嵌入式实时系统的可生存性建模

金永贤，钱雯雯，温兴辉
（浙江师范大学 数理与信息工程学院，浙江 金华 321004）

摘 要：在验证嵌入式实时系统可生存性的过程中，为了避免实验验证和数学模型假设中存在的错误，保证所建模型的准确性，对所建模型的每个组件进行了可生存性分析，从而减小了模型的复杂度，进一步提出了模型故障概率函数，并结合马尔科夫链模型的特点建立了验证嵌入式实时系统可生存性模型。该模型能够根据嵌入式实时系统故障概率密度分布函数，逐个修复或排除高发生率的故障，从而达到增强嵌入式实时系统可生存性的要求。

关键词：嵌入式实时系统；可生存性；故障概率；马尔科夫链
中图分类号：TP393 **文献标识码**：A **文章编号**：1674-7720(2014)02-0063-03

The embedded real-time system survivability modeling

Jin Yongxian, Qian Wenwen, Wen Xinghui
(College of Mathematics, Physics and Information Engineering, Zhejiang Normal University, Jinhua 321004, China)

Abstract: In the process of verifying the embedded real-time system survivability, in order to avoid the error of experimental validation and mathematical models assumption, and ensure the accuracy of the model, the paper analyzes every component of the survivability in the model, thus reducing the complexity of the model, and further proposing the model fault probability function, combining with the characteristics of Markov chain model to establish a validation of embedded real-time system survivability model. Based on the embedded real-time fault probability density function, the model can repair or exclude the high incidence of failure, to achieve the enhancement of requirements of embedded real-time system survivability.

Key words: embedded real-time system; survivability; fault probability; Markov chain

嵌入式实时系统的可生存性[1]是指以计算机技术为基础的嵌入式系统在遭受网络攻击、意外事故或重大灾难等事件时，系统仍然能够在规定的时间约束内完成其基本任务能力，以及外部或内部、同步或异步时间做出响应的能力。由于近年来，嵌入式实时系统在航空、通信和国防等高科技尖端领域的广泛应用，使得研究嵌入式实时系统在发生故障和意外灾难等情况下的可生存能力变得尤为重要。

目前，研究系统可生存性的主要成果有：Barlow 和 Proschan[2] 以及 Siewiorek 和 Swarz[3] 在数学理论的基础上，详细讨论了系统在相应耗损和维护策略下的使用寿命分布(如故障率分布)，并以最小的假设建立了计算机系统的可生存性模型；参考文献[4]中 SHIN K G 等建立了一个关于计算机系统错误检测处理的分析模型，通过该模型检测系统的可生存性能力；林闯[5]从可信网络概念的角度分析了网络安全性、网络可生存性和网络可控性之间的相互关系；参考文献[6]提出了利用多样化分布式动态备份技术和主动漂移机制构建系统的可生存性模型；王慧强[7]提出了开展面向关键任务的分布式信息系统可生存性研究，建立了基于 PST 的分布式信息系统可生存性模型；参考文献[8-9]介绍了关于网络信息系统的可生存性设计的两种主要思路：一是从设计阶段开始就引入可生存性需求，将可生存性需求作为系统设计的先决条件，贯穿于系统开发设计的整个生命周期，最后形成全新的具有可生存能力的系统；二是在原有系统基础上，加入可生存性增强技术(如入侵检测、故障隔离、冗余和自适应等技术)，提高和增强某种系统的可生存性。

1 马尔科夫链模型的应用

通过对模型故障率的分析，可以将系统模型表示成如图 1 所示的马尔科夫链的形式，其中状态表示可用处理器的个数。图中所示的系统在 t 时刻处在状态 i(假设嵌入式

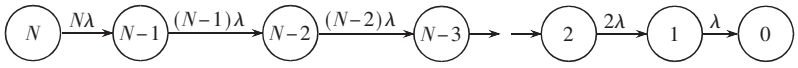

图 1 马尔科夫链模型

实时系统开始于 0 时刻的状态 N)的概率 $\pi_i(t)$由 $\pi_i(t)=\sum_{i=0}^{N-i} A(i,j)e^{-(i+j)\mu t}$ 给出。其中 $A(i,j)=(-1)^j\frac{N!}{i!j!(N-i-j)!}$。

