2015 International Conference on Computing, Networking and Communications, Communication QoS and System Modeling Symposium

Network Bandwidth Utilization Forecast Model on High Bandwidth Networks

Wucherl Yoo and Alex Sim
Lawrence Berkeley National Laboratory, Email: {wyoo,asim}@lbl.gov

Abstract—**With the increasing number of geographically distributed scientific collaborations and the growing sizes of scientific data, it has become challenging for users to achieve the best possible network performance on a shared network. We have developed a model to forecast expected bandwidth utilization on high-bandwidth wide area networks. The forecast model can improve the efficiency of resource utilization and scheduling of data movements on high-bandwidth networks to accommodate ever increasing data volume for large-scale scientific data applications. A univariate forecast model is developed with STL and ARIMA on SNMP path utilization data. Compared with traditional approach such as Box-Jenkins methodology to train the ARIMA model, our forecast model reduces computation time by 83.2%. It also shows resilience against abrupt network usage changes. Its forecast errors are within the standard deviation of the monitored measurements.**

Keywords*—*Forecasting, Network, Time series analysis

I. Introduction

With advances in large scale experiments and simulations, the data volume of scientific applications has rapidly grown. Even with advances in network technology, it has become more challenging to efficiently coordinate network resources and to achieve best possible network performance on a shared network. It is also challenging to build a forecast model for network bandwidth utilization with accurate and fine-grained prediction due to the computational complexities. To support efficient resource management and scheduling data movement for ever increasing data volume in extreme-scale scientific applications, we have developed an analytical model in order to characterize and forecast [1] bandwidth utilization on a high-bandwidth wide area network (WAN).

The forecast model can improve the efficiency of network bandwidth resource utilization. In addition, it can help efficient resource scheduling and path finding for data transfers. The goal of this paper is to model the network bandwidth utilization between two sites to support data flow timing and parameter decisions as well as network topology or link planning. Our modeling efforts can help systematic data transfer decisions without over/under-provision. One of our previous works proposed a network reservation framework to provide guaranteed bandwidth on ESnet [4]. Our forecast model can complement this type of reservation system or a system to select alternate paths for large data transfer.

It is better for the model to be computationally efficient and comparably accurate in order to forecast multiple paths of users' interests. We select a size of an appropriate training set that shows relatively accurate forecast error with manageable computational overhead. In addition, we have studied the effect of variability of the bandwidth usage on the forecast accuracy and the appropriate threshold to make our model resilient against the abrupt usage change.

The experimental data on Simple Network Management Protocol (SNMP) link utilization has been collected by ESnet [1] in 2013 and 2014 on each router. Our experiments use SNMP data from 6 directional paths connecting a pair of large data facilities described in Sec. IV-A. The SNMP data consists of the size of bandwidth utilization and time-scale as 30 seconds interval. The maximum size of bandwidth utilization is extracted at each interval from the routers in each path, which represents bandwidth utilization in the path.

We have developed the forecast model as a univariate time series model. The first step is to remove the seasonality in the measurement data, and we use Seasonal Decomposition of Time Series by Loess (STL) [7]. STL decomposes the SNMP data into the time series of seasonality, trend and remainder. After deducting seasonality component, we use AutoRegressive Integrated Moving Average (ARIMA) on the seasonally adjusted time series. The orders of ARIMA model are selected in an automated mechanism based on the assumption of stationary time series about the SNMP data. We have observed that there are no significant changes in the average bandwidth utilization in the training dataset window (up to 8 weeks) throughout 2013 and 2014. We show that our stationary assumption is appropriate for the SNMP data in Sec. IV-C. Our forecast model reduces computation time for forecast by 83.2% compared to the traditional approach such as Box-Jenkins methodology [6] to find the best fit forecast model using ARIMA.

The rest of paper is organized as follows. Sec. II presents related work. Sec. III demonstrates the model design and implementation. Sec. IV presents experimental evaluation of the forecast model, and the conclusion is in Sec. V .

