

高效蒙特卡罗方法在偏微分方程 初边值问题中的应用*

游 皎, 李万爱

(中山大学中法核工程与技术学院, 广东 珠海 519082)

摘 要: 介绍了蒙特卡罗方法求解偏微分方程初边值问题的基本原理, 并针对该方法在多节点、多游动次数计算中速度慢的问题, 提出了一种改进方案。通过理论分析和算例验证, 在相同计算条件下与传统方法相比, 该改进方案不仅大大减少了计算时间, 而且降低了误差, 这将使蒙特卡罗方法得到更广泛的应用。

关键词: 初边值问题; 蒙特卡罗方法; 随机游动

中图分类号: O21 **文献标志码:** A **文章编号:** 0529-6579(2015)06-0037-05

The Efficient Monte-Carlo Method in Solving the Initial-Boundary Value Problem of Partial Differential Equations

YOU Jiao, LI Wanai

(Sino-French Institute of Nuclear Engineering & Technology, Sun Yat-sen university, Zhuhai 519082, China)

Abstract: The primary principle of Monte Carlo method in solving the initial-boundary value problem for partial differential equations is introduced. And an improved strategy to solve the slow speed problem in computations with large grid points and random walks is proposed. Through the theoretical analysis and numerical validation, the improved strategy not only takes much less computational time, but also gives more accurate results comparing to the traditional methods under the same conditions.

Key words: initial-boundary value problem; Monte Carlo method; random walk

蒙特卡罗方法是一种利用计算机进行统计模拟的数值计算方法, 已经被应用到包括物理、生物、化学和金融等众多领域^[1-2]。而在科学计算中常常需要求解偏微分方程初边值问题, 而该类问题传统上均使用确定性方法如有限元方法。蒙特卡罗与有限元方法的区别在于: ①后者需要使用方程的弱解形式进行离散, 而前者可以使用各种有效的游走规则; ②后者求解大型线性方程组获得求解点值, 而前者利用概率统计理论建立求解值与已知值的关系; ③后者在非确定性问题上的求解需要建立相关随机模型, 而前者自然地具有概率性求解的特点。故一些文献也开始探索如何有效地利用蒙特卡罗方法来求解微分方程^[3-8], 为了使结果更加精确, 利

用蒙特卡罗法时往往需要对求解域进行细密的网格划分。而节点数增大, 游走次数增多后, 计算量变得很大, 需要较长计算时间, 这严重限制了蒙特卡罗方法的实际应用。文献[4]提出了一种改进方法, 该方法已经比已有方法(文献[5-6])在计算效率上有较大改进。本文继续对文献[4]方法加以改进, 方程边界条件处理采用文献[7]方法, 并通过理论分析和算例结果说明, 本文的改进方法在相同的条件下, 不仅可以大大减少计算时间, 而且降低了误差。这些改进将使蒙特卡罗方法能应用到更为占用计算资源的初边值偏微分方程计算中。

* 收稿日期: 2015-01-17

基金项目: 国家自然科学基金青年基金资助项目(11402313); 超级计算应用领域资助项目(141gjc10)

作者简介: 游皎(1992年生), 男; 研究方向: 核工程与技术; 通讯作者: 李万爱; E-mail: liwai@mail.sysu.edu.cn

1 基本原理及改进方案

1.1 传统方案

用蒙特卡罗方法求解微分方程, 首先要构造方程对应的随机概率模型。为具体起见, 本文首先以二维拉普拉斯方程为例, 对利用蒙特卡罗方法解决边值问题进行描述。设问题的控制方程为

$$\nabla^2 u = 0, (x, y) \in \Omega \quad (1)$$

其中 u 为要求解物理量, Ω 为求解域。它满足 Dirichlet 边界条件

$$u|_{\partial\Omega} = u_p(x, y)$$

其中 $\partial\Omega$ 为 Ω 的边界, $u_p(x, y)$ 为边界值函数。首先对求解域 Ω 进行网格划分, 然后相应的离散格式作为蒙特卡罗方法的游动规则。这里以正规结构化网格上的差分格式为例, 采用五点差分格式离散上面的偏微分方程, 可得

$$u_{i,j} = p_{i+}u_{i+1,j} + p_{i-}u_{i-1,j} + p_{j+}u_{i,j+1} + p_{j-}u_{i,j-1} \quad (2)$$

