基于随机时间 Petri 网的安全性分析方法

彭 颖[1] 姚淑珍[1] 谭火彬[2]
(北京航空航天大学计算机学院 北京 100191)[1] (北京航空航天大学软件学院 北京 100191)[2]

摘 要 在分析了现有的 Petri 网与安全性结合的方法的缺陷后,提出了一种基于随机时间 Petri 网(stochastic Time Petri Nets,sTPN)的系统安全性分析方法,利用 sTPN 建立的系统模型不局限于指数分布和确定分布的变迁,也不局限于一般分布的变迁的使能限制。通过修改后的瞬态随机状态类图以及 sTPN 的瞬态分析算法可以得到基于路径的安全性指标。最后给出核反应堆冷却循环系统的例子,说明了所提方法的可用性和合理性。

关键词 随机时间 Petri 网,安全性分析,瞬态分析

中图法分类号 TP301 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2016.11.011

Safety Analysis Method Based on Stochastic Time Petri Nets

PENG Ying[1] YAO Shu-zhen[1] TAN Huo-bin[2]
(School of Computer Science and Engineering, Beihang University, Beijing 100191, China)[1]
(School of Software, Beihang University, Beijing 100191, China)[2]

Abstract After analyzing the shortage of current methods combining safety analysis with Petri net, a system safety analysis method based on stochastic time Petri nets(sTPN) was proposed. System model built by sTPN is neither limited to exponential and deterministic transitions nor to enabling restrictions for generally distributed transitions. Safety metrics based on path can be obtained through modified transient stochastic state classes graph and transient analysis algorithm of sTPN. Experimental results are reported to show the usability and reasonability of the method.

Keywords Stochastic time Petri nets, Safety analysis, Transient analysis

1 引言

安全性(safety)是指系统避免人员伤亡、职业病、设备或财产损坏或丧失、环境损坏的能力[1]。安全攸关系统(Safety-Critical System,SCS)[2]泛指具有潜在破坏力的一类系统。此类系统一旦失效,就能够造成严重的后果,比如人员伤亡、财产损失或环境破坏等。为此,在安全攸关系统开发的早期,特别是设计期间就通过对影响安全性的关键特征进行建模、分析和评估,提供一套自动的、有效的安全性验证方法很有意义。在系统实际投入使用之前评估其安全性,针对可能的问题给出相应的对策,能够在很大程度上避免系统失效。

Petri 网的并发性、异步性和不确定性具有很强的动态建模和分析能力,适合在设计期间对系统建模以及进行安全性分析。

N. G. Leveson 最早介绍了利用基本 Petri 网进行安全性分析的方法[3],其主要步骤是:定义系统的危险状态并评估其严重程度,将危险状态分为高风险状态和低风险状态,安全性分析的目标就是分析高风险状态是否可达;根据可达图,由高风险状态往前推找到关键状态;最后给出相应的对策。文献[4]在上述方法的基础上引入 S-不变量等式辅助高风险状态可达性的分析。利用基本 Petri 网进行安全性分析的局限在于只能进行定性分析。基于时间 Petri 网模型的安全性分析[5]主要依靠仿真得到在规定时间内完成一系列变迁到达目标状态(否则系统将处于高风险状态)的概率,仿真无法考察系统的所有可能情况;并且由于时间 Petri 网只给定了变迁实施延时的区间,但并未指明区间内的概率分布,得到的仿真结果的准确性依赖于延时分布假设的准确性。基于随机 Petri 网的安全性分析方法[6],通过计算 GSPN 或 DSPN 的稳定状态概率,得到系统在长时间运行时到达高风险状态的概率。利用着色 Petri 网进行安全性分析的方法与基本 Petri 网类似[7]。以上方法中只有基于随机 Petri 网的安全性方法由于给每个变迁关联了服从一定概率分布的实施时延,因此可应用于安全性定量分析。但是随机 Petri 网只包含瞬时变迁和时延能服从指数分布和确定分布的变迁,大大限制了所能建模的系统的范围。

本文提出了一种基于随机时间 Petri 网的安全性分析方法,由 sTPN 建模的系统所包含的变迁不局限于指数分布和确定分布,概率分布函数为 e 的多项式(expolynomial)形式的

到稿日期:2015-07-18 返修日期:2016-01-11 本文受航空科学基金项目(2013ZC51023)资助。

彭 颖(1991-),女,硕士生,主要研究方向为软件工程;姚淑珍(1965-),女,博士,教授,CCF 会员,主要研究方向为软件工程、Petri 网理论与应用;谭火彬(1979-),男,博士,讲师,主要研究方向为软件工程。

