A Priority-Aware MAC Protocol for the Smart City

A Priority-Aware MAC Protocol for the Smart City

Manishika Agaskar
Department of EECS
Massachusetts Institute of Technology
Cambridge, MA, USA
magaskar@mit.edu

Vincent W. S. Chan, *Life Fellow*
Department of EECS
Massachusetts Institute of Technology
Cambridge, MA, USA
chan@mit.edu

***Abstract*—We propose a contention-based priority wireless reservation MAC protocol that guarantees latency and throughput for users with time-critical applications. The protocol is designed to accommodate two anticipated requirements of Smart City networks: first, that critical traffic be served within a specified delay without excessive throttling of non-critical traffic; and second, that surges in critical traffic from a geographically-concentrated region be handled gracefully without expansive overprovisioning.**

***Index Terms*—Smart Cities, Internet of Things, wireless access, medium access control**

I. Introduction

The rapid maturation of the Internet of Things and the advent of the Smart City present an opportunity to revolutionize emergency services as both reactive and preventative. A well-designed Smart City will synthesize data from multiple heterogeneous sensors, relay information to and between emergency responders, and potentially predict and even preempt crises autonomously [1].

The existing body of IoT research illustrates several examples of low-latency requirements for various applications. In the industrial IoT, consequences of excessive packet delay could range from pipe overflows due to failed valve closures to chip malfunctions due to faulty wafer placement on a fabrication line [10]. Abnormal body sensor readings from eHealthcare applications may require a rapid response [6]. Unmanned or remote controlled devices, intelligent transportation systems, and augmented reality systems could have latency requirements measured in single-digit milliseconds [9].

In this paper, we specifically consider the time-critical requirements of the Smart City. We note first that medium and large-scale emergencies in a city will likely result in huge surges of critical traffic from a geographically-concentrated region. These surges must be handled gracefully. The seemingly obvious solution is to significantly overprovision for severe traffic surges and/or throttle the communication of non-priority users and sensors. However, this is costly and disprutive of normal IoT traffic. In an ideal network architecture, priority data would be serviced within a specified delay, while other data would be serviced fairly and not excessively blocked.

In this paper, we focus on the network edge, where wireless access will likely be a throughput bottleneck in the short-term. (In the long-term, we suspect a bottleneck may emerge further upstream, and require modifications at the routing and transport layers.)

We propose a MAC protocol that will guarantee priority users in a particular cell be granted channel access within some time $D_{\max}$, assuming the total number of priority users in the cell does not exceed an upper limit $N_{\max}$. The network monitors the total number of users and priority users in each cell using an adaptive back-off algorithm and, if the number of priority users N_p threatens to exceed $N_{\max}$, reassigns a fraction to different cells.

The MAC protocol supports the following Smart City network requirements and characteristics:

1) Latency and throughput guarantees for multiple critical devices;
2) Fairness amongst all critical users;
3) Heterogeneity of critical devices and applications;
4) Time-varying number of critical devices, that occasionally stresses provisioned network resources;
5) Maintenance of best effort service for non-critical users.

Specifically, we propose a TDMA scheme whereby all priority users are allocated a scheduled data slot every D seconds for the duration of their known priority status. Since the access point/central controller may not know all users that are in a critical state, unscheduled priority users must contend for priority channel access during a short contention period every D seconds. We use cognitive networking techniques to increase the efficiency of our priority access scheme.

II. Related Work

Reservation via contention improves the throughput of the canonical ALOHA contention protocol, as long as the access point has the cognitive capability to schedule transmissions and contending nodes are capable to differentiate between scheduled and unscheduled time slots.

Goodman et al. proposed Packet Reservation Multiple Access (PRMA) in the early 1990s to accommodate both the periodic nature of voice communications and the more random arrivals of data packets [5]. Time is divided into frames that are in turn divided into a constant number of time slots. Prior to each slot, the base station broadcasts whether the slot is "reserved" or "available." Terminals with periodic data requirements (e.g. phone calls) contend for an available slot with fixed probability p; upon successful contention, the base station reserves the slot in all subsequent frames until it is released by the successful terminal. This guarantees a fixed delay for the terminal's periodic traffic after successful capture

of a reservation slot. Terminals with nonperiodic data contend for the same available slots but do not reserve the slot for future use.

Since a collision results in the waste of an entire data slot, Chua et al. proposed subdividing available PRMA slots into "minislots" [3]. In Minislotted PRMA, terminals contend for the first minislot of an available slot. If there is a collision, terminals can contend again at the start of the next minislot (assuming feedback arrives quickly). Terminals contend for minislots until a successful contention or the beginning of the next reserved slot. If a terminal captures the channel prior to the end of the slot, it can use the remaining minislots to transmit data. Neither of these two protocols includes any mechanism for fairness or priority, and neither protocol dynamically adjusts to time-varying traffic patterns.

