

结构可靠性分析的自适应子集模拟方法

陈向前*, 董 聪, 闫 阳

(清华大学 土木工程系, 北京 100084)

摘 要:结构可靠性分析需要精确高效的失效概率计算方法。为解决高维非线性可靠性分析问题中的失效概率计算问题,本文提出了两种以失效概率估计精度为停机控制参数的自适应子集模拟方法。理论分析和数值算例表明:(1)两种自适应子集模拟方法能根据失效概率的估计精度要求自适应调整样本量;(2)考虑样本量优化的自适应子集模拟方法能进一步减少总样本量,提高计算效率。本文所提方法为研究者对结构进行精确高效的可靠性分析提供了一条可行途径。

关键词:结构可靠性;失效概率;子集模拟方法;自适应;样本量优化

中图分类号:TB114.3;O213.2 **文献标志码:**A **doi:**10.7511/jslx201305006

1 引言

近年来,伴随着经济建设的快速发展,大跨度桥梁、海洋平台及输电铁塔等大型复杂工程结构大量兴建,重大工程结构的安全性与耐久性问题受到工程与学术界广泛关注^[1]。结构可靠性理论考虑了结构参数的不确定性,可指导工程技术人员和研究者对结构进行更合理地分析、设计与安全评估。失效概率的计算作为结构可靠性理论的核心问题之一,受到了研究者高度重视^[1-14]。

随着计算机硬件水平的不断提升和结构分析技术的成熟,采用数值模拟方法估计结构失效概率渐成主要趋势。大型复杂工程结构通常包括众多随机变量,高维随机空间中的可靠度计算方法是近十年来发展起来的一个重要研究课题^[2-13]。针对上述问题,研究者发展了一系列失效概率计算的随机模拟方法,如子集模拟方法 SS(Subset Simulation)^[2-7]、线抽样方法 LS(Line Sampling)^[8-11]、球子集模拟方法 S³(Spherical Subset Simulation)^[12]及赛马算法 HSA(Horseracing Simulation Algorithm)^[13]等。其中,子集模拟方法具有通用性强、效率高的优点,是目前应用较广泛的一类随机模拟方法。

对于结构失效概率计算的数值模拟方法而言,

精度和效率是算法设计时需考虑的关键因素,在给定的工程精度范围内尽可能提高计算效率一直是该领域研究者努力的方向。子集模拟方法存在以下不足:不能根据要求的计算精度自适应调整样本量。针对子集模拟方法,本文提出了两种根据计算精度自适应调整样本量的方法。最后,利用数值算例验证了上述方法的可行性。

2 子集模拟方法

结构失效概率可表示为

$$P_F = \int_F f(\boldsymbol{\theta}) d(\boldsymbol{\theta}) = \int I_F(\boldsymbol{\theta}) f(\boldsymbol{\theta}) d(\boldsymbol{\theta}) \quad (1)$$

式中 $\boldsymbol{\theta} \in \mathbf{R}^d$ 为 d 维独立标准正态随机矢量,服从概率密度 $f(\boldsymbol{\theta})$; $F = \{\boldsymbol{\theta} | g(\boldsymbol{\theta}) < 0\}$ 为失效域, $g(\boldsymbol{\theta})$ 为功能函数, $I_F(\boldsymbol{\theta})$ 为示性函数;当 $\boldsymbol{\theta} \in F$ 时, $I_F(\boldsymbol{\theta}) = 1$, 否则, $I_F(\boldsymbol{\theta}) = 0$ 。

采用直接蒙特卡罗模拟 DMCS(Direct Monte Carlo Simulation)估计 P_F 时,

$$P_F \approx \hat{P}_F = \frac{1}{N} \sum_{i=1}^N I_F(\boldsymbol{\theta}^{(i)}) \quad (2)$$

式中 $\boldsymbol{\theta}^{(1)}, \boldsymbol{\theta}^{(2)}, \dots, \boldsymbol{\theta}^{(N)}$ 为按概率密度函数 $f(\boldsymbol{\theta})$ 抽取的 N 个独立样本。 \hat{P}_F 的精度可由变异系数 c. o. v. (coefficient of variation) 估计得

$$\delta(\hat{P}_F) = \sqrt{(1 - P_F)/(NP_F)} \approx \sqrt{(1 - \hat{P}_F)/(N\hat{P}_F)} \quad (3)$$

采用 DMCS 估计 P_F 的优点在于,失效概率的估计精度只与 P_F 值和抽取的样本量 N 有关,与问题维数 d 和功能函数的非线性程度无关;缺点在于,结构失效问题是小失效概率问题, P_F 的量级通常在 $10^{-6} \sim 10^{-3}$ 之间,为达到变异系数为 0.1 的

