A Sphere-Packing Model for the Optimal Treatment Plan

Long Yun
Ye Yungqing
Wei Zhen
Peking University
Beijing, China

Advisor: Liu Xufeng

Abstract

We develop a sphere-packing model for gamma knife treatment planning to determine the number of shots of each diameter and their positions in an optimal plan.

We use a heuristic approach to solve the packing problem, which is refined by simulated annealing. The criteria for an optimal plan are efficiency, conformity, fitness, and avoidance. We construct a penalty function to judge whether one packing strategy is better than the other. The number of spheres of each size is fixed, the total number of spheres has an upper bound, and critical tissue near the target is avoided.

Computer simulation shows that our algorithm fits the four requirements well and runs faster than the traditional nonlinear approach. After detailed evaluation, we not only demonstrate the flexibility and robustness of our algorithm but also show its wide applicability.

Introduction

We develop an effective sphere-packing algorithm for gamma-knife treatment planning using a heuristic approach, optimized by simulated annealing.

In our model, we take into consideration the following basic requirements:

1. At least 90% shot coverage of the target volume is guaranteed. This requirement is the main standard for evaluating our algorithm, or an *efficiency* requirement.

2. Minimize the non-target volume that is covered by a shot or by a series of delivered shots. This requirement is a *conformity* requirement.

The UMAP Journal 24 (3) (2003) 339–350.

3. Minimize the overlapped region of the delivered shots in order to avoid the hot spots as well as to economize shot usage. This is a a *fitness* requirement.
4. Limit the dosage delivered to certain critical structures close to the target. Such requirements are *avoidance* requirements.

The traditional model for radiosurgery treatment planning via nonlinear programming assumes that the weights of the shots conform to a certain distribution, from which the construction of the objective function is possible. To avoid the complicated computation of nonlinear programming, we devise a more feasible and rapid heuristic algorithm without reducing any precision of the outcome.

- We consider an optimal sphere-packing plan for a given number of spheres in each size, satisfying requirements 1–3. That is, in this step, we assume that the lesion part is far from any critical organ and try to find an optimal position for a fixed set of the spheres using the heuristic sphere-packing algorithm.
- We try all possible combinations of up to 15 spheres; for each, we use the above algorithm to get an optimal plan. We develop a criterion to select from the different combinations the best packing solution for our model, which is optimized by simulated annealing.
- We consider the real situation in field practice, in which the effect of a critical organ is added. Accordingly, we modify the judgment criterion so that requirement 4 is satisfied.
- Finally, to apply the above method to more general situations, we add the weights of the shots.

Though we admit that the inherent limitations of this model due to the simplification of the problem and the restriction of the hardware capacity are unavoidable, we believe that our model has successfully solved the given problem. Our algorithm is not only fast in generating solutions but also flexible in allowing parameter settings to solve more difficult tasks.

Assumptions

- Shots can be represented as spheres with four different diameters: 4, 8, 14, and 18 mm.
- The target volume is of moderate size with a mean spherical diameter of 35 mm (and usually less) [The Gamma Knife . . . n.d.]
- The maximum number of shots allowed is 15.

- The target object is represented as a three-dimensional digital map with $100 \times 100 \times 100 = 1$ million pixels.
- The volume of a object is measured by the total number of pixels in it.
- The dose delivered is above the lower bound of the effective level to kill the tumor.

Table 1. Description of the variables.

N	total number of shots
n_i	number of shots of type i
s	the sth shot used, $s = 1, \ldots, N$
(x_s, y_s, z_s)	position of the sth shot center
Position	matrix storing all the positions of the shot centers
M	average shot width
Radius	vector storing the four types of radius: [9 7 4 2]
Bitmap	$M \times M \times M$ boolean matrix storing information information from the CT/MRI image
Dose	dose delivered, a linear function of exposure time satisfying $\theta \leq \text{Dose}(i, j, k) \leq 1$, where θ is the lower bound of the isodose contour
Covered	total number of covered pixels in the target volume; directly reflects the efficiency requirement
Miscovered	total number of covered pixels in the normal tissue; directly reflects the conformity requirement
Overlap	total number of overlapped pixels among different shots; directly reflects the fitness requirement
Ratio	percentage of the target volume covered
SphereInof	vector representing the number of each type of shot
SphereRadius	vector representing the radius of N shots

Background Knowledge

Gamma knife radiosurgery allows for the destruction of a brain lesions without cutting the skin. By focusing many small beams of radiation on abnormal brain tissue, the abnormality can be destroyed while preserving the normal surrounding structures. Before the surgery, the neurosurgeon uses the digitally transformed images from the CT/MRI to outline the tumor or lesion as well as the critical structures of the surrounding brain. Then a treatment plan is devised to target the tumor.

