31.1 介绍

在本章中,我们讨论求解渲染方程的理论,重点介绍各种方法的数学以及这些方法涉及的近似类型,并将实现细节推迟到下一章.幸运的是,许多数学可以通过类比来理解,而问题要简单得多.渲染时,我们试图计算的值,辐射场或涉及许多值的组合(通常是积分)的表达式.因此,未知数是整个函数.这与诸如

一类的代数问题,其中未知数是.尽管如此,这些简单的方程为在寻找的更复杂任务中进行的近似提供了有用的模型;我们先讨论这些,然后再将这些想法应用于渲染.

31.2 方程的近似解

即使有中等程度的复杂度,也无法完全解决任何场景的渲染方程.相反,我们被迫近似解决方案,图形中通常使用四种常见的近似形式:

1. 近似方程
2. 限制域
3. **使用统计估计量**
4. 二等分/牛顿的方法

由于最后一个在渲染中使用不多,因此我们简短处理。然而，统计方法现在占据着渲染的主导地位,它将占据本章的其余大部分内容.

我们将在一个更简单的问题中讨论这些问题:找到下列方程的一个正实数:

该方程的数值解为,但我们假装不知道,我们只能进行手动计算,例如加,减,乘,除,求实数的整数幂.

31.3 方法1:近似方程

与其求解(这不涉及提取第2.1个根),我们可以求解“附近”方程,例如

简化为4,得到答案.由于乘法和乘幂都是连续的,因此对此“摄动”方程的解与原始解非常接近也就不足为奇了(见图31.1).解决扰动方程很容易.

你可能会顾虑上一段中“附近”一词没有定义.作为另一个示例,请考虑解决

解为.但是,如果我们稍微改变方程式,使右侧为0而不是0.1,则解为:方程中的小扰动导致解中的大扰动.确定方程对摄动的敏感性(对于更复杂的方程,例如渲染方程)通常非常困难;在实践中,这是通过说“看起来很明显的,如果月亮是椭圆形而不是圆形的,穿过我封闭的卧室窗帘的月光看起来并没有太大的区别.”换句话说,这是通过使用领域专业知识来完成的.确定哪种近似可能只对结果产生较小的扰动.

渲染中的一个示例是通过Lambert反射模型近似任意表面的反射，或者通过“从该点可以看见光源吗？”始终回答“是”来近似函数.第一个导致解决方案,使所有物体看上去都没有光泽,第二个导致解决方案没有阴影.每种近似通常比全黑图像更好，而较差的近似通常比根本没有解更好.

31.4 方法2:限制域

例如求解公式,我们可以问,“有正整数满足(或几乎满足)它吗？”(见图31.2).这样的域限制可以极大地简化事情.在此等式的情况下,我们看到左侧是x的递增函数,并且当时,其值已经是50.因此,任何整数解都必须位于0和1之间.我们只需要尝试这两种可能的解决方案,看看哪个可行(或者,如果都不可行,则是“最佳”).我们很快发现给出,这太小,而得到50,这太大.

然后,我们有两种选择:我们可以报告受限域中的“最佳”解决方案(),或者我们可以说:“理想的解决方案位于0到1之间,0比1更接近;线性插值给出作为最佳猜测答案.”(见图31.3).

我们对线性插值的使用错误地假设左侧的值在和之间作为的函数几乎线性变化,这就是为什么估计的答案不太接近真正的一个原因.更一般而言,如果某个变量的域为,并且我们限制到子集,则从的近似解估计的解需要足够大,使得的任何点 距离点足够近,因此可以从的附近点的值很好地推断出.

讨论光能传递时，我们将看到一个渲染中的域限制示例.请注意,方法1和2都违反了“近似解决方案”原理:它们近似于问题而非解决方案.

31.5 方法3:使用统计估计量

第三种方法是从统计角度“估计”解决方案,即找到一种方法来生成值的序列使得每一个是一个可能的解,并且使得的平均值随着越大越来越接近真实解.

在这种情况下,我们尝试解决公式.

