# SONETOR: a Social Network Traffic Generator

César Bernardini, Thomas Silverston and Olivier Festor
Université de Lorraine, LORIA, UMR 7503, Vandoeuvre-les-Nancy, F-54506, France
Inria Nancy – Grand Est, Villers-les-Nancy, F-54600, France
Email: {cesar.bernardini, thomas.silverston, olivier.festor}@inria.fr

***Abstract*—The Online Social Networks (OSN) have become an important trend in current networks. Due to the susceptible nature of the private data available in OSN, the acquisition of data sets is not an easy task. In this paper and based on the state of the art of measurement studies, we present SONETOR, a synthetic social network traffic generator, characterized by ease of use and flexibility. SONETOR represents current social network behavior such as user publishing and consuming content, users sharing and commenting on information with their friends or viral contagion of videos. Using SONETOR, we also proceed to study the impact of OSN in the network traffic. In particular, our social network traffic traffic generator allows capturing the effect of the flash crowd phenomenon. SONETOR is an open-source and multi-platform tool freely available. The generated traces can be widely distributed without restrictions and they are still privacy compliant.**

## I. Introduction

Online Social Networks (OSNs) are currently massively used in the Internet. OSNs allow users to create virtual communities and to share information about their life (pictures, job update, relationships), news or content. As an example, Facebook arose in 2007 and it counts already a billion of users, among which 700 million users connect to the website at least once a day [1] while Twitter or LindkedIn are also among the most popular websites in the Internet. Following this trend, every social event defines a communication plan through OSNs. Presidential election uses Twitter to spread information to a larger audience [2]. TV shows encourage audience to participate through OSNs by using specific keywords in Twitter. Most if not all the Internet services improve the users' experience through the addition of social features to rapidly spread interesting content. Companies also invest strongly into OSNs in order to promote new products or attract new customers [3] [4]. The Internet has become a Social-oriented Network.

The network workload is an essential characteristic to take into account while modeling or simulating new mechanisms or protocols for the Internet [5]. The OSNs have taken an important share of the overall Internet traffic. Indeed, Facebook, Twitter and LinkedIn are among the ten most accessed websites in the Internet [6]. It is therefore essential to include social characteristics into network traffic model.

In this paper, we present SONETOR, a *social network traffic generator* that statistically models users' interaction within social network. SONETOR extracts parameters from social network measurement studies and allows generating sequences of users' activities within their communities. Our generator is also able to capture traditional effects observed in the Internet such as the massive popularity of content in a short period of time (i.e. flash crowds). The obtained OSN synthetic traces can be directly included into simulations or models and are essential for further evaluation of new mechanisms or protocols. To the best of our knowledge, we are the first ones to produce synthetic social network traffic that accurately reflects the users' interaction within a social network.

We then use SONETOR to study the impact of social network on the content popularity. Indeed, the content popularity on the Internet is traditionally modeled with MZipf distribution function and we show through the use of our generator that the users' interaction within social network has an important impact on the distribution parameters.

The rest of this paper is organized as follows. We review in Section II the related work on network traffic models and the users' behavior into social networks. Then, in Section III, we present our new generator SONETOR and its internal model. Section IV introduces the content popularity model and the simulation environment; and we then show the impact of social networks into the popularity model. Finally, in Section V, we sum up our findings, conclude the article and expose the future work.

## II. Related Work

Numerous studies intend to model the network traffic in diverse environments: Web traffic to test Web servers and internet architectures [5]; e-mail traffic to investigate SPAM [7]; P2P traffic to study improvements and changes on the protocols [8]; Future Internet architectures are not implemented yet, and aim also to be evaluated through simulation experiments with future Internet traffic, which is an extrapolation of the current Internet traffic [9]. Nevertheless, there does not exist utilities that represent the users' interaction in online social networks.