收稿日期:2012-06-11;修改稿收到日期:2012-09-15。
基金项目:国家电网公司科技项目(GC71-12-001)资助。
作者简介:陈向前*(1985-),男,博士生
(E-mail: cxq04@mails.tsinghua.edu.cn);
董 聪(1964-),男,博士,教授,博士生导师;
闫 阳(1985-),男,硕士生。

目标,需要的样本量约为 $100/P_F$, 计算效率不高。

为解决高维非线性随机系统小失效概率计算问题, Au 等^[2-4] 提出了子集模拟方法。该算法的基本策略是, 将失效概率表示为一系列中间失效概率的乘积, 采用马尔科夫链蒙特卡罗方法 MCMC (Markov Chain Monte Carlo) 高效地生成中间失效域样本以估计中间失效概率。

首先, 将失效域表示为一系列中间失效域的子集:

$$F = F_m \subset F_{m-1} \subset \dots \subset F_1 \subset F_0 = \mathbf{R}^d \quad (4)$$

则失效概率 P_F 可表示为

$$P_F = \prod_{i=1}^m P(F_i | F_{i-1}) = \prod_{i=1}^m P_i \quad (5)$$

适当选取中间失效域, 可使中间失效概率 P_i 较大(例如达到 0.1), 采用少量样本即可使中间失效概率达到较高估计精度。 F_0 内样本采用 DMCS 生成, 为提高计算效率, 中间失效域 $F_i (i=1, 2, \dots, m-1)$ 内的样本采用 MM 算法 (Modified Metropolis algorithm) 或 MMH 算法 (Modified Metropolis-Hastings algorithm) 生成。

MMH 算法属于 MCMC 模拟算法。Metropolis 等首先提出了 Metropolis 算法, Hastings 等将建议分布函数推广到更一般形式, Au 等对随机向量进行了分组, 采用各组分别生成的方式提高了高维随机空间中样本被接受的概率, 形成了 MM 算法和 MMH 算法。

本文简略介绍 MMH 算法。将 d 维随机矢量 $\boldsymbol{\theta}$ 分为 n_G 组, $\boldsymbol{\theta} = [\boldsymbol{\theta}^{(1)}, \boldsymbol{\theta}^{(2)}, \dots, \boldsymbol{\theta}^{(n_G)}]$ 。设 $\boldsymbol{\theta}_k$ 为当前状态点, $\boldsymbol{\theta}_k$ 服从条件分布 $q(\boldsymbol{\theta}) = \pi(\boldsymbol{\theta} | F) = \prod_{i=1}^{n_G} q(\boldsymbol{\theta}^{(i)})$, $h_i(\boldsymbol{\xi}^{(i)} | \boldsymbol{\theta}^{(i)})$ 为建议概率密度函数 (proposal PDF, 简称建议分布)。采用 MMH 算法可从当前状态点 $\boldsymbol{\theta}_k$ 出发生成下一个状态点 $\boldsymbol{\theta}_{k+1}$, 使 $\boldsymbol{\theta}_{k+1} \sim q(\boldsymbol{\theta})$ 。MMH 算法可描述为^[3]

① 生成候选状态点

$$\bar{\boldsymbol{\theta}}_{k+1} = [\bar{\boldsymbol{\theta}}_{k+1}^{(1)}, \bar{\boldsymbol{\theta}}_{k+1}^{(2)}, \dots, \bar{\boldsymbol{\theta}}_{k+1}^{(n_G)}]$$

对 $i=1, 2, \dots, n_G$:

a) 利用建议分布 $h_i(\cdot | \boldsymbol{\theta}_k^{(i)})$ 生成 $\boldsymbol{\xi}_{k+1}^{(i)}$ 。

b) 计算接收率

$$r_{k+1}^{(i)} = [q_i(\boldsymbol{\xi}_{k+1}^{(i)}) h_i(\boldsymbol{\theta}_k^{(i)} | \boldsymbol{\xi}_{k+1}^{(i)})] / [q_i(\boldsymbol{\theta}_k^{(i)}) h_i(\boldsymbol{\xi}_{k+1}^{(i)} | \boldsymbol{\theta}_k^{(i)})] \quad (6)$$

c) 以概率 $\min\{1, r_{k+1}^{(i)}\}$ 取 $\bar{\boldsymbol{\theta}}_{k+1}^{(i)} = \boldsymbol{\xi}_{k+1}^{(i)}$, 以概率 $1 - \min\{1, r_{k+1}^{(i)}\}$ 取 $\bar{\boldsymbol{\theta}}_{k+1}^{(i)} = \boldsymbol{\theta}_k^{(i)}$ 。

② 判断候选状态点是否属于中间失效域

取 $\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k$ 。若 $\bar{\boldsymbol{\theta}}_{k+1} = \boldsymbol{\theta}_k$, 结束; 否则, 若 $\bar{\boldsymbol{\theta}}_{k+1} \in F$, 取 $\boldsymbol{\theta}_{k+1} = \bar{\boldsymbol{\theta}}_{k+1}$ 。