到 t 时刻嵌入式实时系统发生故障的概率可由其所处状态(或 1 或 0)的概率表示,即:

$$\text{Prob}\{\text{系统发生故障}\}=\sum_{i=0}^{1}\pi_i(t)$$

因此不需要区分状态 1 或 0,可以将状态 1 和状态 0 定义为故障状态,计算 t 时刻进入此状态的概率。

单个组件故障对整个系统可生存性造成的冲击与嵌入式实时系统的结构有关。关于给定组件发生故障的保险范围是系统能从该故障中成功恢复的概率,现结合嵌入式实时系统使用硬件冗余,建立嵌入式实时系统可生存性模型。

2 系统三元组故障表决

考虑一个嵌入式实时系统,将其处理器配置成多个三元组,即 $N=3$ 的 NMR 群。当一个三元组中的一个处理器发现故障时,就将它从活动状态移除并用备份代替它。表决过程的另一作用是检测故障。表决每 τ 个单位时间发生一次,被表决过程发现的故障处理器将立即被备份替换。假设只发生持久性故障,处理器故障根据速率为 λ 的泊松过程相互独立。有两个基于故障延迟时间的是或不是指数分布的情况。

目前的系统建模,都假设故障延迟时间是 0,即在发生故障的瞬间就产生错误。但实际情况并不如此,直到故障实行,故障才会产生错误。潜在故障的问题是此类故障不产生错误,所以对系统来说是无形的。当一个单元被检测到故障,此单元就会被隔离出系统。如果不能及时检测,可能导致潜在故障单元在嵌入式实时系统中积累起来。

设故障延迟时间满足均值为 $1/\mu$ 的指数分布。假设备件无限,那么三元组不会因为没有替换硬件而产生故障。只有在至少两个处理器在同一表决期间内同时发生错误的情况下,三元组才会发生故障。图 2 展示了这样的一个时间序列。

图 2 导致三元组故障的一事件序列

一个三元组只有在至少两个处理器在同一表决期内出现错误时才会发生故障。因此要注意每个表决瞬间的嵌入式实时系统状态。

首先构造一个可以捕获每个表决瞬间系统状态的马尔科夫链,这条链可将表决瞬间的系统状态由上一表决瞬间的系统状态的函数给出,是一个离散的“内嵌于表决期的不完全马尔科夫链”。用二元组(x,y)来定义系统的状态,其中 x 是三元组中无故障处理器个数,y 是三元组中存在潜在故障的处理器个数(即一个已经发生故障却没有产生错误的处理器),因为三元组中处理器总数是 3 个,则产生错误的处理器个数就是 $3-x-y$。三元组发生故障的概率可用状态的函数形式表达如下:

$$\text{Prob}\{\text{发生故障}\}=\text{Prob}\{\text{系统状态}\in\{(0,0),(0,1),(1,0)\}\}$$

为了得到这些概率,需要链上的状态转移概率,即需要在给定的第$(i-1)$次表决的状态为(x_1,y_1)的前提下,获得在第 i 次表决的状态为(x_2,y_2)的概率 $p_{(x_1,y_1)(x_2,y_2)}$。表决后如果有两个或更多的处理器产生错误,在表决时可发现错误,涉及的处理器被清除出嵌入式实时系统,并由一个新的处理器(假设是无故障的)代替。如果没有处理器产生错误,则不会重新配置。可以假设忽略这里的重新配置操作时间。按照上面的步骤,三元组继续运行应用程序,直到下一次表决再次重复以上过程。

