The Gamma Knife Problem
伽玛刀问题

Darin W. Gillis
David R. Lindstone
Aaron T. Windfield

COMAP Mathematical Contest in Modeling
February 10, 2003
University of Colorado Boulder, CO

翻译：周吕文

[0]若发现翻译问题，请邮件告知，谢谢。

Abstract

The human brain is the most important and vulnerable organ in the body. Therefore, the treatment of brain cancer requires extreme care. Non-invasive Gamma Knife radiosurgery treatment attacks brain tumors using spherical radiation dosages (shots). We have developed methods that design optimized treatment plans using four fixed-diameter dosages. If healthy tissue is damaged during treatment, the damage is irreversible. Thus, our algorithms strictly adhere to the following rule: shots cannot violate tumor boundaries or overlap each other. From a mathematical perspective, the problem becomes a matter of filling an irregularly-shaped target volume with a conglomeration of spheres. We make no assumptions about the size and shape of the tumor. By maintaining complete generality, our algorithms are flexible and robust. The basic strategies of the algorithms are deepest sphere placement, method of steepest descent, and adaptation. We designed representative 3D models to test our algorithms. We found that the most efficient packing strategy was an adaptive algorithm that utilized our steepest descent technique with an average coverage percentage of 39.86% over 100 test cases while not threatening any healthy tissue. One variation on the steepest descent method covered 55.92% of one test case, but had a large standard deviation across 100 test cases. It also produced results 4 times as fast as the adaptive method.

人脑是人体内最重要，也是最脆弱的器官。因此，脑部肿瘤的治疗需要非常小心。非侵入式伽玛刀放射外科治疗使用球形放射靶点来杀死颅内肿瘤。针对只含四种固定直径靶点的伽马刀系统，本文开发了一套算法用于优化伽马刀手术治疗方案。如果健康组织在治疗过程中受损，损害是不可逆转的。因此，本文的算法严格遵守以下规则：靶点不能超出肿瘤边界或彼此重叠。从数学的角度来看，这个问题变成了用球体组合来填充不规则形状目标体积的问题。本文关于肿瘤的大小和形状不作任何假设。通过保持完整的通用性，本文的算法灵活而稳定。本文算法的基本策略包括深度优先法、最速下降方法和自适应法。我们构造了典型的三维肿瘤模型来测试本文的算法。我们发现最有效的填充策略是一种基于本文的最速下降技术的自适应算法，在 100 多个测试算例中，该算法在不威胁任何健康组织的情况下平均覆盖率为 39.86% 。最速下降法在其中一个测试算例中覆盖了 55.92% 的肿瘤区域，但是在 100 个测试算例中表现出很大的标准差。同时，最速下降法给出结果的速度是自适应法的 4 倍。

Contents

1 Introduction | 引言

Each year approximately 1.2 million Americans are diagnosed with cancer, and nearly 40% of these people will undergo radiation therapy [1]. Stereotactic radiosurgery provides an efficient method of treating brain tumors while avoiding invasive surgery and protecting surrounding tissue. We will focus on the implementation of the gamma knife, a specific modality of stereo-tactic radiosurgery. The goal of radiosurgery is to destroy cancerous cells and preserve normal structures.

每年大约有 120 万美国人被诊断出患有癌症，其中将近 40% 的人需要接受放射治疗 [1]。立体定向放射手术提供了一种有效的治疗颅内肿瘤的方法，同时避免了侵入式手术并保护了肿瘤周围正常的脑组织。我们主要关注于伽玛刀手术的实施，伽马刀是放射外科治疗手术的一种。放射外科的目标是破坏癌细胞并保证正常脑组织不受侵害。

1.1 Background Information | 背景

The subjects of cancer and radiation therapy are extremely broad, and generally beyond the scope of our approach. However, some basic knowledge is necessary in order to fully understand this report.

癌症和放射治疗的对象非常广泛，一般超出我们的方法范围。不过，为了能够读懂本文，需要掌握一些基础知识。

1.1.1 Brain Tumors | 颅内肿瘤

Brain tumors exhibit a wide variety of sizes and shapes. We found that the average volume of a tumor operable by radiosurgery was about 15 cm^3 [2]. We will generate 3-dimensional tumor models of approximately this volume with varying physical dimensions.

颅内肿瘤具有各种大小和形状。我们发现接受放射治疗的肿瘤平均体积约为 15 cm^3 [2]。根据这个体积，我们将生成具有不同物理尺寸的三维肿瘤模型。

1.1.2 The Gamma Knife | 伽马刀

The gamma knife unit consists of 201 individual cobalt-60 radiation sources situated in a helmet. The 201 beams converge at an isocenter creating a spherical dose distribution (“shot”). Four sizes of spheres are possible: 4, 8, 14, and 18 mm in diameter. A radiosurgery plan is used to map out shots that will destroy the tumor without harming the patient. Following successful treatment, surviving cancer cells lose their ability to grow. In fact, many partially destroyed tumors shrink or even disappear in time [3].

伽马刀单元包括位于头盔中的 201 个独立的钴-60 辐射源。201 个光束在一个等中心会聚，形成球形剂量分布（“靶点”）。四个可用大小的球体（“靶点”）为：直径 4、8、14 和 18 mm。所谓放射手术方案就是确定出破坏肿瘤而不伤害病人的靶点直径和靶心位置。在成功治疗之后，存活的癌细胞将失去生长能力。事实上，许多被部分破坏的肿瘤会缩小甚至消失 [3]。

2 The Problem | 问题重述

We are designing algorithms for constructing radiosurgery plans. The plans should arrange radiation doses such that tumor destruction is maximized, healthy tissue is protected, and hot spots are avoided. Thus, the algorithms are subject to the following constraints:

- Prohibit shots from penetrating outside the target area
- Prohibit overlap of shots, preventing hotspots
- Maximize the percentage covered in the tumor, or target volume
- Use a maximum of 15 shots

我们正在设计构建放射外科治疗方案的算法。治疗方案应该确定辐射剂量，使肿瘤破坏最大化，保护健康组织，避免热点。因此，算法受以下限制：

- 禁止靶点超出目标区域外
- 禁止靶点重叠，防止热点
- 最大化靶点覆盖肿瘤的百分比或目标体积
- 最多使用 15 个靶点

2.1 Assumptions | 假设

- The tumor is homogeneous; it is equally productive to treat any part of the tumor
- The tumor is modeled discretely using a three-dimensional image
- No assumptions are made about the shape of the tumor
- Tumor cells have either been radiated or not; there are no partial dosages

- 肿瘤是均匀的; 治疗肿瘤的任何部分效果都是相同的
- 使用三维数字图像对肿瘤进行离散建模
- 没有关于肿瘤形状的假设
- 肿瘤细胞要么已经被辐射，要么没有被辐射; 没有部分剂量的概念。

3 Problem Approach | 问题分析

We have divided the problem into three different pieces:

1. Create a variety of 3-dimensional brain tumor models
2. Develop and refine sphere-packing algorithms
3. Test and compare algorithms using tumor models

我们把这个问题分成了三个不同的部分：

1. 构造多种三维颅内肿瘤模型
2. 开发和改进球体填充算法
3. 使用肿瘤模型测试和比较算法

4 Data Models | 数据模型

Our data consists of $100 \times 100 \times 100$ arrays that represent a 1000 cm^3 space around the brain tumor. We used 1's to indicate tumor presence in that space, and 0's to represent healthy brain tissue. This paper commonly refers to each element of the tumor matrix as a voxel (term used for a 3-Dimensional pixel). Each voxel represents 1 mm^3 of brain tissue [4]. We populated the arrays with tumor models as described below.