对任意 $i=1, 2, \dots, n_G$, 若都满足 $h_i(\boldsymbol{\theta}^{(i)} | \boldsymbol{\xi}^{(i)}) = h_i(\boldsymbol{\xi}^{(i)} | \boldsymbol{\theta}^{(i)})$, 则 MMH 算法退化为 MM 算法。

不失一般性, 取中间失效域 $F_i = \{\boldsymbol{\theta} | g(\boldsymbol{\theta}) < b_i\}$, $b_1 > b_2 > \dots > b_m = b = 0$, 则子集模拟方法可描述为^[2,3,7]

① 选择各层抽样数 N , 中间失效概率 p_0 , 原概率密度 $f(\boldsymbol{\theta})$, 建议分布 $h_{i0}(\boldsymbol{\xi}^{(i0)} | \boldsymbol{\theta}^{(i0)})$, $n_c = N p_0$, $n_s = 1/p_0$ 。

② 置 $i=0$, $n_F(i) = 0$, 利用 DMCS 从原分布 $f(\boldsymbol{\theta})$ 抽取 N 个样本 $\boldsymbol{\theta}_1^{(i)}, \boldsymbol{\theta}_2^{(i)}, \dots, \boldsymbol{\theta}_N^{(i)}$ 。

③ 统计 $g(\boldsymbol{\theta}_j^{(i)}) (j=1, 2, \dots, N)$ 中满足 $g(\boldsymbol{\theta}_j^{(i)}) < b=0$ 的个数, 记为 $n_F(i)$ 。

④ 若 $n_F(i) < n_c$, 执行⑤~⑦; 否则, 执行⑧。

⑤ $i = i + 1$, 将 $\boldsymbol{\theta}_j^{(i-1)} (j=1, 2, \dots, N)$ 按 $g(\boldsymbol{\theta})$ 升序排序并重编号为 $\boldsymbol{\theta}_j^{(i-1)} (j=1, 2, \dots, N)$ 。

⑥ 确定条件边界 $b_i = [g(\boldsymbol{\theta}_{n_c}^{(i-1)}) + g(\boldsymbol{\theta}_{n_c+1}^{(i-1)})] / 2$, 对 $j=1, 2, \dots, n_c$, 执行⑦。

⑦ 令 $\boldsymbol{\theta}_{(j-1) \times n_s + 1}^{(i)} = \boldsymbol{\theta}_j^{(i-1)} \sim f(\cdot | F_i)$, 对 $k=2, 3, \dots, n_s$, 从 $\boldsymbol{\theta}_{(j-1) \times n_s + k-1}^{(i)}$ 出发, 用 MMH 算法生成 $\boldsymbol{\theta}_{(j-1) \times n_s + k}^{(i)} \sim f(\cdot | F_i)$, 返回③。

⑧ $P_F = \hat{P}_F \approx \hat{p}_0 (n_F(i) / N)$ 。

3 自适应子集模拟方法

3.1 样本量与估计精度的关系

对于 SS, 第 i 层 ($i=1, 2, \dots, m$) 中间失效概率估计值 \hat{P}_i 对应的 c. o. v. 为^[2]

$$\delta_i = \sqrt{[(1 - P_i) / P_i N] (1 + \gamma_i)} \quad (7)$$

式中

$$\gamma_i = 2 \sum_{j=1}^{n_s-1} (1 - j/n_s) [R_i(j) / R_i(0)] \quad (8)$$

$$R_i(j) = E[I_{F_i}(\boldsymbol{\theta}_k^{(i-1)}) I_{F_i}(\boldsymbol{\theta}_{k+j}^{(i-1)})] - P_i^2 \quad (9)$$

可将式(7)中 $N/(1 + \gamma_i)$ 理解为有效样本数, 则式(7)与式(3)一致。由式(8)和式(9)可知, γ_i 反映了样本间的相关性, 样本间的相关性越大, γ_i 越大, 算法计算效率越低。若忽略从不同初始状态出发的马氏链状态间的相关性, 则式(9)可通过式(10)估计得

$$R_i(j) \approx \hat{R}_i(j) = \frac{1}{N - j n_c} \sum_{k=1}^{n_c} \sum_{l=1}^{n_s-j} I_{F_i}(\boldsymbol{\theta}_{(k-1)n_s+l}^{(i-1)}) \cdot I_{F_i}(\boldsymbol{\theta}_{(k-1)n_s+l+j}^{(i-1)}) - \hat{P}_i^2 \quad (10)$$

