2018 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)

Large Data Transfer Predictability and Forecasting using Application-Aware SDN

Deepak Nadig, Byrav Ramamurthy, Brian Bockelman and David Swanson
Department of Computer Science and Engineering,
University of Nebraska-Lincoln, Lincoln, NE, 68588 USA
Email: {deepaknadig, byrav, bbockelm, dswanson}@cse.unl.edu.

***Abstract*—Network management for applications that rely on large-scale data transfers is challenging due to the volatility and the dynamic nature of the access traffic patterns. Predictive analytics and forecasting play an important role in providing effective resource allocation strategies for large data transfers. We propose a predictive analytics solution for large data transfers using an application-aware software defined networking (SDN) approach. We perform extensive exploratory data analysis to characterize the GridFTP connection transfers dataset and present various strategies for its use with statistical forecasting models. We develop a univariate autoregressive integrated moving average (ARIMA) based prediction framework for forecasting GridFTP connection transfers. Our prediction model tightly integrates with an application-aware SDN solution to preemptively drive network management decisions for GridFTP resource allocation at a U.S. CMS Tier-2 site. Further, our framework has a mean absolute percentage error (MAPE) ranging from 6% to 10% when applied to make rolling forecasts.**

I. Introduction

Software defined networking (SDN) and network functions virtualization (NFV) play a vital role in service provisioning and orchestration [1] by acting as catalysts for service innovation and in enabling end-to-end services management. However, for the network operators to successfully balance user expectations with services' resource requirements, a policy-based approach supported by predictive analytics is necessary. Network operators can create consistent user experiences by ensuring a balance between user requirements and (optimized) network resources. The pivotal role of predictive analytics for network management is often overlooked by SDN/NFV based architectures, which instead focus on network services orchestration. To provide quality of service (QoS) and a consistent end-user experience, next-generation network/service management architectures must proactively address user, service, and resource provisioning constraints.

The dynamic nature of large data transfer workflows (e.g. data-intensive science) demand agility in the allocation of service and network resources. The volatility present in traffic usage patterns further compounds resource provisioning problems. Although aggregate and long-term traffic patterns can be identified, they fail to mitigate short-term traffic variability impacts. This results in the deployment of static threshold-based automatic scaling methods for resource provisioning. Autoscaling solutions are reactive in nature, with resource scaling occurring in response to real-time network events. Autoscaling approaches are inefficient as they fail to respond gracefully to sudden spikes or outlier events. Predictive analytic approaches on the other hand combine trend analysis with historical information to predict future usage requirements. Thus, predictive analytics can be used to preemptively drive network management decisions by providing valuable insights into metrics and performance indicators associated with user/network behavior.

In this paper, we propose a predictive analytics approach to forecasting large data transfers to/from high-performance computing centers in campus networks. To evaluate our proposed predictive analytics models for large data transfer forecasting, we rely on data transfer information obtained from data-intensive science workflows such as the Laser Interferometer Gravitational-Wave Observatory (LIGO) [2] and the Compact Muon Solenoid (CMS) [3]. An application-aware SDN solution previously proposed in [4] is used to obtain GridFTP [5] data transfer information. We begin by performing exploratory data analysis to understand the temporal properties of the transfer information dataset. We then develop and evaluate an autoregressive integrated moving average (ARIMA) model for forecasting future data transfers and evaluate its performance.

The main contributions of this paper are:

1) *Exploratory data analysis of the transfer dataset*: We present a systematic approach to exploratory data analysis of the data-intensive science transfer dataset. Our analysis is critical to assessing the quality of the dataset and its suitability for use with the statistical prediction models. Further, the analysis provides valuable insights that aids in model selection, parameter estimation and forecasting.
2) *ARIMA model for forecasting large data transfers*: We develop a non-seasonal autoregressive integrated moving average (ARIMA) model for forecasting large data transfers from data-intensive science workflows. The model also incorporates automatic parameter estimation and model checking to ensure that the temporal structure is preserved post-estimation. We also demonstrate how the model can be employed to make *rolling forecasts*.
3) *Real-world evaluation with a large-scale dataset*: We demonstrate the scalability of our predictive analytics framework to forecast transfer connections by deploying it on a U.S. CMS Tier-2 site. Further, our solution tightly integrates with an application-aware SDN solution.