我们的数据由 $100 \times 100 \times 100$ 数组构成，代表脑肿瘤附近 1000 cm^3 的空间。我们用 1 表示肿瘤存在的空间，用 0 表示健康的脑组织。本文通常将肿瘤数组的每个元素称为体素（体素是体积元素的简称，或者三维像素）。每个体素代表 1 mm^3 的脑组织 [4]。我们用如下所述的肿瘤模型初始化数组。

4.1 Sphere Tumor Model | 球体肿瘤模型

Our first model was based off of the simple equation for a sphere, $(x—x_0)^2 + (y—y_0)^2 + (z-z_0)^2 = r^2$, where the center of the sphere is represented by (x_0, y_0, z_0) with radius r. We filled in the voxels representing the tumor by applying the inequality, $(x—x_0)^2 + (y—y_0)^2 + (z-z_0)^2 \leq r^2$ throughout the test volume.

我们的第一个模型是基于简单的球面方程 $(x—x_0)^2 + (y—y_0)^2 + (z-z_0)^2 = r^2$，其中球体中心是由 (x_0, y_0, z_0) 表示，半径为 r 。我们通过在整个测试体积中应用不等式 $(x—x_0)^2 + (y—y_0)^2 + (z-z_0)^2 \leq r^2$ 来填入代表肿瘤的体素。

4.2 Ellipsoid Tumor Model | 椭球体肿瘤模型

The ellipsoid model used the same principle as the spherical model. We used the inequality $\frac{(x—x_0)^2}{a^2} + \frac{(y—y_0)^2}{b^2} + \frac{(z-z_0)^2}{c^2} \leq r^2$ to represent the interior of an ellipsoid. The spherical and ellipsoid models were used as a basis for the mutated sphere tumor model.

椭球模型使用与球模型相同的原理。我们用不等式 $\frac{(x—x_0)^2}{a^2} + \frac{(y—y_0)^2}{b^2} + \frac{(z-z_0)^2}{c^2} \leq r^2$ 来表示一个椭球的内部。球形和椭球体模型将作为变异球体肿瘤模型的基础。

4.3 Mutated Spherical Tumor Model | 变异球体肿瘤模型

We learned from research that tumor shapes can be modeled by unions of ellipsoids [5]. Thus, our most accurate model is created by intersecting several different ellipsoids at random locations. We start with a small spherical tumor (see Spherical Tumor Model). Then we create three discrete uniformly distributed random variables U_x, U_y, and U_z. (U_x, U_y, U_z) represents a randomly chosen voxel within the sphere. This point becomes the center of an ellipsoid that is added to the tumor. The a, b, and c parameters that define the dimensions of the ellipsoid are defined by three other random variables U_a , U_b, and U_c. These are continuous random variables chosen in the range of $[5, 15]$ which should represent several varying types

我们从研究中了解到，肿瘤的形状可以通过多个椭球的组合来模拟 [5]。因此，我们最精确的模型是通过在随机位置布置几个相交的不同椭球来构建的。我们从一个小的球体肿瘤开始（见球体肿瘤模型）。然后我们生成三个均匀分布的离散随机变量 U_x、U_y 和 U_z，并用 (U_x, U_y, U_z) 表示球体内随机选择的体素。这一点成为添加到肿瘤的椭球体中心。定义椭圆体维数的参数 a、b 和 c 由三个其他随机变量 U_a，U_b 和 U_c 定义。它们都是在 $[5, 15]$ 范围内选择的连续随机变量，从而能够表示多种不同类型的椭球，使得肿瘤更加真实。

of ellipsoids that make the tumor more realistic.

$$U_{x,y,z} \sim [1, 2, \cdots, 100] \qquad U_{1,2,3} \sim [5, 15] \tag{1}$$

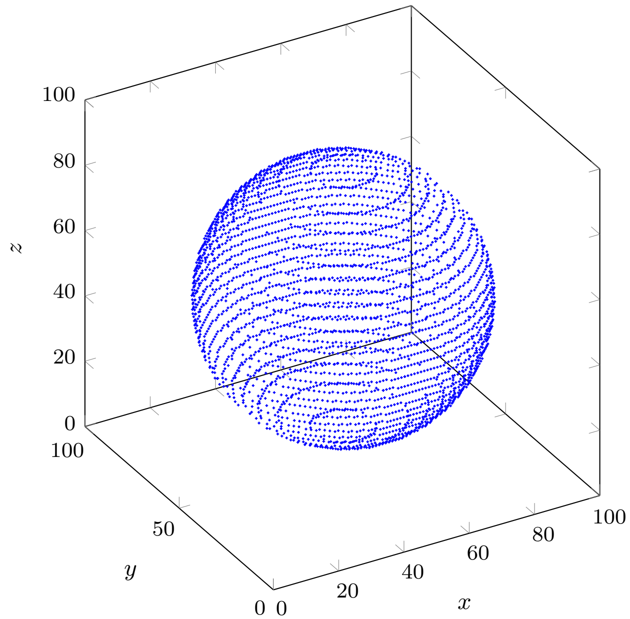

Figure 1: Spherical Tumor | 球形肿瘤

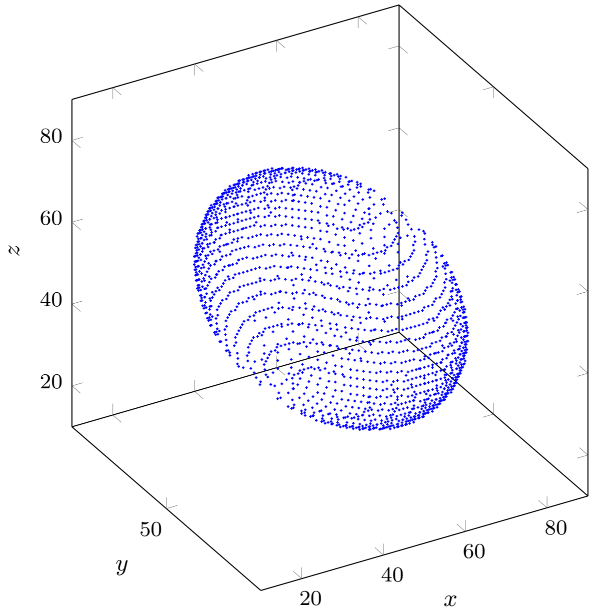

Figure 2: Ellipsoid Tumor | 椭球形肿瘤

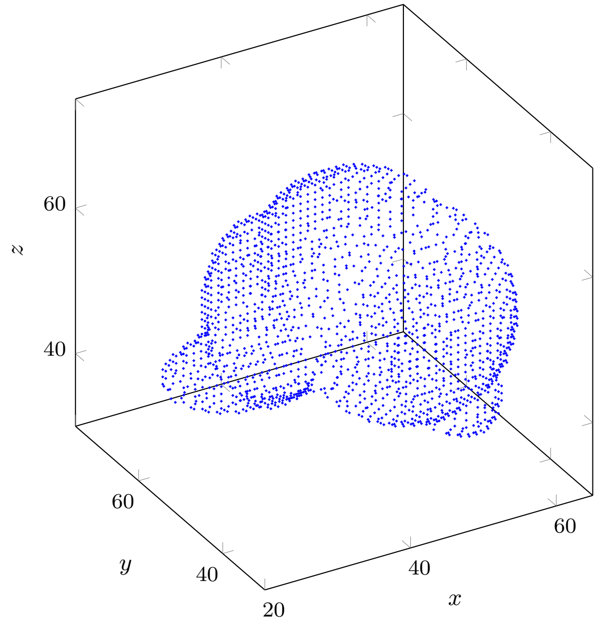

Figure 3: Mutated Tumor | 变异肿瘤

5 Sphere-Packing Algorithms | 球体填充算法

We tried various approaches to solving the problem of packing spheres (from a small set of discrete sizes) into an irregular volume. In practice, tumors are usually represented as a 3D image obtained with an MRI (magnetic resonance imaging). Therefore, we have taken the approach of discretizing the tumor and the removal spheres for processing. We have explored four different methods:

