嵌套式动态容错协议的研究与设计*

史殿习, 吴泉源, 王怀民, 邹 鹏

(国防科学技术大学 计算机学院,湖南 长沙 410073)
E-mail: dxshi@etang.com
http://www.nudt.edu.cn

摘要: 从软件容错的角度出发,在基于网络的分布计算环境下,针对军事指挥系统及银行管理系统的特点,为了满足这些应用对可靠性的要求,以组通信为基础,采用进程复制技术,提出了一个嵌套式动态容错模型;进而提出了一个动态容错算法,该算法保证当主服务进程发生失效时,能够动态地选择一个新的主服务进程,并保证所有后备服务进程的状态保持一致.

关 键 词: 复制;容错;组通信;虚拟同步
中图法分类号: TP393 文献标识码: A

在很多分布式应用系统如军事指挥系统、银行管理系统等中,系统中的关键服务在为客户应用或其他应用提供服务时,可能还要访问系统中其他的关键服务才能完成一个应用请求,服务之间形成了嵌套调用,而且嵌套调用可能是多层次的.现有的基于复制的主备份复制方法和状态机复制方法都没有考虑这种情况[1].为满足上述应用对可靠性的要求,根据这些应用的特点,我们提出了一个嵌套式的动态容错模型,进而以该模型为基础,提出了一个嵌套式的动态容错协议,该协议将主备份复制方法和状态机复制方法有机地结合起来,当主服务进程发生失效时,利用组通信中的虚拟同步机制来动态地保证所有后备服务进程状态的一致性,并动态地选择一个后备服务进程作为新的主服务进程,从而保证系统正确地运行.

1 嵌套式容错模型

嵌套式动态容错模型如图 1 所示.该模型以组通信技术为基础[2],采用进程复制技术,将系统中的每个关键服务复制多份,分布在系统中不同的结点上,形成一个副本组,在每个副本组内指定一个服务进程作为主服务进程(primary),其余服务进程作为后备服务进程.{$primary_1$,$replica_{1,1}$,$replica_{1,2}$,…,$replica_{1,n}$}为一个完整的服务进程组,其中,$Primary_1$ 为主服务进程,{$replica_{1,1}$,$replica_{1,2}$,…,$replica_{1,n}$}为同组的后备服务进程集合.

客户应用与主服务进程、服务进程与服务进程之间的交互主要通过如下 4 类消息:

(1) req_c^i:客户向主服务进程发出的第 *ith* 请求;

(2) res_c^i:服务进程向客户返回的第 *ith* 请求的处理结果;

(3) req_s^i:服务进程向其他服务进程发出的第 *ith* 请求;

(4) res_s^i:服务进程向其他服务进程返回的第 *ith* 请求的处理结果;

当客户请求服务进程为其提供服务时,主服务进程接收请求并进行处理,同时通过组播通信原语 RMcast( )

* 收稿日期: 2000-03-28; 修改日期: 2000-07-26
基金项目: 国家自然科学基金资助项目(69833030);国家 863 高科技发展计划资助项目(863-306-ZD02-01-2);国家 863 青年基金资助项目(863-306-QN2000-3)

作者简介: 史殿习(1966－),男,山东龙口人,博士,讲师,主要研究领域为分布计算;吴泉源(1942－),男,上海人,教授,博士生导师,主要研究领域为分布计算,人工智能;王怀民(1964－),男,江苏南京人,教授,博士生导师,主要研究领域为分布计算,人工智能;邹鹏(1958－),男,福建福州人,教授,博士生导师,主要研究领域为操作系统,分布计算.

可靠地将客户请求组播给同组的其他后备服务进程.当主服务进程发生失效时,该模型启动虚拟同步协议进行同步处理,保证所有后备服务进程执行状态的一致性.主服务进程在对请求处理期间,可以向其他服务进程发送服务请求,并等待服务结果.

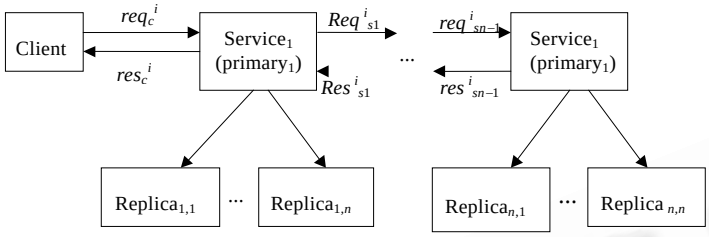

Fig. 1 The nested dynamic fault-tolerant model

图 1 嵌套式动态容错模型

2 嵌套式容错模型

在嵌套式动态容错模型中,主服务进程的角色与其他后备服务进程的角色是不同的,它负责接收客户的请求,并将请求可靠地组播给同组的其他后备服务进程,最后将处理结果返回给客户.由于客户不直接与后备服务进程进行交互,因此,后备服务进程的失效并不影响系统的正常运行;但当主服务进程发生失效时,必须从后备服务进程中选择一个服务进程来承担主服务进程的角色.为此,需要对副本组中的成员关系进行动态的管理,以便在主服务进程发生失效时,能够:(1) 定义一个新的、唯一的服务进程来充当主服务进程的角色;而且,(2) 新的主服务进程的状态与其他所有后备服务进程的状态是一致的.为了满足上述两个条件,我们将组通信中的虚拟同步机制[2]引入到嵌套式动态容错模型中.