II. Related Work

The self-similarity of network traffic has been studied in LAN [16], WAN [20], and the Internet [9]. It allows to use past history to forecast near-term future. Qiao et al. [21] presented an empirical study of the forecast error on different time-scales, showing that the forecast error does not monotonically decrease with smoothing for larger time-scale.

Benson el al. [5] studied network traffic patterns in data centers using SNMP data. Balman et al. [4] proposed a network reservation framework to provide guaranteed bandwidth. Our forecast model complements these works by providing traffic forecast information.

Available bandwidth can be estimated by sending probe packets. Shriram et al. [24] conducted a comparison study on the

[1] We explicitly make a distinction between forecast and prediction. We use forecast as an estimation of *future* values based on the analytical model built from the past observations. On the other hand, we use prediction as an estimation of values based on analytical model.

estimation of available bandwidth from various measurement tools. Aceto et al. [2] proposed end-to-end available bandwidth measurement infrastructure. Our model focuses on the forecast of available bandwidth using passive measurements from routers instead of estimation from probing packets.

Several prediction models of TCP data transfers have been proposed. Throughput prediction models were proposed for large TCP transfers [11]. Mirza et al. [18] used a machine learning mechanism to predict TCP throughput. While these works are restricted to predict TCP data transfers, our model forecasts aggregated throughput for a network path.

Several models have been proposed to forecast network traffic. Sang et al. [22] proposed short-term (a few minutes) forecast model using ARMA with 1 sec time-scale data. Papagiannaki et al. [19] proposed long-term (1 year) forecast model of Internet backbone traffic using ARIMA with 1 week time-scale data. Krithikaivasan et al. [14] proposed mid-term (1 day) forecast model using ARCH model with 15 minute time-scale data. Our model focuses on mid-term (1 day) forecast of the bandwidth utilization using 30 second time-scale data. The number of forecast points is order of magnitude larger than these models due to smaller time-scale. Therefore, our forecast model requires more computation and accuracy than these proposed models. Our model overcomes these challenges by seasonal adjustment and stationary assumption, which are not discussed in the previous models.

III. Model Development

We have developed the forecast model as a univariate time series model. A forecast model estimates the future values using the observed SNMP data up to time n $(x_1, x_2, \cdots, x_n)$. The forecast of h steps ahead is denoted as $\hat{x}_n(h)$ at time $n+h$. When the observed value (x_{n+h}) is available at time $n+h$, we calculate the forecast error denoting $e_n(h)$ as $x_{n+h} - \hat{x}_n(h)$.

A. Logit Transformation

The theoretical maximum value of possible size of the SNMP is 10^{10} bits, and the minimum value is 0 bit in 100 Gbps bandwidth of the current ESnet. Logit transformation is applied to the SNMP data x to set the lower and upper bounds based on these limits. Time series data x containing n observations is transformed to time series data y with lower bound a and upper bound b (10^{10} bits/second) as denoted as $y = \text{logit}(x) = \log(\frac{x-a}{b-x})$. The lower bound, a is approximated to 1 instead of 0 bit/s. While there are very few observed cases when no transfer occurs, this lower bound approximation can be ignored in the 100 Gbps networks.

B. Seasonal Adjustment

After the logit transformation, the transformed SNMP data y is seasonally adjusted. Removing seasonal components from the time series allows the analysis of the non-seasonal trend. This is essential to project the trend and the seasonality of past history to future values. We use Seasonal Decomposition of Time Series by Loess (STL) [7] for this seasonal adjustment. STL decomposes the logit transformed SNMP data into the components of seasonality S, trend T and remainder R as denoted as $y = y_t = S_t + T_t + R_t$.

STL applies a sequence of smoothing from Loess (Locally Weighted Regression Fitting) [8], which estimates nonlinear relationships among S, T, and R. This smoothing sequence progressively refines and improves the estimates of the seasonal and trend components. There exist several parameters to derive the STL model. The seasonal cycle is evaluated with possible choices such as minute, hour, day and week. The smoothing windows for the seasonality (n_s) for trend (n_t) are evaluated with different values. After the decomposition, we seasonally adjust SNMP data by deducting seasonality component denoted as $y\prime = y_t\prime = y_t - S_t = T_t + R_t$.

