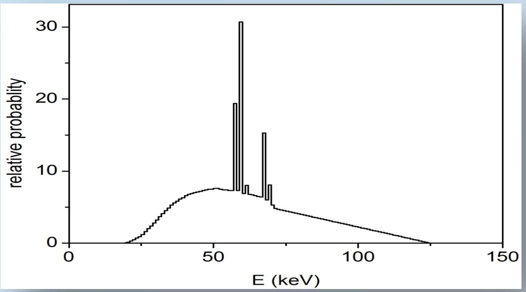
- Fast kV MC simulation for CT/CBCT scans
  - gCTD: assess radiation dose received during CT scans



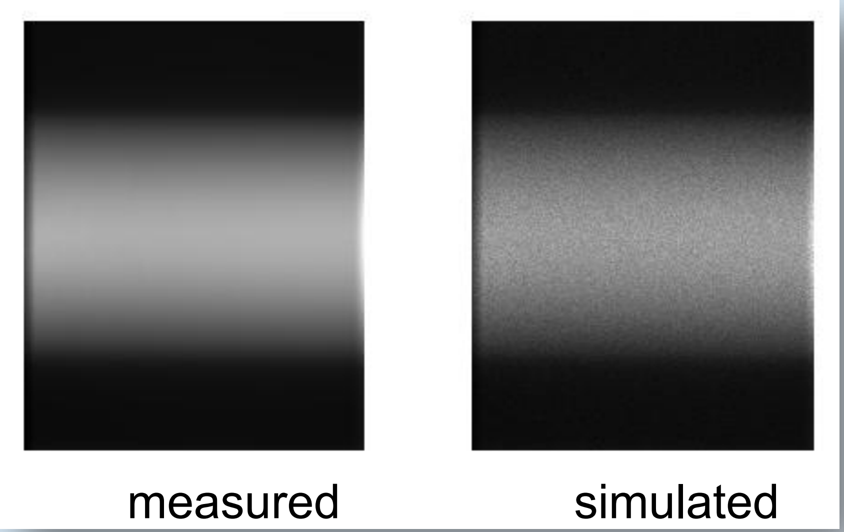
  - gMCDRR: simulate x-ray projections
- Developed based on gDPM but with simpler physics
  - Only photon transport
  - Secondary particle is not needed, so no stack
- gCTD: record dose to voxel
- gMCDRR: record photon energy fluence at imager

Source simulation

- Generate source particle energy according to a known spectrum
  - Generate particles in the entire energy range
  - Simulate each energy bin sequentially

