NTP 自主配置的自组织途径

包秀国[1)] 蒋宗礼[2)] 张 永[1)] 胡铭曾[1)]

[1)](哈尔滨工业大学计算机科学与技术学院 哈尔滨 150001)

[2)](北京工业大学计算机学院 北京 100022)

摘 要 该文研究 NTP 协议用于超大规模网络时间同步遇到的自主配置问题,NTP 自主配置是指依网络的当前状态自适应地刷新运行参数.该文将自主配置作为一类自组织过程,建立了一个生命周期模型,在 NTP 基础上设计了一个附加协议.节点通过该协议能够自动地搜索、选择和调整运行参数.仿真实验和应用表明,该方法与其它同类方法相比有更多的优点且对 NTP 的完善有参考价值.

关键词 时间同步;自主配置;自组织;服务生存性

中图法分类号 TP393

Self-Organizing Paradigm for NTP Autonomous Configuration

BAO Xiu-Guo[1)] JIANG Zong-Li[2)] ZHANG Yong[1)] HU Ming-Zeng[1)]

[1)](*School of Computer Science and Technology, Harbin Institute of Technology, Harbin* 150001)

[2)](*College of Computer Science, Beijing University of Technology, Beijing* 100022)

Abstract This paper studies autonomous configuration problem of time synchronization met when the NTP protocol is used in very large-scale networks. NTP autonomous configuration means the adaptive refreshment of the run-configuration in response to current state of networks. This paper takes the autonomous configuration as a class of self-organizing process. A life-cycle model is presented, and a special extended protocol for NTP is designed. With this protocol, a node is able to automatically search, choose and tune its running parameters. The simulations and applications show that this method has more advantages over others and it is valuable for NTP improvement.

Keywords time synchronization; autonomous configuration; self-organizing; service survivability

1 引 言

网络时间同步是网络安全、网络应用、网络管理涉及的重要问题之一,其研究的历史可追溯到计算机网络诞生之初.传统的同步研究领域主要集中在各种同步算法[1~3],最近,Ad hoc 网络、传感器网络以及计算网格中的时间同步问题已经成为新的研究热点[4~6].时间同步目前最广泛采用的协议是 NTP (Network Time Protocol)协议[7].基于 NTP 协议的网络时间同步的一般用法是:首先设置一个或几个互为备份的网络主机,称为根节点,使其本地时钟直接同步于标准时间 UTC(如通过 GPS 卫星、专用的原子钟等获得);然后根据网络结构选择一组合适的节点作为时间服务器,使其同步于这些根节点;网络上的其它节点则选择其中的一个或几个时间服务

收稿日期:2004-04-10;修改稿收到日期:2005-02-19.本课题得到国家"八六三"高技术研究发展计划项目基金(2002AA142020)和国家重点预研项目基金(413150703)资助.包秀国,男,1963 年生,博士,主要研究领域包括分布式计算、网络安全技术.E-mail:bxg@pact518.hit.edu.cn.蒋宗礼,男,1956 年生,教授,主要研究领域包括分布式计算、计算机教育.张 永,男,1980 年生,硕士研究生,主要研究方向包括分布式计算、网络安全技术.胡铭曾,男,1935 年生,教授,主要研究领域包括体系结构、网络安全技术.

器作为时间同步基准的来源.本文中称一个节点使用的时间服务器为同步源.

组建一个时间同步网时,通常需要在整体规划的基础上,通过手工方式静态地为运行于每一个网络节点上的时间同步进程(常称为 NTP 进程)选择配置一些参数,如:同步源个数、同步源 IP 地址、轮循间隔、精度要求等,使时间同步网构成一个以根节点为树根的层次型网络,且符合设计的其它指标.这种以人工配置为主的应用模式对于一般的中小型网络以及 Internet 上的时间同步十分适用.但对于军事、金融、通信等领域的超大规模网络,这种用法是不方便的,也是不切实际的.比如,当网络的局部用途发生变更,物理拓扑结构调整,网络规模扩大或缩减,以及随机故障、恶意攻击等因素频繁影响同步网络运行时,该种应用模式使得 NTP 受到巨大的挑战!这一新出现的问题被称之为 NTP 自主配置(NTP autonomous configuration)[7].另一方面,具有开放层次式系统特点的时间同步网络存在健壮性(或生存性)缺陷,易遭受恶意攻击[8,9].