其中 (i, j) 为网格点编号, 在均匀网格中有

$$p_{i+} = p_{i-} = p_{j+} = p_{j-} = \frac{1}{4}$$

这个差分格式求解可以看作是一种随机游动过程, 即位于点 (i, j) 子分别以概率 $p_{i+}, p_{i-}, p_{j+}, p_{j-}$ 向四个相邻点游动, 方向取决于按这四个概率产生的随机数。定义随机变量 X , 当粒子游至边界 Q 点时, X 取边界值 $u_Q^{(1)}$ 。重复 m 次游动, 则 X 有 m 个取值 $u_Q^{(k)}, k = 1, \dots, m$ 。根据大数定理, 当 m 趋于正无穷时, 其平均值收敛于 X 的数学期望 $E(X)$, 而 $E(X)$

正是 $u_{i,j}$ 的无偏估计量, 即 $u_{i,j} = \frac{1}{m} \sum_{k=1}^m u_Q^{(k)}$ 。同理可以算出每一个网格点的物理量值。我们记这个随机游动过程为随机游动 1, 是用传统蒙特卡罗方法解决边值问题的思路。

可以将这种思路推广到微分方程初边值问题中, 以扩散方程为例,

$$\frac{\partial u}{\partial t} = \lambda \nabla^2 u, (x, y) \in \Omega \quad (3)$$

其中方程的边值和初值如下

$$u(t, (x, y) \in \partial\Omega) = f(t, x, y), \\ u(0, x, y) = g(x, y).$$

对该方程采用时间后差、空间中心差的差分格式 (BTCS) 并整理可得

$$u_{i,j}^{n+1} = \lambda [u_{i,j}^n + \sigma_x(u_{i+1,j}^{n+1} + u_{i-1,j}^{n+1}) + \sigma_y(u_{i,j+1}^{n+1} + u_{i,j-1}^{n+1})] = p_0 u_{i,j}^n + p_1(u_{i+1,j}^{n+1} + u_{i-1,j}^{n+1}) + p_2(u_{i,j+1}^{n+1} + u_{i,j-1}^{n+1}) \quad (4)$$

其中

$$\sigma_x = \frac{\lambda \Delta t}{\Delta x^2}, \sigma_y = \frac{\lambda \Delta t}{\Delta y^2}, \lambda = \frac{1}{(1 + 2\sigma_x + 2\sigma_y)},$$

$$p_0 = \lambda, p_1 = p_0 \sigma_x, p_2 = p_0 \sigma_y$$

同理想想一种随机游动过程, 粒子以 p_1 概率向 $(i-1, j)$ 和 $(i+1, j)$ 游动, 以 p_2 概率向 $(i, j-1)$ 和 $(i, j+1)$ 游动, 同时以 p_0 概率回到上一个时间点 $u_{i,j}^m$ 。当游动到边界 $\partial\Omega$ 或初始时刻 $t = 0$ 时就停止, 并取值 $u_Q^{(1)}$ 作为第一次随机游动获得值。实施 m 次该随机游动, 最终该点的值将趋近于所有游动的期望值 $u_{i,j}^{n+1} = \frac{1}{m} \sum_{k=1}^m u_Q^{(k)}$ 。记这个随机游动过程为随机游动 2。

1.2 改进方案

上面使用蒙特卡罗方法求解偏微分方程时需要使用差分格式来对方程进行离散。在工程计算里常常需要采用较密网格来得到可靠的结果, 同时由于随机游动过程具有随机性, 通常 m 要取得很大时才能得到较为准确的结果。因此这种方法的计算量比较大, 需要一些提高计算速度的方法来加速这个计算过程^[1,4]。而文献 [4] 的方法在加快计算速度及提高精度上都更有优势, 它的改进步骤为下面的随机游动 3:

假设已通过 m 次游动获得某一点 (i, j) 的函数值 (记为 $u_{i,j}$), 当以另一点 (p, q) 为起点的粒子游动到 (i, j) 时停止游动, 并取本次游动的值为 $u_Q^{(k)}(p, q) = u_{i,j}$, 然后开始下一次游动。

