GPU-Based Radiotherapy Treatment Planning

Steve Jiang, Ph.D.

UC San Diego Radiation Oncology
CART UCSD Center for Advanced Radiotherapy Technologies

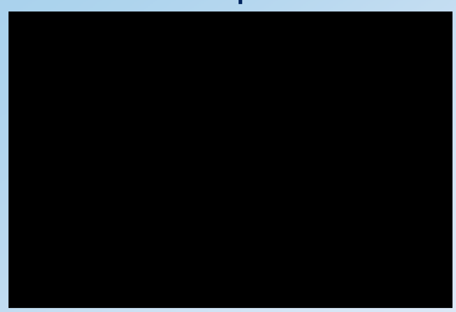
Basic Idea

- So many of our society's technological advances revolve around improving entertainment.
- Some of them can be used to solve medical problems!

What Is GPU?

- GPU - Graphics Processing Unit
- It is a specialized processor that offloads 3D graphics rendering from the microprocessor
- It is used in embedded systems, mobile phones, personal computers, workstations, and game consoles
- More than 90% of new desktop and notebook computers have integrated GPUs
- Video game industry is the main market for GPU

GPU for Computer Games



1.3 billions polygons per second being rendered with >1000 light sources

---

This Cannot Be Done with A Single CPU!

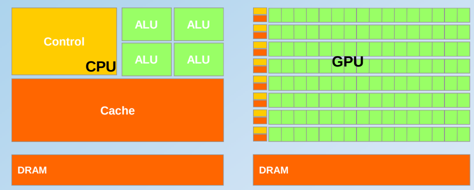
- CPU - Central Processing Unit
  - the "brain" of a computer
  - computes all the logic operations
- Moore's law
  - Introduced in 1965 by Intel co-founder Gordon E. Moore
  - The number of transistors on integrated circuits (including CPUs) doubles every two years
  - Processing speed is strongly linked to the number of transistors
  - Increase CPU speed by increasing the number of transistors
  - Limitation:
    - heat dissipation has become a bottleneck for increasing the CPU clock speed
  - **New paradigm:**
    - **Increase the number of computing units (or cores) in CPU instead of increasing the clock speed**

What You Need Is A Supercomputer!

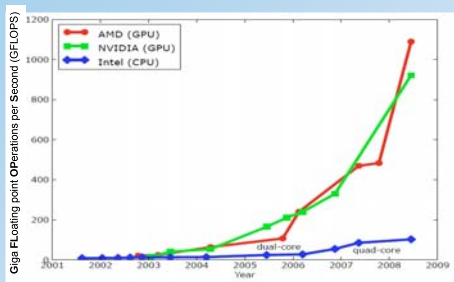
Need A Dedicated Hardware Architecture

- Data level parallelism
  - apply the same operations to all the vertices
  - apply the same operations to all the pixels
- Simple Instruction Multiple Data (SIMD) hardware architecture
  - perform the same operations on multiple data simultaneously
  - SIMD instructions embedded in GPU to process 3D graphics

CPU vs GPU: Different Design Philosophies

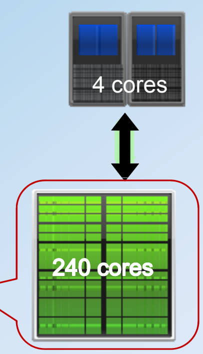


CPU vs GPU



Graphics Processing Unit

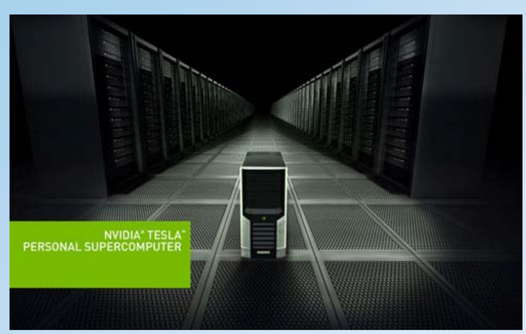
- Personal supercomputer
  - Tesla C1060
  - 240 processors
  - 1.3 GHz clock speed
  - 4 GB memory
  - Single precision floating point



GPUs We Have

NVIDIA GPU card	GeForce 9500 GT	GeForce GTX 285	Tesla C1060	Tesla S1070	Tesla C2050
Price ($)	~50	~300	~1,000	~8,000	~2,500
Memory (GB)	1	1	4	16	3
Computing power (Gflops)	134	1062	936	4320	520 (dp)
# of Processors	32	240	240	960	512

Personal Supercomputer



A Big Word of Thanks!