变迁均有闭合解，并且不局限于一般分布的变迁的使能限制。本文将详细阐述如何利用系统的 sTPN 模型得到基于路径的安全性指标。

2 sTPN

2.1 sTPN 定义

一个 sTPN 由一个十元组构成：$sTPN=\langle P,T,A^-,A^+,A^\cdot,m_0,EFT,LFT,F,C\rangle$。与时间 Petri 网一样，$P$ 是库所的集合，T 是变迁的集合，$A^-\subseteq P\times T$，$A^+\subseteq T\times P$，$A^\cdot\subseteq P\times T$ 分别代表着前置条件、后置条件和抑制条件的集合；m_0 是初始标识。$EFT:T\to R_0^+$ 和 $LFT:T\to R_0^+\cup\{+\infty\}$ 表示每个变迁的最小实施时延和最大实施时延。F 代表着每个变迁的实施时延在 $[EFT,LFT]$ 区间内的概率分布函数，C 是变迁的权重的集合。

sTPN 的状态表示为 $s=\langle m,\tau\rangle$，其中 $m:P\to N$ 表示当前标识，$\tau:T\to R_0^+$ 表示每个变迁的实施时延。一个变迁 t_0 是使能的(enabled)当且仅当它的所有输入库所都包含至少一个托肯并且它的所有抑制库所都不包含托肯，一个变迁 t_0 是可实施的(firable) 当且仅当它是使能的并且它的实施时延 $\tau(t_0)$ 不高于其他使能的变迁。当多个变迁可实施时，利用权值 C 决定哪个变迁发生。

$$Prob\{\text{选择 } t_0\}=\frac{C(t_0)}{\sum_{t_i\in T^f(s)}C(t_i)}$$

其中，$T^f(s)$ 表示系统处于状态 s 时可实施的变迁的集合。

当一个变迁 t_0 实施后，状态 $s=\langle m,\tau\rangle$ 变为 $s'=\langle m',\tau'\rangle$，标识 m 到标识 m' 的变化过程和普通 Petri 网一样：

$$m_{tmp}(p)=\begin{cases}m(p)-1, & \text{if } \langle p,t_0\rangle\in A^-\\ m(p), & \text{otherwise}\end{cases}$$

$$m'(p)=\begin{cases}m_{tmp}(p)+1, & \text{if } \langle t_0,p\rangle\in A^+\\ m(p)+1, & \text{otherwise}\end{cases}$$

在中间标识 m_{tmp} 和最终标识 m' 中都使能的变迁称为持续使能(persistent)，而那些在 m' 中使能但在 m_{tmp} 中不使能的变迁称为新使能(newly enabled)。如同时间 Petri 网中的定义，如果变迁 t_0 在它自己实施后依然使能则认为它是新使能的。

在 t_0 实施后，每个持续使能的变迁的实施时延将会减去在之前状态中逗留的时间，即 $\tau'(t_i)=\tau(t_i)-\tau(t_0)$，而对于新使能的变迁 t_a，它的实施时延是根据函数 $F_{t_a}()$ 从时延区间中采样获得的，即

$$EFT(t_a)\leqslant\tau'(t_a)\leqslant LFT(t_a)$$

$$Prob\{\tau'(t_a)\leqslant x\}=F_{t_a}(x)$$

当 $EFT(t)=LFT(t)=0$ 时，变迁 t 被称为瞬时变迁，其他变迁被称为时间变迁，其中时间变迁若不满足在 $[0,\infty)$ 区间中服从指数分布，则称为服从一般分布(Generally Distributed，GEN)的变迁。确定变迁是 GEN 变迁的一个特例，$EFT(t)=LFT(t)>0$。

由 sTPN 的定义可知它包含了瞬时变迁、指数变迁和确定变迁以及其他一般分布的变迁，因此 sTPN 所能描述的系统的范围大于 GSPN 和 DSPN。

2.2 sTPN 等价的随机过程

由于包含了 GEN 变迁，sTPN 的标识变化等价的随机过程是有时间记忆的，属于广义半马尔可夫过程(generalized semi-Markov process)。下面阐述广义半马尔可夫过程与半马尔可夫过程以及马尔可夫再生过程的区别。

假如所有的 GEN 变迁在每一次变迁发生后都不持续使能，那么此随机过程在每一步再生，它属于半马尔可夫过程的范畴。

假如模型最终会到达一个再生点，则其随机过程属于马尔可夫再生过程。马尔可夫再生过程的一种特殊情况是模型满足使能限制(enabling restriction)，即每个标识至多只有一个 GEN 变迁使能，此时它包含一系列嵌入的连续时间马尔可夫链，对于这类模型，可以采用马尔可夫再生理论或者补充变量的方法来分析。

假如模型有可能不会到达再生点，则其随机过程属于广义半马尔可夫过程。

在多个 GEN 变迁使能的情况下，它们的变迁实施时延是相互依赖的变量，这些变量的联合分布是一个差分界限矩阵(Difference Bounds Matrix，DBM)[8]，也就是说这些变迁的实施时延分布在一个以一系列线性不等式表示的区域内，如下式所示：

$$D=\begin{cases}\tau_i-\tau_j\leqslant b_{ij}\\ \tau_*=0\\ \forall i\neq j\in[0,N-1]\cup\{*,age\}\end{cases}$$

其中，$b_{ij}\in R\cup\{+\infty\}$。一个非空的 DBM 区域有唯一的标准形式，使得 b_{ij} 与 $\tau_i-\tau_j$ 所能表示的最大值一致。是否为标准形式可以用三角不等式鉴别：$b_{ij}\leqslant b_{ih}+b_{hj}$，$\forall i,j,h\in\{*,0,1,\cdots,N-1\}$ 并且 $i\neq j,i\neq h,h\neq j$。