事实上, 所有从 (p, q) 出发并经过 (i, j) 的游动, 其最后取值的数学期望可取成 $u_{i,j}$ 。这种改进方法利用已经获得的值来缩短随机游动的链长度, 避免了所有经过点 (i, j) 后游动到边界的计算, 大大减少了计算时间。这里要指出的是, 这种利用已知函数值的思路同时也可以推广到初值问题 (如方程) 的求解, 即得到下面的随机游动 4:

假设已经计算 n 时刻所有点的函数值 $u_{i,j}^n$ 。计算 $n+1$ 时刻的函数值时, 在一点 (i, j) 为起点的粒子游动计算中, 根据离散格式, 粒子向点 $(i-1, j)$ 、 $(i+1, j)$ 、 $(i, j-1)$ 、 $(i, j+1)$ 和上一步 n 时刻 (i, j) 游动的概率为 p_1 、 p_1 、 p_2 、 p_2 和 p_0 。只要满足下面任一条件, 游动停止并取得该点函数值 $u_Q^{(k)}$: ①到达边界点; ②到达上一时间步即 n 时刻; ③该点为 $n+1$ 时刻已经获得的函数值。游动停止条件归纳起来就是游动到的点函数值已经获得。

在这个误差减少的分析基础上, 本文提出一种先计算“大网格”的方法, 通过改变网格点的计

算顺序以达到更好的优化。具体来说，若求解域划分为 50×50 的网格，可以首先计算 $i = 10, 20, 30, 40$ 和 $j = 10, 20, 30, 40$ 这 4 行 4 列的函数值，相当于把求解域分成了 25 个小求解域，其边界由外边界和已知近似值的点构成，这样大大缩短了其它点的马尔可夫链长度，因而缩短了计算时间。此外，在计算“大网格”时可以适当增大游动次数以获得更准确的值。本文中采用的改进版均是首先计算了间隔为 5 的网格点的值，再进行其他点的计算。下面将这种方法称为 reorder 方法。

1.3 误差分析

随机游动 4 其实是随机游动 3 在初边值问题的推广。虽然文献 [4] 已经验证过随机游动 3 的精度比随机游动 1 更为提高，但是没有进行数学原理上的分析证明。这里将给出随机游动 3 比 1、随机游动 4 比 2 能够计算更准确的分析。

这里有两个误差的问题。首先是游动规则带来的误差，本文的游动规则是通过有限差分方法给出的，其空间误差为截断误差

$$u_{xx} = \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\Delta x^2} + O(\Delta x^2),$$

$$u_{yy} = \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{\Delta y^2} + O(\Delta y^2)$$

时间误差为 $O(\Delta t)$ 。

第二个误差为游动过程的概率统计误差。首先需要研究对于传统方法，游动次数与误差之间的关系。利用文献 [6] 的类似分析，以拉普拉斯方程为例，对于在求解域 Ω 内任意一个网格点 A ，由公式可以推得满足方程的近似解为

$$u^*(A) = \sum_{j=1}^s u_p(Q_j)P(A, Q_j) \quad (5)$$

式中 s 为边界点个数， $u_p(Q_j)$ 为边界点 Q_j 对应的函数值 $u_p(x, y)$ ； $P(A, Q_j)$ 表示粒子从 A 点出发后游动到边界 Q_j 的概率。另外，定义随机变量

$$\xi_{i,j} = \begin{cases} 1, & \text{第 } i \text{ 个粒子到达边界点 } Q_j \\ 0, & \text{第 } i \text{ 个粒子不到达边界点 } Q_j \end{cases}$$

$\xi_{i,j}$ 的期望 $E(\xi_{i,j})$ 为

$$E(\xi_{i,j}) = 0 \times P(\xi_{i,j} = 0) + 1 \times P(\xi_{i,j} = 1) = P(A, Q_j)$$

方差 $V(\xi_{i,j})$ 为

$$V(\xi_{i,j}) = E(\xi_{i,j}^2) - E(\xi_{i,j})^2 = P(A, Q_j)(1 - P(A, Q_j)) \leq \frac{1}{4}$$

