Shelling Tumors with Caution and Wiggles

Team Number 25

April 14, 2003

Abstract

We discuss a simple model for tumor growth, which results in realistic tumor shapes. We then address the question of efficient irradiation of tumors in terms of packing their volume with disjoint spheres of given radii. We develop three algorithms, based on a shelling method, which give recommendations for optimal sphere packings. These packings are interpreted as treatment plans, and the corresponding radiation dosages are analyzed for various qualities.

We analyze the effectiveness of our algorithms in recommending optimal dosage plans. Several problems with the algorithms are addressed. Finally, we conclude that all of our algorithms have a common difficulty covering very large tumors, and that while one of our algorithms is effective as a means of packing the volume with spheres, it does not translate into an efficient treatment plan.

Contents

1 Introduction

The Gamma Knife is a highly effective means of destroying tumor tissue contained within a person's skull while at the same time causing a minimal effect on the surrounding normal tissue. A series of 201 beams, which send ionizing radiation produced by cobalt-60 sources, converge and deliver to a specific region of the brain a powerful dose of radiation without harming the surrounding tissue.

Before the procedure can be performed it is necessary to determine the position where the beams should converge, so the patient's brain is scanned with an MRI or CT imaging device in order to produce a three dimensional image of the tumor. An optimal treatment plan is determined by analyzing this image and deriving a collection of points inside the tumor. Then doses concentrated at those points are administered by the Gamma Knife resulting in a uniform radiation dose of the tumor, while at the same time not subjecting the surrounding normal tissue to an unhealthy amount of radiation. The process of selecting an optimum treatment plan can be lengthy and difficult.

This paper seeks to automate the process of selecting an optimal treatment, thereby reducing the amount of time taken to select a treatment plan. We describe an algorithm to take this three dimensional image and determine an optimal choice for the position and size of the spheres that will cover the tumor as much as possible, while at the same time minimizing the number of shots utilized.

The basic ingredients of our solution include:

- Three tumor growth models representing three different degrees of non-uniformity
- Three sphere-packing algorithms, each representing a different spin on the concept of 'shelling'
- Testing of the algorithms against three different tumor sizes

2 Exploring the Layout of the Problem

Before we address the problem directly, we must describe some of the parameters involved and some of the simplifications and assumptions that we have made in order to solve the problem.

2.1 Collimators

The gamma knife exposes a person to a very large dose of radiation from Cobalt-60 sources. If this exposure was not controlled in some way, the radiation would kill both the person undergoing treatment and the doctor administering it. In order to direct the radiation, a collimator helmet is introduced to direct the radiation towards a specific point in the patient's head. Cobalt-60 as a radiation source emits photons in an isotropic fashion, so by introducing a long cylinder between the source and the target, the beams are constrained to a certain angular distribution. Over the short distance from the end of the cylinder to the target location, the radiation is confined approximately to a cylinder or column. This collimation of the radiation allows specific targeting and reduced exposure.

The collimator helmet is essentially a spherical array of cylinders, each of which directs radiation through a common point. This configuration allows 201 low intensity beams to enter the brain, and intersect in one well defined location. Since the beams are positioned around the skull in an even manner and they are relatively weak, no part of the brain receives a high dose from any one of the beams, and thus the region of the brain outside where the beams converge is relatively unaffected. However, when the beams intersect in the tumor, a huge dose of radiation is delivered. Because the collimator only has four different settings, we are physically restricted to only four different dose sizes.

2.2 Radiation distribution

Because the Cobalt-60 sources are evenly spaced, their intersection can be represented by a spherically symmetric radiation profile. This profile can be well approximated by error functions [1]. Thus, we define the radial distribution of dose intensity to be

$$f_\rho(d) = \sum_{i=1}^{2} \lambda_i (1 - \mathrm{erf}(\frac{\mathrm{d} - \mathrm{r_i}}{\sigma_\mathrm{i}}))$$

Radius	λ_1	r_1	σ_1	λ_2	r_2	σ_2
2mm	0.6492	1.366	4.414	0.5998	2.662	0.6683
4mm	0.4010	7.037	5.702	0.6489	4.849	1.149
7mm	0.3637	13.97	7.197	0.6578	8.200	1.321
9mm	0.3818	17.68	8.195	0.6346	10.32	1.442

Table 1: Fitting Data (source: [1])