C. Bandwidth Utilization Forecast

The forecast model is developed by using AutoRegressive Integrated Moving Average (ARIMA) on the seasonally adjusted time series, $y\prime$. ARIMA model consists of the orders of autoregressive process (p), differences (d) and moving average (q). They are selected in an automated mechanism as follows. First, the stationarity of the time series is confirmed by Kwiatkowski-Phillips-Schmidt-Shin (KPSS) test [15]. When the stationarity is confirmed, d is selected as 0. Otherwise, d is selected as 1, which is sufficient to make the non-stationary time series to stationary in the experiments. We use Akaike's Information Criterion (AIC) [3] to automatically select the modeling parameters as shown in the Box-Jenkins methodology [6]. AIC represents the sum of the maximum log likelihood for the estimation and the penalty from the orders of selected model. This combination allows simpler models with less numbers of orders unless the they show severely low likelihood for the estimation. We calculate AIC with different combinations of p and q incrementing from 1 until the sum of p and q reaches to a certain maximum value. The model choice from AIC converges, and is asymptotically equivalent to that of cross-validation [23]. The best model with p and q is chosen with the least value of AIC. In our case, the maximum sum of p and q is 10, which is the smallest sum that can result in reasonably accurate forecast from the experiments.

After the orders of the ARIMA model are selected, we fit the model with the seasonally adjusted time series $(y_n\prime)$ and with the training set of n observed data.The forecast of h time steps ahead $(\hat{y}_h)$ is computed by adding seasonal component to the fitted model, which is denoted as $\hat{y}_h = \hat{y}_h\prime + \hat{S}_{n+h}$. The seasonality forecast $(\hat{S}_{n+1}, \hat{S}_{n+2}, \cdots, \hat{S}_{n+h})$ is computed by simply repeating the decomposed seasonal component $(S_1, S_2, \cdots, S_n)$. Then, these forecast values are converted to the original scale using the reverse logit transformation.

IV. Experimental Results

A. Experimental Setup

In the experiments, we used 6 directional paths connecting two sites on ESnet in the US. The paths consist of 6 or 7 links connected with the routers between NERSC in CA, ORNL in TN and ANL in IL. PID is the path identification: `P1` and `P2`, paths between NERSC and ORNL, `P3` and `P4`, those between NERSC and ANL, and `P5` and `P6` between ORNL and ANL. The measured SNMP data represents the bandwidth utilization during a 30 second interval at routers in each path. We selected the maximum value on a link in each path for an aggregated measurement. The experiments were conducted on a machine with 8-core CPU, AMD Opteron 6128 and 64 GB memory. To reduce overall execution time, we parallelized the computational tasks of parameter searching, fitting and calculating the forecast error.

The SNMP data was used from Feb. 10 to Feb. 16, 2014 as a test dataset. The duration for the training dataset is from 1 to 8 weeks prior to Feb. 10, 2014. We evaluated the forecast error by a cross-validation mechanism for time series data proposed by Hijorth [12]. The original method by Hijorth computes a weighted sum of one-step-ahead forecasts by rolling the origin when more data is available. Similarly, we computed the average forecast error for 1 week by forecasting one target day ($h = 1, \cdots, 2880$) and rolling 6 more days. We compared the cross-validation results of the forecast errors as Root Mean Squared Error (RMSE) in the following sections. RMSE [2] is in bit/s, and calculated with $RMSE(h) = sqrt(\frac{1}{h} \cdot \sum_{i=1}^{h} (e_n(i))^2)$. After forecasting the first day of the test set $(\hat{x}_1, \cdots, \hat{x}_{2880})$, the forecast error for the first target day $RMSE(h_{day1}) = RMSE(h)$ was computed. The forecast error for the second target day $RMSE(h_{day2}) = MAE(h + h)$ was computed by adding the observations from the first target day to the previous training set $(x_1, \cdots, x_n, \mathbf{x_{n+1}}, \cdots, \mathbf{x_{n+h}})$. This process was repeated for the next 5 target days to compute the average of RMSE for the one-week test set.

