An Approach to Controller Load Forecasting in Software-Defined Networks

Lomonosov Moscow State University

Faculty of Computational Mathematics and Cybernetics

Yulia Skobtsova
Lomonosov Moscow State University
Moscow, Russia
xenerizes@lvk.cs.msu.su

Vasily Pashkov
Lomonosov Moscow State University
Moscow, Russia
pashkov@lvk.cs.msu.su

Introduction

SDN controller undergoes a load from the switches, which includes the processing of incoming control messages and active flow support.

An **overload** means the situation when amount of data delivered to controller is larger than controller capacity – an amount of data that can be processed in time.

Controller overload causes:

- increase in incoming messages processing time;
- disruption of services for end users;
- loss of network control.

Controller load real-time forecasting allows to determine the possible overload in advance and win some time to prevent it.

Controller load forecasting problem

Let h be the time discretization step. Controller interaction with switches $\{s_1, \dots, s_i, \dots, s_k\}$ at time point t can be described by:

1. $p_1^i(t)$ – incoming OpenFlow-messages count from s_i (received in $(t-h;\ t]$);
2. $p_2^i(t)$ – outgoing OpenFlow-messages count to s_i (transmitted in $(t-h;\ t]$).

At time point t **controller state** p is:

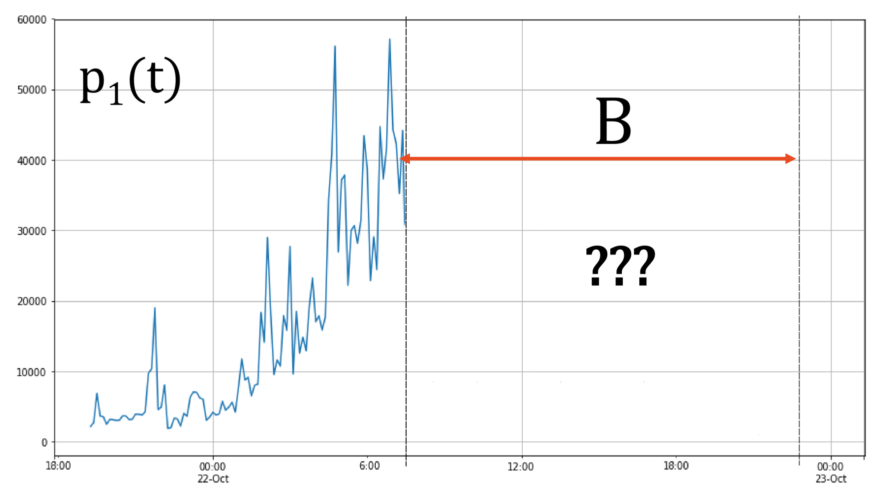
- two time series $\{p_1^i(t), p_2^i(t)\}$ with W observations for each switch s_i.

Given:

- controller state at time point t;
- controller load function

$L(t) = \{\sum p_1^i(t), \sum p_2^i(t)\}$;

- operation time B.



Find:

- estimation $\hat{L}(t)$ of controller load function $L(t)$ for time points $\{t+h, \dots, t+Bh\}$, where $\frac{1}{n}\sum_{i=1}^{n}\left(L(t)-\hat{L}(t)\right)^2 \to min$.

Real-time load forecasting method

Method operates step by step. On each step a forecast with $F, F \ll B$ points is created. The data is approximated by **ARIMA** [2] forecasting model. Actions performed at the *step N* are:

- previous forecast quality check (quality is calculated by correlation distance, RMSE and MAE);
- model order selection (if recommended by quality check);
- model coefficients selection to satisfy updated history;
- forecast with F points creation;
- forecast correction (the latest $\mathrm{M}, \mathrm{M} < F$ points values replacement).