3 系统各组件状态的确立

表决期可分成两段,第一段执行所有重新配置的操作,可忽略执行时间;第二段运行应用程序,执行时间为 τ,如图 3 所示。

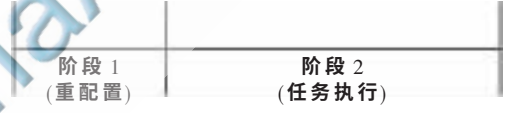

图 3 表决期的分区

如果在表决时系统状态为(x_1,y_1),则重新配置后的状态可由下式给出:

$$\begin{cases}(x_1,y_1) & x_1+y_1=3\\(x_1+1,y_1) & x_1+y_1=2\\\text{发生故障} & x_1+y_1\leqslant 1\end{cases}$$

如果 $x_1+y_1=3$,则三元组中没有产生错误的处理器,因此不会检测到错误,重新配置后的状态与原状态一样。如果 $x_1+y_1=2$,则显然三元组中有一个处理器在前面的表决期内产生了错误,系统会识别出该处理器并用无故障的备件替换。如果 $x_1+y_1\leqslant 1$,则三元组中至少有两个处理器在前面的表决期内产生了错误,则系统发生故障。第一阶段结束时,系统状态为故障或满足 $x_1+y_1=3$ 的(x,y)。该段的状态转移矩阵 P_1 如表 1 所示。

在第一段结束时,系统状态可能是以下 5 个状态之一:(0,3)、(1,2)、(2,1)、(3,0)和故障。给出了重新配置段结束时的系统状态后,计算下一表决瞬间的状态概率,并不能通过审查来确定。为了对第二段发生故障和错误的过程建模,需要一个连续时间的马尔科夫链。如图 4 所示。

表 1 状态转移矩阵 P_1

	故障	(0,0)	(0,1)	(1,0)	(0,2)	(1,1)	(2,0)	(0,3)	(1,2)	(2,1)	(3,0)
(0,0)	1	0	0	0	0	0	0	0	0	0	0
(0,1)	1	0	0	0	0	0	0	0	0	0	0
(1,0)	1	0	0	0	0	0	0	0	0	0	0
(0,2)	0	0	0	0	0	0	0	0	1	0	0
(1,1)	0	0	0	0	0	0	0	0	0	1	0
(2,0)	0	0	0	0	0	0	0	0	0	0	1
(0,3)	0	0	0	0	0	0	0	1	0	0	0
(1,2)	0	0	0	0	0	0	0	0	1	0	0
(2,1)	0	0	0	0	0	0	0	0	0	1	0
(3,0)	0	0	0	0	0	0	0	0	0	0	1