其解为

这可以在计算机上轻松评估,但是我们假设缺乏实数整数幂意外的计算能力.(当我们查看渲染方程式时,相应的陈述将是:“假设我们没有能力计算除整数倍的光线反射之外的任何东西.”这是非常合理的:很难想象这可能意味着什么来计算2.1盎司的光线!)我们仍然可以使用二项式定理找到一个解,即

其中

其中是任意实数,是非负整数.我们将这种情况应用于且的情况下,因此,这样,对公式31.8进行求值将得出公式31.7中的解的值.

为此,需要对无穷级数求和.认识到可以通过查看序列的各个元素来估计无限序列的总和.

31.5.1 通过采样和估算对级数求和

现在,我们从概率论的一些简单应用开始为所有蒙特卡洛方法进行渲染奠定基础.

31.5.1.1 有限级数

假设我们有一个有限级数

然后我们要估算总和.我们可以执行以下操作:选择一个介于1和20之间的随机整数(选择每个可能数字的概率为1/20),并令.那么是一个随机变量.其期望值是其值的加权平均值,权重是概率,即,

我们有一个随机变量,其期望值是我们寻求的总和!通过实际抽取此随机变量的样本并将其平均,我们可以近似求和.

通常,的方差与序列中项的变化量有关:例如,如果所有项都相同,则不会有方差.这也与我们选择项的方式有关,这恰好是均匀的,但是稍后我们将在其他示例中使用非均匀的示例.当我们将这些想法应用于渲染时,我们最终将在光线可以传播的各种路径中进行采样.计算的值将是沿路径的光传输.由于某些路径携带大量光(例如,从光源到您的眼睛的直接路径),而某些路径携带的光很少,因此存在很大的差异.为了使估算精确,将需要大量样本,或采取其他减少方差的方法.对于基本的光线跟踪器,这意味着您可能需要每个像素跟踪许多光线,才能很好地估计到达单个图像像素的辐射率.

31.5.1.2 无穷序列

倾向于将无穷级数以一种明显的方式一般化:选择一个非负整数,然后让;令的所有选择都相等,则的期望值应.但是,这有两个问题.首先,缺失因子为20.在有限示例中,我们将每个乘以20,因为选择它的概率为1/20.这意味着在无限情况下,我们需要将每个乘以无穷大,因为选择它的概率是无穷小的.这根本没有任何意义.其次,以随机方式均匀选择一个正整数的想法听起来不错,但从数学上讲是不可能的.我们需要一种受公式31.11启发的稍微不同的方法，其中该系列的每个项乘以该项的选择概率（1/20）乘以该概率的倒数（20）.我们需要做的就是放弃均匀分布的想法.

无穷序列之和

我们可以选择概率为的非负整数,因此选择的概率为1/2,选择的概率为.(之所以选择这种特殊的概率,是因为它很容易使用,而且很明显,这些概率之和为1,但是其它任何合计为1的正数集合都可以很好地工作.）

按照上述讨论,我们令

和以前一样,的期望值为

和以前一样,估算中的方差与序列项有关.如果恰好是,则方差为零,估计量很大.如果,则方差要大得多,我们需要对大量样本取平均,以得出结果的低方差估算值.

就像我们说的那样,我们选择时所做的特定选择很简单,以概率选择,但是我们可以对正整数使用其他概率分布;根据我们选择的分布,估算器可能具有较低或较高的方差.

在将这种方法应用于渲染时,的选择将成为“光需要反射多少次”的选择.如果我们有一个场景,其中每个表面的反照率约为50％,那么我们预计仅 沿着长度k +1的路径传播的光的数量是沿着长度k +1的路径传播的光的一半。在这种情况下，将每条连续路径长度的概率分配一半是有意义的。 通常，选择正确的采样分布是使此类Monte Carlo方法行之有效的核心.

31.5.1.3 使用随机方法解

将这些方法应用于我们的特定方程,我们知道

通常,我们可以使用二项式定理变换方程的右侧,根据公式以及,我们得到

现在,为了估计一个解,我们选择一个概率为的正整数,并评估第项.在撰写本章时,我们掷硬币并计数了直到正面的掷骰次数,生成了序列3,3,1;因此,我们对的三个估计分别是该系列的第三项,第三项和第一项,分别乘以8,8和2:

回想一下,该问题的正确解决方案是0.5265.我们三个样本的平均值为x = .3024，这显然不是一个很好的解决方案估计.当我们使用10,000个项时，估计值为0.5217，这非常接近（见图31.4）.