时间同步网的参数自主配置问题最早在时间同步协议 DTSS 中涉及[10],Mills 等曾提出一种基于组播的方法①,但组播技术的固有缺陷使得其应用受到很大的限制,如需要相关路由器支持组播报文转发等[11].文献[12,13]将同步源作为一种资源,通过分布式资源搜索实现配置参数的自动选择,但缺少理论模型分析.本文进一步基于自组织原理,将时间同步网的自主配置看作是一类自组织过程.我们用生命周期模型描述这一过程,称为自组织同步模型.基于该模型,通过在 NTP 基础上设计一种附加协议,称为 NTP 自组织自主配置协议 SANP(Self-organizing Autonomous-configuration for NTP Protocol),实现 NTP 参数的自主配置.

本文第 2 节首先给出自组织时间同步模型;第 3 节简要介绍 SANP 协议的设计目标;第 4,第 5 节侧重对 SANP 中的服务生存性、消息转发振荡进行深入讨论;第 6 节给出其中的一些仿真结果和分析;最后为结论.本文以同步源 IP 地址为例讨论.

2 自组织同步模型

自组织是一种普遍存在的现象,如生物病毒自我复制进化等.自组织原理的核心思想是:遵循一组简单规则的多个个体之间能够自主地、异步地相互作用,整体显现出反应或自适应方式的特性[14].它是通过个体微观行为实现系统宏观目标的一种方法.时间同步也可以看作是一种自组织现象.即,时间同步网是以每个 NTP 进程作为一个节点(个体),以根节点为核心,按一定规则自主地、异步地相互作用,以根节点时钟为基准修正节点本地时钟作为整体目标的一类自组织网络.应用自组织原理解决 NTP 自主配置问题的基本思路是:每个节点独立地遵循相同的一组规则(即 SANP 协议),自动地感知节点周围环境的变化,确定调整参数的时机并选择合适的参数,使得形成的时间同步网整体上具有期望的特性,如:各个 NTP 进程是否可以实现参数配置优化,时间同步网对恶意攻击的抵抗能力等.

首先建立以下模型.用有向图 $G=\langle V, NS, WS, E\rangle$ 表示时间同步网,V 是网络中一些节点的集合,$NS \cup WS = V$, $NS \cap WS = \Phi$, $E \subseteq NS \times NS$,其中

本地时钟直接同步于标准时间 UTC 的节点称为根节点;

$\forall v \in NS$,v 为根节点或者已经或能够间接同步于基准时间的节点,称为内节点;

$\forall v \in WS$,v 为需要但尚不能同步于基准时间的节点,称为外节点,又称为孤立节点;

如果节点 u 在节点 v 的 NTP 进程配置文件中是一个时间服务器,则称 u 为 v 的同步源,表示为有向边 (u,v);

$\forall (u,v) \in E$,表示 u 是 v 的一个同步源,二元组 (u,v) 称为 v 的同步关系.

有向图 G 具有以下基本特性:

(1)根节点为永久同步源,根节点的入度为 0(忽略与标准时间 UTC 的直接同步关系).

(2)任何内节点均可以被选为同步源;一个节点可以同时拥有多个同步源,任何内节点至少拥有一个同步源.

(3)当一个内节点丢失其所有同步源时成为一个外节点.

(4)一个节点的全部同步源数量等于节点的入度,一个节点直接服务的节点个数,即以该节点为同步源的同步关系个数等于节点的出度.

时间同步可以抽象为图 G 中的节点按统一的生命周期活动的过程.如图 1,设 v 为任意一个节点,v 的同步动作从时间上可分为三个阶段:创建期、生存期和消失期.在创建期,v 由一个普通网络

① http://www.eecis.udel.edu/~mills/autocfg.html

节点成为 G 的一个外节点；当 v 处在生存期时，v 由一个外节点通过自主配置成为内节点，从而其本地时钟同步于基准时间；在消失期，v 离开同步网 G 成为普通网络主机，结束一个生命周期. 在一个大规模网络中，从整体上看，每一时刻不断有新的节点进入同步网 G，同时又有一些节点离开同步网.