- First Deepest Method

我们尝试了多种方法来解决（选自包含少量几种离散尺寸的）球体填入不规则体积的问题。实际上，肿瘤通常用 MRI（磁共振成像）获得的 3D 图像表示。因此，我们采取了离散化肿瘤、并去除球体的处理方法。我们探索了四种不同的方法：

- 深度优先法

- Steepest Descent Method
- Improved Steepest Descent Method
- Adaptive Method

- 最速下降法
- 改进的最速下降法
- 自适应方法

5.1 Grassfire Algorithm | 烧草算法

All of our sphere-packing methods employ a useful technique called the grassfire algorithm [4]. The grassfire method progressively marks the layers of the tumor from the outside in, analogous to a fire burning away an object on layer at a time. In terms of our model, a 1 represents the existence of cancer at that location, while 0's represent healthy brain tissue.

我们所有的球体填充方法都使用了一种称为烧草算法的有用技术 [4]。烧草算法从外部逐层地标记肿瘤层，类似物体被火从外到内一层一层地燃烧。就我们的模型而言，1 表示在该位置存在肿瘤，而 0 表示健康的脑组织。

For each 1-valued voxel, all surrounding voxels are surveyed. If any of the surrounding voxels are 0-valued (outside the tumor), the current voxel is set to a depth of 2, which represents the boundary of the tumor. This process is repeated for every voxel in the 3D test volume, with layer numbers progressively increasing until all of the 1's in the array have been consumed. The grassfire method calculates an approximate measure of depth for each voxel in the tumor. This gives an easy measure of the largest sphere that can be placed at any given point without violating the tumor boundary. If the voxel is at depth 8 or 9, then a 7mm sphere should be used, and so on. The basic operation of grassfire (shown in two dimensions) can best be seen in the Figure 4. For readability, the boundary layer has been left at a value of one, and the data arrays represent a much smaller area than the plots. Figure 4 shows the effect of grassfire on a circle. Depth is indicated by shade, darker is deeper. The arrows show progression through initial grassfiring through the removal of a small circle from the center (just as spheres are removed from the tumor). Notice that when grassfire is applied after removal, the maximum depth is smaller than that of the original circle.

对于每个值为 1 的体素，所有周围的体素都被检查。如果其周围任何体素的值是 0（在肿瘤之外），则当前体素的深度被设置为 2，表示肿瘤边界。对于 3D 测试体积中的每个体素重复该过程，层数逐渐增加，直到数组中的 1 全部被替换完。烧草算法计算出了肿瘤中每个体素的深度近似值。对于任意给定点，其深度简单地给出了该点可放置不超出肿瘤边界的最大球体半径。如果体素深度为 8 或 9，则应使用 7mm 的球体，依此类推。烧草算法（显示的是二维版）的基本操作可以在图 4 中看到。为了便于阅读，边界层的值保持为 1，上方的数组表示比相应下方的图小得多的面积。图 4 显示了烧草算法对一个圆的处理过程。深度由颜色表示，颜色越深表示深度越深。箭头显示了从大圆中心移除一个小圆的烧草算法过程（就像从肿瘤中移除球体）。注意，移除小圆后再次应用烧草后，最大深度将小于原来大圆的最大深度。

Although it is simple, grassfire provides the foundation for all of our sphere packing algorithms.

虽然烧草算法简单，但其为本文接下来所有的球体填充算法提供了基础。

5.2 Sphere Placement Methodology | 球体安置方法

After grassfiring the tumor model, the deepest point in the tumor is easily found. Reasonably, the deeper the point in the tumor, the more likely it is that a

在肿瘤模型经烧草算法处理后，肿瘤中深度最大的点很容易找到。显然，肿瘤中深度越大的点，能够放置较大半径

large radius sphere can be placed without harming normal tissue. Large (particularly 9 mm) spheres being placed in the tumor increase the coverage of the solution. Conversely, the smallest sphere (2 mm radius) is the least efficient in eradicating cancerous tissue. For the average tumor size, the 2 mm sphere removes less than 1% of the volume. Considering the constraint of the problem that requests that we limit the number of shots, we wish to use the 2 mm sphere the least of the four. Therefore, we place as many large spheres as possible before placing smaller spheres.

球体而不损害正常组织的可能性越大。在肿瘤中放置较大的球体（特别是半径为 9 mm）将有助于提高治疗方案的覆盖率。反之，最小的球体（半径半径为 2 mm）在消除癌性组织方面效率最低。对于平均尺寸的肿瘤，2 mm 的球体去除不到 1% 的体积。考虑到题目要求我们限制靶点（球体）的数量，在 4 种尺寸的球体中，我们希望最少地使用 2 mm 的球体。因此，在放置较小的球体之前，我们得尽可能多地放置较大的球体。

5.3 “First Deepest” Method | 深度优先法

The first deepest method begins by applying the grassfire algorithm to the tumor data. We can then generate a list of the deepest voxels (nearly all volumes will have multiple ‘deepest’ points after the layering process). This method simply takes the first voxel off that list and places the removal sphere at that location. The radius of the sphere that is used is determined from the depth value at that voxel. For example, if a voxel was 8 layers deep (and thus 8 mm deep) then a 7 mm radius sphere can be removed from that location without harming any healthy brain tissue.

深度优先法是基于应用烧草算法处理后的肿瘤数据。然后，我们可以生成一个最深的体素列表（几乎所有肿瘤数据在分层后都会有存在多个“最深”点）。深度优先法简单地选择列表中的第一个体素，并将填充球体放置在该位置。所使用球体的半径由该体素的深度值确定。例如，如果体素是 8 层深（因此深度为 8 mm），则可从该位置移除 7mm 半径的球体而不损害任何健康的脑组织。

5.3.1 Step by Step | 算法步骤

This algorithm has a few basic steps:

1. Grassfire the tumor data (tumor represented by 1’s, healthy tissue by 0’s)
2. Grab one of the points at the deepest level
3. Calculate the equation for the sphere centered at that point with the largest acceptable radius
4. Set all voxels within the radius of the sphere to zero (effectively removing a spherical portion of tumor)

深度优先法的基本步骤如下：

1. 用烧草算法处理肿瘤数据（以 1 表示肿瘤，以 0 表示健康组织）
2. 找到深度最大的一点
3. 计算以该点为中心可接受的最大半径的球体方程
4. 将该球体半径内的所有体素设置为零（有效去除肿瘤在该球体内的部分）

