Improving 802.11p for Delivery of Safety Critical Navigation Information in Robot-to-Robot Communication Networks

Jennifer Gielis and Amanda Prorok
Department of Computer Science and Technology, University of Cambridge
Cambridge, United Kingdom
{jag233, asp45}@cam.ac.uk

Abstract—**Real-world robot networks create hostile radio environments due to their highly dynamic communication topologies and stringent real-time transmission requirements. In this article, we propose improvements to contention based networks that utilize inter-robot navigation information for safer robot motion control in dense multi-robot settings. Towards this end, we introduce a Crash Risk Prioritization (CRP) index, which is employed as the core metric in a scheme of dynamic beacon rate control. Key to our approach is a coupled evaluation setup that allows us to evaluate the impact of network-level changes on the safety of robot navigation. Our results demonstrate that the co-design and simulation of robotic control algorithms and network protocols provide key insights and metrics into the network-level requirements for V2V systems. These insights allow us to make informed conclusions about how such networks can be best served by emerging standards.**

Index Terms—**802.11p, V2V, autonomous robot navigation, multi-robot systems**

I. INTRODUCTION

In settings where airtime is contested, deciding *which robot* should have priority to communicate at *which time* is crucial to delivering safe, crash-free autonomous navigation. Real-world robot networks operate in hostile radio environments [1], due to their highly dynamic communication topologies and stringent real-time transmission requirements. These factors make implementing zero-contention medium access impractical by current methods without extensive infrastructure, such as with cellular systems, though these are currently immature for use with robotic navigation. Without a guarantee of reliable beacon delivery, or mechanisms for losses to be detected, contention-based medium access along with probabilistic delivery estimations must be relied upon.

IEEE 1609 WAVE is a protocol standard currently used for vehicle to vehicle (V2V) networks, which depends upon 802.11p to provide fine-grained, distributed access to the radio medium in a manner that minimizes packet losses due to contention with other robots attempting to send their data, though it has serious flaws for some traffic types that are important in V2V or **robot-to-robot (R2R)** settings (where we use the term R2R to generalize beyond four-wheeled ground vehicles, e.g., drones).

802.11p is the current medium access control (MAC) method used in major V2V standards [2], which supports quality of service (QoS) based priority classes, however there is no mechanism for selecting different QoS priorities for different packets of a uniform class. V2V safety beacons are an example of such a packet class, where the urgency of any individual packet is highly variable based on the situation. We posit that additional metrics must be introduced to control such prioritization.

It is worthy of note that, despite the adoption of 802.11p by higher level standards, LTE-V2V remains a strong competitor in terms of adoption [3], with 802.11p generally providing lower latency and LTE-V2V better packet delivery rates (PDR).

Ultra reliable low latency communications (URLLC) is the class of services made available by the 3GPP 5G specification designed to meet the requirements of R2R and other robotic control applications, and is a possible platform for future LTE-V2V updates. It focuses upon delivering sub 1ms latencies for end-to-end delivery while maintaining packet error rates below 10^{-5}. The mechanisms used to accomplish such impressive figures are numerous and interesting [4]; however we believe that there are particular limitations when measuring the reliability of safety critical networks without reference to the robotic outcomes; notably safety and robotic task performance.

There is a lack of studies that actually employ robot motion control algorithms that are actively using the output of the network that is being simulated; resulting in optimizations that may be optimal for the network but not for autonomous navigation, in addition to failing to detect network failure modes that the robotic control systems are sensitive to. This work contributes towards this literature gap.

In this article, we propose improvements to 802.11p that utilise *inter-robot navigation information* for safe, crash-free robot navigation in dense multi-robot settings. Towards this end, we introduce a **Crash Risk Prioritization (CRP)** index, which integrates proximity and estimated collision times between robots in its computation to provide a concrete estimation of crash risk which is employed as the core metric in a scheme of dynamic beacon rate control, along with finely grained 802.11 parameter tuning over time. Our goal is to demonstrate that a decentralized MAC method can accomplish highly scalable performance that does not compromise the quality of navigation data available to the robot controller, even when a high number of robots are interacting on the network simultaneously in close physical proximity.

Key to our approach is the coupled evaluation setup that allows us to evaluate the impact of network-level changes on vehicular safety, since the effectiveness of the network at delivering safety critical information affects the navigation of robots in the simulation. Our hypothesis is that the codesign of robot control systems and communication protocols is important, as previous works have either disregarded any robotic navigation information, or calculated statistics that are only indirectly connected to crash risk, such as positional error of navigational information.

Our results demonstrate that the proposed prioritization scheme is highly effective at reducing network level contention effects, with more than four times the rate of successful packet delivery versus 802.11p at high robot counts. Additionally, we show that 802.11p results in much higher rates of robot crashes compared to CRP, when using our robotic control scheme.

II. BACKGROUND

This section reviews the background of V2V networking and robot navigation algorithms, as employed in our work, in addition to setting out relationships with newly emerging technologies.

A. URLLC and R2R networks

R2R networks must correct for the problem of medium contention in highly dynamic environments. For packets without strict latency requirements, 5G R2R networks can employ non-URLLC transmission methods that permit high spectral efficiency, contention-free, communications. These cannot be employed in highly decentralised R2R situations because the latency of acquiring radio resources for a particular device-to-device communication can be significant.

URLLC minimises latency by avoiding the requirement to pre-acquire radio resources via the four-step access procedure, instead dropping down to a simple two-step process at minimum, however doing so re-introduces the problems of contention-based medium access. These problems are mitigated in a few ways. First, URLLC uses small (32 byte) packet lengths combined with fine scheduling intervals (mini-slots) that inherently limit the probability of collision. Very small packets also allow URLLC frames to be pro-actively repeated for increased reliability. Secondly, receiver diversity, high bandwidth symbol transmission and various coding schemes can permit the reception of a frame, even when multiple stations have broadcasted simultaneously.

URLLC frames are too small to permit any reasonably secure public-key security scheme on a per-frame basis. This implies that for secure transmissions to take place, some additional communications must occur to initialise a secure state. For an application that is limited in scope, this is not an issue. For any scenario that operates in an open public area, such as V2V applications, security exchanges between nodes may have to occur quite frequently, or rely upon base station provided services for a given area. The implications of a hostile agent broadcasting fake messages that force robots into safety compromising actions should not be underestimated. The scheme introduced in this article assumes the presence of public key security data in every transmitted frame.