… to the millions of computer game enthusiasts worldwide

Who demand an utmost of performance and realism of their game engines

And who create a market force for high performance computing that beats any federal-funded effort (DOE, NASA, etc.)

Slide Courtesy of Klaus Meuller

GPU-based Treatment Planning @ UCSD

- Dose calculation
  - g-FSPB: finite size pencil beam model
  - g-DC-FSPB: finite size pencil beam model with 3D density correction
  - gDPM v1: direct porting of DPM MC code to GPU
  - gDPM v2: optimized for GPU but still with the same physics
- Plan optimization
  - gFMO: fluence map optimization
  - gDAO: direct aperture optimization
  - gVMAT: optimization for volumetric modulated arc therapy

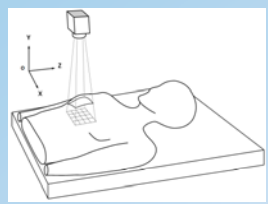
14

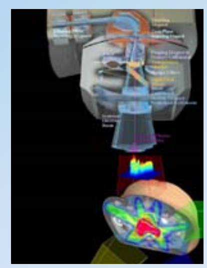
Development of GPU-based Real-time FSPB Dose Calculation

g-FSPB: Gu et al *Phys Med Biol* 54(20) 6287-97, 2009
g-DC-FSPB: Gu et al *Phys Med Biol* 2011 (Submitted)
(http://arxiv.org/ftp/arxiv/papers/1103/1103.1164.pdf)

15

Finite-size Pencil Beam (FSPB) Model



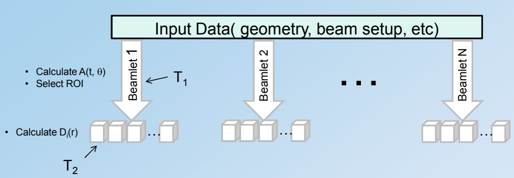


[illegible]

- Original model - Bourland and Chaney, *MP* 1992
- 3D density correction - Jelen and Alber, *PMB* 2007
- Parameters are commissioned with MC simulation data

16

Highly Data-Parallel Task



Naïve Computational Time Estimation

$$T_{CPU} = N_{beamlets} \cdot (T_1 + N'_{voxels} \cdot T_2)$$

$$T_{GPU} = \frac{N_{beamlets}}{N^{(1)}_{threads} \cdot N^{(1)}_{blocks}} \cdot T_1' + \frac{N_{beamlets}}{N^{(2)}_{blocks}} \cdot \frac{N'_{voxels}}{N^{(2)}_{threads}} \cdot T_2'$$

17

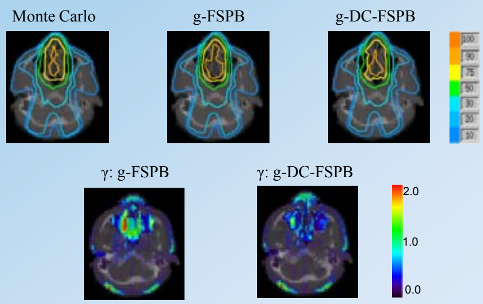
Patient Cases

Table 1. Tumor site, number of beams, and case dimension for 5 head-and-neck (H1-H5) cases and 5 lung (L1-L5) cases.

