Adaptive Mobile Traffic Offloading

Filippo Rebecchi, Marcelo Dias de Amorim, Vania Conan

To cite this version:

Filippo Rebecchi, Marcelo Dias de Amorim, Vania Conan. Adaptive Mobile Traffic Offloading. ALGOTEL 2014 – 16èmes Rencontres Francophones sur les Aspects Algorithmiques des Télécommunications, Jun 2014, Le Bois-Plage-en-Ré, France. pp.1-4. hal-00976877

HAL Id: hal-00976877

https://hal.archives-ouvertes.fr/hal-00976877

Submitted on 11 Apr 2014

Adaptive Mobile Traffic Offloading

Filippo Rebecchi[1,2], Marcelo Dias de Amorim[2], Vania Conan[1]

[1] *Thales Communication & Security, 4 Av. des Louvresses, 92230 Gennevilliers, France*
[2] *LIP6/CNRS – UPMC Sorbonne Universités, 4 Place Jussieu, 75005 Paris, France*

Cellular operators count on the potential of offloading techniques to relieve their overloaded radio access networks. In this paper, we propose, design, and evaluate a re-injection strategy to finely control the opportunistic distribution of popular contents throughout a hybrid mobile network. The idea is to use the infrastructure resources as seldom as possible. Unlike existing techniques that bind re-injection to statically defined objective functions, our proposal adapts to the current network topology. This turns out to be particularly effective in highly dynamic scenario, where clustering prevent contents to diffuse properly. We assess the performance of our strategy by re-running a realistic large-scale (more than 10,000 nodes) vehicular dataset to disseminate contents under different tolerances to delay. The results show significant savings in the infrastructure load between 55% and 63%.

Keywords: Mobile data offloading, delay-tolerant networks, re-injection strategies.

---

1 Introduction

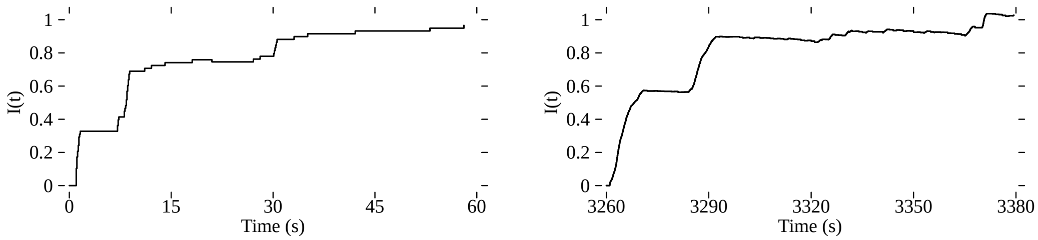
The recent boom in the smart devices market calls for efficient offloading strategies, as the global mobile traffic is expected to increase significantly in the near future [Ind13]. When many co-located users are interested in the same contents, it is possible to benefit from their mobility to shift a portion of the traffic from the primary (cellular) channel onto an alternative (terminal-to-terminal) channel, so to alleviate the load on the operator's cellular infrastructure by reducing redundant traffic.

We propose an adaptive re-injection based offloading scheme to efficiently distribute popular contents to a multitude of mobile users. The strategy adapts to the heterogeneous individual mobility pattern of nodes and to the current evolution of the dissemination process. The motivation behind this work is that this heterogeneity in mobility is responsible for a stepwise evolution of content dissemination, alternating between flat zones (plateaux) and steep periods of diffusion, as we will show in Section 2. The proposed algorithm detects the formation of plateaux in the content diffusion evolution, and, if needed, adaptively re-injects additional copies in the system to finely control the pace at which contents are disseminated. To this extent, the concept of a persistent feedback channel, connecting mobile users with a central offloading coordinator, becomes instrumental, allowing at each instant the infrastructure operator to track the content dissemination status and to anticipate appropriate re-injections.