It is also not clear how well URLLC communications will scale once there are several hundred participants within reception range. Once sufficient nodes are competing for individual mini-slots that the physical layer improvements are no longer enough to correctly receive multiple competing frames, there is no clear strategy for backing off re-transmissions to prevent continuous, unbounded failures. A simple answer to this is to limit transmission range or to rely upon frequency diversity (or other resource allocation method, a well studied problem) between neighbouring areas, however the extent to which this is possible varies based upon the speed of vehicles. Future applications may have extremely high speed robots interacting in or near high-density areas such that minimum distances for low latency requirements are larger than is optimal for network resource allocation.

802.11p, and therefore the existing standards, depend upon CSMA/CA techniques, which are essentially probabilistic collision avoidance. These are useful as they are computationally simple and can operate with no overhead, but are often limited with regards to flexibility, being somewhat brittle to changes in operating scenarios. The simplicity of probabilistic contention management, along with the widespread deployment of 802.11-style networks, allows us to draw conclusions that are likely to be applicable in real world deployments.

While the general trend of research is moving towards more cellular based solutions over 802.11-like networks, many of the physical layer improvements in the cellular world that are applicable to large scale distributed networks can, and have, also been applied to 802.11. Certainly, some discussion of how to handle high load contention loss scenarios for URLLC is warranted, and research experience from contention-based technologies could be applied.

The work in this article assumes the use of a IEEE 1609 WAVE environment sitting above the 802.11p contention mechanism. This is worthy of mention as several aspects of the communications system are impacted by the limitations of that environment. In particular, safety beacons are sent exclusively on the control channel, which is a 10MHz wide radio channel at approximately 5.9GHz. This control channel is separated into $100ms$ synchronisation intervals (SI), which are further subdivided into the control channel interval (CCHI) and service channel interval (SCHI). Since beacons can only be sent during CCHI, it can be seen that there are severe limitations on the available bandwidth for beacons.

B. Related Methods

In order to reduce contention related losses and delays, this work explores a method that alters both the rate at which safety beacons are generated and EDCA parameters on the basis of a robot's risk of colliding with other objects. Several previous works have explored similar schemes; however, it is rarely demonstrated whether changes to network performance are reflected in actual robot safety, even in cases where metrics are used that are intuitively likely to improve safety. A survey of works involved in beacon management can be found in [5], which also includes a more comprehensive overview of network level issues involved.

Tracking error is one metric that has been previously explored, as in [6] and [7], which show the impact of a proposed rate adaptation mechanisms on the performance of robotic navigation tracking performance, though neither makes an attempt to directly derive the effects on robot safety beyond the assumption that tracking errors of other vehicles is linked to navigational safety. While our results do show a relationship between robot safety and tracking errors, this holds true for robots only at high risk of crashing. Without the coupled simulators for networking and robotics, and the embodied real-world validation, this insight could only be intuited.

C. Navigation in R2R Networks

Controlling a robot on a collision-free path is a well-studied problem in robot navigation, with a large bulk of the work focusing on single-robot path planning algorithms. However, as robot densities and interaction frequencies increase, more sophisticated *cooperative* methods are needed to avoid collisions as well as oscillations [8], [9]. Key to this line of work is the necessity to share navigation information among robots, e.g., velocity and position data, facilitating decentralized control schemes.

To date, a majority of R2R navigation techniques rely on the *velocity obstacle* method, as first introduced in [8]. Reciprocal collision avoidance methods based on velocity obstacles rely on inter-robot communication of velocity and position information to enable the computation of velocity obstacles, which, in turn, are used to compute safe alterations to individual robots' trajectories. The technique has the desirable properties of producing optimal (efficient) paths and guaranteeing crash avoidance, while depending only on decentralized computations.

Even though the inter-robot communication requirements of such reciprocal collision avoidance approaches seem relatively simple, they make several strong assumptions about the communications network used to deliver the necessary information. In particular, there is an assumption of zero delivery delay, perfect reliability and the capacity to deliver information at rates sufficient to ensure distributed lock-step computation, often under full-mesh exchange conditions. No wireless communications standard can even approximate such performance when robot counts grow above non-trivial numbers, or require high-frequency distributed algorithmic execution. Clearly, compromise is required.

Swarming algorithms with similarities to our approach which employ local communications generally assume a fixed radius of communications, or sensing, within which perfect information is available. As will be demonstrated below this is a flawed assumption, at least for the case of *explicit*, data network based, communications. A review of the issues in swarm robotics can be found in [10]. Whilst our approach also depends upon information about nearby robots to operate, the set of robots which contribute to navigational decisions is based both upon collision risk and communications limitations, which are arguably both more effective and realistic for any scenario with a significant number of robots.

III. CRASH RISK PRIORITIZATION FOR 802.11P PACKET SEQUENCING

In order to reduce contention related losses and delays, this work proposes a method called Crash

