

# 基于自适应心跳算法的分布式系统故障检测器

王 明 张春熹 伊小素

(北京航空航天大学 惯性技术国防科技重点实验室, 北京 100191)

**摘 要:** 故障检测是容错分布式系统中的关键技术之一. 为了提高故障检测的性能, 提出一种新型的故障检测器——自适应心跳检测器(SA-HD, Self-Adaptive Heartbeat Detector). SA-HD 采用了基于拉式(pull)的自适应心跳算法, 在考虑故障检测性能的同时也考虑了心跳检测所占用的网络资源对网络性能的影响. SA-HD 能够根据网络负载调节自身发送心跳消息的频率, 提高了心跳检测的网络环境适应能力, 尤其是在高负载的环境下, 能够有效改善心跳检测的性能. 建立了 SA-HD 的模型, 对其性能进行了仿真分析, 并通过试验验证了 SA-HD 性能要优于传统推式(push)的心跳检测器.

**关 键 词:** 分布式系统; 故障检测; 自适应心跳检测算法

**中图分类号:** N 945.17

**文献标识码:** A

**文 章 编 号:** 1001-5965(2013)07-0952-05

## Fault detector of fault-tolerant distributed systems based on self-adaptive heartbeat algorithm

Wang Ming Zhang Chunxi Yi Xiaosu

(Key Laboratory of National Defense on Inertial Technology, Beijing University of Aeronautics and Astronautics, Beijing 100191, China)

**Abstract:** The detection of failures is one of the fundamental issues for fault tolerant distributed systems. In order to improve the performance of the fault detection, a novel failure detector called self-adaptive heartbeat detector (SA-HD) was proposed. SA-HD employed a self-adaptive heartbeat detecting algorithm based on pull mode, in which the performance of the failure detection and the network resource caused by heartbeat detecting were both considered. The algorithm could change the frequency of transmitting the heartbeat messages and enhance the adaptability for the network environment, especially in a heavy network payload. The model of SA-HD was built and simulations were carried out to compare the performance between SA-HD and the approach based on push mode. The simulation analysis and experiment results show that the performance of SA-HD is better than the latter's.

**Key words:** distributed systems; failure detection; self-adaptive heartbeat detecting algorithm

由于分布式系统在航空航天、军事、交通等多种领域的广泛应用,其安全性和可靠性成为系统设计中的关键问题<sup>[1-2]</sup>.随着分布式系统的规模不断扩大,系统中各部分组件相互之间的依赖关系也不断复杂化,使得系统发生故障的概率也不断增加,甚至某些故障将导致整个系统的瘫痪.因此,容错设计成为分布式系统中的关键设计因素.而故障检测作为分布式系统容错技术的重要支撑

技术,能够及时并正确检测系统故障,为系统容错组件提供故障信息.

在故障检测方法中,心跳检测算法是分布式系统最常用的方法之一<sup>[1]</sup>.随着分布式系统信息量的增加,传统的周期发送心跳消息的方式已经不能满足故障检测的服务质量(QoS, Quality of Service)需求,而目前大多数故障检测算法集中于改善传统的检测结构,针对故障检测 QoS 的研究

收稿日期: 2012-08-02; 网络出版时间: 2013-01-23 09:48

网络出版地址: [www.cnki.net/kcms/detail/11.2625.V.20130123.0948.001.html](http://www.cnki.net/kcms/detail/11.2625.V.20130123.0948.001.html)

作者简介: 王 明(1985-) 男,河北泊头人,博士生, wangming@aspe.buaa.edu.cn.

成果则较少<sup>[3-10]</sup>. 文献[6-8]中在针对推式(push)的故障检测QoS进行了研究,但是在push方式下被测进程周期性主动发送心跳信息,自适应能力差,尤其是在网络高负载环境下,无法调节心跳检测的调度算法,降低了网络的性能和效率.

本文在考虑故障检测性能的同时,还将网络性能作为重要的因素,定义了一种分布式系统故障检测的性能评价指标,提出了一种基于拉式(pull)的自适应心跳检测器(SA-HD, Self-Adaptive Heartbeat Detector). SA-HD能够有效提高分布式系统故障诊断和信息传输的综合性能.

