

随机微分方程数值模拟的一种算法介绍

Desmond J. Higham

摘要 本文给出了一种实用的容易理解的对随机微分方程数值方法的介绍.

假定读者熟悉确定性微分方程的 Euler (欧拉) 方法, 并且至少对随机变量的概念有一个直觉; 但是不需要有高级概率理论和随机过程理论知识. 文章围绕 10 个 MATLAB 程序展开, 涵盖随机积分, Euler-Maruyama (欧拉-丸山) 方法, Milstein 方法, 强弱收敛性, 线性稳定性和随机链法则.

1. 引言

随机微分方程 (SDE) 模型在包括生物, 化学, 流行病学, 力学, 微电子学, 经济学和金融学的一系列应用领域发挥突出的作用. 对 SDE 理论的完全理解需要熟悉高等概率和随机过程理论; 学习这些材料可能会使一个典型的学习应用数学的学生沮丧. 但是, 即便只有关于确定性常微分方程的 Euler 方法的背景知识和对随机变量的直觉理解, 也可以了解数值模拟 SDE 的基础理论. 并且, 关于数值方法的经验给出了 SDE 基础理论有用的第一步. 因此, 在本文中, 我们介绍如何对一个 SDE 应用简单的数值方法并从实用的观点讨论一些概念, 例如收敛和线性稳定性. 我们主要的目标读者包括高年级的本科生和刚入学的研究生.

我们的目标是使用最少的理论. 但是, 我们倚赖一个基本的假设, 即假设读者对随机变量, 独立性, 期望值和方差至少有一个粗略的感觉, 并且, 特别地, 对正态分布的随机变量的概念熟悉. 我们的数值试验使用蒙特卡罗 (Monte Carlo) 方法: 用一个随机数发生器模拟随机变量, 用计算的平均值来近似期望值.

通过例子学习是学习的最好途径, 所以本文以 10 个 MATLAB 程序 [3, 13] 为基础, 使用类似于 [14] 的基本原理. 在网址 <http://www.maths.strath.ac.uk/~aas96106/algfiles.html> 可下载这些程序. MATLAB 对这种类型的处理是理想的环境, 不仅因为它能生成高水平的随机数和方便制图. 这些程序尽可能地合理的短并且被设计得运行快速 (在现代台式电脑上的运行时间小于 10 分钟). 为了达到这些要求, 我们发现必须“向量化” MATLAB 编码. 我们希望程序中的文本行和我们对文本中重要部分的讨论使程序列表对所有的有一些科学编程经验的读者来说是易懂的.

在下一节我们介绍 Brown (布朗) 运动的概念并且计算离散化的 Brown 路径. 在 §3 我们以关于 Brown 运动积分为想法做试验, 并且说明 Itô (伊藤) 积分和 Stratonovich (斯特拉托诺维奇) 积分的不同之处. 我们在 §4 描述如何使用 Euler-Maruyama 方法模拟一个 SDE. 我们在 §5 介绍强弱收敛的概念, 并且在数值上验证 Euler-Maruyama $1/2$ 阶强

译自: SIAM Review, Vol.43 (2001), No.3, p.525-546, Desmond J. Higham: An Algorithmic Introduction to Numerical Simulation of Stochastic Differential Equations, figure number 7. Copyright ©2001 Society for Industrial and Applied Mathematics. Reprinted with permission. All rights reserved. 美国工业和应用数学学会与作者授予译文出版许可.

收敛和 1 阶弱收敛. 在 §6 我们考虑 Milstein 方法, 为了达到 1 阶强收敛, 这个方法在 Euler-Maruyama 方法上增加了一个修正项. 在 §7 我们介绍两类截然不同的关于 Euler-Maruyama 方法的线性稳定性. 为了强调随机计算本质上不同于确定性计算, 我们在 §8 提出并且在数值上验证随机链法则. §9 简要提及了一些其他重要问题作为结束, 这些问题中很多代表了活跃的研究领域.

我们将提到一些关键资料, 而不是反复的引用而使得文章更加艰深. 对那些被启迪学习更多关于 SDEs 及其数值解的人, 我们推荐 [6] 作为综合的参考书, 这本书包括概率和随机过程必需的材料. 评论文章 [11] 有最新的关于数值方法的文献. 其他 3 个关于 SDEs 的易懂的参考书是 [1], [8] 和 [9], 前两本给出了一些对数值方法的讨论. [10] 的第 2 章和第 3 章给出了一个 SDEs 的自包含的处理和它们的数值解, 这些数值解在聚合体流中有一些应用. [5] 中深刻阐述了 Brown 运动和随机计算的基础理论. §7 中关于线性稳定性的材料基于 [2] 和 [12].

2. Brown 运动

$[0, T]$ 上的一个标量的 标准 Brown 运动, 或 标准 Wiener(维纳) 过程, 是一个连续依赖于 t 的随机变量 $W(t)$, 并且满足以下 3 个条件.

1. $W(0) = 0$ (概率为 1).

2. 对 $0 \leq s < t \leq T$, 由增量 $W(t) - W(s)$ 给出的随机变量是均值为 0, 方差为 $t - s$ 的正态分布; 等价地, $W(t) - W(s) \sim \sqrt{(t-s)}N(0, 1)$, 这里 $N(0, 1)$ 表示一个均值为 0, 方差为 1 的正态分布的随机变量.

3. 对于 $0 \leq s < t < u < v \leq T$, 增量 $W(t) - W(s)$ 和 $W(v) - W(u)$ 是独立的.

为了计算的目的, 考虑离散的 Brown 运动是有帮助的, 这里 $W(t)$ 在离散 t 值上给定. 于是对一些正整数 N , 我们设 $\delta t = T/N$, 并且让 W_j 表示 $W(t_j)$, $t_j = j\delta t$. 条件 1 指出 $W_0 = 0$ 的概率为 1, 条件 2 和 3 告诉我们

$$W_j = W_{j-1} + dW_j, \quad j = 1, 2, \dots, N, \quad (2.1)$$

这里每个 dW_j 是一个形如 $\sqrt{\delta t}N(0, 1)$ 的独立随机变量.

表 1 中 MATLAB 程序 M-file `bpath1.m` 执行了一个在 $[0, 1]$ 上 $N = 500$ 的离散 Brown 运动的模拟. 这里使用了随机数发生器 `randn`——每次访问 `randn` 产生一个 $N(0, 1)$ 分布的独立“伪随机”数. 为了使得试验是可重复的, MATLAB 允许任意设置随机数发生器的初始状态. 我们用命令 `randn('state', 100)` 将初始状态设为 100. 接着运行 `bpath1.m` 会产生相同的结果. 通过重新设置初始状态, 如 `randn('state', 200)`, 可执行不同的模拟. 由 `randn` 产生的数字被 $\sqrt{\delta t}$ 尺度化, 并且被用作 `for` 循环中的增量, 产生 $1 \times N$ 的数组 `W`. 有一个小的不方便: MATLAB 从指标 1 开始排列数组, 而不是从指标 0 开始. 因此, 我们用 `W(1), W(2), \dots, W(N)` 表示 `W`, 并用 `plot([0:dt:T], [0, W])` 作图, 从而在图形中包括了初始值 $W(0)=0$. 图 1 显示了结果; 注意到为了视觉效果, 我们把离散的点连接成了直线. 我们将由 `bpath1` 中的算法得到的数组 `W` 称作 离散化的 Brown 路径.

我们可以通过把循环指令 `for` 替换为更高级的“向量化”命令来更优美更有效率地执

表 1 M-file bpath1.m

%BPATH1 Brownian path simulation

```

randn('state',100)           % set the state of randn
T = 1; N = 500; dt = T/N;
dW = zeros(1,N);             % preallocate arrays ...
W = zeros(1,N);               % for efficiency

dW(1) = sqrt(dt)*randn;       % first approximation outside the loop ...
W(1) = dW(1);                 % since W(0) = 0 is not allowed
for j = 2:N
    dW(j) = sqrt(dt)*randn;    % general increment
    W(j) = W(j-1) + dW(j);
end

plot([0:dt:T],[0,W],'r-')     % plot W against t
xlabel('t','FontSize',16)
ylabel('W(t)','FontSize',16,'Rotation',0)

```

表 2 M-file bpath2.m

%BPATH2 Brownian path simulation: vectorized

```

randn('state',100)           % set the state of randn
T = 1; N = 500; dt = T/N;

dW = sqrt(dt)*randn(1,N);     % increments
W = cumsum(dW);                % cumulative sum

plot([0:dt:T],[0,W],'r-')     % plot W against t
xlabel('t','FontSize',16)
ylabel('W(t)','FontSize',16,'Rotation',0)

```

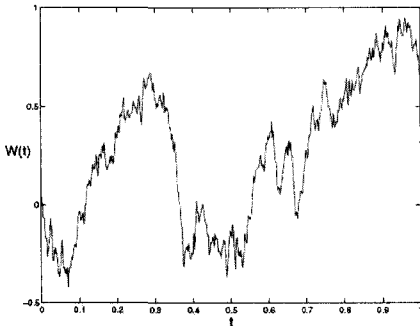


图 1 离散化的从 bpath1.m 到 bpath2.m 的 Brown 路径

行相同的计算,如表 2 中 bpath2.m 所示. 这里,我们给随机数发生器提供两个变量:randn(1,N) 产生一个独立的 $N(0,1)$ 样本的 $1 \times N$ 数组. 函数 cumsum 计算了它的变量的累积和,所以按照要求, $1 \times N$ 的数组 W 的第 j 项是 $dW(1)+dW(2)+\dots+dW(j)$. 避免循环指令 for, 因此直接用数组计算而不是独立的分量计算是写出高效的 MATLAB 编码 [3, Chapter 20] 的关键. 本文中的一些 M-files 如果写成非向量化形式,将慢上好几个量级.