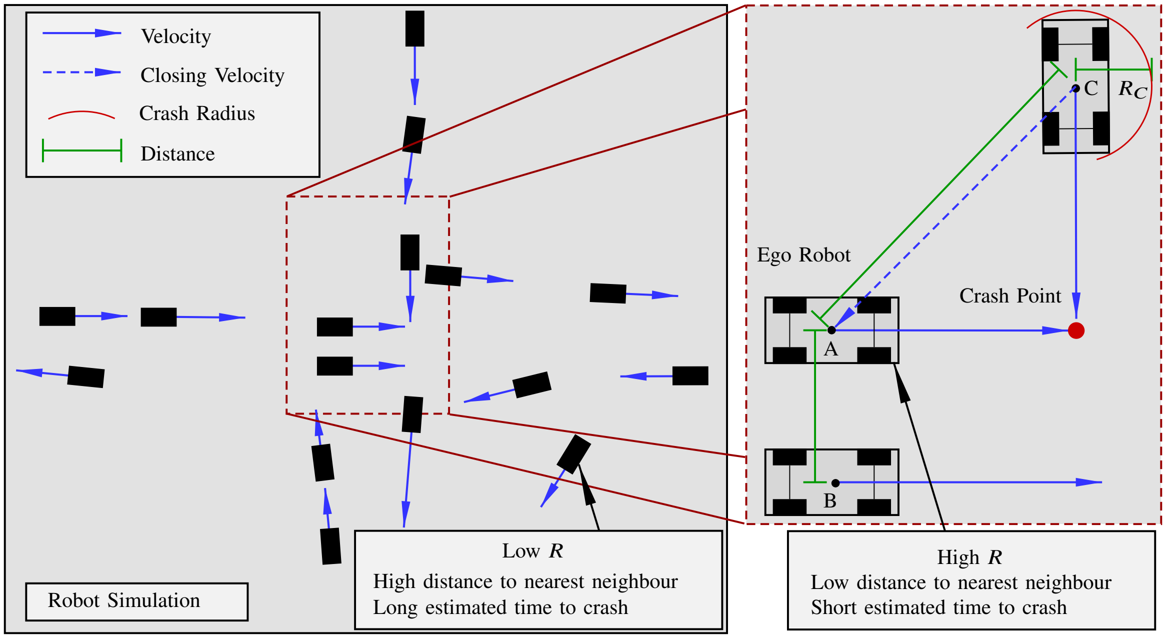


Fig. 1: A representation of the simulation in progress, highlighting the process of deriving R for robot A, given information about the position and velocity of robots B and C. It can be seen that A and C are on course for a collision, while A and B are not though they are closer to each other. Finally R is selected on the basis of the lowest crash time and distance values.

Risk Prioritization which seeks to alter both the rate at which safety beacons are generated and EDCA parameters on the basis of a robot's risk of colliding with other robots. The principle behind this is that robots that are at a high risk of crashing with other robots should receive priority access to the channel.

A. *Implementation Details*

Incorporating both information about *minimum time* until crash and *smallest distance* to another robot are crucial for the estimation of overall crash risk. Simply put, if robots are in close proximity, a sudden maneuver that increases the closing velocity could occur in a small time frame and so the frequency and reliability of beacon delivery are critical. Similar reasoning applies for robots that are distant, but travelling at high closing velocities. The risk index R therefore incorporates both factors in order to maximize robot safety.

1) Computing the risk index: We calculate the value of our risk index, R, by considering it to be 1 minus the minimum of either the lowest time to crash with another robot, or the lowest distance to another robot, for all robots within communication range; higher R values indicate higher risk up to a maximum value of 1. The factors that contribute towards R are normalised between 0 and tunable maximum values representing limits on crash time and distance. The crash time limit is the time interval above which the risk of crashing is no longer considered to reduce and is therefore the minimal priority for transmission. The distance limit operates in the same manner, but for proximity to the nearest robot. The process of computing R is shown visually in Fig. 1 where the ego robot, A, must calculate R given the position and velocity of both B and C within its local navigational database. All safety beacons sent from a particular robot until the next navigational update occurs will share the R value and therefore EDCA parameters.

2) Dynamic EDCA parameter adaptation: CRP accomplishes its central function of prioritization through controlling EDCA parameters, in that robots with high R attempt to transmit sooner within CCHI, and robots with low R wait longer and have much larger numbers of possible transmit slots.

The EDCA parameters that are adjusted by R are the contention window size, CW and the QoS priority class which impacts the length of the AIFS interval. CW is specified as a fixed value rather than a window size because safety beacons are broadcast traffic. Fig. 2 shows both CW and the impact of priority classes in terms of timing generally, but also shows how values of R are used to divide traffic into the four 802.11p defined QoS classes as well as the specific CW values used in each case.

3) Channel interval adjustment: The third value adjusted by CRP is the channel wait interval. The wait interval is a counter that is set immediately after a robot queues a beacon for transmission and it represents the number of CCHI occurrences to wait before sending the next beacon, such that a value of 1 results in transmission in every CCHI. This also means that CRP does not permit a greater than 10Hz transmission rate, which is the *minimum* rate

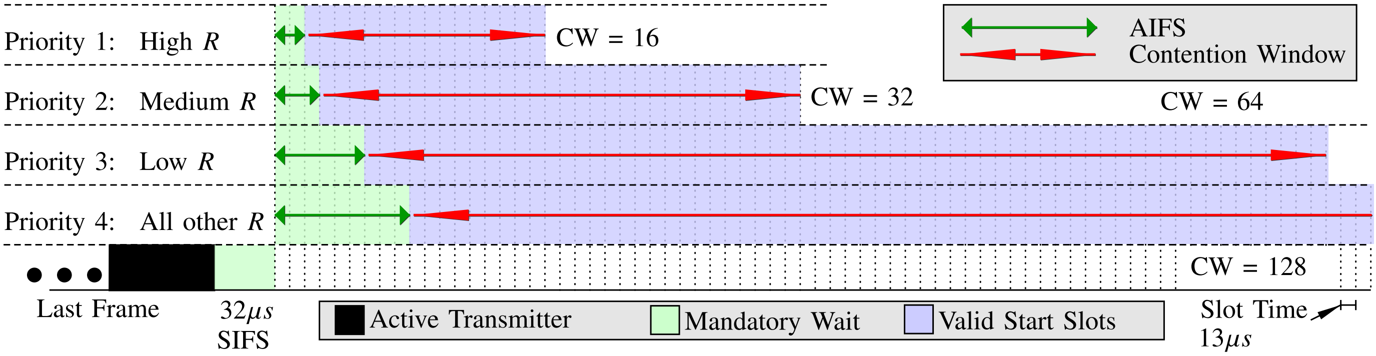


Fig. 2: The specifics of how CRP alters EDCA parameters in response to varying R values. Threshold values for High, Medium and Low priorities are tunable parameters. The general process of selecting CW sizes and AIFS lengths occurs by finding which threshold values R falls into. Higher R values tend towards higher priority, lower AIFS lengths and CW sizes.

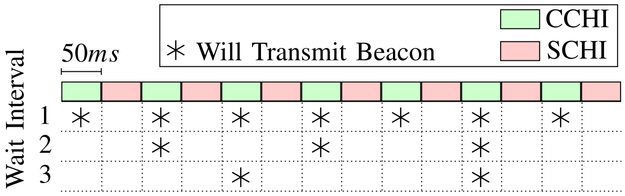


Fig. 3: Shows the effect of wait interval on beacon send rate; increasing values result in larger gaps between beacons.

in IEEE 1609 WAVE. Fig. 3 shows this selection process, showing that an increasing value for the wait interval results in sending a beacon in fewer CCHIs than 802.11p.

The reduction in beacon rate transmission is important to note, as this work aims to demonstrate the counter-intuitive notion that decreasing the frequency at which safety critical information is sent will not *necessarily* negatively impact robotic safety, and may improve it, particularly when combined with the other adjustments made with CRP.