## 1 心跳检测原理及性能评价指标

### 1.1 心跳检测原理

心跳检测是分布式系统中常用的故障检测方法,系统中的心跳检测器通过监督各节点发送的心跳消息来判断这些节点是否发生故障.心跳检测有两种方式:push方式和pull方式.

在push方式下,被检测节点周期性地向心跳监视器发送心跳消息.如果心跳监视器在规定的时间内没有心跳信息,则认为被检测节点发生故障,如图1所示.

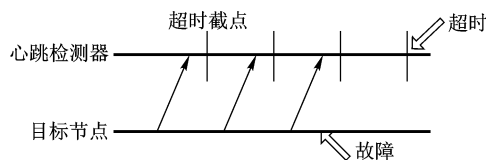


图1 push方式

在pull方式下,心跳检测器向目标节点发送心跳消息后,目标节点对消息进行应答.如果心跳检测器发送请求后收到应答则表示目标节点正常,反之则表示该节点发生故障,如图2所示.

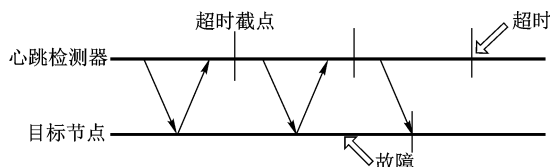


图2 pull方式

### 1.2 故障检测性能指标

为了衡量故障检测器的检测能力,文献[9]提出了故障检测器最基本的两个属性:①完整性(completeness):故障检测系统最终怀疑故障进程的时间;②准确性(accuracy):故障检测系统不会将正确的进程误判为故障的时间.

传统的心跳检测器主要针对以上两个性能指标设计检测算法,心跳检测周期越短,其性能越好.但是在分布式系统中,频繁发送心跳消息将导

致网络负载增加,降低网络传输的性能,因此必须考虑心跳检测对网络传输性能的影响.

对于网络传输,消息平均等待时间是网络性能的重要指标,心跳检测算法设计中不仅要考虑故障检测的完整性和准确性,还要考虑消息的平均等待时间.故障检测性能指标定义如下.

定义1 设故障检测性能参数为 $K$ 则有

$$K = \frac{1}{k_1 T_C + k_2 T_A + k_3 T_W} \quad (1)$$

其中 $T_C$ 为完整性时间参数; $T_A$ 为准确性时间参数; $T_W$ 为消息平均等待时间; $k_1$ 、 $k_2$ 和 $k_3$ 分别为3个指标所占的比重.根据系统需求定义.在本文提出的算法中 $k_1 = k_2 = k_3 = 1$ ,表示各指标所占比重相同.对一个分布式系统 $K$ 值越大说明故障检测的综合性能越好.

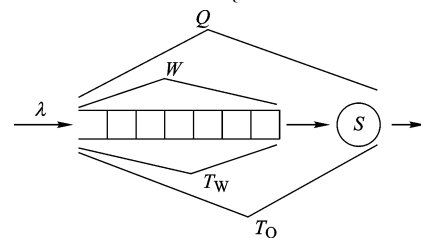
## 2 分布式系统网络消息模型

针对在分布式系统中的网络环境,做以下两个假设:

1) 网络中所有消息的长度相同,即心跳消息和数据消息的长度均为 $L$ ,网络带宽为 $C$ ,则数据包的发送时间 $T_S = L/C$ ,服务速率 $\mu = 1/T_S$ ;

2) 消息的到达过程服从Poisson分布,平均到达速率为 $\lambda$ ,其中心跳消息和数据消息的平均到达速率分别为 $\lambda_B$ 和 $\lambda_D$ ,则 $\lambda = \lambda_B + \lambda_D$ .

基于以上考虑,分布式系统网络消息传输过程服从M/D/1队列过程<sup>[11]</sup>,如图3所示.由图3可知消息的总服务时间 $T_Q = T_W + T_S$ .



$Q$ —某时刻网络中所有数据消息的数量;

$W$ —等待发送的数据消息的数量.