$\hat{P}_i (i=1, 2, \dots, m)$ 通常并不独立, 但研究表明, \hat{P}_i 对应的 c. o. v. 可由式(11)较好地近似为^[2]

$$\delta = \sqrt{\sum_{i=1}^m \delta_i^2} \quad (11)$$

对于失效概率计算的数值模拟算法而言,建立与失效概率估计精度对应的停机准则十分必要。这样的停机准则一旦建立,研究者就可根据对失效概率的估计精度要求确定何时停机,最大限度减少结构分析次数,提高计算效率。 \hat{P}_F 对应的变异系数 δ 反映了 \hat{P}_F 的估计精度,因此以该参数作为停止迭代控制参数更具合理性。对于 SS 而言,预先确定每层样本量 N ,经一次分析可根据式(7~11)估计 \hat{P}_F 对应的 δ ,该 δ 可能大于要求的变异系数 δ_i ,因此需增加样本以提高 \hat{P}_F 估计精度。

3.2 样本量调整方法

本文以 \hat{P}_F 对应的变异系数 δ 作为控制参数,建立停机准则。SS 采用由中间失效概率确定中间失效域边界的策略,规避了预先确定失效域边界的难点,但同时使算法结束后继续增加样本变得困难。本文建议运行完 SS 后,固定得到的中间失效边界 b_i ,继续增加样本时,更新中间失效概率 \hat{P}_i 和 δ_i ,直至按式(11)计算的 δ 不大于目标变异系数 δ_i 。

本文提出两种以变异系数 δ 建立停机准则的自适应子集模拟方法 ASS(Adaptive Subset Simulation),第 1 种方法更直接简明,第 2 种方法进一步考虑了样本量的优化分配,比第 1 种方法更高效。分别记两种方法为 ASS1 和 ASS2。

3.2.1 ASS1

选定 N 和 p_0 ,运行一次 SS,可得到中间失效域数 m (不失一般性假设 $m \geq 2$;否则,采用 DMCS 即可)。记第 i 层样本数分别为 $N_i (i = 0, 1, 2, \dots, m-1)$,第 i 层样本中落入中间失效域 $F_{i+1} = \{\theta | g(\theta) < b_{i+1}\}$ 的样本数为 $n_{F(i+1)}$,由第 i 层域 ($i = 0, 1, 2, \dots, m-1$) 增加样本而新落入 F_{i+1} 的样本量为 $n_{0c}^{(i+1)}$ 。定义一新变量为

$$T_{i+1}(j) = \sum_{k=1}^{n_c^{(i)}} \sum_{l=1}^{n_s-j} I_{F_{i+1}}(\theta_{(k-1)n_s+l}^{(i)}) I_{F_{i+1}}(\theta_{(k-1)n_s+l+j}^{(i)}) \quad (12)$$

式中 $n_c^{(i)} (i = 0, 1, 2, \dots, m-1)$ 为第 i 层马氏链数目, $j = 1, 2, \dots, n_s - 1$ 。

ASS1 算法可描述:

① 输入 N, p_0, N_{0i} ,原概率密度 $f(\theta)$,建议分布 $h_{i0}(\xi^{(i0)} | \theta^{(i0)}), \delta_i$ 。

② 执行一次 SS,输出 $\hat{P}_F, m, \delta, b_{i+1}, N_i, n_{F(i+1)}, T_{i+1}(j)$,其中 $i = 0, 1, 2, \dots, m-1$ 。

③ 若 $\delta \leq \delta_i$,则精度足够,执行⑨;否则,执行④~⑧。

④ 对于给定的第 0 层样本增加数 N_{0i} ,采用 DMCS 从原概率密度 $f(\theta)$ 生成 N_{0i} 个样本,记新

落入 F_i 的样本数为 $n_{0c}^{(1)}$,对应的样本记为 $\theta_1^{(0)}, \theta_2^{(0)}, \dots, \theta_{n_{0c}^{(1)}}^{(0)}$,更新 \hat{P}_i 和 δ_i 。

$$N_0 = N_0 + N_{0i}, n_{F(1)} = n_{F(1)} + n_{0c}^{(1)} \quad (13,14)$$

$$\hat{P}_1 = n_{F(1)}/N_0, \delta_1 = \sqrt{(1 - \hat{P}_1)/\hat{P}_1 N_0} \quad (15,16)$$

⑤ 对 $i = 1, 2, \dots, m-1$,若 $n_{0c}^{(i)} > 0$,执行⑥和⑦;否则,转至⑧。

⑥ 对 $j = 1, 2, \dots, n_{0c}^{(i)}$,令 $\theta_{(j-1) \times n_s + 1}^{(i)} = \theta_j^{(i-1)} \sim f(\cdot | F_i)$,对 $k = 2, 3, \dots, n_s$,从 $\theta_{(j-1) \times n_s + k-1}^{(i)}$ 出发,用 MMH 算法生成 $\theta_{(j-1) \times n_s + k}^{(i)} \sim f(\cdot | F_i)$ 。