B. Simulation Design

Our experiment consists of a large number of robots that are programmed to navigate a four-way intersection while avoiding inter-robot collisions, for which they are required to exchange navigation information in order to update their motion control. The reader may find it instructive to view a video demonstrating our simulation in action at https://youtu.be/ucDAlxsQ78o. All code written for the simulation is written in MATLAB. We co-simulate the R2R communication network and the reciprocal robot motion control.

1) Timings and Co-simulation Design: The network and the underlying physical medium over which it operates are simulated at $1\mu s$ intervals. This value was chosen as all EDCA timings have it as a common denominator. The robotic simulation steps are processed every $50ms$. Robot motion is not interpolated between robotic steps, meaning that for the purposes of distance calculations at the network level, robots are stationary between robotic ticks.

2) Physical Medium: All wireless transmissions are emulated over a physical medium that accounts for free space path loss between pairs of transmitters and receivers, but otherwise has no noise or losses. As per IEEE 1609 WAVE specifications the simulation assumes that channel 178, which has a bandwidth of 10MHz, is used. All robots transmit at a fixed 33dBm in the simulation, the maximum permissible under the standard for non-emergency use. Finally, a constant transmission rate of 6.5Mbps is employed. We empirically tune the values for QoS class selection, and set them to $R > 0.9$, $R > 0.85$ and $R > 0.7$ for high, medium, and low respectively.

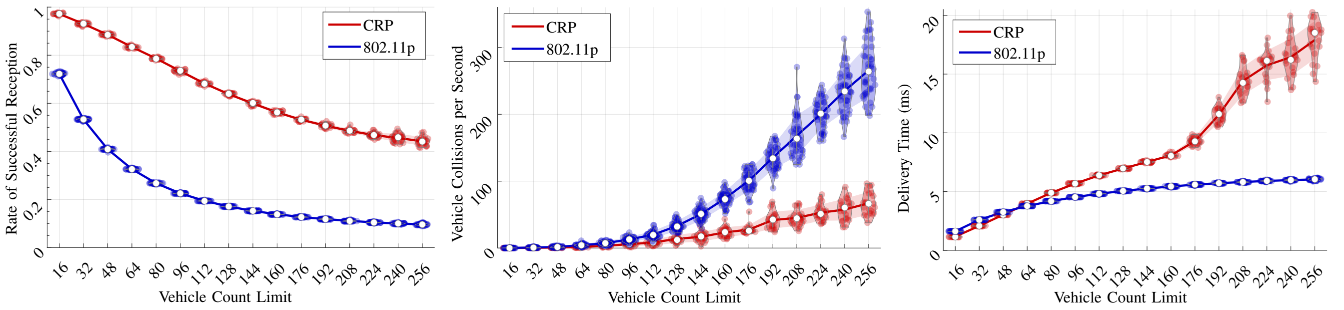
3) Navigation Information: Whenever a safety beacon is successfully decoded by a receiving robot the information contained within, which is the position and velocity of the transmitting robot, is immediately placed within the navigation database of the receiver. This overrides any previous information that the receiver has about the transmitter. This will usually occur in between robotic simulation ticks, though navigation database updates have no impact until the next robotic simulation tick.

In order to reduce the impact of navigational entry aging, any entry within the navigational database that has not been updated via the reception of a beacon since the previous robotic time step is updated by adding the current entry's velocity onto the entry's position, thus effectively treating other robots in the scene as inertial particles with no forces acting upon them.

IV. RESULTS

A. Packet Delivery Rates

Fig. 4a shows the PDR for CRP and 802.11p. Since we report broadcast traffic, each frame may be received many times overall, and each successful reception is counted.



(a) Packet Delivery Rates

(b) Robot Collisions

(c) End-to-end latency

Fig. 4: Results of simulation comparing 802.11p and CRP-modified 802.11p. Panel (a) shows the overall packet delivery rates for safety beacons sent under each scheme, with CRP delivering considerable increases in successful reception. Panel (b) shows the average number of robots that were involved in crashes each second. Panel (c) shows the end-to-end delays, from application send to reception. The shaded area represents the results within 1 standard deviation of the mean and violin bars show the data distribution.

The overall PDRs dropping as low as 10% under standard 802.11p supports the conclusion in [11] that the default settings for 802.11p are inappropriate for pure broadcast traffic when considering only the network level metrics. However, since PDRs drop below 50% at as low as 34 participating robots, there are also clear weaknesses even at light loads. It is likely that applications spreading load across CCHI instead of clustering at the start would substantially improve PDRs at low robot counts, though is not likely to improve higher density scenarios.

CRP accomplishes its gains in PDRs by reducing contention rates, which is primarily effected by the extended CW sizes for lower priority robots (with low R), along with the reduction in packet pet second demand. CRP also generates between 30% and 60% of the number of packets that 802.11p does, which results in a proportionate reduction in contention probability.

B. Navigation Results

Fig. 4b shows the average number of robots involved in crashes per second over the simulation run. In this context, a ‘crash’ is an event where the distance between the centre of two neighboring robots is less than a safety distance, rather than actual robot geometry intersections. No scheme we tested has been capable of delivering a crash-free simulation at high robot counts. This is attributable primarily to the combination of a simple control algorithm, and our use of 2nd order physics simulations for robot movement. For the purposes of these results, the most important property of the control scheme is that it is sensitive to tracking errors.

As Fig. 4b demonstrates, standard 802.11p has significantly worse performance than CRP in terms of robot crashes, reaching a factor of four times more crashes at the largest tested robot counts. This is disproportionate to the differences in overall tracking error, but are somewhat similar to the difference in tracking errors for robots within 40m of each other, which is reasonable given the top speed of robots in the simulation is limited. The fact that close distance tracking error does not trend upwards even at the limits of the simulation suggests that CRP may retain superior performance to 802.11p even at significantly inflated robot densities.

C. Interval between successful beacons

802.11p commonly has extremely large intervals between navigational data updates in the worst cases; with this value at large robot counts (>200) growing to be the bulk of the overall simulation length. Such large intervals between updates for some robot pairs is caused by the fact that, even at small robot counts, 802.11p has very high odds of multiple consecutive contention events. One of the most extreme cases of this was observed, when 129.6 seconds of simulation time passed without a successful one-way update between a specific pair of vehicles. Beacons from the sender must have been lost 2,592 consecutive times in order for sufficient losses to occur for this situation to arise.

