

# 선로 이용률 예측 기반의 동적 폴링 기법

조 강 홍<sup>†</sup> · 안 성 진<sup>††</sup> · 정 진 욱<sup>†††</sup>

## 요 약

본 논문에서는 선로의 이용률 예측을 기반으로 하여 폴링 주기를 동적으로 변경시킬 수 있는 새로운 폴링 기법을 제시한다. 폴링은 네트워크 모니터링의 가장 중요한 기능이지만, 네트워크가 congestion 상태일 때 과도한 폴링 데이터는 네트워크를 심각한 congestion 상태로 만든다. 따라서 기존의 여러 폴링 기법들이 이전 시점에 폴링 요청했을 때 수신한 응답 메시지의 Round Trip Time 또는 폴링한 값의 선로 이용률을 통해 네트워크 congestion 또는 에이전트의 로드 여부를 판단하여 폴링 주기를 변경하여 폴링 트래픽을 조절하였으나 이는 이전 시점의 폴링을 근거로 폴링 주기를 변경하는 기법이기에 때문에 폴링하고자 하는 당 시점의 네트워크 상태를 반영하지 못한다. 본 논문에서 제시하는 기법은 과거의 데이터를 근거로 폴링 시점에 폴링 데이터가 폴링 경로 상의 선로 이용률 임계값을 위반하는지를 예측하여 이를 통해 폴링 주기를 변경시킨다. 본 논문에서는 Box-Jenkins의 AR (Autoregressive) 모델을 사용하여 네트워크를 구성하는 각 선로의 이용률을 예측하였고 임계값 위반 여부를 확률로 제시하였다. 또한, 제시한 선로 이용률 예측 기반의 동적 폴링 기법을 실제 네트워크에 적용하여 적합성 여부를 평가하였고, 실험을 통하여 적절한 수준의 선로 이용률 임계값과 임계값 위반 확률을 판단함으로써 본 기법의 성능을 최대화하였다.

## Dynamic Polling Algorithm Based on Line Utilization Prediction

Kang Hong Cho<sup>†</sup> · Seong Jin Ahn<sup>††</sup> · Jin Wook Chung<sup>†††</sup>

## ABSTRACT

This study proposes a new polling algorithm allowing dynamic change in polling period based on line utilization prediction. Polling is the most important function in network monitoring, but excessive polling data causes rather serious congestion conditions of network when network is in congestion. Therefore, existing multiple polling algorithms decided network congestion or load of agent with previously performed polling Round Trip Time or line utilization, changed polling period, and controlled polling traffic. But, this algorithm is to change the polling period based on the previous polling and does not reflect network conditions in the current time to be polled. A algorithm proposed in this study is to predict whether polling traffic exceeds threshold of line utilization on polling path based on the past data and to change the polling period with the prediction. In this study, utilization of each line configuring network was predicted with AR model and violation of threshold was presented in probability. In addition, suitability was evaluated by applying the proposed dynamic polling algorithm based on line utilization prediction to the actual network, reasonable level of threshold for line utilization and the violation probability of threshold were decided by experiment. Performance of this algorithm was maximized with these processes.

키워드: 예측(Prediction), AR, congestion, 선로 이용률(Line utilization), SNMP, MIB-II

### 1. 서 론

네트워크의 상태를 분석하기 위한 네트워크 모니터링의 가장 중요한 기능은 폴링 기법이다. TCP/IP 네트워크 상에서 SNMP(Simple Network Management Protocol)는 폴링을 이용하는 대표적인 프로토콜으로써 MIB에 정의된 관리 오브젝트들을 라우터, 호스트, 스위치 등의 SNMP 에이전트에게 요청함으로써 가져온다. MIB-II에 근거한 대부분의 오브젝트들은 Count 타입의 값으로 t 시점에서의 폴링한 데이터 집합을  $S_i(t)$  ( $1 \leq i \leq n$ ) 라고 할 때, 폴링 주기인  $P(t)$  이후의

t+1 시점에서의 폴링한 데이터 집합  $S_i(t+1)$  ( $1 \leq i \leq n$ )과 의 차이를 통하여 에이전트의 관리 정보들을 분석한다. 반면에 CPU 이용률과 같은 Gauge 타입의 오브젝트들은 단지 그 시점의 값을 폴링함으로써 가져올 수 있다.

그러나, SNMP 폴링의 구조적인 문제점은 관리하고자 하는 관리 대상의 수인  $N$  과 관리 오브젝트의 수인  $n$ , 그리고 폴링 주기  $P(t)$ 에 따라 발생하는 트래픽의 양은 크게 증가하게 된다. 간단하게 표현하면, 하나의 관리 오브젝트가 k 바이트라고 할 때, 한 번 폴링의 경우에 발생하는 총 관리 트래픽의 양은  $2 \times k \times n \times N$  이고, 총 관리 시간  $S$  을

기준으로 할 경우에 전체 트래픽  $\frac{2 \times k \times n \times N \times P(t)}{S}$  이

다. 이와 같은 관리 트래픽의 증가는 네트워크가 congest-

<sup>†</sup> 준 회원: 성균관대학교 대학원 전기전자 및 컴퓨터 공학부

<sup>††</sup> 종신회원: 성균관대학교 컴퓨터교육과 교수

<sup>†††</sup> 종신회원: 성균관대학교 전기전자 및 컴퓨터공학부 교수

논문접수: 2002년 3월 21일, 심사완료: 2002년 6월 25일

tion일 경우에는 네트워크를 좀 더 심각한 congestion 상태로 만들게 되며, 네트워크 관리라는 개념과 상충되는 기능을 수행하게 된다.

따라서, 이와 관련된 다양한 폴링 기법들이 계속해서 연구되어지고 있다. [1-4]는 이전 시점 폴링 메시지의 Round Trip Time을 근거로 네트워크 congestion을 판단한 후에 다음 시점의 네트워크 폴링 주기를 결정한다. 또한 [5, 6]은 에이전트의 CPU 또는 메모리 이용률 등의 프로세싱 로드를 몇 개의 레벨로 구분하여 폴링 주기를 결정한다. 또한, [7]은 폴링하는 네트워크 관리 데이터 값의 변화량과 네트워크 관리 bandwidth를 고려하여 폴링 주기를 결정하는 기법을 제시하였다.