For industrial wireless sensor networks (WSNs) with strict time-deadline requirements, Yoo et al. proposed dividing "superframes" into contention-access periods and contention-free periods [11]. Nodes contend for access via CSMA/CA, and the central controller separately guarantees slots to specific nodes with periodic traffic. The superframe structure in this protocol is comprised of a fixed number of contention slots and a fixed number of guaranteed slots. The number of guaranteed slots is limited in order to accommodate new devices and to maintain some level of fairness across all users.

Since this protocol does not distinguish between critical and non-critical traffic, Shen et al. proposed PriorityMAC to handle different classes of priority traffic and accommodate the more dynamic nature of time-critical communications in an industrial setting [8]. The lowest priority nodes use TDMA for periodic data updates that can be preempted by higher priority data transmissions. Each TDMA slot is divided into subslots, the first two of which comprise a "High Priority Indication Space." Low-priority nodes must listen to both of these subslots. Nodes with the highest priority send an indication in the first subslot to reserve the slot. The protocol assumes that there is never more than one node in this priority tier at a time. If the first subslot is idle, then nodes of the second highest priority class can send an indication in the second subslot to reserve the channel. If there is a collision in the second subslot, the nodes use an exponential backoff scheme to determine when they next contend. If both subslots are idle, then the regularly scheduled low priority node can transmit.

The assumption of no collisions of the highest priority packets is not realistic in the Smart City, where geographically concentrated emergencies will require time-critical service for multiple devices at once. PR-MAC takes advantage of the spatial correlation between data generated by sensors in a industrial WSN, and therefore only transmits data from a subset of nodes after an event [4]. "Events" are classified based on priority, with different interframe spacing for different event types. At the completion of a data transmission, packets about lower priority events must wait slightly longer before contending for the subsequent frame than packets about higher priority events. Thus, only equal priority nodes can contend simultaneously. In order to minimize collisions, nodes contend via "burst pulses," or non-slotted short pulses that indicate to other users that the node has data to send. Other nodes then backoff some pre-determined amount of time – enough for the successful node to send one data packet. The length of the contention time can be scaled based on the expected number of nodes in each priority class.

PR-MAC and its variants do not address a few key goals of a Smart City network. First, there is no guaranteed fairness amongst equally prioritized nodes and similarly no latency guarantee for specific devices. A node who successfully captures a frame has the same probability of successfully capturing the next frame. This is not considered a flaw in industrial WSNs in which data from one node is reflective of the data from the relatively homogeneous set of other contending nodes. In the Smart City, heterogeneous nodes and applications will have time-critical requirements following an emergency event. Lastly, there is no mechanism in PR-MAC for dynamic resource allocation based on the size of the event or the number of devices with critical data.

We propose a MAC protocol for time-critical Smart City communications that satisfies latency requirements fairly amongst priority nodes for a range of event magnitudes, while also accommodating non-critical (aka best effort) users to the extent possible.

III. Determining User Criticality

Before we can determine how nodes should contend for access, we must consider how the priority status of each node is determined. We assume that all users are noncompromised and acting in good faith. Therefore, we do not account for jamming or otherwise "false" messaging. A realistic implementation would likely require algorithms to detect malicious or malfunctioning nodes and isolate them, e.g. via antenna nulling. There are three general modes by which a node's state can change.

Mode 1: Each node determines and declares its own state. The good faith assumption means that a node will not act selfishly by declaring priority just for better throughput. These priority nodes are discovered by the network via a contention algorithm.

Mode 2: A central controller (perhaps even the access point) determines each node's state. In this case, there would be no need for declaration via contention – the controller would just schedule all traffic and inform each node (via the access point) when or whether to transmit. Discovery of new or mobile nodes would have to be done via the contention algorithm in (1).

Mode 3: Various network elements determine node priority. As edge nodes communicate with different destinations/decision authorities, the destination nodes can either instruct edge nodes to change state (bypassing the access point as a black box and reverting to mode 1) or inform the access point which edge nodes are critical (allowing the access point to unilaterally schedule their transmissions, reverting to mode 2). Note that the latter option would require the destination

node to know enough routing information (e.g. the minimum spanning tree) to know which access point(s) to inform.

These modes differ by when (if) the access point learns the state of a given node. If the access point or central controller knows the identities of all the priority nodes, then priority traffic can be scheduled remotely. If this is not the case, at least some priority nodes must contend for priority access.