表 3 中的 M-file bpath3.m 生成图 2. 这里,我们沿着 1000 个离散化的 Brown 路径计算函数 $u(W(t)) = \exp(t + \frac{1}{2}W(t))$. $u(W(t))$ 在这些路径上的均值用一条蓝色实线标出. 5 条单独的路径用红色虚线标出. M-file bpath3.m 是沿着路径向量化的; dW 是一个 $M \times N$ 数组,使得 dW(i,j) 给出 (2.1) 中的第 i 条路径的增量 dW_j . 我们使用命令 cumsum(dW, 2)

表 3 *M*-file bpath3.m**%BPATH3 Function along a Brownian path**

```

randn('state',100)                                % set the state of randn
T = 1; N = 500; dt = T/N; t = [dt:dt:1];

M = 1000;                                           % M paths simultaneously
dW = sqrt(dt)*randn(M,N);                          % increments
W = cumsum(dW,2);                                   % cumulative sum
U = exp(repmat(t,[M 1]) + 0.5*W);
Umean = mean(U);
plot([0,t],[1,Umean],'b-'), hold on                % plot mean over M paths
plot([0,t],[ones(5,1),U(1:5,:)],'r--'), hold off   % plot 5 individual paths
xlabel('t','FontSize',16)
ylabel('U(t)','FontSize',16,'Rotation',0,'HorizontalAlignment','right')
legend('mean of 1000 paths','5 individual paths',2)

averr = norm((Umean - exp(9*t/8)), 'inf')           % sample error

```

求沿着第二 (列) 维的累积和. 因此, W 是一个 $M \times N$ 数组, 它的第 i 行包含第 i 条路径. 我们用 $\text{repmat}(t, [M \ 1])$ 来产生一个 $M \times N$ 数组, 它的每一行都是 t . 于是 $M \times N$ 数组 U 的第 i 行对应沿着第 i 条路径的 $u(W(t))$. 用 $U_{\text{mean}} = \text{mean}(U)$ 计算每列均值, 所以 U_{mean} 是一个 $1 \times N$ 数组, 它的第 j 项是 $u(W(t_j))$ 样本均值.

在图 2 中我们看到虽然 $u(W(t))$ 沿着单独的路径是非光滑的, 但它的样本均值看起来是光滑的. 这可以被严格证明——事实上, $u(W(t))$ 的期望值是 $\exp(9t/8)$. 在 `bpath3.m` 中, `averr` 记录了在所有点 t_j 的样本均值和精确期望值之间的最大偏差. 我们得到 `averr=0.0504`. 当样本数量增加到 4000, `averr` 减少到 0.0268.

注意到, $u(W(t))$ 作为一个线性 SDE 的解具有 §4 中出现的 (4.6) 的形式. 在一些应用中, 需要给定路径求解——一个所谓的路径解或者强解.

我们在 §5 中将会看到, 一个计算强解方法的优劣一般是由强收敛的阶决定的. 在其他的背景中, 人们只对解的期望值类型的信息感兴趣, 这引出了弱收敛的概念.

3. 随机积分

给定一个合适的函数 h , 积分 $\int_0^T h(t)dt$ 可以用 Riemann (黎曼) 和

$$\sum_{j=0}^{N-1} h(t_j)(t_{j+1} - t_j), \quad (3.1)$$

来逼近, 在 §2 已介绍过, 这里离散点 $t_j = j\delta t$. 事实上, 积分可以由在 (3.1) 中取 $\delta t \rightarrow 0$ 的极限来定义. 用类似的方法, 我们可以考虑一个如下形式的和

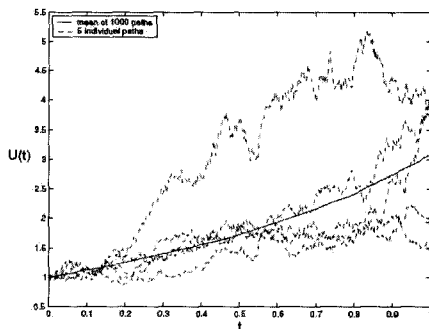


图 2 由 `bpath3.m` 得到的 1000 条 Brown 路径的均值和 5 条单独的路径

表 4 M-file stint.m

```
%STINT Approximate stochastic integrals
%
% Ito and Stratonovich integrals of W dW

randn('state',100) % set the state of randn
T = 1; N = 500; dt = T/N;

dW = sqrt(dt)*randn(1,N); % increments
W = cumsum(dW); % cumulative sum

ito = sum([0,W(1:end-1)].*dW)
strat = sum((0.5*[0,W(1:end-1)]+W) + 0.5*sqrt(dt)*randn(1,N)).*dW

itoerr = abs(ito - 0.5*(W(end)^2-T))
straterr = abs(strat - 0.5*W(end)^2)
```

$$\sum_{j=0}^{N-1} h(t_j)(W(t_{j+1}) - W(t_j)), \quad (3.2)$$

与 (3.1) 类似, 这可以看作随机积分 $\int_0^T h(t)dW(t)$ 的近似. 这里, 我们关于 Brown 运动对 h 作积分.

在表 4 的 M-file stint.m 中, 我们创建了一个 $[0, 1]$ 上 $\delta t = 1/N = 1/500$ 的离散的 Brown 路径, 并且对 $h(t)$ 等于 $W(t)$ 的情形求和 (3.2). 这个积分和被计算为变量 ito. 这里 $*$ 表示按元素的乘积, 所以 $[0, W(1:end-1)].*dW$ 表示一个第 j 个元素是 $W(j-1)*dW(j)$ 的 $1 \times N$ 数组. 于是用函数 sum 来执行所需要的求和, 生成结果 ito=-0.2674.

(3.1) 的另一种表示由

$$\sum_{j=0}^{N-1} h\left(\frac{t_j + t_{j+1}}{2}\right)(t_{j+1} - t_j), \quad (3.3)$$

给出, 这也是一个 $\int_0^T h(t)dt$ 的 Riemann 和近似. (3.2) 对应的另一种表示是

$$\sum_{j=0}^{N-1} h\left(\frac{t_j + t_{j+1}}{2}\right)(W(t_{j+1}) - W(t_j)). \quad (3.4)$$

当 $h(t) \equiv W(t)$ 时, 和 (3.4) 要求计算 $W(t)$ 在点 $t = (t_j + t_{j+1})/2$ 的值. 可以证明, 作 $(W(t_j) + W(t_{j+1}))/2$ 再添加一个独立的 $N(0, \Delta t/4)$ 增量, 给出 $W((t_j + t_{j+1})/2)$ 的一个值, 这个值保持 §2 开始列出的 3 个条件. 用这个方法, 积分和 (3.4) 在 stint.m 中被计算为 strat, 在那里, 我们得到 strat=0.2354. 注意到那两个“随机 Riemann 和” (3.2) 和 (3.4) 给出了显著不同的答案. 用更小的 δt 做进一步的试验显示, 当 $\delta t \rightarrow 0$ 时, 这种不匹配也不消失. 这个现象突出了确定性积分与随机积分之间的重要不同——当定义一个随机积分为 Riemann 和的极限情形时, 我们必须精确地知道积分和是如何构成的. “左手”积分和 (3.2) 导出所谓的 Itô 积分, 而“中点”积分和 (3.4) 则给出 Stratonovich 积分.¹⁾

1) 基于和 $\sum_{j=0}^{N-1} \frac{1}{2}(h(t_j) + h(t_{j+1}))(W(t_{j+1}) - W(t_j))$, 一些作者喜欢用一个与 Stratonovich 积分几乎等价的定义.——原注

精确估计在 `stint.m` 中被逼近的随机积分是可能的. Itô 版本的积分是

$$\begin{aligned}\sum_{j=0}^{N-1} W(t_j)(W(t_{j+1}) - W(t_j)) &= \frac{1}{2} \sum_{j=0}^{N-1} (W(t_{j+1})^2 - W(t_j)^2 - (W(t_{j+1}) - W(t_j))^2) \\ &= \frac{1}{2} \left(W(T)^2 - W(0)^2 - \sum_{j=0}^{N-1} (W(t_{j+1}) - W(t_j))^2 \right) \quad (3.5)\end{aligned}$$