DBM 的标准形式可以利用最短路径问题的解法在多项式时间内获得。

2.3 瞬态随机状态类图

DBM 广泛应用于时间 Petri 网的定性分析中[9]，而随机状态类(stochastic state classes)通过给每个状态类赋予密度函数扩展了 DBM，使其应用于定量分析。扩展后，基于 DBM 的状态类可达图变为随机状态类可达图(简称随机状态类图)，其中包含了状态类之间的变迁发生的概率以及每个状态类内部变迁实施时间的多元分布[10]。瞬态随机状态类(transient stochastic state classes)在随机状态类的基础上又加入了辅助变量 age 来表示从初始状态开始已经过的时间的负数，age 初始设为 0，每个变迁发生后它的值都会减少。

定义 1 一个状态类(state class)是一个二元组 $S=\langle m,D\rangle$，其中 m 是一个标识，D 是指 m 标识下所有使能的变迁的实施时延的向量 $\tau=\langle\tau_0,\tau_1,\cdots,\tau_{N-1}\rangle$，其中 τ_i 表示变迁 t_i 的实施时延。

定义 2 一个瞬态随机状态类是一个三元组 $\langle m,D,f_{\langle\tau_{age},\tau\rangle}()\rangle$，其中 m 是一个标识，$\langle\tau_{age},\tau\rangle$ 称为时钟向量的变量，其中 τ_{age} 是代表已经过时间的负值的标量，向量 $\tau=\langle\tau_0,\tau_1,\cdots,\tau_{N-1}\rangle$ 代表 m 标识下所有使能的变迁的实施时延，$f_{\langle\tau_{age},\tau\rangle}()$ 代表 $\langle\tau_{age},\tau\rangle$ 在 D 上的概率密度函数。

定义 3 给定两个瞬态随机状态类，$\Sigma=\langle m,D,f_{\langle\tau_{age},\tau\rangle}$

()〉和 $\Sigma'=\langle m', D', f'_{\langle\tau'_{age},\tau'\rangle}()\rangle$。我们说 Σ' 是 Σ 经过 t_0 的后续，写作 $\Sigma \overset{t_0}{\Rightarrow} \Sigma'$ 当且仅当满足下列条件：如果网的标识是 m，时钟向量是随机变量 $\langle\tau_{age},\tau\rangle$，它在区域 D 上的概率密度是 $f_{\langle\tau_{age},\tau\rangle}()$，并且 t_0 有不为零的概率发生，当它发生后，模型到达一个新的标识 m' 和一个新的时钟向量 $\langle\tau'_{age},\tau'\rangle$，它在区域 D' 上的概率密度是 $f'_{\langle\tau'_{age},\tau'\rangle}()$。

枚举后续关系 $\overset{t_0}{\Rightarrow}$ 需要识别流出事件和得到后续状态类。变迁 $t_0 \in T$ 是 $\Sigma=\langle m, D, f\rangle$ 的一个流出事件当且仅当在标识 m 下 t_0 使能，并且它的实施时延不高于其他使能变迁的概率，μ_0 不为零：

$$\mu_0 = \int_{D\cap\{x_{age},x|x_0\leqslant x_n,\forall t_n\in T(m)\}} f(x_{age},x)\,\mathrm{d}x_{age}\,\mathrm{d}x > 0$$

m' 是通过 Petri 网的标准方式并通过移出托肯得到的，D' 是通过 DBM 空间的符号状态空间分析计算 D 的后继得到的[9]。状态密度函数 f' 的计算步骤详见文献[11]。

可以证明如果状态类图是有限的，并且它包含的每一个环都至少经过一条代表最早实施时延不为零的变迁（即 $EFT(t)>0$）的边，那么，对于任何时间限制 T，瞬态随机状态类图包含的满足 $(-\tau_{age})\leqslant T$ 的状态的数量是有限的[11]。

3 基于 sTPN 的安全性分析方法

现有的结合 Petri 网与安全性分析的方法大多分析高风险状态的可达性。随机 Petri 网可用于安全性定量分析，即分析高风险状态的稳定状态概率，但是随机 Petri 网所包含的时间变迁的时延只服从指数分布和确定分布。尽管包含指数分布和确定分布的模型能涵盖大量实际应用场景，但依然在很多情况下它们不足以恰当地描述系统的随机行为，并且常用的算法需要确定分布的变迁满足使能限制，即模型的每个标识至多只有一个确定分布的变迁使能，大大限制了随机 Petri 网能建模的系统的范围。

综上所述，选择 sTPN 建立系统的模型，从而能包含更广泛的变迁。基于 sTPN 的安全性分析方法的第一步是建立系统的 sTPN 模型，确定不安全状态并且评估不安全状态的严重性，即将不安全状态分为高风险状态和低风险状态，其中高风险状态代表会导致毁灭性的不可接受的损失的状态。安全性分析的目标就是评估高风险状态的概率并且给出相应对策。

