SCIENCE CHINA
Information Sciences

• LETTER •

April 2017, Vol. 60 049302:1–049302:3
doi: 10.1007/s11432-016-0178-2

General, practical, and accurate models for the performance analysis of cache cascades

Haoqiu HUANG, Lanlan RUI*, Weiwei ZHENG, Danmei NIU & Xuesong QIU

State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876*, China*

Received July 26, 2016; accepted August 12, 2016; published online December 2, 2016

Citation Huang H Q, Rui L L, Zheng W W, et al. General, practical, and accurate models for the performance analysis of cache cascades. Sci China Inf Sci, 2017, 60(4): 049302, doi: 10.1007/s11432-016-0178-2

Dear editor,

Owing to the significant benefits generated by caching in Content Delivery Networks (CDN) and Information-Centric Networking (ICN), the performance analysis of multi-cache systems has received renewed interest by the networking research community [1–5]. To analyze the performance (i.e., cache hit probability), Melazzi et al. [3] proposed an approximate approach that uses a miss stream modeling technique for Least Recently Used (LRU) caches with cascade configurations under renewal traffic. Following the suggestions by Che et al. [2], Melazzi et al.'s approximation considers a cache as a low-pass filter with the cutoff frequency $1/T_{\rm c}$, where $T_{\rm c}$ indicates the cache eviction time and can be approximated with a constant that is independent of the specific item considered. As such, the cache hit probability $P_{\rm hit}^m$ for the item m is trivially given by $P_{\rm hit}^m = P\{t \leqslant T_{\rm c}\}$. Then, $T_{\rm c}$ can be solved by using $\sum_{m=1}^{M} \lambda_m \int_0^{T_{\rm c}} (1 - F_m(t)){\rm d}t = C$, where C represents the storage capacity of a cache, M denotes the total population of items requested by users, λ_m is the average arrival rate of requests at a cache, and $F_m(t)$ indicates the cumulative distribution function of the inter-arrival request time for m.

We remark that the seminal paper of Melazzi et al.'s approximation [3] is missing Leave a Copy Probabilistically (LCP) and Leave a Copy Down (LCD) models for cache cascades. In addition, requests of an item with an inter-arrival request time distribution are essentially gradually filtered out by each cache on the path that the requests traverse, which allows each cache to have a different cache eviction time. As such, the Leave a Copy Everywhere (LCE) strategy, which was previously based on the same cache eviction time for every cache in [3], is also necessary for further investigations.

Contributions. In this letter, we propose general, practical, and accurate models that extend Melazzi et al.'s approximation to analyze the performance of cache cascades. In particular, we analyze a much larger set of practical replication strategies than that considered in Melazzi et al.'s approximation by implementing LCE, LCP, and LCD. We capture the existing state correlations between neighboring caches by considering the effects of the cache eviction time. Each cache can have an individual cache eviction time, which sufficiently reflects current reality.

Models. Let $f_m^i(t)$ be the probability density function (PDF) of the inter-arrival request time, where the request for the item m arrives at the cache i. Similarly, let $f_{\overline{m}}^i(t)$ be the PDF of the

* Corresponding author (email: llrui@bupt.edu.cn)
The authors declare that they have no conflict of interest.

miss stream from the cache i for the item m. In a cascade of caches, each cache is loaded with the miss stream from the previous one, i.e., no other exogenous requests arrive at the cache i except for those from the cache $i-1$. For $i \geqslant 2$, we then obtain $f_{\overline{m}}^{i-1}(t)=f_m^i(t)$. Let $P_{\mathrm{hit}}^m(i)$ be the hit probability for the item m at the cache i. Let T_{c}^i be the cache eviction time for the item m at the cache i.

