Distributed optimization of caching devices with geographic constraints

Konstantin Avrachenkov (Inria, FR)
joint work with
Jasper Goseling and Berksan Serbetci (Uni. Twente, NL)

Applied^2 Probability Workshop, Brisbane, 2/7/2019

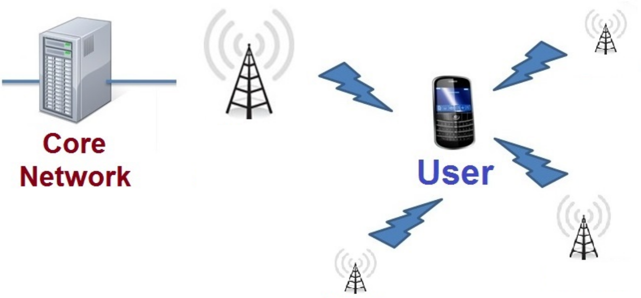
Motivation

In current cellular networks, a user is connected to a single base station (BS).

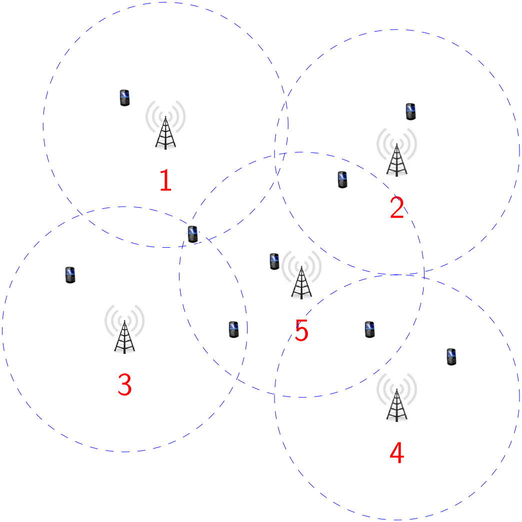
and because of rapid increase in mobile video consumption, the network starts to be saturated.

Motivation

Key elements of potential solution can be (a) simultaneous use of several BSs by same user and (b) availability of cheap memory in base stations to reduce backhaul traffic.