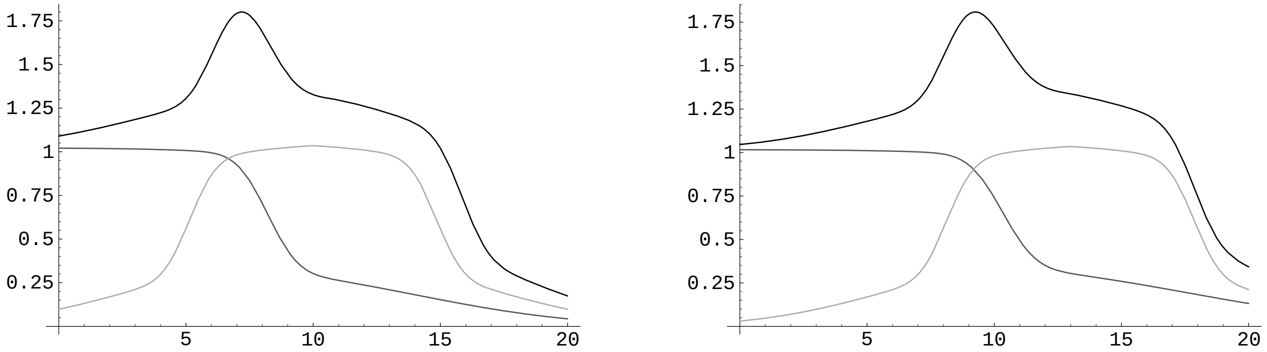

Figure 1: Radiation profiles where spheres meet. In each case, the bottom two curves represent radiation profiles for individual shots, and the bold line represents the total radiation dose received. Left: Spheres of radii 7 and 4 centered at 0 and 11. Right: Spheres of radii 9 and 4 centered at 0 and 13.

where λ_i, σ_i, r_i all depend on ρ, the radius of the sphere, and d is the distance from the center of the sphere. By fitting the values of λ, σ, and r to experimental data, an accurate model for radiation intensity is generated. We follow the results of [1], which can be found in Table 2.2.

Since the tumor should not be over-irradiated, due to observed clinical complications, it would be impractical to design an algorithm which would optimize the placement of these intersections (called shots). This is because it would be computationally intensive and extremely difficult, and the patient must be treated immediately after data collection. Thus, a placement method must be developed which is both accurate and quick. By approximating the shots from collimator helmets as spheres, a much more approachable problem may be formulated: How can we pack the tumor with as many spheres as possible? By using spheres of the same radius as that of the collimator helmet, we agree very well with the drop off point of the radiation distribution defined above. Thus by packing the tumor, we can optimize the placement and number of shots.

2.3 Isosurfaces and isodose

The constructs that define habitability within our brain space are the isodose surfaces, or isosurfaces. These surfaces are the level sets of the scalar field that makes up the radiation distribution in the brain space. Every point on one side of the surface has a higher dosage than any point on the other. In an ideal case, the edge of the tumor would be the Isodose surface, with every point inside the tumor receiving 100% of the prescribed radiation, and every point not in the tumor receiving none. However, this is infeasible in the real world, and so some compromise must be reached. By comparing the regions within isosurfaces to the tumor, it is easy to see how much tumor has been cleared away, how much tumor is left, and how much brain has been sacrificed.

2.4 Digital images of tumors

Typically, an MRI delivers an image of the brain case, with a resolution of approximately 1 cubic millimeter. Since the brain is about one cubic decimeter, our tumor image sits inside a $100 \times 100 \times 100$ voxel space, where each voxel represents 1mm^3. We assume that the shape given by the imaging software is an accurate and conservative approximation of the tumor, so that if we subject voxels denoted as tumorous to doses of radiation we are not damaging normal brain tissue.

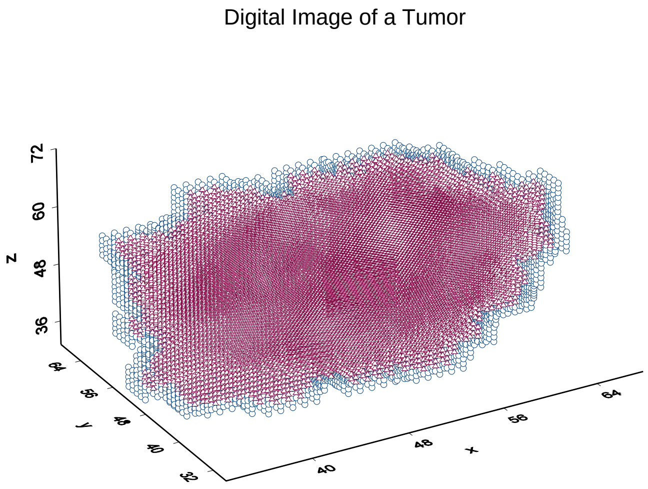

Figure 2: The digital image of an Amok tumor generated by our algorithm

2.5 Problem restatement

Given these considerations, we may restate the problem before us as follows. Given a digital image of tumor tissue, produce a list of at most fifteen spheres (centers and radii) such that:

- The spheres are entirely contained within the tumor tissue
- Each radius is either 2, 4, 7, or 9mm
- The spheres are disjoint (or nearly so to avoid overdosing)
- The total volume of the spheres is as close to that of the tumor as possible

3 Tumor Generation Model

Obviously, before we can effectively build an algorithm to solve this problem, we need digital images of tumors on which to test it. Attempts to find images of actual tumors were unsuccessful, so it became necessary to generate our own. Because we are using fabricated images, we have tried to produce as broad a variety of shapes and sizes as possible, as well as to adhere to existing models of tumor growth.

