Proactive caching in high speed scenarios with mmWave communications

Ludovico Frizziero, Alberto Suman, Anthony Dell'Eva

Department of Information Engineering, University of Padova – Via Gradenigo, 6/b, 35131 Padova, Italy

Email: {ludovico.frizziero,alberto.suman,anthony.delleva}@studenti.unipd.it

Abstract—**This is a template for a scientific research paper. The abstract is a super brief summary of what you do in the paper.**

I. INTRODUCTION

The introduction is structured as follows

- What are we talking about: description of the addressed problem.
- Motivation: why the problem is important.
- Novelty: how you contribute to advance the state of the art
- Results: summary of the main findings

II. RELATED WORK

The Related Work section contains an analysis of the most relevant related literature (remarking the shortcomings that are addressed in your work)

III. SYSTEM MODEL

This section describes the model used for the simulation. In particular we discuss about the highway scenario, the wireless network architecture and, finally, the proposed solution for the proactive caching problem.

A. Highway Scenario

In the simulation we utilize a 1 km straight sector of the highway with three lanes for each direction. The sector is represented in [Fig. 1]. For every iteration we consider a single user entering the road in one of the lane which is randomly chosen. The user's expected speed is v_0, while the instanteneous speed varies every $dt = 0.1\,s$ such as

$$v = v_0 + a \cdot dt \quad [m/s] \tag{1}$$

where a indicates the acceleration, which is chosen randomically each second between $a_{min} = -10\,m/s^2$ and $a_{max} = 5\,m/s^2$. The speed the user can have is also bounded to be in the interval $v \in v_0 \pm v_0 \cdot 10\%$. This is a somewhat hard constraint, although not completely unrealistic.

B. Network Architecture

For this paper we used a 5G wireless network architecture to provide high data rate in most coverage areas. The parameters of the channel are given in the table. Base stations (BS) are deployed in each side of the road according to a Poisson distribution with mean λ_{BS}, hence the total expected amount of base station per sector is $2\lambda_{BS}$. The distance from a BS to the next one within the same row is given as

$$\delta_{i,j} = \frac{1000}{\lambda_{BS}} + u \quad [m] \tag{2}$$

Table I: Parameters of the channel

where u is generated by an uniform distibution in the interval $[-500 \cdot \lambda_{BS}^{-1}, 500 \cdot \lambda_{BS}^{-1}]$. This means that usually BSs tend to be uniformly spaced apart, but it can happen that some BS may end up being very close one another.

The number of users for each BS is renewed every second with a value given by a poisson distribution with parameter λ_{UE}. Thus the available bandwidth for the user at the i-th BS is

$$BW_{UE,i} = \frac{BW}{m_i} \tag{3}$$

where m_i indicates the number of users at the i-th BS. Every dt the rate from the i-th BS to the user equipment (UE) is calculated as

$$r_i = BW_{UE,i} \cdot \log_2(1 + SINR) \tag{4}$$

where in the SINR computation we used the path-loss model given by

$$(PL_i)_{dB} = 32.4 + 20\log_{10}(d)_m + 20\log_{10}(f_c)_{MHz} + \Psi \tag{5}$$

with $\Psi \sim \mathcal{N}(0, 6)$ taking into account the shadowing. We consider also an outage threshold, below which it is set $r = 0$.

Furthermore, the BSs and the UE are provided with electronically steerable directional antennas. BSs take advantage of MIMO techniques, as spatial multiplexing and spatial beamforming to improve, respectively, rate and channel gain. The latter is computed from the Channel State Information matrix, which is completely updated every $3 \cdot dt$ seconds, while signal scattering due to subpaths is recomputed every dt.

Finally, the BSs' network drives the handover in an optimal way: the users is forced to connect only to those BS that have memory available for him, in an order induced by the UE's travel direction, and only when it is proper to due so, that is only when the UE is about to enter the support region of the next BS. The concept of support region is introduced later in [eq. 8].