B. Seasonality Analysis

The STL model was derived by using the parameters described in Sec. III-B. The seasonal cycle was evaluated with possible cyclic periods such as minute, hour, day and week. The smoothing parameter for the seasonality (n_s) was evaluated with possible values such as the same value with n_p or multiples or inverse multiples of n_p. The smoothing parameter for trend (n_t) was also evaluated with different values. With larger n_t, the Interquartile Range (IQR) of the trend component got smaller. This is because smoothing from Loess [8] of the trend component gets smoother with larger n_t, and this result is increasing IQR of the remainder component.

Different values of seasonality smoothing window (n_s) showed the similar forecast accuracy. The IQR of the seasonal component did not change with different n_s. In addition, trend smoothing window (n_t) changed the shape of trend, but did not change forecast accuracy. As a result, we selected n_s and n_t as the same as n_s. While the shape and IQR were changed with different n_t, the forecast error was still similar. This suggests that the ARIMA is more crucial component than STL in our forecast modeling. However, fitting with STL is essential since it removes seasonal component from the original time series. Using the Seasonal ARIMA or the ARIMA without STL appeared to be another possible choice, however, computation time of the modeling these choices took too long to conduct the experiments. Only after seasonal adjustment, the computation time of the ARIMA modeling was viable.

It is well known that Internet traffic has cyclic self-similarity in daily interval [10]. The average forecast error (RMSE) with daily seasonality was 4.9% better than that of weekly seasonality and 2.8% better than that of hourly seasonality. This result shows that SNMP data has stronger daily self-similarity than hourly and weekly periods, similar to the Internet traffic. The average of Hurst parameters [9] from `P1` to `P6` were 0.92, 0.94, 0.93, 0.89, 0.94, and 0.87 respectively, which confirms the self-similarity. While the remainder of STL decomposition did not pass the Ljung-Box test [17] to check whether autocorrelation still exists, which led us to use ARIMA to remove existing autocorrelation from the seasonally adjusted time series.

C. Bandwidth Utilization Forecast

We compared possible modeling choices including parameter selections. The model was developed based on the Box-Jenkins methodology [6], using ARIMA on seasonally adjusted SNMP data by STL. The orders of ARIMA model (p,d,q) were selected in the automated mechanism in Sec. III-C. After fitting the forecast model with the selected parameters, Ljung-Box test was conducted to check whether the overall residuals are similar to the white noise, and the residuals of the forecast model passed the test.

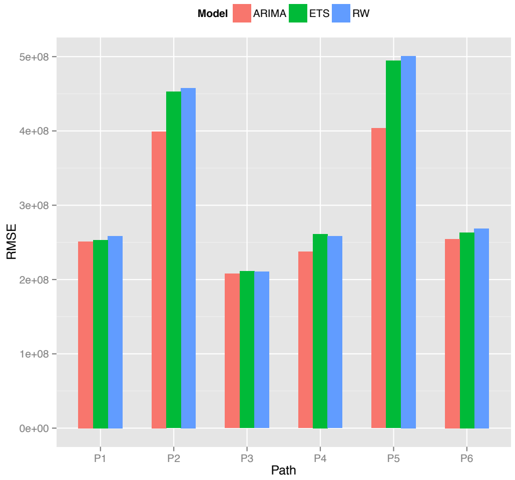

Fig. 1. Forecast Error Comparison for Different Forecast Models on Seasonally Adjusted Data: The training set duration is 8 weeks. The seasonal cycle is one week.

Forecast Methodology: We tested the possible forecast methods on seasonally adjusted time series data by STL. Fig. 1 illustrates the comparison of forecast errors for different forecast models, ARIMA, Exponential smoothing state space model (ETS) [13] and Random Walk (RW) [6]. The forecast error of ARIMA is the lowest, which led us to use ARIMA in the forecast model.