用 n_j 表示 m 个游动粒子中到达边界点 Q_j 的粒子数目，有

$$n_j = \sum_{i=1}^m \xi_{i,j}, m = \sum_{j=1}^s n_j$$

则粒子从 A 点出发后游动到边界 Q_j 的频率为 $\frac{n_j}{m}$ ， $u(A)$ 的计算值为

$$u(A) = \sum_{j=1}^s u_p(Q_j) \frac{n_j}{m}$$

由契比雪夫不等式，结合式和式，对任意一给定的正数 $\varepsilon_1 > 0$ ，有

$$P\left\{\left|\frac{n_j}{m} - P(A, Q_j)\right| \geq \varepsilon_1\right\} \leq \frac{V(\xi_{i,j})}{\varepsilon_1^2 m} \leq \frac{1}{4\varepsilon_1^2 m}$$

结合式和式，对任意小的正数 ε_2 ，有

$$P\{|u(A) - u^*(A)| \geq \varepsilon_2\} = P\left\{\left|\sum_{j=1}^s u_p(Q_j) \left[\frac{n_j}{m} - P(A, Q_j)\right]\right| \geq \varepsilon_2\right\}$$

由于 $\xi_{i,j} (i = 1, 2, \dots, m)$ 具有相同的概率分布，并且相互独立，因此有

$$P\left\{\left|\sum_{j=1}^s u_p(Q_j) \left[\frac{n_j}{m} - P(A, Q_j)\right]\right| \geq \varepsilon_2\right\} \leq \sum_{j=1}^s P\left\{\left|u_p(Q_j) \left[\frac{n_j}{m} - P(A, Q_j)\right]\right| \geq \varepsilon_2\right\} \leq \sum_{j=1}^s P\left\{\left|\left[\frac{n_j}{m} - P(A, Q_j)\right]\right| \geq \frac{\varepsilon_2}{u_p(Q_j)}\right\}$$

所以

$$P\{|u(A) - u(A)^*| \geq \varepsilon_2\} \leq \sum_{j=1}^s \frac{u_p(Q_j)^2}{4\varepsilon_2^2 m}$$

这个结论说明一点的游动次数 m 越大，则该点误差越小。在离散公式 (2) 在均匀网格为例

$$u_{i,j}^{n+1} = \frac{1}{4}(u_{i+1,j}^{n+1} + u_{i-1,j}^{n+1} + u_{i,j+1}^{n+1} + u_{i,j-1}^{n+1})$$

一点的函数值取决于相邻四点的值。如果以该点 (i, j) 为起点需要游动 m 次，则对于传统方案，可以假定每个相邻点的值都经由 $m/4$ 次游动后得到。若有一个邻点的值是已知的，即它已经经历过至少 m 次随机游动，直接使用邻点的已知函数值相当于过该邻点的游动次数至少为 $m^2/4$ (乘法原理)。平均来说有两个邻点函数值已知则总次数为 $m/2 + m^2/2$ ，这将大大增加随机游动次数从而减少误差。

2 数值例子

2.1 已知精确解的二维泊松方程

考虑一个温度分布 $T(x, y)$ 在域 $\Omega = [0, 1] \times [0, 1]$ 满足方程如下的泊松方程，

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = -2\pi^2 \sin(\pi x) \sin(\pi y)$$

且边界条件为 $T(x, y) = 1 + x$ 。则此方程有解析解

$$T(x, y) = 1 + x + \sin(\pi x) \sin(\pi y)$$

分别采用传统的 (随机游动 1, 对应表 1 方案 0) 和改进后 (随机游动 3 + reorder 方法, 对应表 1 方案 1) 的方案对上述问题进行求解, 并改变参数条件, 进行横向和纵向的对比。由于蒙特卡罗方法具有随机性, 表 1 中每个方案都进行了 3 次实验, 结果取平均值。在表 1 中, N 为网格节点数, m 为每个节点的游动次数, 时间单位为秒, 时间比为方案 1 的时间与方案 0 的时间之比, 误差采用 L_1 范数, 相对误差为误差与精确解之比。可以看出, 在相同的条件下, 改进版不仅在时间上较传统版有巨大优势, 而且在降低误差方面也表现得更好。