We investigate the efficiency of the proposed scheme by mimicking a service in which a content, supposed of general interest, must be distributed to a multitude of users within a given maximum reception delay. In our evaluation, we employ a realistic large-scale vehicular trace derived from multiple fine-grained traffic measurements in the city of Bologna [iTE]. We compare our solution with an oracle (unrealistic, but considered only as a benchmark), as well as other offloading strategies based on static objective functions. The results display that the proposed algorithm substantially outperforms existing objective function-based offloading strategies for any considered delay tolerance, reducing by more than half the infrastructure load. Even more importantly, we also show that our approach performs very close to the oracle.

2 Stepwise epidemic diffusion in real datasets

We plot in Fig. 1 the evolution of the epidemic diffusion for two different datasets : the small-scale Rollernet dataset, composed of 62 nodes [TLB$^+$11] and the already mentioned large-scale Bologna dataset, composed of more than $10,000$ nodes [iTE]. To trigger the diffusion, the infrastructure injects a small



(a) Rollernet, 6 initial copies.

(b) Bologna, 10 initial copies.

FIGURE 1: Epidemic diffusion of the content. The diffusion behavior alternates between steep zones and flat zones, which is the result of heterogeneous node mobility.

number of initial copies to random nodes at t_0, monitoring how the diffusion of the message progresses through subsequent direct contacts. A node that has received the message is considered *infected*, while a node that has not yet received the content is *sane*. The instantaneous ratio of nodes infected $I(t) \in [0,1]$ follows a stepwise pattern, alternating between plateaux (flat areas) and periods of heavy infection (steep areas). Plateaux correspond to periods during which the dissemination does not progress, as no sane node comes into range of the already infected nodes. This phenomenon is intrinsically related to the heterogeneity of contact patterns. If the contact process of nodes were Poisson homogeneous and stationary, the epidemic diffusion in the network would follow the logistics equation shown by Zhang et al. [ZNKT07] – the resulting curve would not exhibit any plateau, which is in contrast with our observation.

To catch the heterogeneity of patterns, we adopt a Marked Poisson Process model, in which the meeting times of any two nodes (i,j) is Poisson with rate $\lambda_{ij} = \lambda \cdot p_{ij}$ [CMC06]. The inter-contact times T_{ij} are thus independent exponentials with parameter λ_{ij}, while matrix $C = (p_{ij})$ captures the patterns of contacts between nodes. In the homogeneous case, C is the identity matrix, i.e., all nodes can see each other with the same probability. Let $\mathcal{S}$ be the set of nodes infected at any given instant of the dissemination process. We are interested in the random plateau duration $T^p_{\mathcal{S}}$ during which the dissemination does not progress (nodes in $\mathcal{S}$ do not meet any other node from $\overline{\mathcal{S}}$). Looking at the set of links between nodes in $\mathcal{S}$ and its complement, one can see that $T^p_{\mathcal{S}} = \inf_{i\in\mathcal{S}, j\in\overline{\mathcal{S}}} T_{ij}$. By Poisson calculus, and noting the cut value $\partial\mathcal{S} = \sum_{i\in\mathcal{S}, j\in\overline{\mathcal{S}}} p_{ij}$, we find that $T^p_{\mathcal{S}}$ is an exponential random variable with parameter $\lambda \cdot \partial\mathcal{S}$. The expected plateauing duration, once set $\mathcal{S}$ has been reached, is thus $(\lambda \cdot \partial\mathcal{S})^{-1}$. This argument shows that $T^p_{\mathcal{S}}$ is directly related to the structural properties of the contact matrix C, providing a natural connection between the community structure of the contact graph and the progression (or lack of progression) of the opportunistic dissemination process. *This motivates our further investigation of adaptive offloading that re-injects copies whenever the diffusion evolution runs into a plateau.*

3 Re-injecting copies under rare encounters

Overall operation. We consider a publish-subscribe information system, where users subscribe (unsubscribe) to the service upon entering (exiting) the interest area. A small subset of users receives the content through the cellular channel at t_0. The content is disseminated opportunistically to encountered nodes via direct ad hoc communications. Whenever a node receives content from a neighbor, it acknowledges its reception using the infrastructure network, forming a feedback loop in the system. This simple mechanism allows monitoring the evolution of the content dissemination process, consuming at the same time very limited resources as acknowledgments are relatively lightweights compared to the expected size of content messages. The coordinator continually estimates the infection ratio, and may decide to re-inject additional copies of the content in order to boost the diffusion. When the maximum delivery delay D approaches, and the time left is equal to the time P required to send the message through the infrastructure, the system enters into a *panic zone* and pushes the content to all uninfected nodes in order to guarantee total dissemination.