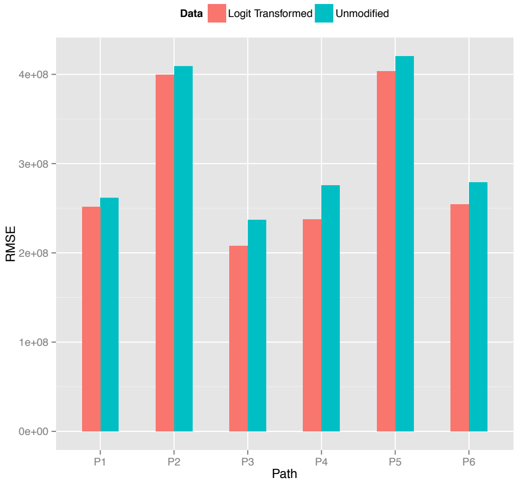

Fig. 2. Forecast Error Comparison for Logit Transformation: The training set duration is 8 weeks. The seasonal cycle is one week.

Logit Transformation: Fig. 2 shows the forecast errors for the logit transformed data compared to the unmodified data.

[2] As the SNMP data is collected in every 30 second interval, the size of bandwidth utilization is normalized by dividing by 30. Note that RMSE is in the same order of the bandwidth. MAE and ME are also shown in Table I.

The forecast errors were derived from the forecast models using STL and ARIMA described in Sec. III-B and Sec. III-C. The forecast error (RMSE) after the logit transformation was consistently more accurate for each path. It was 8.5% better with the logit transformation than without the logit transformation. This is because the logit transformation sets the lower and upper bounds in the modeling and fitting procedures, which helps reduce the potential under-estimation and over-estimation from the forecast.

D. Training Set Size

Intuitively, there is a tradeoff between less computational requirement with smaller training set size and better accuracy with larger training set size. Smaller training set size certainly requires less CPU time and storage including memory and disk. Larger training set size does not always guarantee better forecast accuracy because adding more historical data to the forecast modeling may not improve forecast accuracy due to possible deviances on the older history from the recent history. In addition, past history has lesser degree of impact than recent history on the forecast models such as ARIMA. To find the best possible training set size, we have studied the effect of different sizes of data on forecast accuracy. Fig. 3 shows that training set size with around 4 weeks results in better forecast accuracy. We also tested training set sizes of 8, 16, 24, 32, 40, or 48 weeks. However, the differences of forecast errors when using larger sizes were marginal.

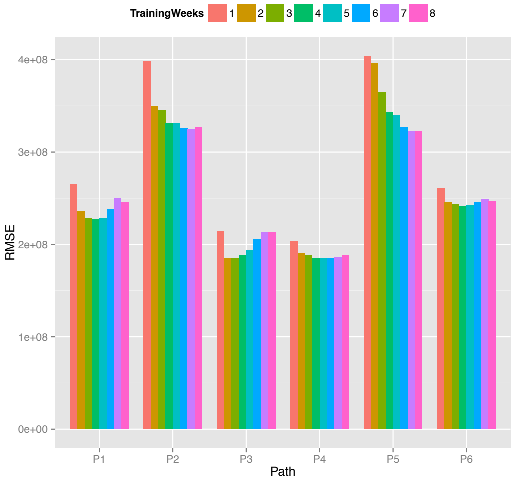

Fig. 3. Forecast Error for Different Training Set Sizes: The training set size varies from 1 to 8 weeks until Feb. 9, 2014. The seasonal cycle is one day.

E. Delayed Model Update

Fig. 4 shows the forecast errors for the delayed model update. We observed that even when KPSS test [15] did not confirm the stationarity, the time series did not drift significantly. Thus, we evaluated whether the stationary assumption of SNMP data was appropriate even when the KPSS test result suggested non-stationary. We observed that the variances of training set and sudden bandwidth utilization changes made the test results inaccurate in some cases. The stationary assumption results the forecast error (RMSE) 10.9% less than that of forecast without the assumption. As we observed stationarity of SNMP data up to 8 weeks in the training dataset, this observation led to a hypothesis that delaying model updates at least one week would not degrade the forecast error, instead of updating and re-fitting the model whenever new measurement data is available. We updated the minimal parts such as auto-correlation and moving averages from the initially fitted model.