The determination of the treatment plan varies substantially in difficulty. When the tumor is large, has an irregular shape, or is close to a sensitive structure, many shots of different sizes could be needed to achieve appropriate coverage of the tumor. The treatment planning process can be very tedious and time-consuming due to the variety of conflicting objectives, and the quality of the plan produced depends heavily on the experience of the user. Therefore, a unified and automated treatment process is desired.

In our model, we reduce the treatment planning problem to an optimal sphere-packing problem by focusing on finding the appropriate number of spheres of different sizes and their positions to achieve the efficiency, conformity, fitness, and avoidance requirements.

Construction and Development of the Model

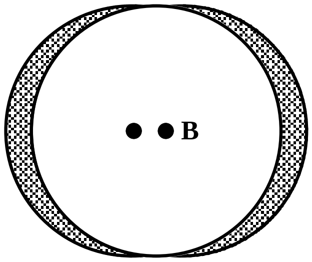
Fixed Set of Spheres

The main idea is to let a randomly chosen sphere move gradually to an optimal position.

In the beginning, N spheres are randomly positioned inside the target volume. Then one sphere is moved to a new position and we estimate whether the new location is better; if so, the sphere is moved, otherwise it remains in place. We repeat this process until a relatively good packing solution is achieved.

To implement our algorithm, we need a criterion to judge a packing solution. According to our four requirements, it is reasonable to take the weighted linear combination of the volume of those covered, miscovered, and overlapped parts as our criterion—that is, a good packing solution means less miscovered, less overlapped, and more covered volumes.

Let sphere A move to location B **Figure 1**. We restrict our consideration to just the pixels in the shaded area, which is very thin and thus has few pixels in it. The program judges which region a pixel belongs to: covered, miscovered or overlapped, and we count the pixels of each kind. We implement this idea using a function `PenaltyJudge` that returns a signed integer indicating whether the change of the packing strategy results in a better solution.



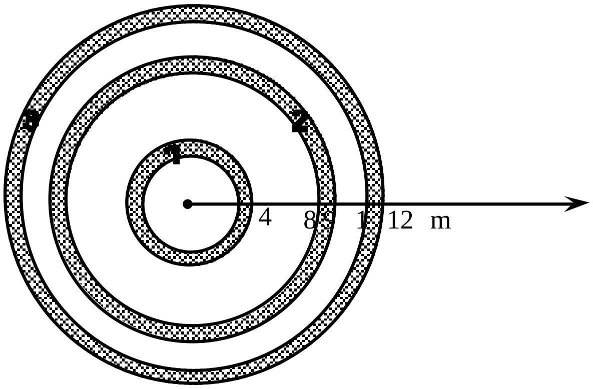


Figure 1. Sphere A moves to location B. **Figure 2.** The centers of the 18 mm spheres are set at O, the the centers of the three smaller spheres at random points in regions 1, 2, and 3, respectively.

How do we set the initial position of the spheres? Our results will be affected significantly if the starting positions are not properly set. Cramming the spheres together will not do any harm, because according to our algorithm, all of the spheres move in different directions and finally scatter through the target volume. But there is one constraint that the initial positions must obey: Larger spheres cannot cover smaller ones. Otherwise, the smaller spheres will never move out of the larger ones, which means they are useless and wasted. Since spheres of the same size will not be covered by each other as long as their centers differ, we need to avoid only coverings between spheres of different sizes. Our technique is to set the spheres of different size in different regions of the target volume, which ensures that the spheres never cover each other.

In **Figure 2**, point O represents the center of the CT image of the target volume. We set the centers of all the 18-mm spheres at point O and center the 14-mm, 8-mm, and 4-mm spheres randomly at the tumor pixels lying in the shadowed regions 1, 2, 3, respectively. Thus, a relatively good starting status is generated.