978-1-5386-8134-3/18/$31.00

The paper is organized as follows: Section II provides a brief overview of application-awareness in the SDN context and its role in predictive analytics and describes related works; Section III presents a detailed exploratory data analysis of our transfer dataset and outlines the various considerations for its use with predictive analytics; In Section IV, we present our ARIMA-based model, our design framework and an algorithm for forecasting GridFTP transfers; Section V presents the performance evaluation results of our model. Finally, in Section VI we conclude our work and discuss the future work.

II. Background and Related Work

A. Application-Aware SDN for Predictive Analytics

Application workflows that rely on large-scale data transfers often place varying and dynamic demands on the underlying network infrastructure. While applications exhibit diverse behavior in resource access/utilization, network management systems must always ensure consistent user experiences. Thus, network management systems must be capable of meeting dynamic user demands on network resources, quality of service (QoS) and network security. While network management systems can employ application-layer metadata to make decisions, information exchange between the network-layer and the application-layer is often limited. We define application-awareness as the exchange of application-layer metadata with the network-layer. This resulting application- and network-layer collaboration can aid network management systems make informed resource allocation decisions. Thus, network management systems can exploit the intelligence provided by application-aware architectures for resource allocation decisions.

By understanding an application's behavior and resource requirements in real-time, a number of network management optimizations are possible. Thus, application-awareness can provide key insights to precisely characterize traffic volatility, resulting in improved forecasting accuracy of resource requirements. In contrast to logging application-layer information, application-awareness provides real-time updates on an application's behavior, requirements and states. Further, the network-layer communicates with the application-layer over a secure communication application programming interface (API) to obtain all application-layer metadata that cannot be obtained by conventional mechanisms. An important advantage of an application-aware solution is that we do not have to resort to external stateful packet processing techniques to obtain application metadata. Traditional techniques such as deep packet inspection (DPI) are both cumbersome and inefficient when employed with large-scale data transfer systems. Also, both stateful packet processing and sniffing fails with applications like GridFTP [5] which use encrypted sessions for connection establishment and data transfer.

