

分层分布式网络故障监视算法研究

张新^{①②} 常义林^① 孙方涛^① 沈中^①

^①(西安电子科技大学综合业务网国家重点实验室 西安 710071)

^②(西安邮电学院电信系 西安 710061)

摘要: 网络管理首先要有效地获得监视信息, 现有的简单网络管理协议(SNMP)网络管理方式中, 一般采用管理者轮询代理以及代理主动向管理者报告事件通知的方法监视网络, 它们都占用较多的网络带宽和系统资源。为了减少网络监视带来的开销, 该文基于分层分布式网络管理体系结构, 提出了一种动态网络监视算法。算法根据轮询和事件通知获得的网络状态数据, 预测发生告警的可能性。如果没有告警的可能性, 则延时轮询, 同时采用告警滞后机制过滤冗余告警信息, 保证了既不漏报告警, 又有效地降低了网络管理的通信负担。仿真结果表明, 该文的监视算法有效地减少了轮询次数, 过滤了重复告警信息, 降低了管理信息占用的带宽, 提高了网络管理的有效性。

关键词: 网络监视; 故障管理; 轮询; 事件通知

中图分类号: TP393.07

文献标识码: A

文章编号: 1009-5896(2007)04-0771-05

A Fault Monitoring Algorithm for Hierarchical Network

Zhang Xin^{①②} Chang Yi-lin^① Sun Fang-tao^① Shen Zhong^①

^①(State Key Lab. of Integrated Service Networks, Xidian Univ., Xi'an 710071, China)

^②(Dept. of Electronic and Information, Xi'an Institute of Posts and Telecommunications, Xi'an 710061, China)

Abstract: One mission of network management is to monitor the network effectively. The monitoring methods in Simple Networks Management Protocol (SNMP) are polling and trap. Both of them consume the system resources and network bandwidth. Based on the hierarchical network architecture, a dynamic monitoring algorithm is proposed so as to minimize the management overhead. By analyzing the probability of the alarm occurrence with the sampled network information, the proposed algorithm prolongs the interval between two consecutive polling when no alarm could occur. It also reduces the redundant alarm information resulted from the signal fluctuation. This method significantly reduces the amount of monitoring traffic and guarantees to detect the abnormal event. The simulation results show that the proposed monitoring scheme reduces both the number of polling and the cost of the network management, consequently improves the network performance.

Key words: Network monitoring; Fault management; Polling; Trap

1 引言

网络监视是网络管理的基础, 它要求采集有关网络配置状态和行为的信息, 用以分析网络性能和网络故障。网络监视作为一种必备的手段, 一方面提供了保障网络正常运行的参数, 另一方面, 监视信息流量叠加在正常的网络流量之上, 增加了网络负担。人们力图研究开销小而有效的网络监视数据获取方法和网络管理信息传输协议; 提高网络管理的有效性以及故障监视的可靠性并有效地降低监视信息传输的负担, 以保证网络的正常业务传输。

现有的SNMP(Simple Network Management Protocol)网络管理系统(包括RMON: Remote network MONitoring),

网络监视数据传送的主要方法是轮询和事件通知。它们都占用较多的系统资源, 增加网络开销。目前的网络管理特别是对于像移动 Ad hoc 网络^[1](Mobile Ad-hoc NETwork: MANET), 其网络带宽受限, 这就更加要求选择合适的网络监视策略。文献[2]分析了SNMP的 MIB(Management Information Base), 采用一种可靠的多播技术以掌握监视信息。在分层分布式网络管理技术^[1, 3, 4]中, 采用了阈值监视和动态轮询技术进行网络监视。但文献[3, 4]没有充分利用事件通知中携带的信息, 并存在着告警信息重复传输的问题。本文结合SNMP的轮询和事件通知机制, 在MANET的分层管理基础上, 提出了分布式动态监视算法, 分析了被管对象属性及其关联关系, 根据收集到的被管对象状态变化情况, 确定网络监视策略, 如轮询周期等; 同时利用告警滞后机制过滤和防止重复信息的多次上报, 以减小网络管理开销, 提高了网络监视的利用率, 优化了网络管理性能。