Case	Tumor Site	# of Beams	# of Beamlets	# of Voxels
H1	Parotid	8 (non-coplanar)	7,264	128×128x72
H2	Hypopharynx	7 (non-coplanar)	4,429	128x128x72
H3	Nasal Cavity	8 (non-coplanar)	3,381	128x128x72
H4	Parotid	5 (coplanar)	4,179	128x128x72
H5	Larynx	7 (non-coplanar)	10,369	128x128x72
L1	Left lung, low lobe(close to pleura)	6 (coplanar)	637	128x128x80
L2	Right lung, low lobe (paravertebral)	6 (coplanar)	1,720	128x128x103
L3	Left lung, upper lobe (close to pleura)	5 (coplanar)	921	128x128x80
L4	Right lung, upper lobe (close to heart)	7 (coplanar)	841	128x128x80
L5	Left lung (middle)	5 (coplanar)	686	128x128x80

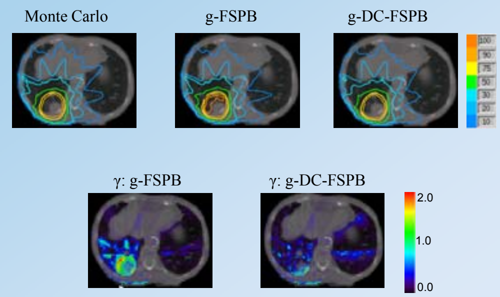
18

Head and Neck Cancer Patient Case (H3)



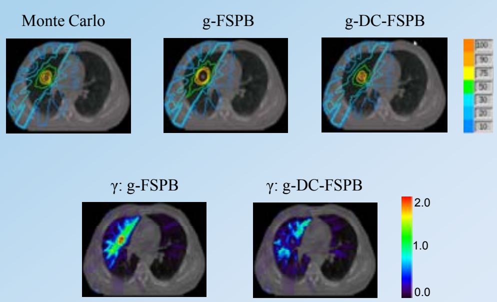
UCSan Diego Radiation Oncology — 19 — CART UCSD

Lung Cancer Patient Case (L2)



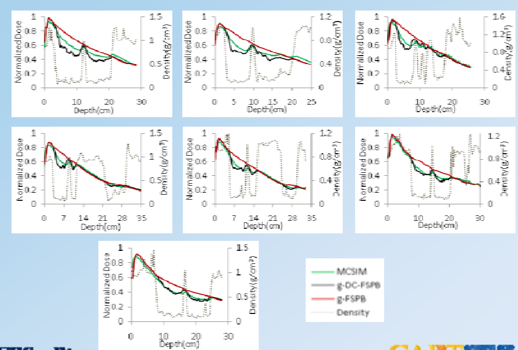
UCSan Diego Radiation Oncology — 20 — CART UCSD

Lung Cancer Patient Case (L4)



UCSan Diego Radiation Oncology — 21 — CART UCSD

Lung Cancer Patient Case (L4)



UCSan Diego Radiation Oncology — 22 — CART UCSD

Accuracy and Efficiency

Table 2. Gamma index evaluation results and dose calculation computation time for 10 testing cases using the g-DC-FSPB algorithm. The corresponding g-FSPB results are given in parenthesis for comparison purpose.

Case #	γ^{max}	γ_{EE}^{avg}	P_{EE}	T_{tr}(sec)	T_{gpu}(sec)	T_{tot}(sec)
H1	2.12 (2.16)	0.30 (0.31)	97.53% (97.32%)	0.20	0.64 (0.55)	0.84 (0.75)
H2	3.44 (4.11)	0.28 (0.28)	97.80% (97.01%)	0.20	0.40 (0.35)	0.60 (0.55)
H3	2.27 (2.36)	0.46 (0.52)	92.29% (86.39%)	0.20	0.38 (0.34)	0.58 (0.54)
H4	3.08 (3.11)	0.61 (0.63)	82.96% (81.56%)	0.19	0.35 (0.32)	0.54 (0.51)
H5	3.33 (3.37)	0.61 (0.61)	86.19% (86.09%)	0.20	1.31 (1.10)	1.51 (1.30)
L1	1.53 (1.92)	0.24 (0.45)	99.35% (94.81%)	0.21	0.22 (0.20)	0.43 (0.41)
L2	2.35 (3.30)	0.36 (0.71)	96.64% (76.38%)	0.22	0.40 (0.36)	0.62 (0.58)
L3	1.68 (3.07)	0.32 (0.75)	99.16% (76.60%)	0.21	0.30 (0.25)	0.51 (0.46)
L4	2.70 (4.59)	0.63 (1.53)	81.33% (28.55%)	0.18	0.25 (0.23)	0.43 (0.41)
L5	2.19 (4.34)	0.49 (1.13)	90.24% (57.03%)	0.21	0.33 (0.29)	0.54 (0.50)
Median	2.32(3.20)	0.41(0.62)	94.46%(83.83%)	0.20	0.37 (0.33)	0.56 (0.53)

