Trunked Radio Systems: Traffic Prediction Based on User Clusters

Hao Chen
School of Computing Science
Simon Fraser University
Burnaby, BC V5A 1S6
Canada
lcheu@cs.sfu.ca

Ljiljana Trajković
School of Engineering Science
Simon Fraser University
Burnaby, BC V5A 1S6
Canada
ljilja@cs.sfu.ca

Abstract— **Studies of individual network users' behavior patterns may seem of little relevance to predicting the entire network's traffic. Clustering techniques, however, help bridge this apparent gap. In this paper, we analyze data collected from a deployed network and use clustering techniques to characterize patterns of individual users' behavior. A network traffic prediction approach is then developed based on user clusters. We analyze three months of continuous network log data from the E-Comm network, an operational trunked radio system. After extracting traffic data from the raw data logs, we identify user clusters by employing the AutoClass tool and the K-means algorithm. Based on the identified user clusters, we use the Seasonal Autoregressive Integrated Moving Average (SARIMA) model to forecast the network traffic by aggregating the predicted traffic of each user cluster. The predicted network traffic shows good agreement with the collected traffic data.**

I. Introduction

The analysis of traffic from operational wireless networks provides useful information about the user behavior patterns. The analysis enables network operators to better understand the behavior of network users and ultimately provide better quality of service.

Traffic prediction is important to assess future network capacity requirements and to plan future network developments. Traditional prediction of network traffic usually considers aggregation of traffic from individual network users and assumes a constant number of network users. This approach cannot easily adapt to dynamic network environments where the number of users varies. One alternative approach, which focuses on individual users, is impractical in predicting the aggregate network traffic because of the high computation cost in cases where the network consists of thousands of users. Employing clustering techniques for predicting aggregate network traffic bridges the gap between these two approaches.

Prior analysis of local-area and metropolitan-area wireless network traffic data analyzed user behavior and mobility patterns [1] and discovered diurnal and weekly behavior patterns. A study of billing records from a Cellular Digital Packet Data (CDPD) mobile wireless network discovered similar cyclic user behavior [2]. An analysis of a trunked radio network traffic revealed that the call holding time distributions are approximately lognormal while the call inter-arrival times are closely approximated by an exponential distribution [3].

In this paper, we describe the analysis of traffic data collected from E-Comm network [4], an operational trunked radio system serving as a regional emergency communication system. We analyzed the raw network log data collected over 92 days: from March 1^{st}, 2003 to May 31^{st}, 2003. In order to achieve better clustering and prediction performance, we removed irrelevant data fields and redundant records. By combining multiple records, we extracted the traffic data and created new records. The footprint of network usage for talk groups was represented by their hourly number of calls. We then used the AutoClass tool [5] and the K-means [6], [7] algorithm to classify network talk groups into clusters. After clustering network users into various clusters characterized by distinct user behavior patterns, we used the Seasonal Autoregressive Integrated Moving Average (SARIMA) model [8] to predict the behavior of each user cluster. We then used aggregation to predict the overall network behavior. The proposed prediction approach led to better prediction results than prediction based on the aggregate network traffic.

In Section II, we present an overview of the E-Comm network architecture. Data cleaning and extraction processes are examined in Section III. We discuss the clustering algorithm and the results of user clustering in Section IV. The SARIMA model used for traffic prediction and the results of the user clusters based traffic prediction are given in Section V. We conclude with Section VI.

II. E-Comm network

A. *E-Comm network architecture*

E-Comm is the regional emergency communications center for Southwest British Columbia. It provides emergency dispatch/communication services for a number of police, ambulance, and fire departments in the Greater Vancouver Regional District, the Sunshine Coast Regional District, and the Whistler/Pemberton area. E-Comm serves agencies such as Royal Canadian Mounted Police, fire and rescue, local police departments, ambulance, and industrial customers such as BC Translink. In total, sixteen agencies use the E-Comm network and each agency has a number of affiliated talk groups.

The E-Comm network employs Enhanced Digital Access Communications System (EDACS), developed by M/A-COM (formerly Comnet-Ericsson) in 1988. EDACS is a group-oriented communication system that allows groups of users to communicate irrespective of their physical locations.