When to re-inject copies ? Re-injection decisions are driven by a derivative-based algorithm. The algorithm needs to keep in memory a short snippet of past infection ratio values. Each content has an associated tracker that stores the evolution of the infection ratio for a temporal sliding window of size W (i.e., at time t the values that will be considered are the ones between $[t-W,t]$). The size of the sliding window trades off how far in time the algorithm looks back and dictates the reactivity to sudden changes in the infection ratio. At evaluation time, the offloading coordinator performs a forward difference quotient on the instantaneous infection ratio $I(t)$ that approximates to a discrete derivative :

$$\Delta_I(t) = \begin{cases} \frac{I(t)-I(t-W)}{W}, & t-t_0 \geq W, \\ \frac{I(t)}{t-t_0}, & t-t_0 < W. \end{cases} \tag{1}$$

Note that $I(t)$ is not monotonically increasing, since nodes may exit the simulation area at any time. $\Delta_I(\cdot)$ approximates the slope of the infection ratio and is one of the parameters that influence the re-injection decision. Re-injection of additional copies of the content occurs whenever the discrete derivative $\Delta_I(\cdot)$ is below a Δ_{lim} threshold computed, on the fly, as the ratio between the fraction of sane nodes and the time remaining before entering the panic zone. This is the reason why a steeper slope is needed when time gets closer to panic zone or the infection ratio lags (different from when we are at the beginning of the infection process). Formally speaking, we have $\Delta_{\text{lim}}(t) = \frac{1-I(t)}{(D-P)-(t-t_0)}$.

As a final step, the injection rate $r_{\text{inj}}(t)$ is computed as a piecewise function, depending on the ratio of the current $\Delta_I(t)$ value and the Δ_{lim} threshold :

$$r_{\text{inj}}(t) = \begin{cases} c, & \Delta_I(t) \leq 0, \\ c\left[1 - \frac{\Delta_I(t)}{\Delta_{\text{lim}}(t)}\right], & 0 < \Delta_I(t) \leq \Delta_{\text{lim}}(t), \\ 0, & \Delta_I(t) > \Delta_{\text{lim}}(t), \end{cases} \tag{2}$$

where $c \in [0,1]$ is a clipping value used to limit the overall amount of re-injected copies in the case of negative values of Δ_I. Finally, $r_{\text{inj}}(t)$, which represents the percentage of uninfected nodes that need to be targeted, is multiplied to the number of uninfected nodes to find the number $\mathcal{R}(t)$ of copies to re-inject at t :

$$\mathcal{R}(t) = \lceil (1-I(t)) \times |N(t)| \times r_{\text{inj}}(t) \rceil, \tag{3}$$

where $|N(t)|$ is the instantaneous number of nodes subscribed to the content update.

4 Evaluation and results

Evaluation scenario. We consider a traffic information service, where a central entity periodically issues a new content to be delivered within a maximum delay D. A single content is active in the system at a time. Possible contents of interest include popular geo-relevant data, such as traffic and roadworks alerts, public utility information, geographic advertising or the distribution of software updates. To evaluate our strategy, we use a large-scale vehicular mobility trace, representing the city of Bologna (Italy), and consisting of $10,333$ nodes [iTE]. In the simulator, we employ a simple contact-based ad hoc MAC model, where a node may transmit only to a single neighbor at a time. Transmission intervals are deterministic, and depend only on the message size and the link data rate. We set cellular downlink and uplink bit rates to 100 KB/s and 10 KB/s, respectively. The bit rate for the ad hoc link is set to 1 MB/s. The size of each content update is set to 100 KB. The size of `ack` messages is 256 bytes, as they carry very little information (content and node identifiers). We heuristically tune the parameters of the algorithm, using $c = 0.05$, and $W = 5$ s in our evaluation. Finally, the panic time duration P is set at 1 s.