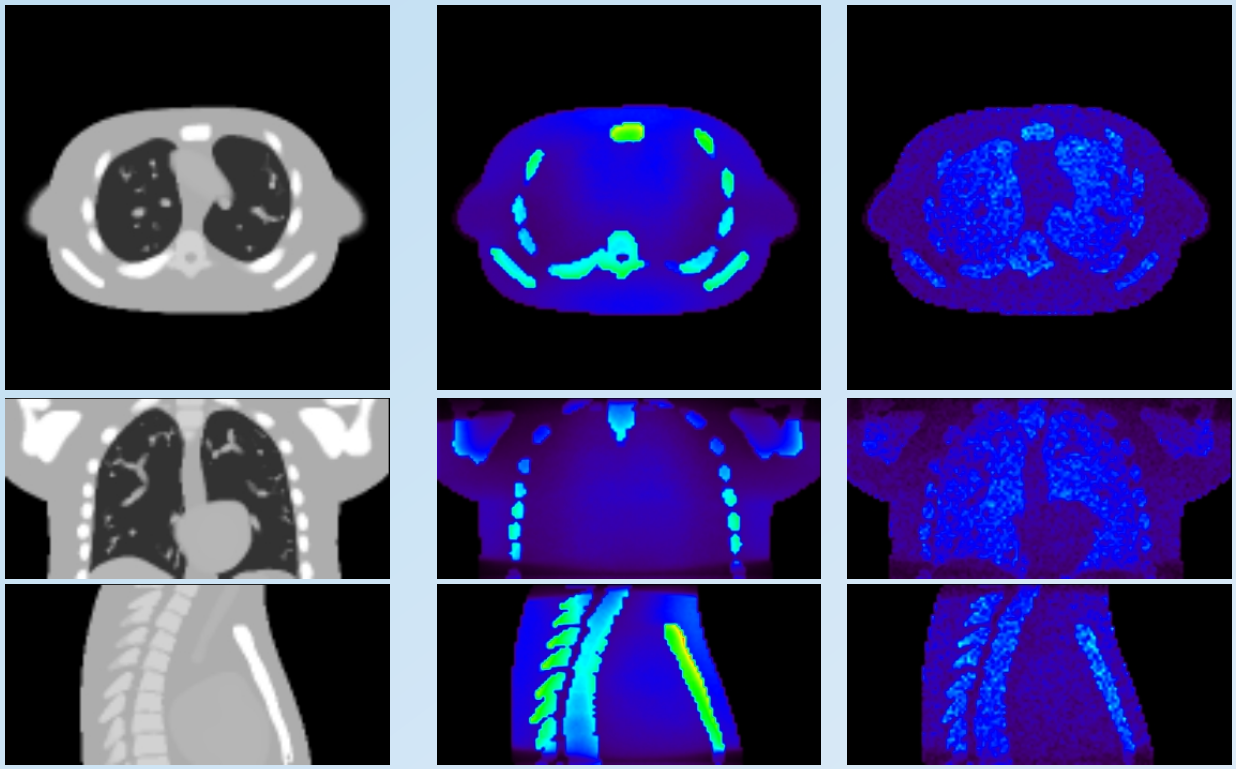


- Generate source particle direction according to measured air scan
  - Using Metroplis algorithm
  - Example: full fan bowtie filter



gCTD Results

- NCAT phantom
  - CBCT scan, gantry angle: 0~360 degree



density   mean   Uncertainty

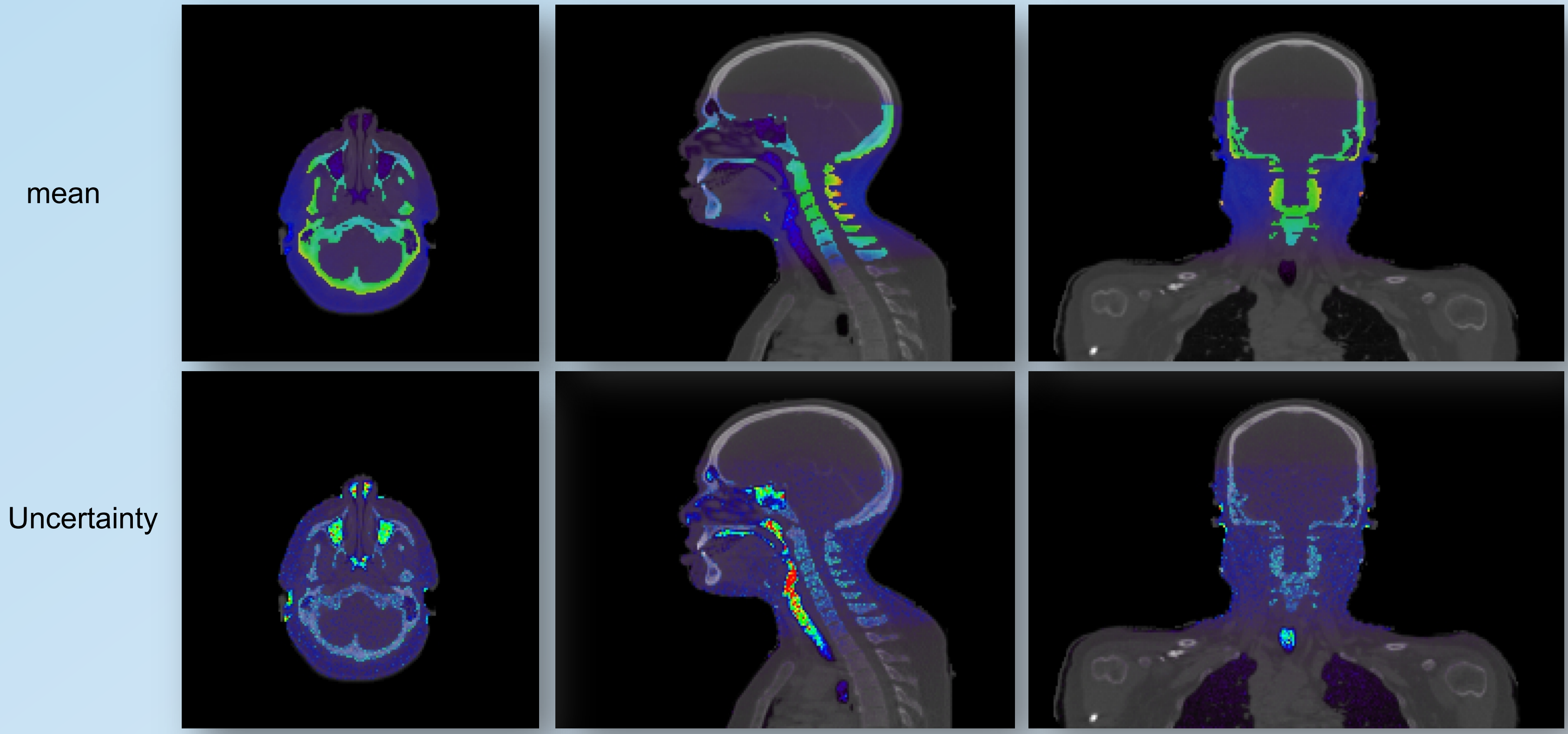
Uncertainty is amplified by 50 times for clear visualization