C. User Behavior

The user needs to download a huge file F (such that it can't be downloaded entirely within the premises of the highway sector we consider) from the internet. We consider the file to be compressed, such that to require a minimum rate $\mu_{UE} \in [0.06, 0.15]$ Gbits/s in order to be played back/consumed
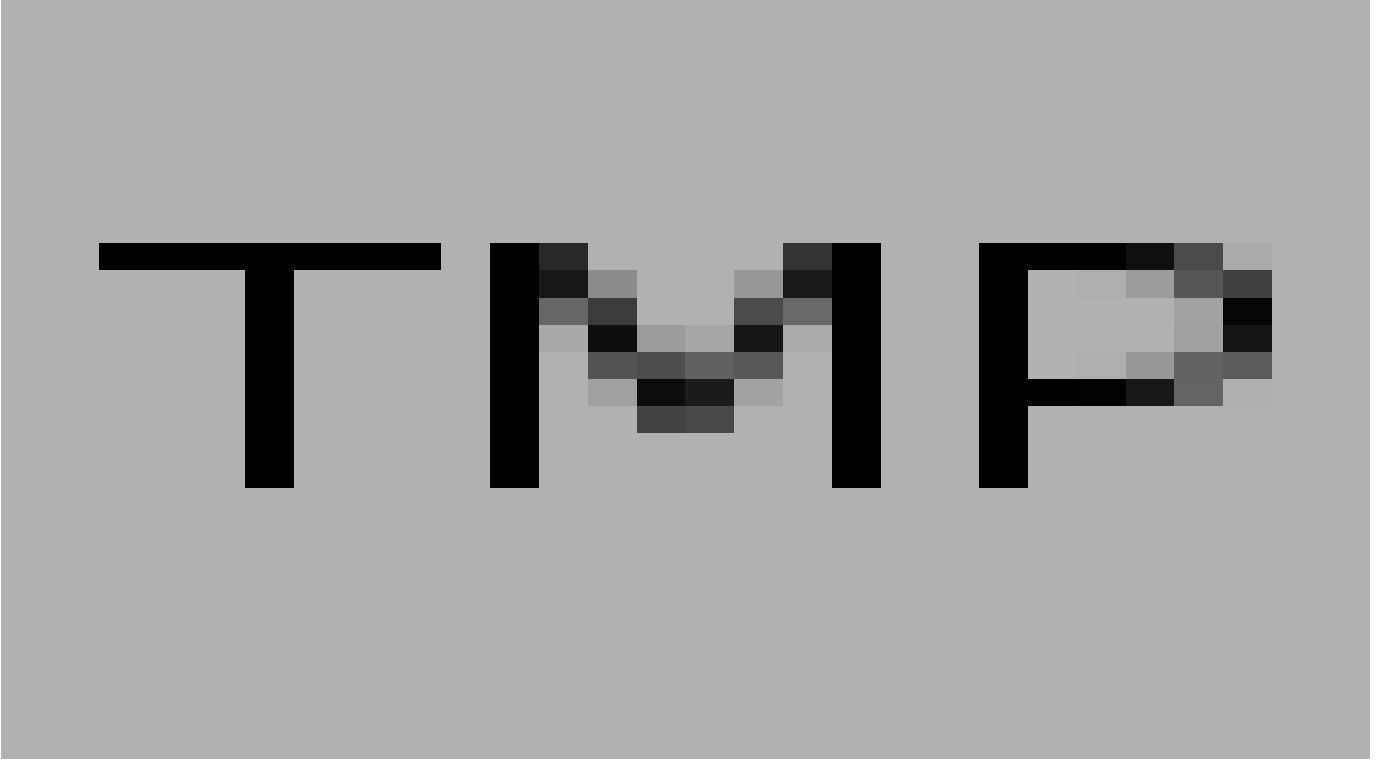
Figure 1: representation of the highway sector.

correctly by the user. The user also has at its disposal a buffer Ω that allows to store 15 seconds of content, that is

$$\Omega = 15 \cdot \mu_{UE} \quad [bits] \tag{6}$$

As soon as the UE is connected to a BS that has memory allocated for him, it starts downloading the available chunk of file with the rate r_i provided by the connection, and to store it in its buffer. There is no flow control, therefore it may happen some portion of the file are lost due to buffer overflow. Lost data are not currently retransmitted. We also consider the download process as a stream of bits rather that packets, since channel errors are not considered. This allows to evaluate for maximum and ideal performances of our system.