In the last years and due to its exponential growth, there has been a huge amount of studies and analysis of Online Social Networks. [10] shows evolution on Twitter relationships while [11] study the first year of operation of Google+; [12] performs an in-deep evaluation on popularity of Youtube videos; [13] has a similar approach but using HTTP requests to consequently model the popularity of videos; [14] deepens on study of trends in Twitter: evolution in time, participation of users. [15], [16], [17] characterize user interaction and patterns of usage in OSNs. Although these studies allow understanding the social networks, their results are difficult

```
Timestamp 1    User a    Activity(d 1, ..., d m)    [opt 1, opt 2...]
Timestamp 2    User b    Activity(d 1, ..., d m)    [opt 1, opt 2...]
Timestamp 3    User c    Activity(d 1, ..., d m)    [opt 1, opt 2...]
...
```

Fig. 1: Format of Sonetor traces

to compare and even more difficult to cross-validate between them. Most of them are based on real OSN data obtained through measurement studies, which is not publicly available. For instance, [15], [16], [17], [13] analyze users' behavior in Orkut, Facebook, LinkedIn, collecting the data through private agreements with the OSNs. Otherwise, active measurement through the use or web crawlers are also a way to collect data from OSN websites such as Twitter[18], [10], [14], Youtube[12], Google Plus[11].

As we have pointed it out, there are no tools available to model OSN traffic and most of the studies depend on traces that are not openly available. In this work, we propose *SONETOR*, a SOcial NEtwork Traffic generatOR. It aims at offering an open and modifiable tools to generate network traffic workload with OSN characteristics.

## III. SONETOR Architecture

### A. Overview

We present an open-source and multi-platform SOcial NEtwork Traffic generatOR, called *SONETOR*, characterized by ease of use and flexibility.

SONETOR generates synthetic traces, and a sample is presented in Figure 1. This Figure describes a trace where each line has a timestamp corresponding to an user $User_x$ performing an activity. Each activity can also count several dependent parameters. It is noteworthy to mention that there are other options ($opt_i$) for further extension of our generator tool. Such extensions can take into account mobility of users, or their geographical coordinates for more sophisticated scenarios.

In the rest of the section, we describe in-detail the SONETOR architecture and the three models it relies on: a social network model capturing the social relationships between users, a users' interaction model capturing the activity scheduling for each user, and a model for each type of activity and its dependent parameters.

### B. Social Relationships Model

As we model a social network, it is required to represent the connections between users. The connection between users can be described by a social network graph. This social network graph may represent friend relationships as well as work belonging depending on the social network to be modeled (e.g.: Facebook, LinkedIN, etc.) *SONETOR* can include several graph models such as small-world or random graphs. Besides graph models, SONETOR includes also realistic social network graphs by extracting social relationships from data set publicly available for Facebook [19].

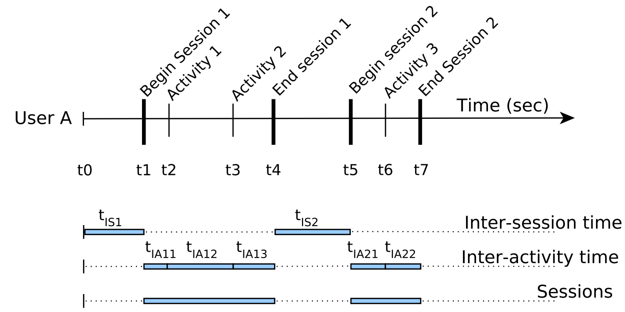



Fig. 2: Users' interaction model. Users can perform several activities within a session

### C. Users' Interaction Model

Our interaction model is based on several major research studies on social networks [16], [17].

The interactions of a user in a social network are depicted in Figure 2 and can be summarized as follows: a user may start $NS$ independent and consecutive sessions. The interval time between each session is calculated with an inter-session time $t_{IS_i}$ from the time origin $t0$ or the previous session ending time. In every session $i$, the user performs a finite number of activities, $NA_i$, such as publications or retrievals from friends. These activities are separated by an inter-activity time $t_{IA_{ij}}$. In the Figure 2, we illustrate an example of user interactions in-which User A has two sessions ($NS = 2$); the first session starts after the inter-session time $t_{IS_1}$ and contains two activities ($NA_1 = 2$), which are separated by three different inter-activity times $t_{IA_{11}}$, $t_{IA_{12}}$ and $t_{IA_{13}}$ . The second session counts only one activity ($NA_2 = 1$) and is separated from the first session by a second inter-session time $t_{IS_2}$.