的极限情形. 现在可以看到 (3.5) 中的项 $\sum_{j=0}^{N-1} (W(t_{j+1}) - W(t_j))^2$ 有期望值 T 和方差 $O(\delta t)$. 因此, 对很小的 δt , 我们希望这个随机变量接近于常数 T . 更精确的论证导出, 对 Itô 积分, 有

$$\int_0^T W(t) dW(t) = \frac{1}{2} W(T)^2 - \frac{1}{2} T, \quad (3.6)$$

Stratonovich 版本的积分是

$$\sum_{j=0}^{N-1} \left(\frac{W(t_j) + W(t_{j+1})}{2} + \Delta Z_j \right) (W(t_{j+1}) - W(t_j))$$

的极限情形, 这里每个 ΔZ_j 是独立的 $N(0, \Delta t/4)$. 这个积分和简化为

$$\frac{1}{2} (W(T)^2 - W(0)^2) + \sum_{j=0}^{N-1} \Delta Z_j (W(t_{j+1}) - W(t_j)),$$

在这里, 项 $\sum_{j=0}^{N-1} \Delta Z_j (W(t_{j+1}) - W(t_j))$ 有期望值 0 和方差 $O(\delta t)$. 因此, 与 (3.6) 对应, 我们有

$$\int_0^T W(t) dW(t) = \frac{1}{2} W(T)^2. \quad (3.7)$$

在 M-file `stint.m` 中, 量 `itoerr` 和 `straterr` 分别记录了 Riemann 和 `ito` 与 `strat` 与它们各自当 $\delta t \rightarrow 0$ 时的极限情形 (3.6) 和 (3.7) 的差值. 我们发现 `itoerr`=0.0158, `straterr`=0.0186.

在数学建模中, Itô 积分和 Stratonovich 积分都有它们的用处. 在下面的章节中, 我们用 Itô 形式定义一个 SDE (一个从 Itô 积分到 Stratonovich 积分的简单转换).

4. Euler-Maruyama 方法

一个标量, 自治 SDE 可以写成积分形式

$$X(t) = X_0 + \int_0^t f(X(s)) ds + \int_0^t g(X(s)) dW(s), \quad 0 \leq t \leq T. \quad (4.1)$$

这里, f 和 g 是标量函数, 并且初始条件 X_0 是一个随机变量. (4.1) 右端的第 2 个积分是关于 Brown 运动的, 如前一节讨论的一样, 我们假定使用了 Itô 版本的积分. 解 $X(t)$ 是对 t 的随机变量. 我们并不试图进一步解释什么叫 $X(t)$ 是 (4.1) 的解 —— 而是定义一个 (4.1) 的数值解法, 然后我们可以把解 $X(t)$ 当成在数值方法中取 0 步长极限得到的随机变量.

通常, 把 (4.1) 写成微分方程形式,

$$dX(t) = f(X(t))dt + g(X(t))dW(t), \quad X(0) = X_0, \quad 0 \leq t \leq T. \quad (4.2)$$

这只不过是用紧凑的形式说明 $X(t)$ 是 (4.1) 的解. 为了保持惯例, 我们将着重强调 (4.2) 的 SDE 形式, 而不是 (4.1) 的积分形式. (注意到我们不被允许写 $dW(t)/dt$, 因为 Brown

运动处处不可微的概率为 1.) 如果 $g \equiv 0$ 且 X_0 是一个常数, 那么问题就变成确定性的, 并且 (4.2) 简化为常微分方程 $dX(t)/dt = f(X(t))$, $X(0) = X_0$.

在 $[0, T]$ 上对 (4.2) 应用数值方法, 我们首先离散化这个区间. 让 $\Delta t = T/L$, L 是某正整数, $\tau_j = j\Delta t$. 用 X_j 表示 $X(\tau_j)$ 的数值近似. Euler-Maruyama 方法 (EM) 具有形式

$$X_j = X_{j-1} + f(X_{j-1})\Delta t + g(X_{j-1})(W(\tau_j) - W(\tau_{j-1})), \quad j = 1, 2, \dots, L. \quad (4.3)$$

为了理解如何得到 (4.3), 注意到从积分形式 (4.1) 有

$$X(\tau_j) = X(\tau_{j-1}) + \int_{\tau_{j-1}}^{\tau_j} f(X(s))ds + \int_{\tau_{j-1}}^{\tau_j} g(X(s))dW(s). \quad (4.4)$$

(4.3) 右端 3 项的每一项都近似于对应的 (4.4) 右端的项. 我们也注意到, 在确定性情形中 ($g \equiv 0$ 并且 X_0 是一个常数), (4.3) 简化为 Euler 方法.

在本文中, 我们将会计算我们自己的离散化的 Brown 路径, 并且用它们来生成 (4.3) 中需要的增量 $W(\tau_j) - W(\tau_{j-1})$. 为了方便, 我们总是选择数值方法的步长 Δt 是 Brown 路径增量 δt 的 R 倍, R 是一个 ≥ 1 的整数. 这保证了离散化的 Brown 运动的基点集 $\{t_j\}$ 包含点集 $\{\tau_j\}$, 而 EM 解是在 $\{\tau_j\}$ 上计算得到的. 在一些应用中, Brown 路径被指定为问题数据的一部分. 如果有解析路径, 那么我们可以使用任意小的 Δt .

表 5 *M-file em.m*

```
%EM Euler-Maruyama method on linear SDE
%
% SDE is dX = lambda*X dt + mu*X dW, X(0) = Xzero,
% where lambda = 2, mu = 1 and Xzero = 1.
%
% Discretized Brownian path over [0,1] has dt = 2^(-8).
% Euler-Maruyama uses timestep R*dt.

randn('state',100)
lambda = 2; mu = 1; Xzero = 1; % problem parameters
T = 1; N = 2^8; dt = 1/N;
dW = sqrt(dt)*randn(1,N); % Brownian increments
W = cumsum(dW); % discretized Brownian path

Xtrue = Xzero*exp((lambda-0.5*mu^2)*([dt:dt:T])+mu*W);
plot([0:dt:T],[Xzero,Xtrue],'m-'), hold on

R = 4; Dt = R*dt; L = N/R; % L EM steps of size Dt = R*dt
Xem = zeros(1,L); % preallocate for efficiency
Xtemp = Xzero;
for j = 1:L
    Winc = sum(dW(R*(j-1)+1:R*j));
    Xtemp = Xtemp + Dt*lambda*Xtemp + mu*Xtemp*Winc;
    Xem(j) = Xtemp;
end

plot([0:Dt:T],[Xzero,Xem],'r--'), hold off
xlabel('t','FontSize',12)
ylabel('X','FontSize',16,'Rotation',0,'HorizontalAlignment','right')

emerr = abs(Xem(end)-Xtrue(end))
```

我们将对下面的线性 SDE 应用 EM 方法

$$dX(t) = \lambda X(t)dt + \mu X(t)dW(t), \quad X(0) = X_0, \quad (4.5)$$

这里, λ 和 μ 是实常数; 于是在 (4.2) 中, $f(X) = \lambda X$, $g(X) = \mu X$. 这个 SDE 可以从, 例如金融数学中的资产价值模型 [4] 中导出. (事实上, 著名的 Black (布莱克)-Scholes 偏微分方程可以从 (4.5) 中导出.) 众所周知 (例如 [8, p.105]), 这个 SDE 的精确解是

$$X(t) = X(0) \exp \left(\left(\lambda - \frac{1}{2} \mu^2 \right) t + \mu W(t) \right). \quad (4.6)$$

在表 5 的 M-file `em.m` 中, 我们考虑 (4.5), 其中 $\lambda = 2$, $\mu = 1$, 和 $X_0 = 1$ (常数). 我们计算 $[0, 1]$ 上, $\delta t = 2^{-8}$ 的一条离散化的 Brown 路径, 并且将 (4.6) 中解的值计为 `Xtrue`. 这在图 3 中用一条实的紫红色的线标出. 然后我们使用步长 $\Delta t = R\delta t$, $R = 4$ 来应用 EM. 用 EM 方法 (4.3) 计算通常的一步需要增量 $W(\tau_j) - W(\tau_{j-1})$, 这个增量由

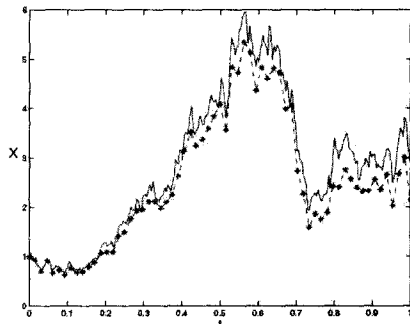


图 3 由 `em.m` 得到的正确解和 EM 近似

$$W(\tau_j) - W(\tau_{j-1}) = W(jR\delta t) - W((j-1)R\delta t) = \sum_{k=jR-R+1}^{jR} dW_k.$$

给出. 在 `em.m` 中这个增量以 `Winc=sum(dW(R*(j-1)+1:R*j))` 呈现. $1 \times L$ 数组 `Xem` 储存了这个 EM 解, 这个解在图 3 中用带星号的红色虚线标出. 精确解和 EM 解在终点 $t = T$ 的误差, 计作 `emerr`, 结果是 0.6907. 取 $\Delta t = R\delta t$, 当 R 是更小的 2 和 1 时, 得到的终点误差分别为 0.1595 和 0.0821.