그러나, 현재까지 제시된 모든 폴링 기법들은 이전 폴링 시점의 네트워크 congestion과 데이터 값의 변화량을 기준으로 다음 시점의 폴링 주기를 결정하였다. 예를 들어, 현재 시점을  $t$ 라고 하면 현재의 폴링 주기는 바로 이전의 폴링 시점인  $t-1$ 에서 폴링했을 때 수신한 응답 메시지의 Round Trip Time 또는 그 메시지를 통해 계산할 수 있는 선로 이용률을 통해 현재  $t$ 시점의 폴링 주기를 결정하였다. 결과적으로 가장 최근에 폴링했던  $t-1$  시점의 네트워크 상태를 현재  $t$ 시점에 적용한 것이다. 이와 같은 방식으로는 다음 시점에 변화된 네트워크 상태를 반영하는 것이 불가능하다. 즉, 이전 폴링 시점인  $t-1$ 에는 네트워크 congestion 상태였기 때문에 폴링 주기를 증가시켰지만, 현재 폴링 시점인  $t$ 에서는 네트워크가 congestion 상태가 해소되었을 수 있다. 또한, 이전 폴링 시점에는 네트워크가 안정된 상태였지만 현재 폴링 시점에서 congestion이 발생되고 있는 상태라면, 폴링 기법의 제 기능을 수행하지 못하게 된다. 이는 폴링 주기가 길면 길수록 이전 네트워크의 상태를 현재 네트워크의 상태로 반영하는 것이 부정확하게 된다.

본 논문에서는 선로 이용률 예측을 기반으로 하여 폴링 주기를 동적으로 변경시킬 수 있는 새로운 폴링 기법을 제시하였다. 본 기법에서는 과거의 데이터를 근거로 폴링 시점에 폴링 데이터가 폴링 경로 상의 선로 이용률 임계값을 위반하는지를 예측하여 이를 통해 폴링 주기를 변경시킨다. 이를 AR 모델을 사용하여 각 선로의 이용률을 예측하였고 그 신뢰 구간을 추정함으로써 확률에 근거하여 선로 이용률 임계값 위반 예측을 수행하였다. 또한, 제시한 모델의 적합성 여부를 평가하였고, 실험을 통하여 적절한 수준의 임계값과 탐지 확률을 수행함으로써 본 기법의 성능을 최대한화하였다.

## 2. 선로 이용률의 예측 모델

선로 이용률은 선로의 bandwidth를 기준으로 하여 현재 사용되고 있는 선로의 이용률을 퍼센트로 나타내는 값으로

라우터, 스위치, 호스트 등과 같은 SNMP 에이전트의 MIB-II 인터페이스 그룹에 포함된  $sysUpTime, IfInOctet, IfOutOctet, IfSpeed$  오브젝트를 폴링함으로써 계산할 수 있다. 이용률과 관련된 여러 가지 방법들이 존재하나, 본 논문에서는 기본적인 MIB 값으로 계산 가능한 다음의 선로 이용률 계산 방법을 사용하였다[8].

$$Z_t = \frac{\text{MAX}\{(IfInOctet_t - IfInOctet_{t-1}), (IfOutOctet_t - IfOutOctet_{t-1})\}}{IfSpeed \times (sysUpTime_t - sysUpTime_{t-1})/100} \quad (1)$$

사용된 오브젝트가 모두 Count 타입의 오브젝트이기 때문에  $t$ 시점에서 폴링한 오브젝트 값과  $t-1$ 시점에서의 오브젝트 값의 차이를 계산함으로써  $[t-1, t]$ 시점에서의 선로 이용률을 계산할 수 있다. 위에서 사용된 오브젝트들은 네트워크 모니터링을 위해 항상 기본적으로 폴링되는 오브젝트들로 추가적인 데이터 폴링없이 사용할 수 있다.

이와 같이 계산된 선로 이용률의 과거 데이터를 근거로 선시점  $t+k$ 의 선로 이용률을 예측할 수 있다. 이를 위해 다양한 예측 모델이 사용될 수 있으나 본 논문에서는 단기 간의 네트워크 트래픽 예측에 적합하고 계산의 복잡성도 높지 않기 때문에 많이 사용되고 있는 AR 모형(Autoregressive model)을 사용한다[9-11].

일반  $p$ 차 AR 모형은 시계열 자체에 대한 회귀 형태를 취하는 모형으로 다음과 같이 표현된다.

$$Z_t = \sum_{j=1}^p \phi_j Z_{t-j} + a_t \quad (2)$$

여기서  $a_t$  백색 잡음으로서 평균 0과 분산  $\sigma_a^2$ 을 가지는 독립이고 동일한 분포(iid : independent, identically distributed)의 확률 변수이고,  $\phi_j$ 는 자기회기 모형(autoregressive)의 모수이다. 즉,  $t$ 시점에서의 선로 이용률  $Z_t$ 는  $p$ 개의 가장 가까운 과거 선로 이용률 데이터들과 이 데이터들로 설명되지 않는 오차항인 백색 잡음  $a_t$ 와의 선형결합이다. 이 모형을 통하여 선시점  $k$ 의 선로 이용률 예측값  $\hat{Z}_t(k)$ 을 계산할 수 있다. 본 논문에서는 AR 모형  $Z_t$ 는 정상적이라고 가정한다.

## 3. 선로 이용률 예측 기반의 동적 폴링 기법

폴링을 수행하는 관리 시스템 a 와 피관리 에이전트 시스템 b 사이의 경로에 있는 선로를  $x, y, \dots, z$  라고 할 때,  $P_{a,b} = L_x, L_y, \dots, L_z$ 과 같이 표기하고,  $t$ 시점에서 선로의 이용률 집합을  $T_{a,b}(t) = Z_x(t), Z_y(t), \dots, Z_z(t)$ 와 같이 나타낸다. 이 때 선로  $x$ 의 이용률  $Z_x(t)$ 는 식 (1)을 이용하여 식 (3)과 같이 나타낸다.

$$Z_x(t) = \frac{\text{MAX}\{(IfInOctet_x(t) - IfInOctet_{x-1}(t)), (IfOutOctet_x(t) - IfOutOctet_{x-1}(t))\}}{IfSpeed \times (sysUpTime_x(t) - sysUpTime_{x-1}(t))/100} \quad (3)$$

식 (3)를 이용하여 선로 이용률을 계산할 수 있으며,  $t$  시점까지의 과거 데이터를 이용하여  $t+1, t+2, \dots, t+l$  시간의 선로 이용률을 예측한다. 시점이  $t$ 이고 선시차가  $l$ 인 시계열  $Z_1, Z_2, \dots, Z_n$ 의 최소 평균제곱오차 예측(minimum mean square error forecast)의 일반식은

$$\hat{Z}_x(l) = E(Z_{x+l} | Z_x, \dots, Z_{x-1}, \dots) \quad (4)$$

이며, 이 때 AR(p) 모형을 통한 선시차  $l$ 의 예측식은

$$\hat{Z}_x(t+l) = \sum_{j=0}^{p-1} \phi_j Z_x(t-j) \quad (5)$$

이다.