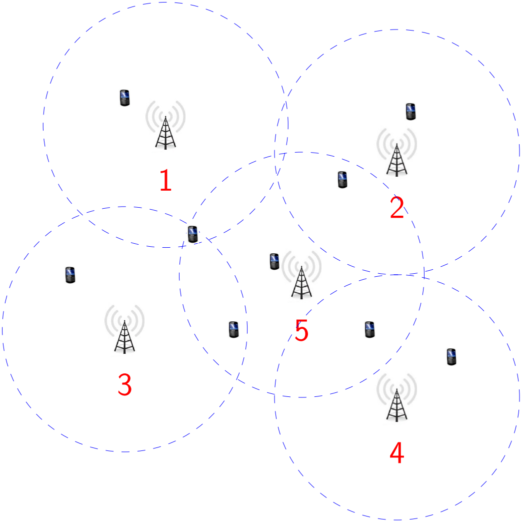


Model



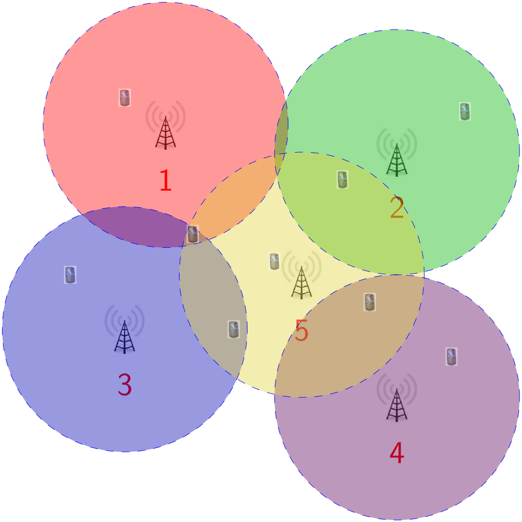
- N BSs

Model



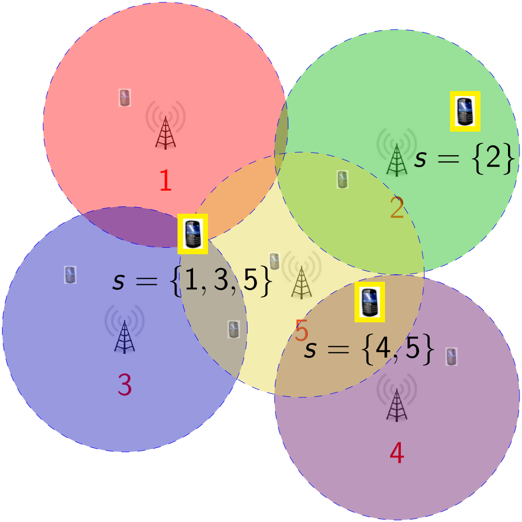
- N BSs
- $\Theta = \mathbb{P}([1:N]) \setminus \emptyset$

Model



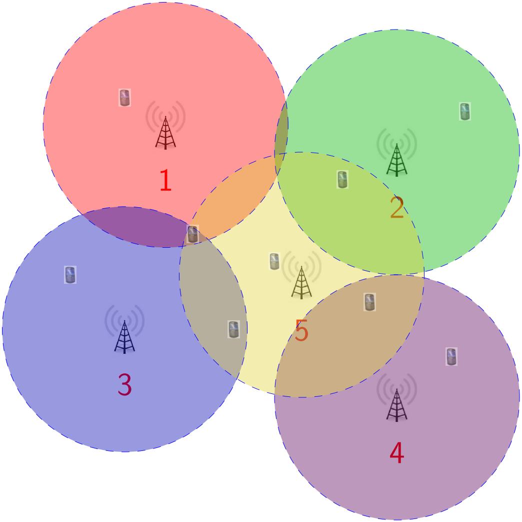
- N BSs
- $\Theta = \mathbb{P}([1:N]) \setminus \emptyset$
- A_s: Area of the plane that is covered only by the caches in subset $s \in \Theta$

Model



- N BSs
- $\Theta = \mathbb{P}([1:N]) \setminus \emptyset$
- A_s: Area of the plane that is covered only by the caches in subset $s \in \Theta$
- $A_{cov} = \cup_{s \in \Theta} A_s$

Model



- N BSs
- $\Theta = \mathbb{P}([1:N]) \setminus \emptyset$
- A_s: Area of the plane that is covered only by the caches in subset $s \in \Theta$
- $A_{cov} = \cup_{s \in \Theta} A_s$
- $p_s = |A_s| / |A_{cov}|$

Content

- Content library $\mathcal{C} = \{c_1, c_2, \ldots, c_J\}$
- Normalized content size 1 (can be relaxed)
- Caches have storage capacity $K \geq 1$
- The content c_j following a popularity distribution a_j

Content

- Content library $\mathcal{C} = \{c_1, c_2, \ldots, c_J\}$
- Normalized content size 1 (can be relaxed)
- Caches have storage capacity $K \geq 1$
- The content c_j following a popularity distribution a_j
- Example: Zipf distribution:

$$a_j = \frac{j^{-\gamma}}{\sum_{j=1}^{J} j^{-\gamma}},$$

where $\gamma \geq 1$.

File Placement

- Content should be placed in caches using knowledge of the request statistics $a_1, \ldots, a_J$.
- The placement policy for cache m:

$$b_j^{(m)} := \begin{cases} 1, & \text{if } c_j \text{ is stored in cache } m, \\ 0, & \text{if } c_j \text{ is not stored in cache } m. \end{cases}$$

- The overall placement strategy for cache m is denoted by $\mathbf{b}^{(m)} = \left[b_1^{(m)}, \ldots, b_J^{(m)}\right]$ as a J-tuple.
- The overall placement strategy for the network is denoted by $\mathbf{B} = \left[\mathbf{b}^{(1)}; \ldots; \mathbf{b}^{(N)}\right]$ as a $J \times N$ matrix.

