软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2015,26(2):305−320 [doi: 10.13328/j.cnki.jos.004783]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

面向安全攸关系统中小概率事件的统计模型检测*

杜德慧, 程 贝, 刘 静

(上海市高可信重点实验室(华东师范大学),上海 200062)

通信作者: 杜德慧, E-mail: dhdu@sei.ecnu.edu.cn

摘 要: 在开放运行环境中,安全攸关系统的不确定性行为有可能导致小概率事件的发生,而此类事件的可靠性指标往往很高,小概率事件一旦发生就会产生灾难性的后果,严重威胁到人们的生命、财产安全.因此,评估、预测小概率事件发生的概率,对于提高系统的可靠性具有重要意义.统计模型检测是一种基于模拟的模型验证技术,结合了系统的快速模拟及统计分析技术,能够有效提高模型检测的效率,适用于验证、评估安全攸关系统的可靠性,但其面临的挑战性问题之一是在可接受的样本数量下,使用统计模型检测技术难以预测、评估小概率事件发生的概率.因此,提出一种改进的统计模型检测框架,设计和开发基于机器学习的统计模型检测器,实现在相对较少的样本数量下预测和评估小概率事件发生的概率.结合轨道交通控制系统中避碰控制案例分析,进一步证明改进后的统计模型检测器能够有效预测和评估安全攸关系统中小概率事件发生的概率.

关键词: 统计模型检测;小概率事件;安全攸关系统;随机混成自动机;机器学习

中图法分类号: TP311

中文引用格式: 杜德慧,程贝,刘静.面向安全攸关系统中小概率事件的统计模型检测.软件学报,2015,26(2):305−320. http://www.jos.org.cn/1000-9825/4783.htm

英文引用格式: Du DH, Cheng B, Liu J. Statistical model checking for rare-event in safety-critical system. Ruan Jian Xue Bao/ Journal of Software, 2015,26(2):305−320 (in Chinese). http://www.jos.org.cn/1000-9825/4783.htm

Statistical Model Checking for Rare-Event in Safety-Critical System

DU De-Hui, CHENG Bei, LIU Jing

(Shanghai Key Laboratory of Trustworthy Computing (East China Normal University), Shanghai 200062, China)

Abstract: In open environment, the stochastic behavior of safety-critical system may lead to occurrence of rare-event, which is critical to the system's reliability. It is very important to estimate the probability of rare-event occurrence. Statistical model checking (SMC) is a simulation-based model checking technology, which integrates the simulation and statistical analysis technique to improve the efficiency of traditional model checking. SMC is used to verify and estimate the reliability of complex safety-critical system. However, the most challenging problem is that it is impossible to estimate and predict the probability of rare-event based on SMC with the acceptable sample size. To solve this problem, this study proposes an improved statistical model checking framework, designs and develops a statistical model checker based on machine learning to estimate and predict the probability of rare-event with fewer sample size. To demonstrate the presented approach, a case study on collision avoidance system in CBTC is discussed. The analysis results show that the proposed approach is feasible and efficient.

Key words: statistical model checking; rare-event; safety-critical system; stochastic hybrid automata; machine learning

安全攸关系统被广泛应用于航空航天、武器装备、医疗设备、交通、工控等与国计民生密切相关的安全攸关领域,其安全性、可靠性要求高,一旦发生系统失效,将严重威胁到人们的生命、财产安全[1,2].验证、分析安全攸关系统的安全性、可靠性已经引起国内外学者的广泛关注,所取得的一系列研究成果对于安全攸关系统的

* 基金项目: 国家自然科学基金(61472140, 61202104); 上海市自然科学基金(14ZR1412500, 13511503100)
收稿时间: 2014-07-02; 修改时间: 2014-10-31; 定稿时间: 2014-11-26

设计、开发具有重要意义.传统方法通常采用模型检测等技术验证安全攸关系统的模型是否满足安全性约束,以提高系统的安全性及可靠性.与传统方法不同,我们更加关注安全攸关系统中的小概率事件(rare-event),评估、预测小概率事件发生的概率,对于提高系统的安全性、可靠性具有重要意义[3].例如,轨道交通控制系统中的避碰(collision avoidance)控制,主要功能是防止列车相撞,此类事件发生的概率极小,但是这种小概率事件一旦发生,就会严重影响到系统的安全性和可靠性.因此,如何使用模型检测技术评估、预测小概率事件发生的概率,是验证、评估安全攸关系统的可靠性面临的挑战性问题之一[3].

传统的模型检测技术通过对系统的状态空间进行全遍历,能够自动验证、分析系统行为的安全性、可靠性,但是,其面临的主要问题是状态空间爆炸(即,随着系统规模的增加,系统的状态空间呈指数级增长).随着系统规模的扩大,传统的模型检测技术很难验证、分析复杂系统行为的正确性、可靠性,也无法进行定量评估分析[4].因此,使用传统的模型检测技术评估、预测安全攸关系统中的小概率事件是不可行的.

统计模型检测技术(statistical model checking,简称 SMC)能够有效评估系统模型满足属性约束的概率区间,可用于定量分析系统模型的性能指标,能够有效缓解传统模型检测面临的状态空间爆炸问题[5].统计模型检测技术主要使用统计方法分析复杂系统的模拟执行路径,其优点是效率高、执行速度快.这项技术的关键点在于:首先,生成系统的模拟路径,并对其进行伯努利实验,判断每次模拟路径是否满足给定的系统属性约束;其次,使用假设检验的方法对系统模拟路径的样本空间进行统计分析,评估系统满足属性约束的概率区间.对于大型、复杂的系统而言,生成系统模拟路径要更快、更容易,而不需要将系统模型转换为验证工具的输入语言,也不需要构造系统的符号模型,例如马尔科夫链(Markov chain).因此,使用统计模型检测技术对大型、复杂的安全攸关系统进行安全性分析、性能评估,为模型检测安全攸关系统的安全性、可靠性提供了广阔的研究前景.目前,SMC 技术引起了学术界和工业界的广泛关注,已成功应用于分析实时调度[6]、系统生物[7,8]、能耗感知的智能家居(energy-aware smart building)[9]等.

但是,统计模型检测面临的主要问题是难以评估小概率事件发生的概率,当小概率事件发生的概率趋近于 0 时(概率取值约为 10^{-10}),统计分析需要的样本空间急剧增加,统计模型检测算法的效率大为降低[10].安全攸关系统的设计、开发对系统的安全性要求较高,人们往往关注的是如何确保系统具有安全性,而对于安全性属性的否命题“系统不安全”这种小概率事件则关注得比较少.如何预测、评估小概率事件发生的概率,对于提高安全攸关系统的安全性、可靠性具有重要意义.然而在实际应用中,使用统计模型检测技术评估、预测小概率事件发生的概率是难以实现的.针对上述问题,我们提出基于机器学习的方法,改进现有的统计模型检测算法,设计、实现面向小概率事件的基于机器学习的统计模型检测框架.我们的目标是:将该模型检测框架应用于评估、预测安全攸关系统中的小概率事件发生的概率,帮助提高轨道交通控制系统模型的可靠性、安全性.

1 面向小概率事件的统计模型检测方法

本节首先介绍面向小概率事件的统计模型检测方法,其中重点讨论我们提出的基于机器学习的统计模型检测框架;然后,分别介绍其两个重要组成部分——学习型模型检测器和自适应统计分析器.学习型模型检测器的核心部分主要包括 Trace 提取算法及基于支持向量机(support vector machine,简称 SVM)的小概率事件预测器;自适应统计分析器集成了几种经典的 SMC 算法,根据事件发生的概率大小,自适应地选择 SMC 算法,以达到高精确度、高效地评估小概率事件发生概率的目的.

1.1 基于机器学习的统计模型检测框架

如图 1 所示是基于机器学习的统计模型检测框架,将目标系统的可执行模型$\mathcal{M}$、系统需要满足的属性约束的概率有界线性时态逻辑(probabilistic bounded linear temporal logic,简称 PBLTL)公式 $P_{\leqslant\theta}(\mathcal{F}')$和统计精度的参数(取决于具体的 SMC 算法)作为输入,通过统计模型检测器,评估在有限的资源约束下,系统模型满足属性约束的概率区间.基于机器学习的统计模型检测器包括两个核心模块——学习型模型检测器和自适应统计分析器.其中,

- 学习型模型检测器接受系统模型$\mathcal{M}$以及有界线性时态逻辑(bounded linear temporal logic,简称 BLTL)

公式$\mathcal{F}'$作为输入,借助经典的模型检测算法判断单个样本 Trace π_i 是否满足属性约束$\mathcal{F}'$:如果满足,则返回结果 1;否则为 0.