5. Reset all nonzero voxels to 1's (resetting the tumor for another grassfire run)

5. 将所有非零体素的值重置为 1（重新设置肿瘤以进行另一次烧草算法处理）

6. Return to step 1

6. 回到步骤 1

0	0	0	1	1	1	0	0	0
0	0	1	1	1	1	1	0	0
0	1	1	1	1	1	1	1	0
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1	0
0	0	1	1	1	1	1	0	0
0	0	0	1	1	1	0	0	0

1's re present location of tumor

0	0	0	1	1	1	0	0	0
0	0	1	2	2	2	1	0	0
0	1	2	1	1	1	2	1	0
1	2	1	1	1	1	1	2	1
1	2	1	1	1	1	1	2	1
1	2	1	1	1	1	1	2	1
0	1	2	1	1	1	2	1	0
0	0	1	2	2	2	1	0	0
0	0	0	1	1	1	0	0	0

After one iteration

0	0	0	1	1	1	0	0	0
0	0	1	2	2	2	1	0	0
0	1	2	3	3	3	2	1	0
1	2	3	4	4	4	3	2	1
1	2	3	4	5	4	3	2	1
1	2	3	4	4	4	3	2	1
0	1	2	3	3	3	2	1	0
0	0	1	2	2	2	1	0	0
0	0	0	1	1	1	0	0	0

Completion of grassfire

0	0	0	1	1	1	0	0	0
0	0	1	2	2	2	1	0	0
0	1	2	1	1	1	2	1	0
1	2	1	0	0	0	1	2	1
1	2	1	0	0	0	1	2	1
1	2	1	0	0	0	1	2	1
0	1	2	1	1	1	2	1	0
0	0	1	2	2	2	1	0	0
0	0	0	1	1	1	0	0	0

After removal of center portion and processing by grassfire algorithm

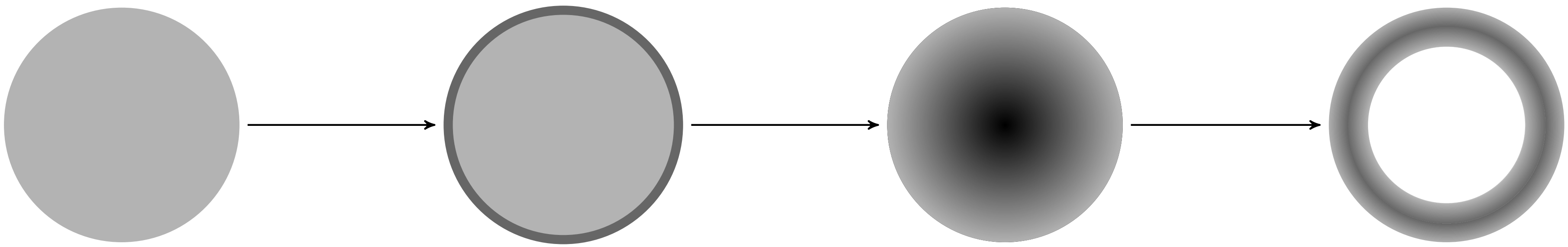

Figure 4: Grassfire Algorithm Flow Chart | 烧草算法流程图

This method is very robust, when a sphere is removed, it is simply seen as a new tumor boundary, so any of the variables such as shot size, number of shots, or tumor shape can change and the algorithm will still work.

这种方法是非常强大的，当一个球体部分从肿瘤中被删除后，剩余的肿瘤被简单地看作是一个新的肿瘤边界，因此任何变量（如球体大小和数量、肿瘤形状）的改变，都不会影响算法的正常工作。

5.3.2 Attempted Variations | 尝试改进

After seeing our first model be semi-successful, we tried to improve the method by looking down the list of the deepest voxels to find a more appropriate sphere center. We accomplished this by giving each voxel a score based on the depths of its neighboring points. Essentially, the algorithm tried to place the sphere at the greatest possible average depth. This did not improve the total coverage; in fact, this algorithm was inferior to the "first deepest" method. We realized that placing the sphere as deep as possible would reduce the depth of the next iteration, preventing more large spheres from being placed. A better strategy would be placing the sphere as shallow as possible (see Figures 5, 6 for a 2D example), in an effort to leave room for more large spheres.

在看到第一个模型略有小成之后，我们试图通过在最深的体素列表中寻找一个更合适的球体中心来改进算法。我们的改进是通过给每个体素评分来完成的，每个体素的得分是依据其相邻点的深度评定的。本质上，该算法试图将球体放置在最大可能的平均深度。这并没有提高总的覆盖率。实际上这个算法还不如“深度最优”方法。我们意识到，尽可能深的放置球体会减少下一次迭代后的深度，从而不利于放置更多的大球体。一个更好的策略是将球体尽可能地放浅（参见图 5 和 6），以便为更大的球体留出空间。

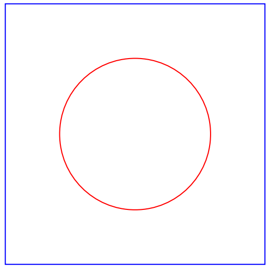

Figure 5: A single large circle in the center prevents placement of any more large circles. | 在正中间放置一个大圆不利于放置更多的大圆。

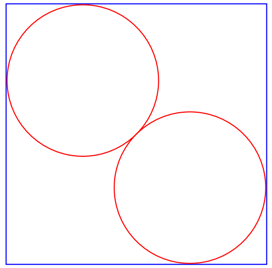

Figure 6: If the first circle is placed far from the center, a second large circle can fit. | 如果第一个圆放置在远离中心的位置，则可以再放置一个大圆。

5.4 Method of Steepest Descent | 最速下降法

The method of steepest descent tries to place the largest possible sphere (as determined by grassfire) close to the tumor boundary. The steepest descent uses a scoring function to find the best location for the biggest sphere.

Starting from the deepest voxel, we calculate the gradient of the score function and proceed along the steepest path until a local max is reached, and this point is used as a sphere isocenter. This is implemented as follows:

最速下降的方法试图将最大可能球体（由烧草决定）放置在靠近肿瘤的边界。最速下降法使用评分函数来找到最大球体的最佳位置。

从最深的体素开始，我们计算评分函数的梯度，并沿着梯度方向直到达到局部最大值，并将此点用作球体等中心点。具体实施步骤如下：

1. Calculate the score of the deepest voxel
2. Calculate the score of all surrounding voxels
3. If the original voxel is the highest scoring, it becomes an isocenter, otherwise move to the highest scoring voxel and go back to step 1.

1. 计算最深体素的分数
2. 计算所有周围体素的得分
3. 如果原始体素得分最高，则其变成等中心，否则移动到得分最高的体素并返回步骤 1

5.4.1 Scoring Function | 评分函数

This method is only as good as the scoring function. We have two factors, W_1 and W_2 that figure into the score of a given voxel.

The W_1 factor measures the depth of any nearby voxels; more specifically, it is an estimation of the depth-density of a sphere centered at that voxel. More rigorously defined, we estimate the W_1 at voxel (x_0, y_0, z_0) to be:

这种方法的好坏只取决于评分函数的好坏。我们有两个因子 W_1 和 W_2，可以用于计算给定体素的得分。

因子 W_1 估计了所有邻近体素的深度; 更具体地说，它是以该体素为中心的球体深度密度的估计。更严格地说，我们估计体素 (x_0, y_0, z_0) 的 W_1 为：

$$W_1 \approx \frac{\iiint_{S(x,y,z)} D(x,y,z)\,\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z}{\text{Total Volume of sphere}} \tag{2}$$

where $S(x_0, y_0, z_0)$ is a sphere centered at (x_0, y_0, z_0). $D(x, y, z)$ is the depth at (x, y, z), so effectively W_1 represents the average depth throughout the sphere's volume. To speed up the evaluation time of the scoring function, we estimate this volume integral by averaging the depth values for a cube surrounding the point. The sphere is inscribed within our cube of estimation, and given that the scoring function will only be a basis of relative comparison; the level of error is tolerable.