The four parameters we extracted for this users' interaction model are the number of sessions $NS$, the inter-session time $t_{IS_i}$, the number of activities per session $NA_i$ and the inter-activity time $t_{IAij}$.

### D. Activities and Dependent Parameters Model

We presented how to model the social relationship between users (Section III B) and the activity scheduling (Section III C), we now present how to model the type of activity. From the trace format (Figure 1), every user $User_x$ executes an $Activity$ at instant $Timestamp_i$. The parameter $Activity$ corresponds to the type of activity being performed by the user. An activity has also dependent parameters $d_i$ depending on the type of activity. For instance, a picture publication (*Activity*) includes necessarily other dependent parameters to the activity such as a text description ($d_1$), a title ($d_2$) and the image itself ($d_3$).

SONETOR includes several Markov Chain that represent distinct users' behavior. For instance some users are more likely to published pictures than video (or vice versa). Every modeled users has an assigned Markov Chain. The user selects the type of activity to perform. Depending on the type of activity, its dependent parameters are computed.

Our tool *SONETOR* is freely available at [20]. As the traces have been obtained with statistical information, traces respect policies of privacy of every single user. The generated traces can also be freely distributed and be replayed by other experiments.

## IV. Impact of OSN into Content Popularity Model

Once we have shown *SONETOR*, we use it to generate social network activity traces. In this section, these traces are analytically analyzed in order to study changes induced by the social networks on the content popularity model.

The content popularity model is a function that establishes the popularity of every piece of content: how often every single piece of content is going to be requested. A popular file is more likely to be requested than an unpopular one. The popularity model is commonly represented with a probability distribution function. Measurements studies have shown that millions of users are attired by only a few select sites, while they give little or not attention to millions of others. This fact is consistent with the power law used in most of content popularity models.

Zipf and Maldelbrot-Zipf belong to this family and they have been chosen in most of the simulation scenarios[21], [9]. MZipf is a generalization of Zipf. With $\alpha = 0$, Zipf is equal to an uniform distribution. There are not consensus about the final and right values for the function. In the literature, its most representative parameter ($\alpha$) varies from 0.6 to 2.5. The catalog of the PirateBay is represented with $\alpha = 0.75$, DailyMotion content using $\alpha = 0.88$ while the VoD in China has an $\alpha$ ranged between 0.65 and 1.0 [22]. Zipf is commonly used in different simulation environments such as CCNSim or models of Content Delivery Networks (CDNs) [23], to model load of web servers, etc.

Experiments based on MZipf laws assign probabilities to a fixed catalog of content. Thus, every time a piece of content is demanded it is selected with a probability distribution. Depending on the configuration of the $\alpha$ parameter, some pieces of content are more likely to be consumed than others.

In the case of social networks, a (M)Zipf function decides which piece of content to be published. The retrieve of content is decided according to a social graph: users have social relationships i.e. acquaintances or friends. In our experiments, users retrieve their friends content through a timeline composed with the latest publications of friends such as Facebook does. In brief, the users publish pieces of content according to a MZipf function but they retrieve content based on their social connections and their latest published content.

### A. Simulation Parameters

In this section, we describe the notation and simulation parameters that we need to perform the experiments. All the simulation parameters are summarized in the Table I.

| Social Network Activity Traces | |
|---|---|
| Number of sessions $NS$ | Zipf ($\alpha = 1.792, \beta = 0.0$) |
| Number of activities $NA_i$ | Zipf ($\alpha = 1.765, \beta = 4.888$) |
| Inter-session time $t_{IS_i}$ (s) | LogNormal ($\mu = 2.245, \sigma = 1.133$) |
| Inter-activity time $t_{IA_{ij}}$ (s) | LogNormal ($\mu = 1.789, \sigma = 2.366$) |
| **Catalog Configuration** | |
| Size | $10^4$ Pieces of Content |
| MZip Parameters | $\alpha = \{0.65, 1.1\}$ |
| | $\beta = 0.0$ |
| **Social Network graph** | |
| #Users | 4,039 |
| #Links | 88,234 |
| #Avg. Degree | 44 |

TABLE I: Simulation Parameters

All along this paper, we contrast results using different content popularity models with and without social networks. From now and so on, the scenarios with content popularity model based on social networks will be called *OSN* traffic while scenarios with popularity model based on a Zipf/MZipf functions will be called *regular* traffic. All the experiments are performed with a catalog counting 10,000 pieces of content.