Experiments

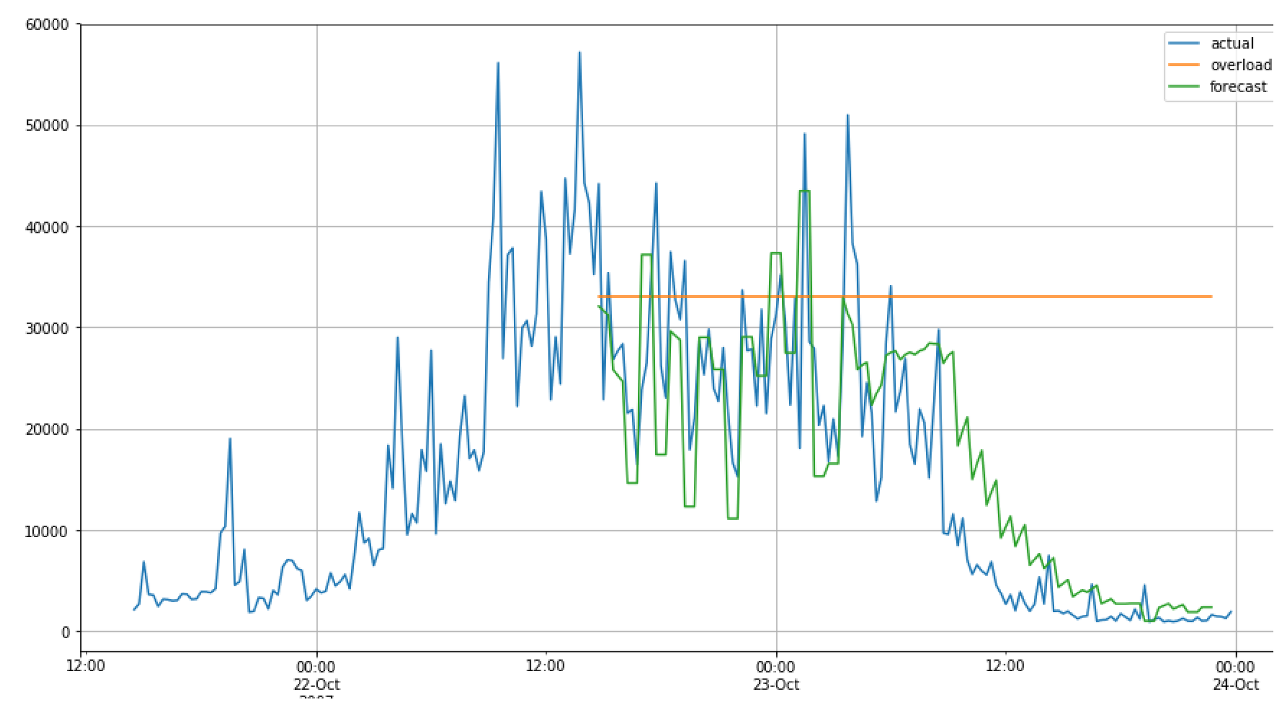
Proposed method is implemented in Python using Pandas [3] StatsModels [4] libraries. Experimental data was collected on Runos SDN controller [5].

Load forecast quality was estimated:

- by ability to determine overloads according to forecast;
- by forecast errors MAE, RMSE.

Controller capacity was selected as follows:

$\alpha \cdot (95th\ percentile\ - 5th\ percentile)$.



Controller load forecast quality was evaluated experimentally for different forecast lengths, monitoring interval sizes, controller capacity and other parameters of proposed method.

Results

Method quality

(FPR – false positive rate, TPR – true positive rate)

α	F = 3		F = 5		F = 7	
	max FPR, %	*mean TPR, %*	*max FPR, %*	*mean TPR, %*	*max FPR, %*	*mean TPR, %*
0.8	5,7	92,3	2,1	92,3	3,3	92,3
0.85	8,07	99,0	7,4	98,9	4,2	97,0
0.95	6,4	94,8	3,5	89,7	16,4	71,4

Forecast errors

	RMSE, packets	MAE, packets	Error rate (MAE, relative to median), %
F = 3	386	497	1,57%
F = 5	370	497	1,16%
F = 7	513	524	1,62%

Conclusions

The quality of controller overload detection showed by evaluation allows to use proposed controller load forecasting method in real systems. An approach is also lightweight, as it does not require complex equipment for implementation and can be used alongside deployed SDN controller.

The further research direction is a study of decision-making systems for distributed control platforms, which can select the controller behavior according to forecasted state data.

Bibliography

[1] Open Networking Foundation, "Specification O. F. S. Version 1.3.0 (wire protocol 0x04)," vol 1, 2012.

[2] Hyndman R.J., Athanasopoulos G., Forecasting: principles and practice. OTexts, 2014.

[3] McKinney W., "pandas: a Python data analysis library," see http://pandas. pydata. org/.

[4] Seabold, Skipper, and Josef Perktold, "Statsmodels: Econometric and statistical modeling with python." Proceedings of the 9th Python in Science Conference, 2010.

[5] A. Shalimov, S. Nizovtsev, D. Morkovnik, R. Smeliansky, "The Runos OpenFlow Controller," in Software Defined Networks (EWSDN), 2015 Fourth European Workshop on (pp. 103-104), IEEE, September 2015.