2005-08-31 收到, 2006-02-24 改回

综合业务网理论与关键技术国家重点实验室基金(00JS63.2.1.DZ01)资助课题

全文分为 5 节,第 2 节建立分层分布式的网络管理模型,第 3 节提出了一种网络监视算法并分析性能,第 4 节给出了采用本文算法的仿真结果和性能对比,最后是结束语。

2 监视模型

图 1 是本文采用的分层分布式网络管理模型,它是由管理者(Manager)、群首(Cluster Head)和代理(Agent)三级组成。管理者 M 是全网管理中心,群首 CH 是分群子网的管理者,代理 A 管理所在节点及周边链路等,多个代理形成一个群并由群首管理,所有群首由网络管理者管理。群首是二级管理域的核心,其监视范围和监视功能由管理者 M 委任,群首驻留有 RMON 探测技术,可对所辖的管理域监视,并把对整个网络有影响的数据提交给管理者 M。

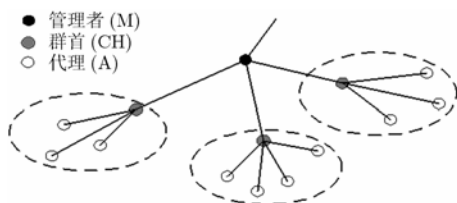


图 1 分层分布式网络管理模型

每个 CH 监视所属群中的哪些对象以及如何监视这些对象属性值,是通过驻留在 CH 上的 RMON 控制表确定的。控制表中包含有这样几个主要内容,以 OID(Object Identifier)表示的被管对象,数据采集间隔时间,总的采集次数。如果要对多个对象监视,则需要在控制表中产生相应的多个控制项,CH 按照控制表规定的内容监视网络资源,生成的控制表可以由上级管理者 M 和本分群子网的 CH 修改,即修改监视对象和数据采集周期等。这种通过修改控制项来确定监视参数的方式为动态监视提供了条件。同时,每个分群子网内的控制表也是相互独立的,即每个 CH 是对所辖子网的资源独立管理,意味着不同的分群子网独立配置,这有利于实现分布式的监视。

3 监视算法

网络故障监视是网络管理者通过轮询网络状态和由代理主动发送事件通知(trap)来发现网络故障的一种网络管理方法。网络故障监视算法的目的是通过确定被管对象的监视范围以及监视策略,在不漏检网络故障的同时,降低轮询以及重复报告带来的网络管理开销。

对于传统的网络管理协议规范,管理者一般采用固定周期轮询的方式,获得所辖域内被管对象的状态信息,同时,代理也可将一些重要的网络异常信息主动报告给管理者。

由于被管对象的状态数据是随着网络性能而动态变化的。对于宽带网络可以允许采用固定周期的轮询方式监视网络,但对于窄带网络,采用固定周期的常规监视方法显然会过多地占用带宽,浪费资源,所以一般采用轮询与事件通知相结合的动态网络监视策略,以降低监视信息对网络资源的占用。一般的轮询与事件通知相结合获取管理信息的方式如

下:在 CH 端,管理者可以执行固定周期的正常轮询,此间,一旦收到由代理 A 发送的被管对象异常的事件通知时,立刻开始轮询。

通常情况下,网络处于正常状态。设网络初始正常运行时,网络管理的固定轮询周期为 T_0 ,经 KT_0 过后,CH 的 MIB 积累了一些数据,根据已有的数据分析如下:

设图 1 网络管理系统中网络状态采样的时间集合为 $T = \{t | t = t_i, i = 1, \dots, j\}$,而 CH 对代理轮询的时间集合为 $T_p (T_p \subseteq T)$ 。某个 CH 在轮询时刻 t 同时获得所辖分群子网内的 n 个被管对象状态 $X_t = \{x_{1,t}, x_{2,t}, \dots, x_{i,t}, \dots, x_{n,t}\}$, $x_{i,t}$ 表示被管对象 i 在时刻 t 的状态值。定义 f_t 为告警函数,它与 t 时刻的被管对象状态 X_t 有关, $f_t = (x_{1,t}, x_{2,t}, \dots, x_{n,t})$,被管对象状态之间的关系可以是加性的也可以是乘性的,乘性的关系可以通过取对数的方式变为加性的。为了简化讨论,设加权系数 $c_{i,t} = 1$, $f_t = \sum_{i=1}^n (c_{i,t} \cdot x_{i,t})$,例如,它可以表示子网中各个节点发出的业务流量与整个子网发出的流量的关系。

告警条件就是判断告警函数是否超过告警阈值。设告警函数 $f_t = \sum_{i=1}^n x_{i,t}$ 有告警阈值 TH_u 和预警阈值 TH_d ,设置 TH_u 和 TH_d 是为了防止由于监视变量数据的波动而产生的冗余告警, $f_t < TH_d$ 时,CH 等待轮询; $f_t \geq TH_d$ 时,进入预警状态。进入预警状态的 CH 是否向上级产生告警这可由驻留在 CH 处的 RMON 探测器运行 RMON 规范中的告警滞后机制实现。如果当前告警函数的值大于等于告警值 TH_u 而上次轮询后告警函数的值小于 TH_u ,则称告警函数的值超过了上限阈值,告警条件为如果变量值是增加的,且告警函数 f_t 超过上限阈值,则告警。监视变量的变化趋势是通过提取驻留在 CH 上的告警表中的标量对象 `alarmStartupAlarm` 的值得到的。

代理的事件通知的触发机制有两种:一种是当被管对象在 t 时刻的状态值 $x_{i,t}$ 超过了 MIB 中规定的相应阈值,代理 A 向 CH 发送事件通知;另一种是当被管对象在 t 时刻的状态值改变量 $(x_{i,t} - x_{i,t-1})$ 超过了某个阈值,向 CH 发送事件通知。

在实际网络中,更多关注的是被管对象状态的改变量,设单位间隔时间(定义每两个相邻采样点时间间隔为一个单位间隔时间)内被管对象 i 的状态值 $x_{i,t}$ 的改变量为 Δ_i , $\Delta_i = x_{i,t} - x_{i,t-1}$,设 Δ_i 的阈值为 δ_i ,正常情况下 $x_{i,t} - x_{i,t-1} < \delta_i$,不发送事件通知,当 $x_{i,t} - x_{i,t-1} \geq \delta_i$ 时,代理向 CH 发送事件通知。在轮询与事件通知相结合的算法中,当 CH 收到代理发来的事件通知后,立刻触发 CH 开始对全子网的所有监视对象轮询,通过轮询获得 t 时刻的 $f_t = (x_{1,t}, x_{2,t}, \dots, x_{n,t})$ 值,并根据告警函数 f_t 做出是否发出告警消息的判断。