We perturb the location of one sphere by a step (i.e., one pixel) in the North, South, East, West, Up, and Down directions. If a perturbation in one of these directions generate a better packing, we move the sphere one step in that direction. Then we choose another sphere and repeat the process. Applying this process to all of the spheres successively is one iteration. Our program generally generates a relatively good packing in about 10–15 iterations.

Results and Data Analysis

Heuristic Method

To test the effectiveness of our algorithm, we construct a 3D target object with 100 × 100 × 100 pixels through the combination of two spheres and a segment of a circle. For the effect of live simulation, we blur and distort the edge of the object through photoprocessing software so that it is very similar to the shape of a real tumor. The simulated results and the solution given by our program are excitingly good, as shown in **Table 2**.

Table 1.

Final distribution of shots from the heuristic algorithm on a simulated target.

Iterations	% covered	% miscovered	% overlapped	Time consumed (s)
0	37	39	14	—
5	75	25	8	20
10	96	11	5	40
15	96	11	5	62
20	96	11	5	83

Visualization of the Results

Plotting the resulting bitmap, we can see clearly from **Figures 3–4** the evolution of the locations of the spheres as well as the stability and robustness of our program.

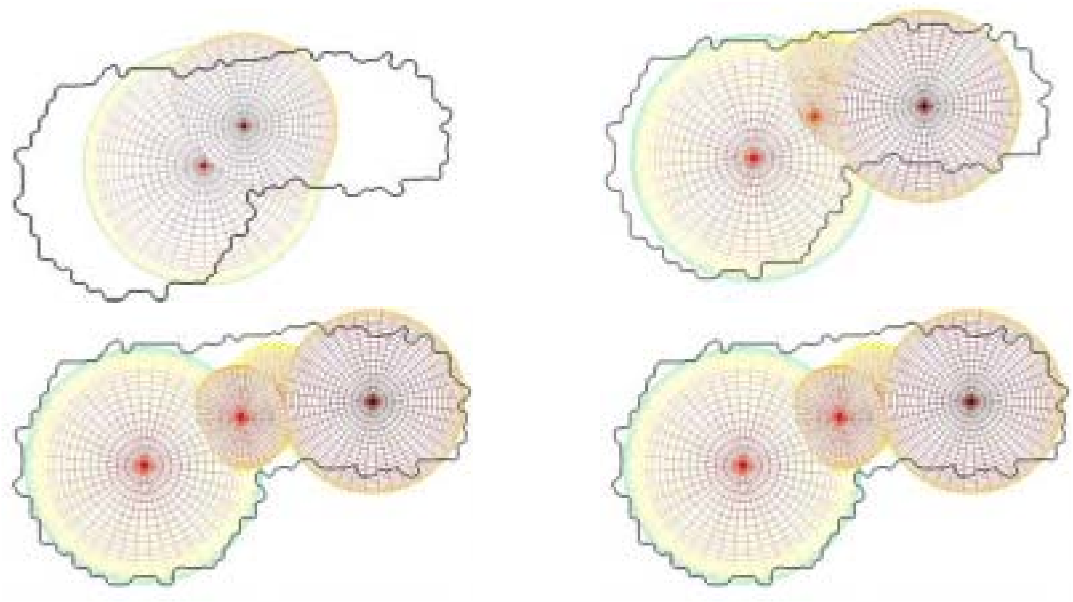


Figure 3. Distribution of spheres within the target after 0, 5, 10, and 15 iterations.

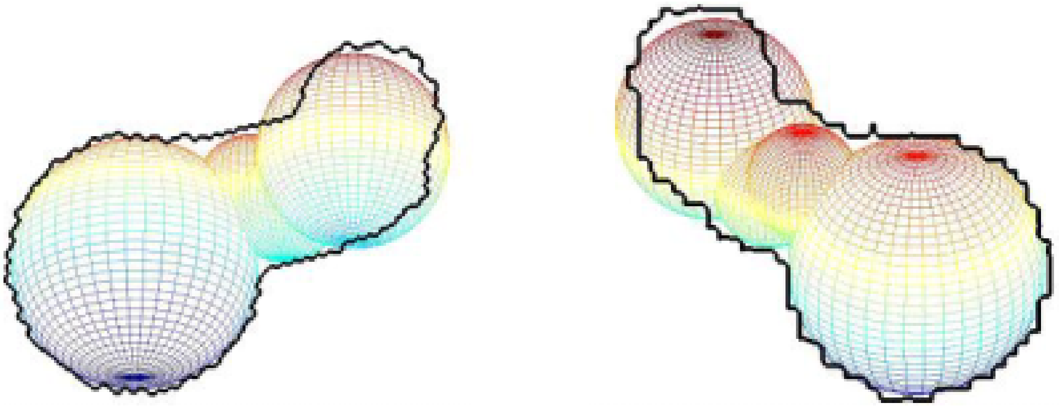


Figure 4. Three-dimensional views of final placement of spheres.