Currently, the E-Comm network consists of 11 cells. Each cell covers one or more municipalities, such as Vancouver, Richmond, and Burnaby. The basic talking unit in the E-Comm network is talk group: a group of individual users working together on common tasks. The E-Comm network is capable of both voice and data transmissions. We analyze only voice traffic because it accounts for more than 99% of network traffic.

B. Group and multi-system calls

A *group call* is a standard call made in a trunked radio system. Groups are sets of users who need to communicate regularly. For example, within a single city-wide system, the North and South fire services may each have one talk group, while the police may be subdivided into several talk groups. EDACS network operators have observed that more than 85% of calls are group calls [9].

A *multi-System Call* is a single group call involving more than one system (cell). In EDACS terminology, *system* is a synonym for cell. A user may initiate a group call without knowing the physical locations of the group members. When all members of the talk group are within one system, the group call is a single system call occupying only one traffic channel in the system. However, when group members are distributed over multiple systems, their group call becomes a multi-system call that occupies one traffic channel in each system. Hence, the major difference between a single-system call and a multi-system call is that the latter occupies additional channels and consumes more system resources. From the collected data, we observed that more than 55% of group calls are multi-system call.

III. Traffic Data

A. Data collection and data format

The raw E-Comm traffic data emanates from a distributed event log database that records every event occurring in the network, such as call establishment, channel assignment, call drop, and emergency call. The collected data contains continuous data records from March 1^{st} 2003 00:00:00 AM to May 31^{st} 2003 00:00:00 AM. The size of the original database is $\sim$ 6 GBytes, with 44,786,489 record rows for the 92 days of data. From the 26 original fields in the database, the 9 fields listed in Table I are of particular interest for our analysis.

TABLE I
SELECTED TRAFFIC DATA FIELDS

Name	Content
event_utc_at	time-stamp of the event recorded
duration$ms	duration of the event in ms
system_id	identification of system involved
channel_id	radio channel occupied
caller	caller's id in the event
callee	callee's id in the event
call_type	call type info., such as group call, emergency call
call_state	call status, such as call assign and call drop
multi_system_call	flag indicating if the event is a multi-system call

B. Data cleaning and extraction

Data preprocessing is a mandatory step for data analysis. It entails cleaning the database, filtering the outliers, and removing redundant records. For example, records with call_type = 100 or with duration = 0 are filtered because they are incorrectly recorded in the database. Records with call_state = 1, which implies a "call drop" event, are redundant because each "call drop" event has a corresponding "call assign" event in the database, while the reverse is not the case. Records with channel_id = 0 are removed because they represent the control channel and those events are not related to users' behavior. Approximately 55% of records were removed from the database in the data cleaning phase.

In the original event log, a multi-system call involving several systems creates several database entries. Each entry represents a call in one system. For example, if a user in system 1 calls three group members currently located in systems 1, 2, and 3, three database entries will be created, with system IDs 1, 2, and 3, respectively. This recording mechanism is appropriate for logging the event activity in a distributed environment. However, when capturing users' calling behavior, a multi-system call with multiple entries in the database may be counted as multiple calls. Hence, we developed an algorithm for the extraction of accurate user calling activity by comparing the call_state, call_type, caller, callee, and call_duration fields in multiple records.

C. Talk group data

The basic talking unit in the E-Comm network is the talk group. Its behavior patterns represent users' behavior. Although talk groups are organized within an agency, the organizational structure does not necessarily represent the usage pattern. Talk groups in different agencies may have similar behavior, while talk groups in the same agency may have different behavior pattern. Clustering of talk groups irrespective of their agencies may help discover various user patterns beneath the apparent agency structure.

An important calling behavior pattern in the voice network is the number of user calls. A common metric employed in telecommunication industry is the hourly number of calls. It may be regarded as a footprint of users' calling behavior. Time scales finer than an hour (minute) are too small to record the calling activity since a call usually lasts 3 – 5 minutes. Time scales larger than an hour (day) are too coarse to capture users' behavior patterns. The collected 92 days of traffic data (2,208 hours) permitted each talk group's calling behavior to be captured by the 2,208 hourly calls. The hourly number of calls for two talk groups over 168 hours is shown in Fig. 1. The distinct calling patterns of the two talk groups are the data samples used for classification.