轮询与事件通知结合的算法很多,本文在文献[3]的基础上,提出一种新的网络状态监视轮询策略,算法

如下: 设 t 为获得 $x_{i,t}$ 的上一轮询时间, 令 $t_m = t + \left\lceil \frac{\text{TH}_d - \sum_{i=1}^n x_{i,t}}{\sum_{i=1}^n \delta_i} \right\rceil$, 如果在时间段 $[t, t_m]$ 内, CH 没有收到事件通知, 则等待直到 t_m 时刻开始轮询, 这与文献 [3] 的处理方式相同。如果在时间段 $[t, t_m]$ 内, 例如 $t'(t < t' < t_m)$ 时刻, CH 收到含有某被管对象 j 的状态值 $x_{j,t'}$ 的事件通知, 则计算 $x_{j,t'} + \sum_{i=1, i \neq j}^n [x_{i,t} + (t' - t) \cdot \delta_i]$ 的值, 如果满足 $x_{j,t'} + \sum_{i=1, i \neq j}^n [x_{i,t} + (t' - t) \cdot \delta_i] < \text{TH}_d$, 推迟轮询时刻为 $t' + m'$, 其中 $m' = \frac{1}{\sum_{i=1}^n \delta_i} \left\{ \text{TH}_d - \left[x_{j,t'} + \sum_{i=1, i \neq j}^n [x_{i,t} + (t' - t) \cdot \delta_i] \right] \right\}$; 否则, 立刻轮询, 算法流程见后。本文和文献 [3] 不同的是, 当 CH 收到一个事件通知后, 不是立即轮询, 而是根据判决条件决定是否立即轮询, 文献 [3] 则是不做判断立即轮询。

本文算法的思路是充分利用已知信息来减少未知信息的不确定性, 在 CH 收到事件通知时, 也就收到了事件通知中携带的一个状态最新值 $x_{i,t'}$, 再在最近一次已轮询获得的其它被管对象的状态值的基础上, 按这些状态变化的最坏情况考虑, 增加它们的状态变化量到最大上限值, 判断是否到达预警的可能。只要有预警的可能, 则立即轮询所有状态, 如果不会有预警的可能发生, 则等待一个时间再轮询。以这种方式保证在不漏报的情况下, 减少轮询次数, 并根据告警阈值的上下限的设置, 采用告警滞后机制过滤重复告警。下面证明该算法的正确性。

引理 1 设 CH 在 $t'(t < t' < t_m)$ 时刻收到上一次轮询后的第一个事件通知消息, 消息中含状态值 $x_{j,t'}$, 当 $x_{j,t'} + \sum_{i=1, i \neq j}^n [x_{i,t} + (t' - t) \cdot \delta_i] < \text{TH}_d$ 时, 不轮询。

证明 在 t' 时刻 CH 收到含有 j 的最新状态值 $x_{j,t'}$ 的事件通知后, 此时 CH 无法获得除去 j 以外其余的 $n-1$ 个被管对象的此时的状态信息。但是, 由于只有 j 发送了事件通知, 所以其它被管对象从时刻 t 到时刻 t' 的每个单位间隔时间内状态的变化量 Δ_i 都小于 δ_i , 因此 $\forall i \neq j$, $(x_{i,t'} - x_{i,t}) < (t' - t) \cdot \delta_i$, 对于 t' 时刻的 n 个被管对象有 $\sum_{i=1, i \neq j}^n (x_{i,t'} - x_{i,t}) < \sum_{i=1, i \neq j}^n [(t' - t) \cdot \delta_i]$, 亦即 $\sum_{i=1}^n x_{i,t'} = \sum_{i=1, i \neq j}^n x_{i,t'} + x_{j,t'} < \sum_{i=1, i \neq j}^n [x_{i,t} + (t' - t) \cdot \delta_i] + x_{j,t'}$ 。当 $x_{j,t'} + \sum_{i=1, i \neq j}^n [x_{i,t} + (t' - t) \cdot \delta_i] < \text{TH}_d$ 满足时, 一定有 $\sum_{i=1}^n x_{i,t'} < \text{TH}_d$ 。由预警条件知, 系统不会进入预警状态, 此时不必轮询, 可以等待下一个轮询时间到达。 证毕

在 CH 收到一条事件通知的时候, 我们无法保证分群子网内的被管对象不进入预警状态, 所以考虑接收到事件通知后 CH 可以获得的最新状态值 $x_{j,t'}$, 按最坏情况下计算, 由此得出在 t' 时刻, 当 $x_{j,t'} + \sum_{i=1, i \neq j}^n [x_{i,t} + (t' - t) \cdot \delta_i] < \text{TH}_d$ 时, CH 不必轮询。这意味着在 t' 时刻, 通过计算运算量较