IV. PROPOSED SOLUTION

This section describes the problem we want to solve, then it provides the solution, correlated with pseudocode for the solver for the related optimization problem.

A. Problem Statement

We want to achieve as uniform as possible performances of our system across all range of speeds the user can have, regardless of the underlying BS density and memory availability in each of them, while keeping the communication overhead among the central controller (CC), required for coordination, and all the BSs as low as possible.

The overhead is reduced thanks to two main assumption: each highway sector is thought as independent from the others, and therefore each of them has it's own local controller; every BS manages it's memory regardless of the others. When a UE is going to enter the sector, the relative CC is notified.

It then computes an optimal size of memory to allocate to the eventually selected BS based on the actual mean speed of the UE and the density of base stations on the sector, as

$$m = \alpha_{v_0, \lambda_{BS}} \cdot \tau \cdot \mu_{UE} \quad [bits] \tag{7}$$

where $\tau = (1000/\lambda_{BS})/v_0$ is the mean connection time to a BS the UE can experience on the sector, and $\alpha_{v_0, \lambda_{BS}} > 0$ is a scale factor used to tune performances. In particular it is strictly tied to the optimal number of base stations the CC ends up allocating along the highway. Every BS within the premises of the CC must then only report if it has such memory space available or not. This eases communication overhead because the CC doesn't have to keep a record of the memory state for each base station constantly updated.

Now that the CC knows all the candidate BSs, an optimal number among them must be chosen to receive a chunk of the file requested by the UE. This is possible by defining the concept of support of the i-th BS as

$$S_i = g_i(m) \frac{v_0}{\mu_{UE}} \quad [m] \tag{8}$$

that is, for how many meters the UE can travel only relying on its buffer after he downloaded all the memory $g_i(m)$ the i-th BS had allocated for him, before experiencing file playback freezing. This measure, due to v_0, is of course, only in expectation. The support is also thought as centered on its relative BS, but this is not important as long as its starting point is decided with a common rule for each BS. Instead, $g(\mathbf{m})|_i \doteq g_i(m)$, where $\mathbf{m} \in \{0, m\}^N$ is the vector containing the memory report for each BS in the sector, is a function introduced later, that determines how much memory the CC
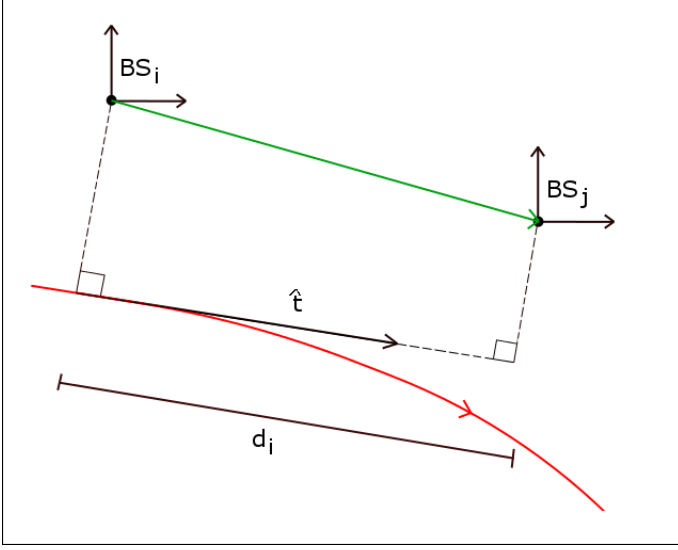
Figure 2: representation of minimum distances from [eq. 9]

eventually allocates to the i-th BS. The optimal number of BSs to choose is then found as $\mathbf{S}^T\mathbf{x} \approx 1000$, where $\mathbf{x} \in \{0,1\}^N$ is the (binary) choice vector outputted by the CC, and $\mathbf{S} \in \mathbb{R}_+^N$ is the vector that contains the support for each BS.