- 统计分析器收集到 0/1 结果作为统计分析的样本,动态执行 SMC 算法,根据现有样本信息是否足以使算法终止于所需精度,确定是否需要更多样本.如果需要更多样本,则重新返回学习型模型检测器,继续生成新的样本 Trace.如此迭代执行,直至结束.

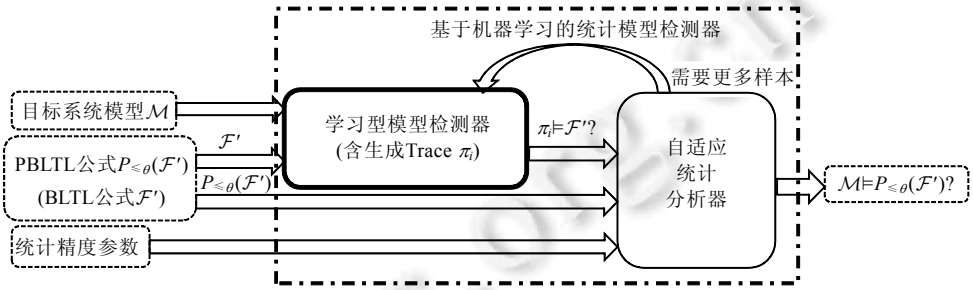


Fig.1 Framework of statistical model checking based on machine learning (ML-SMC)

图 1 基于机器学习的统计模型(ML-SMC)检测框架

基于机器学习的统计模型检测框架的特点是:借助机器学习的方法提高检测单个 Trace 是否满足属性约束的效率,并优化模拟执行的步长;此外,根据统计分析的情况,动态、合适地选取统计分析算法,实现具有自适应能力的统计分析器.基于机器学习的统计模型检测器具有统计分析与预测的能力,能够有效地定量评估系统的性能、定性验证系统的可靠性并预测小概率事件发生的概率区间.

1.2 学习型模型检测器

学习型模型检测器是本文的主要贡献,如图 2 所示.

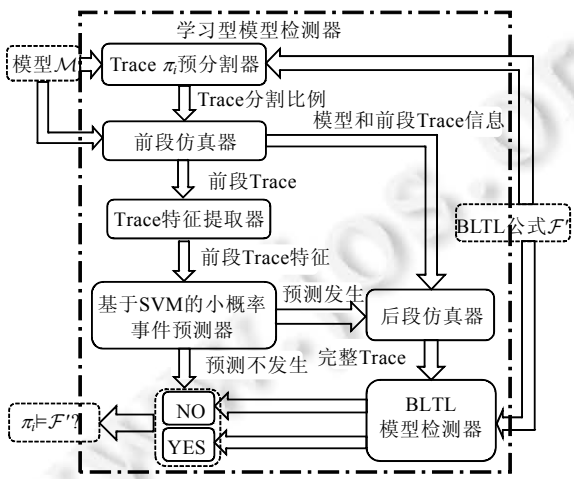


Fig.2 Learning-Based model checker

图 2 学习型模型检测器

学习型模型检测器主要包括 6 个组成部分:Trace 预分割器、前段仿真器、Trace 特征提取器、基于 SVM 的小概率事件预测器、后段仿真器以及 BLTL 模型检测器.首先,Trace 预分割器根据模型$\mathcal{M}$和 BLTL 公式$\mathcal{F}'$的特点决定合适的 Trace 前段与后段的分割比例,比例的选取直接影响 SVM 预测器的学习效率和预测的准确性,此部分将在第 1.3 节详细讨论.根据系统模型与分割比例,前段仿真器执行前段仿真并获得相应的 Trace 信息.

然后,由 Trace 特征提取器根据前段 Trace 提取特征并做一些预处理,将每个状态下的相关变量组成的向量作为 Trace 的特征,输入到基于 SVM 的小概率事件预测器,该特征的选取对 SVM 的学习和预测有很大影响,此部分内容将在第 1.3 节详细讨论.基于 SVM 的小概率事件预测器通过对预先给定的样本集进行学习,以具备一定的能力来预测后段 Trace 的状态信息.因此,对于预测小概率事件不太可能发生的 Trace,直接判定结果为 0,以提高整体效率.SVM 预测器需要提前学习一部分样本才能具备预测能力,所以训练样本的数量和质量也对预测过程有较大的影响,此部分将在第 1.4 节详细讨论.只有当预测器预测到小概率事件发生时才会执行后段仿真器与 BLTL 模型检测器,其过程与传统的统计模型检测类似,通过后段仿真器得到完整 Trace 后,再由 BLTL 模型检测器判断 Trace π_i 是否满足公式$\mathcal{F}'$,并将判定结果返回给统计分析器.

我们特别关注随机混成系统的行为,因此采用随机混成自动机(stochastic hybrid automata,简称 SHA)建模系统的行为,作为学习型模型检测器的输入模型.SHA 是混成自动机具有随机语义的版本,其状态处的时间延迟(delay)可以是某一时间区间内的任意时间点,即,时间延迟的密度函数服从均匀分布;或者时间延迟服从参数为 λ的指数分布 exp(λ).此外,状态之间的迁移动作可以是离散概率分布.随机混成自动机已被成功地应用于建模随机混成系统,能够准确地建模混成系统的随机行为[11].

定义 1. 随机混成自动机$\mathcal{H}$是一个元组$\mathcal{H}=(L,l_0,X,S,E,F,I)$,其中,

(1) L 表示位置(Location)的集合.

(2) $l_0\in L$ 为初始位置.

(3) X 为连续变量的有限集.

(4) 动作的有限集$\Sigma=\Sigma_i\uplus\Sigma_o$ 包括输入动作集合Σ_i与输出动作集合Σ_o.

(5) E 表示迁移的有限集,其中每条迁移边可表示为(l,g,a,φ,l'),其中,l 和 l'表示位置,g 为定义在$\mathbb{R}^X$上的谓词,动作标签 $a\in\Sigma,\varphi$是定义在$\mathbb{R}^X$上的二元关系.

(6) 对于每一个位置 $l\in L,F(l)$为状态处的时间延迟函数.

(7) I 为任意位置赋值一个不变式 $I(l)$.

需要指出的是,这里,时间延迟函数 $F(l)$及不变式 $I(l)$使用的是较为通用的定义,具体到模型验证工具 UPPAAL-SMC[12]中,延迟函数被用于定义一系列待求解的 ODEs 公式,可建模系统的混成行为.其详细的建模形式请参见本文的案例分析部分.

定义 2. 随机混成自动机的语义模型.

使用基于时间标签转换系统(timed labeled transition system,简称 TLTS)描述 SHA 的随机语义模型,其中,系统的状态表示为$(l,v)\in L\times R^X$且 $v\vDash I(l)$,状态之间的迁移有两种形式:

- 时间的延迟迁移为 $(l,v)\xrightarrow{d}(l,v')$, 其中,$d\in\mathbb{R}_{\geqslant 0}$,且 $v'=F(d,v)$.
- 动作的迁移为 $(l,v)\xrightarrow{a}(l',v')$, 表示存在一条迁移边$(l,g,a,\varphi,l')$,使得 $v\vDash g$ 且$\varphi\in(v,v')$.

对于给定的 SHA $\mathcal{H}$,假定某个状态处的时间延迟、输出存在如下的概率分布:

(1) 实数集$\mathbb{R}_{\geqslant 0}$上时间延迟的概率密度函数μ_s,表示迁移何时离开此状态的概率,且 $\int\mu_s(t)\cdot\mathrm{d}t=1$.

(2) 状态迁移的输出概率函数γ_s 表示状态迁移发生时,其输出为动作集 $o\in\Sigma_o$ 中某个输出动作的概率分布,$\Sigma_o\gamma_s(o)=1$.