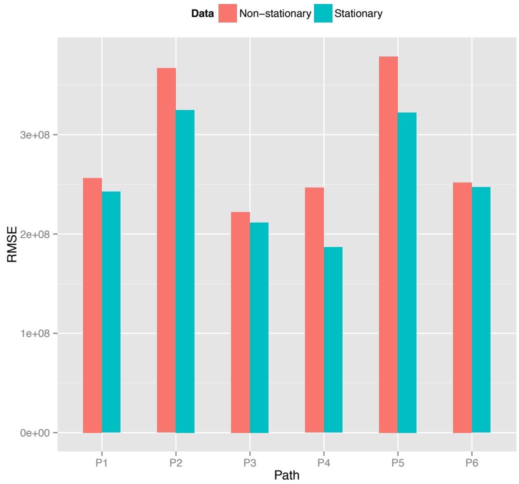

Fig. 4. Forecast Error Comparison for Stationary Assumption: The training set size is 8 weeks until Feb. 9, 2014. The seasonal cycle is one day.

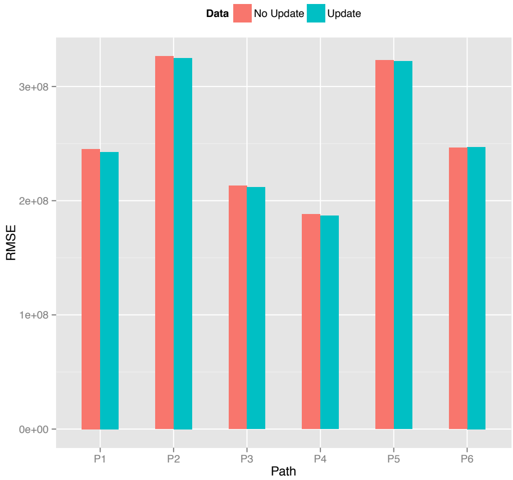

Fig. 5. Forecast Error Comparison for Delayed Model Update: The training set size is 4 weeks. The seasonal cycle is one day.

Fig. 5 shows the forecast errors for our delayed model update. The result shows that the accuracy was not degraded, and it improved the computation time by 83.2% compared to traditional approach such as the Box-Jenkins methodology [6] with updating the models in daily period. The average computation time from the delayed model update took 158 seconds to forecast 7 days duration ahead per path compared to 938 seconds from the model updated once for each day.

Fig. 6 shows the forecast result of the delayed model update for one day test set on Feb. 10, 2014 in `P1`. It shows that our blue-colored forecast values are close to the black-colored observed data. Table I shows the standard deviations of the training set of 4 weeks and the test set from Feb. 10 to Feb. 16 in 2014. This result validates the efficacy of our forecast model. When sudden spikes in the bandwidth utilization were observed from the training set, our forecast model was resilient to those sudden changes and accurate with RMSE within the standard deviations of the test sets.

Since Mean Error (ME) is much closer to 0 than Mean Absolute Error (MAE) in Tab. I, the forecast would be more accurate for large data transfers. This is because the forecast errors are mixed with positive and negative values. When the transfer time is longer than 30 seconds (10 TB transfer takes 800 seconds at theoretical maximum throughput on a 100Gbps network.), the aggregated forecast errors from the large data transfer would decrease.

TABLE I. FORECAST ERROR METRICS. THE VALUES ARE EXPRESSED AS GBPS (10^8 BIT/S). SD_{Train} IS THE STANDARD DEVIATION OF THE TRAINING SET. SD_{Test} IS THAT OF THE TEST SET. RMSE, MAE, ME IS THE DIFFERENT TYPES OF FORECAST ERRORS OF CROSS-VALIDATION.