图3 网络消息模型

该队列模型的状态转移过程构成一个嵌入马尔可夫链.该马尔可夫链的一步转移矩阵表达为

$$P = \begin{bmatrix} a_0 & a_1 & a_2 & a_3 & \cdots \\ a_0 & a_1 & a_2 & a_3 & \cdots \\ 0 & a_0 & a_1 & a_2 & \cdots \\ 0 & 0 & a_0 & a_1 & \cdots \\ 0 & 0 & 0 & a_0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \quad (2)$$

由此可得出其状态转移图,如图4所示.

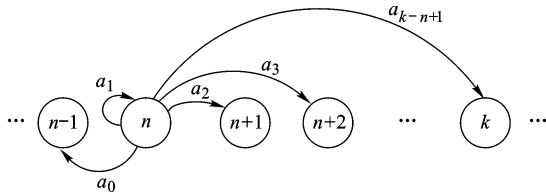


图4 马尔可夫链状态转移图

通过该 M/D/1 模型的马尔可夫链状态转移过程,可获得  $T_Q$  和  $T_W$  的期望:

$$E(T_Q) = T_s + \frac{\lambda T_s^2}{2(1 - \lambda T_s)} \quad (3)$$

$$E(T_W) = \frac{\lambda T_s^2}{2(1 - \lambda T_s)} \quad (4)$$

### 3 SA-HD 心跳检测算法

在心跳检测方法中,多数算法采用 push 方式,通过图1和图2对比,在一次检测期间 push 方式只发送一次心跳消息,而 pull 则需要两次,因此 push 方式占用网络资源较少.但是采用 push 方式心跳检测器没有主动权,只能被动等待目标节点的心跳消息.基于 push 的自适应算法通常根据网络状态推算心跳消息的到达时间而达到对网络状态的自适应.

本文提出的 SA-HD 采用了 pull 方式设计自适应心跳算法.根据网络负载调节心跳消息的发送速率  $\lambda_B$ ,即在高负载时适当降低  $\lambda_B$  或在低负载时适当提高  $\lambda_B$ ,使得式(1)中故障检测性能指标的  $K$  获得最优化.

#### 3.1 $\lambda_B$ 推导

在故障检测过程中,SA-HD 以频率  $\lambda_B$  向目标节点发送心跳检测信息.设心跳检测时间  $T_D$  为在一次检测过程中从发送心跳消息到接收到目标节点的心跳应答的时间.在不考虑 SA-HD 和目标节点的处理时间的情况下,有

$$E(T_D) = 2E(T_Q) \quad (5)$$

即  $T_D$  的期望心跳消息发送和应答的时间的期望之和,其中心跳发送和应答时间的期望均为  $T_Q$ .

SA-HD 中存在一个滑动窗口  $W$ ,其保存最近  $w$  个心跳检测的检测时间,用于计算  $E(T_D)$ .

由式(6)和式(3)可知  $E(T_D)$  的大小由网络负载的大小决定,通过  $E(T_D)$  能够获得当前网络的负载情况,因此,存在一个函数能够获得使式(1)中的  $K$  在当前网络负载环境下达到最优的  $\lambda_B$ :

$$\lambda_B = F(E(T_D)) \quad (6)$$

为了获得该函数,下面对  $K$  进行分析.

在最坏的情况下,即在一次心跳检测中当目标节点发送完心跳响应消息后立刻发生了故障,从故障发生到 SA-HD 判断发生故障的最晚时间为  $T_C$ ,而  $T_A$  为在无故障情况下从 SA-HD 发送心跳消息到确定目标节点无故障的最晚时间,如图5所示.图中  $T_L$  表示心跳等待超时时间,即当  $T_D > E(T_D) + T_L$  时,则 SA-HD 判断目标节点发生故障.在本算法中规定  $T_L = T_Q$ .