B. Related Work

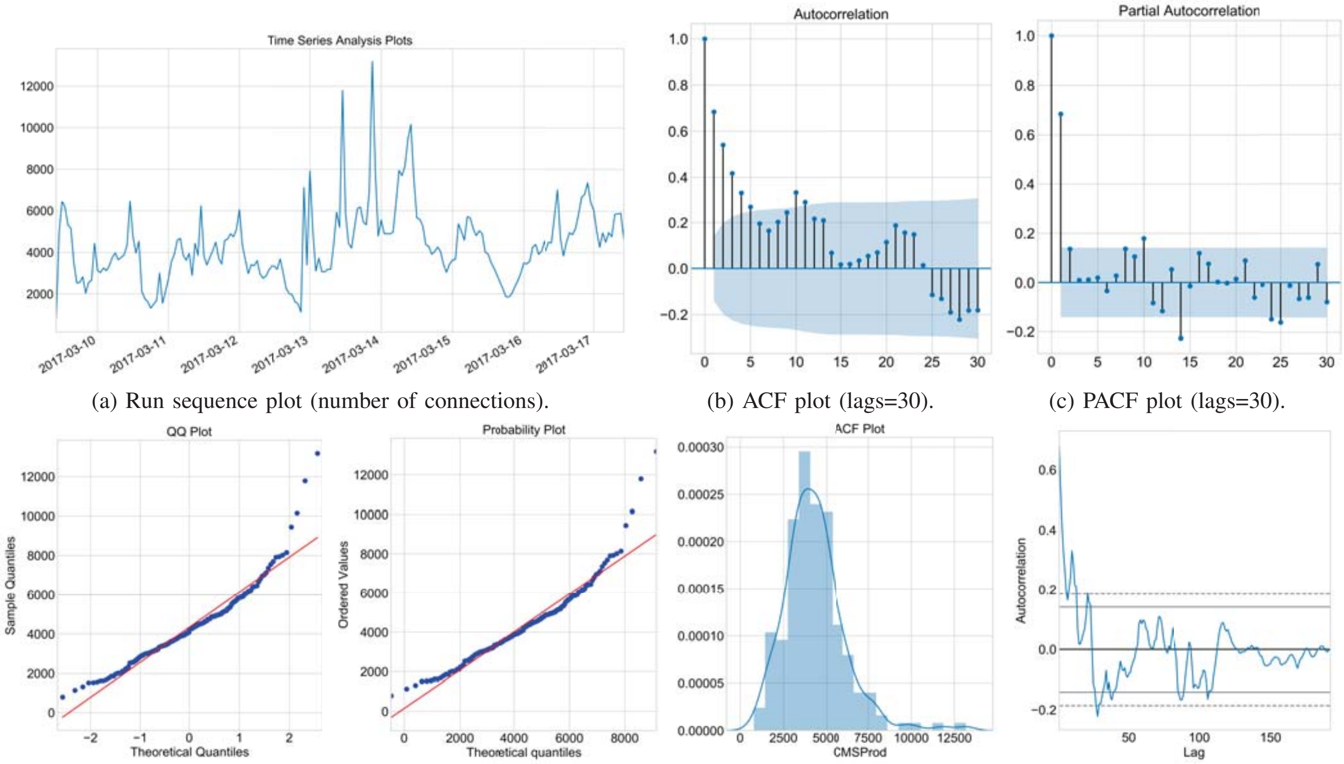
A number or recent studies have focused on the use of statistical thresholds, autoscaling and linear models for resource allocation and provisioning. The authors in [6] look at an example of Neflix using Amazon Web Services (AWS) and propose an autoscaling system for dynamic minimum bandwidth reservation from multiple data centers. Due to the limitations of the use of static thresholds in handling outlier events, more rigorous mechanisms are required. The authors in [7] explore a regression model for autoscaling. Other works such as [8], [9], [10] explore linear models for autoscaling. As non-linear techniques such as exponentially weighted moving average (EWMA) are too expensive computationally, predictive analytic techniques using ARIMA for forecasting cloud resources have been proposed in [11], [12]. Unlike the works described above, our work focuses on developing a predictive analytics solution for managing resources in an application-aware SDN architecture for large-scale data transfers.

III. Exploratory Data Analysis

We use application-awareness to obtain reliable and accurate GridFTP data transfer and connection information. This includes transfer/connection information from both CMS and LIGO workflows. The CMS computing model [3] defines various roles and services for data access. The GridFTP connection information obtained using application-awareness is classified based on both the user-role and its corresponding workflow. Our initial dataset contains connection information classified based on a total of four CMS user-roles and a single LIGO user-role. The dataset includes GridFTP data transfer and connection information measured at a single U.S. CMS Tier-2 site over a period of 15 months. Of the four CMS user-roles, we only use the transfer information from a single-role for testing in our predictive analytics framework. This test dataset (henceforth referred to as the "dataset") contains connection transfers associated with the CMS production workflows representing project-level information of a specific particle-physics project. However, we note that other user/workflow data can be used to similar effect with our proposed predictive analytics framework.

We employ time series analysis on our dataset to understand the nature of its phenomenon and for forecasting future values of the observed sequence. The future/predicted values are also referred to as "out-of-sample" forecasts. We begin by analyzing the dataset for the presence of components such as trends, seasonality, cyclic patterns and random irregular patterns. We assume that the GridFTP transfer information dataset exhibits a systematic pattern (i.e. trend, seasonal and cyclical information) combined with random noise, making the pattern identification difficult. Therefore, our objective is to identify the presence of non-stationary processes in the time series through statistical testing. The removal of systematic patterns from the dataset results in a residual component referred to as "random shock" or "error". We assume that the residual component is independent identically distributed (i.i.d.) and can be modeled using linear regression on exogenous variables for forecasting future data transfers.

Ensuring stationarity of the transfer dataset is an important step in our exploratory data analysis. The sample statistics of a stationary time series do not undergo systemic shifts over



(a) Run sequence plot (number of connections). (b) ACF plot (lags=30). (c) PACF plot (lags=30).