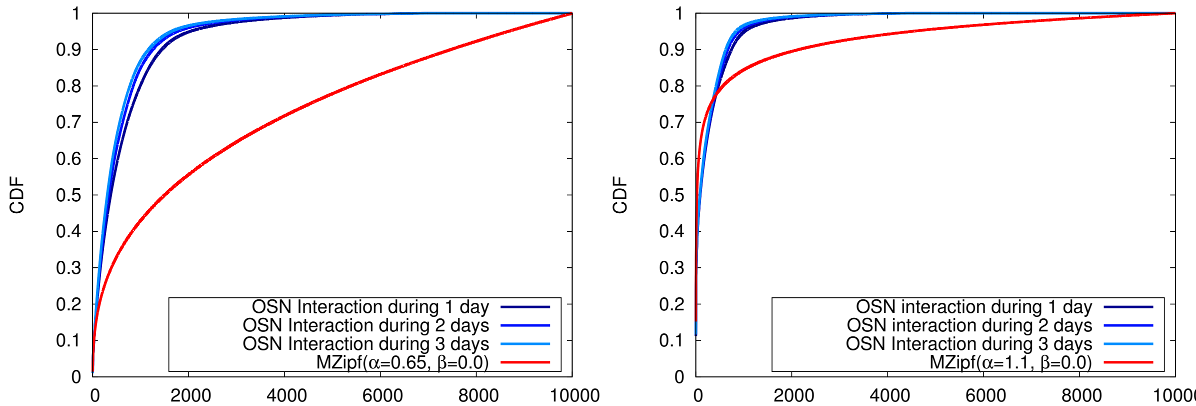
In order to model the social relationships between users, we resort to a Facebook data set publicly available [24]. Facebook is the most popular OSN and the data set consists of 4,039 users, 88,234 friend relationships and each user counts in average 44 relationships.

Online Social Networks (OSN) allow users to publish content at their own will and share it with their acquaintances (i.e., *friends*). Friends may always be updated through a timeline. Thus, we model a *social network* by a network where users can publish, retrieve and share information with their communities, according to their personal preferences. In our social model, each user has therefore two functionalities to interact with its community: *Publish* and *Retrieve*, as defined as follows:

- *Publish*: the production of new content. After retrieving a content, users may share it again with their friends (i.e., *re-tweet* a message).
- *Retrieve*: this function allows users to receive the last content issued by all their friends.

As we mentioned in Section III-D, the activities have dependent values. In case of *Publish* activity, the dependent values are content name which is decided with a popularity model explained in the following section and its filesize. In case of *Retrieve* activity, the dependent values are the user whom to retrieve last updates and it is decided with the social graph. It always selects all their friends. The appearance of the SONETOR traces is shown in the Figure 3.

In the following, we analyze the impact of social networks into the content popularity model. First, we investigate the probability distribution found on a stand-alone OSN traffic scenario. Second, we study a mixed scenario where OSN and regular traffic coexist. And last but not least, we show the presence of flash crowds in the stand-alone OSN traffic scenario.