事实上，与高风险状态的可达性或稳定状态下处在高风险状态的概率相比，安全性分析更希望了解一定时间范围内第一次到达高风险状态的情况，例如飞机在起飞后降落前这一段时间内发生严重故障导致不安全后果的概率。

利用时序逻辑可以很好地将这类安全性需求转化为形式化表达。我们利用 CSL 中一类表示时间限制的路径公式来表达这类安全性需求：$s \models P_{\leqslant p}(\phi_1 U^{\leqslant t} \phi_2)$，此公式等价于 $Prob\{s,\phi_1 U^{\leqslant t}\phi_2\}\leqslant p$。上述公式的意义是从状态 s 出发，模型在时间范围 $[0,t]$ 内经过每个状态都满足公式 ϕ_1 的路径第一次到达满足公式 ϕ_2 的状态的概率小于或等于 p。

关于 CSL 的详细语法和语义可参考文献[12]。

基于 sTPN 的安全性分析方法的第二步是给出基于路径的安全性需求的 CSL 公式，形如 $Prob\{s,\phi_1 U^{\leqslant t}\phi_2\}\leqslant p$。利用 CSL 公式表示安全性需求主要有 3 个优点：1）相比基于自然语言描述的安全性需求的模糊性，形式化的表示使得含义清晰明了；2）不需要人工地根据所要计算的安全性指标更改模型，一个模型就可以计算不同的指标；3）基于这类公式的不同的安全性指标可以利用同一个算法逻辑得出计算结果。

下面具体阐述如何计算此类基于路径的安全性指标。

对于 sTPN 的瞬态随机状态类图 M 和公式 φ，我们定义 $M[\varphi]$ 为修改 M 中所有满足公式 φ 的状态为吸收状态（吸收状态是指系统的某些状态，一旦进入这些状态就不再向外转移）得到的瞬态随机状态类图。

若 $\varphi=\neg\phi_1 \vee \phi_2$，则 $Prob^M\{s,\phi_1 U^{\leqslant t}\phi_2\}=Prob^{M(\varphi)}\{s,ttU^{\leqslant t}\phi_2\}$ 并且 $Prob^{M(\varphi)}\{s,ttU^{\leqslant t}\phi_2\}$ 可以转换为求时刻 t 满足 ϕ_2 的状态的瞬态概率。证明如下：对于 $\varphi=\neg\phi_1 \vee \phi_2$，若 φ 为真，则 ϕ_1 为假或者 ϕ_2 为真。所以 $M[\varphi]$ 中的吸收状态为 ϕ_1 为假或者 ϕ_2 为真的状态，这些状态不再有其他后续状态。

M 中共有 4 类路径：

（1）在时间 t 内沿着满足 ϕ_1 的状态，到达满足 $\neg\phi_1 \wedge \phi_2$ 的状态，这类路径的概率应包含在 $Prob^{M(\varphi)}\{s,ttU^{\leqslant t}\phi_2\}$ 中。

（2）在时间 t 内沿着满足 ϕ_1 的状态，到达满足 $\phi_1 \wedge \phi_2$ 的状态，这类路径的概率应包含在 $Prob^{M(\varphi)}\{s,ttU^{\leqslant t}\phi_2\}$ 中。

（3）在时间 t 内经过状态都满足 $\phi_1 \wedge \neg\varphi_2$，这类路径的概率不应包含在 $Prob^{M(\varphi)}\{s,ttU^{\leqslant t}\phi_2\}$ 中。

（4）在时间 t 内沿着满足 ϕ_1 的状态，到达满足 $\neg\phi_1 \wedge \neg\phi_2$ 的状态，这类路径的概率不应包含在 $Prob^{M(\varphi)}\{s,ttU^{\leqslant t}\phi_2\}$。

对于第一种情况，满足公式 $ttU^{\leqslant t}\phi_2$，由于不满足 ϕ_1 的状态为吸收状态，因此在时刻 t 依然处于此状态。对于第二种情况，满足公式 $ttU^{\leqslant t}\varphi_2$，由于满足 ϕ_2 的状态为吸收状态，因此在时刻 t 依然处于此状态。对于第三种情况，不满足公式 $ttU^{\leqslant t}\phi_2$。对于第四种情况，不满足公式 $ttU^{\leqslant t}\phi_2$，由于不满足 ϕ_1 的状态为吸收状态，因此在时刻 t 依然处于此状态，即一直未到达满足 ϕ_2 的状态。所以，$Prob^M\{s,\phi_1 U^{\leqslant t}\phi_2\}=Prob^{M(\varphi)}\{s,ttU^{\leqslant t}\phi_2\}$。并且对于前两种情况，时刻 t 时的状态满足 ϕ_2；对于后两种情况，时刻 t 时的状态不满足 ϕ_2。因此问题转换为求解在时刻 t 满足 ϕ_2 的状态的瞬态概率。

sTPN 的瞬态概率分析方法详见文献[11]。当变迁的实施时延的概率密度函数 $f_t()$ 是一个 e 的多项式函数时，即 $f_t(y)=\sum_{k=1}^{K} c_k y^k \mathrm{e}^{-\lambda_k y}$，可达的瞬态随机状态类的密度函数有闭合解，可以利用 Sirio 框架[13]有效地进行瞬态分析。易知，上述函数包含了指数分布和确定分布，另外还包含了常用的函数分布如均匀分布、伽马分布和爱尔朗分布等。