离散的动作迁移 $(l,v)\xrightarrow{a}(l',v)$, 存在迁移$(l,g,a,Y,l')$,其中,$v\vDash g$ 且 $v'\vDash v[Y]$.对于每个状态(l,v),迁移的输出概率函数γ_s是集合 $\{o:(l,g,o,-,-)\in E^j\wedge v\vDash g\}(o\in\Sigma_o^j)$ 上的均匀分布.SHA 详细的概率语义模型参见文献[13].

在 UPPAAL-SMC 中,建模状态处的时间延迟的概率分布可采用两种形式:

(1) 当状态处的时间延迟使用有界的数组$[d(l,v),D(l,v)]$表示时,delay 服从区间$[d(l,v),D(l,v)]$上的均匀分布,$d(l,v)$是时间延迟的下确界,$D(l,v)$是上确界;

(2) 当状态处定义参数λ时,时间延迟的密度函数μ_s服从参数为λ的指数 exp(λ).

1.3 Trace的分割与特征提取

使用 SVM 技术进行 Trace 预测.首先,需要确定预测器的输入信息,包括分割 Trace 并提取 Trace 特征.Trace 分割的目的在于找到一个合适的时间点,使其前段 Trace 在尽可能短的条件下最大程度地包含对预测有用的信息,以便 SVM 预测后段 Trace 的信息.Trace 分割的思想源于一种经典的小概率事件模拟技术 Splitting (RESTART)[14],即,基于条件概率.假设在 A 事件发生的情况下,B 事件更有可能发生,如果 $P(A)>P(B)$,则先在模拟 Trace 的过程中找到 A 事件的发生点,在此前提下向后进行多次模拟,以提高捕获到 B 事件的可能性.本文提出基于学习的方法是以此为依据,利用机器学习的技术,通过 A 事件来预测 B 事件的发生,即,条件概率理论为学习和预测的成功提供了理论上的可能性.

由于我们使用 BLTL 来描述属性,因此需要在时间或步长约束的基础上进行分割.BLTL 是 LTL 的扩展,逻辑公式语法表示如下:

$$\mathcal{F}'::=(\phi_1\vee\phi_2)|\neg\phi_1|(\phi_1 U^t\phi_2),\phi::=l|v\circ u,$$

其中,$l\in L$,L 表示离散状态的集合;$v\in V$,V 表示连续变量的集合,则 $v\circ u$ 表示对 V 上变量的一种约束,$\circ\in\{\geqslant,\leqslant,=\}$,$u\in\mathbb{R}$;$t\in Q_{\geqslant 0}$ 表示时间或步长约束.通常,我们将“在时间 t 内最终会”的属性记为 F^t,即 $F^t\psi=\text{True}U^t\psi$.

假定已经获得训练集样本 Π^T,其中,少数类样本集合为 Π^Y,根据 Π^Y 可以得到小概率事件发生的时间点的分布 D^Y,求得 D^Y 的期望 E^Y.这里,我们引入一个预测因子 ρ,设分割点为 τ,则

$$\tau=E^Y-\rho,\rho\in(0,E^Y).$$

ρ 值越大,分割点离小概率事件发生点就越远,预测难度也就越大,但预测正确的收益也更大.ρ 的取值可以通过经验手动调整来获得更好的预测效果,但通常应受到 *bound*(时间或步长约束)的制约,本文采用的方法是根据 E^Y 在 *bound* 范围内的位置来动态确定 ρ,即

$$\rho=\frac{E^Y}{bound}\cdot E^Y.$$

可以看出:E^Y 离 *bound* 越远,ρ 值就越小,即越接近 E^Y.

接着进行特征提取,考虑 3 种解决方案:

(1) 将所有的状态和变量信息作为输入向量,即保留全部 Trace 信息.但输入向量巨大会导致训练速度缓慢甚至训练失败,并且特征信息也不明显.

(2) 只以某些关键事件的发生时间点或时间延迟作为输入向量,但信息有可能过少,导致预测不准确.

(3) 折中方式:在关键时间点上某些关键的位置和连续变量作为输入向量.

本文采用折中方式,下面将介绍如何获得系统中的关键时间点上的关键信息.根据统计模型检测技术的特征,我们提出基于属性约束和随机行为的特征提取方法,此方法主要分为两部分:首先,确定需要提取的特征变量;其次,根据特征变量提取训练集中 Trace 的具体特征.

首先,为了便于描述特征变量确定过程,我们对系统中变量的关键程度进行分级,直接出现在属性约束中的变量称为 1 级变量,直接影响 1 级变量变化的称为 2 级变量,即,直接影响 $n-1$ 级变量变化的称为 n 级变量.在 SHA 中,存在连续变量与位置(即离散变量),其中,连续变量的影响关系可以通过微分方程找到,即,$x^n=\frac{\mathrm{d}x^{n-1}}{\mathrm{d}t}$ 中,当 x^{n-1} 为 $n-1$ 级变量时,x^n 为 n 级变量;位置的影响关系可以通过状态间的迁移关系找到,即,(l^n,t,l^{n-1})中,当 x^{n-1} 为 $n-1$ 级变量时,x^n 为 n 级变量.因此,在确定特征变量时,首先将 $\mathcal{F}'$ 中的所有变量(属于 1 级变量)加入特征集 Φ.对于连续变量 $x^n\in X^n$,遍历模型中相关微分方程 $x^n=\frac{\mathrm{d}x^{n+1}}{\mathrm{d}t}$,对于 $x^{n+1}\notin\Phi$,将其加入 Φ 中;对于位置 $l^n\in L^n$,通过遍历模型中 l^n 的所有入边(l^{n+1},t,l^n),对于 $l^{n+1}\notin\Phi$,将其加入 Φ(X^n 和 L^n 分别为连续变量和位置的集合).迭代上述过程,可依次将 1 级~n 级特征变量加入集合 Φ.在本文案例(参见第 2 节)中,1 级特征变量为属性约束中直接出现的路程变量 *pos*0 和 *pos*1,2 级特征变量为与路程直接相关的速度变量 $v0$ 和 $v1$,3 级特征变量为与速度直接相关的加速度变量 $a0$ 和 $a1$.

其次,要确定每条 Trace 上的关键时间点,并提取 Trace 上关键时间点的特征变量,生成特征向量.在随机系统中,最终结果的不确定性由随机行为导致,因此,随机行为发生的时间点为关键时间点.由 SHA 定义可知,其随机行为体现在:(1) 位置上的时间延迟;(2) 多个目标位置之间的离散概率选择.因此,对于每条 Trace,检测其前段即$[0,\tau]$内上述随机行为发生的时间点,将这些时间点以及其对应时间上的特征变量值拼接,得到此 Trace 的特征向量.在本文案例(参见第 2 节)中,关键时间点主要为列车与无线闭塞中心 RBC 的延迟发送通信消息、丢失消息,以及列车制动控制失败的时间点.下面我们给出特征提取算法的伪代码.

算法 1. Trace feature extraction part I.

//Determine what features we need

Input: BLTL property $\mathcal{F}'$, target model $\mathcal{M}$.

Output: feature set Φ.

1. Φ:=$\varnothing$;
2. Get the set V of all locations and clock variables in $\mathcal{F}'$;
3. For all v in V
4. If (v is a clock variable)
5. For all differential equations $v'=\frac{\mathrm{d}v}{\mathrm{d}t}$
6. If ($N^{\Phi}\leqslant N_{\max}\wedge v'\notin\Phi$)
7. Φ:=$\Phi\cup\{v'\}$;
8. End if
9. End for
10. Else //v is a location
11. For all ingoing edges (v',t,v)
12. If ($N^{\Phi}\leqslant 10\wedge v'\notin\Phi$)
13. Φ:=$\Phi\cup\{v'\}$;
14. End if
15. End for
16. End if
17. End for.

算法 2. Trace feature extraction part II.

//Extract features of one trace

Input: target model $\mathcal{M}$, feature set Φ, trace π.

Output: trace's feature vector α.