2.1 虚拟同步的定义

虚拟同步机制与组的成员关系变化是密切相关的.为了表示上的方便,我们引入视图的概念,视图是对组成员关系的一种抽象.

定义 **1**(视图). 假设一个组 g,它的一个视图是指由组 g 中当时处于正确的运行状态且彼此之间可以进行相互通信的成员进程组成的进程子集,而且该子集被其中的所有成员进程都认同.

我们用 $V_i(g)(i\in N)$来表示组 g 的一个视图,每一个视图都有唯一的标识.组 g 的视图变化是一个线性序列.

在一个组中,组成员之间的组播通信随时受组视图变化的影响,对于一个组成员来说,不仅要处理成员之间的组播消息,而且还要处理视图变化通知消息,并且这两类消息相互交织穿插在成员所处理的消息流中.虚拟同步机制的作用就是对这两类消息进行协调处理,保证正确的组成员之间对这两类消息的同步处理.下面给出虚拟同步的严格定义.

定义 **2(**虚拟同步). 假设组 g, 视图 $V_i(g)$和 $V_j(g)$且 $V_j(g)$是 $V_i(g)$的直接后继,我们称组 g 中的通信是虚拟同步的当且仅当如果存在一个进程 $p\in V_i(g)$在视图 $V_i(g)$中已经传递(处理)了消息 m 且已安装了下一个视图 $V_j(g)$,则所有已经安装下一个视图 $V_j(g)$的进程在安装 $V_j(g)$之前已经传递(处理)了消息 m.

2.2 虚拟同步算法

为了在组成员之间实现虚拟同步通信,我们将虚拟同步问题转化为一致性问题(consensus)[3],并对文献[3]中的一致性算法(将其标记为◊S-Consensus)[3]进行修改,将修改后的算法标记为◊SM-Consensus.算法◊SM-Consensus 与算法◊S-Consensus 的不同之处在于:◊SM-Consensus 算法对◊S-Consensus 算法中的第 2 阶段进行了修改,即协调者 p_c 根据接收到的各个成员的提议消息来计算新的估计值,p_c 关于下一个视图的估计值是当前所有正确的成员关于下一个视图的估计值的交集,这样可以保证下一个估计视图中不包含被怀疑失效的成员;p_c 在当前视图内传递的消息集合是所有正确的成员在当前视图内传递的消息集合的并集,这样可以保证

当前所有正确的成员不遗漏一条在当前视图中被其他成员传递的消息.

以◊SM-Consensus 算法为基础,我们设计了一个新的虚拟同步算法 VSC_Mcast( ),该算法的优点在于,在参与同步的成员的失效个数不超过一半的情况下,该算法的的运行都是正确的,并且满足终止性、一致性及有效性[3]等特性.

3 嵌套式动态容错算法

在对虚拟同步机制研究的基础上,以虚拟同步机制为基础,我们设计了一个嵌套式动态容错协议,该协议分为请求接收过程、请求发送过程、请求队列维护过程及主服务进程失效处理过程等 4 个部分.下面分别给出这 4 个过程的算法描述.

当服务进程向客户或其他服务进程发送消息 m 时,则激活消息发送过程 *SendMsg*(m)进行消息的发送,该过程的算法描述如算法 1 所示.

算法 **1**. 消息发送过程 *SendMsg*(m)

```
if 发送者为主服务进程
then   (1) 主服务进程将消息 m 发送给目的地
       (2) 主服务进程通过原语 Rmcast( )将消息 m 可靠地组播给同组内所有其他成员
if 发送者为后备服务进程
then 以消息 m 作为参数调用消息队列维护过程 UpdateMsgQ(m),对消息队列进行维护和更新
```

当服务进程接收到来自客户或其他服务进程的请求消息 m 时,则激活消息接收过程 *HandleRecvMsg* (m)对接收到消息进行处理,该过程的算法描述如算法 2 所示.