(d) QQ plot. (e) Probability plot. (f) Kernel density plot. (g) Autocorrelation (lags=30).

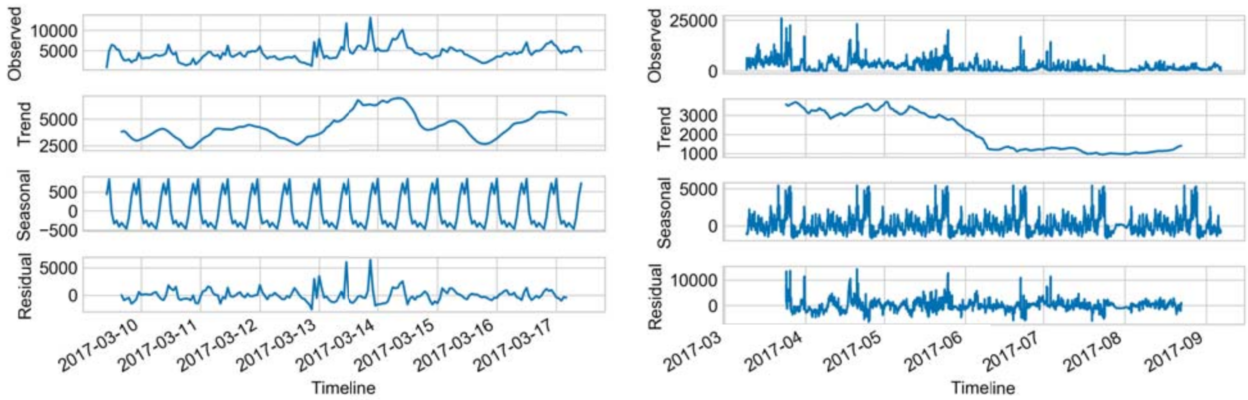
Fig. 1: Exploratory data analysis of the GridFTP transfer dataset.

the observation period. Our statistical prediction models (see Section IV) rely on the stationarity of the time series to make reasonable forecasts of the future data transfers. We use a number of tools to identify systematic patterns in our dataset. The non-repeating linear/non-linear components, i.e. trends, are analyzed using data smoothing. We analyze the periodic variations, i.e. seasonals, using run sequence plots.

To verify the stationarity of our dataset, we use the Augmented Dickey-Fuller (ADF) test. The ADF test also helps us determine whether differencing is required to remove the trends and seasonals from the transfer dataset. The test works by checking a univariate process for a unit root when serial correlation is present. The null hypothesis of the test represents the presence of a unit root and vice versa with the alternative hypothesis. A *p-value* > 0.05 accepts the null hypothesis and indicates a non-stationary series. The *p-values* are obtained using the method described in [13]. The ADF test is configured to automatically choose the number of lags to maximize the Akaike Information Criterion (AIC) [14].

Figure 1 presents the exploratory data analysis of our GridFTP transfer dataset. Figure 1a shows the run-sequence plot of the measured total number of data transfer connections over time. The dataset is presented with an aggregated sampling interval of one hour. We present the correlograms representing the autocorrelation function (ACF) and partial autocorrelation function (PACF) in the Figures 1b and 1c respectively. The shaded bands represent a 95% confidence interval (CI), with lags (i.e. data points) outside of this area showing statistically significant correlation. The lags inside the confidence intervals are due to random-shock. From the correlograms we observe serial dependencies in the transfer dataset. Differencing can be employed to remove these serial dependencies. For example, we can transform each element i in a sample to $(i - k)$, where k is lag value. Thus, through differencing, we can not only identify the hidden seasonal dependencies in the series, but also ensure that the autocorrelations for consecutive lags are independent of each other. This has the effect of making the seasonal components more apparent. Further, by removing the seasonal dependencies, we can make the time series stationary. This is a necessary requirement for the forecasting technique used in Section IV.