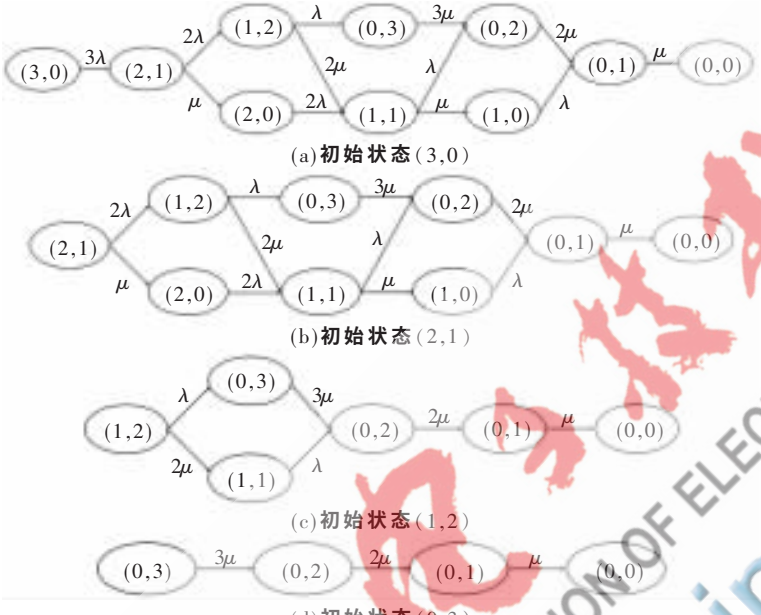

图 4 对第二段发生故障和错误的过程建模

设 $\pi_{x,y}(t)$ 表示在时刻 t 状态为 (x,y) 的概率，那么图 4 中由这些瞬态马尔科夫链可写出状态概率的微分方程为：

$$\frac{d\pi_{3,0}(t)}{dt}=-3\lambda\pi_{3,0}(t)$$

$$\frac{d\pi_{2,1}(t)}{dt}=3\lambda\pi_{3,0}(t)-(2\lambda+\mu)\pi_{2,1}(t)$$

$$\frac{d\pi_{1,2}(t)}{dt}=2\lambda\pi_{2,1}(t)-(\lambda+2\mu)\pi_{1,2}(t)$$

$$\frac{d\pi_{0,3}(t)}{dt}=\lambda\pi_{1,2}(t)-3\mu\pi_{0,3}(t)$$

$$\frac{d\pi_{1,1}(t)}{dt}=2\mu\pi_{1,2}(t)+2\lambda\pi_{2,0}(t)-(\lambda+\mu)\pi_{1,1}(t)$$

$$\frac{d\pi_{0,0}(t)}{dt}=\mu\pi_{0,1}(t)$$

$$\frac{d\pi_{0,2}(t)}{dt}=3\mu\pi_{0,3}(t)+\lambda\pi_{1,1}(t)-2\mu\pi_{0,2}(t)$$

$$\frac{d\pi_{1,0}(t)}{dt}=\mu\pi_{1,1}(t)-\lambda\pi_{1,0}(t)$$

$$\frac{d\pi_{0,1}(t)}{dt}=2\mu\pi_{0,2}(t)+\lambda\pi_{1,0}(t)-\mu\pi_{0,1}(t)$$

$$\frac{d\pi_{2,0}(t)}{dt}=\mu\pi_{2,1}(t)-2\lambda\pi_{2,0}(t)$$

可以把直接求解的值 $\pi_{3,0}(t)=e^{-3\lambda t+C}$ 作为第二个方程的输入，可以得到第二个方程中关于 $\pi_{2,1}(t)$ 的解，再作为第三个方程的输入，求出关于 $\pi_{1,2}(t)$ 的解，以此类推，可以得到第二阶段的状态转移矩阵 P_2，如表 2 所示。

将两段的状态转移概率合并可得到整个表决期的状态转移概率 $P=P_1\times P_2$，该状态转移矩阵显示了嵌入式实时系统每个组件的故障概率状态，从而可以推出整体