UCSan Diego Radiation Oncology — 23 — CART UCSD

Development of GPU-based Fast Monte Carlo Dose Calculation

Jia et al *Phys Med Biol* 55(11): 3077–3086, 2010
Jia et al *Phys Med Biol* 2011 (to be submitted)

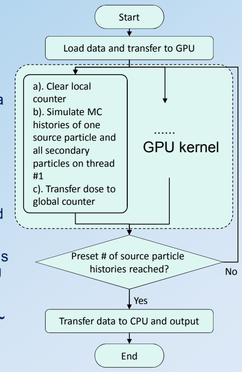
UCSan Diego Radiation Oncology — 24 — CART UCSD

gDPM project

- Started on July 2009
- 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
- Parallelize MC simulation on GPU
  - MC is known as task parallelization
  - GPU favors data parallelization --- SIMD

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



Improved Areas in gDPM v2

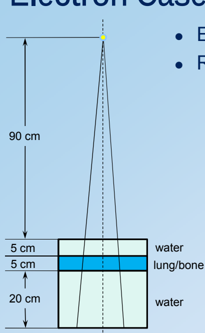
- Branching issue on GPU
  - Different physics between photons and electrons
  - Different paths/interactions between particles of same type
- Random number generator efficiency
- Interpolation of cross section data
- Global memory access speed and access conflict
  - Dose deposition between GPU threads
  - Secondary particles stacking

Other Components

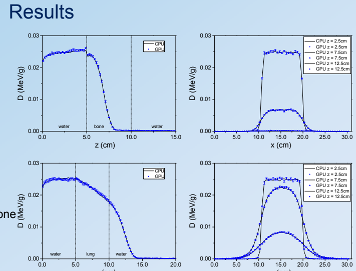
- Load egs4phan format to define patient anatomy (voxel materials and structure information)
- Enable gantry, couch, collimator rotations
- Flexible source function
  - User can supplement with their own realistic linac source model or phase space file
- Enable simulating fluence map and MLC

Dose caculation in realistic IMRT & VMAT treatment plans

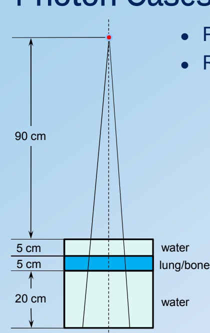
Electron Cases



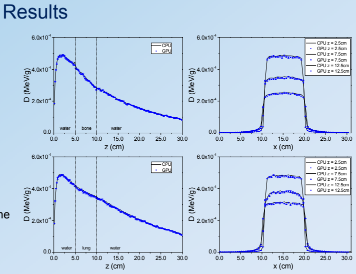
- Electron point source, 20 MeV
- Results