Of particular interest is the fact that there is no significant upwards trend in tracking errors for CRP in simulation runs where 256 robots were considered, for robots within 40m. This can be explained by the fact that large spikes of delays between tracking updates occur far less often under CRP, limiting the effect where the worst tracking results pull the average tracking error up.

A larger simulation, with thousands of robots in a bigger test area would be required to explore the limits of this effect; but it is likely to impact URLLC in a similar manner to 802.11p if no mitigation is introduced.

D. End-to-end latency

An interesting effect of CRP is, on average, an increase in the latency of end-to-end delivery. This can be seen in Fig. 4c, and is a result of a trade-off between PDRs and latency. 802.11p appears to perform well here however, because the results do not consider failed packets, this is heavily biased towards packets that were received due to otherwise contending packets being sent from a larger distance away, permitting the SNR of the nearest transmitting neighbour to be high enough for successful decoding. CRP tends towards an average of 25ms, or half of CCHI duration, as it continues to manage low contention losses even when large number of robots are contending for radio resources which evenly spreads successful transmissions over time.

The relatively high latency is not an issue that causes additional robotic crashes when using CRP because the early transmission slots in each CCHI are given to high priority (high R) packets. Such high R packets were typically delivered with lower latency than 802.11p in comparable setups.

V. CONCLUDING REMARKS

We have shown that CRP is an effective scheme for reducing contention related losses in an 802.11p setting, with successful packet delivery varying between an improvement of 1.35x up to nearly 6x, depending upon the number of participating robots. We have simultaneously demonstrated the impact that such improved network behaviour has on our simulated robot group, delivering up to four times fewer crashes. Finally, we demonstrated that tracking error should be evaluated in combination with a simulation that couples network and robotic simulations in order for improvements in network standards to be correctly evaluated, and that latency and packet delivery probability alone are too simple as metrics to correctly infer safe network operations.

If the concepts of this work were applied to 5G LTE-V2V we could elect to transmit only high priority frames using URLLC schemes, while using non-contention based schemes for the remaining traffic. Without network and robotic algorithm co-design and simulation, we would have no solid grounds upon which to base such optimization decisions.

There is a dearth of work that completely evaluates either V2V or R2R networks at large scales with all communications load in place. Before the implementation of such networks in the real world, comprehensive evaluations in simulators that are as true to life as is possible must be conducted.

ACKNOWLEDGMENTS

A. Prorok was supported by the Engineering and Physical Sciences Research Council (grant EP/S015493/1).

REFERENCES

[1] T. Blazek, G. Ghiaasi, C. Backfrieder, G. Ostermayer, and C. F. Mecklenbräuker, “Ieee 802.11p performance for vehicle-to-anything connectivity in urban interference channels,” in *12th European Conference on Antennas and Propagation (EuCAP 2018)*, 2018, pp. 1–5.

[2] “Ieee standard for wireless access in vehicular environments (wave) – multi-channel operation,” *IEEE Std 1609.4-2016 (Revision of IEEE Std 1609.4-2010)*, pp. 1–94, 2016.

[3] F. Arena, G. Pau, and a. Severino, “A review on ieee 802.11p for intelligent transportation systems,” *Journal of Sensor and Actuator Networks*, vol. 9, p. 22, 04 2020.

[4] P. Popovski, C. Stefanović, J. J. Nielsen, E. de Carvalho, M. Angjelichinoski, K. F. Trillingsgaard, and A. Bana, “Wireless access in ultra-reliable low-latency communication (urllc),” *IEEE Transactions on Communications*, vol. 67, no. 8, pp. 5783–5801, 2019.

[5] S. A. A. Shah, E. Ahmed, F. Xia, A. Karim, M. Shiraz, and R. M. Noor, “Adaptive beaconing approaches for vehicular ad hoc networks: A survey,” *IEEE Systems Journal*, vol. 12, no. 2, pp. 1263–1277, 2018.

[6] C. Huang, Y. P. Fallah, R. Sengupta, and H. Krishnan, “Information dissemination control for cooperative active safety applications in vehicular ad-hoc networks,” in *GLOBECOM 2009 - 2009 IEEE Global Telecommunications Conference*, 2009, pp. 1–6.

[7] R. K. Schmidt, T. Leinmuller, E. Schoch, F. Kargl, and G. Schafer, “Exploration of adaptive beaconing for efficient intervehicle safety communication,” *IEEE Network*, vol. 24, no. 1, pp. 14–19, 2010.

[8] J. Van Den Berg, S. J. Guy, M. Lin, and D. Manocha, “Reciprocal n-body collision avoidance,” in *Robotics research*. Springer, 2011, pp. 3–19.

[9] J. Alonso-Mora, A. Breitenmoser, M. Rufli, P. Beardsley, and R. Siegwart, “Optimal reciprocal collision avoidance for multiple non-holonomic robots,” in *Distributed autonomous robotic systems*. Springer, 2013, pp. 203–216.

[10] N. Nedjah and L. S. Junior, “Review of methodologies and tasks in swarm robotics towards standardization,” *Swarm and Evolutionary Computation*, vol. 50, p. 100565, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2210650217308398

[11] R. Stanica, E. Chaput, and A. Beylot, “Enhancements of ieee 802.11p protocol for access control on a vanet control channel,” in *2011 IEEE International Conference on Communications (ICC)*, 2011, pp. 1–5.

BIOGRAPHIES

Jennifer Gielis received her GDipIT, MIT and MRes degrees in Computer Science at Macquarie University, Australia. She is currently a graduate student at the University of Cambridge, UK.

Amanda Prorok received her BSc, MSc and PhD degrees in Computer Science at the Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland. From 2015-2017, she was a postdoctoral researcher at the General Robotics, Automation, Sensing and Perception (GRASP) Laboratory at the University of Pennsylvania, USA. She is currently a University Lecturer (Assistant Professor) at the University of Cambridge, UK.