Finally, it is also necessary to evenly distribute the chosen BSs along the sector, to assure buffer overflows are minimized and playback freezing is avoided. To this avail the final optimization problem has a cost term that depends on a vector $\mathbf{d} \in \mathbb{R}_+^N$ which captures the relevant distances among chosen base stations, that is, given $\mathbf{x} \in \{0,1\}^N$, each component is defined as

$$d_i = \min_{j \text{ s.t. } x_j=1} |< pos_w(BS_i) - pos_w(BS_j), \hat{t}_{w,i} >| \quad (9)$$

Here $\hat{t}_{w,i}$ is the unitary norm tangent to the travel direction (predicted along the sector) of the UE in world coordinates, evaluated on the perpendicular of the i-th BS, while $pos_w(BS_i)$ is the position of the i-th BS in world coordinates. A representation is visible in [Fig.2].

B. Optimization Problem

The proposed optimization problem therefore is

OPTIMIZATION PROBLEM

$\mathbf{x} \in \underset{x \in \{0,1\}^N}{argmax} \; \sum_i g_i(m)x_i e^{-\frac{(\mathbf{S}^T\mathbf{x}-1000)^2}{2}} - \lambda \left(\frac{\mathbf{d}^T\mathbf{x}}{2} - 1000 \right)^2$

such that:

$\sum_i g_i(m)x_i \leq F$

$0 < g_i(m) \leq min(m, \Omega) \; \forall i \text{ s.t. } x_i = 1, 0 \text{ otherwise}$

---

where $\lambda > 0$ is a tradeoff parameter between memory allocation and its even distribution along the sector. The function $g(\cdot)$ allows for some flexibility from the CC side in distributing the memory, such that in some BS it may end up being less

than what originally requested if needed for any reason. This is especially useful to avoid having overlapping supports between chosen base stations, because this may cause buffer overflow at the user side. In particular a useful expression for $g(\cdot)$ that minimizes such overlaps, given the almost uniform BSs' deployment assumed in this paper, is

$$g_i(m) = min \left(\frac{1000\mu_{UE}}{v_0||\mathbf{x}||_0}, m \right) \quad (10)$$

because this makes the support S_i become exactly the expected distance among BSs after having observed which one has been chosen, that is

$$S_i = \mathbb{E}[\delta_{i,j}|\mathbf{x}] = \frac{1000}{||\mathbf{x}||_0} \quad (11)$$

The expression $||\cdot||_0$ is the zero norm of a binary vector. The discussion made so far presents the solution that best maximizes our optimization problem given the geometric assumption it relies on. It is nonetheless possible to lessen the uniform constraint over m, or over $g(\cdot)$, but the solution becomes suboptimal.

Finally we provide the pseudocode that solves the optimization problem in [Alg. 1], that runs approximately in $O(N^4)$ due to the update on $\mathbf{d}$. At first it solves the optimization problem by considering the requested amount m for each BS. This way an optimal number of stations is selected, then it applies $g(\cdot)$ in order to minimizes support regions overlapping, while also maximizing the exponential to 1 as a (desired) byproduct.