您可能担心我们假设可以为编写幂级数,但是当到达渲染方程式时,这样的重写可能并不容易.幸运的是,对于渲染方程式,将其重写为无穷级数实际上非常容易,尽管估计结果级数的总和仍涉及相同的随机方法.

31.6 方法4:二分法

我们求解方程的最后一种方法是找到一个的值,使得小于13(例如),再找到另一个值使得大于13(例如).由于是关于的连续函数,它的值13必须介于x = 0和x = 1之间.我们可以求的值,并且发现它小于13.现在我们知道解在1/2和1之间.在我们知道的包含答案的区间的中点重复评估,然后根据结果舍弃一半的区间,迅速收敛于包含解决方案.

尽管有这些方法的吸引力，但对于函数方程式(即答案是一个函数而不是一个数字),就像渲染方程式一样，并没有简单的模拟方法.没有简单的方法可以将一个数字介于两个其他数字之间的概念推广到更通用的功能类别.

31.8 回顾渲染方程

回想一下，辐射场是场景中所有表面点的集合上的一个函数,它取一个点和方向,并返回一个实数,该实数表示离开的沿着方向离开的射线的辐射.我们的辐射场模型是一个函数.

我们在第29.4节中讨论了一个微妙之处：有些“表面点”是两个表面的一部分.例如，如果我们有一个实心的玻璃球体(见图31.5),那么最好将球体北极的点视为两个点:一个在“外部”,另一个在“内部”.在外点向北传播的光线被反射或透射，而在内点向北传播的光线到达那里，并且即将通过玻璃-空气界面传播或反射.正如我们建议的那样,我们可以增强光场的概念,使其包含三个参数：一个点，一个方向以及一个定义该点“外部”的法线向量，但是在本章的其余部分中，我们将而是只讨论反射（除了一些仔细指出的点之外）,因为（a）一处两点的想法使表示法复杂化，这种表示法已经足够复杂了,并且（b）程序中的实际更改我们将在第32章中看到的有关传输的说明相对较小且简单明了。关于保留北极的两个单独副本的问题，实际上，正如我们将在第32章中讨论的那样，我们将仅保留一个几何副本，并且不会明确表示光场。另一方面，到达的光线的含义及其处理方式将取决于其方向与单位法线n的点积，从而在我们的程序中产生多个if-else子句.

不过，我们将继续为散射函数编写,但是你需要记住,在透射的情况下,某些项可能需要在其上带有绝对值符号.

渲染方程表示场景中的辐射场的特征是,某个表面点在某个方向上的辐射是(a)该点发出的辐射与(b)该点在该方向上散射的所有入射光之和.该方程为

回忆一下这些术语的含义:

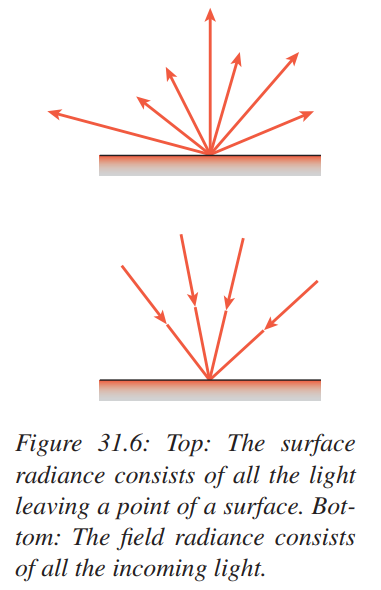
1. 是发射的辐射场,除非是照明器上的某个点,否则,并且是该照明器从发射光的方向.
2. 是由于产生的散射辐射场;是当整个场景的辐射场为时从点沿方向散射的光.

具体来说,定义为

对于我们当前的讨论,该表达式的关键特征是出现在积分中.