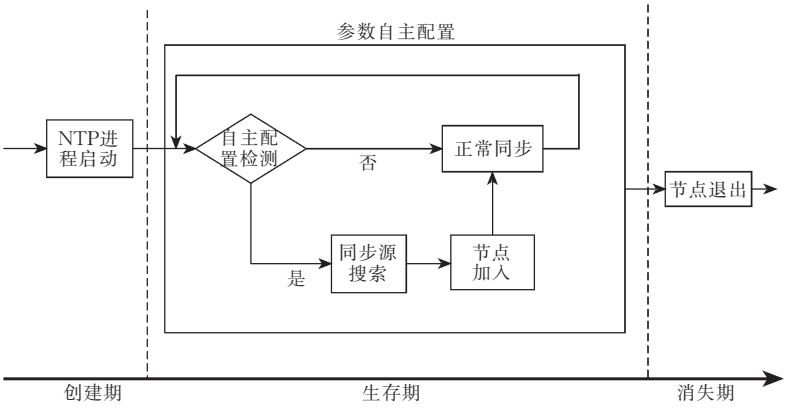


图 1　自组织同步生命周期模型

为分析图 1 中的各种操作对同步网络拓扑的作用，设 $G=\langle V,NS,WS,E\rangle$ 为动作前的拓扑图，$G'=\langle V',NS',WS',E'\rangle$ 为动作后的拓扑图，节点在生命周期内的不同动作对网络拓扑的影响主要有以下几个方面：

(1)NTP 进程启动. 物理网络中的一个主机 v 因某种服务的需要启动含有自主配置能力的 NTP 进程，此时图 G 变成 G'，其中 $V'=V\cup\{v\}$，$WS'=WS\cup\{v\}$.

(2)自主配置检测. 节点周期性检测有关状态变量，确定是否需要调整 NTP 进程参数. NTP 进程启动后，自动进入配置状态检测，此时图 G 不发生变化：$G'=G$.

(3)同步源搜索. 一个外节点 v 在 G 中搜索并筛选合适的一个内节点子集 $APS(v)=\{u_1,u_2,\cdots,u_k\}$ 的操作，k 为正整数. 当自主配置检测动作结果为真时，启动同步源搜索操作，此时图 G 也不发生变化：$G'=G$.

(4)节点加入. 节点 v 调整 NTP 进程参数，与 $APS(v)$ 中每个节点建立同步关系，此时，$NS'=NS\cup\{v\}$，$WS'=WS-\{v\}$，$E'=E\cup\{(u_1,v),(u_2,v),\cdots,(u_k,v)\}$，节点 v 由外节点变成内节点.

(5)正常同步. 节点 v 与其所有同步源交互，应用标准 NTP 协议，使其本地时钟同步于 $APS(v)$ 中的一个最佳同步源的时钟.

(6)节点退出. 一个内节点因某种原因，导致同步进程终止或无法正常同步的动作. 导致节点退出的原因是多方面的：可分为节点主动退出和被动退出两种情形. 如 NTP 进程的人为关闭、对应主机停机等为主动退出情形；节点 v 的所有同步源退出后 v 也必须退出为被动退出情形. 值得指出的是：由于恶意攻击，少数几个甚至单个节点的退出可能导致与其有直接或间接同步关系的大量节点的被动退出，即一个节点的退出可能导致网络结构出现“雪崩效应”形式的变化. 以下分布式递归算法 left(v) 描述了节点退出的一般情形.

left(v)
{$NS'=NS-\{v\}$; $WS'=WS\cup\{v\}$;
$E'=E-\{(u_1,v),(u_2,v),\cdots,(u_k,v)\}$;
if v 不是任何其它节点的同步源 then return
else {for $\forall u,(v,u)\in E$, do $E=E-(v,u)$;
if $\forall w\in NS$ 均有 $(w,u)\notin E$ then left(u)}

3　SANP 协议

基于以上模型，我们对应设计了一个 NTP 基础上的附加协议 SANP[9]. 它主要定义节点在需要调整 NTP 进程运行参数时的规则，节点在正常同步阶段仍遵循传统的 NTP 协议. SANP 协议设计的主要目标是：

(1)零人工配置. 节点自动检测当前运行参数是否合适，自适应确定和调整新的运行参数；

(2)服务可生存. 服务生存性指系统在遭受攻击、出现故障或人为失误等缺损情况下仍可以及时提供基本服务的能力[15]. 在时间同步系统中，当部分节点因随机故障或遭受恶意攻击失效后，给其它节点的服务受到的影响应最小化；

(3)兼容 NTP. 使用包括 SANP 协议的节点与仅使用传统 NTP 协议的节点可运行在同一时间同

步网络中;

(4)低开销. 运行 SANP 协议应具有较低的资源消耗.

SANP 协议设计包括许多方面,如报文格式设计、同步源 IP 地址等参数作为一种分布式资源目标的搜索效率和选择优化方法等. 以下侧重讨论其中的两个问题:如何提高在恶意攻击下的服务生存性以及分布式资源搜索中请求报文的转发振荡消减.