$t$  시점에서 산출된 선시차  $l$ 의 예측값  $\hat{Z}_x(t+l)$ 이 계산이 되면  $\hat{Z}_n(t+l)$ 이 그 시점의 선로 이용률의 포화 상태 여부를 판별하는 기준이 되는 임계값인  $\lambda_x(t+l)$ 을 넘어설 확률을 정규분포를 통해 계산할 수 있다. 이를  $\hat{P}_n(t+l)$ 라고 하면

$$\begin{aligned} \hat{P}_x(t+l) &= P(\hat{Z}_x(t+l) \geq \lambda_x(t+l)) \\ &= P(S_x(t+l) \geq \frac{\lambda_x(t+l) - \hat{Z}_x(t+l)}{\sigma_x(t+l)}) \end{aligned} \quad (6)$$

이다. 여기서  $S_n(k)$ 은 정규 분포로 표준화 된 확률 변수이다.

즉,  $t$  시점에서  $a$ 와  $b$  구간 사이의 선시차가  $l$ 인 선로 이용률 예측값의 집합은

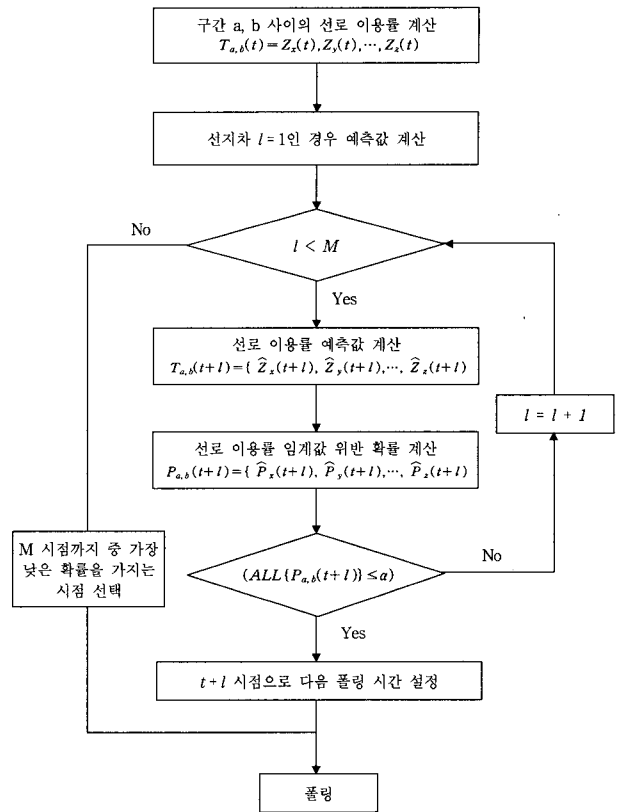
$$\hat{Z}_{a,b}(t+l) = \{ \hat{Z}_x(t+l), \hat{Z}_y(t+l), \dots, \hat{Z}_z(t+l) \} \quad (7)$$

라고 나타낼 수 있으며, 이 예측값이 선로 이용률 임계값을 넘을 확률의 집합은

$$\hat{P}_{a,b}(t+l) = \{ \hat{P}_x(t+l), \hat{P}_y(t+l), \dots, \hat{P}_z(t+l) \} \quad (8)$$

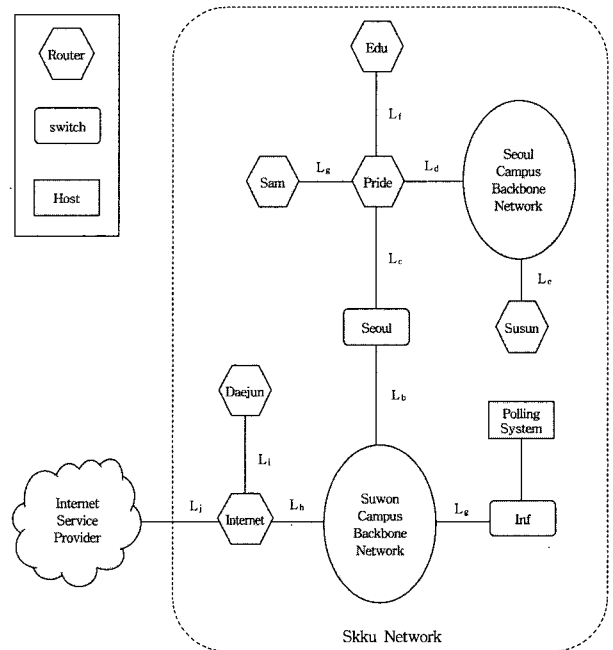
로 나타낼 수 있다.

이와 같이 계산된 확률값을 기준 확률값  $\alpha$ 과 각각 비교하여  $\alpha$  값을 넘었을 경우에는 다음 시점으로 폴링 시점을 변경하고, 이를 반복하여 폴링 시점을 찾아낸다. 즉,  $t$  시점에서 계산된  $P_{a,b}(t+l)$  ( $l=1, 2, \dots, M$ )의 확률 중에서 모든 선로의 확률값이  $\alpha$  값을 넘어서지 않는 가장 가까운 시점을 폴링 시점으로 선택한다.



(그림 1) 선로 이용률 예측 기반 동적 폴링 기법의 flow chart

#### 4. 실험 및 성능 평가



(그림 2) 성균관대학교 네트워크 구성도

본 논문에서 제시한 알고리즘의 적합성과 그 타당성을 평가하기 위하여 실제 네트워크 상에 적용하였다. (그림 2)

는 성균관대학교의 실제 네트워크 구성도이며 이에 대한 실험 환경은 다음과 같다.