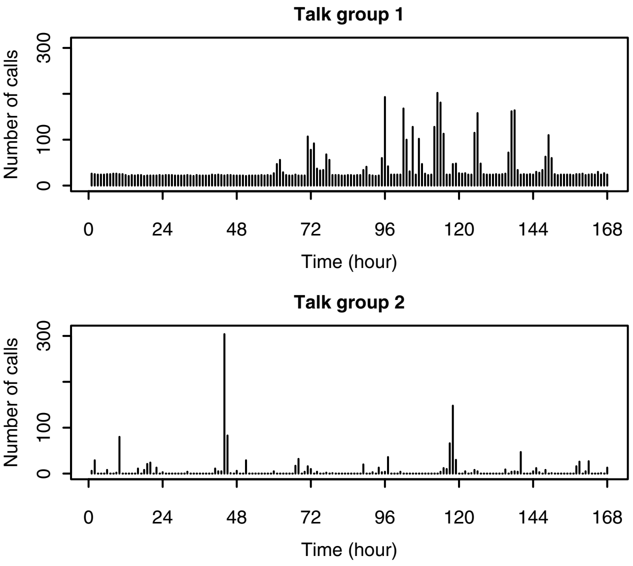

Fig. 1. Calling patterns for talk groups 1 and 2 over a 168-hour period.

IV. User Clustering

A. Clustering algorithms

Clustering analysis has a variety of objectives. It groups or segments a collection of objects into subsets or "clusters". Objects within a cluster are more similar to each other than objects in distinct clusters. An object can be described by a set of measurements or by its relations to other objects. Clustering analysis may, for example, be used to discover distinct customer groups and to characterize customer groups based on their purchasing patterns in business environments. In our study, we aimed to identify distinct user groups with similar network usage patterns. Network users are classified into clusters, according to the similarity of their behavior patterns. We used AutoClass [5] and the K-means [6] algorithm to classify the calling patterns of talk groups.

AutoClass [5] is an unsupervised classification tool based on the classical finite mixture model. The goal of the Bayesian approach to unsupervised classification is to find the most probable set of class descriptions given the data and the prior expectations. AutoClass begins by creating a random classification and then manipulates it into a high probability classification through local changes. It repeats the process until it converges to a *local maximum*. It then starts over again and continues for a specified number of tries. The computed probability is intended to cover the whole volume in the parameter space around this maximum, rather than just the peak. Each new try begins with a certain number of classes and may conclude with a smaller number of classes.

The K-means algorithm [6] is a commonly used algorithm for data clustering. Based on the input parameter k, it partitions a set of n objects into k clusters so that the resulting intra-cluster similarity is high and the inter-cluster similarity is low. Similarity of clusters is measured with respect to the mean value of the objects in a cluster, which can be viewed as the cluster's center of gravity. The algorithm is well-known for its simplicity and efficiency.

B. Clustering results

After 1,100 tries over 18 hours, AutoClass discovered 24 populated clusters of talk groups as the best classification, with the average cluster size of 25.7. The number of talk groups in each cluster (cluster size) is shown in Table II. The hourly number of calls for the talk groups in clusters 5, 17, and 22 are shown in Fig. 2. Talk groups in the three clusters exhibit distinct calling behavior patterns.

TABLE II
AutoClass Results: Cluster Size

Cluster ID	Size	Cluster ID	Size	Cluster ID	Size
1	144	9	20	17	15
2	67	10	20	18	13
3	66	11	19	19	12
4	31	12	19	20	10
5	25	13	18	21	9
6	23	14	18	22	4
7	22	15	18	23	3
8	21	16	17	24	3