The W_2 factor is used to make sure that normal tissue is not contained in the shot. Given the sphere size that will be used for the potential shot,

其中 $S(x_0, y_0, z_0)$ 是一个以 (x_0, y_0, z_0) 为中心的球体。$D(x, y, z)$ 是 (x, y, z) 处的深度，所以 W_1 有效地表示了整个球体体积的平均深度。为了加速评分函数的评估时间，我们对以该点为中心的立方体的深度值进行平均来估计以上体积积分。虽然球体积分被我们用立方体来近似，但给出的评分函数只是相对比较的基础; 误差程度是可以接受的。

因子 W_2 用于确保正常组织不包含在靶点（球体）中。对于给定尺寸（将有可能作为靶点）的球体，

$$W_2 = \begin{cases} 1 & \text{if depth}(x_0, y_0, z_0) > \text{shot radius;} \\ 0 & \text{if depth}(x_0, y_0, z_0) \le \text{shot radius.} \end{cases} \tag{3}$$

This is another place where our decision to prohibit destroying any healthy tissue became a central part our solution. If one wished to improve total coverage of the tumor at the expense of healthy tissue, they could implement a continuous scoring function for the W_2 weight.

本文解决方案的核心是禁止损伤任何健康组织，因子 W_2 是这一核心的另一个种体现。如果有人希望以牺牲健康组织为代价来改善肿瘤的总覆盖率，那么他们可以把 W_2 修改为一个连续函数。

Finally, the total score is given by W_2/W_1. This scoring function will reward the shot for being at a closer distance to the tumor edge (or a removed sphere, since this looks like an edge to our algorithms) while still being entirely contained within the tumor.

最后，总得分由 W_2/W_1 给出。这个评分函数将奖励完全包含在肿瘤内且距离肿瘤边缘较近靶点（或去除的球体，因为对于我们的算法，这看起来像肿瘤的边缘）。

5.5 Improved Steepest Descent Method | 改进最速下降法

This algorithm is very similar to the aforementioned Steepest Descent Method, but allows spheres to be placed closer to the tumor boundary. The only changes lie in how the W_1 and W_2 weights are calculated in the score function.

该算法与前面提到的最速下降法非常相似，但允许球体更靠近肿瘤边界。唯一的差别就是如何计算评分函数中的 W_1 和 W_2。

5.6 Altered Score Function | 改进评分函数

The improved scoring function calculates the W_2 score factor in a more rigorous manner. In the previous scoring function, we used the depth (determined by the grassfire method) to determine if the shot would fit or not fit. After analyzing the results, we found that the grassfire depth is actually a conservative depth estimate–there can be more distance between the voxel and the boundary than is indicated. In order for our sphere to fit more tightly, we constructed a list of the points on the boundary which is consulted each time W_2 is calculated. Now,

改进的评分函数以更严格的方式计算 W_2 评分因子。在之前的评分函数中，我们使用深度（由烧草算法确定）来确定靶点是否合适。在分析结果之后，我们发现由烧草算法得到的深度实际上是一个保守的深度估计 - 体素和边界之间的距离可能会比所示的更大。为了使我们的球体更加紧密地填入，我们在每次计算 W_2 时在构建了一个边界点列表。则有，

$$W_2 = \begin{cases} 0 & \text{if any shot voxels} \subset \{(x,y,z)|(x,y,z) \text{ is on tumor boundary}\}; \\ 1 & \text{else.} \end{cases} \tag{4}$$

5.7 Adaptive Method | 自适应法

The Adaptive Method generates an initial sequence of shots using the Steepest Descent Method (note that coverage could be improved by using the Improved Steepest Descent Method; but the simulations would run an order of magnitude slower). The initial sequence is then changed one sphere at a time and re-packed until each shot in the sequence has been changed once. Theoretically, taking this action allows for the exchange of a large sphere for many smaller spheres, which may be more effective. It follows the idea that perhaps some spheres need to be placed poorly initially in order to allow smarter shots to be placed down the line.

自适应法使用最速下降法生成初始靶点序列（注意，通过使用改进的最速下降法可以提高覆盖率; 但是计算的速度会降低一个量级）。然后每次只更换初始序列中一个球体并重新填入，直到序列中的每个球体都被改变一次。从理论上讲，这样的操作可以将较大球体替换成更多较小的球体，这可能更有效。它遵循这样的想法：也许一些球体刚开始需要放在不太好的位置，以便让接下来更好的放置其它球体。

For instance, consider an initial sequence of length N that starts with the following shots {9 mm sphere, 4 mm sphere, 4 mm sphere, $\cdots$}. Using the same initial tumor, the Adaptive Method will run the Steepest Descent Method, but the new sequence MUST start with a 7 mm sphere. On the second iteration, the sequence MUST start with a 9mm sphere followed by a 2 mm sphere. The third iteration will start with 9mm sphere, 4mm sphere, and then a 2 mm sphere. This continues until all $N-1$ possible sequences have been generated. After looking at each of these new generated sequences, we look to find the one with the maximum coverage and keep that sphere packing sequence.

例如，考虑一个长度为 N 的初始序列，该靶点序列如下：{9 mm sphere, 4 mm sphere, 4 mm sphere, $\cdots$}。使用相同的初始肿瘤，自适应方法将运行最速下降法，但新的序列必须以 7 mm 的球体开始。在第二次迭代中，序列必须从 9 mm 球体开始，然后是 2 mm 球体。第三次迭代将以 9 mm 球体，4 mm 球体和 2 mm 球体开始。迭代一直持续到所有 $N-1$ 可能的序列都已经产生。在查看这些新生成的序列之后，我们寻找具有最大覆盖率的序列并保存该球体填充序列。

6 Results | 结果

In order to test our solutions, we simulated our algorithms numerous times to find out the statistical characteristics of our methods.

为了测试我们的解决方案，我们多次运行了我们的算法，以找出我们算法的统计特性。

6.1 Quantitative Results | 定量结果

Table 1: Timing Performance of Sphere-Packing Algorithms | 球体填充算法的运行时间性能

Method	Avg Time Elapsed	Relative Speed
First Deepest	83.14	1.0
Steepest Descent	104.21	1.25
Improved Steepest Descent	229.08	2.8
Adaptive	1025.40	12.3

Table 2: Side-by-side Comparison of Sphere-Packing Algorithm Performance | 球体填充算法的性能比较

Method	Trials	Ave. % Covered	Standard Deviation	Minimum	Maximum
First Deepest	100	33.6127	5.1973	20.0951	45.2376
Steepest Descent	100	38.1984	2.9443	32.2990	44.8696
Improved Steepest Descent	100	37.0520	6.2445	27.5252	55.9551
Adaptive	20	39.8606	2.5211	35.1200	43.7666

Each algorithm has been run on the same suite of 100 testcases, with the exception of the adaptive algorithm. Because of the longer run time of the adaptive algorithm, a subset of 20 cases was used. Because of the relatively large number of testcases performed, we will be able to infer general characteristics of each algorithm in terms of run time and coverage percentage. Table 1 contains a summary of the timing results of each algorithm, and in Table 2 we show the quantitative results for all four algorithms in one place.

