Adaptive TTL-Based Caching for Content Delivery
Soumya Basu¹, Aditya Sundararajan², Javad Ghaderi³, Sanjay Shakkottai¹, and Ramesh Sitaraman²,4
UT Austin¹, UMass Amherst², Columbia University³, and Akamai Technologies⁴

Content Delivery Networks (CDN)
- CDN delivers millions of requests from content provider to users.
- CDN brings content closer to end users by caching content locally.
- CDN handle heterogeneous and ephemeral contents, e.g., webpage, video.

CDN Objectives and Cache Design
- Primary Objective: Guaranteeing Quality of Service (QoS) to the end users.
  - QoS ⇒ Local Availability ⇒ Cache Hit rate*
- Ambitious Objective: Enabling Pricing Schemes for content providers.
  - QoS + Dedicated Resources ⇒ Local Availability + Dedicated Memory ⇒ Cache Hit rate* + Cache Size*

Challenges in Cache Design
- Content Request Process
  1. Millions of Objects spanning thousands of types
  2. Correlated Arrivals with complex inter-arrival and distribution
  3. Complex and non-independent content popularity
  4. Non-stationary arrivals – (i) one-hot wonders, (ii) Flash Crowds
- Cache Design: Trace-based methods and theoretical methods

Theoretical methods
- Design Parameter: f (Hit rate, size rate, Request process param)
- Evaluation on traces via simulation
  1) Very few results with non-stationarity
  2) Knowledge of Model for closed form
  3) Request parameters high-dimensional
  4) Long traces necessary for accuracy

Overcoming Challenges: Adaptive Caching
- Model-oblivious and Target-driven, Online Adaptation of the parameters
  - Time-to-live (TTL) caches for adaptation with hit-rate guarantees
  - Circumventing non-convexity: Achieving size-rate, not optimizing
  - Higher degrees of freedom for size-rate control: Two level TTL caches.
- Adaptive Filtering: Non-stationary content to limit wastage of size

Dynamic TTL Cache (d-TTL)
- Single Level TTL Cache
  - One ‘TTL’, θ₁ for each type t
  - On Miss, cache with TTL, θ₁
  - On Hit, reset the TTL to θ₁
  - On timer Expiry, evict object
- Single Level TTL Adaptation
  - Hit-rate target for each type t, h₁ < 1
  - Adaptive TTL, θ₁(t) on f₁ request
    - On f₁ request: Type t object, a ∈ (0.5, 1)
      - If Miss, θ₁(t) = θ₁(t - 1) + \frac{1}{2} h₁
      - If hit, θ₁(t) = θ₁(t - 1) + \frac{1}{4} (h₁ - 1)

Caching Terminology
- Meta-data: Id of an Object
  - Type: E.g., data, audio, video
  - Hit(t): Object in cache (i), i=1.2
  - Miss: Object not in (any) cache.
- Virtual Hit: Object not in both cache, but object-id in cache (2).
- Hit rate = \# Cache hit / \# Requests
  - Avg cache size (Gb)
  - Arrival rate (Gbps)

Filtering TTL Cache (f-TTL)
- Two Level TTL Cache
  - Two levels of caches: Cache (1) and (2)
  - ‘TTL pair’, (θ₁, θ₂) for each type t
  - ‘TTL pair’ satisfies θ₂ ≥ θ₁ for all t
  - On Miss, i) Cache object in (2) with timer θ₂
  - On Hit, reset the timer to θ₁
- Two Level TTL Adaptation
  - Hit-rate target for each type t, h₁ < 1
  - Size-rate target for each type t, s₂
  - Adaptive TTL, (θ₁(t), θ₂(t)) on f₁ request.
  - On f₁ request: Type t object, a ∈ (0.5, 1)
    - If Miss or Virtual hit,
      - θ₁(t) = θ₁(t - 1) + \frac{1}{2} h₁
    - If Hit, denote the timer value as θ₂ > 0
      - θ₂(t) = θ₂(t - 1) + \frac{1}{4} (h₂ - 1)

Performance on Akamai Traces
- Akamai trace: Duration. 9 days. #Requests. 504m. #Objects, 25m.
- Targeted hit rate for d-TTL and f-TTL: 40%, 50%, 60%, 70% and 80%.
- Targeted size rate for f-TTL: 50% of size rate of d-TTL.
- ACCURACY: Error in Achieved hit rate: 1.3%, Achieved size rate: 2%.

Adaptation Insights
- The larger the value of θ₁ the higher the hit-rate.
- On cache hit decrease θ₂ value and on cache miss increase.
- The second ‘TTL’, θ₂, enables filtering of rare objects.
- Lower θ₂ + hit-rate target ⇒ Smaller cache (2) but larger cache (1).
- Thumb Rule: Filtering reduces total size under high non-stationarity.

Performance Guarantee
System Model
- Finite no. of types and finite no. of ‘recurrent’ objects of each type.
- Arrival of ‘rare’ objects at a non zero rate, e.g., flash-crowd, one-hot objects
- Content request modeled as Markovian Arrival Process
- Inter arrival time with any absolutely contd. pdf with connected support
- d-TTL* and f-TTL* attains the target hit-rate asymptotically almost surely
- f-TTL* attains size-rate ≤ the target or collapses to θ₂ = 0 a.a.s.

Proof Techniques
- Stochastic approximation technique used for TTL adaptation
- Projected ODE based proof technique to show convergence of d-TTL
- Two timescale Actor-Critic framework for convergence of f-TTL