5. EM 方法的强弱收敛性

在 `em.m` 上面的例子中, 当 Δt 减少时, EM 解与真实解更加接近 —— 似乎可以收敛. 紧记 $X(\tau_n)$ 和 X_n 是随机变量, 为了使收敛的概念精确, 我们必须决定如何度量它们的差别. \mathbb{E} 表示期望值, 用 $\mathbb{E}|X_n - X(\tau_n)|$, 可以导出强收敛的概念. 如果存在一个常数 C , 使得

$$\mathbb{E}|X_n - X(\tau)| \leq C\Delta t^\gamma \quad (5.1)$$

对任意固定的 $\tau = n\Delta t \in [0, T]$, 和充分小的 Δt 成立, 则该方法被称作有阶为 γ 的强阶收敛. 如果 f 和 g 满足适当的条件, 可以证明 EM 有 $\gamma = 1/2$ 的强阶收敛. 注意到, 这区别于确定性的情形 —— 如果 $g \equiv 0$, 并且 X_0 是常数, 那么期望值可以从 (5.1) 的左端删去, 并且不等式在 $\gamma = 1$ 时成立.

在我们的数值试验中, 我们将把焦点集中在终点 $t = T$ 的误差上, 于是我们让

$$e_{\Delta t}^{\text{strong}} := \mathbb{E}|X_L - X(T)|, \quad \text{其中 } L\Delta t = T, \quad (5.2)$$

表示这种强意义下 EM 的终点误差. 如果 (5.1) 的界在 $\gamma = 1/2$ 时, 对 $[0, T]$ 上的任意固定点成立, 那么它当然在终点时成立, 于是对充分小的 Δt , 我们有

$$e_{\Delta t}^{\text{strong}} \leq C\Delta t^{\frac{1}{2}}. \quad (5.3)$$

表 6 M-file emstrong.m

```

%EMSTRONG Test strong convergence of Euler-Maruyama
%
% Solves      dX = lambda*X dt + mu*X dW,   X(0) = Xzero,
%             where lambda = 2, mu = 1 and Xzero = 1.
%
% Discretized Brownian path over [0,1] has dt = 2^(-9).
% E-M uses 5 different timesteps: 16dt, 8dt, 4dt, 2dt, dt.
% Examine strong convergence at T=1: E | X_L - X(T) |.

randn('state',100)
lambda = 2; mu = 1; Xzero = 1;      % problem parameters
T = 1; N = 2^9; dt = T/N;          %
M = 1000;                            % number of paths sampled

Xerr = zeros(M,5);                  % preallocate array
for s = 1:M,                          % sample over discrete Brownian paths
    dW = sqrt(dt)*randn(1,N);        % Brownian increments
    W = cumsum(dW);                  % discrete Brownian path
    Xtrue = Xzero*exp((lambda-0.5*mu^2)*dt+mu*W(end));
    for p = 1:5
        R = 2^(p-1); Dt = R*dt; L = N/R;      % L Euler steps of size Dt = R*dt
        Xtemp = Xzero;
        for j = 1:L
            Winc = sum(dW(R*(j-1)+1:R*j));
            Xtemp = Xtemp + Dt*lambda*Xtemp + mu*Xtemp*Winc;
        end
        Xerr(s,p) = abs(Xtemp - Xtrue);        % store the error at t = 1
    end
end

Dtvals = dt*(2.^([0:4]));
subplot(221)                          % top LH picture
loglog(Dtvals,mean(Xerr),'b*-'), hold on
loglog(Dtvals,(Dtvals.^(-.5)),'r--'), hold off % reference slope of 1/2
axis([1e-3 1e-1 1e-4 1])
xlabel('\Delta t'), ylabel('Sample average of | X(T) - X_L |')
title('emstrong.m','FontSize',10)

XXXX Least squares fit of error = C * Dt^q XXXX
A = [ones(5,1), log(Dtvals)']; rhs = log(mean(Xerr));
sol = A\rhs; q = sol(2)
resid = norm(A*sol - rhs)

```

表 6 的 M-file emstrong.m 中使用与 em.m 中相同的 λ , μ , 和 X_0 , 考察 SDE (4.5) 的 EM 的强收敛性. 我们计算 $[0, 1]$ 上, $\delta t = 2^{-9}$ 时, 1000 条不同的离散化的 Brown 路径. 对每一条路径, 用 5 个不同的步长 $\Delta t = 2^{p-1}\delta t$, $1 \leq p \leq 5$ 来应用 EM 方法. 第 s 个抽样路径使用的第 p 个步长的终点误差储存在 $Xerr(s, p)$ 中; 所以 $Xerr$ 是一个 1000×5 的数组. 函数 `mean` 用来求所有的抽样路径的均值: 求 `mean(Xerr)` 得到一个 1×5 的数组, $Xerr$ 的每一列都被它的列均值代替. 因此, `mean(Xerr)` 的第 p 个元素是一个对 $\Delta t = 2^{p-1}\delta t$ 时的 $e_{\Delta t}^{\text{strong}}$ 的近似.

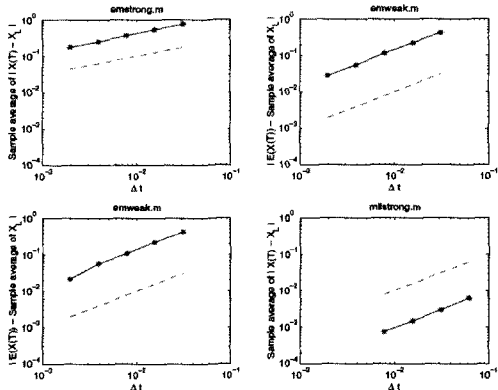


图 4 强的和弱的错误绘图: 红色虚线是每种情况下合适的参考斜率. 左上和右上是对 EM 的结果, 左下是对弱 EM 的结果, 右下是对 Milstein 的结果

了 $\log C$ 和 q 的最小平方拟合, 给出 q 值是 0.5384, 最小二次剩余是 0.0266. 因此, 我们的结果与 $1/2$ 阶的强收敛是一致的.

在 `emstrong.m` 中监测误差 $e_{\Delta t}^{\text{strong}}$ 时, 我们隐含地假设其他一些误差源是可忽略的, 它们包括以下的:

抽样误差: 误差来自通过抽样均值近似一个期望值.

随机数偏差: 随机数发生器固有的误差.

舍入误差: 浮点舍入误差.

对于一个典型的计算, 抽样误差可能是这 3 个误差中最大的. 在准备本文的程序时, 我们发现做某个试验需要样本的数量充分大, 并且对预计阶的收敛充分小的时间步长是可观察到的. (抽样误差像 $1/\sqrt{M}$ 一样衰减, 这里 M 是使用的抽样路径的数量.) [7] 中的一项研究显示, 当 Δt 减少时, 随机数发生器生成的样本缺乏独立性, 导致在舍入误差变得显著之前, 典型地退化计算.

尽管 (5.1) 中强收敛的定义涉及一个期望值, 它蕴含着单体模拟. Markov (马尔可夫) 不等式指出, 如果一个随机变量 X 有有限期望值, 那么对任意 $a > 0$, $|X| \geq a$ 的概率有 $(\mathbb{E}|X|)/a$ 的上界, 即

$$\mathbb{P}(|X| \geq a) \leq \frac{\mathbb{E}|X|}{a}.$$

因此, 取 $a = \Delta t^{1/4}$, 我们发现 EM 的 $\gamma = 1/2$ 阶强收敛的推论是

$$\mathbb{P}(|X_n - X(\tau)| \geq \Delta t^{1/4}) \leq C\Delta t^{1/4},$$

或者, 等价地,

$$\mathbb{P}(|X_n - X(\tau)| < \Delta t^{1/4}) \geq 1 - C\Delta t^{1/4}.$$

这表明 “在 $[0, T]$ 上一个固定点的误差很小” 的概率接近 1.