⑦ 对 $\theta_j^{(i)} (j = 1, 2, \dots, n_{0c}^{(i)} n_s)$ 按 $g(\theta)$ 升序排序为 $\theta_j^{(i)} (j = 1, 2, \dots, n_{0c}^{(i)} n_s)$,统计 $\theta_j^{(i)}$ 中满足 $g(\theta_j^{(i)}) < b_{i+1}$ 的个数,记为 $n_{0c}^{(i+1)}$,更新 \hat{P}_i 和 δ_i 。

$$N_i = N_i + n_{0c}^{(i)} n_s \quad (17)$$

$$n_{F(i+1)} = n_{F(i+1)} + n_{0c}^{(i+1)} \quad (18)$$

$$\hat{P}_{i+1} = n_{F(i+1)}/N_i \quad (19)$$

$$T_{i+1}(j) = T_{i+1}(j) + \sum_{k=1}^{n_{0c}^{(i)}} \sum_{l=1}^{n_s-j} I_{F_{i+1}}(\theta_{(k-1)n_s+l}^{(i)}) \cdot I_{F_{i+1}}(\theta_{(k-1)n_s+l+j}^{(i)}) \quad (20)$$

$$R_{i+1}(j) \approx \hat{R}_{i+1}(j) = [1/(N_i - j p_0 N_i)] T_{i+1}(j) - \hat{P}_{i+1}^2 \quad (21)$$

$$\gamma_{i+1} = 2 \sum_{j=1}^{n_s-1} (1 - j/n_s) [\hat{R}_{i+1}(j) / \hat{R}_{i+1}(0)] \quad (22)$$

$$\delta_{i+1} = \sqrt{(1 - \hat{P}_{i+1} / \hat{P}_{i+1} N_i) (1 + \gamma_{i+1})} \quad (23)$$

⑧ 按式(11)更新 δ ,返回③。

⑨ 精度足够,迭代终止,计算

$$\hat{P}_F = \prod_{i=1}^m \hat{P}_i \quad (24)$$

从式(20~23)可以看出,利用新变量 $T_i(j)$ 后, δ_i 的计算变得更为快捷,无需保存上一次迭代过程中的样本和功能函数值。

3.2.2 ASS2

将式(7)变换为如下一般形式

$$N_{i-1} = [(1 - P_i) / P_i \delta_i^2] (1 + \gamma_i) \quad (25)$$

式中 $i = 1, 2, \dots, m$ 。引入记号

$$a_i = [(1 - P_i) / P_i] (1 + \gamma_i), x_i = \delta_i^2 \quad (26,27)$$

则总样本量 N_{total} 可表示为

$$N_{total} = \sum_{i=1}^m N_{i-1} = \sum_{i=1}^m a_i / x_i \quad (28)$$

考虑给定 δ 时使总样本量 N_{total} 最小,则对应的优化问题可表示为

$$\begin{aligned} \min N_{total} &= \sum_{i=1}^m a_i / x_i \\ \text{s. t. } \sum_{i=1}^m x_i &= \delta^2 \end{aligned} \quad (29)$$

由柯西不等式得

$$\left(\sum_{i=1}^m \frac{a_i}{x_i}\right)\left(\sum_{i=1}^m x_i\right) = \left[\sum_{i=1}^m (\sqrt{a_i/x_i})^2\right]\left[\sum_{i=1}^m (\sqrt{x_i})^2\right] \geq \left[\sum_{i=1}^m \sqrt{a_i/x_i} \sqrt{x_i}\right]^2 = \left[\sum_{i=1}^m \sqrt{a_i}\right]^2 \quad (30)$$

结合式(29)得

$$N_{\text{total}} \geq \left(\sum_{i=1}^m \sqrt{a_i}\right)^2 \frac{1}{\delta^2} \quad (31)$$

式中,等号当且仅当

$$x_1/\sqrt{a_1} = x_2/\sqrt{a_2} = \dots = x_m/\sqrt{a_m} \quad (32)$$

时成立,此时有

$$x_i = (\sqrt{a_i} / \sum_{j=1}^m \sqrt{a_j}) \delta^2 \quad (33)$$

$$N_{i-1} = a_i/x_i = (\sqrt{a_i}/\delta^2) \sum_{j=1}^m \sqrt{a_j} \quad (34)$$

$$N_{\text{total}} = \left(\sum_{j=1}^m \sqrt{a_j}\right)^2 \frac{1}{\delta^2} \quad (35)$$