UC San Diego
RADIATION ONCOLOGY

CART UCSD CENTER FOR ADVANCED RADIOTHERAPY TECHNOLOGIES

gCTD Results

- Head-and-Neck patient
  - CBCT scan, gantry angle: 0~200 degree



Uncertainty is amplified by 10 times for clear visualization

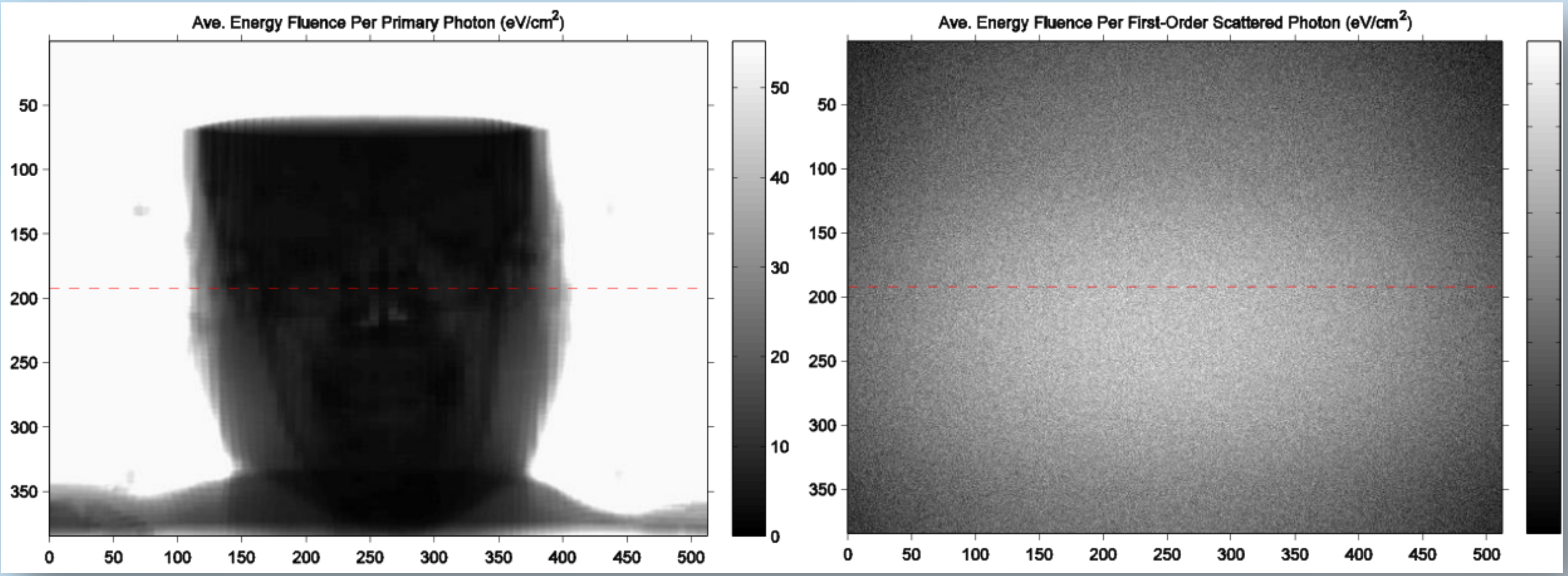
gCTD Results

Case	Resolution	$\langle \sigma_D/D \rangle$ (%)	T(sec)
NCAT	128×128×60	0.47	57.5
HN Patient	256×256×160	0.67	128

