AbstractID: 4966 Title: Correlated Sampling for accelerating CT-based Monte Carlo dose calculations for brachytherapy treatment planning

Purpose: To investigate the potential of correlated sampling variance reduction technique for accelerating CT-based Monte Carlo (MC) simulations for calculating 3D brachytherapy dose distributions.

Method and Materials: Correlated MC (CMC) simulations generate photon histories in the homogeneous geometry. By recomputing particle weights to account for non-water composition of the CT voxels and applicator and seed components, a second highly correlated set of histories is constructed, resulting in a lower variance estimate of the dose difference, $\Delta D=D_{het}-D_{hom}$ To evaluate the accuracy and efficiency of CMC, a clinical permanent prostate implant with 78 I-125 seeds was simulated using both CMC and UMC versions of our accelerated CT-based MC dose-computation code, PTRAN_CT for voxel sizes ranging from $1\times1\times1mm^3$ to $2\times2\times2mm^3$. Mean efficiency gains were estimated for regions with minimum doses greater than 20%, 50% and 90% of D_{90}, as well as different anatomical regions.

Results: Systematic differences between UMC and CMC PTRAN_CT were less than 0.4%. Efficiency gains ranged from 4.68 to 15.76 depending on the voxel size and region. CMC can achieve a 2% average precision with 2 mm cubic voxels in 23 seconds on a single P4 processor for voxels with doses > $50\%D_{90}$. Voxels with very low doses and/or large dose perturbations can experience efficiency losses. Because ΔD is a relatively smoothly varying quantity compared to D_{het}, CMC efficiency maybe enhanced using coarser voxel sizes than UMC for the same level of volume-averaging artifact.

Conclusion: Correlated sampling MC can reduce CPU times by an additional 4-15 fold compared to accelerated uncorrelated MC. In practice, larger efficiency enhancements can be achieved because correlated sampling volume-averaging errors are smaller. MC-based brachytherapy treatment planning, requiring only a few seconds of CPU time, is achievable. (Supported by NIH-R01CA46640)