In this paper, we consider the case of a wireless multiple-access system with a mix of centralized and distributed control. In particular, a centralized controller schedules priority users but does not know a priori how many or which users have priority requirements. Users independently determine that they have priority data to send and then contend for a reservation. During the contention period, the central controller estimates (and broadcasts) the current number of contending users in order to minimize the expected number of contention slots required.

The Markov chain shown in Figure 1 classifies each user into one of two states: priority or regular. With probability λ, a regular user turns priority during a "cycle" of the MAC protocol and with probability κ, a priority user reverts to regular during the same time. If all network control functions are executed at the start of each of these cycles, then we can ignore the (negligible) probability of a device turning critical and then non-critical (or vice versa) during the same cycle, since the network would never know.

Though the state of each user is independent of its neighbors, we assume that users in a particular geographic area share the same stochastic parameters. In an emergency, these parameters would change for all devices in a particular region. This geographic area could be the size of a single access cell, or it could encompass multiple cells or fractions thereof.

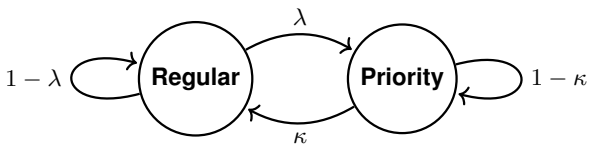

Fig. 1: A two-state Markov model represents the criticality of a node.

Because we assume device independence (given the geographically-based stochastic parameters), we can analyze the state evolution of a single device to understand the evolution of the total number of priority users $N_p(t)$ in a cell of N users. It is trivial to show that as $t \to \infty$,

$$\mathrm{E}[N_p(t)] = \frac{N\lambda}{\lambda + \kappa} \tag{1}$$

$$\mathrm{var}[N_p(t)] = \frac{N\kappa\lambda}{(\lambda + \kappa)^2} \tag{2}$$

IV. TDMA Critical Reservation Scheme

We define a "cycle" as comprised of one contention period, one scheduled priority transmission period, and one best effort transmission period. Any regular users that become priority users in cycle $t-1$ contend during cycle t and (with high probability) also transmit their first scheduled data packet during cycle t. Meanwhile, known priority users with scheduled transmissions during the $t-1$ priority period must explicitly announce during their last priority transmission that they are reverting back to their regular state.

The network controller tracks the number of priority users in the system at any given time, and also keeps a record of the number of priority users in the previous cycle. If the number of priority users per access point grows too large, or if a significant upward trend in priority users is identified, then additional resources need to be deployed to satisfy system requirements. This can be implemented by pushing traffic to adjacent cells, beamforming to access farther away cells for temporary service, or deploying additional relay nodes.

Figure 2 depicts two cycles of the TDMA contention protocol. At the start of cycle t, let $N_p(t)$ be the total number of priority nodes, $N_c(t)$ be the number of contending nodes, and $N_r(t)$ be the number of best effort users with data to send. The length of a priority transmission period is $\frac{N_p(t)L}{C}$, with L the length of a critical data packet and C the channel bit rate. The length of the contention period is $\frac{R_c(t)l}{C}$, where $R_c(t)$ is the number of slots of length l in the contention period.

Assuming any regular users that turn into priority users during a contention period can contend during that period itself, the maximum delay a user could experience between turning critical and transmitting its first data packet would span two data transmission periods and one contention period. This maximum would occur if (1) the node turned critical at the start of a data period, (2) the node were afforded the last reserved slot in the next reserved period, and (3) the total number of priority users were $N_{\max}$, the upper limit of priority transmissions in a cycle.

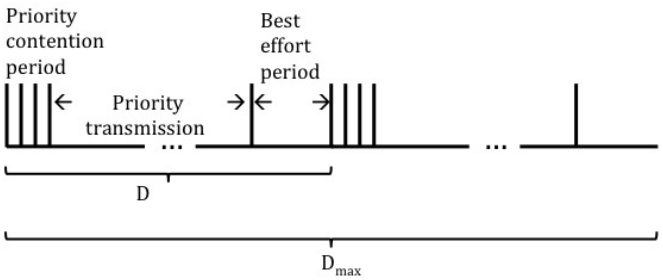

Fig. 2: Diagram of the reservation protocol.

Suppose $D_{\max}$ is the maximum allowed delay for a priority packet. As in Figure 2, we can divide time into cycles of length $D = D_{\max}/2$. Additional resources are needed to satisfy delay requirements if:

$$\frac{N_p(t)L + R_c(t)l}{C} > D \tag{3}$$

A. Protocol Description

We recommend using the well-known pseudo-Bayesian stabilized Aloha protocol by Rivest [7] to dynamically modify the transmission probability of contending users before each contention slot. Each slot is long enough for a short contention packet of length l bits, with enough time for feedback from the access point.