采用 SS 预抽样时选取的每层样本量通常并不多,得到的估计参数 $\sqrt{a_i}$ 会有一些误差,从而导致式(34)中各层最优样本量 N_i 发生偏差,以上问题可通过迭代方式逐步增加样本量解决。为叙述方便,再引入几个变量,设经 n_{it} 次迭代使 $\delta \leq \delta_t$, 记由式(34)得到的各层最优样本量为 $N_e^{(j)}(i)$, 第 j 次迭代前各层样本量为 $N^{(j-1)}(i)$, 第 j 次迭代第 i 层样本量计划增至 $N_t^{(j)}(i)$, 其中 $j = 1, 2, \dots, n_{it}$, $i = 0, 1, \dots, m-1$, $N_t^{(j)}(i)$ 按如下方式选取为

$$N_t^{(j)}(i) = \begin{cases} \text{当 } N_e^{(j)}(i) \geq N^{(j-1)}(i), \\ N^{(j-1)}(i) + \text{round}[(N_e^{(j)}(i) - \\ N^{(j-1)}(i)/n_{it} + 1 - j)] \\ \text{当 } N_e^{(j)}(i) < N^{(j-1)}(i), \\ N^{(j-1)}(i) \end{cases} \quad (36)$$

式中 $\text{round}[\cdot]$ 表示按四舍五入方式取整。经一次 SS 预抽样后,可得 $N^{(0)}(i) = N$, N 为 SS 预抽样选取的每层样本数。第 j 次迭代第 i 层样本量实际增至 $N^{(j)}(i)$, $N^{(j)}(i)$ 与 $N_t^{(j)}(i)$ 有微小差别,取决于后面介绍的样本增加方式。

上述过程解决了样本量应取多少问题,对算法设计而言,还需采用合适的方法增至所需样本量。

第 0 层中间域采用 DMCS 增加样本,因此第 0 层第 j 次迭代直接增加 $N_t^{(j)}(0) - N^{(j-1)}(0)$ 个样本。

第 i 层中间域 ($i = 1, 2, \dots, m-1$) 采用了 MMH 算法增加样本,样本量增加需要一些特殊处理,为此首先引入剩余种子点概念和几个新记号。将每条马氏链的初始状态点称为种子点,各中间域样本由两部分组成:一部分是长度为 n_s 的马氏链

状态点(取 $n_s = 1/p_0$),另一部分是还未采用 MMH 算法延长马氏链长度的种子点,将这部分种子点称为剩余种子点。记第 i 层中间域样本中,剩余种子点数为 $n_{01}(i)$,以马氏链状态点存在的样本数为 $n_{02}(i)$ 。在第 i 层中间域进行第 j 次迭代前,第 i 层中间域样本中剩余种子点由两部分构成:一部分是第 i 层中间域进行第 $j-1$ 次迭代后未用完的种子点,另一部分是第 $i-1$ 层中间域完成第 j 次迭代后新进入第 i 层中间域的样本点。给定马氏链链长 n_s ,根据 $n_{01}(i)$ 、 $n_{02}(i)$ 和 $N_t^{(j)}(i)$ 之间的关系,分三种情况加以考虑。

① 当 $n_{01}(i) + n_{02}(i) \geq N_t^{(j)}(i)$ 时,表明第 i 层第 j 次迭代所需样本量足够,无需增加样本。

② 当 $n_{01}(i) + n_{02}(i) < N_t^{(j)}(i)$ 且 $n_{01}(i) \times n_s + n_{02}(i) \geq N_t^{(j)}(i)$ 时,表明剩余种子点足够,无需增加种子点。从 $n_{01}(i)$ 个剩余种子点中随机选择 n_{ic} 个种子点,其中

$$n_{ic} = \lceil [N_t^{(j)}(i) - n_{01}(i) - n_{02}(i)] / (n_s - 1) \rceil \quad (37)$$

式中 $\lceil \cdot \rceil$ 为取不小于该数的最小整数。分别从这 n_{ic} 个种子点出发,利用 MMH 算法生成长度为 n_s 的马氏链。更新 $n_{01}(i)$ 、 $n_{02}(i)$, 计算 $N^{(j)}(i)$:

$$\begin{aligned} n_{01}(i) &= n_{01}(i) - n_{ic} \\ n_{02}(i) &= n_{02}(i) + n_{ic} \times n_s \\ N^{(j)}(i) &= n_{01}(i) + n_{02}(i) \end{aligned} \quad (38)$$

③ 当 $n_{01}(i) \times n_s + n_{02}(i) < N_t^{(j)}(i)$ 时,表明剩余种子点不够,需增加剩余种子点,共需种子点数为

$$n_{ic} = \lceil [N_t^{(j)}(i) - n_{02}(i)] / n_s \rceil \quad (39)$$

分别从 $n_{01}(i)$ 个种子点出发用 MMH 算法生成长度为 n_s 的马氏链,然后,采用迭代方式增加种子点和马氏链状态点。即对 $k=1, 2, \dots, n_{ic} - n_{01}(i)$ 依次执行:在第 i 个中间域当前所有 $(n_{01}(i) + k - 1) \times n_s + n_{02}(i)$ 个样本中随机选取 1 个样本点,从该样本点用 MMH 算法生成一个状态点,取新生成的状态点为种子点,再用 MMH 算法生成一条长度为 n_s 的马氏链。更新 $n_{01}(i)$ 、 $n_{02}(i)$, 计算 $N^{(j)}(i)$:

$$\begin{aligned} n_{01}(i) &= 0, \quad n_{02}(i) = n_{02}(i) + n_{ic} \times n_s \\ N^{(j)}(i) &= n_{01}(i) + n_{02}(i) \end{aligned} \quad (40)$$