To generate digital images of tumors, we created an algorithm to simulate their growth in time-steps. This algorithm makes use of a cellular automaton to model cellular division [2]. For our lattice, we take a simple three-dimensional grid. Ideally the lattice would have a structure similar to that of a biological fabric, but we take the simplified model for several reasons. First, it is computationally convenient—not only is simulation reduced to an elementary level, but the output resembles a digital image without any further processing. Second, we are interested only in studying the final shape of tumors, and not their growth. Our algorithm is sufficient to produce volumes representative of tumor shapes.

Each node of the lattice represents an idealized mathematical cell. We begin by placing a single cell in the center of the space, allowing the cells to 'divide' (generate a new cell) at each time step until the tumor reaches the desired size. Whether or not it divides will depend on several factors which vary between the three different generation methods discussed below.

Note that in all cases, we smooth out unrealistic or undetectable holes in each tumor after it has finished growing to the desired size. It is still theoretically possible for a tumor to reconnect with itself in a toroidal shape, but this does not happen in practice.

radius	9mm	7mm	4mm	2mm
coverage	3000mm^3	1300mm^3	250mm^3	12mm^3

Table 2: Shot Radius vs. Coverage

3.1 The Uniform Pressure tumor

This simulation supposes that the tumor is growing in a uniformly pressurized environment. The probability for a given cell to divide at each time step is proportional to $1 - r/R$, where r is the distance from the cell to the origin, and R is a bound on the radius of the tumor. Since we are interested only in the final tumor shape and not development, the constant of proportionality is essentially irrelevant. The resulting tumor shape is generally a ragged sphere.

3.2 The Varied Pressure tumor

Suspecting that the preceding model may be too symmetric, we created a second algorithm which varies the pressure against the tumor, corresponding to the affect the surrounding area of the brain would have on the growth of a tumor. To do this, we specify certain random points x_i as pressure points, and curb tumor growth near these points. The probability for a given cell x to divide is now proportional to the product $(1 - r/R) \prod |x - x_i|/M$, where M is the maximum width of the tumor space, and r and R are as before.

Tumors generated with this model are slightly more elongated and pointed than before. Some curve into a lima-bean shape around the pressure points.

3.3 The Running Amok tumor

One model of tumor growth found in [4] postulates that once a tumor begins to jut in one direction, cells in the out-cropping are more likely to receive nutrients from the surrounding brain tissue, and hence are more likely to grow, whereas the less extended regions of the tumor are less likely to grow. The end result is that the tumor is more likely to grow in a direction that it has grown out before, and results in a tumor that is spiny in appearance.

Thus, in this third model, we have removed the outside pressures and specified that more recently generated cells shall be more likely to divide than older ones. The resulting shape is a bulky tumor with tendrils showing recent and active growth.

4 Packing Spheres into Volumes

The sphere packing problem is ages old and yet unsolved. The very problem with which we are faced has been shown to be NP-complete [6]. When faced with the problem of packing rounds into a barrel, the military has been rumored to pack them as tight as they can and proceed to drive the barrel in a large truck over a bumpy road. This would clear up enough space for another handful of shots.

Unfortunately, the spheres which we are packing do not carry mass, and we certainly may not propose to drive the Gamma Knife, with its patient, over a bumpy road. For these reasons, we implement an algorithmic approximation.

4.1 The general approach

Our plan is based on two fundamental principles. The first is

- Largest spheres first

That is, in all of our algorithms, we pack large spheres into the volume before considering smaller ones. The reason for this is the steep fall-off of coverage as shot radius decreases (see Table 2). One may notice that it takes twice as many 7mm spheres to do the job of a 9mm sphere, consuming more diametric space in the tumor and complicating the dosage plan. Since one of our stated goals is to minimize the number of doses, the principle of utilizing larger spheres first is clearly a good one.

The next principle which we have tried to follow is

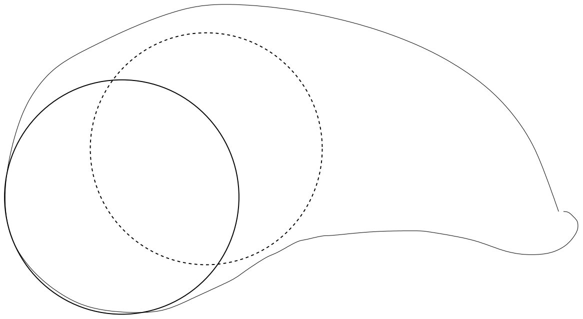

Figure 3: The sphere on the left is a better choice than the one on the right.