After 10 iterations, all the spheres go into a relatively stable position. Such fast and stable convergence occurs in all of our simulations. Hence, we can reasonably assume that after 15 iterations any initial packing will turn into an optimal one.

Further Development: The Best Set of Spheres

Difficulties and Ideas

So far, we have used our personal judgment about which sets of spheres to pack. A natural idea is to enumerate all combinations of the four types of spheres and find an optimal one. There are $\binom{19}{4} = 3,876$ nonnegative integer solutions to the equation $n_1 + n_2 + n_3 + n_4 \leq 15$. The runtime for our program to check them all, at 83 s each, would be 89 h, excluding the time to read the bitmap and plot the graphs.

So, to get a near-optimal but efficient solution, we turn to simulated annealing, which we use not only to find the optimal combination of the spheres but also to determine the direction of the spheres to move in each step.

Simulated Annealing to Find the Optimal Combination

Simulated annealing (SA) is a Monte Carlo approach used in a wide range of problems concerning optimization, especially NP-complete problems, which can approximate the global extremum within the time tolerance.

SA is a numerical optimization technique based on the principles of thermodynamics. The algorithm starts from a valid solution, randomly generates new states for the problem, and calculates the associated cost function. Simulation of the annealing process starts at a high fictitious temperature (usually manipulated according to practical need). A new state is randomly chosen from the "neighborhood" of the current state (where "neighborhood" is a map defined as $N:\ \text{State} \to 2^{\text{State}},\ i \to S_i$ satisfying $j \in S_i \iff i \in S_j$) and the difference in cost function is calculated. If (CurrentCost $-$ NewCost) ≤ 0, i.e., the cost is lower, then this new state is accepted. This criterion forces the system toward a state corresponding to a local—or possibly a global—minimum. However, most large optimization problems have many local minima and the optimization algorithm is therefore often trapped in a local minimum. To get out of a local minimum, an increase of the cost function is accepted with a certain probability, i.e., the new state is accepted even though it is a little hotter. The criterion is

$$\exp\left(\frac{\text{CurrentCost} - \text{NewCost}}{\text{Temperature}}\right) > \text{Random}(0,1).$$

The simulation starts with a high temperature, which makes the left-hand side of the equation close to 1. Hence, a new state with a larger cost has a high probability of being accepted.

The change in temperature is also important in this algorithm. Let $\beta_n = 1/\text{Temperature}$. Hwang et al. [1990] prove that if $\beta_n/\log n \to 0$ as $n \to \infty$, then $P(\text{NewState}_n \in \text{global extremum}) \to 1$. But in practice, we usually reduce the temperature, according to $\text{temperature}_{n+1} = 0.87\,\text{temperature}_n$, for convenience.

We apply SA to determine both the next direction to move a specified shot and whether a shot should be deleted or not according to our judgment function. In the case of direction determination, we have

$$\%(\text{CurrentCost} - \text{NewCost}) = 2 \times \text{SphereRadius} - \text{PenaltyJudge};$$

and in the case of determining whether to delete a shot or not, we have

$$\%(\text{CurrentCost} - \text{NewCost}) = \text{RatioCovered} - 0.7.$$

After this adjustment, the results and the speed improve dramatically.

Visualization of the Results

This time, we use a tumor image from the Whole Brain Atlas [Johnson and Becker 1997]. Using 20 two-dimensional slices of the tumor, we construct a 3D presentation of it. We visualize the optimization process in **Figures 5–7**.

Using Matlab, we seek out the contour of the tumor through reading the bitmap of all the pixels. Finally, we get a bitmap of $50 \times 50 \times 50 = 125000$ pixels, which is within the capacity of our computer.