V. RESULTS

blablablablablablabla

VI. CONCLUSIONS

Conclusions are a superbrief summary of what has been done and highlighting of the "take home message"
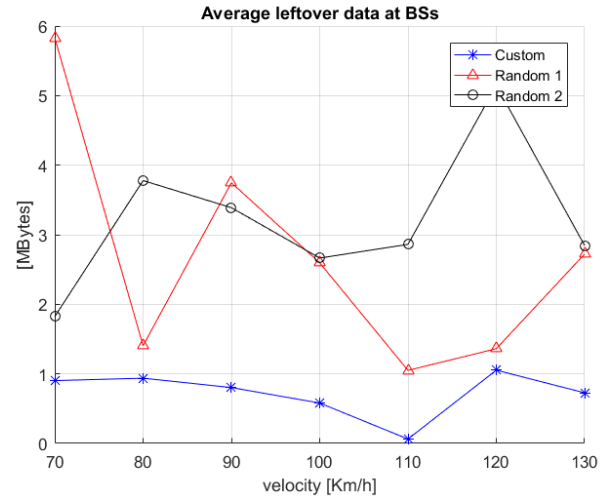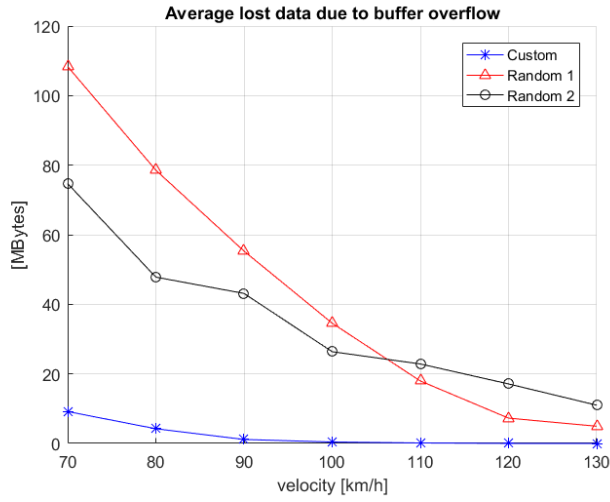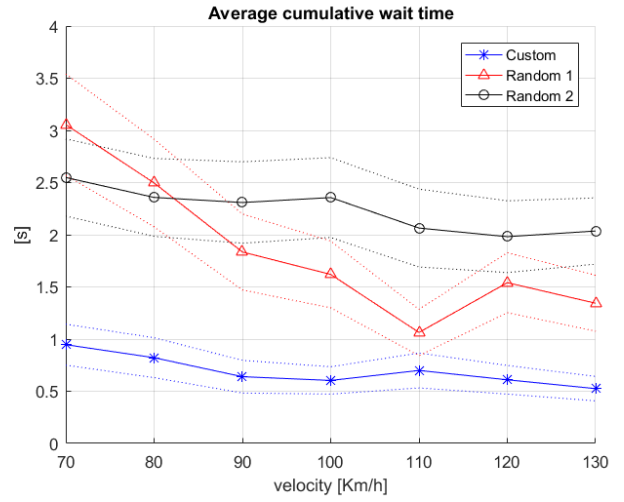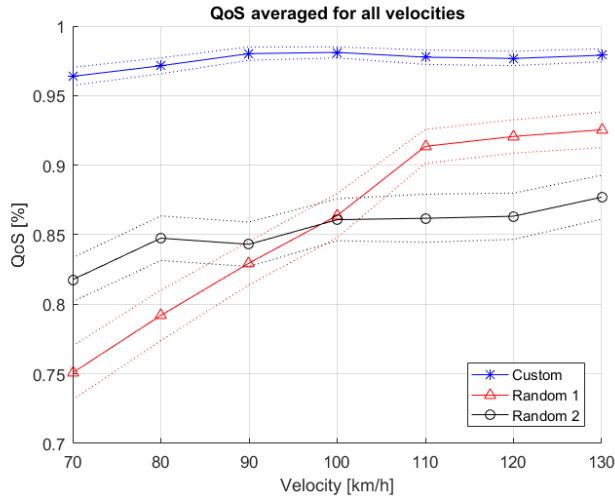
Figure 3: Statistics for a sector with density of 14 BS. Confidence Intervals at 0.05% are shown were relevant.
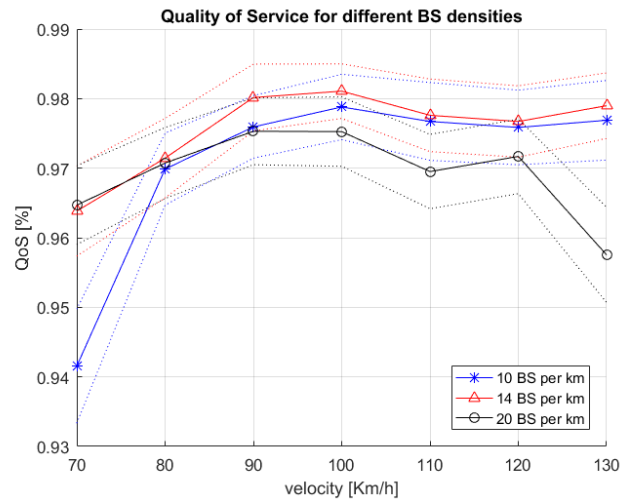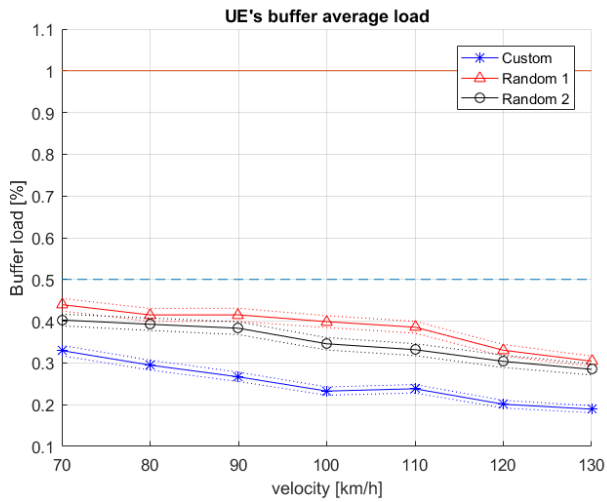