- Spheres should hug the tumor edge

We have assumed that shots from the Gamma Knife deliver a spherical dose. However, due to physical constraints of the Gamma Knife, we may consider only shots of radius 2, 4, 7, and 9mm. With this limitation in mind, note that it is very likely for there to be a many-way tie for the largest allowable sphere fitting inside a given volume. But not all choices are created equal—a centrally placed sphere may preclude the possibility of fitting other large spheres in the volume (see Figure 3).

For example, it would be unwise to place a 9mm shot at the center of a spherical tumor which is 16mm in radius, as this would not allow us to place any 7mm spheres into the tumor, violating our previous principle.

4.2 A discrete sphere packing model

Because the tumor data is given to us as a digital image, we chose to pack spheres into the image directly. That is, we select spherical subsets of voxels from the given tumor, and consider the optimum means of placing spheres centered at voxels inside the tumor, that do not intersect with each other (or with the edge of the tumor.)

The algorithm acts by a 'shelling method' [5], which we describe in detail. The first step is to identify the boundary of the tumor, marking the voxels on a bounding surface with value of 0. We then assign every tumorous voxel a positive value denoting its distance from the boundary nodes by the following method:

Algorithm 1 The depth evaluation algorithm

```
$X$ is the set of tumorous voxels
$E_0$ is the set of boundary voxels
$d := 0$
while $X \neq \varnothing$ do
  for $x \in X$ do
    if $x$ neighbors a point of $E_d$ then
      move $x$ from $X$ to $E_{d+1}$
    end if
  end for
  increment $d$
end while
```

Now, the sphere we wish to place inside the tumor should be as large as possible, but also as close to the boundary as possible. We therefore wish to take for the center of this sphere a point at depth r from the boundary, where r is the greatest of 2, 4, 7, and 9 available. This way, when we shell a sphere at depth r, the shelled area will be along the edge of the tumor as much as possible.

In any case, our algorithm must choose the center x of the sphere from the set E_r. The way this is done will vary through the three different algorithms that we have tested, and will be discussed below.

Having chosen our candidate shell center x, we remove from the tumor the sphere centered at this voxel of radius r. Since our space is made up of discrete points, and the coordinate of a voxel is taken to be the center of that voxel, this translates to removing all voxels in the tumor whose coordinates are within r millimeters of the center of the sphere. If this is the case, then we consider the voxel to be part of the interior of the sphere, and thus removing the sphere from the tumor consists of removing exactly these points.

We then iteratively apply this shelling method to the volume of tumor which remains. The algorithm terminates after it has shelled 15 times, since that is what we have chosen as our maximum number of allowable shots. However, we keep track at each stage of what percent of the tumor has been eliminated so that fewer shots may be recommended if they are deemed sufficient.

4.3 A Hasty algorithm

This is the simplest algorithm. Each time we must choose a center for our shell of radius r, we simply grab one at random. In some sense, this algorithm acts as a control for the next two.

4.4 A Careful algorithm

This next algorithm improves upon the previous one in an obvious way. At each stage we have to select a candidate center x from the set E_r. We do so by determining which choice of center will be most beneficial towards placing the remaining spheres. Since the placement of the largest spheres will have the greatest effect on the percentage of the volume (Again, see Figure 3), we determine which choice of center results in the largest depth by applying the shelling process to what remains of the tumor after the sphere centered at that point has been removed. We want the remaining region to have as high a depth as possible, since this will make it more likely that we will be able to fit large spheres in the surrounding area.

While it would be preferable to compare all of the possible choices, this is in most cases too computationally intensive. We therefore select a number of our centers in E_r, in this case 5, compute the depth of the volume of the tumor minus the sphere centered at that point in each case, and from these 5 pick the one whose remaining tumor has voxels with the highest depth.

4.5 A Wiggling algorithm

Prototyping these two algorithms revealed that our depth search algorithm does not always choose sphere centers which are as close to the tumor boundary as possible. The root of this problem will be discussed in detail below, when we evaluate the different algorithms.

The Wiggling algorithm was developed as a partial solution to the this problem. It rests on the theory that the depth algorithm may still in fact be used to find spheres which closely hug the boundary.

To do this, the Wiggling algorithm chooses a voxel x from E_r at random, and places a sphere of radius r centered at the voxel inside the tumor. It then transports the sphere, one unit at a time, starting off in a random direction and continuing until the sphere meets the boundary as closely as possible. (That is, the algorithm simply transports the sphere until it discovers that the next step would land it outside the tumor.) By no accident, this method very much resembles that of the truck on the bumpy road.

Unfortunately, the methods of this algorithm cannot be combined efficiently with those of the previous algorithm. Any care exerted would simply be undone by wiggling.

4.6 Output

One cannot distinguish from the three dimensional images the differences between our three algorithms. Figure (4) shows a cut-away plot of tumor wall, together with 15 spheres selected by the careful algorithm.