Performance metric

Total miss probability:

$$f(\mathbf{B}) = \sum_{j=1}^{J} a_j \sum_{s \in \Theta} p_s \prod_{\ell \in s} (1 - b_j^{(\ell)}).$$

Optimization problem

Our goal is to find the optimal placement strategy minimizing the total miss probability as follows:

Problem 1

$$\begin{aligned} &\min \;\; f\left(\mathbf{B}\right) \\ &\textbf{s.t.} \quad b_1^{(m)} + \cdots + b_J^{(m)} = K, \quad b_j^{(m)} \in \{0,1\}, \quad \forall j, m. \end{aligned}$$

Optimization problem

Our goal is to find the optimal placement strategy minimizing the total miss probability as follows:

Problem 1

$$\begin{aligned} &\min \ f(\mathbf{B}) \\ &\textbf{s.t.} \quad b_1^{(m)} + \cdots + b_J^{(m)} = K, \quad b_j^{(m)} \in \{0,1\}, \quad \forall j, m. \end{aligned}$$

$\rightarrow$ Not convex.

ROBR Algorithm

- A distributed asynchronous algorithm;
- Iteratively updating the placement policy at each cache;
- Repeatedly perform best response dynamics;
- Random Order Best Response Algorithm (ROBR):
  At each iteration, a random cache is chosen uniformly and updated.

Best response dynamics

Let $f^{(m)}$ denote the miss probability for a user that is located uniformly at random within the coverage region of cache m, *i.e.*,

$$f^{(m)}(\mathbf{b}^{(m)}, \mathbf{b}^{(-m)}) = \sum_{j=1}^{J} a_j(1 - b_j^{(m)}) \sum_{\substack{s \in \Theta \\ m \in s}} p_s \prod_{\ell \in s \setminus \{m\}} (1 - b_j^{(\ell)})$$
$$= \sum_{j=1}^{J} a_j(1 - b_j^{(m)}) q_m(j),$$

where $\mathbf{b}^{(-m)}$ is the placement policies of all players (caches) except player (cache) m and

$$q_m(j) = \sum_{\substack{s \in \Theta \\ m \in s}} p_s \prod_{\ell \in s \setminus \{m\}} (1 - b_j^{(\ell)}).$$

Structure of the best response dynamics

Each cache tries selfishly to optimize the payoff function $f^{(m)}(\mathbf{b}^{(m)}, \mathbf{b}^{(-m)})$. That is, given a placement $\mathbf{b}^{(-m)}$ by the other caches, cache m solves for $\mathbf{b}^{(m)}$ in

Problem 2

$$\min \quad f^{(m)}\left(\mathbf{b}^{(m)}, \mathbf{b}^{(-m)}\right)$$
$$\textbf{s.t.} \quad b_1^{(m)} + \cdots + b_J^{(m)} = K, \quad b_j^{(m)} \in \{0,1\}, \quad \forall j.$$

Structure of the best response dynamics

Relaxed version of Problem 2 is given by

Problem 3

$$\begin{aligned} \min \quad & f^{(m)}\left(\mathbf{b}^{(m)}, \mathbf{b}^{(-m)}\right) \\ \textbf{s.t.} \quad & b_1^{(m)} + \cdots + b_J^{(m)} = K, \quad b_j^{(m)} \in [0,1], \quad \forall j. \end{aligned}$$

Structure of the best response dynamics

Relaxed version of Problem 2 is given by

Problem 3

$$\min \; f^{(m)}\left(\mathbf{b}^{(m)}, \mathbf{b}^{(-m)}\right)$$
$$\mathbf{s.t.} \quad b_1^{(m)} + \cdots + b_J^{(m)} = K, \quad b_j^{(m)} \in [0,1], \quad \forall j.$$

Lemma 1

Problem 3 is a convex, in fact linear, optimization problem.