4 服务生存性

当时间同步网络因恶意攻击等原因分裂为多个子网时,除与根节点处于同一子网的节点可以正常同步外,其它节点的服务功能将丧失. NTP 协议采用冗余技术提高服务的生存性或健壮性. 即一个节点允许同时配置多个同步源,只要其中任何一个同步源正常运行,则该节点可以正常同步. 冗余方法是以牺牲宝贵的网络资源为代价的,同时它也无法抵抗恶意攻击. 比如,当攻击使同步网中某一层节点全部失效后,则多数节点的服务将停止. 总之,基于传统 NTP 提供的服务生存能力较差,其中的一个原因是由于节点没有自愈能力. 在 SANP 协议中,节点具有在线自愈功能.

4.1 自愈方法

自愈操作的第一步是如何检测自愈状态. 当一个节点与根节点不在同一子网时,根据 NTP 协议,节点的层级变量(Stratum)自动变为协议常数 STRATUM_UNSPEC(0～255 之间,缺省取 16),表明节点已不能正常同步. 节点可利用层级变量的这种变化,准确判定自愈操作的状态和时机.

自愈操作的第二步是在检测到自愈状态后,与根节点所在子网如何重新建立连接(Rewiring). 显然,存在许多自愈方法. 一种可以降低自愈开销的直观方法是:非根节点所在子网的一些节点增加几个与根节点所在子网的连接即可,其它节点维持原有邻接关系. 然而,这种表面上简单且效率很高的方法在 SANP 中却难以实现. 其原因是:如果采用这种方法,自愈节点之间首先必须协商选择一些"代表节点"去与根节点所在子网建立连接. 可以推知,这种协商过程不仅增加协议的复杂性,同时也增加额外的通信开销. 我们使用一种简单有效的方法.

当一个自愈节点发现自愈状态后,并不是立即而是随机延迟一段时间后才执行自愈操作. 换句话说,当攻击导致一部分节点失效后,并不是所有自愈节点同时,而是陆续地进行自愈操作;即开始时只有少量节点进行自愈操作,随着网络自愈的进展,更多的节点开始自愈操作. 这样做的理由是:被割裂子网中只要有一个节点完成自愈操作,同一子网的其它节点,可维持原有连接自动恢复正常状态;节点的这种"侥幸"可以减少自愈开销,同时不需要复杂的协商.

4.2 自愈时间

在时间同步网中,以上自愈方法的采用将产生一种复杂动态过程. 以下简要分析自愈时间. 设网络中单个节点完成自愈操作所需的最长时间为 a,网络自愈时间为 T,我们以 a 为时间片单位,如果网络在经 m 个时间片后完成自愈,则 $T \leq ma$. 在这种假定下,网络自愈时间的估计等价于计算时间片数 m.

当每个自愈节点在检测到自愈状态后立即进行自愈操作时,则经一个时间片后网络完全自愈,网络自愈时间为 $T \leq a$. 这种方式的优点是自愈速度快;但当攻击强度较大,即自愈节点数量较多时,由于自愈操作的并行性,自愈产生的负载很大,会出现通信尖峰. 我们将节点在检测到自愈状态后的活动时间以时间片 a 为单位划分,设置一组 0/1 二元随机变量 Z^k 控制节点自愈过程,$k=1,2,3,\cdots,k$ 为时间片编号. 仅当 $Z^k=1$ 时,节点开始自愈. 通过调整在不同时间片进入自愈操作的概率 $P(Z^k=1)=p_1$, $p_2,\cdots$ 控制网络自愈过程. 比如,假定希望控制网络最大自愈时间片数为 L,则只要 $P(Z^L=1)=1$,则网络在至多 L 个时间片内完全自愈,网络的最坏自愈时间 $T=L\times a$.

5 转发振荡与消减

5.1 转发振荡

分布式资源搜索主要通过请求转发实现,最常用的算法为广播(Flooding). 即任何节点对受到的资源搜索请求在所有邻居中扩散,直到控制扩散范围的跳数 TTL 为 0. 在请求转发过程中,存在同一请求消息被重复转发处理的现象.

先从最简单的例子讨论. 假定一个消息 Q 进入如图 2(a)和图 2(b),$Q.TTL=4$,转发算法为 Flooding. 则在没有冗余消息处理情况下不同时段产生的消息数量如图 3.