PID	SD_{Train}	SD_{Test}	RMSE	MAE	ME
P1	4.13	2.36	2.27	1.72	0.29
P2	4.51	3.37	3.31	2.59	-0.58
P3	4.01	2.07	1.88	1.45	0.47
P4	3.03	2.06	1.85	1.46	0.10
P5	4.64	3.40	3.42	2.74	-1.04
P6	4.00	2.54	2.42	1.79	0.30

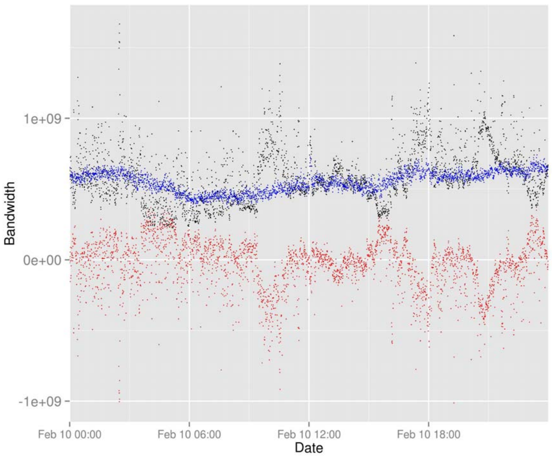

Fig. 6. Bandwidth Utilization Forecast for P1: The x axis is the date. The y axis is the bandwidth (bit/s). Black colors are for the observed data. Blue colors are for the forecasts. Red colors are for the forecast errors.

V. CONCLUSIONS

We present a network bandwidth utilization forecast model. It can support efficient network resource utilization, efficient scheduling and alternate path finding, and planning on network link/bandwidth provision for high-bandwidth networks. Since data sharing opportunities over the wide-area network increase for large-scale scientific data applications that generate large volume of data, it is challenging to efficiently coordinate network resources on a shared network. In addition, sudden bandwidth utilization change makes forecast more challenging. We observe that the network traffic behavior for the scientific networks shows stationarity and self-similarity in daily periodicity. Logit transformation and stationary assumption show effectiveness in reducing the forecast error by 8.5% and 10.9% respectively. Our experimental results show that the delayed model update reduces the computation time by 83.2% compared to the traditional Box-Jenkins approach. It does not show the degradation of the forecast error when reducing the frequency of the model updates, and it shows the resiliency when there are sudden network bandwidth utilization changes. Our forecast model is accurate having RMSE within the standard deviations of the observed measurements. It can be applicable to other time series data with daily seasonality such as vehicular traffic data. The future work includes the adaptive forecast model based on the long-term/short-term trend changes of bandwidth utilization and the application of the forecast model to the network provisioning.

VI. ACKNOWLEDGMENTS

This work was supported by the Office of Science, the U.S. Dept. of Energy under Contract No. DE-AC02-05CH11231.

REFERENCES

[1] "Energy Sciences Network (ESnet)," http://www.es.net/, 2014.

[2] G. Aceto, A. Botta, A. Pescapé, and M. D'Arienzo, "Unified architecture for network measurement: The case of available bandwidth," *J. Netw. Comput. Appl.*, vol. 35, no. 5, pp. 1402–1414, Sep. 2012.

[3] H. Akaike, "A new look at the statistical model identification," *IEEE Trans. Automat. Contr.*, vol. 19, no. 6, pp. 716–723, 1974.

[4] M. Balman, E. Chaniotakisy, A. Shoshani, and A. Sim, "A Flexible Reservation Algorithm for Advance Network Provisioning," in *Proc. Int. Conf. High Perform. Comput. Networking, Storage Anal.* ACM/IEEE, Nov. 2010.

[5] T. Benson, A. Akella, and D. A. Maltz, "Network traffic characteristics of data centers in the wild," in *Proc. Conf. Internet Meas. - IMC '10*. ACM, Nov. 2010, p. 267.

[6] G. E. P. Box, G. M. Jenkins, and G. C. Reinsel, *Time Series Analysis: Forecasting and Control*, 4th ed. John Wiley & Sons, 2013.