Finally, we also note that the analysis presented in Figure 1 represents a non-stationary series as evidenced by subjecting the dataset to the ADF test. The ADF test on the time series in Figure 1a results in *p-values* of 0.116 and 2.628×10^{-4} pre- and post-differencing (first-order) respectively. Thus, we can conclude that first-order differencing can stationarize the dataset. Figures 1d and 1e present the normal quantile-quantile (QQ) plot and the normal probability plot for our dataset respectively. From the graphs, we observe that the dataset is right skewed in comparison to a normal distribution. The corresponding kernel density plot is also shown in Figure 1f. Lastly, in Figure 1g, we present the autocorrelation plot with confidence intervals of both 95% and 99% respectively. The



(a) Decomposition plot of the short-term aggregate.

(b) Decomposition plot of the long-term aggregate.

Fig. 2: Short-term and long-term time series decomposition plots.

number of lags outside the 95% CI are indicative of the AR model order p. The time series decomposition plots for a short-term aggregate (one hour aggregates over a period 8 days) and a long-term aggregate (one hour aggregates over a period of 6 months) are shown in Figures 2a and 2b respectively. The STL [15] method is used in this decomposition. Next, we provide some guidance on data aggregation, dataset length and forecast objectives, and its impact on design choices.

A. Dataset considerations for model estimation

1) Effect of Data Aggregation: The GridFTP transfer information is measured with a granularity of microseconds. We perform data aggregation and use the aggregate statistics to make predictions as the original dataset granularity does not permit meaningful analysis. We use resampling to aggregate data transfers with a granularity of one hour and use the obtained aggregate statistics in our predictive analytics framework. The choice of the resampling period and the corresponding aggregate granularity is subjective and depends on the application's forecasting objectives.

2) Dataset length: Another important design choice is the length of the dataset used to fit the forecasting model. The number of data points in the sample depends on the amount of random fluctuations in the data and also on the number of model parameters [16]. We rely on the model's performance with out-of-sample data and use both AIC and the out-of-sample mean absolute percentage error (MAPE) to choose the dataset length.

3) Other considerations: Lastly, whether predictive analytics and forecasting are applied to per-aggregate (total transfers), per-user, per-flow or per-project depends on how the forecast information is used to influence network management decisions. We implement a per-role forecast to predict the user-roles' resource requirements.

IV. An ARIMA model for Predictive Analytics

In this section, we develop a predictive analytics framework to accurately forecast GridFTP transfer connections. First, we train the model using the stationarized dataset obtained from Section III to estimate its parameters. Initially, we use a historical dataset partitioned into a training set and a validation set to estimate the model parameters. On finding a suitable model, we verify its performance on the validation set using mean absolute percentage error (MAPE) as the metric. The model is then applied to make rolling forecasts with a real-time dataset consisting of the current observation (x_t) and k time-lagged observations (i.e. $\{x_{t-k}, x_{t-k+1}, \cdots, x_{t-1}, x_t\}$). The model generates n out-of-sample forecasts (see Section IV-B) that are used by the SDN controller to make resource allocation decisions. Finally, the SDN controller effects data-plane changes over the OpenFlow protocol to preemptively adapt to the changing resource requirements. Automatic model checking and re-selection are employed to ensure that the forecast accuracy is consistently maintained.

The autoregressive integrated moving average (ARIMA) [17] model includes two common processes: an autoregressive process (AR) and a moving average (MA) process. The AR and MA processes are combined with an "integrated" component. The integrated component is used to replace data values with their corresponding *d-th* order differenced values. The differencing process ensures stationarity of the dataset and the integration process reverses this effect post-fitting. Due to the presence of serially dependent data points in our dataset, we can use the time-lagged information to estimate future values using the AR process. Unlike the AR process, in the moving average (MA) process, each observation is also affected by past random shock. Thus, each observation is a linear combination of past random shocks. The ARIMA method is a generalization of the ARMA process and the model is composed of three parameters namely: (i) the AR parameter (p), (ii) the number of differencing passes (d), and (iii) the MA parameter (q).