Suppose $N_c(t)$ users turn critical during cycle $t-1$ and contend during cycle t. During each contention slot, each contending user transmits with probability $p = \frac{1}{\hat{n}_c}$, where $\hat{n}_c$ is the estimated number of unscheduled critical users, broadcast by the access point. If $\hat{n}_c \leq 0$, users transmit with probability $p = 1$. The contention protocol stops when a transmit probability of $p = 1$ results in an idle slot. The access point operates as follows:

Algorithm 1 Access Point Contention Protocol

Input: $N_c(t-1)$
Output: $N_c(t)$
  $\hat{n}_c = N_c(t-1)$
  $N_c(t) = 0$
  isContending = True
  while isContending **do**
    broadcast $\hat{n}_c$
    listen
    if SUCCESS **then**
      $\hat{n}_c = \hat{n}_c - 1$
      $N_c(t) = N_c(t) + 1$
    else if COLLISION **then**
      $\hat{n}_c = \hat{n}_c + (e-2)^{-1}$
    else if IDLE **then**
      if $(\hat{n}_c \leq 0)$ **then**
        isContending = False
      end if
      $\hat{n}_c = \hat{n}_c - 1$
    end if
  end while
  return startPriorityTransmission

After all users have successfully contended, the $N_c(t)$ newly scheduled users join existing priority sessions in transmitting during the priority transmission period during their reserved frames. The access point broadcasts the end of the priority transmission period, after which best effort users can attempt to access the network until the next contention period starts at the beginning of cycle $t+1$.

B. Contention Period

In this section, we will show that the required size of the contention period is approximately linear with the number of contending users, so it may not be impractical for all (old and new) priority users to contend every period. We can use this fact to upper bound the length of the contention period.

Rivest analyzed his contention algorithm by assuming Poisson arrivals of packets and approximating the packet backlog as also Poisson-distributed. By contrast, we utilize the algorithm as a scheduling tool where we have a fixed number of packets that have already arrived at the start of the contention algorithm. In this case, the "arrival rate" is zero, but we can still approximate the backlog as Poissonian using the following justification.

At the start of cycle t, $N_c(t)$ is the number of regular users $N_r(t-1)$ that have transitioned to critical in the previous slot. Based on our Markov model for devices, this transition happens with probability λ. Since $N_r(t-1)$ is large and λ is small, the resulting binomial distribution for $N_c(t)$ can be approximated as a Poisson distribution.

We simulated the algorithm under these conditions for varying N_c and $\hat{n}_c$ and show the results across 10000 simulations in Figure 3 below.

Figure 3 shows (across 10000 simulations) the mean and maximum number of slots required for complete reservation when (1) the initial estimate is uniformly set to 0 and (2) there is no initial estimation error. For comparison, we also plot the expected number of slots required if we know a priori the exact number of contending users, $\sum_{n=1}^{N_c} \frac{1}{(1-\frac{1}{n})^{n-1}}$.

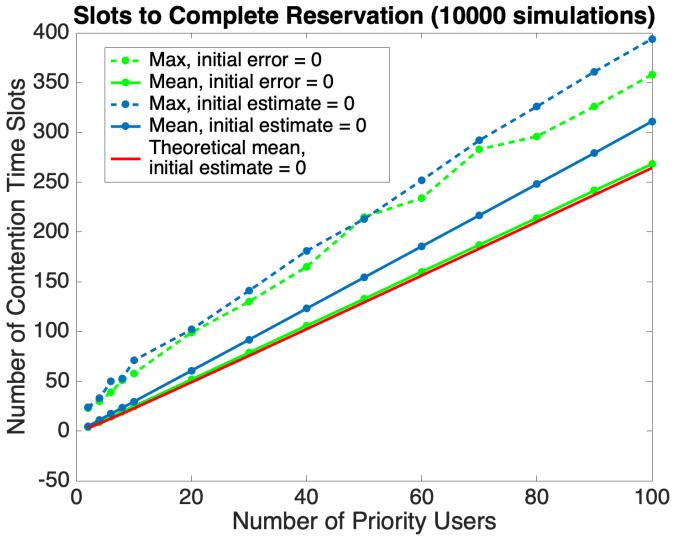

Fig. 3: A linear relationship exists between the number of contending users and the average number of contention time slots required for complete reservation.

The mean number of slots required grows linearly with the number of contending users. For the 10000 simulations of case (1), we find $y = 3.1N_c - 1.6$ with a correlation coefficient of 0.999998. For case (2), we find $y = 2.7N_c - 2$ with a correlation coefficient of 0.999996.