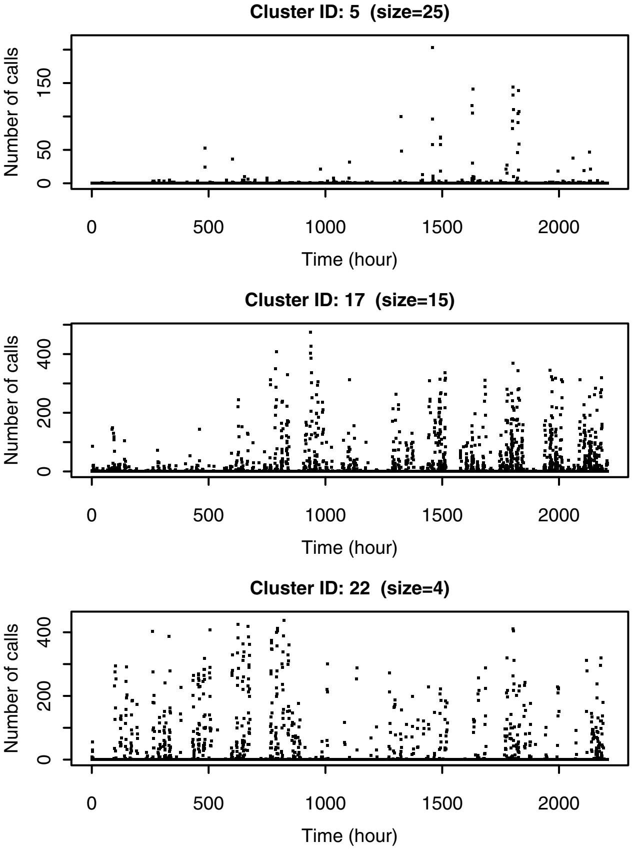

Fig. 2. Cluster data plot for AutoClass clusters with IDs 5, 17, and 22.

The K-means algorithm requires an *a priori* number of clusters. We tested the performance of the K-means algorithm for three choices: k = 3, 6, and 16. The Euclidean distance was used as the distance function in the K-means algorithm to measure the similarity among talk groups. The clustering results with three clusters (k = 3) proved most useful.

Table III shows the properties of the three clusters discovered by the K-means algorithm. The first cluster consists of 17 talk groups, contributing to 59% of the calls in the network, with an average number of calls ranging from 94 to 208 per hour. Talk groups in this cluster are the busiest in the network in terms of the number of calls. They are the dispatch groups that assign and schedule other talk groups for specific tasks. The second cluster with 31 talk groups, contributing to 26% of the calls, is composed of average network users. The third cluster consists of the least frequent users who contribute to only 15% of the network calls. These users account for over 90% of talk groups.

TABLE III
K-MEANS RESULT: CLUSTER PROPERTIES (k = 3 CLUSTERS)

Cluster size	Range of min. num. of calls	Range of max. num. of calls	Range of avg. num. of calls	Total number of calls	Number of calls (%)
17	0 - 6	352 - 700	94 - 208	5,091,695	59
31	0 - 3	135 - 641	17 - 66	2,261,055	26
569	0	1 - 1613	0 - 16	1,310,836	15

V. TRAFFIC PREDICTION

The aggregate network traffic was predicted using the classical SARIMA model [8]. We also employed cluster-based prediction and compared the results to the prediction based on aggregate traffic.

A. SARIMA modeling

A network traffic trace consists of a series of observations in a dynamical system environment. Fig. 1 shows a time series corresponding to the hourly number of calls for talk groups 1 and 2. In time series analysis, a classical SARIMA model is used to model and predict traffic. SARIMA time series model was applied to NSFNET traffic [10] and sub-networks [11] for traffic model fitting and forecasting.

The general ARIMA model contains autoregressive (AR) and moving average (MA) parts and explicitly includes differencing in the formulation of the model. The model parameters are: the autoregressive parameter (p), the number of differencing passes (d), and the moving average parameter (q). ARIMA models are classified as ARIMA (p, d, q) [8]. Due to the presence of natural diurnal and weekly patterns in the E-Comm traffic data, we used Seasonal ARIMA (SARIMA), a variation of the ARIMA model. The model introduces into the ARIMA model a seasonal period parameter (S), a seasonal autoregressive parameter (P), the number of seasonal differencing passes (D), and a seasonal moving average parameter (Q). A SARIMA $(p, d, q) \times (P, D, Q)_S$ model can be represented as:

$$\phi(B^s)\phi(B)(1 - B^s)^D(1 - B)^d X_t \ = \ \theta(B^s)\theta(B)Z_t,$$

where $\phi(B)$ and $\theta(B)$ represent the AR and MA parts, $\phi(B^s)$ and $\theta(B^s)$ represent the seasonal AR and seasonal MA parts, respectively. B is the back-shift operator $(B^i X_t \ = \ X_{t-i})$.

B. Traffic prediction based on aggregate traffic

The aggregate network traffic consists of all network users' traffic. We use the R system [12] to identify, estimate, and verify the SARIMA model for the aggregate users' traffic. Because the network traffic possesses inherent diurnal and weekly patterns, we selected both 24-hour (one day) and 168-hour (one week) intervals as seasonal period parameters.

