LIMERIC: A Linear Adaptive Message Rate Algorithm for DSRC Congestion Control

Gaurav Bansal, *Member, IEEE*, John B. Kenney, *Member, IEEE*, and Charles E. Rohrs

***Abstract*—Wireless vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication holds great promise for significantly reducing the human and financial costs of vehicle collisions. A common characteristic of this communication is the broadcast of a device's core state information at regular intervals (e.g., vehicle speed and location or traffic signal state and timing). Unless controlled, the aggregate of these broadcasts will congest the channel under dense traffic scenarios, reducing the effectiveness of collision avoidance applications that use transmitted information. Active congestion control using distributed techniques is a topic of great interest for establishing the scalability of this technology. This paper defines a new adaptive congestion control algorithm that can be applied to the message rate of devices in this vehicular environment. While other published approaches rely on binary control, the LInear MEssage Rate Integrated Control (LIMERIC) algorithm takes advantage of full-precision control inputs that are available on the wireless channel. The result is provable convergence to fair and efficient channel utilization in the deterministic environment, under simple criteria for setting adaptive parameters. This "perfect" convergence avoids the limit cycle behavior that is inherent to binary control. We also discuss several practical aspects associated with implementing LIMERIC, including guidelines for the choice of system parameters to obtain desired utilization outcomes, a gain saturation technique that maintains robust convergence under all conditions, convergence with asynchronous updates, and using channel load to determine the aggregate message rate that is observable at a receiver. This paper also extends the convergence analysis for two important cases, i.e., measurement noise in the input signal and delay in the update process. This paper illustrates key analytical results using MATLAB numerical results and employs standard NS-2 simulations to demonstrate the performance of LIMERIC in several high-density scenarios.**

***Index Terms*—Congestion control, dedicated short-range communications (DSRC), safety, simulation, wireless networks.**

I. Introduction

TO enable intelligent transportation system applications, including collision prevention, the U.S. Federal Communication Commission has allocated 75 MHz of spectrum in the 5.9-GHz band for dedicated short-range communications (DSRC) among vehicles [vehicle-to-vehicle (V2V)] and between vehicles and roadside infrastructure [vehicle-to-infrastructure (V2I)] [1], [2]. The technology is being standardized in the IEEE 802.11 working group (WG) [3], [4], the IEEE 1609 WG [5], [6], and the SAE DSRC committee [13], [14]. Similar spectrum allocations and standards are being pursued in Europe.

DSRC uses the 802.11 medium access control (MAC) protocol, which is based on carrier-sense multiple access/collision avoidance (CSMA/CA). When a node has a packet to send, it first listens to the channel. If the channel is idle, the node transmits the packet. If the channel is busy, the node waits for a random backoff time before transmitting the packet. MAC frame collision probabilities are reduced via this mechanism; however, they remain nonzero. The probability of collision events increases with the number of nodes competing to access the wireless channel.

DSRC-based vehicle CA in the U.S. is based on a paradigm of frequent broadcasts of core state data from each vehicle. These utilize the basic safety message (BSM) [13], which includes information such as location, speed, and brake status, as well as path history and prediction. These messages are typically on the order of 300–400 bytes, including all lower layer overhead [7]. By default, they are transmitted over a single-hop 300–500-m range, with a 6-Mb/s data rate and a 10-Hz message rate. BSMs are sent on a 10-MHz "safety channel." In a high-density traffic scenario, there may be 150 or more vehicles present within the reception range of a transmitting vehicle, with even more vehicles within the interference range. In these scenarios, the packet error ratio (PER) will rapidly increase with vehicle density, as a result of increased frame collision probabilities. Field tests conducted with DSRC radios measured a PER as high as 79.5% for links with received signal strength between -80 and -85 dBm and with 360 emulated vehicles [12]. Increased probabilities of packet loss make it more difficult for a vehicle to model the movement of its neighbors and recognize vehicle collision threats. Hence, congestion on the DSRC safety channel can severely impact safety application performance. As a result, techniques to control and mitigate channel congestion are among the highest priorities for moving DSRC technology toward deployment. The Vehicle Safety Communications 3 Consortium is currently researching congestion control and scalability for the DSRC safety channel in cooperation with the U.S. Department of Transportation [15]. In the following section, we provide details about related work that has been conducted in the literature on congestion control. We also provide details about the organization of this paper.