一般地,对于由 N 个节点组成的闭环,如图 2(c),不失一般性,假定一个消息 Q 进入任一个节点 Λ,$\Lambda.left$,$\Lambda.right$ 分别表示其左右两个邻居节点,

$Q.TTL=cc$. 由此产生的消息总量 $S(\Lambda,cc)$ 等于 $\Lambda.left,\Lambda.right$ 分别收到 TTL 为 $cc-1$ 的请求产生的消息总量加上 2. 用递推方程表示为

$$S(\Lambda,cc)=2+S(\Lambda.left,cc-1)+S(\Lambda.right,cc-1) \tag{1}$$

由于在闭环中任何节点的结构特征是相同的，产生的消息量也必然是相等的，故方程式(1)可简化表示为

$$S(\Lambda.left,cc-1)=S(\Lambda.right,cc-1)=S_{cc-1},$$
$$S_{cc}=2+2S_{cc-1},\quad S_1=2.$$

求解该递推方程式为

$$S_{cc}=2^1+2^2+\cdots+2^{cc}=2^{cc+1}-2 \tag{2}$$

有意义的是，S 与环的节点个数无关，只与 TTL 有关.

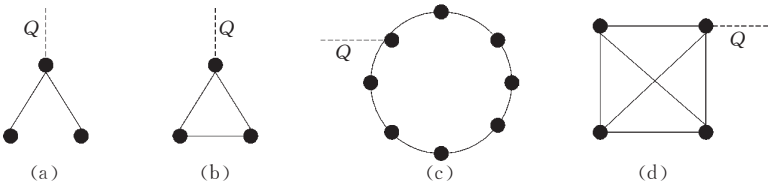


图 2　转发振荡示例网络

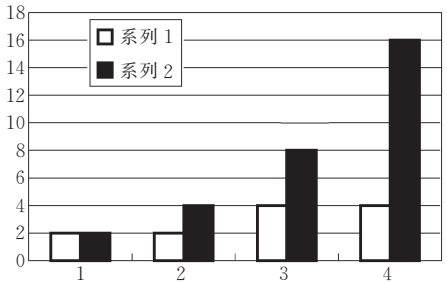


图 3　在图 2(a)(系列 1)和 2(b)(系列 2)中转发消息 Q 的数量

对于由 N 个节点的完全连通图，如图 2(d)，同理推导如下：任一节点有 $N-1$ 个邻居节点，由于结构的对称性，每一个邻居节点产生的消息总量是相同的，故

$$S_{cc}=N-1+(N-1)S_{cc-1},\quad S_1=N-1 \tag{3}$$

求解该递推方程式(3)

$$S_{cc}=(N-1)+(N-1)^2+(N-1)^2+\cdots+(N-1)^{cc}=(N-1)^{cc+1}-2 \tag{4}$$

一般地，假定节点 Λ 有 L 个邻居，分别标志为 $\{0,1,\cdots,L-1\}$，这些邻居节点的度数分别为 d^0，$d^1,\cdots,d^{L-1}$，节点 Λ 在收到一个请求后产生的消息总量可表示为

$$\begin{aligned}S(\Lambda,cc)&=L+S(0,d^0,cc-1)+S(1,d^1,cc-1)+\cdots+S(L-1,d^{L-1},cc-1)\\&=L+\sum_{k=0}^{L-1}S(k,d^k,cc-1)\end{aligned}$$

根据上式(3)～(5)，对于任一节点，采用 Flooding 算法时产生的消息总量为 $O(N^{cc})$，其中，N 为网络节点总数，cc 为请求的 TTL 值.

不难看出，在没有冗余消息处理情况下，Flooding 算法将导致严重的消息冗余转发，我们称之为消息转发振荡. 而且，这一现象不仅出现在 Flooding 转发算法中，所有采用转发原理的算法均会产生类似的问题！

重复转发问题在实际网络中，由于存在网络延迟的不一致和随机性，比上述的分析更为复杂.

5.2　消减方法

消减转发振荡的基本方法如下：在每个节点中设置一个带消息池的过滤器，该过滤器对接受的消息进行过滤处理，对判定为重复消息的(不一定完全准确，见以下分析)丢弃；反之，按搜索算法指定的邻居转发该请求. 过滤器中采用的转发振荡消减算法的基本原理如图 4 所示.

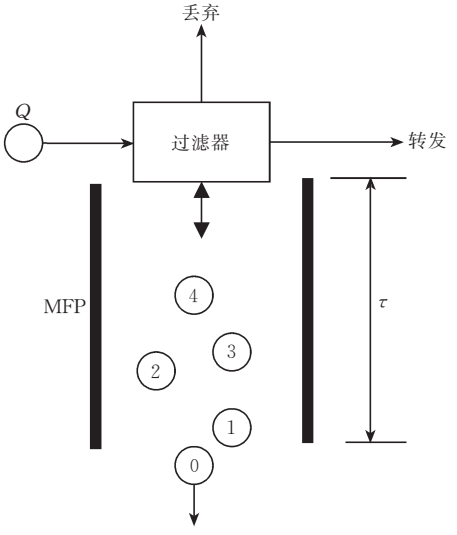


图 4　转发振荡消减原理

图中，MFP(Message Forward Pool)为一个消息池，其长度为 τ，其物理意义是任何消息在 MFP 中的生命期. 过滤器对任何收到的请求在接受时附加相应的时间戳，当一个新的请求 Q 到来时，首先检查其在 MFP 中是否存在，如果不存在，则执行转