5 Analyzing Sphere Placement Plans

5.1 Computing dose distributions

After completing calculations using discrete spheres and voxel models of tumors, the actual dose distribution from the proposed treatment plan is calculated. Using the centers of the spheres, and the radially symmetric radiation

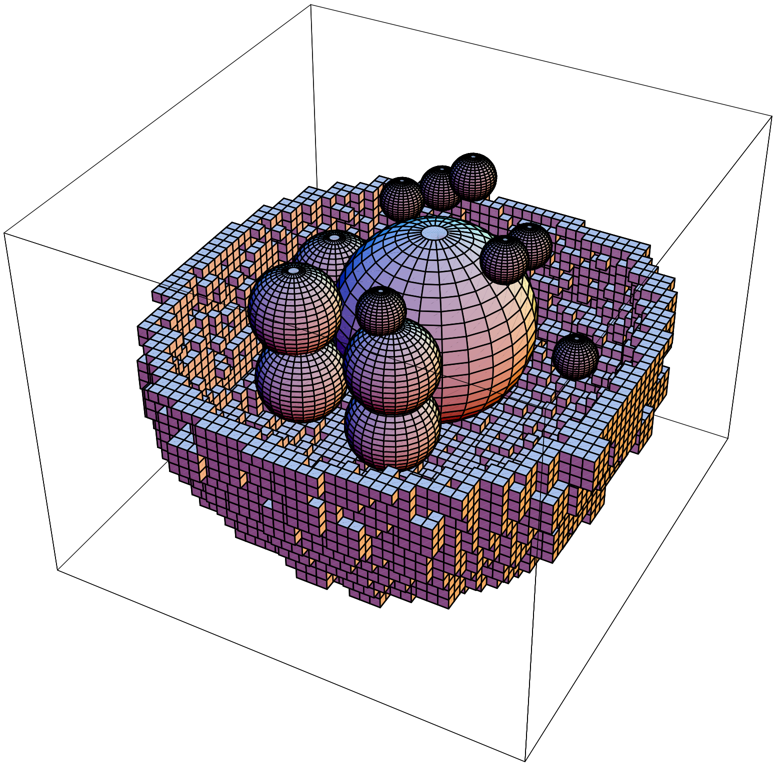

Figure 4: Spheres packed in a tumor

functions presented earlier, a scalar field is constructed in brain space defined by the formula

$$D(x,y,z) = \sum_{i=1}^{n} f_{r_i}(\sqrt{(x-x_i)^2+(y-y_i)^2+(z-z_i)^2})$$

where f is the equation given in equation (2.2), and (x_i, y_i, z_i) is the center of the i^{th} sphere with radius r_i.

This formula is then approximated with discrete values at the center of each voxel. These values are simply the value of D at that point. Of course, it would be more accurate to integrate over the entire cube, and use the computed value as the representative number for the voxel, but this was too computationally intensive and the accuracy gained on test cases was minimal. Using this scalar field, it is possible to calculate isosurfaces of any value, and the volume of the region contained in this isosurface. By taking various different isosurfaces, the efficiency of the spherical covering plan can be calculated. The amount of damage to the surrounding tissue is also available.

5.2 Using contour profiles to determine the number of shots

Because each step in our algorithm is executed in the same way whether or not there will be a subsequent shot, computing every shot and then using some set of parameters to determine the value of each successive shot enables us to compute the ideal number of shots for each tumor. Since neither of these methods is incredibly time consuming (it took us 10-15 minutes of computing time to analyze one of the large tumors in full), it is certainly feasible to run this process in the hospital and deliver a dose profile to a patient in moments.

5.3 Visualizing dose distribution

In order to facilitate the visualization of dose distribution, we constructed a program to slice the field of dose distribution and superimposed an image of the tumor on it, as in Figure 5. By taking multiple slices, it is possible to see a three dimensional isosurface profile of the tumor. The white areas in the images of Figure 5 refer, from

Table 3: Percentage of tumor receiving 30/40/50% of shot intensity by the Hasty algorithm

Shape	Size (mm^3)		
	10,000	20,000	36,000
Uniform	96/86/71	96/86/68	96/86/68
Varied	95/82/70	96/89/70	76/69/54
Amok	94/87/70	91/82/63	78/66/49

Table 4: Percentage of tumor receiving 30/40/50% of shot intensity by the Careful algorithm

Shape	Size (mm^3)		
	10,000	20,000	36,000
Uniform	96/85/69	98/91/70	99/92/73
Varied	99/96/84	92/82/69	92/86/72
Amok	94/85/64	88/79/62	79/66/50

Table 5: Percentage of tumor receiving 30/40/50% of shot intensity by the Wiggling algorithm

Shape	Size (mm^3)		
	10,000	20,000	36,000
Uniform	99/95/82	91/83/69	88/81/67
Varied	99/96/84	92/82/69	92/86/72
Amok	91/85/72	79/71/56	76/65/51

top-left clockwise, to the regions of the tumor that receives a radiation dosage above 30, 50, 80%, and the region of potential overdosing, i.e a region with an intensity higher than 2.5 times that of a single shot, referred to as a 'hotspot.'