Structure of the best response dynamics

Theorem 1

The optimal solution to Problem 3 is given by

$$\bar{b}_j^{(m)} = \begin{cases} 1, & \text{if } \pi_m^{-1}(j) \leq K, \\ 0, & \text{if } \pi_m^{-1}(j) > K, \end{cases}$$

where $\pi_m : [1, J] \to [1, J]$ *satisfies* $a_{\pi_m(1)} q_m(\pi_m(1)) \geq a_{\pi_m(2)} q_m(\pi_m(2)) \geq \cdots \geq a_{\pi_m(J)} q_m(\pi_m(J))$,

Recall that

$$q_m(j) = \sum_{\substack{s \in \Theta \\ m \in s}} p_s \prod_{\ell \in s \setminus \{m\}} (1 - b_j^{(\ell)}).$$

Content placement game

- Each cache continues to optimize its placement strategy until no further improvements can be made, that is until no player can take an advantage from the other players.
- At this point, **B** is a *Nash equilibrium* strategy.
- We will refer to this game as the *content placement game*.
- ROBR algorithm then can be viewed as the *best response dynamics* in the content placement game.

Convergence

Theorem 2

The content placement game is a potential game. Furthermore, if we schedule each cache infinitely often, the best response dynamics converges to a Nash equilibrium in finite time.

Convergence

Theorem 2

The content placement game is a potential game. Furthermore, if we schedule each cache infinitely often, the best response dynamics converges to a Nash equilibrium in finite time.

- Finite number of placement strategies;
- Each non-trivial best response providing a positive improvement in hit probability;
- Updating caches repeatedly guaranteeing convergence to a Nash equilibrium in finite time.

Complexity

Theorem 3

Consider discrete placement of caches, polynomial scaling of file popularities and round-robin scheduling of caches. Then, the best response dynamics of the content placement game converges to a Nash equilibrium in at most $\kappa_1^{-1} N^2 J^{\kappa_2}$ *iterations, with* κ_1 *and* κ_2 *constants.*

Complexity

Theorem 3

Consider discrete placement of caches, polynomial scaling of file popularities and round-robin scheduling of caches. Then, the best response dynamics of the content placement game converges to a Nash equilibrium in at most $\kappa_1^{-1} N^2 J^{\kappa_2}$ iterations, with κ_1 and κ_2 constants.

- The locations of caches:$\subset d\mathbb{Z} = \{(i_1 d, i_2 d) \mid i_1, i_2 \in \mathbb{Z}\}$
- The coverage area: The disk of radius r around the location of the cache

Complexity

Theorem 3

Consider discrete placement of caches, polynomial scaling of file popularities and round-robin scheduling of caches. Then, the best response dynamics of the content placement game converges to a Nash equilibrium in at most $\kappa_1^{-1} N^2 J^{\kappa_2}$ iterations, with κ_1 and κ_2 constants.

- The locations of caches:$\subset d\mathbb{Z} = \{(i_1 d, i_2 d) \mid i_1, i_2 \in \mathbb{Z}\}$
- The coverage area: The disk of radius r around the location of the cache
- In general it is possible that $a_j \to 0$ as $J \to \infty$
- Assumption: $\forall j$, a_j decreases at most polynomially fast in J

Stochastic simulated annealing (SSA)

- Best response dynamics may converge to local optimum at some scenarii;
- We may use SSA approach[1] to guarantee global convergence;
- However, SSA is not often used because convergence is slow.

[1] B. Hajek, "Cooling schedules for optimal annealing ", *Mathematics of operations research*, vol. 13, no. 2, pp. 311-329, May 1988.

Deterministic simulated annealing (DSA)

- An initial τ is set.
- $b_j^{(m)}$ can now take values from the closed set $[\tau, 1-\tau]$ instead of the closed set $[0, 1]$.
- τ is decreased at each iteration.