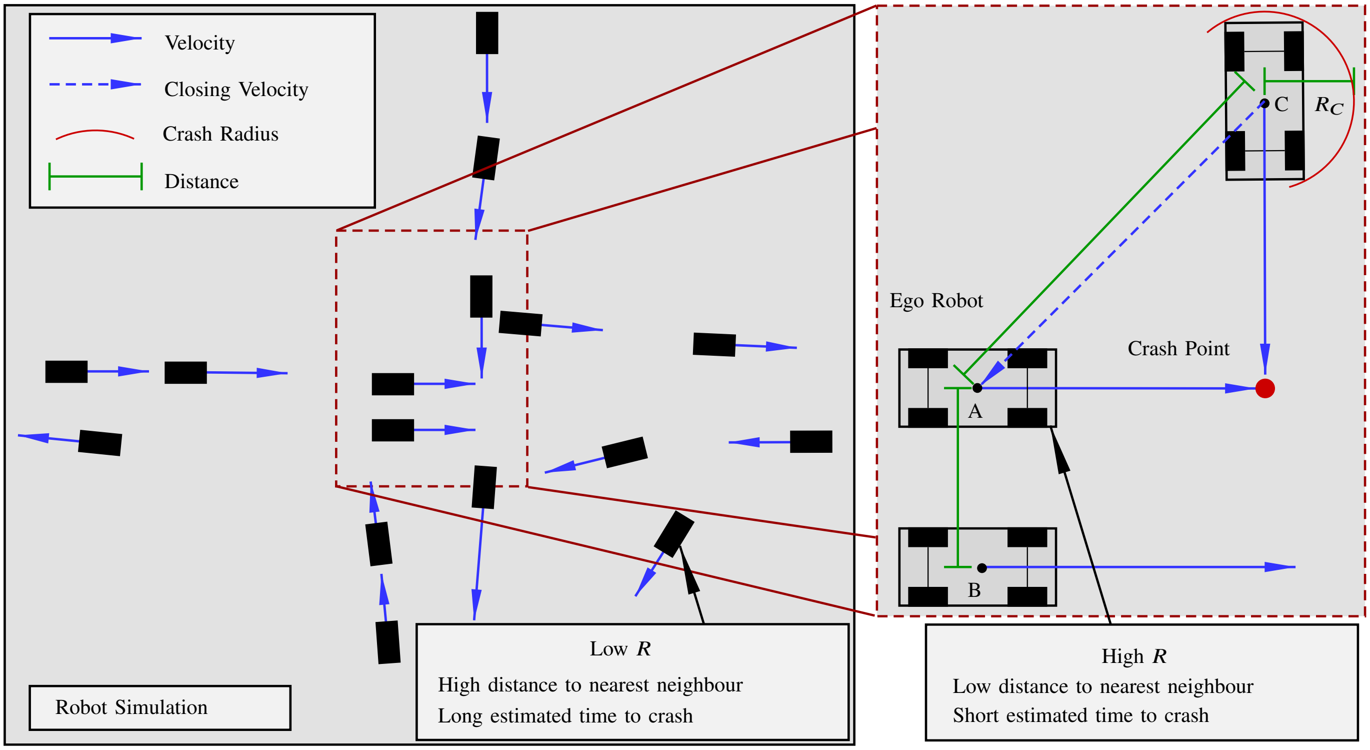


Fig. 5: A representation of the simulation in progress, highlighting the process of deriving R for robot A, given information about the position and velocity of robots B and C. It can be seen that A and C are on course for a collision, while A and B are not though they are closer to each other. Finally R is selected on the basis of the lowest crash time and distance values.

1

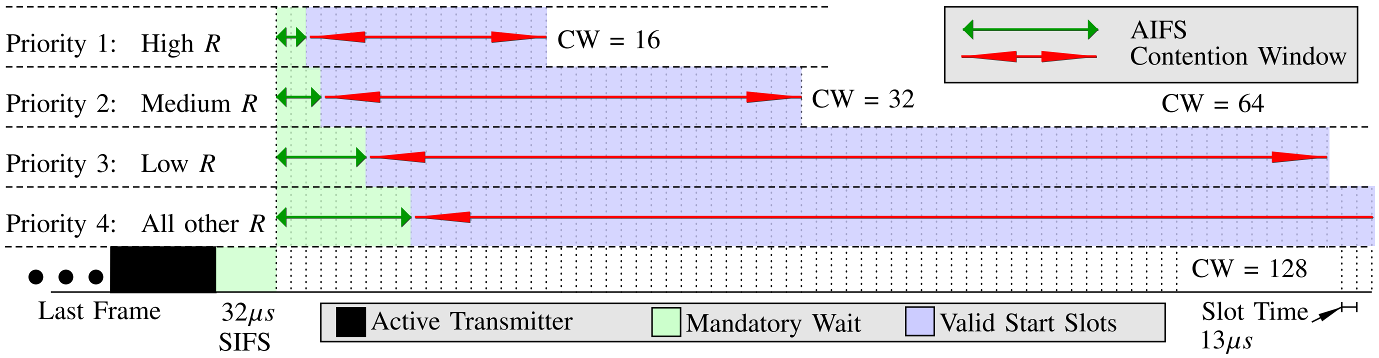


Fig. 6: The specifics of how CRP alters EDCA parameters in response to varying R values. Threshold values for High, Medium and Low priorities are tunable parameters. The general process of selecting CW sizes and AIFS lengths occurs by finding which threshold values R falls into. Higher R values tend towards higher priority, lower AIFS lengths and CW sizes.

2

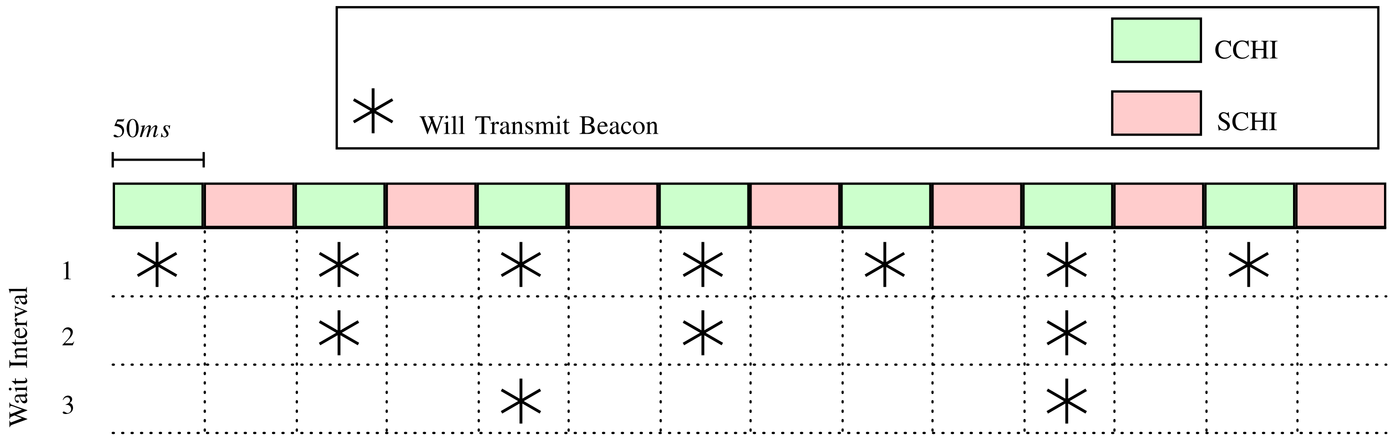


Fig. 7: Shows the effect of wait interval on beacon send rate; increasing values result in larger gaps between beacons.