1. Get the set L^a of delayed locations and the set L^b of subsequent locations of branches;
2. For all states $s(t)$ of π
3. For all $l^a\in L^a$
4. If (s contains l^a)
5. $\forall\phi\in\Phi$, get $\phi(t)$
6. Form vector $\alpha^a=(l^a,\phi_1^a(t),...,\phi_n^a(t),t)$;
7. End if
8. End for
9. For all $l^b\in L^b$
10. If (s contains l^b)

11. $\forall \phi \in \Phi$, get $\phi(t)$
12. Form vector $\alpha^b = (l^b, \phi_1^b(t), ..., \phi_n^b(t), t)$;
13. End if
14. End for
15. End for
16. //form the final feature vector
17. $\alpha = (\alpha_1^a, ..., \alpha_n^a, \alpha_1^b, ..., \alpha_n^b)$.

对于算法 1,我们基于属性约束,结合微分方程和状态迁移关系,能够快速定位与属性约束相关的变量,具有一定的通用性;对于算法 2,由于其基于随机行为出现的时间点提取特征,因此并不适用于随机行为过少,甚至没有随机行为的系统.算法 1 与算法 2 相结合,特别适合于随机混成系统(如列车控制系统)的 Trace 特征提取.为了保证 SVM 学习效率,Trace 特征不宜过多,根据经验,一般取 3 级特征,并限定一个最大特征数,如 10,和一个最大的随机行为时间点数,如 100,以此保证每条 Trace 的特征数量不超过 1 000(SVM 完全可以处理此数量的特征).

1.4 基于SVM的小概率事件预测器

SVM 是 Cortes 等人于 20 世纪 90 年代提出的一种机器学习模型[15],其本质上是求解一个凸优化问题.它遵循结构风险最小化原理,综合考虑先验风险与置信风险,与传统的机器学习方法相比有很多优势,如容易训练、能够解决小样本及局部极小值等问题.SVM 分为线性和非线性两种:对于线性问题,其基本思想是构造一个最大化分类间隔的超平面(最优超平面);对于非线性问题,则通过一个输入变换将输入向量空间的数据映射到高维特征空间中(通过引入核函数),再在高维特征空间中对输入向量进行与线性分类类似的操作,即,构造最优超平面.详细的 SVM 分类原理参见文献[16].

近年来,SVM 已经广泛应用于许多领域,但在模型检测领域则很少有人关注.事实上,SMC 与机器学习有着共同的特征,即,对样本进行统计分析,且 SMC 最终判定属性是否满足,可看作一个两类分类问题(SVM 本质上是两类分类器).而不同之处在于,SMC 并没有对已经分析的样本进行记录和学习.通过对大量样本 Trace 的观察,可以发现:(1) 多条 Trace 之间有大量的重复信息;(2) 一条 Trace 的前后也有某种潜在的关联性,或者说事件的发生具有某种潜在的累积性,如在特定时间多次错误的累计出现才会导致某事件发生.因此,我们认为,对统计样本的记录和学习能够有效提高 SMC 算法的性能.

本算法引入了机器学习来预测 Trace 结果,因此会导致 SMC 算法得到的统计结果不准确.为了尽量减少预测准确性给 SMC 带来的影响,我们利用 SVM 中分类点到分类超平面的距离 x[17,18]来区分可信与不可信样本.给定一个分类阈值ε(文献[17]中建议设为 1 左右),当 $x<\varepsilon$,即,离分类平面较近时,我们认为分类结果不可信.不可信的 Trace 样本会进入后段仿真器模拟出完整 Trace,以得到确定结果.如图 3 所示为 SVM 小概率事件预测器的学习和预测流程.

首先,进行一定量的完整 Trace 模拟,得到初始样本集(注:初始样本集必须包含一定量的正样本和负样本,即小概率事件发生的样本,否则无法开始 SVM 的学习.如果在 SMC 算法结束都未得到足够负样本,此过程则不会开始,本文的算法退化为普通 SMC 算法).利用初始样本训练得到初始 SVM S_0,置信度为 W_0.

此后,每次产生新 Trace 都进行一次预测,根据前文所述,判断是否为可信样本:如可信,则使用预测结果,直接开始下一次模拟;如不可信,则进入后段仿真器,模拟出完成 Trace,检查其真实结果.接着对比真实结果与预测结果:如果预测正确,则开始下一次模拟;如果预测错误,则将此 Trace 加入错误样本集,待错误样本达到一定量时,在现有 S_0 基础上增量训练新的 SVM S_i,置信度为 W_i.迭代上述过程,直至 W_i 达到一定值时,便不再训练新的 SVM,直接在 S_i 的辅助下完成 SMC 验证.关于 SVM 置信度的具体内容,参见文献[18];关于 SVM 增量学习的具体内容,参见文献[19].

对于 SMC 小概率事件验证问题,由于小概率事件在模拟中很难出现,所以最终得到的训练样本集中两类样本数量严重不平衡,类似于机器学习领域的非平衡数据集的分类问题.文献[20]对目前解决非平衡类问题的多

种方法进行了分析和比较,认为虽然方法众多,但尚无一种最优方法.SVM 的诸多优良特性使得它在处理非平衡类问题时同样具有一定的优势.通常的非平衡类问题研究主要集中于数据挖掘等方面,其中很多问题并不会出现在 SMC 中,如数据碎片、噪声等,我们面临的唯一问题是绝对样本稀缺.

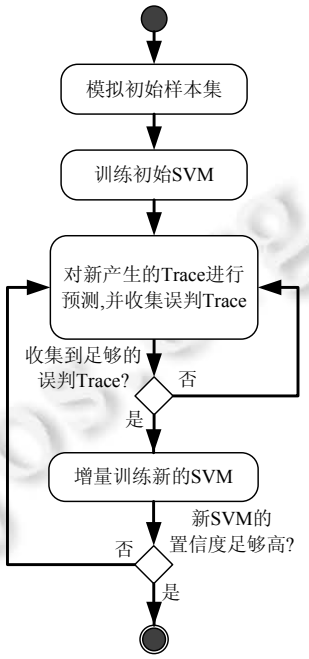


Fig.3 Learning and prediction flow of SVM-based rare-event predictor

图 3 SVM 小概率事件预测器的学习和预测流程

本文将采用下面两种方法来解决 SVM 的非平衡数据学习:

(1) 代价敏感性.1997 年,Osuna 等人就提出了 C-SVM,即,用两个正则化参数 C^+和 C^-来控制两类的错误惩罚;1999 年,Veropoulos 等人又对 C-SVM 进行改进,给两类赋予不同的代价.2003 年,Raskutti 等人又对代价的选取方式做了进一步的讨论和改进.本文算法基于 LIBSVM 实现.LIBSVM 是一个高效且易用的 SVM 软件包,已实现 C-SVM,其具体内容参见文献[21].

(2) 重采样.主要分为过采样(over-sampling)和欠采样(under-sampling).本文采用欠采样的方法,通过丢弃一部分多数类样本来减小数量差距.由于 SMC 是对同一场景进行反复模拟,因此获得的样本 Trace 会有很高的相似度.通过欠采样的方法,可以自动排除高相似度的 Trace,以精简多数类样本集,提高预测的准确率.本文预设了一个可接受比例系数,即, $Ratio=\frac{N^+}{N^-}$ (如 $Ratio$=4),N^-与 N^+分别表示少数类和多数类样本数量.

1.5 自适应统计分析器

为了提高统计模型检测的效率及评估结果的精确度,我们提出了自适应统计分析算法,即,在自适应统计分析器中集成几种常用 SMC 算法.文献[22]对简单取样规划(simple sampling plan,简称 SSP)[10]、顺序概率比检验(sequential probability ratio test,简称 SPRT)[10]、贝叶斯假设检验(Bayesian hypothesis testing,简称 BHT)[23]以及贝叶斯区间估计检验(Bayesian interval estimation testing,简称 BIET)[23]这 4 种算法做了详细的实验对比.根据实验结果可知:SSP 作为 SPRT 的退化算法,可应用性较差;而 SPRT,BHT 和 BIET 这 3 种算法则各有优势.总之,SPRT 和 BHT 算法所需样本数量较少,因此耗时较少,但统计分析的结果精度较低.BHT 算法和 SPRT 算法存在两大问题:(1) 当概率阈值θ逐渐靠近边界 0 或 1 时,所需样本数量会先急剧增长,然后在某点急剧减少,之后所需

样本数量会极少,统计结果变得很不准确;(2) 随着 SPRT 中容错范围α和β或 BHT 中贝叶斯因子可接受阈值 T 的增大,结果的准确度会有所提高,但所需样本数量会呈指数增长,因此,很难在实际应用中做精确统计.而 BIET 与其他方法的不同之处在于,BIET 是基于区间估计的算法,可以给出统计的概率区间,所需样本数量较多,但所得结果也更准确,很适合评估小概率事件发生的概率.