因此基于 sTPN 的安全性分析方法的第三步是得到 sTPN 在时刻 t 内的瞬态随机状态类图，将满足 $\neg\phi_1 \vee \phi_2$ 的状态修改为吸收状态。第四步是求解修改后的瞬态随机状态类图在时刻 t 的瞬态概率。至此，得到了基于路径的安全性指标。

值得注意的是，公式 $Prob\{s,\phi_1 U^{\leqslant t}\phi_2\}\leqslant p$ 中的 s 可以为初始状态类，也可以为再生的瞬态随机状态类（regenerative transient stochastic state class），其定义如下。

瞬态随机状态类是指状态类中所有使能的变迁要么是新

使能的，要么实施时延是指数分布或确定分布的，或者此变迁的实施时延与另一个变迁的实施时延有一个确定的延迟，而后一个变迁属于前三类。

原因是再生的瞬态随机状态类失去了时间记忆，之后状态的变化只取决于当前状态而与之前的状态无关，所以可以把它当做初始状态类，从它开始生成瞬态随机状态类图。

以其他类为初始化类 s 求解公式 $Prob\{s,\phi_1 U^{\leqslant t}\phi_2\}\leqslant p$ 的方法有待探究。

4 实验

下面给出一个安全攸关系统的例子来具体阐述基于 sTPN 的安全性分析方法。示例描述的是核反应堆冷却循环系统[14]，若系统进入不安全状态可能导致危及生命的严重后果。没有修复过程的系统自然运行的状态图如图 1 所示。

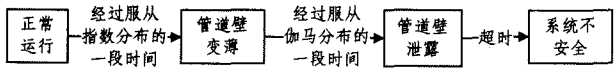

图 1　系统状态图

若没有定期地监测和修复，系统长时间的运行最终会导致管道壁变薄。如果管道壁变薄没有被发现和修复，那么管道最终会泄露。下一步，假如管道泄露没有在短时间内被发现并且及时修复，则冷却循环系统无法提供正常的冷却功能，核反应堆系统将进入不安全状态。

系统的检查子系统的流程图如图 2 所示。

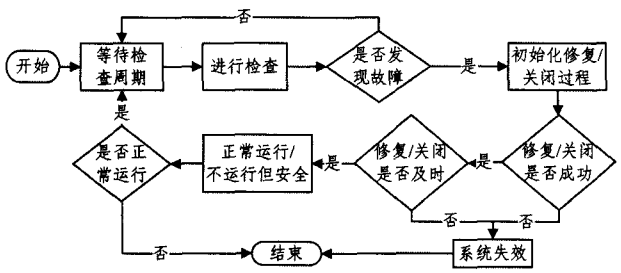

图 2　系统的检查子系统的流程图

在检查子系统中，检查定期触发，故障可能被发现也可能被漏掉。如果发现故障，系统将分析情况并决定进行修复或者执行关闭程序。修复和关闭过程并不是零风险的，可能会导致失去系统的控制，而没有得到在管道泄露时间过长之前将系统恢复到正常运行状态或将系统安全关闭的期望结果。管道的 sTPN 模型如图 3 所示。

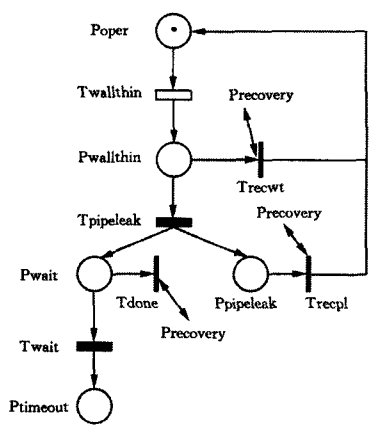

图 3　管道的 sTPN 模型

初始时刻管道处于正常运行状态，Poper 有一个托肯，经过服从指数分布的一段时间后，管道壁变薄，托肯从库所 Poper 转移到 Pwallthin。当管道壁变薄后，有两种可能的情况：第一种是检查子系统成功完成修复，Trecwt 发生，系统恢复到正常运行状态；第二种是经过服从伽马分布的一段时间后，Tpipeleak 发生，管道泄露，此时 Pwait 和 Ppipeleak 各有一个托肯。Twait 是一个确定变迁，代表管道泄露的安全时间。如果检查子系统成功完成修复早于 Twait 变迁实施，那么 Twait 被抢占，Tdone 和 Trecpl 将发生，系统恢复到正常运行状态。