除了自适应算法，每种算法都使用了同一套 100 个用例进行测试。由于自适应算法的运行时间较长，所以使用了 20 个用例。由于执行的测试用例数量相对较多，我们可以根据运行时间和覆盖百分比来推断每个算法的一般特征。表 1 包含了每种算法的耗时结果，表 2 显示了所有四种算法的定量结果。

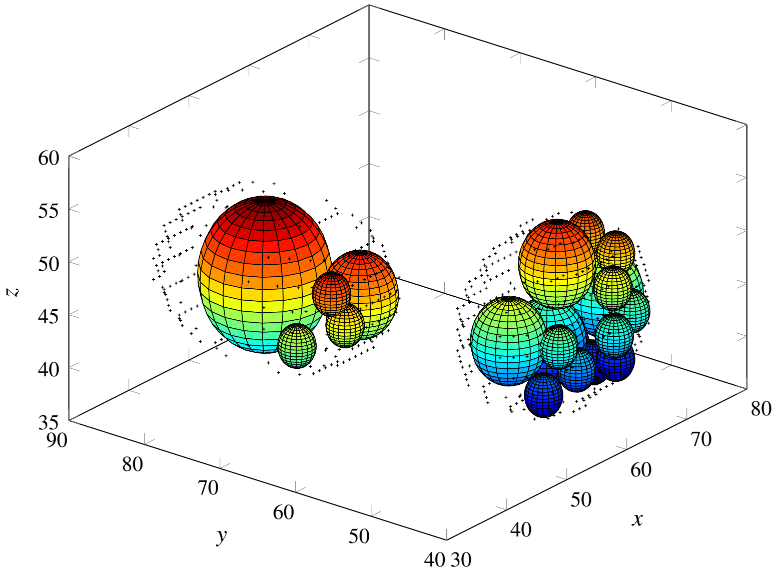

Figure 7: Sphere Packing for Two Disjoint Tumors | 两个分离肿瘤的球体填充

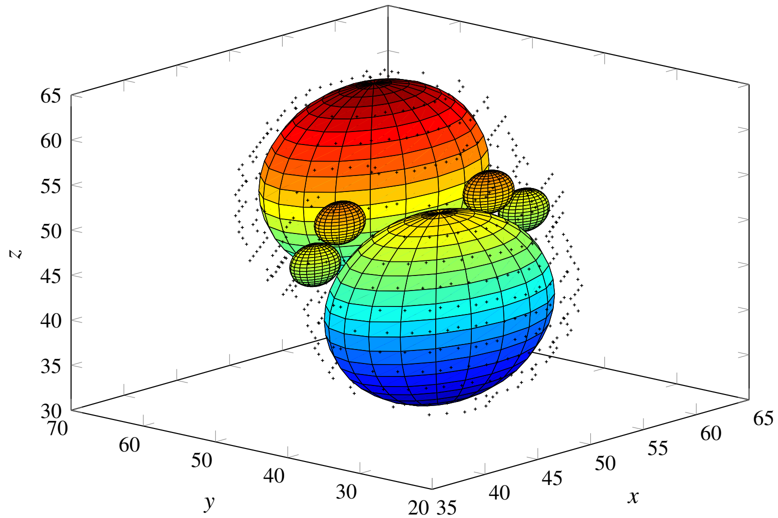

Figure 8: Best Coverage Sphere-Packing | 最佳覆盖率的球体填充

One of the features that makes our modeling approach distinct is that our algorithms work on algorithm of arbitrary shape even disjoint tumors. The way we modeled the data and the sphere-packing approach we took means that we can handle any discretized 3D shape–including disjoint volumes. The result of one such pack is shown in Figure 7.

The maximum coverage we could achieve was 55.9% without harming any

我们模型明显不同于其它建模方法的一个特点是：我们的算法可以处理任意形状甚至不相连的肿瘤。我们模拟数据的方式以及我们采用的球体填充算法意味着我们可以处理任何离散化的 3D 形状—甚至是不连续的体积。一个不连续肿瘤的球体填充结果如图 7 所示。

在不伤害任何健康的脑组织的前提下，我们使用四种离

healthy brain tissue, and using the discrete set of sphere sizes. The sphere pack we used is below in Figure 8, and a set of slices of the 3D object is shown in Figure 9. More qualitative detail on each algorithm is in the following sections.

散尺寸球体可达到的最大覆盖率是 55.9%。我们使用的填充球体如图 8 所示，三维对象的一组切片如图 9 所示。每种算法的更多定性细节见接下来的部分。

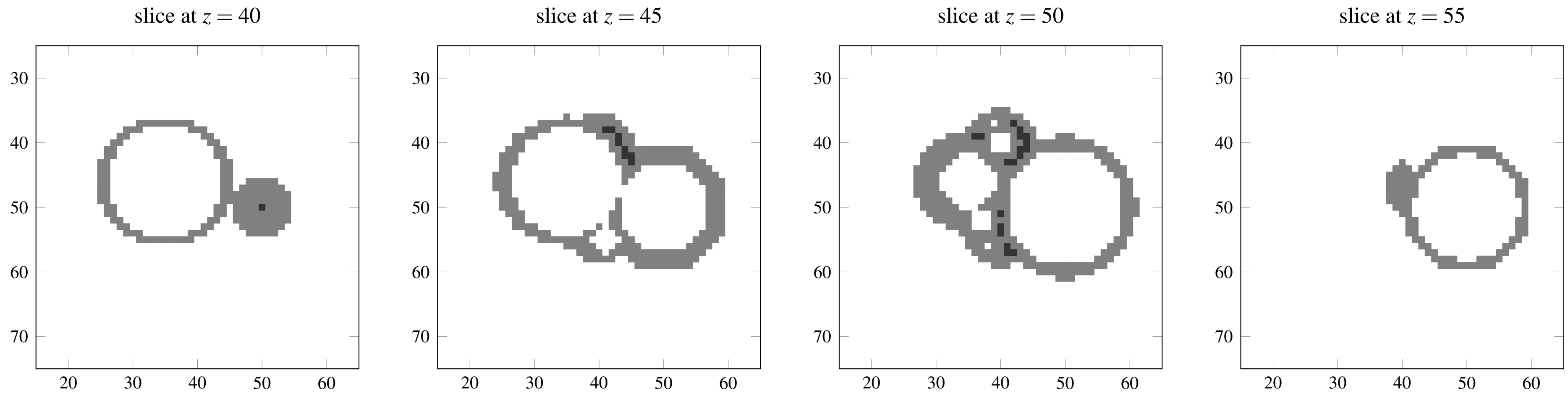

Figure 9: Slices of Best Coverage. Grey represents remaining tumor tissue. | 最佳覆盖范围。灰色代表剩余的肿瘤组织。

6.2 Qualitative Results | 定性结果

6.2.1 “First Deepest” Method | 深度优先法

As expected, this algorithm’s performance was quantitatively the weakest. The maximum coverage of 45.3% is actually slightly better than the other methods. However, the minimum value of 20.1%, as well as the average of 33.6% are quite low compared with other methods. It also exhibits large, random variations in coverage, meaning the algorithm is equally likely to fill in a low percentage as it is to fill in a relatively high percentage of a tumor. See Figure 10 for a plot of testcase vs. coverage.

Overall:

- The algorithm was inconsistent and was not as effective as the rest of the algorithms.