小的计算式 $x_{j,t'} + \sum_{i=1, i \neq j}^n [x_{i,t} + (t' - t) \cdot \delta_i]$, 如果满足此不必轮询的条件, 则不必轮询 t' 时刻分群子网内的全部值, 而将轮询时间延长到某个新的时刻。

定理 1 CH 在 t' 时刻收到上一次轮询时刻 t 后的第一个事件通知消息, 若 $x_{j,t'} + \sum_{i=1, i \neq j}^n [x_{i,t} + (t' - t) \cdot \delta_i] < \text{TH}_d$ 并且 $\forall k, k(t' \leq k \leq t' + m')$ 无事件通知, 则 $\sum_{i=1}^n x_{i,k} \leq \text{TH}_d$ 。这里 $m' = \frac{1}{\sum_{i=1}^n \delta_i} \left\{ \text{TH}_d - \left[x_{j,t'} + \sum_{i=1, i \neq j}^n [x_{i,t} + (t' - t) \cdot \delta_i] \right] \right\}$ 。

证明 根据引理 1, 当 $x_{j,t'} + \sum_{i=1, i \neq j}^n [x_{i,t} + (t' - t) \cdot \delta_i] < \text{TH}_d$ 时, CH 所辖的分群子网内的被管对象未达到预警条件, CH 等待轮询时间的到达。 $\forall k, k(t' \leq k \leq t' + m')$, 由于 k 时刻的 $x_{i,k}$ 满足 $x_{i,k} - x_{i,k-1} < \delta_i$, 可以推出对 n 个被管对象有 $\sum_{i=1}^n x_{i,k} < \sum_{i=1}^n [x_{i,t'} + (k - t') \cdot \delta_i]$,

$$\begin{aligned} & \because \sum_{i=1}^n x_{i,t'} < x_{j,t'} + \sum_{i=1, i \neq j}^n [x_{i,t} + (t' - t) \cdot \delta_i] \\ & \therefore \sum_{i=1}^n x_{i,k} < (k - t')n\delta + \sum_{i=1}^n x_{i,t'} \\ & < \sum_{i=1, i \neq j}^n [x_{i,t} + (t' - t) \cdot \delta_i] + x_{j,t'} + (k - t') \cdot \sum_{i=1}^n \delta_i \end{aligned}$$

$$\begin{aligned} & \text{又} \because (k - t') \cdot \sum_{i=1}^n \delta_i \leq m' \cdot \sum_{i=1}^n \delta_i \\ & \therefore \sum_{i=1}^n x_{i,k} \leq \sum_{i=1, i \neq j}^n [x_{i,t} + (t' - t) \cdot \delta_i] + x_{j,t'} + m' \cdot \sum_{i=1}^n \delta_i \end{aligned}$$

$$\text{当 } m' = \frac{1}{\sum_{i=1}^n \delta_i} \left\{ \text{TH}_d - \left[x_{j,t'} + \sum_{i=1, i \neq j}^n [x_{i,t} + (t' - t) \cdot \delta_i] \right] \right\}$$

时, 有 $\sum_{i=1}^n x_{i,k} \leq \text{TH}_d$ 。 证毕

这表明, 在 t' 时刻, 当 CH 收到事件通知时, 如果满足 $x_{j,t'} + \sum_{i=1, i \neq j}^n [x_{i,t} + (t' - t) \cdot \delta_i] < \text{TH}_d$, 则不必立即轮询, 也不会产生告警漏报的情况, 从而证明了本文算法是正确的。从分析知, 可等待至 $t' + m'$ 时刻开始轮询。如果在 $[t', t' + m']$ 的区间内又收到了新的事件通知, 则可以按 t' 时刻的处理方式一样来判断处理。

当同时收到多个事件通知时, 有如下推论:

推论 1 若在 t' 时刻 CH 收到了对象为 j_1, j_2, \dots, j_l 的 $l(l < l \leq n)$ 个事件通知消息, 若 $\sum_{j_k=j_1, j_2, \dots, j_l} x_{j,t'} + \sum_{i=1, i \neq j_k}^n [x_{i,t} + (t' - t) \cdot \delta_i] < \text{TH}_d$, 并且 $\forall k, k(t' \leq k \leq t' + m')$ 无事件通知, 则有 $\sum_{i=1}^n x_{i,k} \leq \text{TH}_d$, 这里 $m' = \frac{1}{\sum_{i=1}^n \delta_i} \cdot \left\{ \text{TH}_d - \left[\sum_{j_k=j_1, j_2, \dots, j_l} x_{j,t'} + \sum_{i=1, i \neq j_k}^n [x_{i,t} + (t' - t) \cdot \delta_i] \right] \right\}$ 。

证明与定理 1 的证明类似。

//本文的轮询告警算法

```

1: function Alarm()
2: {
3:    $f \leftarrow \text{polling all } (x_i)$ ;
4:    $t_m = t + \left\lceil \frac{(\text{TH}_d - \sum_{i=1}^n x_{i,t})}{\sum_{i=1}^n \delta_i} \right\rceil$ ;
5:   If ( $f > \text{TH}_d$ )
6:     A=alarmStratupAlarm;
7:   while( $f_{t-1} < \text{TH}_u$  &  $f_t \geq \text{TH}_u$ )
8:     report ALARM;
9:   while( $(A=1|A=3)$  &  $f_t > \text{TH}_u$ )
10:    report ALARM;
11: While  $A = 2$  &  $f_{t-2} > \text{TH}_u$  &  $f_{t-1} < \text{TH}_u$  &  $f_t > \text{TH}_u$ )
12:   report ALARM;
13: }
14: int  $t_m = 0$ ;  $f_t = 0$ ;  $t' = 0$ ;
15: while(TRUE)
16:   if ( $t < t_m$  & receive trap at  $t'$ )
17:     if  $\left( x_{j,t'} + \sum_{i=1, i \neq j}^n [x_{i,t} + (t' - t) \cdot \delta_i] < \text{TH}_d \right)$ 
18:       function Alarm();
19:     else
20:        $t_m = t' + \frac{1}{\sum_{i=1}^n \delta_i} \left\lceil \text{TH}_d - \left[ x_{j,t'} + \sum_{i=1, i \neq j}^n [x_{i,t} + (t' - t) \cdot \delta_i] \right] \right\rceil$ 
21:       elseif ( $t \geq t_m$ )
22:         function Alarm();

```

本文同时考虑了算法的复杂性, 由于CH处驻留的只是一个远程RMON探测器, 运算能力有限, 本算法在条件判断时, 利用前面的运算结果, 降低了对系统资源的占用。

现分析本文算法的通信开销。设 $P_r(i)$ 为被管对象的变化量 $x_{i,t} - x_{i,t-1} > \delta_i$ 的概率, 则单位时间内不发事件通知的概率为 $\prod_{i=1}^n (1 - P_r(i))$, 至少收到一个事件通知的概率为 $1 - \prod_{i=1}^n (1 - P_r(i))$ 。而在 $[t, t_m]$ 时间内没有收到事件通知时的平均等待时间为 $\text{EXP} \left[\frac{1}{\sum_{i=1}^n \delta_i} (\text{TH}_d - \sum_{i=1}^n x_{i,t}) \right]$ 。设 $P_t(j)$ 为被管对象状态满足 $x_{j,t'} + \sum_{i=1, i \neq j}^n [x_{i,t} + (t' - t) \cdot \delta_i] < \text{TH}_d$ 式的概率, 则收到一个事件通知并开始轮询的平均等待时间为

$$\text{EXP} \left\{ \frac{1}{\sum_{i=1}^n \delta_i} \left\lceil \text{TH}_d - \left[x_{j,t'} + \sum_{i=1, i \neq j}^n [x_{i,t} + (t' - t) \cdot \delta_i] \right] \right\rceil \right\}$$