The models and parameters fitted for the network traffic are shown in Table IV. Based on m past traffic data samples, we were able to forecast the future n traffic data. Normalized mean square error $nmse$ was used to measure the prediction quality by comparing the deviation of the predicted data with the observed data:

$$nmse(a, b) \ = \sum_{i=m+1}^{m+n} \frac{(a_i - b_i)^2}{a_i^2},$$

where a_i is the observed and b_i is the predicted data.

TABLE IV
PREDICTION RESULTS BASED ON AGGREGATE TRAFFIC

p	d	q	P	D	Q	S	m	n	$nmse$
2	0	1	0	1	1	24	1,920	24	0.1941
3	0	1	0	1	1	24	1,920	24	0.1907
2	0	1	0	1	1	24	1,680	168	0.4079
3	0	1	0	1	1	24	1,680	168	0.4081
2	0	1	0	1	1	168	1,920	24	0.0969
3	0	1	0	1	1	168	1,920	24	0.1012
2	0	1	0	1	1	168	1,680	168	0.1745
3	0	1	0	1	1	168	1,680	168	0.1748

Two groups of models, with 24-hour and 168-hour seasonal periods, are shown in Table IV. When used to predict the same period of future data based on the same amount of training data, the four models with a 168-hour seasonal period provided better prediction results (indicated by smaller $nmse$) than the four 24-hour period based models. The 168-hour season-based models perform much better, particularly when predicting long term traffic data. A drawback of the 168-hour based models is the large computational cost. In model fitting and forecasting, they require 200 times more CPU time than the 24-hour based models.

A visual comparison of the prediction results of the 24-hour and the 168-hour models in predicting one future week of traffic based on the 1,680 past hours is shown in Fig. 3. The visual comparison is consistent with the $nmse$ metric. The model with 168-hour period leads to more accurate prediction than the 24-hour period model.

C. Traffic prediction based on user clusters

A key assumption of the developed aggregate users based prediction model is that the adopted model is static: the number of network users and their behavior pattern is constant in time. However, this assumption does not hold when planning further network expansions. Hence, a static model cannot be

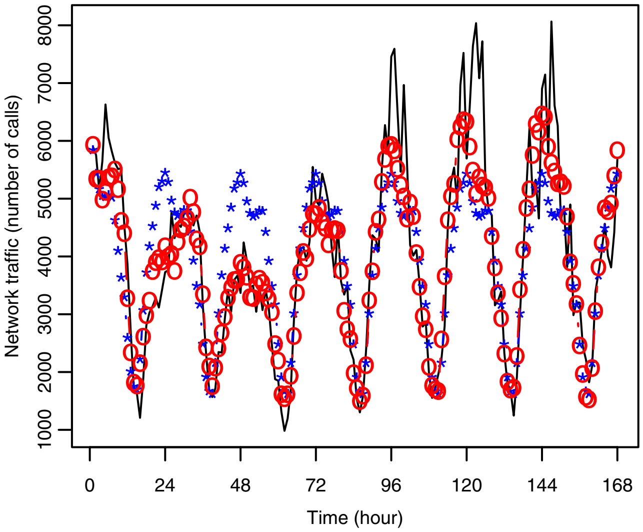

Fig. 3. Prediction of 168 hours of traffic based on 1,680 past hours. Solid line: observation, "o": prediction of 168-hour seasonal model, and "*": prediction of 24-hour seasonal model. Prediction of the 168-hour based model fits the observations better than the prediction of the 24-hour based model.

used to forecast network traffic. Therefore, we employed a user clusters based prediction approach to predict the network traffic by accumulating the prediction results from user clusters. In large networks with many users, it is impractical to predict individual users' traffic and then aggregate the predicted data. With user clusters, traffic prediction is reduced to predicting and aggregating users' traffic from few clusters.

We used the K-means algorithm to partition talk groups into three clusters. K-means produced a fewer number of clusters (3) compared to AutoClass (24). For each cluster, we employed the SARIMA model $(2,0,1) \times (0,1,1)_{24}$ to predict the traffic. The results of predicting network traffic by aggregating the traffic predicted from three clusters of users are shown in Table V.