发操作，并将其放入 MFP，进入 MFP 的消息以 $1/\tau$ 的速度穿过 MFP，即任何一个进入的消息在 MFP 中只能停留的时间为 τ. 假设一个消息 $Q0$ 进入 MFP 的时间戳为 $Q0.\mathrm{STP}$，则在 $Q0.\mathrm{STP}+\tau$ 时刻后，该消息从 MFP 中消失.

消息池的长度 τ 的选择比较复杂. 当 τ 较大时，对消减转发振荡有利，但其代价有两方面：一是节点需要的缓存和处理开销将随着 τ 的增大而增加；另一方面，当 τ 过大超出相同请求发出的最短间隔时间时，可能导致正常的请求被误判为重复消息而丢弃，从而影响搜索操作. 显然，τ 越小，转发振荡消减的效果越差，$\tau=0$ 时过滤器没有任何作用. 理想情况下，参数 τ 的选择应当在 $T1<\tau<T2$. $T1$ 为产生重复转发的最小通信延时，$T2$ 为同一请求发生的最短间隔时间. 由于不同网络差异，传输延迟分布的随机性，不同请求到达一个节点历经的路径差异等原因，合理的 $T1$，$T2$ 难以获得；τ 过小将降低振荡消减的效果，τ 太大将可能使正常的消息转发被丢弃，影响搜索结果. 总之，完全消除转发振荡是不可能的，我们采用实验方法给出 τ 选择的一些经验方法.

6 仿真实验

我们采用 Parsec[17] 进行仿真实验验证. Parsec 是由 UCLA 研究的一种基于 C 语言的离散事件仿真工具. 实验主要模拟了节点按 SANP 协议进行时间同步的过程. 节点模拟为 Parsec 中的逻辑进程 LP[17]. 以下为其中的一些与本文相关的实验结果和分析.

6.1 消息冗余度

为探索转发振荡消减中过滤器参数 τ 的选择方法，在随机生成的网络拓扑中，假定网络的链路延迟分布服从正态分布规律，延迟集中在 100ms 左右，实验的搜索算法为 Flooding. 为测试过滤器的作用，我们假定搜索的目标资源数量稀疏，总量仅为 20 个（资源数量少，转发振荡越严重），在网络中随机分布.

Flooding 算法只有一个参数 TTL，TTL 越大，搜索成功率越高，同时产生的重复消息数量也越多. 由于理想情况下，一个请求消息应在访问的每个节点被转发一次，故我们定义以下指标，称为消息冗余度 RRM，用于衡量过滤器消减转发振荡的程度.

$$RRM=\frac{\text{消息总数}-\text{访问节点总数}}{\text{消息总数}},$$

易知，$0\leq RRM<1$，$RRM=0$ 时，没有任何重复消息. 实验的过滤器以 τ 为参数，τ 为 0 时对应搜索算法中过滤器没有作用. 具体实验方法如下：对不同的 (τ,TTL) 组合，仿真搜索过程，统计消息总数和访问节点总数，并计算 RRM. 其中，$0\leq\tau\leq500$，τ 的步长取 50，TTL 取 4～9. 实验结果如图 5 所示.