所以, 本文算法下, 单位时间内每次轮询 n 个被管对象的期望代价 $g(\delta_i)$ 为

$$g(\delta_i) = \left[\frac{n \sum_{i=1}^n \delta_i}{\text{TH}_d - \text{EXP}(\sum_{i=1}^n x_{i,t})} \right] + n \cdot \left\{ 1 - \prod_{i=1}^n [1 - P_r(i)] \right\} \cdot (1 - P_t(j))$$

$$+ \left\{ \frac{n \sum_{i=1}^n \delta_i}{\text{TH}_d - \text{EXP} \left\{ x_{j,t'} + \sum_{i=1, i \neq j}^n [x_{i,t} + (t' - t) \cdot \delta_i] \right\}} \right\} \cdot P_t(j)$$

这是一个与 δ_i 有关的函数, 在本文中, 无论是 $P_r(i)$ 还是 $P_t(j)$ 都与 δ_i 有关, 而 δ_i 与被管对象在不同时刻的不同状态有关, 所以, 本文的算法在排除了不必要的轮询以减低开销的同时, 充分地利用了网络动态信息。

我们称在每个轮询时刻下对分群子网内所有被管对象均轮询的算法为 O_b 。 O_b 算法的通信代价为 n , 则本文算法与 O_b 算法的比值为: $g(\delta_i)/n$, 通过 $dg(\delta_i)/d\delta_i = 0$, 得到 $\delta_i = \delta_{\text{optimal}}$, 即选取合适的 δ_{optimal} , 可以得到了最优 $g_{\text{min}}(\delta_i) = g(\delta_{\text{optimal}})$ 。

已有的算法是在收到事件通知的 t' 时刻, 立刻开始轮询, 轮询的结果固然有 $\sum_{i=1}^n x_{i,t'} \geq \text{TH}_d$ 的情况, 但也有 $\sum_{i=1}^n x_{i,t'} < \text{TH}_d$ 的情况, 而当 $\sum_{i=1}^n x_{i,t'} < \text{TH}_d$ 时, 不必立即轮询。本文是在利用了事件通知的最新信息 $x_{j,t}$ 对告警条件的可能性做出预测后, 进行判断, 如果 $x_{j,t'} + \sum_{i=1, i \neq j}^n [x_{i,t} + (t' - t) \cdot \delta_i] < \text{TH}_d$ 式成立, 则排除了 $\sum_{i=1}^n x_{i,t'} \geq \text{TH}_d$ 成立的情况下产生的轮询, 同时, 本文算法采用了告警滞后机制, 过滤了由于告警的波动而产生的大量冗余警告, 进一步降低了监视信息占用的带宽, 所以本文的算法比已有的算法代价低, 是优化的算法。

4 仿真实验及分析

本文采集实际Internet网络上的数据, 采集包长在64byte到1518byte之间分布的吞吐量, 每10s采集一次, 持续24个小时的流量作为仿真数据源, 采集10个分布点的数据, 仿真结果如下:

图2所示为本文算法与文献[3]算法的比较, 可以看出在相同的数据源下, 采用本文的算法对网络轮询监视, 与文献[3]相比, 对网络带宽的占用量节省了将近1/2。通过采用合适的策略, 降低轮询的开销, 也就降低了管理信息对网络带宽的占用, 从而有效地降低了管理开销。

图3所示为分群子网内的节点数目变化时, 被管对象状态值的变化量与节省的监视流量的关系, 可以看出, 随着 δ_i 的增加, 节省的监视流量也在增加, 当 δ_i 的值增加到一定值后, 监视流量的增加变缓, 其原因是当被管对象状态量的变化量 δ_i 增加时, 代理向CH发送事件通知的概率减小, 但CH处轮询时间间隔缩短, 增加了管理开销。同时可以看出, 随着节点数目的增加, 节省监视流量的比值也在减少, 表明此监视算法与监视对象的数目有关。