图 4 是检查子系统 Petri 网模型，它描述了试图去发现、隔离、修复故障或者关闭系统的过程。

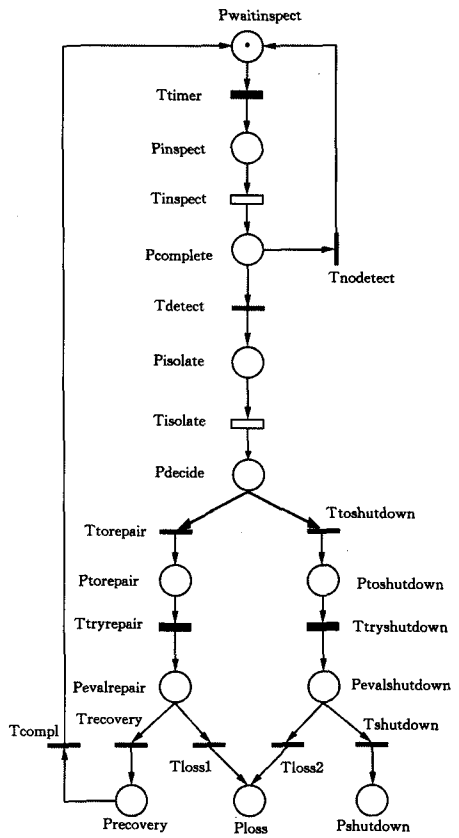

图 4　检查子系统 Petri 网模型

初始时，Pwaitinspect 有一个托肯，系统处于等待检查状态。Ttimer 是确定变迁，即系统定期触发检查过程。检查可能发现故障也可能漏掉故障，这两种情况发生的概率取决于系统的状态：正常运行、管道壁变薄还是管道泄露。如果没有发现故障，Tnodect 发生，检查系统回到等待检查状态 Pwaitinspect。如果发现故障，Tdetect 发生，托肯到达 Pisolate，系统尝试定位故障，定位完成后系统下一步决定(Pdecide)是采取修复措施还是直接关闭系统。如果决定修复，托肯转移至 Ptorepair，完成修复过程(Ttryrepair)后，托肯转移至 Pevalrepair。评估修复是否成功，若成功则托肯转移至 Precovery。Trecwt，Trecpl 和 Tdone 的优先级高于 Tcompl，因此若管道处于管道壁变薄或者管道泄露的状态，将会得到修复，管道回到正常运行状态，之后 Tcompl 发生，检查系统回到等待检查状态 Pwaitinspect；若修复失败，托肯转移至 Ploss，表示失去了对系统的控制。如果决定关闭系统，过程与上述类似，若关

闭成功则托肯转移至 Pshutdown，若关闭失败则托肯转移至 Ploss，表示失去了对系统的控制。Ploss 和 Pshutdown 都是吸收库所。

各瞬时变迁的发生概率列如表 1 所列。

表 1　瞬时变迁的发生概率

变迁名	含义	概率
Tdetect	检测到故障	0.001，如果#(Poper)为 1 0.99，如果#(Pwallthin)为 1 0.999，如果#(Ppipeleak)为 1
Tnodetect	未检测到故障	0.999，如果#(Poper)为 1 0.01，如果#(Pwallthin)为 1 0.001，如果#(Ppipeleak)为 1
Ttorepair	决定修复	0.99
Ttoshutdown	决定关闭系统	0.01
Trecovery	修复成功	0.99
Tloss1	修复失败	0.01
Tshutdown	关闭成功	0.999
Tloss2	关闭失败	0.001

各时间变迁的类型和速率参数如表 2 所列。

表 2　时间变迁的类型和速率参数

变迁名	含义	类型	参数
Twallthin	管道壁变薄	指数分布	$\lambda=10^{-4}$
Tpipeleak	管道泄露	伽马分布	$\alpha=2,\beta=10^3$
Twait	管道泄露超时	确定分布	t=20
Ttimer	触发检查流程	确定分布	t=2
Tinspect	故障检查	指数分布	$\lambda=\frac{1}{2}$
Tisolate	定位故障	指数分布	$\lambda=1$
Ttryrepair	修复故障	指数分布	$\lambda=\frac{2}{3}$
Ttryshutdown	关闭系统	指数分布	$\lambda=1$

对于此模型，不安全状态是当 Ploss 有托肯或者 Pshutdown 没有托肯而 Ptimeout 有托肯的状态，因为 Pshutdown 中有托肯代表系统已关闭，此时虽然系统停止正常运行，但处于安全状态。这两种情况均会导致灾难性后果，因此它们处于系统的高风险状态。假设需要验证的安全性需求是：

$$Prob\{s,\rightarrow(\#Pisolate=1\&\&\#Poper=1)\&\&\rightarrow(\#Pshutdown=1)U^{\leqslant 500}(\#Ptimeout=1\parallel\#Ploss=1)\}\leqslant 10^{-4} \quad (1)$$

$$Prob\{s',\rightarrow(\#Pshutdown=1)U^{\leqslant 500}(\#Ptimeout=1)\}\leqslant 10^{-4} \quad (2)$$