With regard to LCE, the arrival process of requests for the item m at the first cache is an exogenous renewal process with $f_m^1(t)$. The hit probability for m at the first cache can be solved by using Melazzi et al.'s approximation to give $P_{\mathrm{hit}}^m(1)=\int_0^{T_{\mathrm{c}}^1} f_m^1(t)\mathrm{d}t$. The corresponding $f_m^2(t)$ is then given by $f_m^2(t)=f_{\overline{m}}^1(t)=\sum_{k=0}^{\infty}\{[u(t)-u(t-T_{\mathrm{c}}^1)]f_m^1(t)\}^{*k} * [u(t-T_{\mathrm{c}}^1)f_m^1(t)]$, where $u(t)$ denotes the unit-step function, the operator $*$ indicates convolution, and $\{g(t)\}^{*k}$ describes the k-fold convolution of the (generic) function $g(t)$ with itself. Consider $T_{\mathrm{c}}^2 > T_{\mathrm{c}}^1$. A cache hit for the item m only occurs at the second cache at the time t if at least one request arrives at this cache within the interval $[t-T_{\mathrm{c}}^2, t-T_{\mathrm{c}}^1]$. Then, we obtain $P_{\mathrm{hit}}^m(2)=\int_{T_{\mathrm{c}}^1}^{T_{\mathrm{c}}^2} f_m^2(t)\mathrm{d}t$. If instead $T_{\mathrm{c}}^2 \leqslant T_{\mathrm{c}}^1$, the item m can be stored in the first cache at $\tau_1 \in [t-T_{\mathrm{c}}^1, t)$. This implies that no request for m can arrive at the second cache. Hence, we clearly obtain $P_{\mathrm{hit}}^m(2)=0$. Based on the above, for further generalization to an arbitrary cache i, $i \geqslant 2$, we obtain $f_m^i(t)=f_{\overline{m}}^{i-1}(t)=\sum_{k=0}^{\infty}\{[u(t) -u(t-T_{\mathrm{c}}^{i-1})]f_m^{i-1}(t)\}^{*k} * [u(t-T_{\mathrm{c}}^{i-1})f_m^{i-1}(t)]$ and

$$P_{\mathrm{hit}}^m(i)=\begin{cases} \int_{T_{\mathrm{c}}^{i-1}}^{T_{\mathrm{c}}^i} f_m^i(t)\mathrm{d}t, & T_{\mathrm{c}}^i > T_{\mathrm{c}}^{i-1}, \\ 0, & T_{\mathrm{c}}^i \leqslant T_{\mathrm{c}}^{i-1}. \end{cases} \tag{1}$$

Consider LCP. To compute $P_{\mathrm{hit}}^m(1)$, the last request for m needs to arrive at τ_1 and either produces a cache hit or triggers the probabilistic insertion (with the probability q). Thus, we can write $P_{\mathrm{hit}}^m(1)=[P_{\mathrm{hit}}^m(1)+(1-P_{\mathrm{hit}}^m(1))q]\cdot\int_0^{T_{\mathrm{c}}^1} f_m^1(t)\mathrm{d}t$. Given $T_{\mathrm{c}}^2 > T_{\mathrm{c}}^1$, requests for the item m arrive at the second cache provided that (i) requests arrive at τ_1, in which case the requests that do not trigger the insertion of the item m at the first cache are forwarded to the second cache; and (ii) requests arrive in the interval $[t-T_{\mathrm{c}}^2, t-T_{\mathrm{c}}^1)$. Then, all of these requests result in cache misses and thus are immediately forwarded to the second cache. Instead, for $T_{\mathrm{c}}^2 \leqslant T_{\mathrm{c}}^1$, any forwarded requests can arrive at the second cache if and only if the first cache cannot trigger the insertion of the item m at $\tau_2 \in [t-T_{\mathrm{c}}^2, t)$. Thus, for generalization to an arbitrary cache i, $i \geqslant 2$, we have $f_m^i(t)=f_{\overline{m}}^{i-1}(t)=\sum_{k=0}^{\infty}\{[u(t)-u(t-T_{\mathrm{c}}^{i-1})]f_m^{i-1}(t)P_{\mathrm{hit}}^m(i-1)\}^{*k} * \{u(t-T_{\mathrm{c}}^{i-1})f_m^{i-1}(t)+[u(t)-u(t-T_{\mathrm{c}}^{i-1})]f_m^{i-1}(t)[1-P_{\mathrm{hit}}^m(i-1)]\}$ and

$$P_{\mathrm{hit}}^m(i)=\begin{cases} \dfrac{q\left[\int_{T_{\mathrm{c}}^{i-1}}^{T_{\mathrm{c}}^i} f_m^i(t)\mathrm{d}t+(1-q)\int_0^{T_{\mathrm{c}}^{i-1}} f_m^{i-1}(t)\mathrm{d}t\right]}{1-(1-q)\left[\int_{T_{\mathrm{c}}^{i-1}}^{T_{\mathrm{c}}^i} f_m^i(t)\mathrm{d}t+(1-q)\int_0^{T_{\mathrm{c}}^{i-1}} f_m^{i-1}(t)\mathrm{d}t\right]}, & T_{\mathrm{c}}^i > T_{\mathrm{c}}^{i-1}, \\ \dfrac{q(1-q)\int_0^{T_{\mathrm{c}}^i} f_m^{i-1}(t)\mathrm{d}t}{1-(1-q)^2\int_0^{T_{\mathrm{c}}^i} f_m^{i-1}(t)\mathrm{d}t}, & T_{\mathrm{c}}^i \leqslant T_{\mathrm{c}}^{i-1}. \end{cases} \tag{2}$$