[실험 환경]

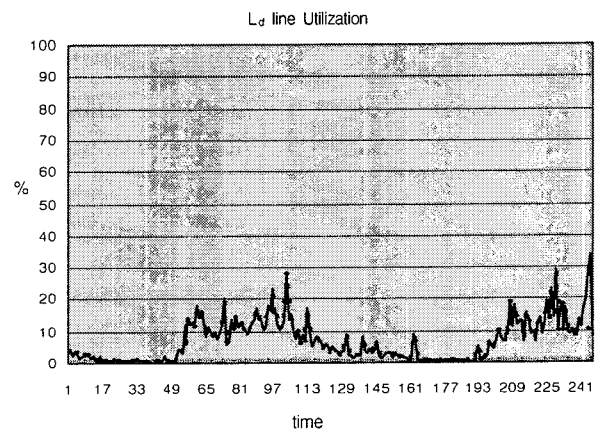
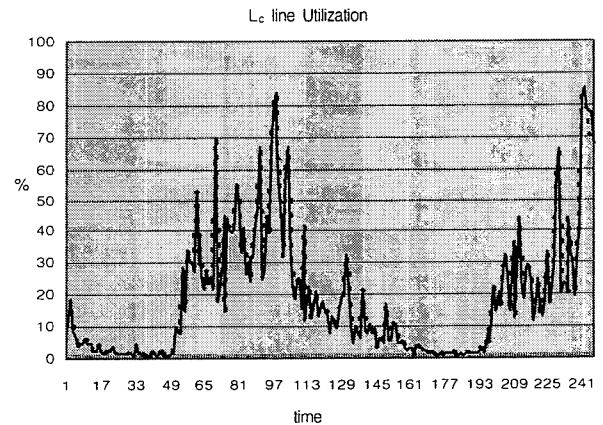
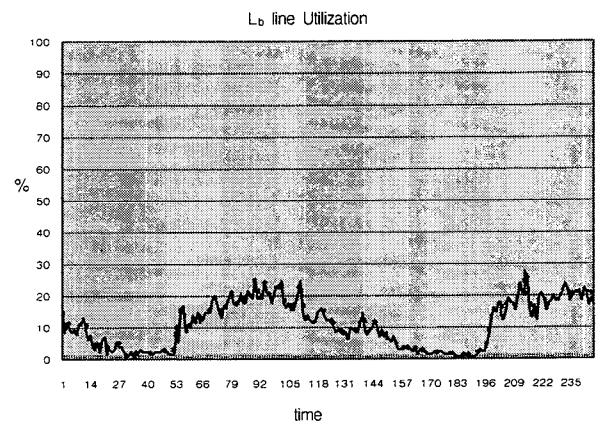
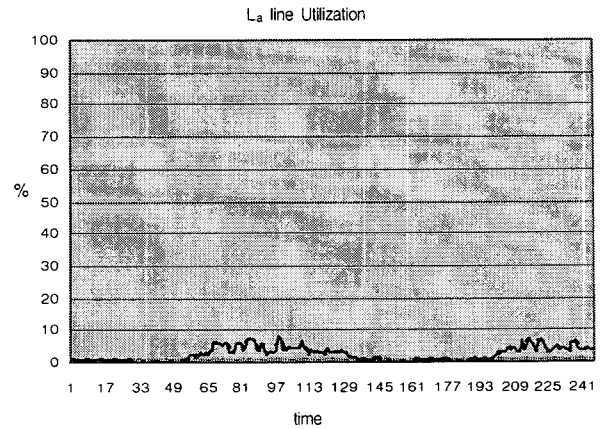
- (1) 성균관대학교의 실제 네트워크(서울 캠퍼스와 수원 캠퍼스)를 구성하는 7개의 라우터, 스위치, 호스트를 폴링 대상으로 하고 이를 연결하는 10개의 선로에 대해 식 (3) 통해 선로 이용률을 계산하였다.
- (2) 실험을 위한 폴링 시스템은 UNIX 서버(Solaris 7)에 설치하였다.
- (3) 폴링 시스템에서는 각 관리 대상 장비들을 10분 주기로 폴링하여 모니터링 하였고, 그 기간은 2001년 12월 10일부터 24일까지 약 2주간이다.
- (4) 폴링 시스템은 자바로 구현하였고, 폴링 데이터를 위해 데이터베이스는 Mysql을 사용하였다.