Based on the plots of the maximum number of slots required, we conclude that a linear upper bound of $4n + 50$ would be sufficient as a worst case estimate for the number of slots required for successful reservation of n contending users. In none of more than 10000 simulations did any contending user remain unscheduled within this bound.

If we have an accurate estimator $\hat{n}_c$, then case (2) should be a good approximation of the performance of the contention

algorithm. The upper bound on the number of slots required will help us provision the network to ensure that delay constraints are satisfied, while the expected value can be used to optimize the distribution of regular users across access points.

V. Cognition and Protocol Gain

Our dynamic priority MAC reservation protocol deploys cognitive networking techniques to provide priority network access for critical users. To quantify the value of cognition in wireless access, we compare the cell capacity C required for our priority MAC protocol with that of a slotted ALOHA protocol and a reservation TDMA protocol without class-differentiated contention.

Note that the slotted ALOHA protocol cannot make latency guarantees even if the number of users is known, so we instead consider the expected delay across all packets. The real gain of our protocol should be even higher than the one we compute below, since we neglect to account for the necessary overprovisioning.

In this section, we define q as the packet arrival rate across all nodes and q_p as the packet arrival rate across priority nodes. Since common IoT sensors tend to have data packets of between 1 and 5 kB, we set $L = 20000$ bits. Assuming an address/ID length of 128 bits, commensurate with IPv6, and supposing a 64 bit timestamp, we set $l = 500$ bits. Unless otherwise noted, these are the values used for the simulation results depicted in any figures that follow.

A. No Cognition: Stabilized Slotted ALOHA

Without network cognition, neither the access point nor the nodes themselves have any information about which nodes are critical. Suppose we use a stabilized slotted ALOHA protocol with no other modifications. Since all users have the same probability of success, the long-term average allocated throughput per user should be equal. Note that there is no guarantee of short-term fairness.

We want to provision $C = C_{aloha}$ such that the average per packet delay of all users does not exceed $D_{\max}$. Since the access point cannot schedule traffic in advance, nodes must contend for packet-sized slots. Each slot is $\frac{L \text{ bits}}{C \text{ bits/sec}}$ seconds. The expected per packet delay of stabilized slotted ALOHA (in slots) can be approximated as [2]:

$$\mathrm{E}[W] = \frac{e - \frac{1}{2}}{1 - \lambda e} - \frac{(e^{\lambda} - 1)(e - 1)}{\lambda(1 - (e - 1)(e^{\lambda} - 1))} \tag{4}$$

where λ is the arrival rate of packets in slots^{-1}. We now set $\lambda \approx \frac{qL}{C}$ and write:

$$\begin{aligned} \mathrm{E}[D] &= \left(\frac{L}{C}\right) \mathrm{E}[W] \qquad (5) \\ &= \frac{L}{C}\left(\frac{e - \frac{1}{2}}{1 - \frac{qL}{C}e} - \frac{(e^{\frac{qL}{C}} - 1)(e - 1)}{\frac{qL}{C}(1 - (e - 1)(e^{\frac{qL}{C}} - 1))}\right) \\ &= D_{\max} \text{ when } C = C_{aloha} \end{aligned}$$

We cannot solve this explicitly for C_{aloha}, but we can find a close approximation. There are two values of C that solve the above equation. One of these is too low and results in an unstable $\lambda = \frac{qL}{C} > \frac{1}{e}$. The other solution results in a λ very close to, but slightly less than, $\frac{1}{e}$.

Thus a simple approximation is $C_{aloha} = qLe$.

B. No Cognition: Class-Agnostic Reservation Scheme

Scheduled packet transmission via reservation can achieve latency guarantees and short-term fairness, and in those respects is superior to the unscheduled Aloha scheme. From the network perspective, there is no distinction between critical and non-critical packets – the priority status of nodes may never be determined. Each access point schedules traffic for all users via a contention period using stabilized slotted ALOHA. In the limit of high data frame size L and low contention packet size l, this protocol performs similarly to TDMA.