表 1 泊松方程的蒙特卡罗求解方法的误差对比

Table 1 Error comparison using MC methods for Poisson equation

N	m	方案	时间/s	时间比/%	平均误差	比值/%	相对误差/%
81^2	1 000	1	16.5		0.005 0		0.3
81^2	1 000	0	1 174.0	1.4	0.014 1	35	0.8
81^2	500	1	8.4		0.007 6		0.3
81^2	500	0	589.6	1.4	0.019 7	38	1.1
41^2	1 000	1	2.6		0.008 1		0.4
41^2	1 000	0	73.6	3.5	0.013 9	58	0.7
41^2	500	1	1.3		0.009 8		0.5
41^2	500	0	36.7	3.5	0.018 1	54	1.0

2.2 二维非定常热传导问题

考虑如下二维非定常热传导方程

$$\partial u / \partial t = \lambda \nabla^2 u, (x, y) \in [0, 1] \times [0, 1], \lambda = 0.5$$

其边界条件及初始条件分别为

$$u(t, 0, y) = u(t, 1, y) = u(t, x, 0) = u(t, x, 1) = 0,$$

$$u(x, y, 0) = \exp(-20(x - 1/2)^2 - 20(y - 1/2)^2)$$

由于非定常问题传统方法 (即随机游动 2) 在二维中计算量过大, 这里的传统方法采用随机游动 2 加上随机游动 4 中的时间改进方案。使用传统方法 (对应表 2 方案 0) 和改进方案 (随机游动 4 + reorder 方法, 对应表 2 方案 1) 求解 $t = 0.05$ 时的分布。用二阶 Crank-Nicolson 差分法取精细条件 ($N = 1\,000 \times 1\,000$, $dt = 0.001$) 作为参考解, 以此计算两种方案的误差。其计算结果及时间、误差的比较如表 2 所示。可以看出改进方案不仅大程度地压缩了运行时间, 而且在一定程度上减少了误差。

为了进一步分析计算结果, 本文还作了温度等值线进行对比, 如图 1 所示。改进方案的等高线更加圆滑, 说明该方案的随机误差更小, 结果更稳定。

表 2 二维非定常热传导问题的蒙特卡罗法误差比较 ($N = 51^2, m = 1\,000$)

Table 2 Error comparison using MC methods for 2D unsteady heat conduction equation

方案	时间/s	时间比/%	误差	相对误差/%
1	4.1		0.007 2	6.4
0	27.2	15.1	0.007 8	7.0

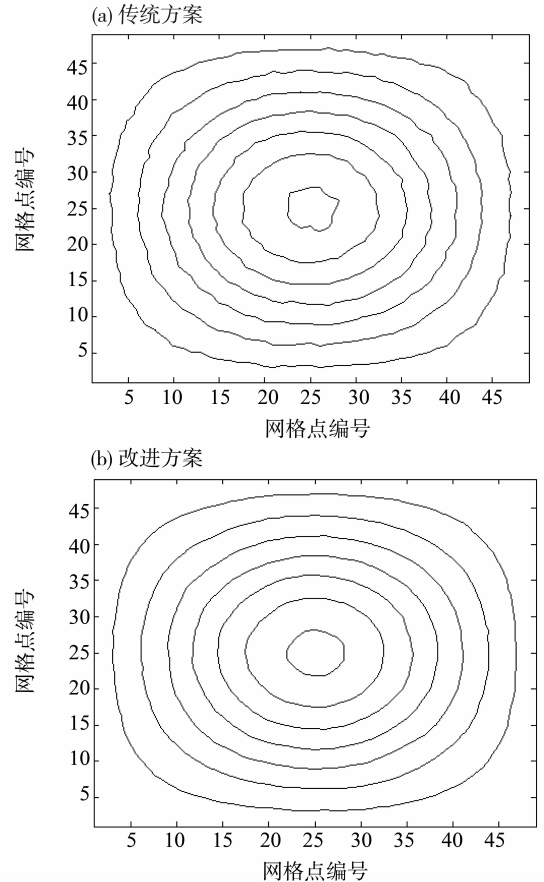


图 1 常热传导方程蒙特卡罗法计算的温度等值线对比

Fig. 1 Thermal isolines comparison using MC methods on unsteady heat conduction equation