正如所料，深度优先法的性能从定量上来看是最差的。45.3% 的最大覆盖率实际上略好于其他方法。然而，与其他方法相比，覆盖率最小值仅为 20.1%，覆盖率平均值为 33.6%。而且深度优先法的覆盖率也表现出较大的、较随机的变化，这意味着该算法既可能出现相对较高的肿瘤覆盖百分比，也可能出现较低的覆盖百分比。测试用例与覆盖率的关系见图 10。

总结来说：

- 该算法一致性不高，效果不如其他算法。
- 该方案给出的平均覆盖率是四种解决方案中较低的。

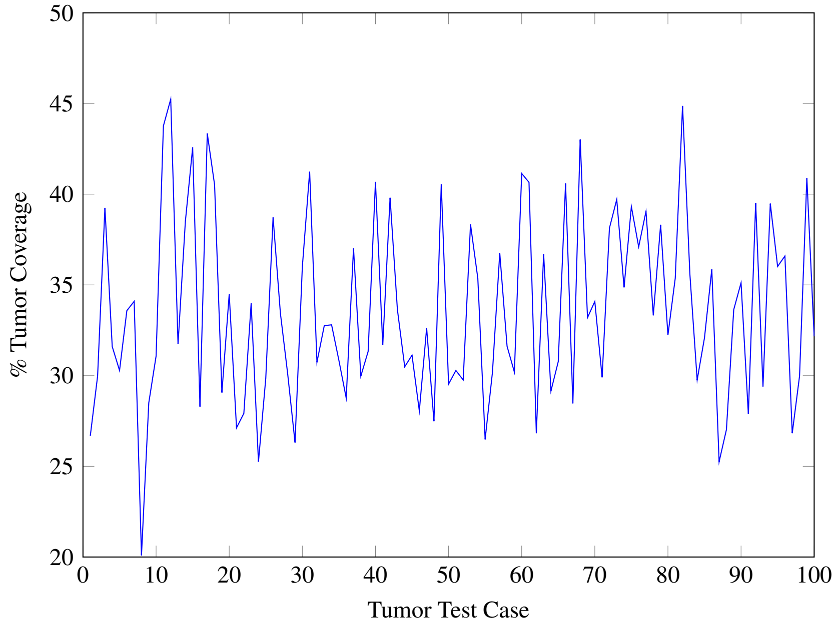

Figure 10: First Deepest Method with 100 Test Cases | 100 个测试用例的深度优先法结果

- The solution yielded a lower average percentage of the four solutions.
- Run time is faster than the other methods.

- 该算法的运行时间比其他算法快。

6.2.2 Method of Steepest Descent | 最速下降法

This algorithm offered a distinct improvement over the “first deepest” method. Although the maximum coverage was slightly lower, the minimum and average coverages were significantly better. The algorithm was also much more consistent. This makes sense because this method adapts to variations in tumor size and shape, rather than using the first available isocenter for each sphere. See Figure 11 for full test results.

Overall:

- The algorithm covers more tumor volume than the “first deepest” method

这个算法为“深度优先”法提供了一个明显的改进。尽管最大覆盖率略低，但最小覆盖率和平均覆盖率明显比深度优先法更好。算法也具有更高的一致性。这是合理的，因为这种方法能够适应肿瘤大小和形状的变化，而不是直接使用每个球体第一个可用的等中心。完整的测试结果见图 11。

总的来说：

- 该算法比“深度优先”法涵盖更多的肿瘤体积

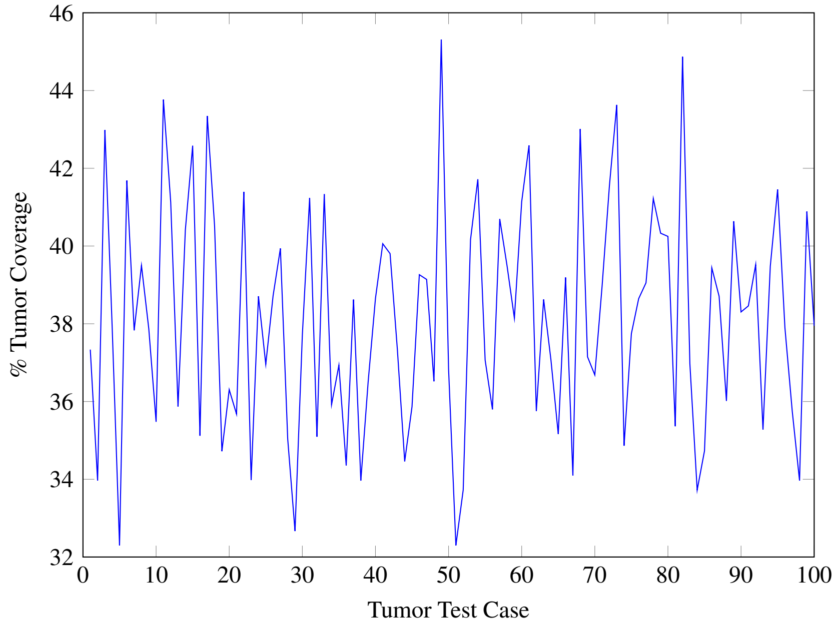

Figure 11: Steepest Descent Method with 100 Test Cases | 100 个测试用例的最速下降法结果

- The algorithm produces relatively consistent coverage
- The algorithm is fast compared to the improved steepest descent and adaptive methods

- 该算法产生相对一致的覆盖
- 与改进的最速下降和自适应法相比，该算法是快速的

6.2.3 Improved Steepest Descent Method | 改进的最速下降法

This algorithm is similar to the method of steepest descent, except for the ability to pack spheres closer to edges and other spheres. This method did poor on the average, doing both worse than the Steepest Descent Method and the Adaptive Method. But, it did yield the best coverage out of all methods–reaching over 50% on four different test cases. Also, this method had the highest deviation of all our methods.

Overall:

除了能够将球体填充得更靠近边缘和其他球体，这个算法类似于最速下降法。平均而言，这种方法效果不佳，它比最速下降法和自适应法差都差。但是，它确实产生了所有方法中最好的覆盖率—在四种不同的测试案例中达到了 50% 以上。而且，这种方法在我们所有的方法中具有最高的偏差。

总的来说：

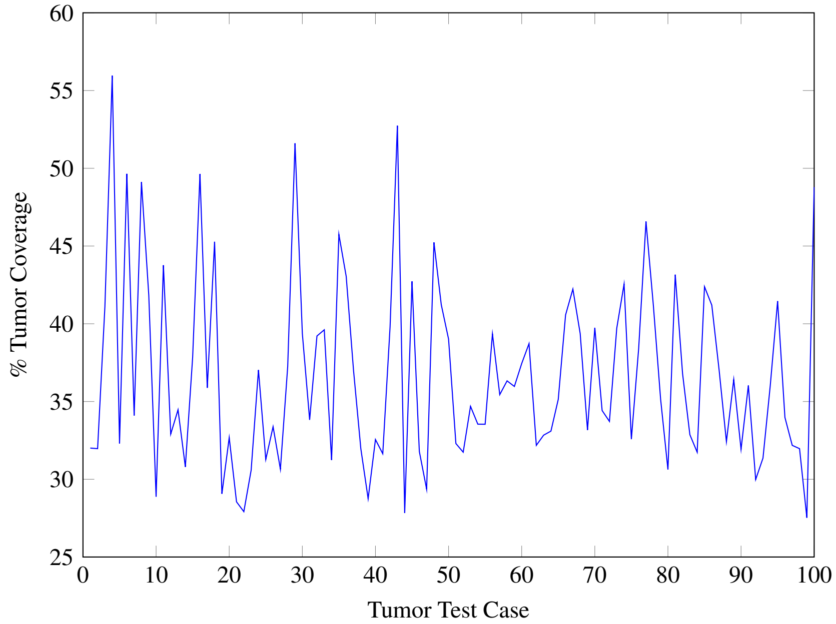

Figure 12: Improved Steepest Descent Method with 100 Test Cases | 100 个测试用例改进的最速下降法