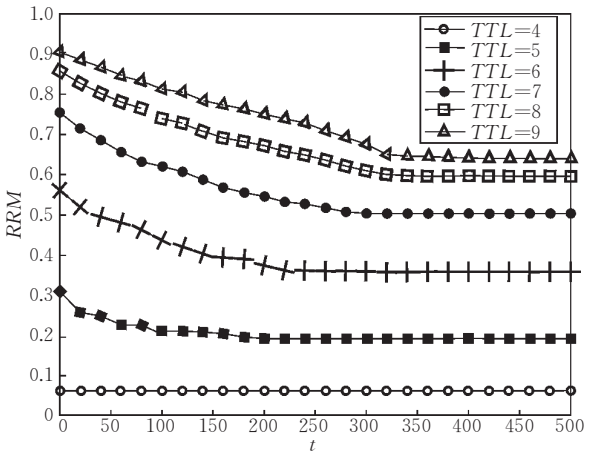


图 5 转发振荡消减效果

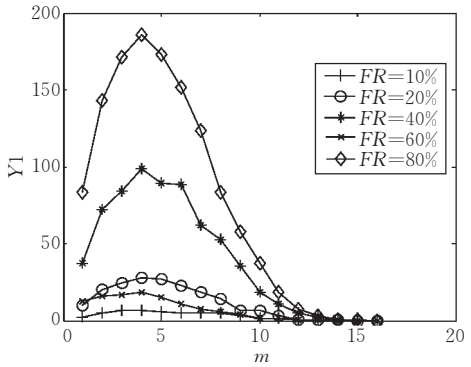
观察实验数据，可得到以下结果：(1)RRM 随着 τ 的增加而下降，这与第 5 节的理论分析是一致的. (2)当 $TTL\leq4$ 时，转发振荡消减的结果不十分显著，但随着 TTL 的增大，转发振荡消减的作用显著. 如当 $TTL=9$ 时，在没有过滤器时，重复消息的数量大约是访问节点个数的 10 倍左右，而采用过滤器后，最大可降到 2.5 倍左右. (3)无论 τ 如何选择，转发振荡是不能完全消除的. 从图中可以看出，在所有的实验案例中，当 τ 超过一定阈值时，RRM 趋向一个大于 0 的恒定值，这验证了第 5 节的另一个分析结果：重复消息存在的原因是多方面的，仅靠过滤器是不能完全消除的. (4)存在一个 τ 的上限阈值. 比较图中的拐点可以看出，随着 TTL 的增大，拐点右移，其规律基本上与 TTL 成线性比例. 这一特性可用于选择合适的 τ 值.

6.2 自愈时间片数

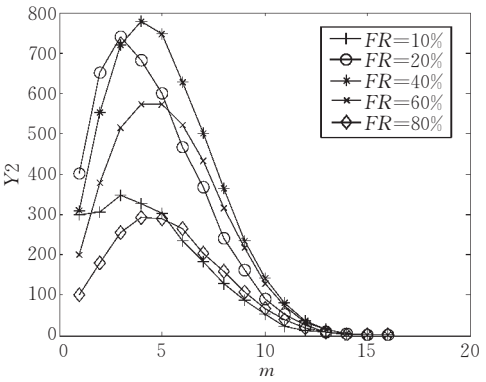
我们用网络受到假定的攻击后自愈时间片数评估 SANP 的服务生存性. 在生成的网络拓扑中，假定有两种典型的失效模型：(1)随机故障. 即网络中每一个节点失效的概率是相同的；(2)恶意攻击. 恶意攻击的种类繁多，这里的恶意攻击仅是指故意攻击节点度数高的节点. 失效的强度统一用失效率 FR 衡量，FR 指随机故障或恶意攻击导致的失效节点数占网络总节点数的百分比.

实验目的是在不同失效模型和失效强度下，检测当网络完全自愈需要的时间片数 m 以及实际进行

自愈操作的节点数. 实验中，每个节点在不同时间片进入自愈操作概率均按线性规律增加，$P(Z^k=1)=0.05,0.10,0.15,\cdots,L=20$，即控制自愈过程在 20 个时间片内完成. 实验结果如图 6 所示.



(a) 随机故障时的自愈过程



(b) 恶意攻击时的自愈过程

图 6　两种失效模式下的自愈过程

观察以上实验数据表明：

(1)概率 $P(Z^k=1)$ 是影响自愈时间的主要因素. 当 $P(Z^k=1)$ 一定时，网络自愈时间确定，自愈时间与失效模型和失效强度基本无关.

(2)在相同的失效率下，恶意攻击导致的实际进行自愈操作的节点数 $Y2$ 远高于随机故障模型下的对应值 $Y1$.

(3)$P(Z^k=1)$ 也是决定网络自愈通信尖峰的主要因素. 如图 6 中，这种情况下进行自愈操作的节点数的高峰均在第 5 时间片左右.

(4)实际自愈节点数变化规律在不同失效模式下是不同的. 恶意攻击下实际的自愈节点数 $Y2$ 随着失效率 FR 上升而增加，当达到一定程度后转为下降，这是由于恶意攻击强度很大时，网络中剩余节点数较少造成的. 随机故障失效模式下，由于失效节点选择的随机性而没有类似的规律，如图 6(a).

(5)SANP 协议的自愈时间是可控的，且自愈时间与失效模型基本无关，表明 SANP 协议具有良好的服务生存性.