A. Design Framework

The design framework is shown in Figure 3. The application-aware SDN architectures similar to those in [4], [18] are used in the data acquisition process to obtain GridFTP data transfer and connection information. The SNAG [4]

(SDN-managed network architecture for GridFTP transfers) architecture provides a convenient mechanism for obtaining accurate pre-classified information about all current/ongoing flows in the campus network. GridFTP connection information from a pool of 13 production GridFTP servers are obtained using this approach. This GridFTP connection information is indexed by a 11 node Elastic cluster for storage, indexing, analysis and visualization. The Elastic stack provides representational state transfer (REST) application programming interfaces (APIs) that are used by the predictive model to request the total aggregate connection transfers for the time window under consideration.

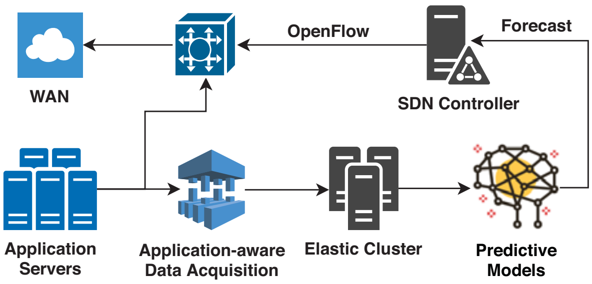


Fig. 3: Predictive Analytics Framework.

The dataset is used to fit an ARIMA-based predictive analytics model. The out-of-sample forecasts from the model and its parameters are communicated to the SDN controller. The SDN controller uses the forecast information to track and adapt the data-plane resources allocated to the corresponding user-role. To ensure consistent performance, the framework incorporates periodic model checking and automatic parameter (re-)estimation (see Section IV-B).

B. Algorithm Design

The ARIMA-Forecast$(\boldsymbol{x}, n)$ is shown in Algorithm 1. The algorithm is divided into two phases namely: (i) model parameter estimation, and (ii) model checking and parameter re-estimation. The *FindBestModel*$(\boldsymbol{x}, n)$ routine is used to identify the best ARIMA model parameters for a dataset $\boldsymbol{x}$. The search space of the model order is upper bounded by $[P, D, Q]$, with $P, Q = \{i, 0 < i < 10\}$ and $D \in (0, 1)$. For each run, we compute the model's Akaike Information Criterion (AIC) and choose the model with the least AIC. The best model (i.e. mdl_{out}) and its order (O_{meta}) are stored and used to forecast n out-of-sample steps. Both the forecast information and the model parameters are used by the SDN controller for network management decisions as outlined in Section IV-A. The first phase is responsible for both model identification and model parameter estimation. The model is then used to forecast out-of-sample steps (i.e. predictions). Thus, the algorithm finds the best effective (yet computationally frugal) model order O_{meta} using maximum likelihood estimation to minimize the sum of squared residuals.

We perform model checking in the second phase. We use a *rolling forecast* model to adapt and track changes in traffic conditions. The model identified in the first phase is used to make out-of-sample forecasts. However, to ensure consistency in the model's performance, we periodically check the accuracy of the out-of-sample forecasts with real observations. Automatic parameter re-estimation is performed if the model's mean absolute error (MAE), ε, falls below a predefined threshold, τ. This process is scheduled periodically to ensure that the out-of-sample predictions are accurate and usable by the SDN controller for decision making.

Algorithm 1 ARIMA-Forecast$(\mathbf{x}, n)$

Require: Connection information vector $(\mathbf{x})$, out-of-sample forecast step size (n).
Output: F_n out-of-sample forecasts.

$AIC_{out} = \infty$
$O_{meta} = None$
$mdl_{out} = None$
Phase 1: Model parameter estimation
FindBestModel$(\mathbf{x}, n)$:
for p in 0 **to** P **do**
  for d in 0 **to** D **do**
    for q in 0 **to** Q **do**
      $mdl_{tmp} =$ ARIMA$(\mathbf{x}, n, [p, d, q])$
      if AIC$(mdl_{tmp}) < AIC_{out}$ **then**
        $AIC_{out} =$ AIC (mdl_{tmp})
        $O_{meta} = [p, d, q]$
        $mdl_{out} = mdl_{tmp}$
      end if
    end for
  end for
end for
return $F_n = FORECAST\,(mdl_{out}, n)$
Phase 2: Model Checking
while True **do**
  $t = t_{now}$
  if $\varepsilon\,(mdl_{out}) < \tau$ **then**
    FindBestModel$(\mathbf{x}, n)$
  end if
  Wait $t + \delta t$
end while