(5.1) 的强阶收敛度量了 “误差均值” 在 $\Delta t \rightarrow 0$ 时衰减的速度. 放松要求的话, 我们可以度量 “误差均值” 的衰减速度. 这导出了弱收敛的概念. 如果存在一个常数 C , 使得

如果不等式 (5.3) 的约等情形成立, 那么取对数:

$$\log e_{\Delta t}^{\text{strong}} \approx \log C + \frac{1}{2} \log \Delta t. \quad (5.4)$$

`loglog(Dtvals, mean(Xerr), 'b*-')`——`emstrong.m` 中的命令标出 \log - \log 尺度下 $e_{\Delta t}^{\text{strong}}$ 相对 Δt 的近似. 这产生了图 4 左上蓝色星号连接的实线. 作为参考, 添加了一条斜率是 $1/2$ 的红色虚线. 我们看到两条曲线的斜率看起来匹配得很好, 这表明 (5.4) 是正确的. 我们通过假定存在对某个常数 C 和 q 的一个幂关系 $e_{\Delta t}^{\text{strong}} = C\Delta t^q$ 对此进行进一步检验, 从而 $\log e_{\Delta t}^{\text{strong}} = \log C + q \log \Delta t$. 在 `emstrong` 结尾, 计算

表 7 *M-file* emweak.m

```

%EMWEAK Test weak convergence of Euler-Maruyama
%
% Solves    dX = lambda*X dt + mu*X dW,    X(0) = Xzero,
%           where lambda = 2, mu = 1 and Xzer0 = 1.
%
% E-M uses 5 different timesteps: 2^(p-10), p = 1,2,3,4,5.
% Examine weak convergence at T=1:    | E(X_L) - E(X(T)) |.
%
% Different paths are used for each E-M timestep.
% Code is vectorized over paths.
%
% Uncommenting the line indicated below gives the weak E-M method.

randn('state',100);
lambda = 2; mu = 0.1; Xzero = 1; T = 1; % problem parameters
M = 50000; % number of paths sampled

Xem = zeros(5,1); % preallocate arrays
for p = 1:5 % take various Euler timesteps
    Dt = 2^(p-10); L = T/Dt; % L Euler steps of size Dt
    Xtemp = Xzero*ones(M,1);
    for j = 1:L
        Winc = sqrt(Dt)*randn(M,1);
        % Winc = sqrt(Dt)*sign(randn(M,1)); %% use for weak E-M %%
        Xtemp = Xtemp + Dt*lambda*Xtemp + mu*Xtemp.*Winc;
    end
    Xem(p) = mean(Xtemp);
end
Xerr = abs(Xem - exp(lambda));

Dtvals = 2.^([1:5]-10);
subplot(222) % top RH picture
loglog(Dtvals,Xerr,'b--'), hold on
loglog(Dtvals,Dtvals,'r--'), hold off % reference slope of 1
axis([1e-3 1e-1 1e-4 1])
xlabel('\Delta t'), ylabel('| E(X(T)) - Sample average of X_L |')
title('emweak.m','FontSize',10)

%% Least squares fit of error = C * dt^q %%
A = [ones(p,1), log(Dtvals)']; rhs = log(Xerr);
sol = A\rhs; q = sol(2)
resid = norm(A*sol - rhs)

```

对某类函数 p , 对任意固定的 $\tau = n\Delta t \in [0, T]$ 和充分小的 Δt , 有

$$|\mathbb{E}p(X_n) - \mathbb{E}p(X(\tau))| \leq C\Delta t^\gamma \quad (5.5)$$

成立, 则称该方法有阶为 γ 的弱阶收敛. 通常, (5.5) 中所允许的函数 p 必须满足光滑性和多项式增长条件. 我们将把重点放在当 p 是恒等函数的情形. 对适当的 f 和 g , 可以看到 EM 有 $\gamma = 1$ 阶的弱阶收敛.

模拟我们的强收敛试验, 我们让

$$e_{\Delta t}^{\text{weak}} := |\mathbb{E}X_L - \mathbb{E}X(T)|, \quad \text{其中 } L\Delta t = T, \quad (5.6)$$

表示 EM 中弱的终点误差. 所以 (5.5) 对 $p(X) \equiv X$, $\gamma = 1$, 直接蕴含对充分小的 Δt ,

$$e_{\Delta t}^{\text{weak}} \leq C\Delta t. \quad (5.7)$$

我们检查表 7 中 M-file `emweak.m` 的 EM 弱收敛. 这里我们对 $[0, 1]$ 上, $\lambda = 2$, $\mu = 0.1$, $X_0 = 1$ 解 (4.5). 在 EM 中, 我们对 50000 条离散化的 Brown 路径进行抽样并且使用 5 个步长 $\Delta t = 2^{p-10}$, $1 \leq p \leq 5$. 这个代码与 `emstrong` 相比有一个额外的向量化水平——我们同时计算所有 50000 条路径. 这增加了额外的存储需要花费的执行时间. 作为补偿, 我们对每个 Δt 使用了不同的路径, 于是只需要存储当前的增量, 而不是整条路径. 为了额外的效率, 我们进一步选择路径增量 $\delta t = \Delta t$. 近似 $\mathbb{E}X_L$ 的抽样均值存储在 `Xem` 中. 由 (4.6) 得到, 对真实解, $\mathbb{E}X(T) = e^{\lambda T}$, 并且对每个 Δt , `Xerr` 存储了相应的弱终点误差. 图 4 中右上的图显示了弱误差在 log-log 尺度下随 Δt 的变化. 添加了一条斜率是 1 的红色虚线作为参考. 看起来 (5.7) 的约等式似乎成立. 如 `emstrong.m` 中一样, 我们做最小平方拟合, 得到 `q=0.9858`, 并且 `resid=0.0508`, 证明了 (5.7) 的约等式.

值得强调的是, 对于 `emweak.m` 中的计算, 我们对每一个步长 Δt 使用了不同的路径. 这是非常合理的. 弱收敛只关注解的均值, 所以对 (4.3) 中的每一步增量 $W(\tau_j) - W(\tau_{j-1})$, 我们可以自由地使用任何 $\sqrt{\Delta t}N(0, 1)$ 的样本. 事实上, 如果这个增量用一个独立的双点随机变量 $\sqrt{\Delta t}V_j$ 代替, 这里 V_j 以相同的概率取 $+1$ 和 -1 , 那么弱收敛的阶是保持的. (注意到, $\sqrt{\Delta t}V_j$ 与 $\sqrt{\Delta t}N(0, 1)$ 有相同的均值和方差). 在这种方式下用 $\sqrt{\Delta t}V_j$ 代替 Brown 增量, 导出弱 Euler-Maruyama 方法 (WEM), 这种方法有 $\gamma = 1$ 阶的弱收敛, 但是, 由于它没有用到按路径的信息, 所以达不到强收敛. WEM 背后的动机是, 从 V_j 中抽取样本的随机数发生器比从 $N(0, 1)$ 中抽取样本的更有效. 在 M-file `emweak.m` 中, 我们有注释行,

```
% Winc=sqrt(Dt)*sign(randn(M,1)); %% use for weak E-M %%
```

去掉首字符 %, 从而变成了非注释行, 执行 WEM, 因为 `sign(randn(M, 1))` 等概率地等于 $+1$ 或 -1 . (显然, 因为我们使用内置的正态随机数发生器, 在这种情形下, 没有提高效率.) 结果误差如图 4 的左下图所示. 最小平方拟合给出 `q=1.0671`, `resid=0.2096`.

6. Milstein 高阶方法

在前一节中我们看到, 在 (5.1) 中, EM 有 $\gamma = 1/2$ 阶的强收敛, 而基本的确定性 Euler 方法有经典的阶为 1 的收敛. 通过在随机增量中增加一个修正项, 可以将 EM 的强收敛的阶提高到 1, 这产生了 Milstein 方法. 修正项的产生源于传统的 Taylor (泰勒) 展开在 Itô 微积分的情形下必须做修改. 应用 Itô 的结果, 能生成一个所谓的 Itô-Taylor 展开, 这个展开是随机微积分中的一个基本工具. 在一个适当的点截取 Itô-Taylor 展开, 给出了 SDE (4.2) 的 Milstein 方法:

$$\begin{aligned} X_j &= X_{j-1} + \Delta t f(X_{j-1}) + g(X_{j-1})(W(\tau_j) - W(\tau_{j-1})) \\ &\quad + \frac{1}{2}g(X_{j-1})g'(X_{j-1})((W(\tau_j) - W(\tau_{j-1}))^2 - \Delta t), \quad j = 1, 2, \dots, L. \end{aligned} \quad (6.1)$$

表 8 的 M-file `milstrong.m` 中, 对 SDE 应用 Milstein 方法,

$$dX(t) = rX(t)(K - X(t))dt + \beta X(t)dW(t), \quad X(0) = X_0, \quad (6.2)$$

上式在人口动力学 [9] 中出现. 这里 r , K 和 β 是常数. 我们取 $r = 2$, $K = 1$, $\beta = 0.25$, $X_0 =$