通过分析可知,现有的统计模型检测算法各有优缺点.因此,我们提出一种具有自适应能力的统计分析算法.首先,根据用户对先验分布的提供情况选择使用 SPRT/BHT 算法进行粗略分析,如果有先验分布时使用 BHT 算法(先验分布有利于基于贝叶斯的方法快速收敛),否则使用 SPRT 算法.当 SPRT/BHT 的计算误差变大或是样本数量急剧增大时,若尚未得到判定结果或有更高精度的需要,则采用 BIET 算法进行统计分析.算法流程如图 4 所示.

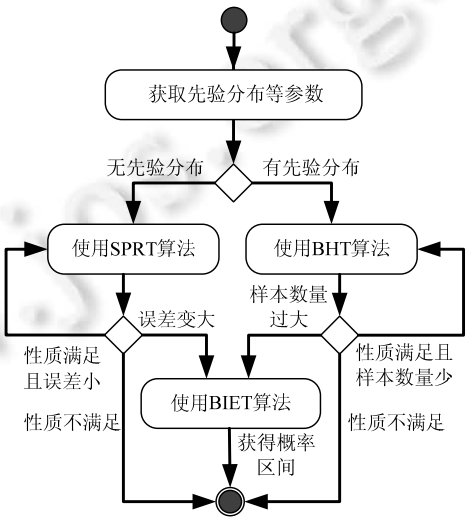


Fig.4 Adaptive statistical model checking algorithm flowchart

图 4 自适应统计模型检测算法流程图

自适应统计模型检测算法的伪代码如下:

算法 3. Adaptive statistical model checking.

Input: PBLTL property $P_{\geqslant\theta}(\mathcal{F}')$, prior density g, half-interval size $k\in\left(1,\frac{1}{2}\right)$ in BIET, interval coverage coefficient $c\in\left(\frac{1}{2},1\right)$ in BIET.

Output: Whether or not the PBLTL property is satisfied, and the probability $\hat{p}$ if satisfied.

```
1.  n:=0;  //The number of traces used this time
2.  m:=0;  //The number of traces used last time
3.  trs:=null;  //The set of traces drawn so far
4.  Boolean sat:=true;
5.  While (sat==true & not (n<<m))
6.   m:=n;
7.   If (g==null)
8.    Call SPRT(sat,n,trs);  //Sat and n are output  //Record all traces analyzed in trs
9.   Else
10.   Call BHT(sat,n,trs);  //Sat and n are output  //Record all traces analyzed in trs
```

11. End if
12. $\theta:=\theta\cdot 0.5$;
13. End while
14. If (*sat*)
15. Call $BIET(trs,\hat{p})$; //Reuse the traces for SPRT or BHT
16. Return $\hat{p}$;
17. Else
18. Return false.

本算法特别适用于属性约束是否满足或概率区间的范围完全未知的情况.算法在执行中会进行自适应调整,迅速定位概率区间,并选取合适的算法得到更精确的结果.本算法的设计基于大量的实验经验[12,22,23],具有一定的通用性.

2 列控系统案例分析

2.1 列控系统避碰控制简介

基于通信的列车控制系统(communication based train control system,简称 CBTC)已经广泛应用于城市轨道交通控制系统的设计和开发,通过高精确度的列车定位技术和双向、大容量的车地通信技术,实现列车按照移动闭塞制式追踪运行,使列车能以更短的追踪间隔安全运行[24].CBTC 的核心功能模块——自动列车防护(automatic train protection,简称 ATP)主要负责列车状态信息以及数据信息的处理并控制列车运行.目前,CBTC 系统已广泛应用于我国城市轨道交通控制系统,其主要的供应商有西门子、阿尔斯通、阿尔卡特等.为了进一步证明本文方法的可行性及有效性,我们将重点围绕该系统的核心模块 ATP 中的避碰控制展开讨论.

在 CBTC 中,轨道控制中心的任务由无线闭塞中心(radio block center,简称 RBC)处理.在整个系统中,每辆列车实时监控自己的速度、位置信息,并发送给负责当前区域的 RBC.每一个 RBC 则实时接收并处理自己管辖区域的列车,为它们规划前进路线,即,RBC 会赋予每辆列车一段安全移动范围,称作移动授权(move authority,简称 MA).如图 5 所示,Train0 和 Train1 都以一定的速度行驶,我们需要计算无线通信截止距离(deadline)以及当没有收到新的移动授权消息时需启动紧急制动的安全刹车距离(brake distance).Error 表示一段容错距离,以应对诸如位置测量偏差等问题.Position in RBC 为 Train0 上一次发送位置消息的占位,Train1 此时认为 Train0 在 Position in RBC 的位置.

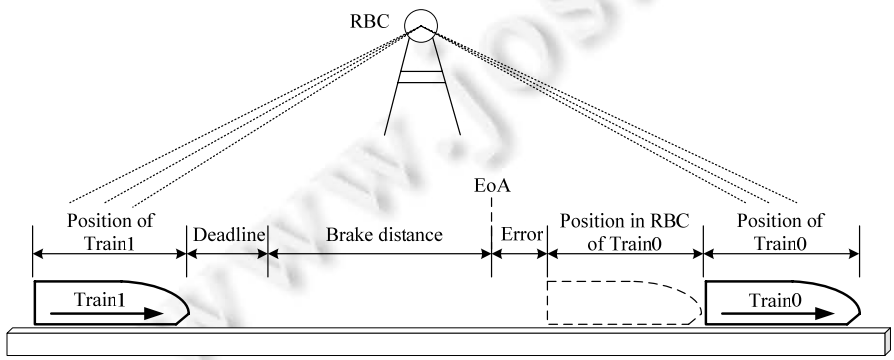


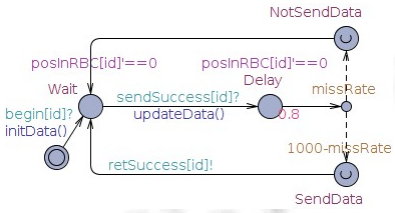
Fig.5 Distance control between trains in CBTC

图 5 CBTC 中的列车间距控制

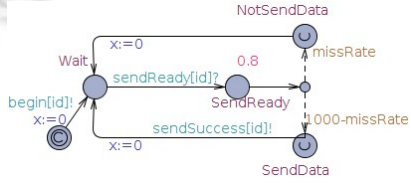
在该系统中,无线数据传输与处理必须是实时、高效的.车与 RBC 之间通过无线通信(global system for mobile communications-railway,简称 GSM-R)实现数据交换,是整个列车系统能否安全、高速运行的关键[25].但

是基于 GSM-R 的通信并不是安全、可靠的,数据包可能被延迟甚至丢失,因此,列车必须在某一截止距离 Deadline 内判断前方轨道是否安全,如果不安全,则必须启动紧急刹车.这种关键数据包被延迟或丢失将严重影响列车运行的整体效率,在某些极端情况下(如刹车控制延迟或失灵)甚至可能导致撞车的发生,我们需要评估并预测在该系统中此类小概率事件发生的可能性.