算法 **2**. 消息接收过程 *HandleRecvMsg*(m)

```
if 接送者为主服务进程
then   (1) 主服务进程通过原语 Rmcast( )将接收的消息 m 可靠地组播给同组内所有其他成员
       (2) 主服务进程对消息 m 进行处理
if 接收者为后备服务进程
then
    if 消息 m 的类型为 res_c^i 或 req_s^i
    then 对接收到的消息进行处理,并将消息放入消息缓冲队列 MsgQ_p^i 中
    if 消息 m 的类型为 res_c^i 或 req_s^i
    then 以消息 m 作为参数调用队列维护过程 UpdateMsgQ(m),对消息队列进行维护和更新
```

消息队列处理维护过程主要用于处理后备服务进程发出的类型为 res_c^i 和 req_s^i 的消息,以及来自主服务进程的类型为 res_c^i 和 req_s^i 的消息,这些消息只在主服务进程与后备服务进程之间流动,目的是让后备服务进程时刻了解主服务进程的执行状态.通过将后备服务进程发送的消息与接收到的来自主服务进程的消息进行比较,可判断后备服务进程当前的执行状态是否与主服务进程的执行状态一致.其算法描述如算法 3 所示.

算法 **3**. 消息队列维护过程 *UpdateMsgQ*(m)

```
if  in MsgQ_p^i then.
    delete m from MsgQ_a^i ;
else
    add m to the MsgQ_p^i
```

主服务进程失效处理过程的功能是当后备服务进程接收到来自失效监测器的怀疑主服务进程失效的消息时,启动虚拟同步处理过程 *VSC_Mcast*( )进行同步处理,*VSC_Mcast*( )返回的结果为$\{V_{i+1}(g),MsgQ^i\}$,其中,$V_{i+1}(g)$表示新的组视图,$MsgQ^i$ 表示服务进程在视图 $V_i(g)$中所传递的消息集合.其算法描述如算法 4 所示.

算法 **4**. 主服务进程失效处理过程

```
HandlPrimaCrash( )
{V_{i+1}(g) ,MsgQ^i} ← VS_Pro( );
UnHandledMsgSet ← MsgQ^i – MsgQ_a^i;
while TempMsgQ != ϕ do
    m ← GetMsgFromQ(UnHandledMsgSet);
    if  m = req_c^i or res_s^i then dlvr(m) //对消息进行处理
    else UpdateMsgQ(m)
```

endwhile

$prima \leftarrow SelectPrima(V_{i+1}(g))$;

if $a = prima$ and $MsgQ_a^i != \phi$ then

$\forall m \in MsgQ_a^i$:send(m) to dst;

$MsgQ_a^i \leftarrow \phi$

4 结束语

为了支持军事指挥系统的开发,我们在组通信开发工具 GCS[4]之上,设计实现了一个嵌套式容错服务支持系统 NFTS,该系统实现了上面所提出的算法,并且已经验证该算法是正确的.目前,我们正在 NFTS 之上开发一个军事指挥系统,其目标是当主指挥所在作战中被炸毁后,能够保证系统继续正确地运行.

References:

[1] Guerraoui, R., Schiper, A. Software-Based replication for fault tolerance. Computer, 1997,30(4):68~74.

[2] Van, Renese, R., Birman, K.P., Maffeis, S. Horus: a flexible group communication system. Communications of the ACM, 1996, 39(4):76~83.

[3] Chandra, T.D., Toueg, S. Unreliable failure detector for reliable distributed systems. Journal of the ACM, 1996,43(1):225~267.

[4] Shi, Dian-xi, Wu, Quan-yuan, Wang, Huai-min, *et al*. The research and design of group communication service in the cooperative applications. Journal of Computer-Aided Design and Computer Graphics, 2000,12(1):76~80.

附中文参考文献:

[4] 史殿习,吴泉源,王怀民,等.协同应用中组通信服务的研究与设计.计算机辅助设计与图形学报,2000,12(1):76~80.

Research and Design of the Nested Dynamic Fault-Tolerant Protocol*

SHI Dian-xi, WU Quan-yuan, WANG Huai-min, ZOU Peng

(*School of Computer*, *National University of Defence Technology*, *Changsha* 410073, *China*)

E-mail: dxshi@etang.com

http://www.nudt.edu.cn

Abstract: From the view of the software fault-tolerance, in the distributed computing environment based on the network, for the property of the military command system and bank management system and in order to satisfy the reliability requirement of these applications, based on the group communication technology and by using the process replication technology, in this paper, a nested dynamic fault-tolerant model and a dynamic fault-tolerant algorithm are proposed. When the primary service process crashes, this algorithm can dynamically select one of backup process as a new primary service process and assure the state of all backup service processes is consistent.

Key words: replication; fault-tolerance; group communication; virtual synchronization

* Received March 28, 2000; accepted July 26, 2000

Supported by the National Natural Science Foundation of China under Grant No. 69833030; the National High Technology Development 863 Program of China under Grant No.863-306-ZD02-01-2; the Youth Foundation of the National High Technology Development 863 Program of China under Grant No.863-306-QN2000-3