- The algorithm yielded the best coverage out of any algorithm
- On the average, this algorithm ranks second to the worst in coverage
- The method proved to have the highest standard deviation

- 在所有算法中，改进的最速下降法给出了最佳的覆盖率
- 平均而言，该算法在覆盖率上排在第二位
- 该方法被证明具有最高的标准偏差

6.2.4 Adaptive Method | 自适应法

The adaptive algorithm is based off of the Steepest Descent Method, that performed better than the ordinary Steepest Descent Method. The average coverage of approximately 40% was the best out of all four algorithms; and it also had the lowest standard deviation near 2.5.

Overall:

- The method had the highest average coverage percentage

自适应算法基于最速下降法，比普通最速下降法表现得更好。在四种算法中，该算法具有最佳平均覆盖率约为 40 %; 而且该算法也具有最低的标准差约为 2.5 。

总的来说：

- 该方法的平均覆盖率最高

- Algorithm was the most consistent method out of the four
- Adaptive Method was the slowest and most complicated algorithm

- 该算法是四种算法中一致性最高的算法
- 自适应法是最慢、最复杂的算法

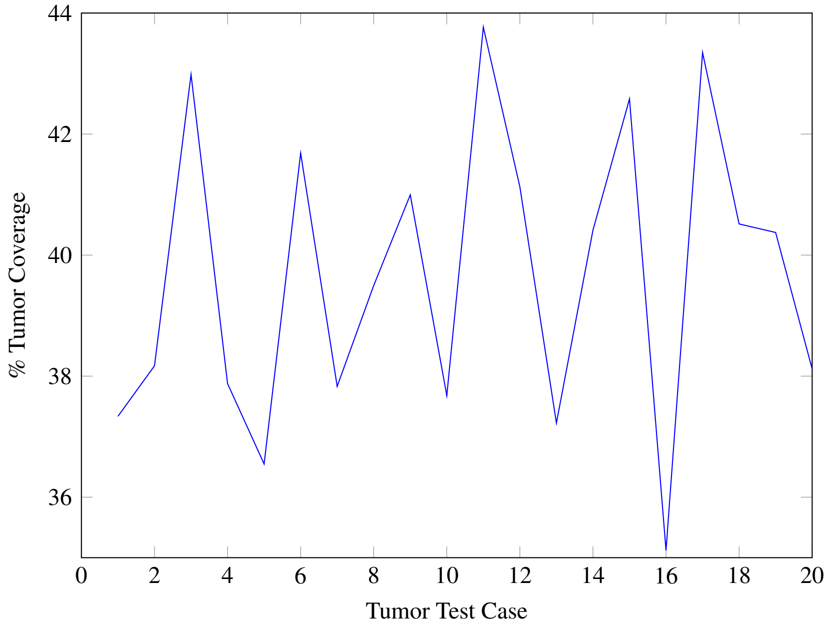

Figure 13: Adaptive Method with 20 Test Cases | 20 个测试用例的自适应方法

7 Conclusion | 结论

In retrospect, we found that all of our algorithms had strengths and weaknesses. The "first deepest" method was quick in run time, while the method of steepest descent was consistent in the coverage of the tumor. The improved steepest descent method yielded the highest percentage of the tumor covered, while the adaptive algorithm maintained the highest average percent covered with the smallest deviation.

回想起来，我们发现本文所有的算法都各有优缺点。“深度优先”法运行速度快，而最速下降法的肿瘤覆盖率的一致性较高。改进的最速下降法给出了最高的肿瘤覆盖率，自适应算法则具有最高的平均覆盖百分比，且具有最小的偏差。

7.1 Weaknesses | 缺点

Because of the fact that we were so concerned with the prohibiting of hot spots and radiating healthy brain tissue, our algorithms did not cover as much of the tumor as may be desired.

由于我们绝对禁止热点和放射健康的脑组织，因此我们的算法并没有尽可能多地覆盖肿瘤。

7.2 Strengths | 优点

Our algorithms can process any possible tumor. Our algorithms are "patient-friendly" – they don't destroy anything outside of the tumor, nor do they produce spheres that intersect each other. They are also simple and robust. Someone with more specific planning goals could use our algorithms as a strong foundation.

我们的算法能处理任何可能的肿瘤。我们的算法是“对病人友好的”—它们不会破坏肿瘤以外的任何东西，也不会产生相互交叉的球体。他们也简单而强大。有更具体的规划目标的人可以将我们的算法作为一个坚实的基础。

7.3 Future Work | 未来的工作

It would be nice to merge our Adaptive algorithm with that of the Steepest Descent, and try to build up our percentage covered with both techniques. Also, efficient geometrically based algorithms could be augmented with ours to produce excellent results in the presence of specific cases, while maintaining the ability to handle a case of arbitrary complexity.

将我们的自适应法与最速下降法合并，并尝试构建包含两种技术的覆盖百分比是非常好的想法。此外，有效的基于几何的算法可以增强我们的算法，在特定情况下产生出色的结果，同时保持处理任意复杂情况的能力。

References

[1] Michael C Ferris, Robert R Meyer, and Warren D' Souza. Radiation treatment planning: Mixed integer programming formulations and approaches. In *Optimization Technical Report 02-08*. University of Wisconsin, 2002.

[2] John YK Lee, Ajay Niranjan, James McInerney, Douglas Kondziolka, John C Flickinger, and L Dade Lunsford. Stereotactic radiosurgery providing long-term tumor control of cavernous sinus meningiomas. *Journal of neurosurgery*, 97(1):65–72, 2002.

[3] Andrew H Kaye and Edward R Laws Jr. *Brain Tumors*. Churchill Livingstone, New York, 1995.

[4] Thomas H Wagner, Taeil Yi, Sanford L Meeks, Francis J Bova, Beverly L Brechner, Yunmei Chen, John M Buatti, William A Friedman, Kelly D Foote, and Lionel G Bouchet. A geometrically based method for automated radiosurgery planning. *International Journal of Radiation Oncology* Biology* Physics*, 48(5):1599–1611, 2000.

[5] AL Asachenkov. *Disease dynamics*. Springer Science & Business Media, 1994.

[6] Tomaso Aste and Denis Weaire. *The pursuit of perfect packing*. Institute of Physics Publishing, 2000.

[7] John Horton Conway and Neil James Alexander Sloane. *Sphere packings, lattices and groups*. Springer Science & Business Media, 1998.

[8] Anuraag R Kansal, S Torquato, GR Harsh, EA Chiocca, and TS Deisboeck. Simulated brain tumor growth dynamics using a three-dimensional cellular automaton. *Journal of theoretical biology*, 203(4):367–382, 2000.

[9] Manual MatLab. The language of technical computing. *The MathWorks, Inc. http://www. mathworks. com*, 2012.

[10] Taeil Yi. A tree in a brain tumor. *Gainesville, FL: University of Florida. The Gamma Knife Problem*, 363, 2000.