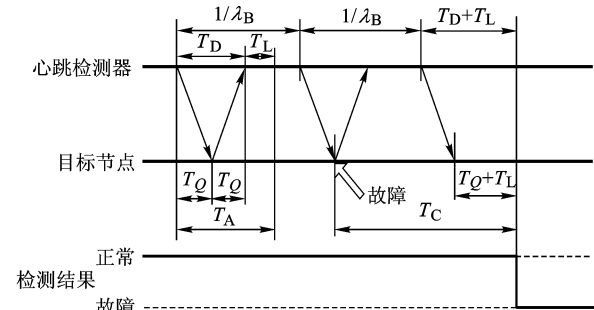


图5 马尔可夫链状态转移图

根据图5以及式(3)、式(4)可得

$$T_C = \frac{1}{\lambda_B} + 2T_s + \frac{(\lambda_B + \lambda_D) T_s^2}{1 - (\lambda_B + \lambda_D) T_s} \quad (7)$$

$$T_A = \frac{3}{2} \left[ T_s + \frac{(\lambda_B + \lambda_D) T_s^2}{1 - (\lambda_B + \lambda_D) T_s} \right] \quad (8)$$

再由式(1)可得

$$K = \frac{1}{\lambda_B} + 3 \left[ \frac{2T_s - (\lambda_B + \lambda_D) T_s^2}{1 - (\lambda_B + \lambda_D) T_s} \right] - T_s \quad (9)$$

对  $K$  求导,当  $K' = 0$  时得到最优  $\lambda_B$ :

$$\lambda_B = \frac{(\sqrt{3} - 1)(1 - \lambda_D T_s)}{2T_s} \quad (10)$$

根据式(3)、式(6)可以获得  $\lambda_B$  和  $\lambda_D$  的关系:

$$\lambda_D = \frac{E(T_D) - 2T_s}{E(T_D) T_s - T_s^2} - \lambda_B \quad (11)$$

将式(11)代入式(10),可获得式(6)描述的函数:

$$\lambda_B = F(E(T_D)) = \frac{(\sqrt{3} - 1)}{(3 - \sqrt{3}) T_s} \left( 1 - \frac{E(T_D) - 2T_s}{E(T_D) \cdot T_s - T_s^2} T_s \right) \quad (12)$$

#### 3.2 SA-HD 心跳检测算法伪代码

SA-HD 心跳检测算法包括2个过程:心跳检测过程和心跳应答过程.分别用于 SA-HD 和目标节点.而心跳检测过程包括心跳消息发送和应答检测2个部分,具体算法如下所示.

For heartbeat detector:

Initial:  $\lambda_B = A$ ;  $i = 0$ ;  $t_i = 0$ ;

Upon  $t_{\text{current}} = t_i + 1/\lambda_B$ :

```

 $i = i + 1;$ 
 $t_i = t_{\text{current}};$ 
transmit a heartbeat message;
Upon receive the heartbeat response:
  If  $(t_{\text{current}} - t_i) < E(T_D) + E(T_Q)$ 
    begin
      If  $i < w$ 

$$E(T_D) = \frac{1}{i} \sum_{j=1}^i T_D(j);$$

      else
        begin
 $T_D = t_{\text{current}} - t_i;$ 
        Add  $T_D$  to  $W$ ;

$$E(T_D) = \frac{1}{w} \sum_{j=1}^w T_D(j);$$

 $\lambda_B = F(E(T_D));$ 
        end
      end
    end
  else
    A fault occurs on the destination;
  For destination node:
    Upon receive the heartbeat message:
      transmit a heartbeat response message.

```

其中  $A$  为  $\lambda_B$  的初始值, 取  $0 \sim \mu$  之间;  $t_{\text{current}}$  为当前时间。

## 4 仿真及试验

### 4.1 故障检测性能的仿真及分析

本节通过仿真, 对 SA-HD 的性能进行分析。仿真条件如下: ①网络带宽:  $C = 800 \text{ Mbit/s}$ ; ②消息长度:  $L = 1 \text{ KB}$ 。

由以上 2 个条件可知,  $\lambda_B + \lambda_D$  最大值为 100 000, 因此仿真中  $\lambda_D$  取值范围为  $0 \sim 80000/\text{s}$ 。

本文对  $\lambda_B$  为 20 000 和 1 000 的 push 方式进行仿真, 并将 SA-HD 算法的性能与 push 方式对比。图 6 ~ 图 9 分别为在  $\lambda_D$  变化过程中对 3 种心跳检测方式的  $T_C$ ,  $T_A$ ,  $T_W$  以及  $K$  的仿真结果。