表 8 *M-file milstrong.m*

```

%MILSTRONG Test strong convergence of Milstein: vectorized
%
% Solves  $dX = r*X*(K-X) dt + \text{beta}*X dW$ ,  $X(0) = X_{\text{zero}}$ ,
% where  $r = 2$ ,  $K = 1$ ,  $\text{beta} = 1$  and  $X_{\text{zero}} = 0.5$ .
%
% Discretized Brownian path over  $[0,1]$  has  $dt = 2^{-11}$ .
% Milstein uses timesteps  $128*dt$ ,  $64*dt$ ,  $32*dt$ ,  $16*dt$  (also  $dt$  for reference).
%
% Examines strong convergence at  $T=1$ :  $E | X_L - X(T) |$ .
% Code is vectorized: all paths computed simultaneously.

rand('state',100)
r = 2; K = 1; beta = 0.25; Xzero = 0.5; % problem parameters
T = 1; N = 2^(11); dt = T/N; %
M = 500; % number of paths sampled
R = [1; 16; 32; 64; 128]; % Milstein stepsizes are R*dt

dW = sqrt(dt)*randn(M,N); % Brownian increments
Xmil = zeros(M,5); % preallocate array
for p = 1:5
    Dt = R(p)*dt; L = N/R(p); % L timesteps of size Dt = R dt
    Xtemp = Xzero*ones(M,1);
    for j = 1:L
        Winc = sum(dW(:,R(p)*(j-1)+1:R(p)*j),2);
        Xtemp = Xtemp + Dt*r*Xtemp.*(K-Xtemp) + beta*Xtemp.*Winc ...
            + 0.5*beta^2*Xtemp.*(Winc.^2 - Dt);
    end
    Xmil(:,p) = Xtemp; % store Milstein solution at t =1
end

Xref = Xmil(:,1); % Reference solution
Xerr = abs(Xmil(:,2:5) - repmat(Xref,1,4)); % Error in each path
mean(Xerr); % Mean pathwise errors
Dtvals = dt*R(2:5); % Milstein timesteps used

subplot(224) % lower RH picture
loglog(Dtvals,mean(Xerr),'b*-'), hold on
loglog(Dtvals,Dtvals,'r--'), hold off % reference slope of 1
axis([1e-3 1e-1 1e-4 1])
xlabel('\Delta t')
ylabel('Sample average of | X(T) - X_L |')
title('milstrong.m','FontSize',10)

%%% Least squares fit of error = C * Dt^q %%%
A = [ones(4,1), log(Dtvals)]; rhs = log(mean(Xerr));
sol = A\rhs; q = sol(2)
resid = norm(A*sol - rhs)

```

0.5 (常数), 并且使用 $[0, 1]$ 上, $\delta t = 2^{-11}$ 的离散化的 Brown 路径. (6.2) 的解可以写为包含一个随机积分的闭形式表达式. 为简单起见, 我们让 $\Delta t = \delta t$ 的 Milstein 解作为精确解的一个好的近似, 然后将它与取 $\Delta t = 128\delta t$, $\Delta t = 64\delta t$, $\Delta t = 32\delta t$ 和 $\Delta t = 16\delta t$ 在 500 条样本路径上的 Milstein 近似解作比较. 相比 *emstrong.m*, 我们又增加了一个向量化

水平, 我们同时计算所有的路径, 而不是用一个 for 循环来涵盖样本路径. 我们设置 dW 为一个 $M \times N$ 数组, 在这个数组中, $dW(s, j)$ 是第 s 条路径的第 j 个增量. 对时间步长 $R(p)*dt$, Milstein 需要的增量是

$$Winc = \text{sum}(dW(:, R(p)*(j-1)+1:R(p)*j), 2);$$

这里取子数组包含 dW 的所有行和 $R(p)*(j-1)+1$ 列到 $R(p)*j$ 列, 然后对第二维 (列) 求和. 结果是一个 $M \times 1$ 数组, 它的第 j 个元素是在列 $R(p)*(j-1)+1$ 和列 $R(p)*j$ 之间的 dW 的 i 行中的元素的和. $M \times 5$ 数组 $Xmil$ 储存了 M 路径, 5 个步长的所有数值解. 结果的 log-log 误差如图 4 的右下图所示, 伴随着一条斜率为 1 的参考线. 最小平方拟合给出 $q=1.0184$ 和 $\text{resid}=0.0350$.

7. 线性稳定性

强收敛和弱收敛的概念关注一个有限区间 $[0, T]$ 上, 对小步长 Δt 的数值方法的精度. 但是, 在很多长时间 $t \rightarrow \infty$ 的应用中, 人们对 SDE 的行为感兴趣. 在这个场合, 形如 (5.1) 或 (5.5) 的收敛的界是没有价值的, 因为, 一般来说, 常数 C 随着 T 无界增长. 对于确定的 ODE 方法, 已经发展了一大块稳定性理论, 这些理论对固定的 $\Delta t, t_j \rightarrow \infty$ 的极限的数值方法的行为做了深入解析. 作为代表性的, 一个数值方法被应用于一类具有某些定性特征的问题, 并且这个数值方法重现这种特性的能力也被分析了. 虽然已经分析了一大类问题, 最简单的, 也可能是最有启发性, 是线性试验方程 $dX/dt = \lambda X$, 这里 $\lambda \in \mathbb{C}$ 是一个常数参数. 对于 SDE, 如我们现在说明的, 可以发展一个类似的线性稳定性理论.

我们回到线性 SDE (4.5), 参数 λ 和 μ 允许是复数. 在 $\mu = 0$ 并且 X_0 是常数的情形, (4.5) 简化为确定的线性试验方程, 它有形如 $X_0 \exp(\lambda t)$ 的解. 如果我们使用术语稳定的表示对任意的 $X_0, \lim_{t \rightarrow \infty} X(t) = 0$, 那么我们看到, 稳定性由 $\Re\{\lambda\} < 0$ 刻画. 为了把这个思想推广到 SDE 情形, 我们必须更加精确地描述 “ $\lim_{t \rightarrow \infty} X(t) = 0$ ” 的含义——随机变量是无穷维对象, 因此一般来说模是不等价的. 我们将考虑两个最常见的稳定性的度量方法: 均方方法和渐近方法. 假设 $X_0 \neq 0$ 的概率为 1, (4.5) 的解满足

$$\lim_{t \rightarrow \infty} \mathbb{E} X(t)^2 = 0 \Leftrightarrow \Re\{\lambda\} + \frac{1}{2}|\mu|^2 < 0, \quad (7.1)$$

$$\lim_{t \rightarrow \infty} |X(t)| = 0, \text{ 概率为 } 1 \text{ 地} \Leftrightarrow \Re\left\{\lambda - \frac{1}{2}\mu^2\right\} < 0. \quad (7.2)$$

(7.1) 的左端定义了均方稳定性的概念. (7.1) 的右端用参数 λ 和 μ 完全地刻画了这个性质. 类似地, (7.2) 定义和刻画了渐近稳定性. 设 $\mu = 0$, 这两种刻画衰退到相同的条件, 即出自确定的稳定性情形的 $\Re\{\lambda\} < 0$. 从 (7.1) 和 (7.2) 立刻得到, 如果 (4.5) 是均方稳定的, 那么它自动地渐近稳定, 但反之不成立. 因此, 关于这个试验方程问题, 均方稳定性是比渐近稳定性更迫切的需要. 两个稳定性定义在实践中都是有用处的.

现在假设选取参数 λ 和 μ , 使得 SDE (4.5) 在均方意义或渐近意义下是稳定的. 然后一个自然的问题是, “对什么范围的 Δt , EM 的解在类似的意义下是稳定的?” 这个问题在均方意义下是容易分析的. 期望值的简单性质表明, 对 (4.5) 应用 EM 方法, 有

表 9 *M-file stab.m*

```

%STAB Mean-square and asymptotic stability test for E-M
%
% SDE is dX = lambda*X dt + mu*X dW, X(0) = Xzero,
% where lambda and mu are constants and Xzero = 1.

randn('state',100)
T = 20; M = 50000; Xzero = 1;
ltype = {'b-', 'r--', 'm-.'}; % linetypes for plot

subplot(211) %%%%%%%%% Mean Square %%%%%%%%%
lambda = -3; mu = sqrt(3); % problem parameters
for k = 1:3
    Dt = 2^(1-k);
    N = T/Dt;
    Xms = zeros(1,N); Xtemp = Xzero*ones(M,1);
    for j = 1:N
        Winc = sqrt(Dt)*randn(M,1);
        Xtemp = Xtemp + Dt*lambda*Xtemp + mu*Xtemp.*Winc;
        Xms(j) = mean(Xtemp.^2); % mean-square estimate
    end
    semilogy([0:Dt:T], [Xzero,Xms], ltype{k}, 'Linewidth',2), hold on
end
legend('\Delta t = 1', '\Delta t = 1/2', '\Delta t = 1/4')
title('Mean-Square: \lambda = -3, \mu = \surd 3', 'FontSize',16)
ylabel('E[X^2]', 'FontSize',12), axis([0,T,1e-20,1e+20]), hold off

subplot(212) %%%% Asymptotic: a single path %%%%
T = 500;
lambda = 0.5; mu = sqrt(6); % problem parameters
for k = 1:3
    Dt = 2^(1-k);
    N = T/Dt;
    Xemabs = zeros(1,N); Xtemp = Xzero;
    for j = 1:N
        Winc = sqrt(Dt)*randn;
        Xtemp = Xtemp + Dt*lambda*Xtemp + mu*Xtemp*Winc;
        Xemabs(j) = abs(Xtemp);
    end
    semilogy([0:Dt:T], [Xzero,Xemabs], ltype{k}, 'Linewidth',2), hold on
end
legend('\Delta t = 1', '\Delta t = 1/2', '\Delta t = 1/4')
title('Single Path: \lambda = 1/2, \mu = \surd 6', 'FontSize',16)
ylabel('|X|', 'FontSize',12), axis([0,T,1e-50,1e+100]), hold off