Now, we move to LCD. We first consider the simple case of three caches. When $T_{\mathrm{c}}^2 > T_{\mathrm{c}}^1$, the request for the item m arriving at the second cache at the time t can produce a cache hit, which means that the item m is present in this cache at τ_2. Thus, it is necessary to analyze the previous request for m arriving at this cache. There exist two sufficient and necessary conditions for the previous request: (i) upon arrival of this request, either the item m is already in the second cache, or it is stored in the third cache (and not in the second cache) and thus triggers an insertion of m at the second cache after τ_2; and (ii) either this request with the probability $1-P_{\mathrm{hit}}^m(1)$ arrives at the second cache at τ_1, or it arrives in the time interval $[t-T_{\mathrm{c}}^2, t-T_{\mathrm{c}}^1)$ without producing the cache hit at the first cache. Thus, we easily generalize our approximation to an arbitrary cache i, $2 \leqslant i < l$, where the cache l indicates the last cache in the cache cascades, to obtain $f_m^i(t)=f_{\overline{m}}^{i-1}(t)=\sum_{k=0}^{\infty}\{[u(t)-u(t-T_{\mathrm{c}}^{i-1})]f_m^{i-1}(t)P_{\mathrm{hit}}^m(i-1)\}^{*k} * \{u(t-T_{\mathrm{c}}^{i-1})f_m^{i-1}(t)+[u(t)-u(t-T_{\mathrm{c}}^{i-1})]f_m^{i-1}(t)[1-P_{\mathrm{hit}}^m(i-1)]\}$ and (3). Finally, by resorting to backpropagation analysis, we can use $P_{\mathrm{hit}}^m(l)$ as an initial value to iteratively compute $P_{\mathrm{hit}}^m(i)$. The initial value is then given by $P_{\mathrm{hit}}^m(l)=\int_0^{T_{\mathrm{c}}^l} f_m^1(t)\mathrm{d}t$.

$$P_{\mathrm{hit}}^m(i)=\begin{cases} \dfrac{P_{\mathrm{hit}}^m(i+1)\cdot\left[\int_{T_{\mathrm{c}}^{i-1}}^{T_{\mathrm{c}}^i} f_m^i(t)\mathrm{d}t+(1-P_{\mathrm{hit}}^m(i-1))\cdot\int_0^{T_{\mathrm{c}}^{i-1}} f_m^{i-1}(t)\mathrm{d}t\right]}{1-\left[1-P_{\mathrm{hit}}^m(i+1)\right]\cdot\left[\int_{T_{\mathrm{c}}^{i-1}}^{T_{\mathrm{c}}^i} f_m^i(t)\mathrm{d}t+(1-P_{\mathrm{hit}}^m(i-1))\cdot\int_0^{T_{\mathrm{c}}^{i-1}} f_m^{i-1}(t)\mathrm{d}t\right]}, & T_{\mathrm{c}}^i > T_{\mathrm{c}}^{i-1}, \\ \dfrac{P_{\mathrm{hit}}^m(i+1)\cdot\left[(1-P_{\mathrm{hit}}^m(i-1))\cdot\int_0^{T_{\mathrm{c}}^i} f_m^{i-1}(t)\mathrm{d}t\right]}{1-\left[1-P_{\mathrm{hit}}^m(i+1)\right]\cdot\left[(1-P_{\mathrm{hit}}^m(i-1))\cdot\int_0^{T_{\mathrm{c}}^i} f_m^{i-1}(t)\mathrm{d}t\right]}, & T_{\mathrm{c}}^i \geqslant T_{\mathrm{c}}^{i-1}. \end{cases} \tag{3}$$