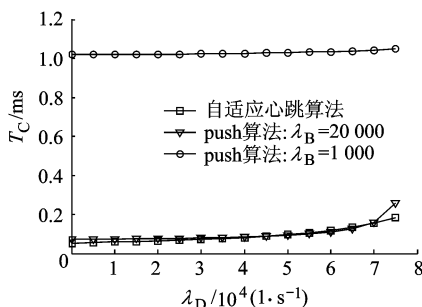


图 6  $\lambda_D$  与  $T_C$  的关系

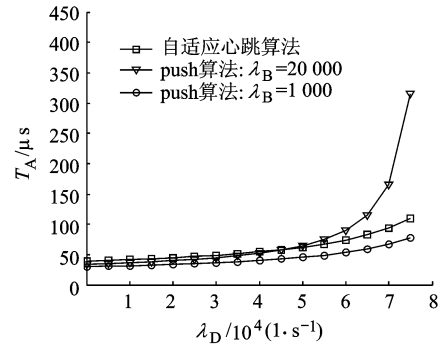


图 7  $\lambda_D$  与  $T_A$  的关系

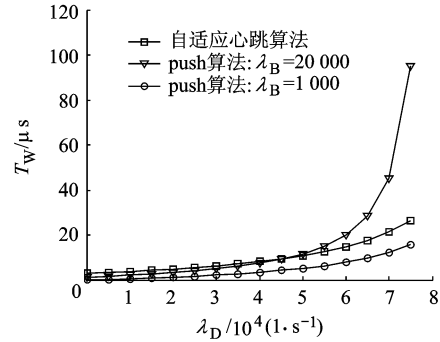


图 8  $\lambda_D$  与  $T_W$  的关系

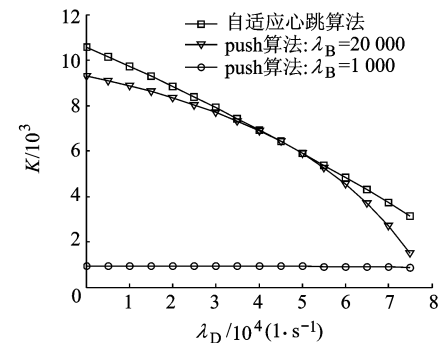


图 9  $\lambda_D$  与  $K$  的关系

从图 6 中可以看出 SA-HD 相对 push 方式具有更短的  $T_C$ , 保持在  $200 \mu\text{s}$  以内。在 push 方式下  $\lambda_B$  是决定  $T_C$  的主要因素,  $\lambda_B$  越大  $T_C$  越短。由于 push 方式没有调节能力, 因此当  $\lambda_B + \lambda_D$  接近最大值 100 000 时,  $T_C$  由于网络阻塞, 迅速增长。而 SA-HD 在高负载情况下通过对  $\lambda_B$  的调节来保持  $T_C$ , 从而提高检测的性能。

图 7 中 SA-HD 的  $T_A$  介于 2 种 push 方式之间, 保持在  $100 \mu\text{s}$  以内。因为决定  $T_A$  的主要因素是网络的负载, 即在相同  $\lambda_D$  条件下,  $\lambda_B$  越小  $T_A$  越小, 所以在 push 方式下,  $T_A$  在  $\lambda_B = 1 000$  时比  $\lambda_B = 20 000$  时小。而在 SA-HD 中,  $\lambda_B$  的选取兼顾心跳检测性能和网络性能, 其  $T_A$  介于其他 2 种方式之间。

与  $T_A$  相同,  $T_W$  的主要决定因素也是网络的负载。如图 8 所示, SA-HD 的  $T_W$  保持在  $30 \mu\text{s}$  以

内,介于 2 种 push 方式之间.

由图 9 可知 SA-HD 的  $K$  是 3 种方式中最优的. 在 push 方式下  $\lambda_B=1\,000$  时,由于  $T_C$  较大,导致  $K$  值较小. 而当  $\lambda_D$  在 40 000 附近时,SA-HD 的最优  $\lambda_B$  为 20 000,因此此时 SA-HD 与  $\lambda_B=20\,000$  的 push 方式具有相同的  $K$  值.

通过对  $T_C$ ,  $T_A$ ,  $T_W$  以及  $K$  的仿真结果对比, SA-HD 是 3 种方法中最优的方法.