(a) MZipf ($\alpha = 0.65$, $\beta = 0.0$) (b) MZipf ($\alpha = 1.1$, $\beta = 0.0$)

Fig. 4: Stand-alone Scenario: impact of the OSN on the content popularity model

```
03.33    UserB    Retrieve(UserA, UserK, UserW)
05.21    UserB    Publish(108, 1024KB)
07.44    UserA    Retrieve(UserB, UserE)
12.35    UserB    Publish(2, 580KB)
```

Fig. 3: Example of a Sonetor trace

### *B. Stand-alone Scenario with OSN Traffic*

We aim at discovering the changes on the content popularity model provoked by the interaction of users within social networks. We start the assessment on a stand-alone scenario where all the traffic comes from social networks. In this experiment, we reproduce *SONETOR* traces to analyze the impact of social networks into the content popularity model. At the beginning of the experiment, we decide a popularity model such as MZipf($\alpha = 1.1$; $\beta = 0$). This popularity model was selected at the generation of synthetic traces and it is used to select content to be published. The consumption of content is realized by the users; users receive an update of their friends' publication. The users do not consume all the content from their friends but only a small subset based on influence model [25]. Then, we bound the synthetic trace to a precise time limit such as 1, 2 or 3 days. Once the bounded synthetic traces are executed, we build a probability distribution with the consumed content. We proceed to draw the probability distribution found and to fit the curve with a new power-law distribution.

In Figures 4b and 4a, we show charts with distinct MZipf configurations. The charts are built with the number of access for every piece of content. The red line represents the original popularity model configuration while the blue lines stand for the popularity model found after 1, 2 and 3 days of social network interaction. The curves correspond to the Cumulative Distribution Function (CDF) of the probability distribution found after 1, 2 and 3 days of interaction in social networks.

Using a popularity model MZipf($\alpha = 0.65$; $\beta = 0$), the 90% of most requested content consists of 7,466 pieces of content. While using OSN, the 90% of most requested content get reduced to only 1,165 pieces. It means the subset of most popular content get reduced to 16% of its original number of pieces. It is important to remark as well that in the OSN case, only 6,992 pieces of content were consumed: it means 30% of the content is completely ignored.

Those results are confirmed using the model MZipf($\alpha = 1.1$; $\beta = 0$), the introduction of OSN produces that the 55% of content is ignored and the subset of most popular content is reduced to 29% of its original number of pieces.

With the two figures, we can draw some conclusions. The viral effect of OSN provokes many users to consume less diverse content and it strengthens the importance of popular elements. As we see in both cases, the subset of most popular contents is reduced in important proportions. The social networks provokes the creation of *super* popular subsets of content. These pieces of content are highly demanded and may have an important impact on the general behavior of the network. In the OSN there are many pieces of content that are completely ignored and never consumed. This fact reveals that many of the content are irrelevant to most of the users. Even more, in the social network, the number of published content is a subset of all the pieces of content found in the popularity model. In other words, the content found in social networks is subset of all the content found in the Internet. In the charts we have seen how the ratio of popular elements got decreased.

Using the least squares method, we fit the distributions to MZipf functions. We present the results in the Table II. The results show good approximations to the function after 1, 2 and 3 days of OSN traffic. As we can observe, the alpha values have grown significantly, which means a few popular content increases its popularity while most of the other stay unpopular.

| Original Content Popularity Model (*input*) | | | Obtained Content Popularity Model (*output*) | |
|---|---|---|---|---|
| Distrib. | Parameters | Period | Distrib. | Parameters |
| MZipf | $\alpha = 0.65$ $\beta = 0.0$ | 1 day | MZipf | $\alpha = 5.27$ $\beta = 2114.89$ |
| MZipf | $\alpha = 0.65$ $\beta = 0.0$ | 2 days | MZipf | $\alpha = 5.59$ $\beta = 2009.75$ |
| MZipf | $\alpha = 0.65$ $\beta = 0.0$ | 3 days | MZipf | $\alpha = 5.19$ $\beta = 1664.63$ |
| MZipf | $\alpha = 1.1$ $\beta = 0.0$ | 1 day | MZipf | $\alpha = 2.43$ $\beta = 181.23$ |
| MZipf | $\alpha = 1.1$ $\beta = 0.0$ | 2 days | MZipf | $\alpha = 2.55$ $\beta = 191.67$ |
| MZipf | $\alpha = 1.1$ $\beta = 0.0$ | 3 days | MZipf | $\alpha = 2.59$ $\beta = 189.62$ |

TABLE II: Distribution Parameters for the Stand-alone Scenarios

### C. Mixed Scenario with OSN and Regular Traffic

We have already shown the impact of social networks in an environment where all the traffic is produced by social networks. Now, we are interested in mixed scenarios where traffic is composed of two types of traffic: social network and regular traffic. OSN and regular traffic are going to coexist in the near future. We argue that the penetration of social networks into the overall traffic will incur changes in the content popularity model. These changes are analyzed in this section.