C. Cognition: Priority Reservation Scheme

In our proposed cognitive MAC protocol, nodes know when they have priority status, and they contend via stabilized slotted ALOHA to reserve priority transmission slots. Recall that each cycle is $D_{\max}/2$ seconds. The expected queue of contending priority users after one cycle ends and at the start of the subsequent contention period is $Q = (q_p D_{\max})/2$. We have established via simulation an upper bound on the number of required contention slots, $R_c < \frac{4q_p D_{\max}}{2} + 50$. Each contention slot is $\frac{l}{C}$ seconds. We want to find the required capacity $C = C_{cog}$ such that the duration of the time to contend combined with the time to transmit is less than the delay bound $D_{\max}$. We assume the worst case here, in which non-critical users do not transmit. We discuss service guarantees for non-critical users in Section VI.

$$\text{Time to contend} = \frac{R_c l}{C} < \frac{2q_p D_{\max} l}{C} + \frac{50l}{C} \tag{6}$$

$$\text{Time to transmit} = \frac{q_p D_{\max}}{2}\frac{L}{C} \tag{7}$$

$$D > \frac{2q_p D_{\max} l}{C} + \frac{50l}{C} + \frac{q_p D_{\max}}{2}\frac{L}{C} \tag{8}$$

$$C > q_p \frac{L}{2} + 2q_p l + \frac{50l}{D_{\max}} = C_{cog} \tag{9}$$

In Figure 4, we vary $D_{\max}$ and plot the gain $G_1 = \frac{C_{aloha}}{C_{cog}}$ of using the priority MAC reservation scheme in lieu of unscheduled ALOHA versus the critical fraction $f = \frac{q_p}{q}$ of the total traffic. The gain G_1 can be approximated as:

$$G_1 = \frac{C_{aloha}}{C_{cog}} \approx \frac{qLe}{q_p\left(\frac{L}{2}+2l\right)+\frac{50l}{D_{\max}}} \tag{10}$$

$$= \frac{2qLD_{\max}e}{q_pD_{\max}L+100l}$$

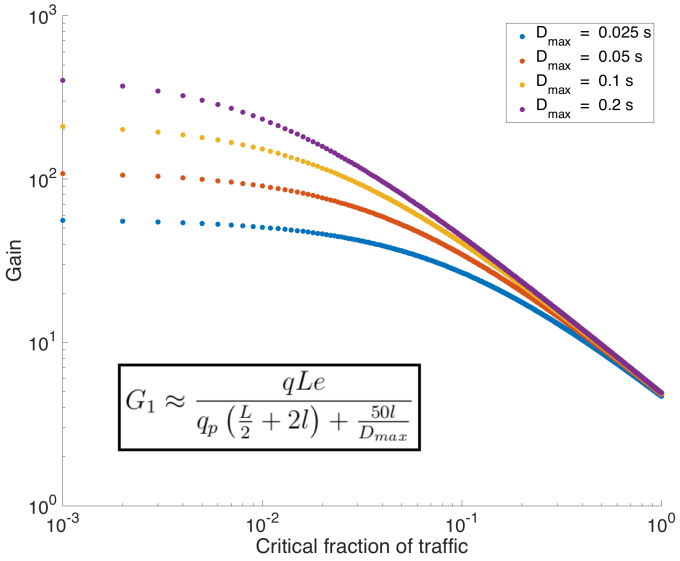

Fig. 4: The critical fraction of traffic is plotted against G_1, the gain of the priority reservation scheme over stabilized slotted ALOHA for various delay bounds $D_{\max}$.

G_1 grows with $D_{\max}$ and L and decreases with $\frac{q_p}{q}$. This indicates that as the critical fraction f of traffic increases, the gain of the reservation scheme over slotted ALOHA decreases. The gain also decreases as the delay requirements become more stringent and as data packets become smaller. If the size of the data packets L increases relative to the reservation slot size l, the gain approaches $\frac{2qe}{q_p}$.

The gain is not independent of q. For $q_p \ll q$, the gain is greater for larger q. As q_p approaches q, if L is sufficiently large, the gain approaches $2e$. Note that this is the same gain as using TDMA (i.e. scheduling) instead of slotted ALOHA (times a factor of 2 due to our protocol's cycle length of $D = D_{\max}/2$).

Suppose G_2 is the gain that results from using priority scheduling instead of the class-agnostic reservation scheme that schedules all users. In other words, G_2 is the gain solely from nodes knowing their priority state.

$$G_2 = \frac{q\left(\frac{L}{2}+2l\right)+\frac{50l}{D_{\max}}}{q_p\left(\frac{L}{2}+2l\right)+\frac{50l}{D_{\max}}} \tag{11}$$

In this case, the gain approaches 1 as q_p approaches q, but approaches $\frac{q}{q_p}$ as $D_{\max}$, q, or L increases, assuming a fixed f. Figure 5 illustrates the gain approaching $\frac{q}{q_p}$ as L increases. Note that $\frac{q}{q_p} = f^{-1}$.