Photon Cases

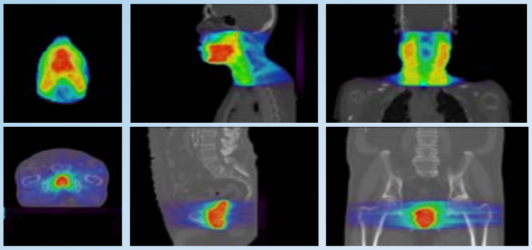


- Photon point source, 6 MV spectrum
- Results



RapidArc Cases

- Photon point source, 6 MV spectrum
- 2 arcs

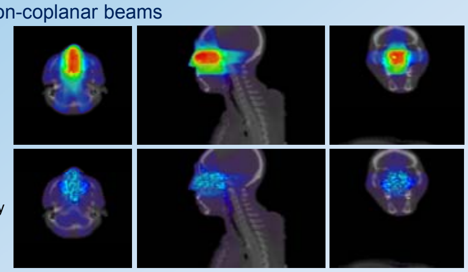


IMRT Case

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

Mean

Uncertainty



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 $P_{3\%}$ and P_t.

Source type	# of Histories	Case	$\langle \sigma_D/D \rangle$ CPU (%)	$\langle \sigma_D/D \rangle$ GPU (%)	$P_{3\%}$ (%)	P_t (%)
20MeV Electron	2.5×10^6	water-lung-water	0.99	0.98	99.3	99.9
20MeV Electron	2.5×10^6	water-bone-water	0.98	0.99	99.8	100.0
6MV Photon	2.5×10^8	water-lung-water	0.71	0.72	98.6	98.5
6MV Photon	2.5×10^8	water-bone-water	0.64	0.64	98.9	96.9
6MV Photon	2.5×10^8	VMAT HN patient	N/A	0.98	N/A	N/A
6MV Photon	2.5×10^8	VMAT Prostate patient	N/A	0.74	N/A	N/A
6MV Photon	2.5×10^8	IMRT HN patient	N/A	0.57	N/A	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	36.7	N/A
6MV Photon	2.5×10^8	VMAT Prostate patient	N/A	39.6	N/A
6MV Photon	2.5×10^8	IMRT HN patient	N/A	36.1	N/A

CPU: Intel Xeon processor with 2.27GHz
GPU: NVIDIA Tesla C2050

Development of A GPU-based Fluence Map Optimization (FMO) Algorithm for IMRT

Men et al *Phys Med Biol* 54(21):6565-6573, *2009*

FMO model and flow chart

$$\min \sum_{j \in V} F_j(z_j^l)$$

Subject to

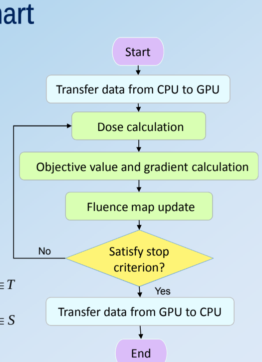
$$z_j^l = \sum_{i \in N} D_{ij}^l x_i \qquad j \in V$$

$$x_i \geq 0 \qquad i \in N$$

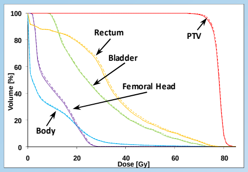
Where

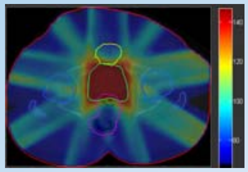
$$F_{s-}(\mathbf{z}) = \sum_{j \in V_s} (\max(0, z_j - z_j^l))^2 \qquad s \in T$$

$$F_{s+}(\mathbf{z}) = \sum_{j \in V_s} (\max(0, z_j^l - z_j))^2 \qquad s \in S$$



Results for GPU-based FMO Algorithm





# Case	beamlet size (mm²)	# beamlets	voxel size (mm³)	# voxels (×10⁴)	# non-zero D_{ij}'s (×10⁶)	GPU time (s)
1	10×10	2,055	4×4×4	3.6	3.1	0.2
2	5×5	6,433	4×4×4	3.6	10.6	0.5
3	5×5	6,433	2.5×2.5×2.5	14.0	43.3	2.8

~40x speedup compared to an Intel Xeon 2.27 GHz CPU

~0.5 sec for re-optimizing a 9-field prostate IMRT plan

Men et al *Phys Med Biol* 54(21):6565-6573, *2009*

Development of GPU-based Direct Aperture Optimization (DAO)