请注意,如果我们针对未知的求解方程31.27,则还没有计划!我们有一个函数,可以在很多地方对其进行评估以构建图片.特别是,我们可能会针对每个像素中心和从针孔相机的针孔到(假设是物理相机,其中胶卷平面在针孔后面)的相应矢量评估.

函数是一个高阶函数:它接受函数并产生新的函数.你已经在演算类中看到了其他这样的高阶函数,例如派生函数,也许还遇到过ML, Lisp和Scheme这样的编程语言,在这些语言中,这种高阶函数很常见.

 让我们从计算机科学的角度考虑这个积分.我们有一个要解决的明确问题(“找到”),我们可以检查问题的难度.首先，即使是相当琐碎的场景，也可以证明没有简单的封闭式解决方案.其次,观察到的域不是离散的,就像我们在计算机科学中看到的大多数事物一样,而是一个相当大的连续体-的参数中存在三个空间坐标和两个方向坐标,因此它是五个实变量函数.(注意:在图形中,通常将其称为“五维函数”,但更准确地说是一个域为五维的函数.)在计算机科学术语中,我们将经典问题称为“一个旅行推销员问题或3-SAT“困难”问题,因为解决该问题的唯一已知方法是从大O角度讲,比枚举所有可能的解决方案更简单.相比之下,由于是连续域,因此渲染方程更加困难,因为即使枚举所有可能的解决方案也不可行.你的下一个想法可能是开发一种不确定的方法来近似解决方案.这是一个很好的直觉,这就是大多数渲染算法的作用.但是,与研究过的许多不确定性算法不同,尽管我们可以表征这些随机图形算法的运行时间,但它本身是没有意义的,因为在一般情况下,近似中的误差是无限的:因为域是连续的,而且我们只能使用有限的多个样本,因此始终可以构建一个场景,其中所有光线都由我们的样本缺少的一些稀疏路径承载.

生成近似解的一种策略是以某种方式离散域,以便我们可以限制误差.这也是一个好主意,因为这样我们便可以枚举解决方案空间中相当大的一部分.我无法查看曲面上每个点的光传输,但可以查看曲面上三角形网眼近似的每个顶点的光传输.图形不是唯一的.当您将计算机科学从纯粹的理论中提取出来并开始将其应用于物理学时，您会发现问题通常不仅具有指数级的复杂性，而且即使在您遇到一般情况下，您也常常需要找到能很好地起作用的良好近似值 不能在所有情况下都限制错误.

回顾一下第29章,将场景中的光分为两类(见图31.6):在表面的一点上,光可能从远处的各个点到达,这种情况称为**场辐射**.这种光撞击表面并被散射.产生的输出辐射称为**表面辐射**.

进一步细分辐射也很有帮助:在点处的表面辐射可以分为那里的发射辐射(仅在照明器处不为零)和反射辐射.这些对应于渲染方程式右侧的两个项.双重地,可以将处的场辐射分为处的直接照明(即,照明器发出的辐射穿过空的空间传播到的辐射)和处的间接照明(即,沿着射线从点到点的辐射,但是点上没有自发光).在下一章中,我们将返回这些术语.

以公式31.27的形式编写渲染公式可以清楚地看出,散射算子将一个辐射场()转换为另一个辐射场().不仅这样做,而且线性地做到这一点:如果我们计算,我们得到,并且对于任何实数,,如公式31.28所示.这种线性并不是由于绘制方程时的一些技巧而引起的.它是一种物理可观察的属性,在物理学中通常被称为叠加原理,对于希望解决该渲染方程的人来说,它是非常幸运的.稍后,在第35章中,我们将看到这种叠加原理适用于基于物理的动画中出现的力,速度和其他事物,并且它将再次大大简化我们的工作.

我们可以用以下形式重写渲染方程

或者

其中表示密度算子:它将转换为,我们使用表示算符在辐射场上的应用.

本章的其余大部分内容介绍了解决此方程的方法.记住,当我们检查这种方法时,每个光源发出的光作为输入,每个表面点的双向反射率分布函数（BRDF）也被作为输入,因此可以计算出算子和未知数是辐射场.