## 4.2 试验结果

试验以 IEEE 1394b 光总线分布式系统为平台,采用故障注入的方式对故障检测性能测试. 为方便与仿真结果比较分析,试验条件与仿真条件相同,即:  $C=800\text{ Mbit/s}$ ,  $L=1\text{ KB}$ .

在不同网络负载下分别对 SA-HD,  $\lambda_B=1\,000$  的 push 方式以及  $\lambda_B=20\,000$  的 push 方式进行测试并统计故障检测性能参数,试验结果如表 1 所示.

表 1 不同  $\lambda_D$  下 3 种方式的  $K$  值

故障检测 方式	$\lambda_D$			
	10 000	30 000	50 000	70 000
SA-HD	9 415	7 552	5 681	3 369
push( $\lambda_B=1\,000$ )	735	713	700	690
push( $\lambda_B=20\,000$ )	8 509	7 486	5 692	2 453

从表 1 中的试验结果可知 SA-HD 相对其他两种 push 方式具有更好的性能,该结果与仿真结果基本吻合. 但是由于在实际分布式系统中,各节点在消息处理过程中以及消息在链路中传递都存在一定的延时,因此导致试验中的性能参数略小于仿真结果.

## 5 结 论

心跳检测方法由于其算法简单、故障检测速度快等优势,是分布式系统中常用的故障检测方法. 本文提出 SA-HD 采用了基于 pull 方式的自适应心跳检测算法,在考虑心跳检测性能的同时,也考虑了心跳检测与网络性能之间的关系,最终确定一个最优心跳检测频率. SA-HD 具有灵活的调节能力,能够适应网络负载的变化对检测性能的

影响,提高了故障检测的性能. 此外,本文通过仿真分析及试验验证将 SA-HD 与传统的 push 方式心跳检测器进行了对比,最终证明了 SA-HD 具有更好的故障检测效果.

## 参考文献 (References)

- [1] Xiong Naixue, Yang Yan. A survey on fault-tolerance in distributed network systems [C]//Proceedings of IEEE International Conference on Computational Science and Engineering. New York: IEEE, 2009: 1065–1070
- [2] Felber P, Defago X, Guerraoui R, et al. Failure detectors as first class objects [C]//Proceedings of IEEE International Symposium on Distributed Objects and Applications. New York: IEEE, 1999: 132–141
- [3] Wiesmann M, Urban P, Defago X. An SNMP based failure detection service [C]//Proceedings of the 25th IEEE International Symposium on Reliable Distributed Systems. New York: IEEE, 2006: 365–374
- [4] Zhu Hao, Chen Haopeng. Adaptive failure detection via heartbeat under hadoop [C]//Proceedings of IEEE Asia-Pacific Services Computing Conference. Jeju: IEEE, 2011: 231–238
- [5] Roberto B, Jean M H, Sara T P. A methodology to design arbitrary failure detectors for distributed protocols [J]. Journal of Systems Architecture, 2008, 54(7): 619–637
- [6] Chen W, Sam T, Marcos K A. On the quality of service of failure detectors [J]. IEEE Transactions on Computers, 2002, 51(1): 13–32
- [7] Naohiro H, Xavier D, Rami Y, et al. The  $\varphi$  accrual failure detector [C]//Proceedings of the 23th IEEE International Symposium on Reliable Distributed Systems. New York: IEEE, 2004: 66–78
- [8] Benjamin S, Andreas P, Wolfgang T, et al. A lazy monitoring approach for heartbeat-style failure detectors [C]//Proceedings of the 3th International Conference on Availability, Reliability and Security. New York: IEEE, 2008: 404–409
- [9] Chandra T D, Toueg S. Unreliable of failure detectors for reliable distributed systems [J]. Journal of the ACM, 1996, 43(2): 225–267
- [10] Fetzer C, Raynal M, Tronel F. An adaptive failure detection protocol [C]//Proceedings of the 8th Pacific Rim Symposium on Dependable Computing. New York: IEEE, 2001: 146–153
- [11] Kleinrock L. Queueing systems, volume 1: theory [M]. New York: John Wiley, 1962

(编辑: 赵海容)