Manuscript received October 1, 2012; revised April 25, 2013; accepted June 25, 2013. Date of publication July 26, 2013; date of current version November 6, 2013. The review of this paper was coordinated by Prof. H. Hartenstein.

G. Bansal and J. B. Kenney are with the Toyota InfoTechnology Center, Mountain View, CA 94043 USA (e-mail: gbansal@us.toyota-itc.com; jkenney@us.toyota-itc.com).

C. E. Rohrs is with the Toyota InfoTechnology Center, Mountain View, CA 94043 USA, and also with Rohrs Consulting, Inc., Newton, MA 02458 USA (e-mail: crohrs@mit.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVT.2013.2275014

A. Related Work

DSRC congestion control has been the subject of recent research. In [9], the vehicle sends BSMs more frequently when it experiences high dynamics so that neighbors can maintain an acceptable tracking accuracy, and it adapts transmit power as a function of the fraction of time that the channel is busy. In [10], a distributed transmit power control method is proposed, which reduces the power of safety message transmissions during congestion to control the safety channel load. In [11], a message-rate-control-based approach is proposed to adapt the BSM transmission rate based on a binary comparison between measured channel load and a target threshold. Binary message rate control is also the subject in [12], in which the co-authors of this paper proposed an additive-increase–multiplicative-decrease message rate update mechanism for DSRC safety communication, including results from prototype radio tests and computer simulations that illustrate effective congestion control for hundreds of emulated or simulated vehicles. DSRC congestion control is expected to be an important topic for international standardization. The European Telecommunications Standards Institute (ETSI) has published a framework for decentralized congestion control (DCC) in [25] and is actively considering extensions and improvements. This framework envisions a variety of potential controls, including message rate, transmit power, data rate, and receive sensitivity. Similar efforts are expected in the U.S. in the future. In [24], Subramanian *et al.* proposed a synchronous algorithm for congestion control based on a time-slotted MAC as well as improvements to the ETSI DCC. In [34], broadcast efficiency is defined as an average rate at which vehicles receive broadcast packets from its neighbors. Yei *et al.* proposed congestion control strategies such that broadcast efficiency is maximized in vehicular ad hoc networks (VANETs). Prioritized contention control where priority is assigned for propagation of emergency warning messages in VANETs is studied in [35]. In [36], Nasiriani *et al.* have conducted fairness and stability analysis for algorithms that adapt transmission rate and power for doing congestion control in VANETs. It has been shown that there is a tradeoff between convergence speed, stability, and fairness for the proposed algorithms. In [37], Tielert *et al.* studied joint rate and power control schemes for achieving congestion control in VANETs and concluded that an efficient strategy is to select transmit power according to the target awareness distance and to adapt the message rate according to channel load. In [38], the effect of different choices of transmission range and message rate is analyzed, and models have been developed to quantify network performance. A transmission range adaptation-based feedback control scheme has been designed, which is robust to network traffic and variations on the road.

There has been also more than a decade of research conducted on Mobile Ad hoc NETworks (MANETs) [30]. However, that literature deals with a significantly different problem than is considered in this paper. Under the MANET label, researchers consider multihop networks with emphasis on routing and backward compatibility with the Transmission Control Protocol (TCP). Even when considering rate control [31] and broadcast applications [32], the MANET model considers how to modify multihop network protocols. Here, as expounded in [33], we consider a network that is fundamentally broadcast both in its shared medium and its applications.