```

$$\lim_{j \rightarrow \infty} \mathbb{E} X_j^2 = 0 \Leftrightarrow |1 + \Delta t \lambda|^2 + \Delta t |\mu|^2 < 1. \quad (7.3)$$

在渐近意义下, 这个问题可以借助强大数律和重对数律来研究, 可导出

$$\lim_{j \rightarrow \infty} |X_j| = 0, \text{ 概率为 } 1 \text{ 地} \Leftrightarrow \mathbb{E} \log |1 + \Delta t \lambda + \sqrt{\Delta t} \mu N(0, 1)| < 0. \quad (7.4)$$

这些结果由表 9 的 *M-file stab.m* 阐述. 为了测试均方稳定性, 我们对两个参数集解 $[0, 20]$ 上, $X_0 = 1$ (常数) 情形下的 (4.5). 第 1 个集有 $\lambda = -3, \mu = \sqrt{3}$. 这些值满足 (7.1),

因此该问题是均方稳定的. 我们对 50000 条离散化的 Brown 路径, 3 个不同的步长 $\Delta t = 1, 1/2, 1/4$, 应用 EM. 只有第 3 个步长值 $\Delta t = 1/4$ 满足 (7.3) 的右端. 图 5 的上面一个图标出了 X_j^2 关于 t_j 的抽样均值. 注意到, 垂直轴是对数尺度的. 在这个图中, $\Delta t = 1$ 和 $\Delta t = 1/2$ 的曲线是随 t 增长的, 而 $\Delta t = 1/4$ 曲线随 t 衰减到 0. 因此, 这个试验正确地蕴含了 EM 在均方意义下对 $\Delta t = 1, 1/2$ 和 $\Delta t = 1/4$ 分别

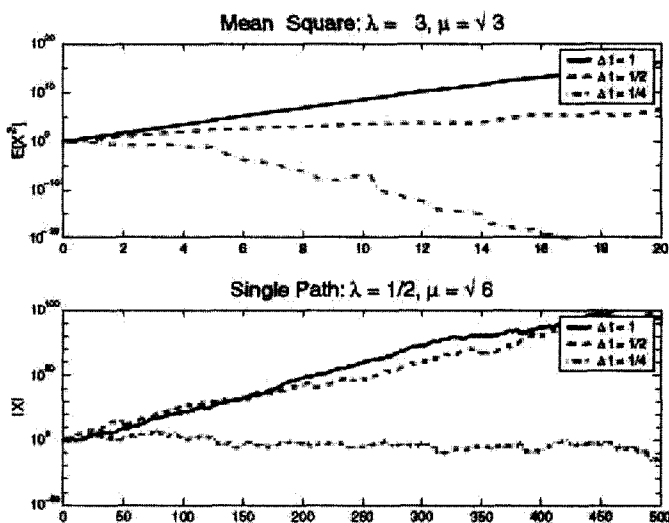


图 5 由 stab.m 得到的均方和渐近稳定性

是不稳定的和稳定的. 但是, 使用的样本数 (50000) 并不足以完全地解析这个行为; 3 条曲线应该是直线. 这突出表明了缺乏深入检查的简单化的抽样可能导致让人误解的结论.

为了检验渐近稳定性, 我们使用参数集 $\lambda = 1/2$ 和 $\mu = \sqrt{6}$. 从 (7.2) 可以得出, SDE 是渐近稳定的 (虽然, 从 (7.1) 得到, 它不是均方稳定的). 因为渐近稳定性关注概率为 1 的事件, 我们对一条单独的离散化的 Brown 路径, $\Delta t = 1, 1/2, 1/4$ 应用 EM, 并且由于计算一条单独的路径是没有价值的, 我们在 $[0, 500]$ 上积分. 可以证明, 只有这些时间步长中最小的, $\Delta t = 1/4$, 满足条件 (7.4)——这在图 6 中标出, 将在下面进行讨论. 图 5 下面的图标出了沿着路径, 关于 t_j 的 $|X_j|$. 我们看到, 只有 $\Delta t = 1/4$ 的解看起来衰减到 0, 和理论上一致.

图 6 提供了对这些计算的深入认识. 这里我们已经标出了对 $\lambda, \mu \in \mathbb{R}$ 的稳定性区域. x 轴和 y 轴分别表示 $\Delta t \lambda$ 和 $\Delta t \mu^2$. 在这个记法中, 从 (7.1) 得到, 对 $y < -2x$, SDE 是均方稳定的 (水平的紫红色阴影, 记作 SDE:ms), 对 $y > 2x$ 是渐近稳定的 (垂直的绿色阴影, 记作 SDE:as). EM 均方稳定性的条件 (7.3) 要求 y 是正的, 并且在抛物线 $-x(2+x)$ 的下方. 抛物线在图中用一条红色实曲线表示, 而相应的 EM 均方稳定

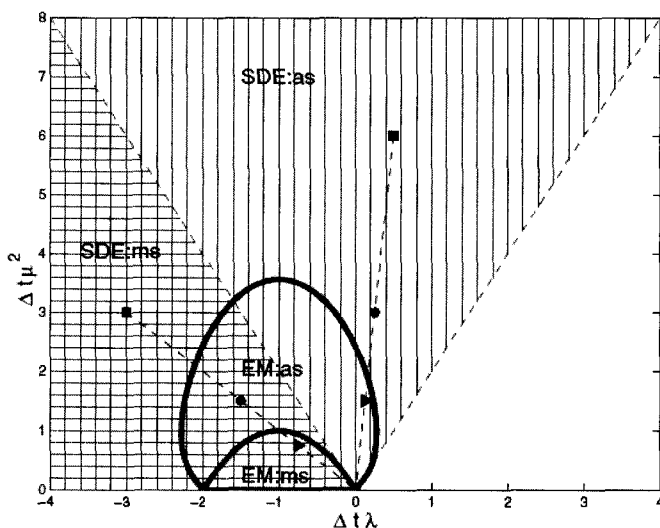


图 6 均方和渐近稳定区域

性区域记作 EM:ms. 由决定 EM 渐近稳定性的条件 (7.4) 导出用蓝色实线表示的拉平的蛋形边界.

结果区域记作 EM:as. 为了解释这个图, 注意到对给定的 λ 和 μ 的值, 点 $(x, y) = (\lambda, \mu^2)$ 对应时间步长 $\Delta t = 1$, 然后根据沿着连接 (λ, μ^2) 与原点的射线移动改变步长 Δt . 在 `stab.m` 中使用的参数集 $(\lambda, \mu^2) = (-3, 3)$ 和 $(\lambda, \mu^2) = (1/2, 6)$ 用红色的正方形表示. 我们看到第 1 个集落在 SDE:ms 中, 第 2 个集落在 SDE:as 中, 但是对 $\Delta t = 1$ 的 EM 都是不稳定的. 把 Δt 从 1 减少到 $1/2$, 得到图中蓝色圆圈标注的点. 现在第 1 个集在 EM:as 中 (但不在 EM:ms 中), 第 2 个集仍在 EM:as 外面. 进一步将 Δt 减少到 $1/4$, 得到图中绿色三角形标注的点. 我们看到现在第 1 个参数集在 EM:ms 中, 第 2 个参数集在 EM:as 中. 所以, 基本 SDE 的稳定性性质在两种情形中都得到重现.

8. 随机链法则

我们在 §3 中看到, 有不只一种方法来将积分的概念拓展到随机背景下. 在本节中, 我们简要提及随机微积分和确定微积分的另一个不同.