图4为本文采用滞后告警机制下的算法性能比较, 文献[3]中当告警条件 $f_t > \text{TH}_d$ 时就告警, 这样的告警机制受小信号波动的影响较大, 降低了带宽的利用率, 而本文根据告警

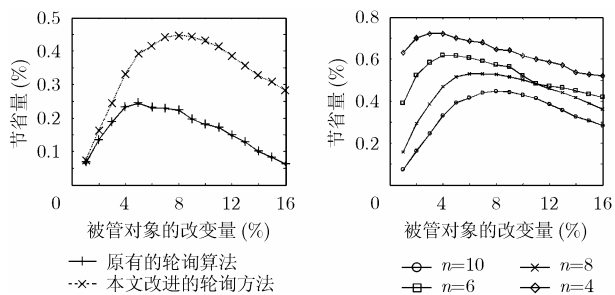


图 2 本文算法对轮询的改进

图 3 不同被管对象数目下的节省量

滞后机制设置了告警阈值和预警阈值, 有效地过滤重复告警信息, 可以看出告警次数明显降低。从而减少了管理开销。

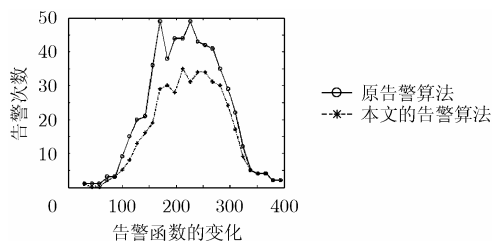


图 4 滞后告警机制下的性能

5 结束语

如何降低网络监视的通信开销是网络管理的一个重要研究方向。轮询和事件通知的方法都要占用一定的网络带宽, 因此在不漏报网络故障的情况下, 尽量减少轮询和事件通知的次数能够降低网络管理的开销。本文在分层分布式网络管理构架下, 提出了一种降低网络轮询流量的监视算法。算法根据轮询和事件通知的约束条件, 用获得的网络状态数据, 预测发生告警的可能性。对于不发生告警的情况, 则延时轮询, 这样做既不产生告警的漏报, 又有效地降低了网络

管理的通信负担。仿真结果验证了本文算法的有效性。进一步分析看出, 网络发生异常时, 往往会有几个被管对象出现异常, 而由一个事件通知触发的轮询代价要高于多个事件通知共同作用的结果, 所以, 下一步工作将研究多个事件通知共同触发下的通信开销, 以期有效降低网络管理通信占用的带宽。

参考文献

- [1] Shen C C, Srisathapornphat C, and Jaikaeo C. An adaptive management architecture for ad hoc networks. *IEEE Communications Magazine*, 2003, 41(2): 108-115.
- [2] Duarte J and Santos A D. Network fault management based on SNMP agent groups. *Proceedings of the IEEE, 21st International Conference on Distributed Computing Systems Workshops, Phoenix, Arizona, ICDCS, 2001: 51-56.*
- [3] Dilman M and Raz D. Efficient reactive monitoring. *IEEE Journal on Selected Areas in Communications*, 2002, 20(4): 668-676.
- [4] Jia J, Naqvi Sh and Raz D, et al. Toward efficient monitoring. *IEEE Journal on Selected Areas in Communications*, 2000, 18(5): 723-732.
- [5] Stallings W. *SNMP, SNMPV2, SNMPV3, RMON1 and 2: Practical Network Management*, Addison Wesley, 1998.

张 新: 女, 1968 年生, 副教授, 研究方向为网络管理。
 常义林: 男, 1944 年生, 教授, 博士生导师, 研究方向为网络管理。
 孙方涛: 男, 1980 年生, 讲师, 研究方向为网络管理。
 沈 中: 男, 1969 年生, 博士, 讲师, 研究方向为网络管理。