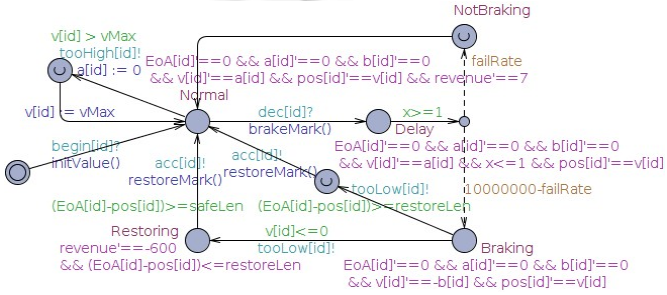
ATP 的避碰控制主要由 RBC、列车通信模块、列车制动装置及列车控制器这 4 个构件实现,使用随机混成自动机 SHA 建模,如图 6 所示.我们着重于验证系统通信和控制的安全问题,因此为了便于分析,对模型做了如下简化:(1) 由于消息延迟,RBC 和各列车之间信息具有不一致性,我们只考虑列车的真实位置与在 RBC 中记录的上一次的位置,随后,列车直接读取 RBC 中信息计算出移动授权终点(end of authority,简称 EoA),忽略其他可能的不一致性;(2) EoA 只限于前方列车的占位、没有道岔等,并且只考虑两辆列车,一前一后;(3) 列车减速制动后不再加速,即,只考虑一辆列车追逐前方列车的短暂情形的安全性;(4) 没有考虑乘客人数、人工操作、列车质量、列车长度等问题.



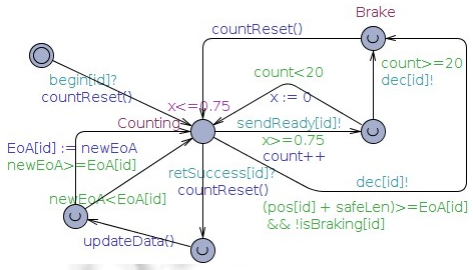
(a) RBC 通信构件



(b) 列车通信构件



(c) 列车制动构件



(d) 列车控制器

Fig.6 SHA model of collision avoidance control case

图 6 避碰控制案例的随机混成自动机模型

构件 RBCcom 建模 RBC 的通信,系统初始化后处于 Wait 状态,接收到列车发来的 sendSusccess 消息后,在 *updateData*(·)中更新 RBC 中记录的此列车信息.由于测量所得到的列车位置信息会有一定的测量误差,模型中使用 *posInRBC*[*id*]=*pos*[*id*]+*random*(*uncertainPos**2)表示列车位置,其中,*random*(*uncertainPos**2)表示测量误差;之后,RBC 进行必要的计算和存储,此操作所需要的时间延迟服从参数为 0.8 的指数分布,即,exp(0.8);最后,发送返回消息 retSuccess,消息丢失率以 missRate 表示,在 SHA 中使用离散概率建模,如图 6(a)中虚线部分所示.构件 TrainCom 建模列车的通信与 RBC 通信模块类似,同样包含信息处理延迟和发送信息的丢失率.

该系统中具有的连续行为主要体现在构件 TrainPlant(如图 6(c)所示),建模列车制动装置的操作功能,用于控制列车运行的最大速度和实现列车制动功能.系统初始化后处于 Normal 状态,并且在未达到最大速度 vMax 前处于加速状态,加速度为 *a*,当达到最大速度后进行匀速行驶;当收到控制器的制动请求 dec 时,列车进入制动模式,有固定 1s 的控制延迟,之后进入制动状态,且控制失败率以 failRate 表示.在成功进入制动 Braking 状态后,以–*b* 的加速度减速,由于列车的实际运行环境可能受天气、冰雪路面条件、最小粘着力和最大顺风等因素的影响,列车的制动加速度(–*b*)有一定的不确定性,模型中以 *b*[*id*]=1–*random*(0.4)表示.需要注意的是,当列车处于

Braking 状态时,不变式“$EoA[id]'==0$ && $a[id]'==0$ && $b[id]'==0$ && $v[id]'==-b[id]$ && $pos[id]'==v[id]$”表示列车处于刹车状态时列车速度的变化情况,如,$v[id]'==-b[id]$表示单位时间内速度的变化率为$-b[id]$.模型中,通过该方式建模随机混成系统的连续行为.进入 Braking 状态后,列车会一直减速直到停止.这里,采用近似于“硬撞墙”的模式[26]:当两车间隔小于或等于安全距离 safeLen 时,当前列车认为前行列车处于停车状态,即,“硬墙”.

构件 TrainController 建模列车控制器,用于控制通信消息的发送并在必要时候发出制动请求.控制器开始处于 Counting 计数状态,每隔 0.75s 要求通信模块向 RBC 发送一次消息,若连续 20 次都未收到 RBC 返回的消息,则自动发出制动请求以保护列车安全;另外,当列车的当前距离与 EoA 点的距离小于或等于安全距离时,也发出制动请求.

2.2 模型验证与分析

为了最大限度地模拟真实场景,我们的部分实验参数设置遵循 IEEE 1474.1TM 的标准规范,如列车位置测量误差、通信消息延迟等;部分参数,如列车的最大速度、安全制动距离、制动加速度等则参考了现行的高速列车系统,具体取值见表 1.此外,我们考虑开放环境下的不确定性因素给列车带来的影响,如因环境造成的制动加速度偏差值等.

Table 1 Parameters of CBTC model

表 1 CBTC 模型的参数

参数名称	参数值
通信消息延时的概率密度	Exp(0.8)
位置测量误差	10m
列车起始间距	4 000m
安全距离	3 000m
列车起始速度	45m/s
列车最大速度	84m/s
列车加速度	0.9m/s
列车制动加速度	−1m/s
因环境造成的制动加速度偏差值	+/-0.4m/s
通信消息丢包率	0.1%
列车制动控制失败率	0.00001%

实验分析结果表明,上述参数设置并不绝对安全.即,存在较小的几率发生碰撞.为了进一步评估碰撞发生的概率区间,我们评估在 200s 内 Train1 超过前方 Train0 的概率小于或等于θ,θ初值取 0.2,用 PBLTL 表示为

$$P_{\leqslant\theta}(F^{200}(pos[1]\geqslant pos[0])).$$

实验采用了本文提出的自适应统计算法,在评估过程中,当θ=0.09 时,SPRT 算法所需样本数量会出现一个峰值;当θ=0.023 时开始执行 BIET 算法,之后,所需样本数稳步上升,最终得到如图 7 所示的碰撞发生的累积概率分布图(BIET 算法参数如下:区间覆盖系数 c=0.9,半区间大小 k=0.00005;总样本数为 182 754,完整样本数为 26 413,负样本数为 34;最终结果的概率区间为[0.00014148,0.00024148],平均概率值为 0.000 191 48).

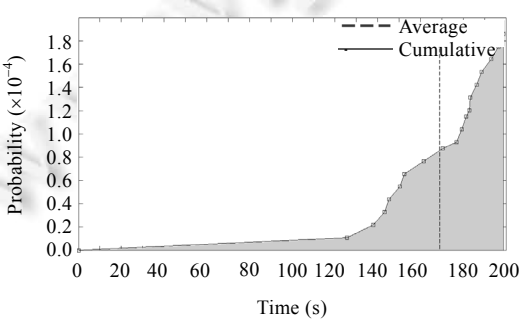


Fig.7 Cumulative probability distribution plot for collision

图 7 碰撞发生的累积概率分布图

由图 7 可知:碰撞的概率在前 120s 内极小;[140s,200s]时间段为两车碰撞的高发区间,且在[145s,155s]与[180s,190s]两段时间的碰撞概率最大.需要注意的是,本文针对的是非线性随机混成自动机,而基于模拟仿真的方法(如本文的方法)在处理连续时间时可能存在一定的误差.一方面由于仿真算法存在误差,验证和评估精度依赖于仿真算法;另一方面,由于引入 SVM,其预测也会存在误报问题.因此,如何控制误差是亟待解决的问题之一.

在现有的 SMC 算法中,当概率接近 0 或 1 时,BIET 算法比其他几种更为有效[22,23].实验结果表证明,仅利用 BIET 算法所需时间仍然较长(得到如图 7 所示的结果耗时约为 15 小时).为了进一步提高算法效率,我们采用了本文提出的基于 SVM 的 SMC 算法,对同样的属性约束进行评估,记录不同参数下的生成样本数、验证时间以及内存占用峰值,得到的对比实验结果见表 2.