5.4 More information from contours

The contours also allow us to see the effects of sphere placement in a way that the discrete functions of spheres cannot. By examining the various points of the space, we can see that even though the spheres do not intersect, there can be places with radiation exposure intensity higher than that of one shot.

6 Applying the Algorithm

Under the problem conditions, we were allowed a maximum of 15 spheres. In all of our simulations, it was possible to place 15 spheres inside the tumors, since they were of relatively high volume compared to a sphere, especially a sphere of radius 2.

From each of our three tumor growth models, we produced 30 tumor images: 10 of size 10,000mm^3, 10 of size 20,000mm^3, and 10 of size 36,000mm^3. We then tested each of our three algorithms against all 90 tumors.

We performed two operations: First, the spherical dose pattern was used to create a scalar field of radiation exposure values throughout the 100×100×100 brain space. This field was then used to calculate the exposures of every cell within our modeled brain and tumor, and decide which of these cells survived the treatment. Using a 30% contour as our cutoff for death and survival, and further calculating the percentage exposures to other isodose cutoff values, we obtained a series of exposure percentages for our various tumor shapes and sphere placement algorithms. Reducing this data leads us to Tables 3, 4, and 5.

Second, we take the spherical profiles generated by the placement algorithms, and estimate the radiation exposure values to all of the brain space for each generation of spherical dose pattern created by the algorithm. When the total dosage of the tumor reaches a specified value, in this case 30%, the spherical dose profile of that