Figure 4: Buffer load at UE side, for 14 BS per sector. Confidence intervals are at 0.05%.

Figure 5: QoS for densities of 10, 14 and 20 BS per sector. Confidence intervals are at 0.05%.

Figure 6: QoS for 20km long journey, 14 BS per sector. Confidence intervals are at 0.05%.

Algorithm 1 pseudocode to solve the optimization problem

Input: $\mathbf{m} \in \{0, m\}^N$, F, positions of each BS with respect to the world
Output: $\mathbf{x} \in \{0, 1\}^N$, $g(\mathbf{m}) \in \{0, g_i(m)\}^N$

$\mathbf{x} \leftarrow \mathbf{0}$, $x_i \leftarrow 1$ for a random i
$\mathbf{S} \leftarrow (\mathbf{m} v_0)/\mu_{UE}$
while $\mathbf{S}^T \mathbf{x} < 1000$ **or** $\mathbf{m}^T \mathbf{x} \leq F$ **do**
   best_i $\leftarrow$ invalid index, best_k $\leftarrow -\infty$
   for $j \in \{1..N\}/\{i : x_i = 1\}$ **do**
      $\mathbf{y} \leftarrow \mathbf{x}$, $y_j \leftarrow 1$
      $\mathbf{d} \leftarrow$ update distances as in [eq. 9] with reference to $\mathbf{y}$
      $k \leftarrow \mathbf{m}^T \mathbf{y} - \lambda \left(\frac{\mathbf{d}^T \mathbf{y}}{2} - 1000 \right)^2$
      if k > best_k **then**
         best_k $\leftarrow k$, best_i $\leftarrow j$
      end if
   end for
   if best_i **not** invalid **then**
      $x_{best\_i} \leftarrow 1$
   else
      CONSTRAINTS NOT MET
   end if
end while
$g(\mathbf{m}) \leftarrow \mathbf{0}$
for all i = {1...N} **do**
   $g_i(m) \leftarrow \frac{1000\mu_{UE}}{v_0||\mathbf{x}||_0}$ if $x_i = 1$, 0 otherwise
end for
return $\mathbf{x}$, $g(\mathbf{m})$