Table 2 Performance comparison between SVM-based SMC and classical SMC

表 2 基于 SVM 的 SMC 与传统 SMC 的性能比较

		基于 SVM 的 SMC 算法		BIET 算法	
		完整样本数/验证时间	内存占用峰值(KB)	样本数/验证时间	内存占用峰值(KB)
BIET 算法参数:区间覆盖系数 c;半区间大小 k	c=0.9 k=0.0005	2 251/4 915s	82 680	2 251/4 833s	41 680
	c=0.9 k=0.00005	26 413/8.1h	172 400	182 754/15h	41 860
	c=0.99 k=0.0005	5 365/3.5h	135 800	6 367/4.3h	41 880
	c=0.99 k=0.00005	33 178/27.8h	181 700	---/>>60h	41 680

运行实验的硬件环境如下:CPU:Intel Core i3-3320,3.3GHz,4 核;内存 3.8GB;操作系统 ubuntu13.10,64 位.

表 2 中需要指出的是:基于 SVM 的 SMC 算法并不会直接减少样本 Trace 的总个数,而是对于大部分可信样本 Trace,模拟一半即可停止模拟并判定结果,从而减少模拟所需的总时间;少数不可信样本 Trace 会进入后段仿真器生成完整 Trace,这类样本称为完整样本.经过实验统计,在 BIET 算法中,超过 99%的时间都用于模拟生成 Trace,统计分析部分计算的时间只有 1s 左右[23];在本文基于 SVM 的 SMC 算法中,SVM 训练和预测过程所需时间约占总时间的 0.3%.可见,统计分析和 SVM 训练、预测两者所需时间远小于模拟生成样本 Trace 所需时间,改进方法可行.

下面将从时间和空间两个方面对两种算法进行分析:

(1) 所需样本数和验证时间

- 当 c=0.9,k=0.0005 时,完整样本数与 BIET 所需样本数相等,原因是模拟的样本 Trace 较少,在 2 251 个样本中并不含负样本(碰撞发生的样本),导致 SVM 无法发挥作用;而验证时间稍有增加,是由保存一定量的 Trace 等额外操作形成的.
- 当 c=0.9,k=0.00005 时,由于接近了真实的概率值,利用传统的 BIET 算法所需样本数和验证时间都急剧增加,此时 SVM 发挥作用,完整样本数为 26 413 个,大约只占 BIET 算法所需样本数的 14.45%,即,85.55%的样本并不需要完整模拟即可判定结果,最终时间大约只占 BIET 算法的一半.
- 同样地,当 c=0.99 时也可以得到类似的结论.

(2) 内存占用峰值

基于 SVM 的 SMC 算法在内存占用上会稍多于传统的 BIET 算法,原因是算法过程初期需要保存一定量的 Trace 以供 SVM 训练,而通常训练样本的数量并不会随着验证所需样本数的增加而增加.从表 2 中也可知,内存占用大约稳定于 170 000KB~180 000KB,因此,内存占用并不会成为本算法的主要瓶颈.

3 相关工作

近年来,使用模型检测技术验证、分析安全攸关系统的安全性、可靠性的研究工作取得了较大的进展,人

们关注的重点是如何使用传统的模型检测技术对系统模型进行验证分析,检测系统模型是否满足安全性属性约束,从而提高系统模型的安全性.然而,对安全攸关系统中小概率事件的研究工作较少,据我们所知,目前对于定量评估、预测小概率事件发生的概率的研究工作还处于起步阶段,还没有形成一套完整的建模、分析方法.特别地,如何使用统计模型检测技术预测、评估小概率事件发生的概率的研究工作较少.这里,我们主要针对此方面的研究进行相关工作的比较.

用于模拟小概率事件比较成熟的技术是重要性取样(importance sampling)[27],可减少系统模型所对应的样本空间,能够比较高效地捕获小概率事件发生的样本 Trace.借助此方法的优势,美国 CMU 的 Edmund M. Clarke 教授领导的团队提出使用统计模型检测技术验证、分析随机混成系统的性能,并重点围绕如何使用重要性取样技术减少样本空间数 N,解决小概率事件的统计分析问题[3],旨在提高统计模型检测的效率.但是,如何在不考虑整个状态空间的前提下找到合适的重要性取样分布是个难点.可能的解决方法之一是,使用交叉熵(cross-entropy)寻找重要性取样分布的优化参数[28].

此外,Gong 等人将 SMC 技术应用于生物系统的分析,用于检测胰腺癌细胞存在的概率[29],这也是使用统计模型检测技术预测、评估小概率事件发生概率的一个典型应用.张德平等人[30]提出了软件可靠性评估的重要抽样方法,将抽样技术用于统计测试,提出了软件可靠性估计的最优测试剖面生成的启发式迭代算法,以提高统计测试的速度.Zimmermann 等人提出使用扩展的 UML 状态图建模 ETCS 系统中的避碰模块,将状态图自动转化为 Petri 网模型,并借助相应的 Petri 网工具分析系统的性能;此外,还提出了小概率事件的模拟方法,以实现系统的可靠性度量[31].

我们与丹麦奥尔堡大学 Larsen 教授领导的团队合作,提出了一种基于 SHA 的随机混成系统建模与验证方法,并在验证工具 UPPAAL-SMC[12]中实现了基于 SHA 的统计模型验证算法,以定量地评估混成系统的性能.该方法已经成功地应用于工业级的案例,如能耗感知的智能家居(energy-aware smart building)[9]、系统生物、实时调度.但目前为止,该工具只实现了 SPRT 算法,统计分析的精确度不高,对于小概率事件则基本无法预测.本文基于我们前期的工作,针对小概率事件改进了统计模型检测器,旨在评估、预测安全攸关系统中小概率事件发生的概率.

4 总 结

安全攸关系统被广泛应用于国防、工控、交通、电子医疗等安全攸关领域,其系统行为通常具有随机混成系统的特性.评估、预测安全攸关系统中小概率事件的发生,对于提高系统的安全性、可靠性具有重要意义.然而,如何评估、预测这类随机混成系统中小概率事件的发生概率,是目前面临的一个难题.针对该问题,我们提出面向小概率事件的统计模型检测框架,结合机器学习及统计模型检测技术,针对小概率事件,设计、优化统计模型检测器,并设计、实现了两种关键算法:一种算法是对样本 Trace 进行学习,提取样本特征,用于减少统计分析所需要的样本空间,以提高统计模型检测的效率;另一种是自适应统计模型检测算法,能够根据概率评估的范围和精度,自适应地选取统计模型检测算法.结合轨道交通控制系统中避碰案例的分析,构建了该系统的随机混成自动机模型,并使用基于机器学习的统计模型检测方法评估、预测小概率事件,即,列车撞车发生的概率区间.初步的实验结果表明,本文所提出的方法具有可行性.下一步的工作重点是继续改进和优化基于机器学习的 SMC 算法,并通过预测小概率事件发生的概率,分析导致其发生的原因,构建安全属性模型.

致谢 我们诚挚地感谢审稿专家所给出的宝贵意见与建议.同时,我们也非常感谢在本文写作过程中,华东师范大学软件学院的陈小红、孙海英老师在 CBTC 列车控制系统模型分析中所给予的建议及帮助.

References:

[1] Mitra S, Wongpiromsarn T, Murray RM. Verifying cyber-physical interactions in safety-critical systems. Security & Privacy, 2013, 11(4):28−37. [doi: 10.1109/MSP.2013.77]

[2] Michael L, Frank O. Teaching and training formal methods for safety critical systems. In: Proc. of the 39th Euromicro Conf. on Software Engineering and Advanced Applications (SEAA). IEEE, 2013. 408−413. [doi: 10.1109/SEAA.2013.54]

[3] Zuliani P, Baier C, Clarke EM. Rare-Event verification for stochastic hybrid systems. In: Proc. of the 15th ACM Int'l Conf. on Hybrid Systems: Computation and Control. ACM Press, 2012. 217−226. [doi: 10.1145/2185632.2185665]

[4] Clarke EM, Klieber W, Nováček M, Zuliani P. Model checking and the state explosion problem. In: Meyer B, Nordio M, eds. Proc. of the LASER 2012. LNCS 7682, Heidelberg: Springer-Verlag, 2012. 1−30. [doi: 10.1007/978-3-642-35746-6_1]

[5] Clarke EM, Zuliani P. Statistical model checking for cyber-physical systems In: Proc. of the Automated Technology for Verification and Analysis. Heidelberg: Springer-Verlag, 2011. 1−12. [doi: 10.1007/978-3-642-24372-1_1]