V. Experimental Study

An example forecast with 95% and 99% confidence bands are shown in Figure 4. The out-of-sample forecasts are generated using an *ARIMA*$(5, 1, 5)$ model estimated using Algorithm 1. The estimated model has a mean absolute percentage error (MAPE) of 6.68% for ten out-of-sample forecasts. The MAPE varies between 6% to 10% for different dataset inputs over time. Further, the MAPE increases monotonically with an increase in the number of out-of-sample forecasts.

The estimated model's residuals are shown in Figure 5. The ACF and PACF plots are shown in Figure 5a. We observe that the correlograms confirm the absence of significant correlation, since the data points fall within the CI bands. The kernel density plot showing the distribution of the residuals is shown in Figure 5b. We note that model residuals do not show normality and their distribution is heavy-tailed. This implies

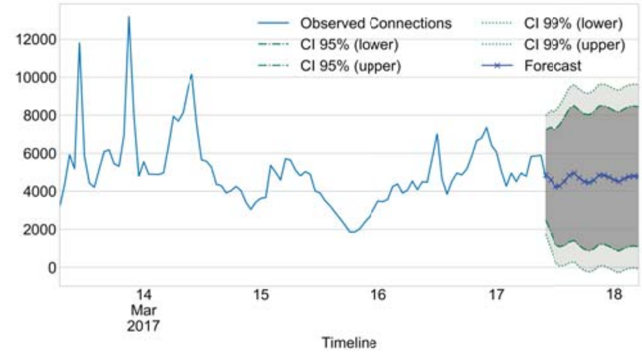
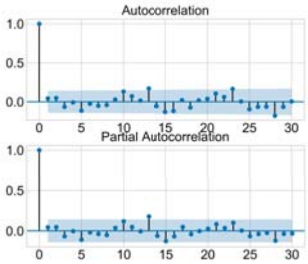
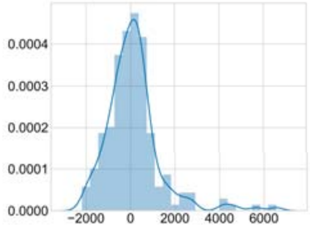


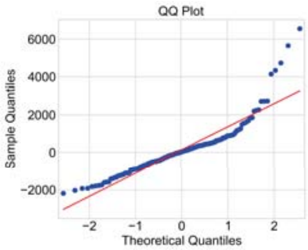
Fig. 4: GridFTP transfer forecast (Duration: 10 minutes).



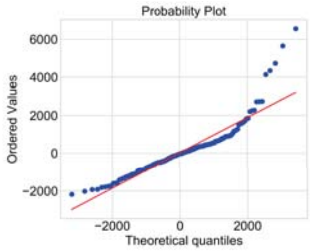
(a) Residual correlograms.



(b) Residual distribution.



(c) Residual QQ plot.



(d) Residual probability plot.

Fig. 5: Forecasting and predictive

that our model does not fully capture the underlying process, but instead show some correlated errors. Lastly, Figures 5c and 5d show the QQ plot and the probability plot of the residuals respectively. These plots also confirm that the residuals cannot be represented using a white-noise process. Thus, the residuals follow a heavy-tailed distribution. This is due to the presence of occasional conditional volatility that is not captured by our model.

VI. Conclusions and Future Work

We proposed a predictive analytics framework for forecasting GridFTP transfer connections using an application-aware SDN approach. We developed an autoregressive integrated moving average (ARIMA) based forecasting algorithm that incorporates automatic parameter estimation and periodic model checking features. When used to make rolling forecasts, our framework with a MAPE performance ranging between 6% to 10%, can provide effective strategies to preemptively drive resource allocation decisions. Thus, our predictive analytics framework using application-aware SDN can lead to the development of proactive network management systems. By tightly integrating our predictive analytics framework with an application-aware SDN solution, we demonstrate a network management approach capable of forecasting the resource requirement of dynamic large-scale data transfers. Our future work will focus on developing multivariate predictive analytic models to characterize heterogeneous user/application traffic behavior.