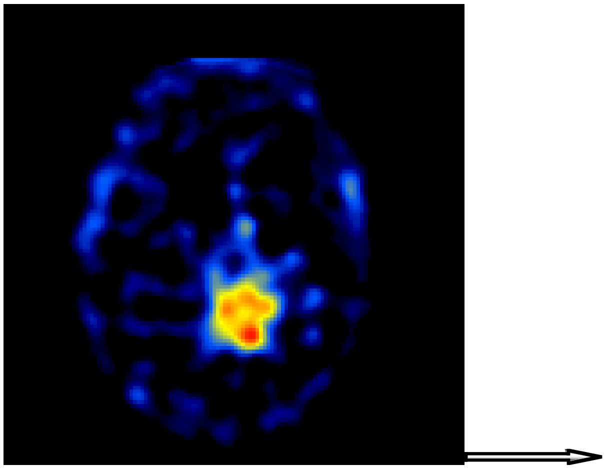


Figure 5. Sample slice of the tumor.

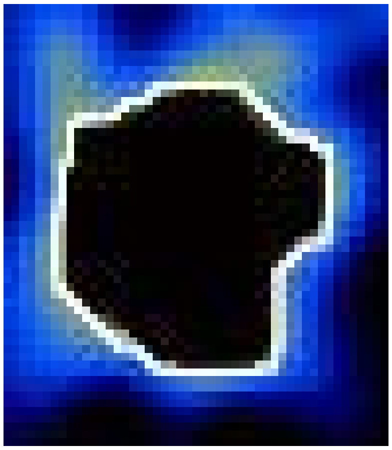


Figure 6. Contour of the tumor

Figure 7 shows the power of our algorithm.

Critical Organ

Finally, we take into consideration the existence of a critical organ. In real medical practice, the maximum dose to critical volumes must be minimized, that is, the dose delivered to any part of a critical organ cannot exceed a particular value

Thus, we modify our judging criterion to meet the avoidance requirement. In our previous algorithm, the criterion is implemented in the function `PenaltyJudge` as a weighted linear combination of the `Covered`, `Miscovered`, and `Overlapped` variables. We change `PenaltyJudge` so that if after a step of the movement, a sphere covers any part of the critical organ, the movement should not be made, even if the `PenaltyJudge` function justifies it (this kind of criterion does not differ from setting a positive value as the maximum dose that can be delivered to a critical part). We can simply give the covered critical part an infinite weight in the linear combination to achieve this goal, which is a demonstration of the flexibility of our program.

The results generated after this change do not differ significantly from the previous ones (because our heuristic algorithm also tries to avoid the protruding of the shots as much as possible), but the existence of critical organ does pose a negative contribution to the final strategy of the treatment planning.

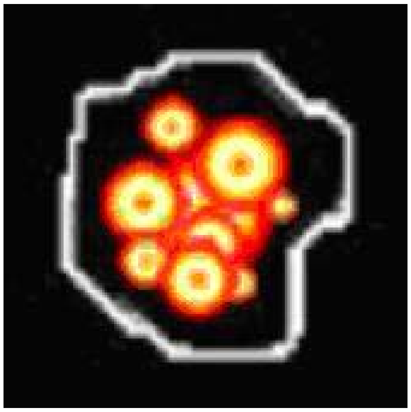


Figure 7a. Initial setting.

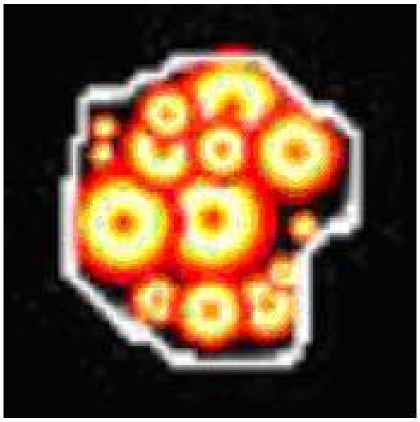


Figure 7b. The scatter process.

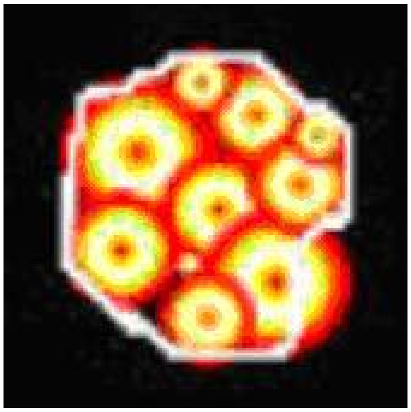


Figure 7c. Deletion and further scattering of the spheres.