[6] David A, Larsen KG, Legay A, Mikučionis M. Schedulability of herschel-planck revisited using statistical model checking. In: Leveraging Applications of Formal Methods, Verification and Validation. Applications and Case Studies. Berlin, Heidelberg: Springer-Verlag, 2012. 293−307. [doi: 10.1007/978-3-642-34032-1_28]

[7] Jha SK, Clarke EM, Langmead CJ, Legay A, Platzer A, Zuliani P. A bayesian approach to model checking biological systems. In: Proc. of the Computational Methods in Systems Biology. Heidelberg: Springer-Verlag, 2009. 218−234. [doi: 10.1007/978-3-642-03845-7_15]

[8] Miskov-Zivanov N, Zuliani P, Clarke EM, Faeder JR. Studies of biological networks with statistical model checking: application to immune system cells. In: Proc. of the Int'l Conf. on Bioinformatics, Computational Biology and Biomedical Informatics. ACM Press, 2013. 728−729. [doi: 10.1145/2506583.2512390]

[9] David A, Du DH, Larsen KG, Mikučionis M, Skou A. An evaluation framework for energy aware buildings using statistical model checking. Science China Information Sciences, 2012,55(12):2694−2707. [doi: 10.1007/s11432-012-4742-0]

[10] Legay A, Delahaye B, Bensalem S. Statistical model checking: An overview. In: Proc. of the Runtime Verification. Heidelberg: Springer-Verlag, 2010. 122−135. [doi: 10.1007/978-3-642-16612-9_11]

[11] David A, Du D, Larsen KG, Legay A, Mikučionis M, Poulsen DB, Sedwards S. Statistical model checking for stochastic hybrid systems. arXiv preprint arXiv:1208.3856, 2012. [doi: 10.4204/EPTCS.92.9]

[12] Bulychev P, David A, Larsen KG, Mikučionis M, Poulsen DB, Legay A, Wang Z. UPPAAL-SMC: Statistical model checking for priced timed automata. arXiv preprint arXiv:1207.1272, 2012. [doi: 10.4204/EPTCS.85.1]

[13] David A, Larsen KG, Legay A, Mikučionis M, Poulsen DB, Van Vliet J, Wang Z. Statistical model checking for networks of priced timed automata. In: Proc. of the Formal Modeling and Analysis of Timed Systems. Heidelberg: Springer-Verlag, 2011. 80−96. [doi: 10.1007/978-3-642-24310-3_7]

[14] Villén-Altamirano M, Villén-Altamirano J. RESTART: A straightforward method for fast simulation of rare events. In: Proc. of the Simulation Conf. IEEE, 1994. 282−289. [doi: 10.1109/WSC.1994.717150]

[15] Cortes C, Vapnik V. Support-Vector networks. Machine Learning, 1995,20(3):273−297. [doi: 10.1007/BF00994018]

[16] Press WH. Numerical Recipes 3rd ed., The Art of Scientific Computing. Cambridge: Cambridge University Press, 2007.

[17] Li R, Ye SW, Shi ZZ. SVM-KNN classifier—A new method of improving the accuracy of SVM classifier. Acta Electronica Sinica, 2002,30(5):745−748 (in Chinese with English abstract). [doi: 10.3321/j.issn:0372-2112.2002.05.035]

[18] Ling P, Zhou CG. For SVM: Confidence online evaluation & decision improvement. Journal of Frontiers of Computer Science & Technology, 2008,2(2):192−197 (in Chinese with English abstract). [doi: 10.3778/j.issn.1673-9418.2008.02.007]

[19] Diehl CP, Cauwenberghs G. SVM incremental learning, adaptation and optimization. Proc. of the Int'l Joint Conf. on IEEE, 2003,4: 2685−2690. [doi: 10.1109/IJCNN.2003.1223991]

[20] Qian HB, He GN. A survey of class-imbalanced data classification. Computer Engineering and Science, 2010,32(5):85−88 (in Chinese with English abstract). [doi: 10.3969/j.issn.1007-130X.2010.05.023]

[21] Chang CC, Lin CJ. LIBSVM: A library for support vector machines. ACM Trans. on Intelligent Systems and Technology, 2011,2 (3):1−27. [doi: 10.1145/1961189.1961199]

[22] Kim Y, Kim M, Kim TH. Statistical model checking for safety critical hybrid systems: An empirical evaluation. In: Armin B, Amir N, Tanja V, eds. Proc. of the Hardware and Software: Verification and Testing. Heidelberg: Springer-Verlag, 2013. 162−177. [doi: 10.1007/978-3-642-39611-3_18]

[23] Zuliani P, Platzer A, Clarke EM. Bayesian statistical model checking with application to Stateflow/Simulink verification. Formal Methods in System Design, 2013,43(2):338−367. [doi: 10.1007/s10703-013-0195-3]

[24] Zhu L, Yu FR, Ning B, Tang, T. Communication-Based train control (CBTC) systems with cooperative relaying: Design and performance analysis. IEEE Trans. on Vehicular Technology, 2013,63(5):2162−2172. [doi: 10.1109/TVT.2013.2291533]

[25] Damm W, Mikschl A, Oehlerking J, Olderog ER, Pang J, Platzer A, Wirtz B. Automating verification of cooperation, control, and design in traffic applications. In: Proc. of the Formal Methods and Hybrid Real-Time Systems. Heidelberg: Springer-Verlag, 2007. 115−169. [doi: 10.1007/978-3-540-75221-9_6]

[26] Zhang YP, Sun YS, Zhao B. Research on train interval control under moving block. Highlights of Sciencepaper Online, 2013,6(12): 1112−1118 (in Chinese with English abstract).

[27] Srinivasan R. Importance Sampling: Applications in Communications and Detection. Heidelberg: Springer-Verlag, 2002.

[28] Jegourel C, Legay A, Sedwards S. Cross-Entropy optimisation of importance sampling parameters for statistical model checking. In: Proc. of the Computer Aided Verification. Heidelberg: Springer-Verlag, 2012. 327−342. [doi: 10.1007/978-3-642-31424-7_26]

[29] Gong H, Wu TT, Clarke EM. Pathway-Gene identification for pancreatic cancer survival via doubly regularized Cox regression. BMC Systems Biology, 2014,8(1):1−9. [doi: 10.1186/1752-0509-8-S1-S3]

[30] Zhang DP, Nie CH, Xu BW. Importance sampling method of software reliability estimation. Ruan Jian Xue Bao/Journal of Software, 2009,20(10):2859−2866 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/3553.htm [doi: 10.3724/SP.J.1001.2009.03553]

[31] Zimmermann A, Trowitzsch J. Reliability evaluation of distributed embedded systems with UML state charts and rare event simulation. In: Proc. of the MBEES. 2009. 128−139.

附中文参考文献:

[17] 李蓉,叶世伟,史忠植.SVM-KNN 分类器——一种提高 SVM 分类精度的新方法.电子学报,2002,(5):745−748. [doi: 10.3321/j.issn:0372-2112.2002.05.035]

[18] 凌萍,周春光.SVM 置信度在线评估以及决策改进.计算机科学与探索,2008,2(2):192−197. [doi: 10.3778/j.issn.1673-9418.2008.02.007]

[20] 钱洪波,贺广南.非平衡类数据分类概述.计算机工程与科学,2010,32(5):85−88. [doi: 10.3969/j.issn.1007-130X.2010.05.023]

[26] 张友鹏,孙永生,赵斌.移动闭塞列车运行间隔控制研究.中国科技论文在线精品论文,2013,6(12):1112−1118.

[30] 张德平,聂长海,徐宝文.软件可靠性评估的重要抽样方法.软件学报,2009,20(10):2859−2866. http://www.jos.org.cn/1000-9825/3553.htm [doi: 10.3724/SP.J.1001.2009.03553]

杜德慧(1979－),女,河南信阳人,博士,副教授,CCF 会员,主要研究领域为可信软件,模型检测.

程贝(1988－),男,硕士生,主要研究领域为可信软件,模型检测.

刘静(1964－),女,博士,教授,博士生导师,主要研究领域为模型驱动软件开发,可信软件.