7　结　论

本文给出了超大规模网络时间同步中必然遇到的 NTP 自主配置问题的一种解决方法. 该方法摈弃了 NTP 时间同步网的静态层次型结构，建立了一个自组织时间同步模型，设计了对应的协议，使静态的时间同步网变为一种动态自组织网络. 对其中的服务生存性、消息转发振荡问题进行了较深入的讨论.

本文的方法已成功应用于解决一个具有数千个节点的超大规模网络时间同步问题，并计划用于解决数万个节点规模网络的时间同步问题，为保障我国信息安全发挥了重要作用. 仿真和应用表明：该方法不仅对于 NTP 协议的完善有参考价值，而且具有重要的实用价值.

致　谢　原北京工业大学计算机学院的徐斌斌同学参与文中的一些早期实验，在此表示感谢！

参　考　文　献

1　Cristian F.. A probabilistic approach to distributed clock synchronization. Distributed Computing, 1989, 3: 146～158

2　Arvind K.. Probabilistic clock synchronization in distributed systems. IEEE Transactions on Parallel and Distributed Systems, 1994, 5(5): 474～487

3　Ramanathan P., Shin K. G., Butler R. W.. Fault-tolerant clock synchronization in distributed systems. IEEE Computer, 1990, 23(10): 33～42

4　Zhao Ying, Zhou Wan-Lei *et al*. Self-adaptive clock synchronization for computational grid. Journal of Computer Science and Technology, 2003, 18(4): 434～441

5　Sivrikaya F., Yener B.. Time synchronization in sensor networks: A survey. IEEE Network, 2004, 18(4): 45～50

6　Sichitiu M. L., Veerarittiphan C.. Simple, accurate time synchronization for wireless sensor networks. In: Proceedings of Wireless Communications and Networking, Atlanta, Georgia, 2003, 1266～1273

7　Mills D.. A brief history of NTP time: Confessions of an Internet timekeeper. ACM Computer Communications Review, 2003, 33(2): 9～22

8　Berthaud Jean-Marc. Time synchronization over networks using convex closures. IEEE/ACM Transactions on Networking, 2000, 8(2): 265～277

9 Bao Xiu-Guo. Study on survivability enhancing techniques for open hierarchical systems[Ph. D. dissertation]. Harbin Institute of Technology, Harbin, 2004(in Chinese)
(包秀国. 开放层次式系统的生存性增强技术研究[博士学位论文]. 哈尔滨工业大学,哈尔滨,2004)

10 Digital Equipment Corporation. Digital time service functional specification. 1989

11 Sharma P., Perry E., Malpani R.. Network. IP multicast operational network management: Design, challenges, and experiences. IEEE Network, 2003, 17(2): 49～55

12 Jiang Zong-Li, Xu Bin-Bin.. Automatic configuration in NTP. High Technology Letter, 2003, 9(4): 70～73

13 Bao Xiu-Guo, Hu Ming-Zeng *et al*. A self-organizing timekeeping network. Journal of China Institute of Communications, 2004, 25(1): 150～157(in Chinese)
(包秀国,胡铭曾等. 一种自组织时间同步网. 通信学报, 2004, 25(1): 150～157)

14 Grover W. D.. Self-organizing broadband transport networks. Proceedings of IEEE, Special Issue on Communications in the 21st Century, 1997, 85(10): 1582～1611

15 Isher J., Linger R.. Survivability: Protecting your critical systems. Internet Computing, 1999, 3(6): 55～63

16 Baryshnikov Y., Coffman E. *et al*. Flood search under the California Split rule. Operations Research Letters, 2004, 32(3): 199～206

17 Bagrodia R. *et al*. Parsec: A parallel simulation environment for complex systems. IEEE Computer, 1998, 31(10): 77～85

BAO Xiu-Guo, born in 1963, Ph.D.. His research interests include distributed computing and network security technology.

JIANG Zong-Li, born in 1956, professor. His research interests include distributed computing and computer education.

ZHANG Yong, born in 1980, M. S. candidate. His research interests include distributed computing and network security technology.

HU Ming-Zeng, born in 1935, professor. His research interests include computer architecture and network security technology.

Background

The Network Time Protocol (NTP) is widely used to synchronize computer clocks. However, management and configuration of large-scale timekeeping networks under this protocol have become almost unworkable. A means is required to automatically configure the hierarchy in response to changing topology, failures and malicious attacks. This paper presents an effective way to achieve autonomous configuring for any node in timekeeping network. This research is supported by the project named "Large-scale Network Intrusion Detection"(National High Technology Research and Development Program (863 Program) under grant No. 2002AA142020) and the project named "Large-scale Network Security Management Techniques"(No. 413150703).