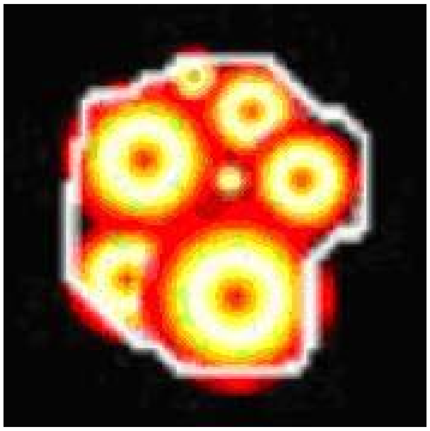


Figure 7d. Final effect; 12 shots are used to cover this large tumor.

Reoptimization

There are two aspects in the reoptimization of our model: the improvement of the quality of the final solution, and the efficiency of the algorithm.

The random starting position of the packing spheres significantly affects the performance of the solution. Although we use a technique to improve the soundness of the starting position, the starting position can still be unsatisfying, since the tumor can be of any irregular shape and size. For example, in **Figure 3**, our program sometimes generates an inferior solution; the result we present is the best from three executions.

To get out of this dilemma, for each starting position we repeat three times the search for the optimal distribution and select the best solution.

The model could also be easily modified to solve more complex situations:

- consideration of the distribution of the dose of the shots, and
- varying the the radii of the available shots according to a continuous interval.

However, with new factors or more pixels, the program slows down. To speed it up, we can use a stepwise optimizing method—that is, first solve the problem with a coarse approximation, then refine and optimize it.

- In our initial model, we evolve a less good packing solution to a better one by pixel movements. In the modification, we make each step 3 pixels; after the packing has evolved to a stable status, we reset the step size to 1 pixel.
- We minimize the drawback of a large number of pixels by managing an image of a smaller size, i.e., an image in which one pixel represents several pixels of the original volume. We use our model to find an optimal solution for the smaller image, then return to the original data to generate a final solution.

Evaluation: Strengths and Weaknesses

Four characteristics can be used in evaluating the algorithm for planning the treatment: effectiveness, speed, flexibility, and robustness.

Since our algorithm focuses mainly on optimizing the final results to meet requirements 1-4, and the data in **Table 2** show satisfying results, our algorithm achieves the effectiveness goal. Our model also simplifies the decision of a good treatment plan to an optimal sphere-packing problem. By using the heuristic approach and the simulated annealing algorithm, we can find the optimal number of spheres of each kind and their positions in a relatively short time.

In addition, we take full consideration of various factors that affect the efficiency and safety of the gamma knife radiosurgery. By summarizing them to four requirements, we construct a penalty function that decides whether a change in the packing plan is desirable or not. Such a penalty function gives our algorithm great flexibility: If more factors are taken into consideration, we can simply add the contribution of that factor to the function. This flexibility is of great value in practice since the requirements of different patients may vary a lot.

Furthermore, the heuristic method used in our program is general. In real medical practice, when many of the assumptions in this particular problem no longer stand, we can still use our algorithm to get an optimal plan. For example, some medical literature [Ferris et al. 2003] mentions that the actual dose delivered is ellipsoidal in nature rather than spherical; we can simply modify our model to handle this situation by changing the four sphere types to ellipsoidal ones—the main outline of our algorithm needs little change.

Finally, our method is strengthened by simulated annealing, which ensures that our solution can reach the global optimum with great probability.

Though we believe that our model solves the problem, there are some limitations:

- The sphere-packing model is too simple; it fails to consider the real dose distribution and the time required for the shots to deliver enough energy.

- Due to the restriction of the hardware, our final solution to for a target consisting of 1 million pixels needs approximately 30 min on a Pentium IV PC, which means any magnification of the scale of this problem is intolerable.

Extension of the Problem

There are five factors that may affect the effectiveness of the treatment:

- How many shots to use (N)?
- Which size shots (radius)?
- Where to deliver the shots (position)?
- What's the distribution of the dose of a particular shot?
- How long to deliver shots (t_s)?