3 总结

本文研究了蒙特卡罗方法在微分方程初边值问题中的应用, 针对传统方法计算速度较慢的问题, 提出了一种适合多节点、多游动次数的改进方法并给出精度提高的分析证明过程。与较传统方法相比, 改进方案无论在计算时间和结果精度上都有明显优势, 适合蒙特卡罗方法在更加复杂工程问题中的应用。下一步工作将结合曲面处理技术^[9], 将该方法推广到一般的几何计算区域中。

- and Clouds (ACDC'09), Barcelona, Spain, 2004. New York, NY, USA: ACM, 2009: 37–42.
- [21] ZHU Q B, DAVIDF M, DEVARAJ C F, et al. Reducing energy consumption of disk storage using power-aware cache management [C] // Proceedings of the 10th International Conference on High-Performance Computer Architecture (HPCA'04), Madrid, Spain, Piscataway, NJ, USA: IEEE, 2004: 118–129.
- [22] MOHAMED Y, TIAN Y Y, OZCAN F, et al. CoHadoop: flexible data placement and its exploitation in Hadoop [C] // Proceedings of the VLDB Endowment VLDB 2011, 2011, 4(9): 575–585.
- [23] DITTRICH J, QUIANE-RUIZ J A, JINDAL A, et al. Hadoop ++: Making a yellow elephant run like a cheetah (without it even noticing) [C] // Proceedings of the PVLDB 2010, 2010, 3(1/2): 518–529.
- [24] ZHAO Y R, WANG W P, MENG D, et al. A data locality optimization algorithm for large-scale data processing in Hadoop [C] // Processings of the ISCC 2012, 2012: 655–661.
- [25] 赵彦荣, 王伟平, 孟丹, 等. 基于 Hadoop 的高效连接查询处理算法 CHMJ[J]. 软件学报, 2012, 23(8): 2023–2041.
- [26] 廖彬, 于炯, 张陶, 等. 基于分布式文件系统 HDFS 的节能算法[J]. 计算机学报, 2013, 36(5): 1047–1064.
- [27] 廖彬, 于炯, 孙华, 等. 基于存储结构重配置的分布式存储系统节能算法[J]. 计算机研究与发展, 2013, 50(1): 3–18.

(上接第 40 页)

参考文献:

- [1] 刘军. 科学计算中的蒙特卡罗策略[M]. 北京: 高等教育出版社, 2009.
- [2] 洪志敏. 基于 Monte-Carlo 技术的积分(微分)方程数值求解方法研究[D]. 内蒙古: 内蒙古工业大学博士学位论文.
- [3] 左应红, 王建国. 蒙特卡罗方法在解微分方程边值问题中的应用[J]. 强激光与粒子束, 2012, 24(12): 3023–3026.
- [4] 周泉, 陈植武, 詹杰民. 蒙特卡罗法的改进: 一种新型的快速算法[C] // 第二十一届全国水动力学研讨会暨第八届全国水动力学学术会议暨两岸船舶与海洋工程水动力学研讨会文集, 2008.
- [5] 林建国, 周俊陶. 蒙特卡罗法的一种快速计算: 在势流问题中的应用[J]. 水动力学研究与进展 A 辑, 2005, 20(3): 405–410.
- [6] 史慧会, 曲小钢. 蒙特卡罗方法在热传导方程中的应用[J]. 重庆工学院学报: 自然科学版, 2008, 22(11): 101–103.
- [7] VAJARGAH B F, VAJARGAH K F. Monte Carlo method for finding the solution of Dirichlet partial differential equations [J]. Applied Mathematical Sciences, 2007, 1(10): 453–462.
- [8] LAUDU D P, BINDER K. A guild to Monte Carlo simulations in statistical physics [M]. Cambridge University Press, 2009.
- [9] JUN S, SUNGHWAN Y, NAM Z C. A Monte Carlo method for solving heat conduction problems with complicated geometry [J]. Nuclear Engineering and Technology, 2007, 29(3): 207–214.