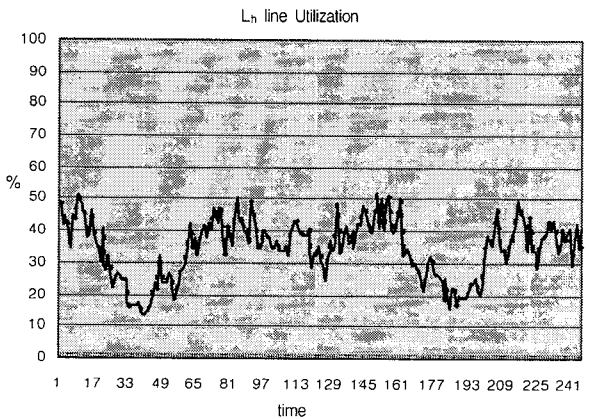
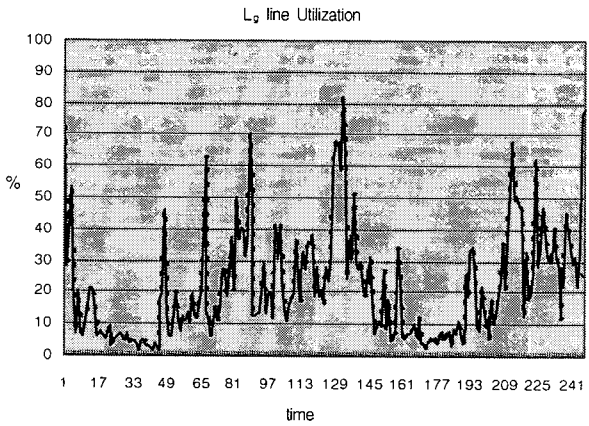
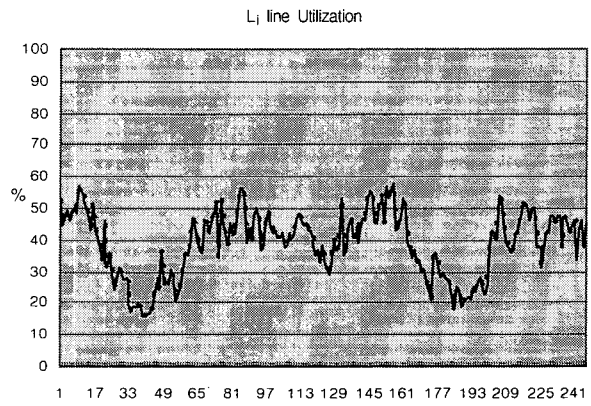
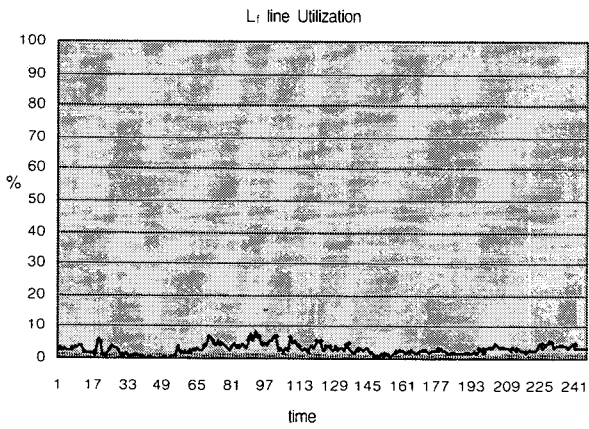
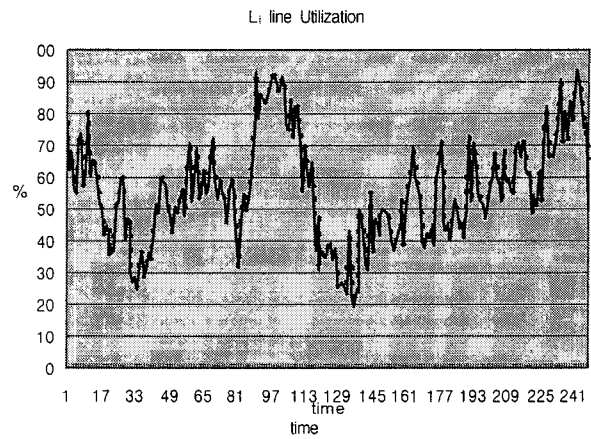
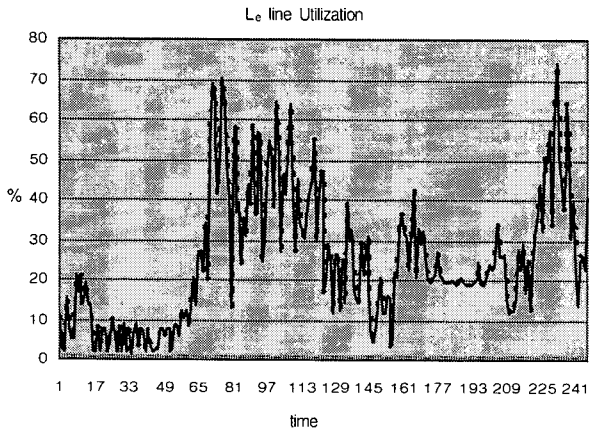
4.1 실험

(그림 1)과 같이 폴링 시스템에서 관리 대상이 되는 각 장비의 집합은 {Seoul, Pride, Sam, Edu, Susun, Internet, Daejun}이며, 이를 연결하는 각 선로의 집합은  $\{L_a, L_b, L_c, L_d, L_e, L_f, L_g, L_h, L_i, L_j\}$ 이다. (그림 3)은 시간에 따른 각 선로의 이용률의 추이를 보여주고 있다. (그림 2)에서 보는 바와 같이  $L_c$ 는 서울 캠퍼스와 수원 캠퍼스를 연결하는 serial 라인으로 선로 이용률이 높은 것을 볼 수 있고,  $L_i$ 는 대전으로 나가는 serial 라인으로 시간에 따라 높은 트래픽을 보인다. <표 1>은 제시한 알고리즘을 사용하기 위한 각 선로의 타입과 속도를 나타내고 있으며, 예측을 위해 사용된 AR order와 AR parameter들을 나타낸다.

<표 1> 선로 정보와 AR parameters

선로 이름	선로 타입	선로 속도	예 측	
			D (order)	AR parameter
$L_a$	ethernet	10Mbps	3	$\phi_1=0.892, \phi_2=-0.189, \phi_3=0.232$
$L_b$	ethernet	100Mbps	2	$\phi_1=0.775, \phi_2=0.177$
$L_c$	serial	2.048Mbps	4	$\phi_1=0.421, \phi_2=0.136, \phi_3=0.164, \phi_4=0.171$
$L_d$	ethernet	10Mbps	3	$\phi_1=0.337, \phi_2=0.419, \phi_3=0.150$
$L_e$	ethernet	10Mbps	4	$\phi_1=0.586, \phi_2=0.025, \phi_3=0.084, \phi_4=0.239$
$L_f$	serial	155.52Mbps	2	$\phi_1=0.656, \phi_2=0.162$
$L_g$	serial	1.544Mbps	2	$\phi_1=0.636, \phi_2=0.139$
$L_h$	ethernet	100Mbps	2	$\phi_1=0.665, \phi_2=0.227$
$L_i$	serial	44.21Mbps	2	$\phi_1=0.695, \phi_2=0.203$
$L_j$	serial	44.21Mbps	2	$\phi_1=0.741, \phi_2=0.155$