表 2 第二阶段状态转移矩阵 P_2

	故障	(0,0)	(0,1)	(1,0)	…	(0,3)	(1,2)	(2,1)	(3,0)
故 障	1	0	0	0	…	0	0	0	0
(0,0)	1	$\pi_{0,0}(r\|0,0)$	$\pi_{0,1}(r\|0,0)$	$\pi_{1,0}(r\|0,0)$	…	$\pi_{0,3}(r\|0,0)$	$\pi_{1,2}(r\|0,0)$	$\pi_{2,1}(r\|0,0)$	$\pi_{3,0}(r\|0,0)$
(0,1)	1	$\pi_{0,1}(r\|0,1)$	$\pi_{0,1}(r\|0,1)$	$\pi_{1,0}(r\|0,1)$	…	$\pi_{0,3}(r\|0,1)$	$\pi_{1,2}(r\|0,1)$	$\pi_{2,1}(r\|0,1)$	$\pi_{3,0}(r\|0,1)$
(1,0)	1	$\pi_{0,0}(r\|1,0)$	$\pi_{0,1}(r\|1,0)$	$\pi_{1,0}(r\|1,0)$	…	$\pi_{0,3}(r\|1,0)$	$\pi_{1,2}(r\|1,0)$	$\pi_{2,1}(r\|1,0)$	$\pi_{3,0}(r\|1,0)$
					…				
(0,3)	0	$\pi_{0,0}(r\|0,3)$	$\pi_{0,1}(r\|0,3)$	$\pi_{1,0}(r\|0,3)$	…	$\pi_{0,3}(r\|0,3)$	$\pi_{1,2}(r\|0,3)$	$\pi_{2,1}(r\|0,3)$	$\pi_{3,0}(r\|0,3)$
(1,2)	0	$\pi_{0,0}(r\|1,2)$	$\pi_{0,1}(r\|1,2)$	$\pi_{1,0}(r\|1,2)$	…	$\pi_{0,3}(r\|1,2)$	$\pi_{1,2}(r\|1,2)$	$\pi_{2,1}(r\|1,2)$	$\pi_{3,0}(r\|1,2)$
(2,1)	0	$\pi_{0,0}(r\|2,1)$	$\pi_{0,1}(r\|2,1)$	$\pi_{1,0}(r\|2,1)$	…	$\pi_{0,3}(r\|2,1)$	$\pi_{1,2}(r\|2,1)$	$\pi_{2,1}(r\|2,1)$	$\pi_{3,0}(r\|2,1)$
(3,0)	0	$\pi_{0,0}(r\|3,0)$	$\pi_{0,1}(r\|3,0)$	$\pi_{1,0}(r\|3,0)$	…	$\pi_{0,3}(r\|3,0)$	$\pi_{1,2}(r\|3,0)$	$\pi_{2,1}(r\|3,0)$	$\pi_{3,0}(r\|3,0)$

系统的可生存性。

根据上述实验结果可以看到，本文所提出的嵌入式实时系统可生存性建模方法能够正确地反应出可生存性的关键属性；系统可生存性不仅与其所受攻击的严重性、攻击强度有关，还与系统对攻击的抵抗、检测及恢复等可生存性能密切相关。可生存性是一个整体性的综合评估值，反应了系统的整体性能。

本文对嵌入式实时系统的可生存性方法中实验验证法和建立数学模型法进行了分析，并提出了所存在的问题，对嵌入式实时系统可生存性模型进行了改进。所建立的模型能够通过分析模型的复杂度确立故障发生概率，因此，根据概率的大小，系统会自动移除概率大的故障，从而保证系统的可生存性。

参考文献

[1] ELLISON R J，FISHER D A，LINGER R C，et al.Survivable network system：an emerging discipline[EB/OL].(2007-11-20)[2013-08-30].http://www.cert.org/research/97tr013.pdf.

[2] BARLOW R E，PROSCHAN F.Mathematical theory of reliability[M].Siam：Society for Industrial and Applied，1996.

[3] SIEWIOREK D P，SWARZ R S.Reliable computer systems：design and evaluation[M].Massachusetts：AK Peters，1998.

[4] SHIN K G，LEE Y H.Error detection process-model，design and its impact on computer performance[J].IEEE Transaction，1984，C-33(6)：529-540.

[5] 林闯，彭雪海.可信网络研究[J].计算机学报，2005，28(5)：751-758.

[6] 黄遵国，卢锡城，胡华平.生存能力技术及其实现案例研究[J].通信学报，2004，25(7)：137-145.

[7] Wang H Q，Liu D X.A holistic approach to survivable distributed information system for critical applications[C].In：The Proc.of ISPA'2005，Nanjing，2005：713-724.

[8] 张乐君，国林，王巍，等.网络系统可生存性评估与增强技术研究概述[J].计算机科学，2007，34(8)：30-33.

[9] Ma Qingkai，Xiao Liangliang，YEN I L，et al.An adaptive multiparty protocol for seccure data protection[C].Makoto T.Proc.of the Paralleland Distributed Systems.Los Alamitos：IEEE Computer Society，2005：43-49.

(收稿日期：2013-09-20)

作者简介：

金永贤，男，1964年生，教授，主要研究方向：实时与嵌入式。

钱雯雯，女，1989年生，硕士研究生，主要研究方向：嵌入式实时系统。

温兴辉，男，1988年生，硕士研究生，主要研究方向：嵌入式。