To this end, we simulate scenarios with different ratio between *regular* and *OSN* traffic (i.e., 100%-0%; 90%-10%; 80%-20%; 50%-50%; 0%-100%). We then present in Figure 5 the CDF of the content popularity obtained with all the mixed scenarios. We represent the execution of the social network activity traces in a three-days period. For lack of space, we only show the chart for regular traffic with MZipf ($\alpha = 0.65$), still in the Table III the other configurations are shown.

We fit the mixed scenarios with a MZipf function. We summarize the obtained MZipf parameters in the Table III. As seen in Fig. 5, the curves for regular traffic and for the mixed scenario with 10% of OSN traffic seems similar. The mixed scenario with 10% of OSN traffic has an apparently minimal impact in the MZipf obtained MZipf parameter: the $\alpha$ parameter passes from 0.65 to 0.71, which means the 20% of the most popular content passes from 151 pieces of content to 142. If we consider the 90% of all the content, it passes from 7,458 to 7,264 pieces of content: a reduction of 9% and 3% in the most popular contents respectively. The same phenomenon is also observed for the OSN traffic ranging from 20% to 100% but at higher scale. For instance, for the 50% OSN traffic, the $\alpha$ parameter passes from 0.65 to 1.14.

From this experiment, we observe that OSN traffic has an impact on the content popularity model. While we increase the ratio of OSN traffic, we observe the growth of the $\alpha$ parameter (Table III). A higher value for the $\alpha$ parameter implies a smaller subset of *super* popular content (i.e. the shape of the curve tends to the left side). Then with OSN traffic, there is a small subset of *super* popular content that is highly requested while the other content stay unnoticed.

| Original Content Popularity Model (*input*) | Obtained Content Popularity Model (*output*) | | |
|---|---|---|---|
| MZipf0.65 + 10%OSN | MZipf | $\alpha = 0.71$ | $\beta = 10.57$ |
| MZipf0.65 + 20%OSN | MZipf | $\alpha = 0.81$ | $\beta = 38.90$ |
| MZipf0.65 + 50%OSN | MZipf | $\alpha = 1.14$ | $\beta = 123.81$ |
| MZipf0.65 + 100%OSN | MZipf | $\alpha = 5.19$ | $\beta = 1664.63$ |
| MZipf1.1 + 10%OSN | MZipf | $\alpha = 1.11$ | $\beta = 0.40$ |
| MZipf1.1 + 20%OSN | MZipf | $\alpha = 1.16$ | $\beta = 1.78$ |
| MZipf1.1 + 50%OSN | MZipf | $\alpha = 1.38$ | $\beta = 16.09$ |
| MZipf1.1 + 100%OSN | MZipf | $\alpha = 2.59$ | $\beta = 185.92$ |

TABLE III: Distribution Parameters for the Mixed Scenario

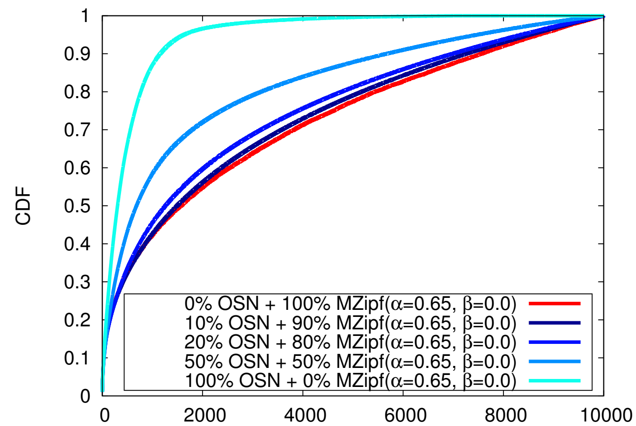



Fig. 5: Mixed Scenario: impact of the OSN on the content popularity model

This observation can have a significant impact for the Caches of the Internet. Indeed, with the rise of caching architectures such as Content Delivery Networks (CDN) and Content Centric Networks (CCN), in-network caching has become an important issue for the Internet [26]. By reducing the subset of *super* popular content, it reduces the number of relevant content to be stored into Caches. It then alleviates the load on the Caches, saves resources (e.g.: memory) and improves the performances of the Internet caches