(그림 3) 선로 이용률

폴링 시스템에서 각 장비의 모니터링을 위해 폴링을 수행할 때 폴링 데이터가 거쳐야 하는 경로는

$$Path_{PollingSystem, Seoul} = \{L_a, L_b\}$$

$$Path_{PollingSystem, Pride} = \{L_a, L_b, L_c\}$$

$$Path_{PollingSystem, Susun} = \{L_a, L_b, L_c, L_d, L_e\}$$

$$Path_{PollingSystem, Edu} = \{L_a, L_b, L_c, L_f\}$$

$$Path_{PollingSystem, Sam} = \{L_a, L_b, L_c, L_g\}$$

$$Path_{PollingSystem, internet} = \{L_a, L_h\}$$

$$Path_{PollingSystem, Daejun} = \{L_a, L_h, L_i\}$$

이다. 따라서, 각 경로에 대한 선로 이용률을 계산해야 하고 이를 근거로 다음 시점의 선로 이용률을 예측한다. <표 2>는 최대 폴링 간격  $M$ 을 한 시간인 6이라고 했을 때 (식 5)를 이용하여 시점  $t$ 에서  $t+6$ 까지의 각 선로의 이용률을 예측한 결과와 이 예측값이 식 (6)에 의해 정규화 되었을 때 임계값  $\lambda_x$ 를 넘을 확률을 나타낸다. 선로 이용률의 임계값은 선로 타입이 ethernet일 경우는 30%일 경우와 40%일 경우로 구분하여 계산하였고, 선로 타입이 serial일 경우에는 80%와 90%의 경우를 계산하였다.

이와 같이 예측된 선로 이용률을 근거로 폴링을 수행할 때의 모든 경로가 congestion이 아닌 상태를 만족시키는 폴

<표 2> 선로 이용률 예측값과 임계값 위반 확률

선로 이름	l (시점)	실제값	예측값			$\hat{P}_n(t+l)$		선로 이름	l (시점)	실제값	예측값			$\hat{P}_n(t+l)$	
			하한값	$\hat{Z}_x(t+l)$	상한값	$\lambda_x = \{30,80\}$	$\lambda_x = \{40,90\}$				하한값	$\hat{Z}_x(t+l)$	상한값	$\lambda_x = \{30,80\}$	$\lambda_x = \{40,90\}$
L <sub>a</sub>	0	3.41						L <sub>f</sub>	0	4.58					
	1	3.76	2.60	3.86	5.12	0.00	0.00		1	2.81	2.67	4.16	5.64	0.00	0.00
	2	3.75	2.22	3.90	5.58	0.00	0.00		2	2.87	2.19	3.97	5.75	0.00	0.00
	3	4.34	1.84	3.69	5.53	0.00	0.00		3	2.91	1.80	3.79	5.77	0.00	0.00
	4	3.66	1.60	3.54	5.59	0.00	0.00		4	2.93	1.51	3.63	5.75	0.00	0.00
	5	3.38	1.42	3.56	5.71	0.00	0.00		5	3.52	1.29	3.60	5.71	0.00	0.00
	6	3.33	1.23	3.50	5.78	0.00	0.00		6	2.99	1.12	3.39	5.66	0.00	0.00
L <sub>b</sub>	0	21.73						L <sub>g</sub>	0	34.1					
	1	18.12	17.43	21.14	24.84	0.11	0.00		1	29.10	14.88	31.76	48.65	0.00	0.00
	2	22.57	16.14	20.71	25.29	0.15	0.02		2	30.97	9.50	29.51	49.52	0.00	0.00
	3	20.48	15.57	20.55	25.54	0.16	0.02		3	22.01	5.47	27.75	49.77	0.00	0.00
	4	16.72	15.00	20.41	25.82	0.17	0.03		4	32.90	3.11	26.32	49.53	0.00	0.00
	5	19.47	14.35	20.19	26.04	0.20	0.04		5	76.44	1.21	25.16	49.12	0.00	0.00
	6	16.66	13.76	19.98	26.21	0.20	0.50		6	78.26	1.00	24.23	48.66	0.00	0.00
L <sub>c</sub>	0	41.11						L <sub>h</sub>	0	39.58					
	1	81.66	68.84	83.11	97.38	0.60	0.30		1	41.19	28.46	35.32	42.19	0.80	0.23
	2	85.11	66.30	81.78	97.26	0.56	0.27		2	29.33	27.26	35.50	43.75	0.64	0.17
	3	78.83	56.32	72.43	88.55	0.33	0.15		3	37.28	25.85	35.29	44.73	0.66	0.23
	4	77.79	53.22	70.11	87.00	0.26	0.10		4	42.78	24.90	35.19	45.48	0.74	0.29
	5	77.46	53.25	71.22	89.19	0.30	0.15		5	34.75	24.12	35.08	46.02	0.71	0.33
	6	68.31	47.16	65.82	84.48	0.20	0.09		6	40.77	23.50	34.97	46.45	0.67	0.33
L <sub>d</sub>	0	13.73						L <sub>i</sub>	0	86.31					
	1	11.39	4.91	9.8	14.69	0.00	0.00		1	93.58	78.94	91.21	103.48	0.83	0.56
	2	17.28	5.69	10.84	16.00	0.00	0.00		2	90.05	75.29	90.24	105.19	0.77	0.52
	3	20.00	4.64	10.41	16.19	0.00	0.00		3	82.75	67.86	85.02	67.87	0.64	0.36
	4	28.80	3.9	10.11	16.34	0.01	0.00		4	75.30	54.11	72.87	91.63	0.13	0.17
	5	33.72	3.43	10.00	16.56	0.01	0.00		5	77.29	53.12	73.14	93.16	0.35	0.18
	6	8.31	2.88	8.77	16.65	0.01	0.00		6	49.58	44.40	65.42	86.44	0.23	0.10
L <sub>e</sub>	0	39.87						L <sub>j</sub>	0	44.88					
	1	21.72	13.63	27.30	40.97	0.40	0.17		1	45.88	32.33	44.60	56.87	0.00	0.00
	2	13.64	10.28	26.77	43.26	0.40	0.19		2	33.37	30.07	45.01	59.96	0.00	0.00
	3	25.30	7.20	25.96	44.72	0.40	0.21		3	42.78	28.88	46.03	63.19	0.02	0.00
	4	25.06	4.96	25.30	45.65	0.38	0.23		4	45.87	28.07	46.83	65.59	0.03	0.00
	5	22.55	3.15	24.70	46.25	0.38	0.23		5	37.17	27.57	47.59	67.60	0.05	0.00
	6	40.43	1.70	24.16	46.63	0.38	0.23		6	45.39	27.26	48.27	69.29	0.06	0.02

링 주기를 찾아낼 수 있다. 즉,  $\alpha$ 가 congestion 상태인지를 판단하기 위한 확률 임계값이라고 할 때  $(ALL(P_{a,b}(t+l)) \leq \alpha)(l=1,2,\dots,6)$ 을 모두 만족하는  $l$ 을 <표 2>를 통해서 찾아낼 수 있다. <표 3>은 기준 확률 임계값에 따라 각 장비를 폴링할 때 가장 적합한 폴링 주기를 나타낸다.