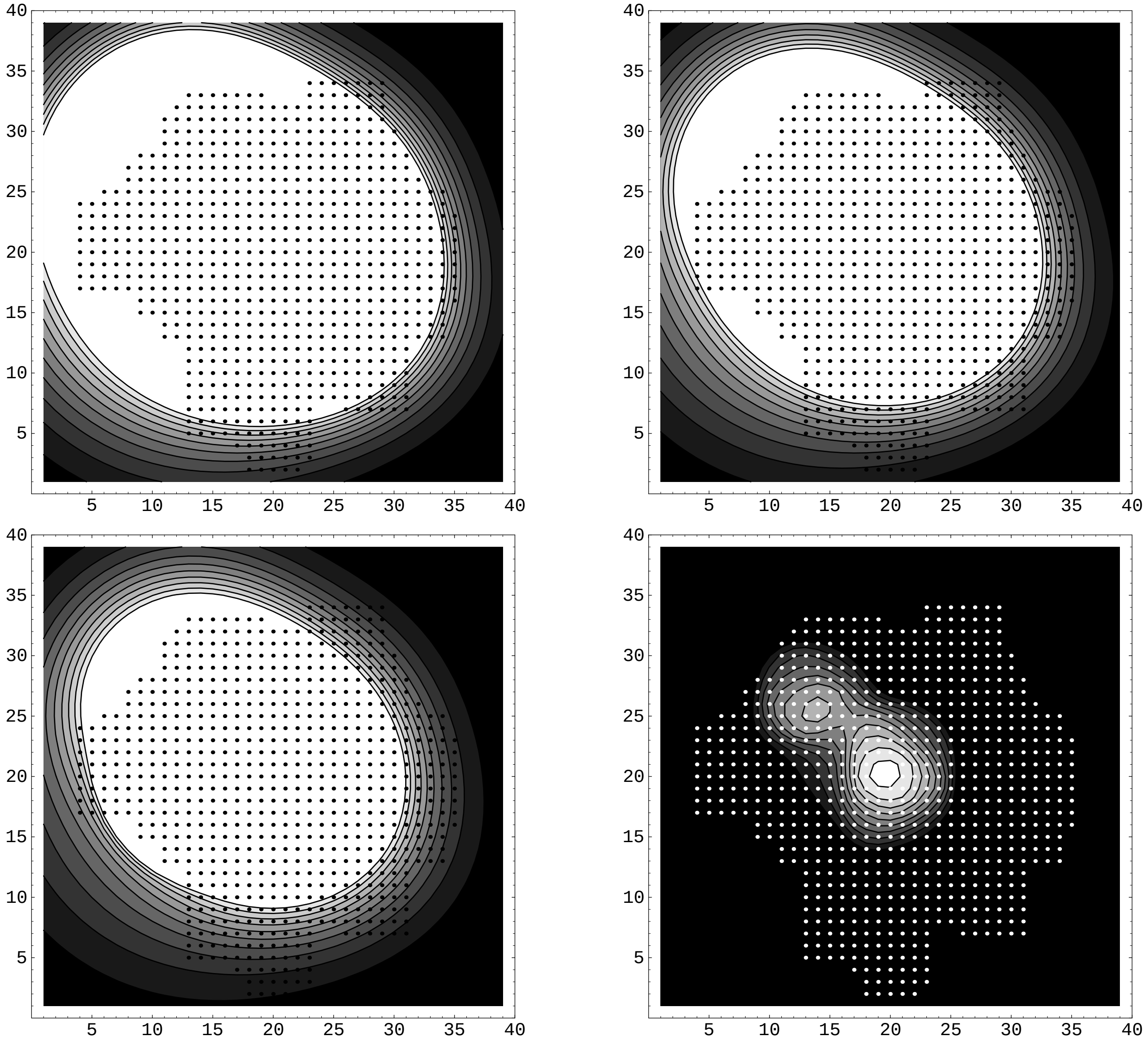

Figure 5: A single cross-sectional slice of a tumor (represented by dots). Top-left: 30% cut-off. Top-right: 50% cut-off. Bottom-left: 80% cut-off. Bottom-right: shows only hot spots.

Table 6: Approximate number of shots needed to kill 90% for the Hasty algorithm

Shape	Volume (mm^3)		
	10,000	20,000	36,000
Uniform	8	15	7
Varied	15	10	9
Amok	10	–	–

Table 7: Approximate number of shots needed to kill 90% for the Careful algorithm

Shape	Volume (mm^3)		
	10,000	20,000	36,000
Uniform	7	12	7
Varied	–	12	9
Amok	10	–	–

Table 8: Approximate number of shots needed to kill 90% for the Wiggling algorithm

Shape	Volume (mm^3)		
	10,000	20,000	36,000
Uniform	10	10	15
Varied	9	11	15
Amok	15	–	–

generation is recorded. Thus the total number of shots delivered to the patient is minimized while maintaining effective coverage of the tumor. Reducing this data leads us to Tables 6, 7, and 8. Note that a dash means that the algorithm was not able to fill 90% of the volume with 15 spheres.

7 Evaluation of Methods and Conclusions

In this section we compare the strengths of our sphere packing algorithms.

7.1 Difficulties reaching 90% coverage

Even for simple volumes, we have concluded that it can be impossible to actually cover 90% of a tumor with shots. To see this, consider a simple volume which is a sphere of 10mm radius. We may shell this with a 9mm sphere, which would leave no room even for a 2mm sphere and would still cover less than 3/4 of the volume. In other words, if one were solving the purely mathematical problem of packing given spheres into $20,000\text{mm}^3$ volumes, one would almost certainly not succeed.

Simply put, the only way our algorithms are able to destroy 90% of the tumors is by allowing hot-spots in the tumor.

7.2 Incompatibility of square lattices and Euclidean distance

Even worse, no instance of our algorithm can see that it is possible to put a 9mm shot in the 10mm sphere. This is because the depth evaluation algorithm assigns depths to the tumorous voxels that does not entirely correspond to the physical distance between the voxels and the tumor edge. One sees in Figure 6 (at left) that when voxels happen to differ via the coordinate axes, then the depth differential coincides with the distance between the voxels. However, in every other case the depth does not quite accurately measure distance. For example, two

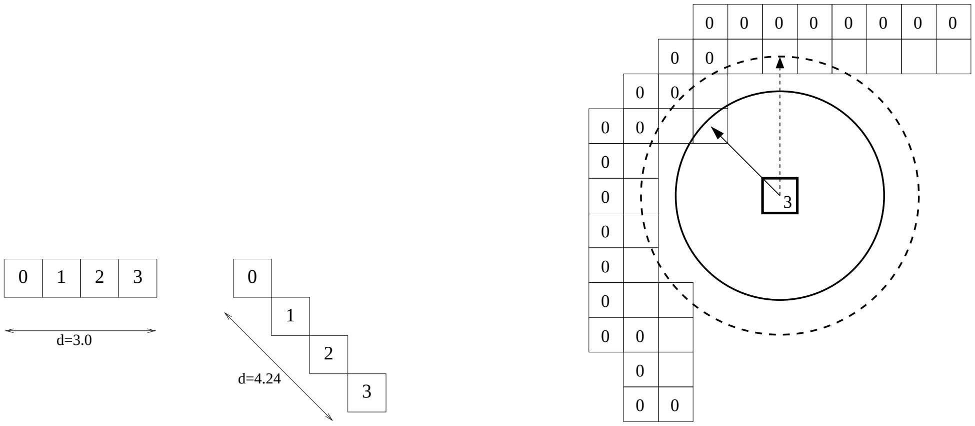

Figure 6: (Left) Metric inconsistencies. (Right) A schematic highlighting a weakness of our depth model.

voxels that are diagonally adjacent will have a difference in depth of 1, whereas the Euclidean distance between the centers of the two voxels is $\sqrt{3}$.

The result in many situations that our centers have a depth given to them by the shelling process that is not an accurate representation of their physical position inside the tumor. For example, in Figure 6 (at right), the highlighted voxel has a depth of 3, and so if we choose that voxel as the center of our circle, then we would place a circle of radius 3 at that point to cover the tumor. But it is clearly possible to squeeze a circle of radius 4 into the tumor. The problem is exaggerated further in three dimensions.