式(1)中 s 表示系统的初始状态，即 Poper 和 Pwaitinspect 各有一个托肯，$\#Px$ 表示库所 Px 中包含的托肯数，式(1)表示系统从初始状态开始运行，在 500 单位时间的时间限制内没有出现管道正常但因检测错误而发现故障，这种检测错误的情况最终到达高风险状态的概率。如第 3 节所述，公式中的 s 不一定为初始状态，也可以是再生点，以 s' 表示 Pwallthin 和 Pwaitinspect 各有一个托肯的状态。式(2)表示系统从管道壁变薄开始，在 500 单位时间的时间限制内到达 $(\#Ptimeout=1)$ 这个高风险状态的概率。

以式(1)为例，将可达图中满足 $\neg\phi_1\vee\phi_2$ 的状态修改为吸收状态，由于 Ploss，Pshutdown 和 Ptimeout 都是吸收库所，因此只需将满足 $(\#Pisolate=1\&\&\#Poper=1)$ 的状态修改为吸收状态即可。问题转换为求解 $t=500$ 时的瞬态概率。式(2)的计算方法类似。图 5 是式(1)的值随时间变化的趋势图，图 6 是式(2)以本文所述方法得到的计算结果以及利用 TimeNET4.3 得到的仿真结果随时间变化的趋势图。

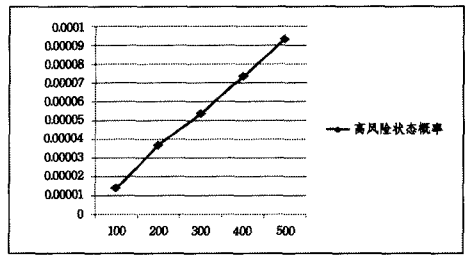

图 5　式(1)的值随时间变化的趋势图

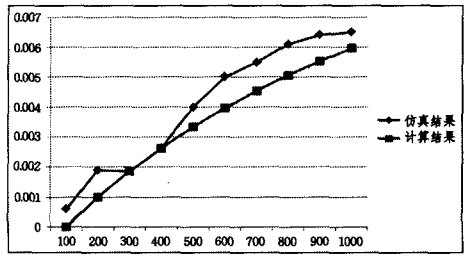

图 6　式(2)的计算结果和仿真结果随时间变化的趋势图

由计算可得以初始状态 s 开始，在 $t=500$ 时处于高风险状态的概率为 9.32×10^{-5}，满足式(1)，即系统满足所验证的安全性需求。以本文所述方法计算可得以 s'(管道壁变薄)为开始状态，在 $t=500$ 时处于 $(\#Ptimeout=1)$ 高风险状态的概率为 3.34×10^{-3}(TimeNET 的仿真结果为 3.9×10^{-3})，不满足式(2)，即不满足所验证的安全性需求。由图 6 易知，式(2)计算结果与仿真结果基本吻合，说明了基于 sTPN 的安全性分析方法的合理性。

基于 sTPN 的安全性分析方法的优点在于利用 sTPN 能有效求解实施时延的概率密度函数是 e 的多项式函数的变迁；基于 CSL 公式的安全性需求清晰明了并且不需要人工地根据所要计算的安全性指标更改模型，一个模型就可以计算不同的指标。缺点在于此方法的效率与 sTPN 的瞬态分析算法的效率紧密相连，现有的研究通过发掘 sTPN 瞬态随机状态类图中的再生点，利用马尔可夫再生过程的理论可以在很大程度上降低瞬态分析的复杂度[11]，但对于复杂系统的 sTPN 模型，瞬态分析的时间复杂度依然是一个瓶颈。

结束语　本文提出了一种基于 sTPN 的系统安全性分析方法，利用 sTPN 建立的系统模型除包含指数分布和确定分布的时间变迁外，还能包含概率密度函数是 e 的多项式函数的一般变迁，并且不局限于一般分布的使能限制。通过瞬态随机状态类图以及 sTPN 的瞬态分析算法可以得到基于路径的安全性指标，给出了安全性分析的新思路。本文进一步的工作包括基于 sTPN 的系统模型稳定状态安全度的分析，以及包含其他概率分布的变迁的模型的安全性分析方法。

参 考 文 献

[1] MIL-STD-882E. System safety[S]. Arlington: Department of Defence, 2012

[2] Storey N R. Safety critical computer systems [M]. Addison-Wesley Longman Publishing Co., Inc., 1996

[3] Leveson N G, Stolzy J L. Safety analysis using Petri nets[J]. IEEE Transactions on Software Engineering, 1987, 13(3): 386-397

(下转第 76 页)

fic Forecast Update[R]. 2014-2019 White Paper, Mar 2015

[2] Enck, William, et al. TaintDroid: an information-flow tracking system for realtime privacy monitoring on smartphones[J]. ACM Transactions on Computer Systems(TOCS), 2014, 32(2): 5-20

[3] Ries, Christian L. Dafit——a new work flow oriented approach for time efficient data preparation, validation and flagging of time series data from environmental monitoring[C]// Proceedings of the 27th Conference on Environment Informatics—Informatics for Environ. 2013: 651-656

[4] Thomas C, Gerhard K, Andrew L, et al. SAP R/3 business blueprint: understanding the business process reference model[M]. Prentice-Hall, Inc., 1997