<표 3>에서 보여주는 결과를 분석해 보면  $\lambda_x = \{40,90\}$ 인 경우는 실제 네트워크 상의 트래픽이 이 임계값을 초과할 정도로 과다하게 발생하지 않았기 때문에 확률 임계값이 낮은  $\alpha=0.5$ 인 경우를 제외하고는 모두 그 다음  $t+1$ 시점이 폴링이 적합한 주기로 나왔다. 그러나,  $\lambda_x = \{30,80\}$

<표 3> 최적 폴링 주기

폴링 대상	$\alpha = 0.5$		$\alpha = 0.6$		$\alpha = 0.7$	
	$\lambda_x = \{30,80\}$	$\lambda_x = \{40,90\}$	$\lambda_x = \{30,80\}$	$\lambda_x = \{40,90\}$	$\lambda_x = \{30,80\}$	$\lambda_x = \{40,90\}$
Seoul	$t+1$	$t+1$	$t+1$	$t+1$	$t+1$	$t+1$
Pride	$t+3$	$t+1$	$t+2$	$t+1$	$t+1$	$t+1$
Susun	$t+3$	$t+1$	$t+2$	$t+1$	$t+1$	$t+1$
Edu	$t+3$	$t+1$	$t+2$	$t+1$	$t+1$	$t+1$
Sam	$t+3$	$t+1$	$t+2$	$t+1$	$t+1$	$t+1$
Internet	$t+2$	$t+1$	$t+2$	$t+1$	$t+2$	$t+1$
Daejun	$t+4$	$t+3$	$t+4$	$t+1$	$t+3$	$t+1$

으로 선로 이용률 임계값을 낮췄을 때는 확률 임계값  $\alpha$ 에 따라 적합한 폴링 시점을 판별할 수 있었다. 먼저  $\alpha=0.5$ 인 경우에는 확률 임계값이 너무 낮게 설정되어 폴링 시점의 변화가 심하고 최대 폴링 주기까지 도달하여 적합한 폴링 시점을 판별하기가 어렵다.  $\alpha=0.5$ 와  $\alpha=0.6$ 일 때 대상 장비가 Internet과 Daejun의 경우에는 선로  $L_h$ 와  $L_i$ 의 이용률이 높기 때문에 최대 폴링 주기 안에서 확률 임계값을 만족시키는 시점을 찾을 수 없었다. 따라서, 이 중 가장 낮은 확률을 가지는 시점인  $t+2$ 와  $t+4$  시점이 선택되어졌다.

결과적으로 확률 임계값은 네트워크의 트래픽의 정도에 따라서 달라질 수 있는데 본 실험의 경우에는  $\alpha=0.7$ 일 경우에 최대 폴링 주기까지 도달하지 않고 적합한 폴링 시점을 선택하여 최대 성능을 나타내었다.

#### 4.2 기존 폴링 기법과의 성능 비교

본 논문에서 제안하는 방법의 성능을 평가하기 위해 기존의 폴링 기법들이 제안하는 방식과 비교하는 것이 요구된다. 기존의 폴링 기법들을 구분하면 네트워크의 congestion과 관련하여 이전 폴링의 Round Trip Time 또는 선로 이용률을 기준으로 폴링 시점을 동적으로 변경시키는 기법들과 폴링 대상이 되는 시스템의 프로세싱 로드를 기준으로 폴링 주기를 변경시키는 방법 등이 제시되었다. 기존의 네트워크 congestion 관점에서의 제시된 폴링 기법들은 이전 시점의 Round Trip Time 또는 선로 이용률을 기준으로 다음 시점의 폴링 주기를 변경하는 것이기 때문에 폴링하고자 하는 시점에서 네트워크의 congestion 상태가 변했을 경우 이를 적용시킬 수 없다. 따라서, 이와 같은 기존 기법들이 제시했던 이전 시점의 상태를 기준으로 하는 기법들과 본 논문에서 제시하는 다음 시점을 예측하여 폴링 주기를 변경하는 기법이 네트워크 congestion을 제대로 반영하지 못한 횟수를 비교하면 그 성능의 차이를 알 수 있다. 본 실험의 경우에는 <표 2>에서  $L_c$ 의  $l=0$ 인 경우 선로 이용률 값이 41.11이기 때문에 이를 기준으로 하면 선로 이용률 임계값을 넘지 않았기 때문에 다음 시점에 현재 폴링 주기를 그대로 사용하여 폴링을 수행할 것이다. 그러나, 실제로 폴링이 일어나는 그 다음 시점  $l=1$ 인 경우의 선로 이용률 값을 보면 81.66으로 임계값 80%를 기준으로 할 경우에 네트워크의 congestion 상태를 알 수 있다. 따라서 이 경우 기존의 폴링 기법은 변화된 선로의 상태를 반영하지 못했다.  $L_h$ 와  $L_i$ 의 경우도 마찬가지로 그 값과 임계값에 따라서 차이가 있을 수 있으나 실제 네트워크 상태를 반영하지 못했다는 것을 알 수 있다. 다음의 <표 4>는 기존의 폴링 기법들과 제안하는 방식이 정확하게 네트워크 상태를 반영하지 못한 횟수를 보여주는 표이다. 이를 보면 제안하는 폴링 기법에 더 정확하게 네트워크 상태를 반영하는 것을 볼 수 있으며 따라서 네트워크 상태에 따른 적

합한 폴링 주기 변경을 수행한다.