在确定性的情形中, 如果 $dX/dt = f(X)$, 那么对任意光滑函数 V , 链法则给出

$$\frac{dV(X(t))}{dt} = \frac{dV(X(t))}{dX} \frac{d(X(t))}{dt} = \frac{dV(X(t))}{dX} f(X(t)). \quad (8.1)$$

现在假设 X 满足 Itô SDE (4.2). (8.1) 对 $V(X)$ 的 SDE 类似是什么? 一个合理的猜测是 $dV = (dV/dX)dX$, 于是, 应用 (4.2),

$$dV(X(t)) = \frac{dV(X(t))}{dX} (f(X(t))dt + g(X(t))dW(t)). \quad (8.2)$$

但是, 应用 Itô 的结果的一个严格分析揭示产生了一个额外项,¹⁾ 并且正确的公式是

$$dV(X(t)) = \frac{dV(X(t))}{dX} dX + \frac{1}{2}g(X(t))^2 \frac{d^2V(X(t))}{dX^2} dt,$$

对上式应用 (4.2), 变成

$$dV(X(t)) = \left(f(X(t)) \frac{dV(X(t))}{dX} + \frac{1}{2}g(X(t))^2 \frac{d^2V(X(t))}{dX^2} \right) dt + g(X(t)) \frac{dV(X(t))}{dX} dW(t). \quad (8.3)$$

我们不试图证明或者仅仅是验证 (8.3), 而是要做一个数值试验.

我们考虑 SDE

$$dX(t) = (\alpha - X(t))dt + \beta\sqrt{X(t)}dW(t), \quad X(0) = X_0, \quad (8.4)$$

这里 α 和 β 是正常数参数. 这个 SDE 是一个反方差过程, 它可以作为资产价值模型 [8, 第 9 章]. 可以证明, 如果 $X(0) > 0$ 的概率为 1, 那么对所有的 $t > 0$, 这个正号都成立. 令 $V(X) = \sqrt{X}$, 应用 (8.3) 得到,

$$dV(t) = \left(\frac{4\alpha - \beta^2}{8V(t)} - \frac{1}{2}V(t) \right) dt + \frac{1}{2}\beta dW(t). \quad (8.5)$$

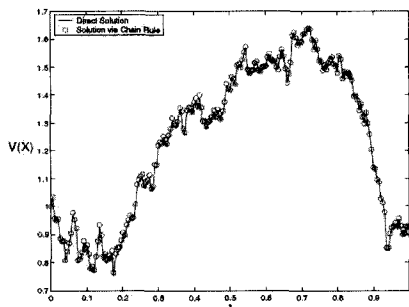


图 7 直接用 (8.4) 得到的 $V(X(t)) = \sqrt{X(t)}$ 的 EM 逼近和用由 `chain.m` 得到的链式法则 (8.5)

1) 事实上, (8.2) 在 Stratonovich 框架下结果是正确的, 但我们在这里使用 Itô 微积分.——原注

表 10 *M-file chain.m*

```

%CHAIN Test stochastic Chain Rule
%
% Solve SDE for  $V(X) = \sqrt{X}$  where  $X$  solves
%  $dX = (\alpha - X) dt + \beta \sqrt{X} dW$ ,  $X(0) = X_{\text{zero}}$ ,
% with  $\alpha = 2$ ,  $\beta = 1$  and  $X_{\text{zero}} = 1$ .
%  $X_{\text{em1}}$  is Euler-Maruyama solution for  $X$ .
%  $X_{\text{em2}}$  is Euler-Maruyama solution of SDE for  $V$  from Chain Rule.
% Hence, we compare  $\sqrt{X_{\text{em1}}}$  and  $X_{\text{em2}}$ .
% Note: abs is used for safety inside sqrt, but has no effect in this case.

randn('state',100)
alpha = 2; beta = 1; T = 1; N = 200; dt = T/N; % Problem parameters
Xzero = 1; Xzero2 = 1/sqrt(Xzero); %

Dt = dt; % EM steps of size Dt = dt
Xem1 = zeros(1,N); Xem2 = zeros(1,N); % preallocate for efficiency
Xtemp1 = Xzero; Xtemp2 = Xzero2;
for j = 1:N
    Winc = sqrt(dt)*randn;
    f1 = (alpha-Xtemp1);
    g1 = beta*sqrt(abs(Xtemp1));
    Xtemp1 = Xtemp1 + Dt*f1 + Winc*g1;
    Xem1(j) = Xtemp1;
    f2 = (4*alpha-beta^2)/(8*Xtemp2) - Xtemp2/2;
    g2 = beta/2;
    Xtemp2 = Xtemp2 + Dt*f2 + Winc*g2;
    Xem2(j) = Xtemp2;
end

plot([0:Dt:T],[sqrt([Xzero,abs(Xem1)])], 'b-', [0:Dt:T],[Xzero,Xem2], 'ro')
legend('Direct Solution','Solution via Chain Rule',2)
xlabel('t','FontSize',12)
ylabel('V(X)','FontSize',16,'Rotation',0,'HorizontalAlignment','right')

Xdiff = norm(sqrt(Xem1) - Xem2,'inf')

```

在表 10 的 *M-file chain.m* 中，我们对 $\alpha = 2$, $\beta = 1$, 并且 $X_0 = 1$ (常数) 时的 SDE (8.4) 应用 EM. 我们令 $\Delta t = \delta t$, 计算一条 $[0, 1]$ 上, $\delta t = 1/200$ 的离散的 Brown 路径. 数值解的平方根在图 7 中用一条蓝色实线标出. 我们还对 V 解 SDE (8.5), 并且用一个红色的符号 “o” 表示这个解. 我们观察到, 两个解有很好的 consistency. 这也可以用最大偏差值证实: $X_{\text{diff}}=0.0151$.

9. 进一步的问题

本文设想用最少的技术细节, 给读者一个关于 SDE 的数值模拟的简要实用的介绍. 于是我们抹去了大量的理论, 省去了很多重要的主题. 查阅 §1 末尾的参考文献是填补这些缺口的好办法. 在这最后一节中, 我们会点到一些被略去的主题.

首先, 我们没有讨论对 (4.2) 中的 f 和 g 加上什么条件可以保证 SDE 有一个唯一解. 典型的存在和唯一性定理 (和数值方法的收敛定理) 比它们在确定性理论中的相应结果有更严格的限制. 第 2, 我们没有提及 SDE 和某些依赖时间的 (下转 308 页)

限宽度会引出某种不确定性. 对于这些数据, 所恢复的长度 L_0 与原来的相对误差有 18%; 其他的长度和质量则要更准确一些.

10. 进一步的探索

我们鼓励读者们利用我们在网站

<http://www.caam.rice.edu/~beads>

上提供的数据进行前瞻性的和反问题的探索, 其中这些数据包括了 2, 4, 6 颗珠子系统的时间序列数据和峰值位置.

你们在自己的实验中将会看到, 我们所描述的反过程有着许多的误差. 首先, 我们的度量带进了不准确: 我们测量长度准确到 $\pm 2\text{mm}$, 张力到 $\pm 10^4\text{dyn}$ (达因), 质量到 $\pm 0.01\text{g}$, 而频率到 $\pm 0.7\text{sec}^{-1}$. 我们的数学模型只是真实系统的一个近似; 我们既没有考虑进非线性影响以及最终使弦停止振动的阻尼. 弦本身既非完全柔韧的也非无质量的, 正如模型所假定的, 珠子也非质点. (图 4 提示了珠子的宽度对于本征值可能的影响.) 这个系统渐近地描述了小的振动, 但是我们的振幅必须足够大到能被光电探测器检测到. 知道了如此多的这些误差, 你可能会对在 §9 描述的实验中取得的准确程度感到惊讶吧!

还有一个误差源对我们的考虑也是有好处的, 特别当我们考虑将这个实验用于有许多珠子的弦时. 在 §5 详述的那个数值算法, 在进行浮点计算机算术时会招致舍入误差. 粗略地说, 当 $n > 50$ 时, 图 5 中的算法在 MATLAB 默认的双精度算术中会出现大的误差. (人们应该适当地定出单位, 这是因为如像 $\tau = 10^7\text{dyn}$ 在 n 变大时会导致溢流.) 读者可以通过利用 §2 的矩阵 \mathbf{M} 和 \mathbf{K} (在 MATLAB 中使用 `eig(K, M)`) 来计算假设是对称弦的本征值的办法来探索这个不稳定性, 然后将这些“准确的”数据提供给反算法. 你能恢复你开始时所用的长度和质量准确到什么程度? 当 n 变得非常大时这个精度是怎样依赖于珠子的个数的? 有兴趣的读者可研究另一种算法, 如像连分数法 (见扩充了的文稿 [7]) 和 Lanczos 方法 (见 [6], §4.2).

致谢, 参考文献 (略)

(胥鸣伟 译 姚景齐 校)

(上接 335 页)

偏微分方程之间的联系. 第 3, 我们没有注意标量问题——当应用到一个 SDE 系统中时, EM 方法看起来是一样的, 但是 Milstein 方法变得更加复杂. 对 SDE 的数值方法的研究在很多方向上是很活跃的, 包括高阶强弱收敛的构造方法或者提高稳定性, 时间步长变量算法的设计和长时间性质的分析如非线性问题的遍历. 最近的文献可以在 [11] 中找到.

致谢, 参考文献 (略)

(李灵芝 译 李艳芳 校)