Our Wiggling algorithm was developed as an attempt to find a partial solution to this problem. By allowing the spheres to move toward the edge of the tumor, we hoped to pack the spheres as tightly as possible.

7.3 Conclusion/Recommendation

First note that for small or medium sized tumors, the Careful algorithm is comparable to the Hasty Algorithm, and thus the Hasty algorithm is preferable for time considerations.

We also see that all of the algorithms, but especially the Wiggling algorithm, have difficulty handling the very large tumors. They are just too large compared to the sphere sizes, and so it is unlikely any combination of spheres would fill the volume of the tumor to the amount that is required by the problem statement.

However, we see that for reasonably sized tumors, the Wiggling algorithm is quite effective, and works especially well towards filling in the Varied tumors. This is because the Varied tumors bulge around the pressure points, forming pockets that are ideal for nesting spheres. When the spheres wiggle, they are more likely to settle into these regions, and thus pack into the tumor tightly. Furthermore, the Wiggling method is a much faster running algorithm, especially compared to the Careful algorithm.

In fact, the Wiggling algorithm does such a good job of packing in spheres, that if we do not put a constraint on the total number of spheres to be used, then the Wiggling method tends to pack a very high number of spheres, resulting in a large dosage. Furthermore, since the spheres are packed to the edge, the radiation from the shot at a sphere will spill out of the tumor into the surrounding normal tissue. We thus conclude that while the Wiggling algorithm is a very good method of sphere packing, it is not a good means of distributing the dosage uniformly and safely across the tumor.

To summarize:

- None of these algorithms should be used to come up with treatment plan for tumors of very large size. Very large tumors should be treated in multiple sessions.
- For moderately sized tumors, the Hasty algorithm should be used in preference to the Careful algorithm.
- The Wiggling method is a speedy and effective means of deriving a sphere packing algorithm, but does not translate well into an effective and safe treatment plan. However, we do recommend it for tumors that are similar in shape to the Varied tumor, because it did prove effective in producing an optimal plan in that situation.

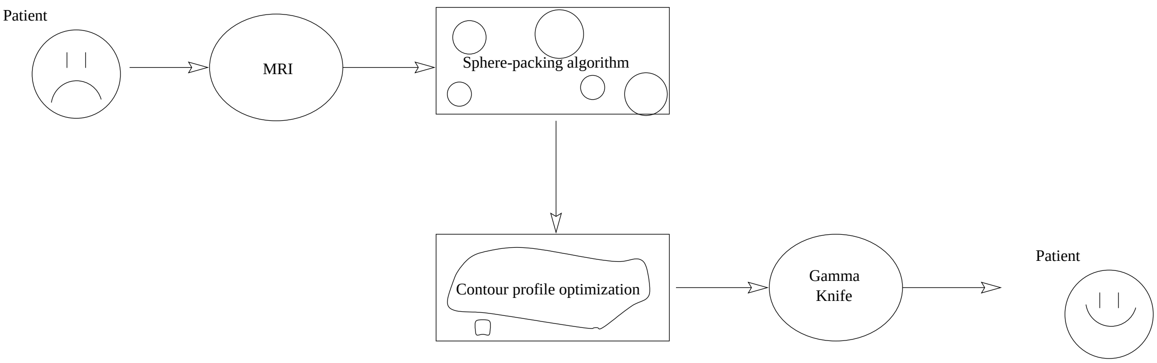

Figure 7: Schematic diagram of treatment process

Thus, by adopting a computerized approach to dose planning as detailed in this paper, the ordeal faced by a victim of brain tumors, as illustrated by the schematic in Figure 7, will be greatly expedited, and recovery can begin as soon as possible.

References

[1] Ferris, Michael C. and David M. Shepard. "Optimization of Gamma Knife Radiosurgery." *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, vol. 55, 2000.

[2] Kansal, A. R., et. al. "Simulated Brain Tumor Growth Dynamics Using a Three-Dimensional Cellular Automaton." *Journal of Theoretical Biology*, 2000.

[3] MkKeran, R. O., and G. D. Wilde. "Mathematical Models of Glioma Growth." *Brain Tumors*. Butterworth & Co., 1980.

[4] Newman, William I., and Jorge A. Lazareff. "A Mathematical Model for Self-Limiting Brain Tumors." `http://www2.ess.ucla.edu/ñewman/LGA.pdf`.

[5] Wagner, Thomas H., et. al. "A Geometrically Based Method for Automated Radiosurgery Planning." *International Journal of Radiation, Oncology, Biology, Physics*, vol. 48.5, 2000.

[6] Wang, Jie. "Packing of Unequal Spheres and Automated Radiosurgical Treatment Planning." *Journal of Combinatorial Optimization*, 1999.

[7] Wu, Q. Jackie. "Sphere Packing Using Morphological Analysis." *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, vol. 55, 2000.