Alternative strategies. We compare our algorithm with two baseline strategies : “infrastructure only” (Infra) and “connected-component oracle” (Oracle). In the Infra strategy, there is no offloading at all, and the infrastructure represents the only means of distributing content. In the Oracle strategy, the offloading coordinator has a non-causal picture of the ad hoc connectivity of the entire network (unrealistic but useful to provide an upper bound on performance). We also compare our strategy with Push-and-Track, which relies on static objective functions [WLL+12]. To be as fair as possible, we consider Push-and-Track's objective function that gives the best result for the considered scenario (namely *linear* and *slow start*).

Results. The derivative re-injection strategy performs very well in terms of reduced infrastructure load, saving between 55% and 63% of traffic for different message delays against Infra, as plotted in Fig. 2. In addition, our strategy pays only a small penalty compared to Oracle. It also always obtains better performance than static objective function-based strategies from [WLL+12], thanks to its adaptive behavior. Although Derivative and Oracle show more or less the same trend, this result is achieved through two completely different strategies. On the one hand, Oracle exploits the perfect knowledge of the connectivity status in the network, pushing the content to very specific nodes. On the other hand, Derivative has a much less complete, and slightly out of sync, view of the system, and employs its smart re-injection algorithm to guess *when* additional copies of the content are required. Note also that Oracle presents always larger confidence interval. This is linked to the mobility and turnover of nodes : the resulting connectivity changes in time influencing the Oracle prediction performance. A mobility and connectivity agnostic framework such as ours is less sensitive to this issue.

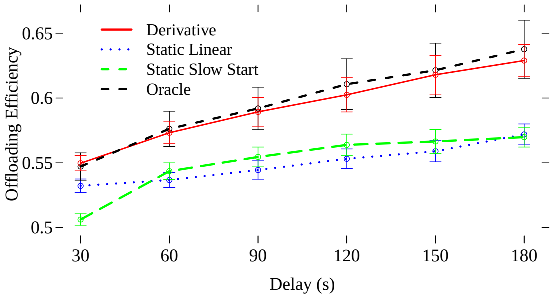


FIGURE 2: Offloading efficiency for the considered scenario. Different maximum reception delays for messages are considered. 95% confidence intervals are plotted.

5 Conclusion

In this paper † we first described the stepwise behavior of the epidemic diffusion in real-life opportunistic networks. To obtain efficient offloading in such a context, we proposed a strategy that adapts to the varying opportunistic dissemination evolution by re-injecting additional copies only when needed. By analyzing this proposal on a large-scale vehicular dataset, we confirmed that it consistently does better than static objective-based offloading strategies, performing very close to an oracle. Future work includes evaluating our strategy on other (synthetic and real) scenarios and implementing the system in a real testbed.

Références

[CMC06] I. Carreras, D. Miorandi, and I. Chlamtac. A framework for opportunistic forwarding in disconnected networks. In *ICST MobiQuitous*, San Jose, CA, USA, July 2006.

[Ind13] Cisco Visual Networking Index. Global mobile data traffic forecast update (2012 – 2017). `http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-481360.pdf`, 2013.

[iTE] FP7 European project iTETRIS : The integrates wireless and traffic platform for real-time road traffic management solutions. `http://www.ict-itetris.eu`.

[TLB+11] P.-U. Tournoux, J. Leguay, F. Benbadis, J. Whitbeck, V. Conan, and M. Dias de Amorim. Density-aware routing in highly dynamic DTNs : The RollerNet case. *IEEE Transactions on Mobile Computing*, 10(12) :1755–1768, December 2011.

[WLL+12] J. Whitbeck, Y. Lopez, J. Leguay, V. Conan, and M. Dias de Amorim. Push-and-track : Saving infrastructure bandwidth through opportunistic forwarding. *Pervasive and Mobile Computing*, 8(5) :682–697, October 2012.

[ZNKT07] X. Zhang, G. Neglia, J. Kurose, and D. Towsley. Performance modeling of epidemic routing. *Computer Networks*, 51(10) :2867–2891, July 2007.

---

†. This work is partially supported by the European Commission in the framework of the FP7 MOTO project (grant 317959).