### D. Flash-Crowd Effect

In the networks, the term *Flash-Crowd* refers to a piece of content actively demanded on the Internet for a short period of time [27]. In this section, we study the *Flash-Crowds* and its correlation with social networks.

We study content consumption for a particular piece of content during one-day period. We track all the timestamps in-which the content has been consumed. We then contrast the number of demands between regular traffic and OSN traffic scenarios.

The histogram is shown in Figure 6, the x-axis represents the time in seconds while the y-axis represents the percentage of requests for the selected piece of content. For sake of clearness, we zoom on a one hour period (from 2 h to 3 h). With regular traffic, the requests for content seem to be uniformly distributed and it seem to represent a dense solid line. The OSN traffic shows separated spikes of demand, which are the so-called Flash Crowds.

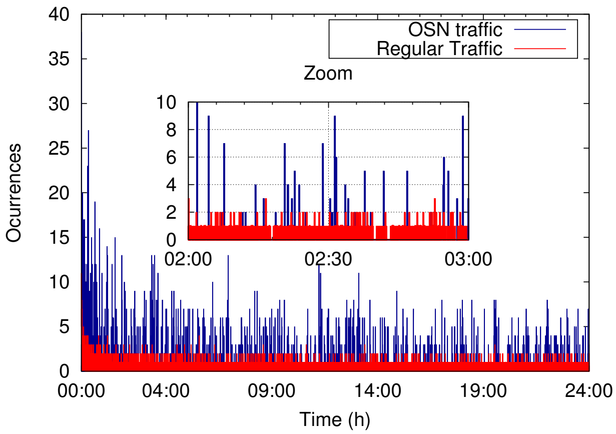



Fig. 6: Capturing Flash Crowd in the Content popularity Model with SONETOR

Using regular traffic (in red), we observe a dense concentration of points and the absence of peaks. This fact points out that the number of requests for certain piece of content is nearly constant with time. Using OSN traffic, the charts behaves in a different manner: The histogram shows high peaks and less dispersion which highlights the presence of multiple *Flash Crowds* using social networks. This fact can be observed clearly in the zoom chart (Figure 6).

We believe the correct handling of flash crowds processes is going to be one of the main issues for the Future Internet. Social networks tend to privilege the distribution of content in a short period of time, this short period of time involves a high number of requests for a piece of content which becomes the content popular. Once the short period of time has passed, the content stop being popular and it is barely demanded. In typical caching and routing problems, the correct handling of flash crowds involves generating copies and updating routing tables in order to evolve the end-user or overall network performance.

## V. Conclusions & Future Work

In this paper, we propose *SONETOR*, an OSN synthetic traffic generator. *SONETOR* is based on a social network model capturing the social relationships between users, a users interaction model for the users' activity scheduling, and an activity model to describe distinct type of activities.

We use *SONETOR* and study the impact of social relationships in the content popularity. The social relationships between users enforce that many pieces of content become popular and are spread massively throughout the OSN while many others passed unnoticed. It has a major impact on the content popularity model traditionally used in the Internet. With SONETOR, we showed that OSN privileges a subset of super-popular content. By reducing the number of popular content, network caches can improve their performances with the same storage capabilities. It is an important results as in-network caching are nowadays largely studied with network architectures such as data-centers, Content Distribution Networks or Information-Centric Networks.

As future work, we aim at studying other components of the Internet traffic. SONETOR may also serve to validate future protocols or networks architectures in a social environment.

## References


[1] "http://www.insidefacebook.com/2013/08/23/canadians-still-the-most-active-facebook-users-in-the-world/."

[2] "http://mashable.com/2012/11/06/obama-wins-twitter."