Men et al *Phys Med Biol* 55(15):4309-4319, 2010

DAO Model

- Model

$$\min_{y_k, A_k} \sum_{j \in V} \alpha_j \max(z_j - T_j, 0)^2 + \beta_j \max(T_j - z_j, 0)^2$$

Subject to

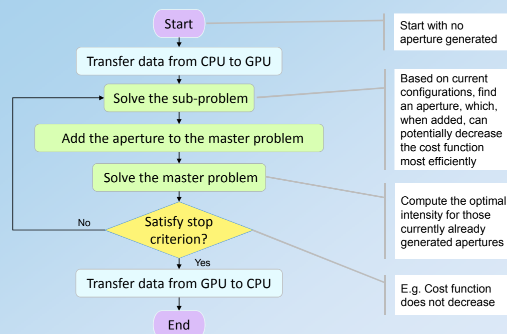
$$z_j = \sum_{k \in A_k} D_{kj} y_k \quad j \in V$$

$$y_k \geq 0 \quad k \in K$$

Optimize w.r.t. both aperture A_k and intensity y_k

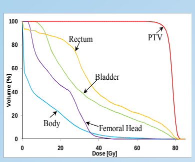
- It is reasonable to expect that only a relatively small number of apertures will actually have positive intensity. The challenge is therefore to identify a small set of apertures that yield a high-quality treatment plan
- We use a formal column generation approach to solving the DAO problem

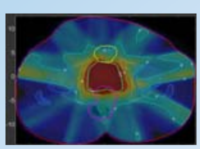
Flowchart



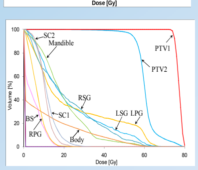
Results

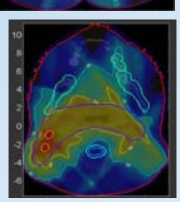
Prostate





Head and Neck





Results

- 5 prostate cases (P1~P5) and 5 H&N cases (H1~H5) are tested

Case	# beamlets	# voxels	# non-zero D_{ij}'s	Running time (sec)
P1	7,196	45,912	2,763,243	1.7
P2	7,137	48,642	2,280,076	0.7
P3	5,796	28,931	1,765,294	0.8
P4	7,422	39,822	2,717,424	2.3
P5	8,640	49,210	3,086,884	1.6
H1	5,816	33,252	1,576,418	1.0
H2	8,645	59,615	3,162,752	2.4
H3	9,034	74,438	3,500,188	1.8
H4	6,292	31,563	1,596,168	1.8
H5	5,952	42,330	2,215,202	2.5

Development of GPU-based VMAT Optimization Algorithm

Men et al *Med Phys* 37(11): 5787-5791, 2010

UCSan Diego Radiation Oncology — 46 — CART UCSD

VMAT model

- Model

$$\min_{y_k, A_k} \sum_{j \in V} \alpha_j \max(z_j - T_j, 0)^2 + \beta_j \max(T_j - z_j, 0)^2 + \mu \sum_k \left[\frac{y_{k+1} - y_k}{\theta_{k+1} - \theta_k} \right]^2$$

Subject to $z_j = \sum_{k \in A_k} D_{kj} y_k \quad j \in V$

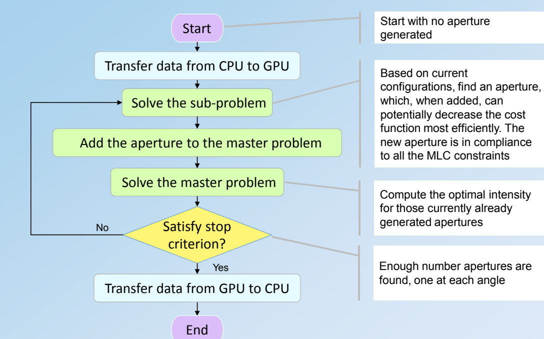
$y_k \geq 0 \quad k \in K$

Optimize w.r.t. both aperture A_k and intensity y_k

- Additional constraints
  - Only one aperture is at one beam angle
  - Aperture shapes at neighboring angles satisfy MLC mechanical constraints
  - The second term in cost function constraints smoothness of y_k between neighboring angles
- We use a formal column generation approach to solving the VMAT problem