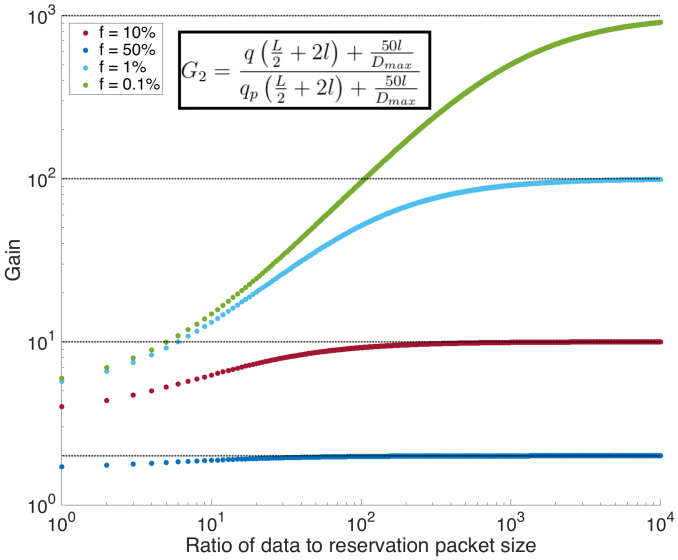

Fig. 5: The ratio of the data packet size L to the reservation slot size l is plotted against G_2, the gain of the priority reservation scheme over a regular reservation scheme.

The main takeaway from our analysis: when the volume of critical traffic is low compared to non-critical traffic, we need significantly less bandwidth to accommodate latency guarantees for critical packets if critical packets are given priority access to the channel. This priority access can be granted via an ALOHA-based contention period for critical users every $D_{\max}/2$ seconds in order to guarantee that newly critical users gain access (i.e. transmit a data frame) within $D_{\max}$ seconds.

VI. Best Effort Service

Critical nodes are known to have high-value data, but non-critical nodes do not necessarily have low-value data; they have data of *unknown* value. If nodes cannot themselves declare criticality, then a controller elsewhere in the network must trigger the regular-to-priority state change. This requires the controller to first receive a data packet that contains the information that triggers the state change. We propose guaranteeing with high probability that regular users get serviced within some time T, comprised of m cycles of length D, so $T = mD$. The following analysis assumes that all sensors are sampling fast enough that they have data to send during every cycle.

Suppose the access point schedules both priority and non-priority transmissions via two separate contention periods. Suppose the total number of contention slots across both periods is fixed and equal $4N + 100$ (based on the upper bound we computed earlier). Since $N_p(t)$ users are guaranteed a slot, the remaining transmission slots are randomly allocated to $N_s(t) = \frac{DC}{L} - N_p(t) - (4N + 100)\frac{l}{L}$ regular users. In each cycle, the number of users that do not receive a slot is a

constant $K = N - (N_p(t) + N_s(t) = N_r(t) - N_s(t)$. We then find the following conditional probabilities:

$$\begin{aligned}&\mathbf{Pr}(\text{Regular user i does not get a data slot}\\&\quad|\ \text{Number of regular users} = N_r(t))\\&\quad= \frac{K}{N_r(t)}\end{aligned} \tag{12}$$

$$\begin{aligned}&\mathbf{Pr}(\text{Regular user i does not get a data slot in m cycles}\\&\quad|\ \{N_r(t), ..., N_r(t+m-1)\})\\&\quad= \frac{\left(N + \frac{(4N+100)l - DC}{L}\right)^m}{\prod_{t'=t}^{t+m-1} N_r(t')}\end{aligned} \tag{13}$$

Given a sequence $\{N_r(t), N_r(t+1), ..., N_r(t+m-1)\}$, we can limit to γ the probability that a regular user never transmits in m cycles, if we have a capacity C_γ where:

$$C_\gamma > \frac{NL}{D} + \frac{(4N+100)l}{D} - \frac{L}{D}\gamma^{\frac{1}{m}}\left(\prod_{t'=t}^{t+m-1} N_r(t')\right)^{\frac{1}{m}} \tag{14}$$

We use the Central Limit Theorem in conjunction with our Markov model of devices to claim that with high probability,

$$\begin{aligned}\left(\prod_{t'=t}^{t+m-1} N_r(t')\right)^{\frac{1}{m}} &> \mathbf{E}[N_r] - 6\sqrt{\text{var}(N_r)}\\&= \frac{\kappa N}{\lambda+\kappa} - 6\sqrt{\frac{\lambda\kappa N}{(\lambda+\kappa)^2}}\end{aligned} \tag{15}$$

The channel rate required to reserve, schedule, and transmit all users in D is:

$$C_{\max} = \frac{NL}{D} + \frac{(4N+100)l}{D} \tag{16}$$

This leads us to an approximate lower bound on C_γ:

$$C_\gamma > C_{\max} - \gamma^{\frac{1}{m}}\frac{L}{D}\left(\frac{\kappa N - 6\sqrt{\lambda\kappa N}}{\lambda+\kappa}\right) \tag{17}$$

Figure 6 shows C_γ for an example cell with $N = 100000$ users, $D = 100$ ms, and $N_p \approx 50$ users ($\lambda = 1.65 \times 10^{-7}$ s^{-1} and $\kappa = 3.33 \times 10^{-4}$ s^{-1}). We see that a significant reduction in required access point bandwidth can be achieved while guaranteeing critical users low-latency access and guaranteeing regular users "not unreasonable" delays, e.g. on the order of 1-2 minutes.