[7] R. B. Cleveland, W. S. Cleveland, J. E. McRae, and I. Terpenning, "STL: A Seasonal-Trend Decomposition Procedure Based on Loess," *J. Off. Stat.*, vol. 6, no. 1, pp. 3–73, 1990.

[8] W. Cleveland and S. Devlin, "Locally weighted regression: an approach to regression analysis by local fitting," *J. Am. . . .*, 1988.

[9] M. Crovella and A. Bestavros, "Self-similarity in World Wide Web traffic: evidence and possible causes," *IEEE/ACM Trans. Netw.*, vol. 5, no. 6, pp. 835–846, 1997.

[10] C. Estan, S. Savage, and G. Varghese, "Automatically inferring patterns of resource consumption in network traffic," in *SIGCOMM '03*. ACM, pp. 137–148.

[11] Q. He, C. Dovrolis, and M. Ammar, "On the predictability of large transfer TCP throughput," in *Proc. Conf. Appl. Technol. Archit. Protoc. Comput. Commun.*, vol. 35, no. 4. ACM, Aug. 2005.

[12] J. Hjorth, *Computer intensive statistical methods: Validation, model selection, and bootstrap*, 1993.

[13] R. J. Hyndman, A. B. Koehler, R. D. Snyder, and S. Grose, "A state space framework for automatic forecasting using exponential smoothing methods," *Int. J. Forecast.*, vol. 18, no. 3, pp. 439–454, Jul. 2002.

[14] B. Krithikaivasan, Y. Zeng, K. Deka, and D. Medhi, "ARCH-Based Traffic Forecasting and Dynamic Bandwidth Provisioning for Periodically Measured Nonstationary Traffic," *IEEE/ACM Trans. Netw.*, vol. 15, no. 3, pp. 683–696, Jun. 2007.

[15] D. Kwiatkowski, P. C. Phillips, P. Schmidt, and Y. Shin, "Testing the null hypothesis of stationarity against the alternative of a unit root," *J. Econom.*, vol. 54, no. 1-3, pp. 159–178, Oct. 1992.

[16] W. Leland, M. Taqqu, W. Willinger, and D. Wilson, "On the self-similar nature of Ethernet traffic (extended version)," *IEEE/ACM Trans. Netw.*, vol. 2, no. 1, 1994.

[17] G. Ljung and G. Box, "On a measure of lack of fit in time series models," *Biometrika*, vol. 65, no. 2, pp. 297–303, 1978.

[18] M. Mirza, J. Sommers, and P. Barford, "A Machine Learning Approach to TCP Throughput Prediction," *IEEE/ACM Trans. Netw.*, vol. 18, no. 4, pp. 1026–1039, Aug. 2010.

[19] K. Papagiannaki, N. Taft, Z.-L. Zhang, and C. Diot, "Long-term forecasting of internet backbone traffic." *IEEE Trans. Neural Networks*, vol. 16, no. 5, pp. 1110–1124, Sep. 2005.

[20] V. Paxson and S. Floyd, "Wide area traffic: the failure of Poisson modeling," *IEEE/ACM Trans. Netw.*, vol. 3, no. 3, 1995.

[21] Y. Qiao, J. Skicewicz, and P. Dinda, "An empirical study of the multiscale predictability of network traffic," in *Proc. Int. Symp. High Perform. Distrib. Comput.* IEEE, 2004, pp. 66–76.

[22] A. Sang and S.-q. Li, "A predictability analysis of network traffic," *Comput. Networks*, vol. 39, no. 4, pp. 329–345, Jul. 2002.

[23] J. Shao, "An asymptotic theory for linear model selection," *Stat. Sin.*, vol. 7, pp. 221–264, 1997.

[24] A. Shriram and J. Kaur, "Empirical Evaluation of Techniques for Measuring Available Bandwidth," in *Proc. Int. Conf. Comput. Commun.* IEEE, 2007, pp. 2162–2170.