[5] Fang Jin, Wang Tie-cheng, Shi Zhi-kuan, et al. UML workflow model[J]. Computer Engineering and Design, 2004, 25(9): 1572-1575(in Chinese)
方进, 王铁成, 石志宽, 等. 基于 UML 的工作流建模[J]. 计算机工程与设计, 2004, 25(9): 1572-1575

[6] Mike P, Bouguettaya D A, Mostafa A H, et al. Class library support for workflow environments and applications[J]. IEEE Transactions on Computers, 1997, 46(6): 673-686

[7] Aalst V, Wil M P. Structural characterizations of sound workflow nets[M]. Eindhoven University of Technology, Department of Mathematics and Computing Science, 1996

[8] Alur R, Holzmann G J, Peled D. An analyzer for message sequence charts[C]// International Workshop on Tools and Algorithms for the Construction and Analysis of Systems. Springer Berlin Heidelberg, 1996: 35-48

[9] Yang Nian-hua, Yu Hui-qun, Sun Hua, et al. Modeling UML sequence diagrams using extended Petri nets[J]. Telecommunication Systems, 2012, 51(2/3): 147-158

[10] Bo Jiang, Xiang Long, Gao Xiao-peng. Mobiletest: A tool supporting automatic black box test for software on smart mobile devices[C]// Proceedings of the Second International Workshop on Automation of Software Test. IEEE Computer Society, 2007: 8-15

[11] Stratos I, Olga P, Surajit C. Overview of Data Exploration Techniques[C]// Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data. ACM, 2015: 277-281

[12] Krishna K V, Arun K K, Kaladhar V, et al. Implementation and evaluation of scalable data structure over HBase[C]// Proceedings of the International Conference on Advances in Computing, Communications and Informatics. ACM, 2012: 1010-1018

[13] Li Jing-feng, Li Yan, Chen Ping. Formal description of the UML sequence diagram[J]. Computer Science, 2002, 29(6): 147-148(in Chinese)
李景峰, 李琰, 陈平. UML 顺序图的形式化描述[J]. 计算机科学, 2002, 29(6): 147-148

[14] Wu Zhe-hui. Petri nets introduction[M]. DynoMedia Inc., 2006(in Chinese)
吴哲辉. Petri 网导论[M]. DynoMedia Inc., 2006

[15] Coalition, Manage W. Terminology glossary[M]// Medicina Sporta. 2006: 513-524

[16] Murata, Tadao. Petri nets: Properties, analysis and applications[J]. Proceedings of the IEEE, 1989, 77(4): 541-580

[17] Yi Xiao-sheng, Jiang Hao. Time petri nets equivalent conversion workflow evaluation[J]. Computer Engineering and Design, 2006, 27(20): 3916-3919(in Chinese)
衣晓升, 姜浩. 时间 Petri 网等效变换的工作流评价[J]. 计算机工程与设计, 2006, 27(20): 3916-3919

(上接第 65 页)

[4] Sun Y, Zhang H. A software safety analysis method based on S-invariant of Petri Net[C]// 2011 9th International Conference on Reliability, Maintainability and Safety(ICRMS). IEEE, 2011: 487-492

[5] Peng Z. The safety analysis of flight landing based on Time Petri Net[C]// 2012 Proceedings-Annual Reliability and Maintainability Symposium(RAMS). IEEE, 2012: 1-5

[6] Luo Peng-cheng. A study on the modeling and analysis technique of system safety analysis based on petri nets[D]. Changsha: National University of Defense Technology, 2001(in Chinese)
罗鹏程. 基于 Petri 网的系统安全性建模与分析技术研究[D]. 长沙: 国防科技大学, 2001

[7] Cho S M, Hong H S, Cha S D. Safety analysis using coloured Petri nets[C]// Asia-Pacific Software Engineering Conference, 1996. IEEE, 1996: 176-183

[8] Vicario E. Static analysis and dynamic steering of time-dependent systems[J]. IEEE Transactions on Software Engineering, 2001, 27(8): 728-748

[9] Berthomieu B, Diaz M. Modeling and verification of time dependent systems using time Petri nets[J]. IEEE Transactions on Software Engineering, 1991, 17(3): 259-273

[10] Vicario E, Sassoli L, Carnevali L. Using stochastic state classes in quantitative evaluation of dense-time reactive systems[J]. IEEE Transactions on Software Engineering, 2009, 35(5): 703-719

[11] Horváth A, Paolieri M, Ridi L, et al. Transient analysis of non-Markovian models using stochastic state classes[J]. Performance Evaluation, 2012, 69(7): 315-335

[12] Aziz A, Sanwal K, Singhal V, et al. Model-checking continuous-time Markov chains[J]. ACM Transactions on Computational Logic(TOCL), 2000, 1(1): 162-170

[13] Carnevali L, Ridi L, Vicario E. Sirio: A framework for simulation and symbolic state space analysis of non-Markovian models[C]// 2011 Eighth International Conference on Quantitative Evaluation of Systems(QEST). IEEE, 2011: 153-154

[14] Tomek L, Mainkar V, Geist R M, et al. Reliability modeling of life-critical, real-time systems[J]. Proceedings of the IEEE, 1994, 82(1): 108-121