[3] K. Song, D. Wang, S. Feng, and G. Yu, "Detecting opinion leader dynamically in chinese news comments," in *WAIM'11*, 2012.

[4] "Health leaders media article. http://goo.gl/ljbwd."

[5] J. Cao, W. Cleveland, Y. Gao, K. Jeffay, F. Smith, and M. Weigle, "Stochastic models for generating synthetic http source traffic," in *IEEE INFOCOM*, 2004.

[6] "http://www.alexa.com/topsites/."

[7] P. Svoboda, W. Karner, and M. Rupp, "Modeling e-mail traffic for 3g mobile networks," in *IEEE PIMRC*, 2007.

[8] S. Naicken, B. Livingston, A. Basu, S. Rodhetbhai, I. Wakeman, and D. Chalmers, "The state of peer-to-peer simulators and simulations," *SIGCOMM Comput. Commun. Rev.*, vol. 37, no. 2, 2007.

[9] D. Rossi and G. Rossini, "Caching performance of content centric networks under multi-path routing (and more)," Telecom ParisTech, Tech. Rep.

[10] V. Arnaboldi, M. Conti, A. Passarella, and R. Dunbar, "Dynamics of personal social relationships in online social networks: a study on twitter," in *ACM COSN*, 2013.

[11] R. Gonzalez, R. Cuevas, R. Motamedi, R. Rejaie, and A. Cuevas, "Google+ or google-?: dissecting the evolution of the new osn in its first year," in *WWW*, 2013.

[12] M. Cha, H. Kwak, P. Rodriguez, Y.-Y. Ahn, and S. Moon, "I tube, you tube, everybody tubes: analyzing the world's largest user generated content video system," in *ACM SIGCOMM IMC*, 2007.

[13] H. Li, H. Wang, J. Liu, and K. Xu, "Video requests from online social networks: Characterization, analysis and generation," in *IEEE INFOCOM*, 2013.

[14] H. Kwak, C. Lee, H. Park, and S. Moon, "What is twitter, a social network or a news media?" in *WWW*, 2010.

[15] F. Schneider, A. Feldmann, B. Krishnamurthy, and W. Willinger, "Understanding online social network usage from a network perspective," in *ACM SIGCOMM IMC*, 2009.

[16] F. Benevenuto, T. Rodrigues, M. Cha, and V. Almeida, "Characterizing user behavior in online social networks," in *ACM SIGCOMM IMC*, 2009.

[17] L. Gyarmati and T. Trinh, "Measuring user behavior in online social networks," *Netwrk. Mag. of Global Internetwkg.*, 2010.

[18] E. Bakshy, J. Hofman, W. Mason, and D. Watts, "Everyone's an influencer: quantifying influence on twitter," in *ACM WSDM*, 2011.

[19] J. McAuley and J. Leskovec, "Learning to discover social circles in ego networks," in *NIPS 2012*, 2012.

[20] "Sonetor website. http://www.loria.fr/~bernardc/sonetor."

[21] L. A. Adamic and B. A. Huberman, "Zipf's law and the Internet," *Glottometrics*, vol. 3, 2002.

[22] C. Fricker, P. Robert, J. Roberts, and N. Sbihi, "Impact of traffic mix on caching performance in a content-centric network," in *IEEE NOMEN*, 2012.

[23] S. Fayazbakhsh, Y. Lin, A. Tootoonchian, A. Ghodsi, T. Koponen, B. Maggs, K. Ng, V. Sekar, and S. Shenker, "Less pain, most of the gain: incrementally deployable icn," in *ACM SIGCOMM*, 2013.

[24] J. McAuley and J. Leskovec, "Learning to discover social circles in ego networks," in *NIPS*, 2012.

[25] M. Granovetter, "Threshold Models of Collective Behavior," *American Journal of Sociology*, vol. 83, 1978.

[26] C. Bernardini, T. Silverston, and O. Festor, "Mpc: Popularity-based caching strategy for content centric networks," in *IEEE ICC 2013*.

[27] M. J. Freedman, "Experiences with coralcdn: A five-year operational view." in *NSDI*, 2010.