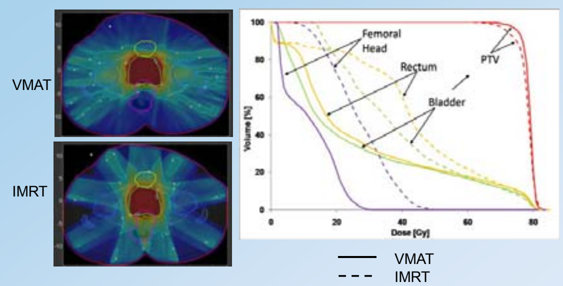
UCSan Diego Radiation Oncology — 47 — CART UCSD

Flowchart



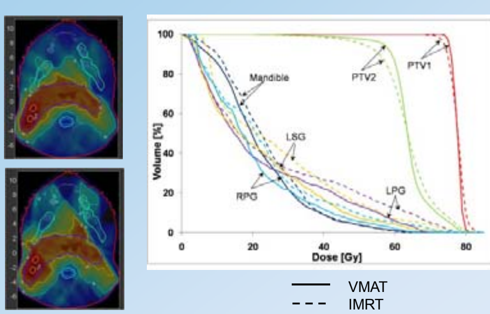
UCSan Diego Radiation Oncology — 48 — CART UCSD

Prostate cancer case



UCSan Diego Radiation Oncology — 49 — CART UCSD

Head-and-neck cancer case



UCSan Diego Radiation Oncology — 50 — CART UCSD

Results

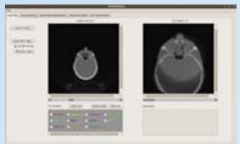
- 5 prostate cases (P1~P5) and 5 H&N cases (H1~H5) are tested

Case	# beamlets	# voxels	# non-zero D_{ij}'s ($\times 10^7$)	CPU time (sec)	GPU time (sec)
P1	40,620	45,912	2.3	340	22
P2	59,400	48,642	3.2	265	18
P3	38,880	28,931	1.8	276	20
P4	43,360	39,822	2.6	410	26
P5	51,840	49,210	3.0	348	23
H1	51,709	33,252	2.5	290	21
H2	78,874	59,615	5.0	468	27
H3	90,978	74,438	5.5	342	25
H4	71,280	31,563	2.6	363	25
H5	53,776	42,330	3.5	512	31

UCSan Diego Radiation Oncology — 51 — CART UCSD

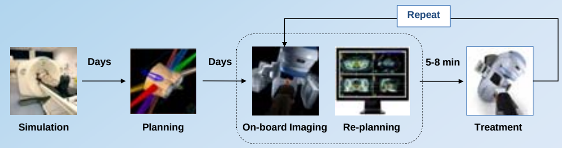
Summary – GPU-based Treatment Planning

- We have developed GPU-based computational tools for real-time treatment planning
- For a typical prostate case
  - The dose calculation takes less than 1 second with FSPB with 3D density correction, less than 40 seconds with Monte Carlo
  - The plan optimization takes less than 1 second with FMO, 2 seconds with DAO, and 30 seconds with VMAT
- Next step
  - Faster → algorithm improvement, multiple GPUs
  - Software integration → A research platform (SCORE: Supercomputing Online Re-planning Environment)
  - Clinical implementation and evaluation



New Treatment Techniques

- Interactive treatment planning
  - Interactively modifying the DVH curves and isodose lines
  - Interactive plan updating
  - Solve the multi-objective optimization problem
- Online adaptive radiotherapy



Acknowledgement

http://radonc.ucsd.edu/Research/CART

http://arxiv.org/

Short Course on GPU Programming