To improve our model so that it can accommodate more practical situations, shot *weights* must be added.

Our previous model mainly focuses on the first three factors, while our improvement also addresses the last two factors. We can obtain the actual shot weights distribution, since in practice it is easy to measure the relative weight of a dose at a certain distance from the shot center as well as to represent its distribution in a 3D coordination.

We fit a nonlinear curve to these measurements, using nonlinear least squares. Suppose that the function of the curve is $D_s(x_s, y_s, z_s, i, j, k)$, which represents the relative dose at pixel (i, j, k) from a shot centered at (x_s, y_s, z_s). The dose at a certain pixel of the CT/MRI image can be calculated by the function

$$\text{Dose}(i, j, k) = \sum_{(s,r)\in S\times\text{Radius}} t_{s,r} D_s(x_s, y_s, z_s, i, j, k),$$

where $t_{s,r}$ is the duration of a shot.

We make our four requirements more precise and practical by setting numerical limitations to the dose that the tumor, normal tissue, and critical part receive. These limitations, set by the neurosurgeon, vary from patient to patient.

A simple refinement is to modify the diameters in the sphere-packing problem. The diameters are no longer 4, 8, 14 and 18 mm but must be calculated using the function D_s when the specified weight of shot is known. For example, if more than 50% of the shot weight is required for the lesion part, the required diameters can be worked out from $D_s = 0.5$. (We assume that the position of the shot won't affect the distribution of shot weight—only the distance from the shot center determines the weight.) If normal tissue can receive only less than 20% of the shot weight, we calculate the diameters D corresponding to the 80% shot weight. Our conformity requirement is reduced to: The distance between the pixels of normal tissue and shot center must be greater than D.

Higher precision may be achieved using the concept of *isodose curve*: A p% isodose curve is a curve that encompasses all of the pixels that receive at least p% of the maximum dose delivered to any pixel in the tumor. The conformity requirement can be represented as the conformity of such an isodose curve to the target volume. We can also approach the shot weight problem by adjusting the amount of shot time, especially when the target is very close to the critical part. Under such circumstances, hitting the critical part is unavoidable. But we can divide the total time required for the treatment into short spans, so that the dose received by the critical part in one time span will do little harm to it while the cumulative dose can kill the tumor.

Anyhow, any improvement cannot be attained without combining real-world practice and must be balanced with the speed and efficiency requirement.

References

Johnson, Keith A., and J. Alex Becker. 1997. The Whole Brain Atlas. `http://www.med.harvard.edu/AANLIB/home.html`.

Buffalo Neurosurgery Group. n.d. Gamma knife radiosurgery. `http://buffaloneuro.com/radio/radio2.html`.

Center for Image-guided Neurosurgery, Department of Neurological Surgery, University of Pittsburgh. 2003. The gamma knife: A technical overview. `http://www.neurosurgery.pitt.edu/imageguided/gammaknife/technical.html`.

Donovan, Jerry. n.d. Packing circles in squares and circles page. `http://home.att.net/~donovanhse/Packing/index.html`.

Ferris, M.C., J.-H. Lim and D.M. Shepard. 2003. Radiosurgery treatment planning via nonlinear programming. *Annals of Operations Research* 119 (1): 247–260. `ftp://ftp.cs.wisc.edu/pub/dmi/tech-reports/01-01.pdf`.

Ferris, M.C., and D.M. Shepard. 2000. Optimization of gamma knife radiosurgery. In *Discrete Mathematical Problems with Medical Applications*, vol. 55 of DIMACS Series in Discrete Mathematics and Theoretical Computer Science, edited by D.-Z. Du, P. Pardolas, and J. Wang, 27–44. Providence, RI: American Mathematical Society. `ftp://ftp.cs.wisc.edu/pub/dmi/tech-reports/00-01.pdf`

Hwang, Chii-Ruey, and Sheu, Shuenn-Jyi. 1990. Large-time behavior of perturbed diffusion Markov processes with applications to the second eigenvalue problem for Fokker-Planck operators and simulated annealing. *Acta Applicandae Mathematicae* 19: 253–295.

Shepard, D.M., M.C. Ferris, R. Ove, and L. Ma. 2000. Inverse treatment planning for gamma knife radiosurgery. *Medical Physics* 27 (12): 2748–2756.