ASS2 算法可描述:

① 输入 N , p_0 , δ_t , 原概率密度 $f(\boldsymbol{\theta})$, 建议分布 $h_{i0}(\boldsymbol{\xi}^{(i0)} | \boldsymbol{\theta}^{(i0)})$, n_{it} 。

② 执行一次 SS, 输出 \hat{P}_F , m , δ , b_{i+1} , $n_{F(i+1)}$, $T_{i+1}(j)$, 其中 $i = 0, 1, 2, \dots, m-1$ 。

③ 若 $\delta \leq \delta_t$, 则精度足够,执行⑥;否则,对

$j=1,2,\dots,n_{it}$, 执行④和⑤。

④ 对 $i=0,1,2,\dots,m-1$, 计算 $\hat{P}_F(i+1)$, $\gamma(i+1)$, 按式(36)估计 $N_i^{(j)}(i)$ 。

⑤ 根据 $n_{01}(i)$ 、 $n_{02}(i)$ 和 $N_i^{(j)}(i)$ 之间的关系增加中间域样本,更新 $N^{(j)}(i)$ 。

⑥ 计算 $\hat{P}_F = \prod_{i=1}^m \hat{P}_i$ 并输出 δ 。

4 数值算例

结构功能函数为 $g(\mathbf{x}) = -x_1 + a \sum_{i=2}^d x_i^2 - b$, 其

中 $a=0.025$, $b=20.27$, $d=1000$, $x_i (i=1,2,\dots,d)$ 服从标准正态分布, x_i 之间相互独立。本算例取自文献[5,6],为一高维非线性算例。

失效概率基准值采用 DMCS 估计,样本量取 10^8 ,对应的变异系数采用式(3)估计。采用 SS 时,取 $p_0=0.1$,建议分布取正态分布,均值参数取当前状态点坐标,标准差参数取 1,取每层样本量为 $N=1000$ 。为与 ASS1 和 ASS2 比较,再取 SS 每层样本量为 $N=5200$ 。采用 ASS1 算法时,取第 0 层样本迭代增量 $N_{0i}=100$,目标变异系数取 $\delta_i=0.10$,其余参数与 SS 同。采用 ASS2 算法时,取迭代次数 $n_{it}=5$,其余参数同 ASS1。SS($N=1000$)、SS($N=5200$)、ASS1 和 ASS2 各运行 500 次,失效概率均值、根据 500 次结果统计的变异系数值、功能函数估计次数均值列入表 1。

表 1 各方法计算结果对比

Tab. 1 Comparison of different methods

方法	P_F	c. o. v.	功能函数估计次数
DMCS	7.1497×10^{-4}	0.0037	1.0×10^8
SS($N=1000$)	7.0908×10^{-4}	0.2669	3635
SS($N=5200$)	7.0197×10^{-4}	0.1269	19221
ASS1	7.0960×10^{-4}	0.1171	22701
ASS2	7.0807×10^{-4}	0.1150	19127

由表 1 可知,采用 SS($N=1000$)运行 500 次,变异系数为 0.2669,结果的离散性太大,这意味着运行 1 次得到的 \hat{P}_F 实际估计值与 P_F 真值之间的偏差可能会很大。

采用 ASS1 和 ASS2 得到的失效概率均值和变异系数值大致相同,但 ASS1 所需的总样本量比 ASS2 多 18.7%,因此可以认为,在相同失效概率估计精度条件下 ASS2 具有更高的计算效率。注意到由 500 次失效概率结果统计的变异系数值比目标变异系数 δ_i 大,这与假设(1)各中间域之间的样本独立和(2)同一失效域不同马氏链之间的样本

独立相一致。原则上,可以进一步考虑同一失效域不同马氏链状态点间的相关性计算 δ ,但考虑中间域样本间的相关性推导 \hat{P}_F 对应的变异系数存在一定困难。经验表明,采用式(11)得到的 \hat{P}_F 变异系数估计值通常与 \hat{P}_F 实际变异系数接近。

为进一步比较 ASS2 与 SS 算法,按总样本量大致相等的原则调整 SS 每层样本量 N ,从表 1 可以看出,当 SS 每层样本量取 $N=5200$ 时,ASS2 与 SS 的功能函数估计次数大致相同。而此时 SS 对应的变异系数为 0.1269,大于 ASS2 对应的变异系数。因此,可以认为在相同的总样本量条件下,ASS2 的计算精度比 SS 更高。