3

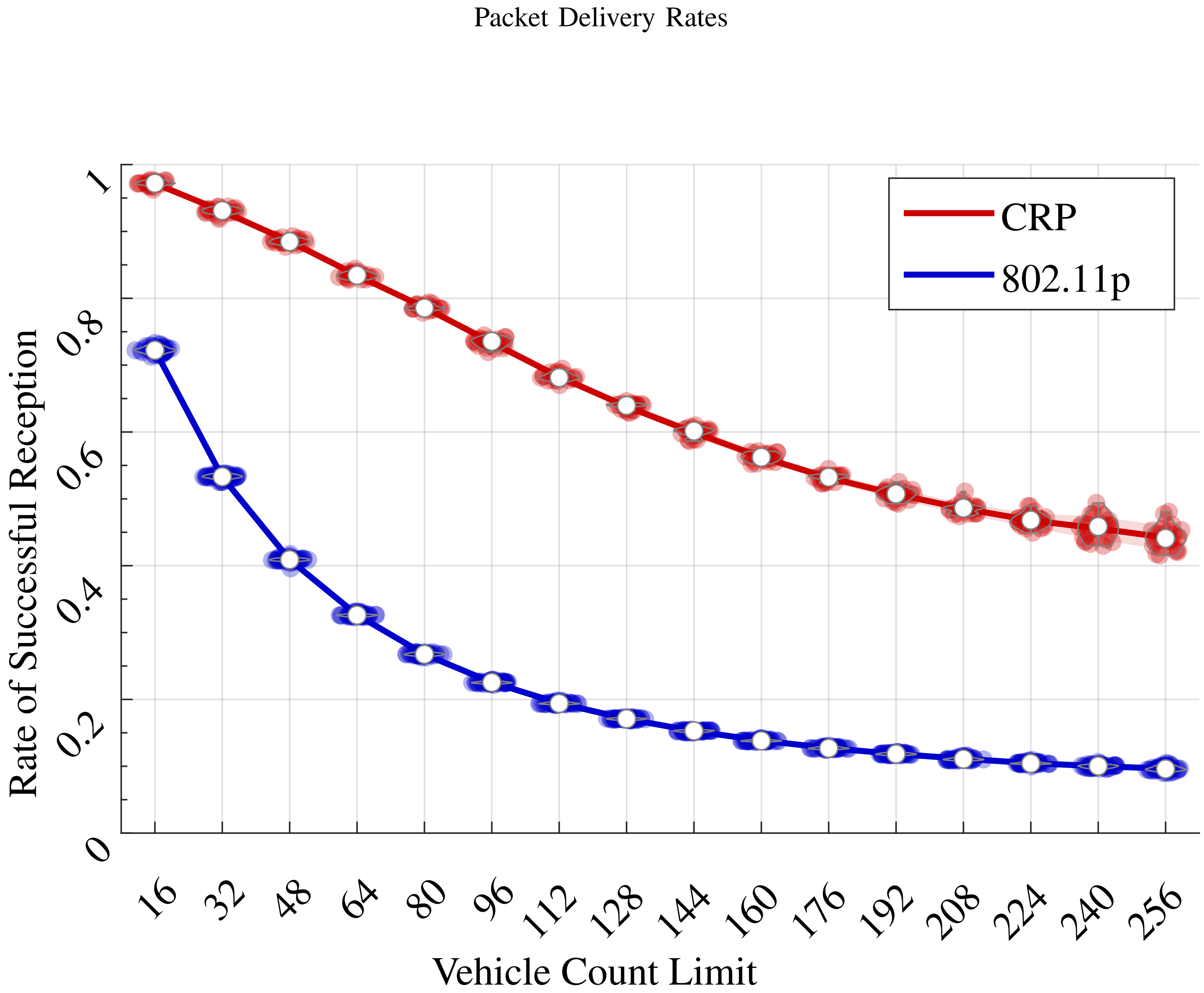
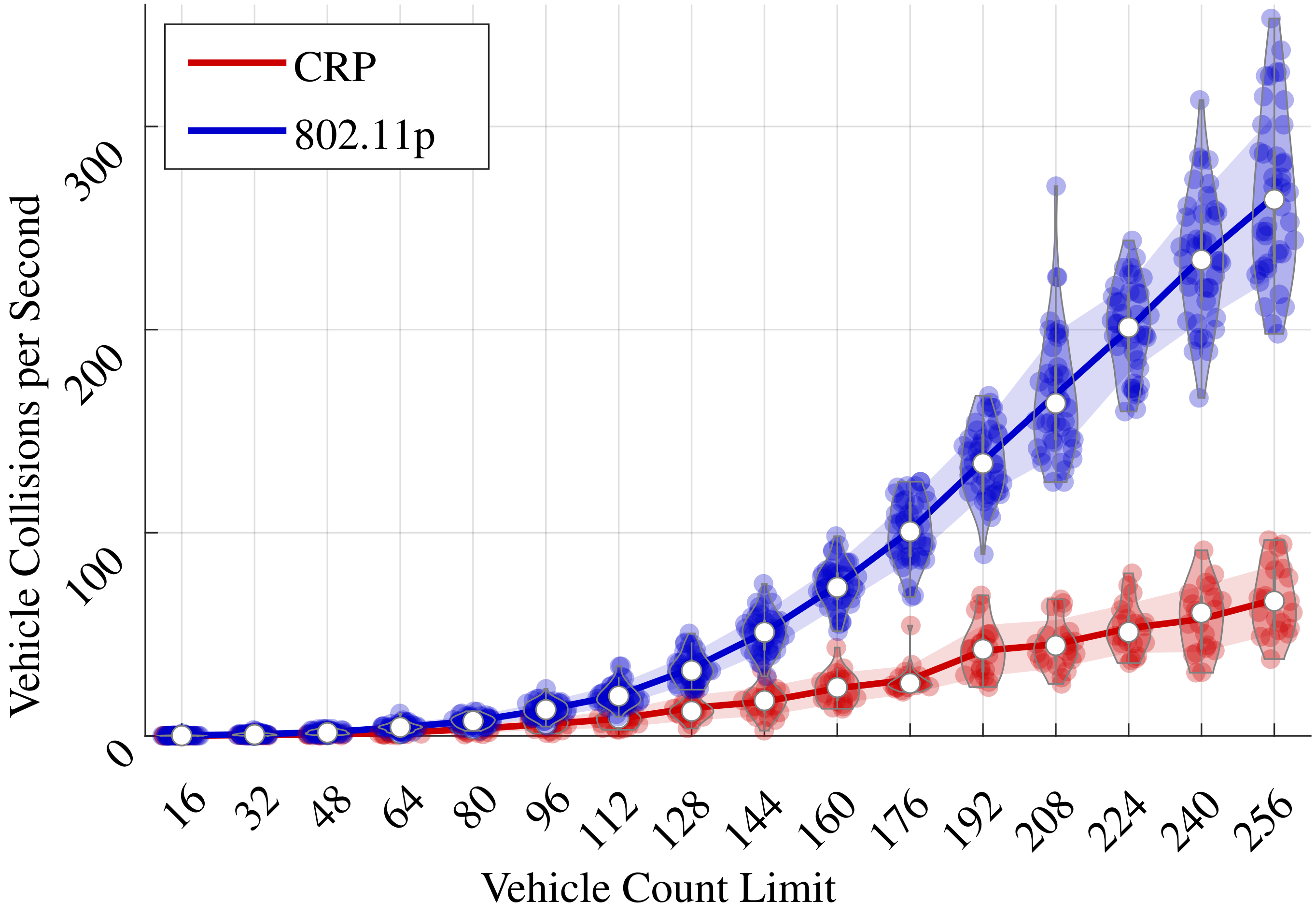