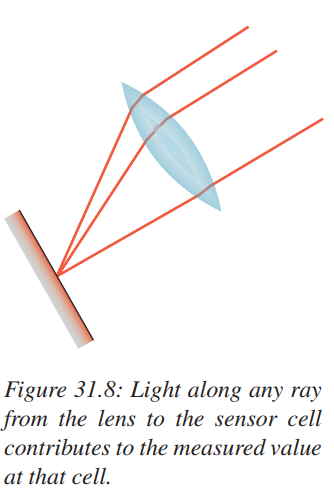
这种表达方式与线性代数中特征值问题的描述方式相似,并非巧合.在解决渲染方程式时,我们将使用许多与研究特征值相同的技术.

31.9 我们要计算的是什么？

渲染方面的许多工作分为以下几类:

1. 开发数据结构以加快射线投射操作,我们将在第36章中进行讨论
2. 选择函数的表示形式,使其足以捕获各种各样的表面所表现出的各种反射率,但又足够简单以允许进行巧妙的渲染优化,这已经在第27章中介绍过了.
3. 确定近似渲染方程解的方法

这是本章中与我们有关的最后一个主题.

 渲染方程式特征化了描述场景中辐射度的函数.我们真的需要了解有关的一切吗?大概散射到外太空(或朝某个完全吸收表面的方向)的辐射对我们来说无关紧要——它可能不会影响我们正在制作的图像.实际上,如果我们要从针孔在点的相机拍摄照片,那么我们真正关心的唯一值就是.为了计算这些值,我们可能需要计算其他值,以便更好地估计我们关注的值.

但是,假设我们要模拟一个实际的相机,它带有镜头和传感器阵列,如许多数码相机中的CCD阵列.为了计算像素处的传感器响应,我们需要考虑将光线传递到P的所有光线-光线从透镜的任何点到与对应的传感器单元的任何点(请参见图31.8).

正如我们在第29章中所述,沿不同光线入射的光可能具有不同的效果:与成角度入射的光线相比,与胶片平面正交的光线可能会产生更大的响应,并且靠近细胞中心的光线可能比到达边缘附近光线更重要—一切都取决于传感器的结构.测量方程式29.15表示

其中是传感器响应函数,它告诉我们像素对沿着方向通过的光线的辐射的响应.

一个完全合理的理想化条件是像素面积是一个很小的正方形,对于穿过该正方形的射线,否则为0.即使有这种理想化,我们也希望像素值是计算是整个像素区域和整个镜头方向的积分.即使我们假设透镜非常小,以至于可以通过单条光线（针孔近似值）准确估算出后者的积分,仍然有一个面积积分可以估算.

估算此积分的一种非常糟糕的方法是在像素区域的中心(即最简单的光线跟踪模型,我们在其中穿过像素中心发射光线)中获取一个样本.使这种方法特别糟糕的是混叠伪影:如果我们正在制作栅栏图片,并且栅栏的间距与像素的间距略有不同,结果将是大块的恒定颜色,眼睛检测到每个像素的近似近似值（见图31.9）.

如果我们改为在每个像素中取一个随机点，则可以大大减少这种混叠现象（见图31.10）。 相反，我们在图像中看到了椒盐噪声.

因为我们的视觉系统通常不会在此类噪声中看到“边缘”，但是很可能会在锯齿图像中看到不正确的边缘，因此在噪声的别名方面进行权衡是一定的改进（参见图31.11）.

在积分的某个域上获取许多(随机)样本并将它们平均的想法在更普遍的情况下适用.我们可以在波长带上进行积分（而不是进行如此普遍的简单RGB计算，这相当于固定样本策略）。 我们可以在整个镜头区域进行积分以获得景深效果和色差.对于移动中的场景,我们可以通过在固定的“时间窗口”上进行积分来模拟在一定时间段内打开的快门的效果.所有这些想法在Cook等人[CPC84]的经典论文中都有描述,它称为进程分布式射线跟踪.由于可能会与分布式处理的概念混淆,我们更喜欢术语库克现在用来描述算法[Coo10]的分布射线跟踪.我们将在第32章中讨论在分布射线跟踪中使用的特定采样策略.