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant Numbers OAC-1541442 and CNS-1817105. This work was completed using the Holland Computing Center of the University of Nebraska, which receives support from the Nebraska Research Initiative. The authors would like to thank Garhan Attebury, Holland Computing Center at UNL for his valuable support.

References

[1] E. Hernandez-Valencia, S. Izzo, and B. Polonsky, "How will NFV/SDN transform service provider opex?" *IEEE Network*, vol. 29, no. 3, pp. 60–67, May 2015.

[2] B. P. Abbott, R. Abbott, R. Adhikari *et al.*, "LIGO: the Laser Interferometer Gravitational-Wave Observatory," *Reports on Progress in Physics*, vol. 72, no. 7, p. 076901, 2009.

[3] D. Bonacorsi, "The CMS Computing Model," *Nuclear Physics B - Proceedings Supplements*, vol. 172, pp. 53 – 56, 2007.

[4] D. N. Anantha, Z. Zhang, B. Ramamurthy *et al.*, "SNAG: SDN-managed Network Architecture for GridFTP Transfers," in *3rd Workshop on Innovating the Network for Data-Intensive Science (INDIS) '16, SC16*, November 2016.

[5] W. Allcock, J. Bresnahan, R. Kettimuthu, and M. Link, "The Globus Striped GridFTP Framework and Server," in *Supercomputing, 2005. Proc. ACM/IEEE SC 2005 Conference*, Nov 2005, pp. 54–54.

[6] D. Niu, H. Xu, B. Li, and S. Zhao, "Quality-assured cloud bandwidth auto-scaling for video-on-demand applications," in *2012 Proceedings IEEE INFOCOM*, March 2012, pp. 460–468.

[7] H. Fernandez, G. Pierre, and T. Kielmann, "Autoscaling web applications in heterogeneous cloud infrastructures," in *2014 IEEE International Conference on Cloud Engineering*, March 2014, pp. 195–204.

[8] N. Roy, A. Dubey, and A. Gokhale, "Efficient autoscaling in the cloud using predictive models for workload forecasting," in *2011 IEEE 4th International Conference on Cloud Computing*, July 2011, pp. 500–507.

[9] B. Bouterse and H. Perros, "Scheduling cloud capacity for Time-Varying customer demand," in *2012 IEEE 1st International Conference on Cloud Networking (CLOUDNET)*, Nov 2012, pp. 137–142.

[10] K. Kanagala and K. C. Sekaran, "An approach for dynamic scaling of resources in enterprise cloud," in *2013 IEEE 5th Intl. Conf. on Cloud Computing Technology and Science*, vol. 2, Dec 2013, pp. 345–348.

[11] A. Amin, A. Colman, and L. Grunske, "An approach to forecasting qos attributes of web services based on arima and garch models," in *2012 IEEE 19th Intl. Conf. on Web Services*, June 2012, pp. 74–81.

[12] A. Adegboyega, "An adaptive score model for effective bandwidth prediction and provisioning in the cloud network," in *2015 IEEE Globecom Workshops (GC Wkshps)*, Dec 2015, pp. 1–7.

[13] J. G. MacKinnon, "Critical Values for Cointegration Tests," Queen's University, Dept. of Economics, Working Papers 1227, Jan. 2010.

[14] H. Akaike, *Information Theory and an Extension of the Maximum Likelihood Principle*. Springer New York, 1998, pp. 199–213.

[15] R. B. Cleveland, W. S. Cleveland, and I. Terpenning, "STL: A seasonal-trend decomposition procedure based on loess," *Journal of Official Statistics*, vol. 6, no. 1, p. 3, 1990.

[16] R. Hyndman and A. V. Kostenko, "Minimum Sample Size requirements for Seasonal Forecasting Models," *Foresight: The International Journal of Applied Forecasting*, no. 6, pp. 12–15, 2007.

[17] G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.

[18] D. N. Anantha, B. Ramamurthy, B. Bockelman, and D. Swanson, "Differentiated network services for data-intensive science using application-aware SDN," in *2017 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, Dec 2017, pp. 1–6.