VII. Summary

In this paper, we have analyzed a MAC protocol for time-critical Smart City communications that satisfies the latency requirements of priority users during geographically-concentrated traffic surges, while accommodating non-critical users with best effort service.

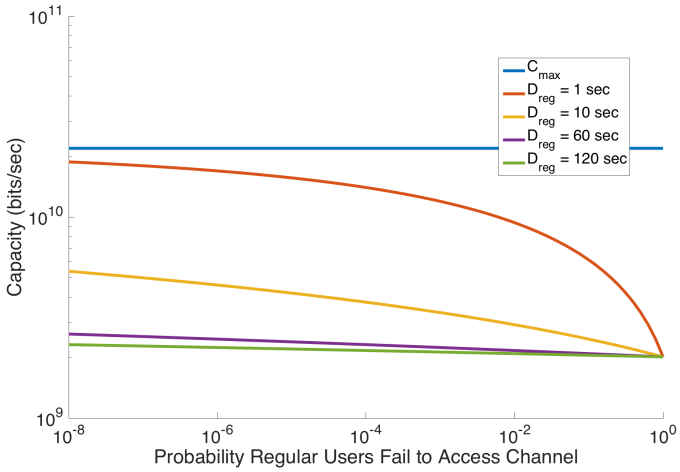

Fig. 6: C_γ is plotted against γ for various $D_{reg} = mD$.

We showed that when the volume of critical traffic is low compared to non-critical traffic, our protocol requires significantly less bandwidth than conventional ALOHA or TDMA protocols. Additionally, this reduction in required access point bandwidth can be achieved while guaranteeing regular users delays no worse than 1-2 minutes.

If a surge in priority users exceeds the provisioned capacity of an access point, further network modifications will be required. At the physical layer, this could include load-balancing between adjacent access points and beamforming to extend the range of devices and access points. Routing and transport layer modifications may also be needed but are not addressed here.

References

[1] M. Agaskar. "Architectural Constructs for Time-Critical Networking in the Smart City". PhD Thesis. Massachusetts Institute of Technology, 2018.

[2] D. P. Bertsekas and R. G. Gallager. *Data Networks*. 1992.

[3] S. T. Chua et al. "Minislotted packet reservation multiple access protocol for integrated voice-data transmission". In: *Singapore ICCS Conference Proceedings*. (Nov. 1994), pp. 320-323.

[4] A. M. Firoze et al. "PR-MAC A Priority Reservation MAC Protocol For Wireless Sensor Networks". In: *2007 International Conference on Electrical Engineering*. (Apr. 2007), pp. 16.

[5] D. J. Goodman et al. "Efficiency of packet reservation multiple access". In: *IEEE Transactions on Vehicular Technology* (Feb. 1991), pp. 170-176.

[6] M. M. Hassan et al. "Resource Provisioning for Cloud-Assisted Body Area Network in a Smart Home Environment". In: *IEEE Access* (2017), pp. 13213-13224.

[7] R. Rivest. "Network control by Bayesian broadcast". In: *IEEE Transactions on Information Theory* (May 1987), pp. 323-328.

[8] W. Shen et al. "PriorityMAC: A Priority-Enhanced MAC Protocol for Critical Traffic in Industrial Wireless Sensor and Actuator Networks". In: *IEEE Transactions on Industrial Informatics* (Feb. 2014), pp. 824-835.

[9] G. Sun et al. "Air-Interface Slice Based Dynamic Resource Reservation for Ultra-Low-Latency IoT Transmissions". In: *IEEE 41st Conference on Local Computer Networks*. (Nov. 2016), pp. 603-606.

[10] H. Yan et al. "Superframe Planning and Access Latency of Slotted MAC for Industrial WSN in IoT Environment". In: *IEEE Transactions on Industrial Informatics* (May 2014), pp. 1242-1251.

[11] S. E. Yoo et al. "Guaranteeing Real-Time Services for Industrial Wireless Sensor Networks With IEEE 802.15.4". In: *IEEE Transactions on Industrial Electronics* (Nov. 2010), pp. 3868-3876.