- Uncertainty is computed in high dose region, D> $0.3D_{max}$
- 10^9 particles simulated using NVIDIA C2050

gMCDRR Results

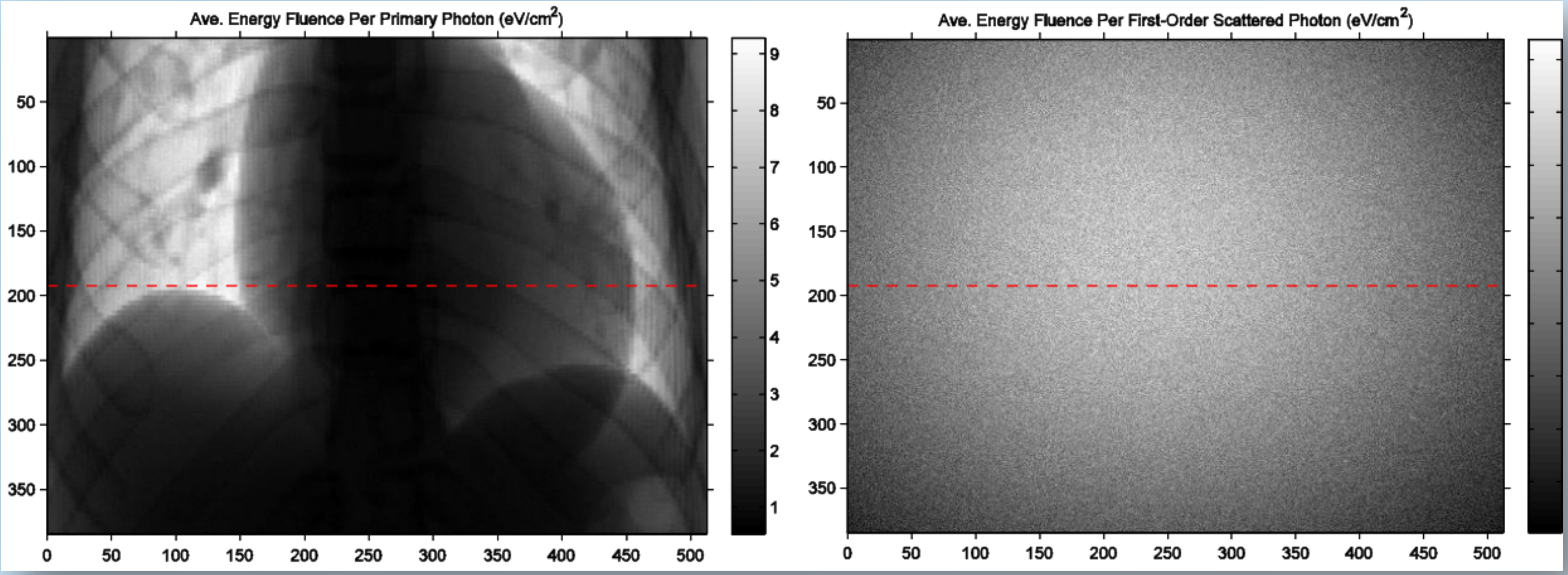
- Head-and-Neck patient



- 10^{10} particles simulated, ~10 min on Tesla C2050

gMCDRR Results

- NCAT phantom



- 10^{10} particles simulated, ~10 min on Tesla C2050

Conclusion

- gDPM: dose calculation for a realistic plan within 1 min or less (with multi-GPU)
- gCTD/gMCDRR: fast dose calculation and kV image simulation for CBCT
- GPU is powerful for MC simulation in radiotherapy
  - Pros: inexpensive, very powerful
  - Cons:

    - Requires careful implementation
    - Not as straightforward as using MPI on a cluster
    - Rewriting/restructuring code is sometimes needed

UC San Diego
RADIATION ONCOLOGY

CART UCSD CENTER FOR ADVANCED RADIOTHERAPY TECHNOLOGIES

Acknowledgement

- Supervisor: Steve Jiang
- Carrie Jiang, Mike Folkerts
- All group members
- SDSC: Dongju Choi, Amit Majumdar