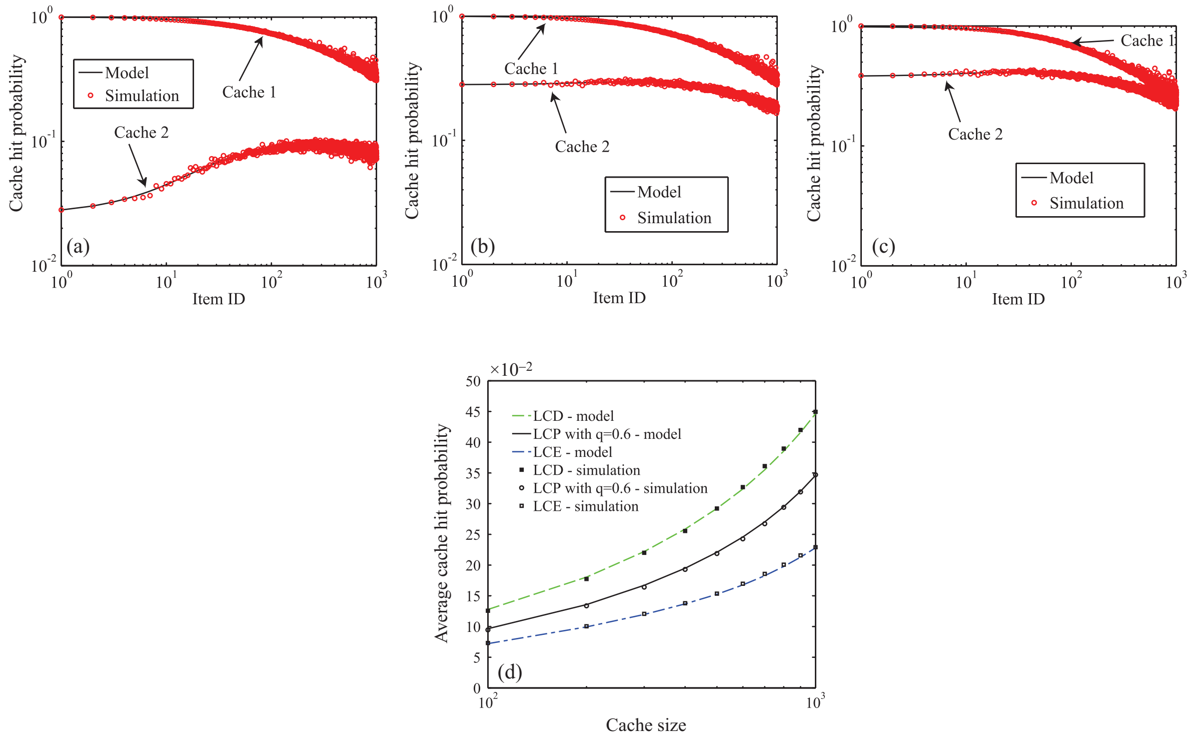

Figure 1 (Color online) Lognormal distribution (CV=4) on cache 1: (a) LCE; (b) LCP with $q = 0.6$; (c) LCD; (d) average cache hit probability vs. cache size.

Numerical results. We considered a chain of six identical caches, a total population of 10^3 content items, a cache of size $C = 100$, a total of 10^7 requests arriving at cache 1, and a Zipf's law slope coefficient of $\alpha = 0.8$. Figure 1(a)–(c) present the per-item cache hit probabilities achieved by LCE, LCP with $q = 0.6$, and LCD on cache 2 for a lognormal distribution (CV $= 4$) on cache 1, respectively. As expected, our models perfectly succeeded in approximating the per-item cache hit probability for the three replication strategies considered. Figure 1(d) compares the performance of the three replication strategies. The analytical predictions and simulation results showed excellent agreement. Triggering the probabilistic insertion (with the probability $q = 0.6$) provided the huge gain of LCP with respect to LCE. This is because it prevents some unpopular items from inserting a cache even if the cache does not produce cache hits for these unpopular items. In addition, LCD significantly outperformed LCP. This is because popular items with multiple requests can be aggregated in the frontend caches to result in a higher average cache hit probability.

Conclusion and future work. In this letter, we show that the performance of cache cascades with practical replication strategies under renewal traffic can be accurately analyzed by extending Melazzi et al.'s original approximation. Still, many extensions of our models are possible, especially for caches with general mesh (arbitrary) configurations. We will leave further investigations to future work.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant Nos. 61302078, 61372108) and Beijing Higher Education Young Elite Teacher Project (Grant No. YETP0476).

References

1 Bianchi G, Detti A, Caponi A, et al. Check before storing: what is the performance price of content integrity verification in LRU caching? ACM SIGCOMM Comput Commun Rev, 2013, 43: 59–67
2 Che H, Tung Y, Wang Z. Hierarchical web caching systems: modeling, design and experimental results. IEEE J Sel Areas Commun, 2002, 20: 1305–1314
3 Melazzi N B, Bianchi G, Caponi A, et al. A general, tractable and accurate model for a cascade of LRU caches. IEEE Commun Lett, 2014, 18: 877–880
4 Huang T, Wang G Q, Fang C, et al. Modeling of miss-probability in content-centric networking. Sci China Inf Sci, 2015, 58: 072301
5 Martina V, Garetto M, Leonardi E. A unified approach to the performance analysis of caching systems. In: Proceedings of the 33rd IEEE International Conference on Computer Communications (INFOCOM), Toronto, 2014. 2040–2048