AbstractID: 8357 Title: IMRT Leaf Sequencing with Intensity-based Segment Weight Optimization

Purpose

Dose distribution based segment weight optimization can significantly improve the quality of IMRT plans after leaf sequencing despite it is computationally very time-consuming. This research aims to develop a leaf sequencing algorithm that has built-in intensity-based segment weight optimization with minimal additional running time.

Methods

The new algorithm is named LSWISW (for leaf sequencing with intensity-based segment weight optimization). Its input includes: (1) (continuous) intensity patterns; (2) MLC leaf leakage ratio (typically ranges from 1% to 3%); either (3) the number of MLC segments, *k*, specified by the user or (4) an upper bound on the error between the computed intensity pattern and original ideal intensity pattern. If (3) is specified, a plan of *k* segments that best approximates the ideal intensity pattern is calculated. If (4) is specified, a plan of the minimum number of segments whose error is within the error bound is produced.

In LSWISW, the leaf sequencing is modeled as a constrained shortest path problem and solved using dynamic programming; segment weight optimization is modeled and solved as a nonnegative least square optimization.

Results

The performance of our leaf sequencing algorithm was tested on three treatment sites (head-and-neck, lung, and prostate). Our target delivery system is the Varian LINAC, whose MLC allows interdigitiation. In all the cases, our algorithm produces IMRT plans that rival those from Pinnacle planning system with afterward dose-based segment weight optimization.

Execution times of no more than 1 minute for our algorithm were observed on a laptop computer with a Pentium M Processor of 2.0 GHz.

Conclusion

We presented an IMRT leaf sequencing algorithm with built-in intensity-based segment weight optimization. Compared with the leaf sequencing and dose-based segment weight optimization modules from Pinnacle planning system, our new leaf sequencing algorithm is much more effective and efficient.