如果采用 DMCS,则达到与 ASS2 相同的变异系数需要的样本量大致为

$$N_{\text{MCS}} = (1 - P_F) / P_F \delta^2 = (1 - 7.1497 \times 10^{-4}) / (7.1497 \times 10^{-4} \times 0.1150^2) = 1.057 \times 10^5 \quad (41)$$

因此,可以认为 ASS2 比 DMCS 的计算效率要高得多。

5 结 语

本文提出了两种以失效概率估计精度为停机控制参数的自适应子集模拟方法。理论分析和数值算例表明:

(1) 两种自适应子集模拟方法(ASS1 和 ASS2)能根据失效概率 P_F 的估计精度要求(δ_i)自适应地调整样本量。

(2) ASS1 比 ASS2 更直接简明,但是考虑样本量优化分配的 ASS2 方法能进一步减少总样本量,提高计算效率。

(3) 采用增量方式计算相关参数能进一步提高计算效率。

本文建议采用 ASS2 处理高维非线性可靠性分析问题中的失效概率计算问题。

参考文献(References):

- [1] 董 聪.现代结构系统可靠性理论及其应用[M].北京:科学出版社,2001.(DONG Cong. *The Theory of Structural Systems Reliability and its Applications* [M]. Beijing: Science Press, 2001. (in Chinese))
- [2] Au S K, Beck J L. Estimation of small failure probabilities in high dimensions by subset simulation[J]. *Probabilistic Engineering Mechanics*, 2001, 16(4): 263-277.

- [3] Au S K, Beck J L. Subset Simulation and its application to seismic risk based on dynamic analysis[J]. *Journal of Engineering Mechanics*, 2003, **129**(8): 901-917.
- [4] Au S K, Ching J, Beck J L. Application of subset simulation methods to reliability benchmark problems [J]. *Structural Safety*, 2007, **29**(3): 183-193.
- [5] Katafygiotis L S, Zuev K M. Geometric insight into the challenges of solving high dimensional reliability problems[J]. *Probabilistic Engineering Mechanics*, 2008, **23**(2-3): 208-218.
- [6] Zuev K M, Katafygiotis L. Modified Metropolis-Hastings algorithm with delayed rejection[J]. *Probabilistic Engineering Mechanics*, 2011, **26**(3): 405-412.
- [7] Zuev K M, Beck J L, Au S K, et al. Bayesian post-processor and other enhancements of subset simulation for estimating failure probabilities in high dimensions[J]. *Computers and Structures*, 2012, **92-93**: 283-296.
- [8] Koutsourelakis P S, Pradlwarter H J, Schuëller G I. Reliability of structures in high dimensions (part I) algorithms and applications [J]. *Probabilistic Engineering Mechanics*, 2004, **19**(4): 409-417.
- [9] Koutsourelakis P S. Reliability of structures in high dimensions. Part II: theoretical validation [J]. *Probabilistic Engineering Mechanics*, 2004, **19**(4): 419-423.
- [10] Pradlwarter H J, Schuëller G I, Koutsourelakis P S, et al. Application of line sampling simulation method to reliability benchmark problems [J]. *Structural Safety*, 2007, **29**(3): 208-221.
- [11] Zio E, Pedroni N. An optimized Line Sampling method for the estimation of the failure probability of nuclear passive systems [J]. *Reliability Engineering & System Safety*, 2010, **95**(12): 1300-1313.
- [12] Katafygiotis L S, Cheung S H. Application of spherical subset simulation method and auxiliary domain method on a benchmark reliability study [J]. *Structural Safety*, 2007, **29**(3): 194-207.
- [13] Zuev K M, Katafygiotis L. The Horseshoe Simulation algorithm for evaluation of small failure probabilities [J]. *Probabilistic Engineering Mechanics*, 2011, **26**(2): 157-164.
- [14] 陈向前, 董聪, 闫阳. 微分等价递归算法的解析格式 [J]. *计算力学学报*, 2011, **28**(5): 688-692, 710. (CHEN Xiang-qian, DONG Cong, YAN Yang. Analytical scheme of differential equivalent recursive algorithm [J]. *Chinese Journal of Computational Mechanics*, 2011, **28**(5): 688-692, 710. (in Chinese))

Adaptive subset simulation for structural reliability analysis

CHEN Xiang-qian*, DONG Cong, YAN Yang

(Department of Civil Engineering, Tsinghua University, Beijing 100084, China)

Abstract: Structural reliability analysis requires accurate and efficient failure probability evaluation. Two Adaptive Subset Simulation (ASS) methods were developed in this paper to deal with the high dimensional nonlinear structural reliability problems. Stopping criteria of the proposed methods were based on the estimated coefficient of variation, which was more reasonable. The sample size of the proposed ASS methods could be adjusted adaptively according to the estimated coefficient of variation. The ASS method with optimal sample size was more efficient. The accuracy and efficiency of the two ASS methods were verified by a numerical example.

Key words: structural reliability; failure probability; Subset Simulation; adaptive; optimal sample size