Resulting placement strategy: An example

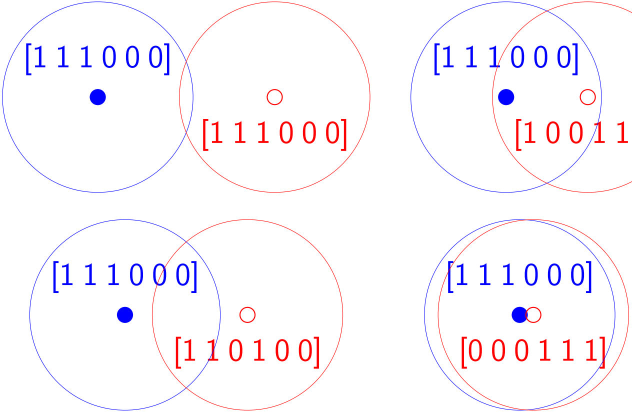


Figure 1: An example of the resulting optimal placement strategies for a small network.

Poisson placement of caches

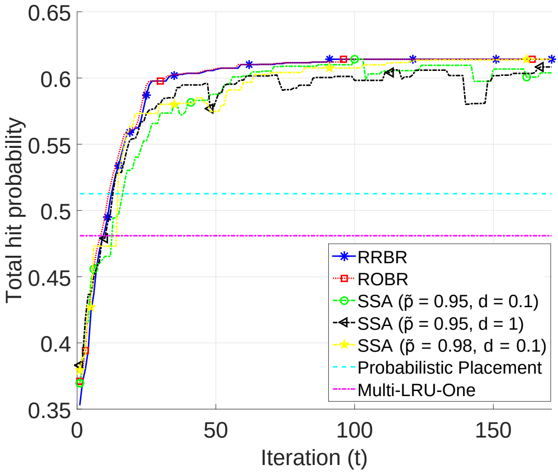


Figure 2: Hit probability evolution ($J = 100, K = 3$).

A real wireless network: Berlin network

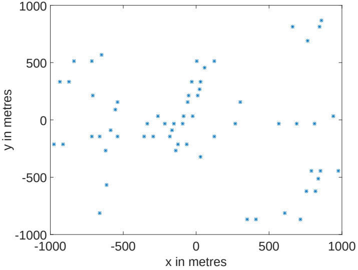


Figure 3: Location of base stations from OpenMobileNetwork dataset.

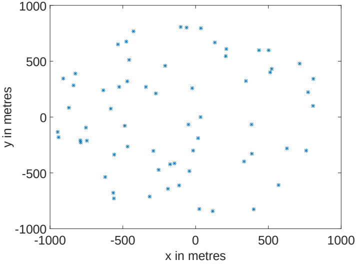


Figure 4: A realization of the Spatial Homogeneous Poisson Process.

A real wireless network: Berlin network

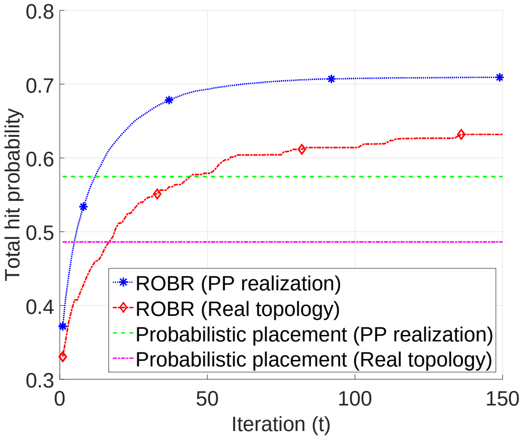


Figure 5: Hit probability evolution ($J = 200, K = 3$).

Discussion & Conclusion

- A low-complexity asynchronously distributed cooperative caching algorithm
- A game theoretic perspective for convergence
- Significantly better hit probability performance
- Often, our algorithm finds the best Nash equilibrium corresponding to the global optimum
- When the algorithm converges to a local optimum, performance gap is very small compared to the global optimum (also a remedial DSA algorithm is provided to escape from the local optimum)
- Future work: Generalization of the model, adapting to dynamic settings, DSA analysis

Thank you!

Details are available in:

Avrachenkov, K., Goseling, J., and Serbetci, B.
A low-complexity approach to distributed cooperative caching with geographic constraints.
Proceedings of ACM Sigmetrics, v.27, 2017.