Fig. 8: Results of simulation comparing 802.11p and CRP-modified 802.11p. Panel (a) shows the overall packet delivery rates for safety beacons sent under each scheme, with CRP delivering considerable increases in successful reception. Panel (b) shows the average number of robots that were involved in crashes each second. Panel (c) shows the end-to-end delays, from application send to reception. The shaded area represents the results within 1 standard deviation of the mean and violin bars show the data distribution.

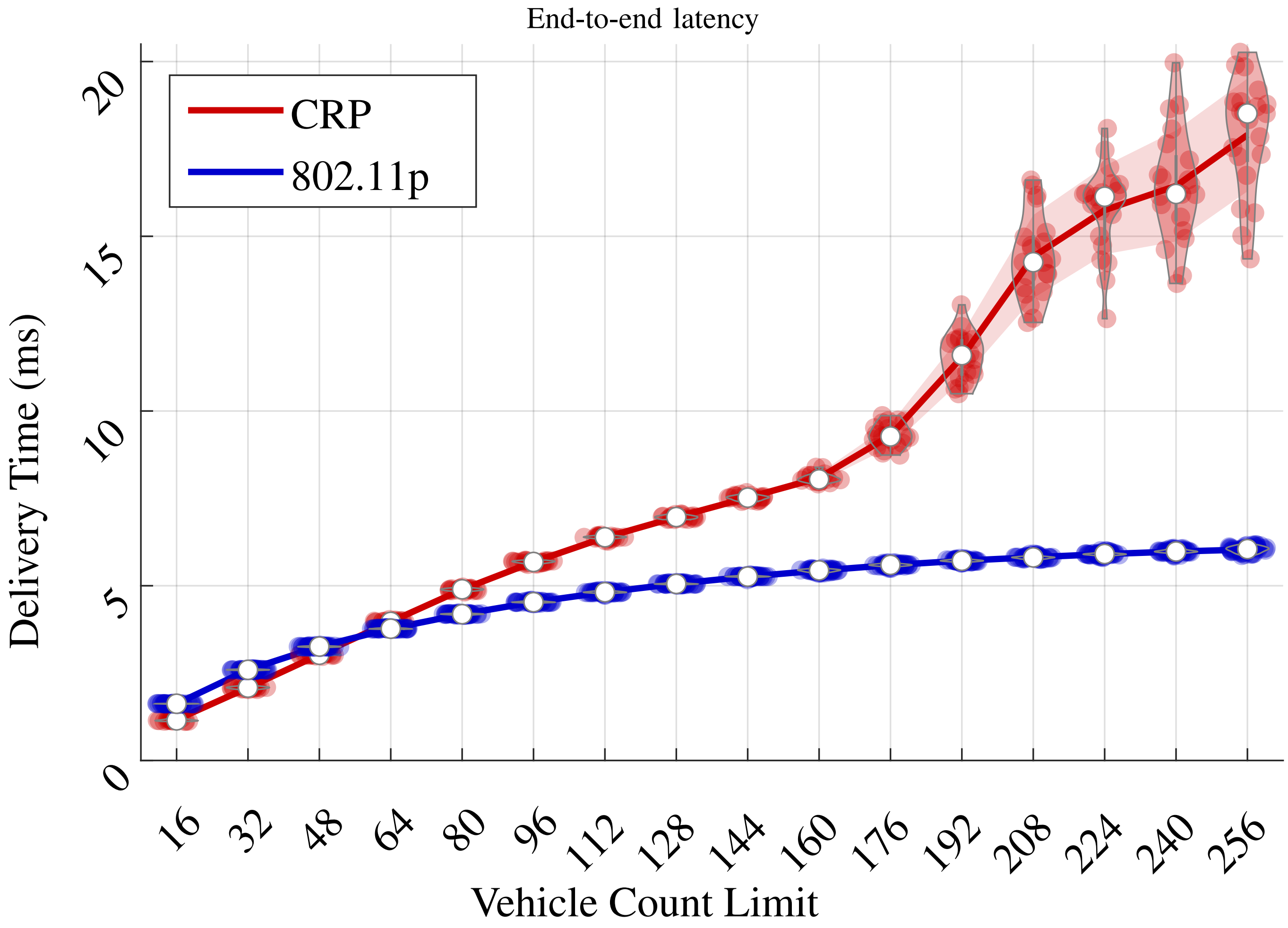
4

Please note that the caption for this figure is also for the following two

Robot Collisions



5


End-to-end latency
CRP
802.11p
Delivery Time (ms)
0
5
10
15
20
16
32
48
64
80
96
112
128
144
160
176
192
208
224
240
256
Vehicle Count Limit

6