<표 4> 네트워크 상태의 부정확 횟수

폴링 횟수	1	5	10	20
기존의 폴링 기법	3	11	21	39
제안하는 폴링 기법	0	3	7	14

### 5. 결 론

네트워크의 상태를 분석하기 위한 네트워크 모니터링의 가장 중요한 기능 중에 하나가 폴링 기법이다. 그러나, 네트워크 상의 트래픽이 많아서 네트워크가 congestion 상태이거나 bottleneck 구간에서는 과도한 폴링 데이터가 네트워크의 상태를 더욱 더 심각한 congestion 상태로 만든다. 따라서, 기존에 다양한 형태의 폴링 기법들이 제시되었다. 이와 같은 기존의 여러 폴링 기법들은 이전에 폴링한 Round Trip Time 또는 선로 이용률을 통해 네트워크 congestion 또는 에이전트의 로드 여부를 판단하여 폴링 주기를 변경하여 폴링 트래픽을 조절하였으나 이는 이전 시점의 폴링을 근거로 폴링 주기를 변경하는 기법이기 때문에 폴링하고자 하는 당 시점의 네트워크 상태를 반영하지 못한다.

본 논문에서는 선로의 이용률 예측을 기반으로 하여 폴링 주기를 동적으로 변경시킬 수 있는 새로운 폴링 기법을 제시하였다. 본 논문에서 제시하는 기법은 과거의 데이터를 근거로 하여 현재 폴링을 수행할 경우 폴링 경로 상의 선로 이용률 임계값을 위반하는지를 예측하여 이를 통해 폴링 주기를 변경함으로써 네트워크가 더욱 심각한 congestion 상태가 되지 않도록 한다. 예측 모델로 AR 모델을 사용하여 네트워크를 구성하는 각 선로의 이용률을 예측하였고 임계값 위반 여부를 확률에 근거하여 판단하였다. 또한, 제시한 기법을 실제 네트워크에 적용하여 적합성 여부를 평가하였고, 이용률 임계값을 ethernet 구간에서는 30%와 40%, 그리고 serial 구간에서는 80%, 90%로 설정하였고, 임계값 위반 확률도 각각 0.5, 0.6, 0.7로 설정하여 경우에 따라 적합한 폴링 시점을 계산하였으며, 이를 통해 적절한 선로 이용률 임계값과 임계값 위반 확률을 판단함으로써 본 기법의 성능을 최대화하였다. 또한, 실제 선로의 이용률과 예측값과의 비교를 통하여 이전 기법들과 제시된 기법과의 성능 비교를 수행하였다. 본 기법은 선로 이용률을 기준으로 하여 최적의 폴링 시점을 판단하였지만, 본 기법을 선로 이용률 이외에 다양한 기준에 적용하는 것도 가능할 것이다.

### 참 고 문 헌

[1] K. Ohta, N. Sun, G. Mansfield, and Y. Nemoto, "Effective polling control for network management using SNMP," IEICE Technical Reports on Information Network, IN94-

135 : 91-96, Nov., 1994.

[2] T. Ika, K. Ohta, N. Kato, G. Mansfield, and Y. Nemoto, "Management information gathering on wide-area network in consideration of information accuracy," Proc. Of the 1997 IEICE General Conference, B-7-46 : 175, Mar., 1997.

[3] P. Moghe and M. Evangelista, "Rap-rate adaptive polling for network management applications," In Proc. Of IEEE NOMS'98, pp.395-399, 1998.

[4] B. Bondi, "A nonblocking mechanism for regulating the transmission of network management polls," In Proc. Of IFIP/IEEE IM'97, pp.565-580, 1997.

[5] P. Dini and R. Boutaba, "Deriving variable polling frequency policies for pro-active management in networks and distributed systems," In Proc. Of IFIP/IEEE IM'97, pp.541-552, 1997.

[6] P. Dini, G. Bochmann, T. Koch and B. Kramer, "Agent based management of distributed systems with variable polling frequency policies," In Proc. of IFIP/IEEE IM'97, pp.553-564, 1997.

[7] K. Yoshihara, K. Sugiyama, H. Horiuchi, S. Obana, "Dynamic Polling Scheme Based on Time Variation of Network Management Information Values," In Proc. of IFIP/IEEE'99, pp.141-154, 1999.

[8] 안성진, 정진욱, "SNMP MIB\_II를 이용한 인터넷 분석 파라미터 계산 알고리즘에 관한 연구," 정보처리학회, 제5권 제8호, pp.2102-2116, 1998.

[9] Cynthia S. Hood, Chuanyi Ji, "Intelligent Agents for Pro-active Fault Detection," IEEE Internet Computing, Vol.22, pp.65-72, 1998.

[10] Cynthia S. Hood, Chunanyi Ji, "Beyond Thresholds : An alternative Method for Extracting Information from Network Measurements," GLOBECOM'97, Vol.1, pp.487-491, 1997.

[11] Hellerstein, J.L, Fan Zhang, shahabuddin, "An approach to Predictive Detection for Service Management," Integrated Network Management, pp.309-322, 1999.

[12] William W. S. Wei, "Time Series Analysis : Univariate and Multivariate Methods," Addison-Wesley Publishing Company, 1990.

[13] C. Chatfield, "The Analysis of Time Series : An introduction," Chapman&Hall, 1989.



**조 강 홍**

e-mail : khcho@songgang.skku.ac.kr  
 1997년 성균관대학교 정보공학과 학사  
 1999년 성균관대학교 전기전자 및 컴퓨터 공학부 석사  
 1999년~현재 성균관대학교 전기전자 및 컴퓨터 공학부 박사과정

관심분야 : 네트워크 관리, 트래픽 분석



**안 성 진**

e-mail : sjahn@comedu.skku.ac.kr  
 1988년 성균관대학교 정보공학과 졸업 (학사)  
 1990년 성균관대학교 대학원 정보공학과 졸업 (석사)  
 1990년~1995년 한국전자통신연구원 연구 전산망 개발실 연구원

1996년 정보통신 기술사 자격 취득  
 1998년 성균관대학교 대학원 정보공학과 졸업 (박사)  
 1999년~현재 성균관대학교 컴퓨터교육과 조교수  
 관심분야 : 네트워크 관리, 트래픽 분석, Unix 네트워킹



**정 진 욱**

e-mail : jwchung@songgang.skku.ac.kr  
 1974년 성균관대학교 전기공학과 학사  
 1979년 성균관대학교 대학원 전자공학과 석사  
 1991년 서울대학교 대학원 계산통계학과 박사

1982년~1985년 한국과학기술 연구소 실장  
 1981년~1982년 Racal Milgo Co. 객원연구원  
 1985년~현재 성균관대학교 전기전자 및 컴퓨터공학부 교수  
 관심분야 : 컴퓨터 네트워크, 네트워크 관리, 네트워크 보안