TABLE V
PREDICTION RESULTS BASED ON USER CLUSTERS

cluster	*(p,d,q)*	*(P,D,Q)*	*S*	*m*	*n*	*nmse*
1	(2,0,1)	(0,1,1)	24	1680	48	1.1954
2	(2,0,1)	(0,1,1)	24	1680	48	2.4519
3	(2,0,1)	(0,1,1)	24	1680	48	0.3701
*	(2,0,1)	(0,1,1)	24	1680	48	0.6298
A	(2,0,1)	(0,1,1)	24	1680	48	0.6256
B	(2,0,1)	(0,1,1)	24	1680	48	**0.4231**

In Table V, rows 1, 2, and 3 show traffic prediction results for user clusters 1, 2, and 3, respectively. The row marked with "*" is the aggregate user traffic prediction obtained without clustering the users, which was also used for prediction shown in Table IV. Row "A" shows the weighted aggregate prediction of network traffic based on the three user clusters. Row "B" shows the optimized weighted aggregate prediction. Note that the $nmse > 1.0$ for clusters 1 and 2 implies that the prediction results for these clusters are worse than the prediction based on the mean value of past data. A much better prediction, shown in row "B", was achieved when the mean value prediction was applied for clusters 1 and 2. Even the unoptimized cluster-based prediction (row "A"), is not worse than the prediction using aggregate traffic. The advantage of the user cluster-based prediction is that prediction of traffic in networks with a variable number of users is possible, as long as the new user groups could be classified into the existing user clusters.

VI. CONCLUSIONS

In this paper, we analyzed traffic data collected from an operational trunked radio network. We used the K-means algorithm and AutoClass to classify network users into user clusters. We predicted network traffic using the SARIMA model based on aggregate user traffic and based on three user clusters that produced better prediction results. The user clusters based prediction approach is also applicable to networks with variable number of users where prediction based on aggregate traffic could not be applied. Our research may enable network operators to predict network traffic and may provide guidance for network expansion.

VII. ACKNOWLEDGMENT

The authors thank D. Sharp from Planetworks and the management and technical staff at E-Comm for providing access to the activity data and technical support for data analysis. This research was supported by the NSERC under Grant 216844-03 and Canada Foundation for Innovation.

REFERENCES

[1] D. Tang and M. Baker, "Analysis of a metropolitan-area wireless network," in *Proc. MOBICOM 1999*, Seattle, WA, USA, Aug. 1999, pp. 13–23.

[2] L. A. Andriantiatsaholiniaina and Lj. Trajković, "Analysis of user behavior from billing records of a CDPD wireless network," in *Proc. IEEE Workshop on Wireless Local Networks (WLN) 2002*, Tampa, FL, Nov. 2002, pp. 781–790.

[3] D. Sharp, N. Cackov, N. Lasković, Q. Shao, and Lj. Trajković, "Analysis of public safety traffic on trunked land mobile radio systems," *IEEE J. Select. Areas Commun, Special Issue on Quality of Service Delivery in Variable Topology Networks*, (in print).

[4] E-Comm–Emergency Communications for SW British Columbia [Online]. Available: http://www.ecomm.bc.ca.

[5] P. Cheeseman and J. Stutz, "Bayesian classification (AutoClass): theory and results," in *Advances in Knowledge Discovery and Data Mining*, U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, Eds., AAAI Press/MIT Press, 1996.

[6] L. Kaufman and P. J. Rousseeuw, *Finding Groups in Data: An Introduction to Cluster Analysis*. New York: John Wiley & Sons, 1990.

[7] J. W. Han and M. Kamber, *Data Mining: Concepts And Techniques*. San Francisco: Morgan Kaufmann Publishers, 2001.

[8] G. E. P. Box and G. M. Jenkins, *Time Series Analysis: Forecasting and Control*. San Francisco: Holden-Day, 1976.

[9] EDACS Explained [Online]. Available:
http://www.trunkedradio.net/trunked/edacs/EDACS_Whitepaper.pdf.

[10] N. K. Groschwitz and G. C. Polyzos, "A time series model of long-term NSFNET backbone traffic," in *Proc. IEEE International Conference on Communications (ICC'94)*, vol. 3, New Orleans, LA, May 1994, pp. 1400–1404.

[11] Y. W. Chen, "Traffic behavior analysis and modeling sub-networks," *International Journal of Network Management*, John Wiley & Sons, vol. 12, 2002, pp. 323–330.

[12] The R Project for Statistical Computing [Online]. Available: http://www.r-project.org.