B. Organization

In this paper, we propose a message rate control algorithm that improves on the binary approach in [11] and [12], i.e., the Linear MEssage Rate Integrated Control (LIMERIC). It uses linear feedback to adapt the message rate and, thus, avoids the limit cycle behavior that is inherent in binary control. In Section II, we introduce the algorithm and use analysis to prove deterministic convergence as a simple function of adaptation parameters. We assume that the algorithm knows the aggregate message rate in a single collision domain; however, the algorithm neither needs to know nor attempts to estimate the number of neighboring vehicles. Convergence is shown both with regard to fairness (vehicle message rates converge to each other) and to efficient channel utilization relative to a target. Section III provides detailed discussions of several practical considerations in using LIMERIC. The first concerns selection of algorithm parameters to achieve desired channel utilization. The second examines the convergence criteria and introduces a gain saturation technique that guarantees stable convergence under all operating environments. With this technique, LIMERIC enjoys convergence to a point in the vast majority of cases but falls back to stable limit cycles in the rare case when device density exceeds conservative estimates. The basic convergence proof assumes synchronous update behavior, i.e., devices update their message rates at same times. The paper also considers the more easily achieved asynchronous update case and offers analysis to show that convergence is even more robust in that regime. Section III concludes with a discussion of the use of a standardized channel activity measure as a substitute for the aggregate rate that is used in the analysis and shows conditions under which the analytical results remain valid. While this algorithm is applicable in a variety of channel congestion environments, we focus on the DSRC safety channel use case. In Section IV, we present computer simulation results that illustrate various facets of system performance considered in Sections II and III. Sections V and VI expand the earlier analysis in two important dimensions. Section V considers the effect of measurement noise on the channel load input. Section VI analyzes convergence when there is delay in the error signal used in LIMERIC adaptation. We present conclusions and directions for future research in Section VII.

II. Linear Message Rate Integrated Control: A Linear Control Algorithm

In general, the objectives of an adaptive congestion control approach are threefold: 1) to converge to a desired channel utilization level (channel load); 2) to achieve local fairness among immediate neighboring vehicles; and 3) to achieve global fairness among all vehicles contributing to congestion. The scope of this paper is limited to objectives 1 and 2, i.e., we simplify the network to a single collision domain where all the nodes measure the same channel load. For such a network, we aim to achieve fairness such that all the nodes converge to the

same message rate. The NS-2 simulations use a topology with a small number of hidden nodes to show that algorithm behavior is essentially the same as predicted for no hidden nodes. Objective 3, which deals with wide-area fairness for a larger network with many hidden nodes, will be studied as future work and will likely depend on the exchange of additional protocol information between vehicles [11]. We briefly comment on this in Section III-D. LIMERIC can be extended to use a more general congestion indicator; however, in this paper, it only uses local information.

In most networks using adaptive control, each user has limited information about the amount of congestion. Usually, this information can be summarized by a binary variable indicating that the network is either congested or not. For example, in the Internet, the TCP uses the absence or late arrival of an acknowledgment (ACK) as an indication of congestion and the timely arrival of an ACK as an implicit indication of the availability of additional capacity [20]. Many other protocols use implicit or explicit binary congestion indicators as well [16], [17], [21]. The limitation of binary information feedback leaves the rate control algorithm with certain known problems concerning convergence to fairness unless the distributed users are assumed to act synchronously [18]. In addition, with binary information feedback, the rates of a system converge only to within a steady-state oscillation or limit cycle around some average rate.

Here, we introduce the LIMERIC algorithm for a network in which each user has full-precision information about the state of the congestion. In particular, we assume that each user can measure the fraction of network capacity that is in use at each moment. To model this mathematically, we let $r_j(t), j = 1, 2, \ldots, K$, be a number between 0 and 1 representing user j's rate of transmission as a fraction of the total channel capacity, assuming there are K active users on the network. For example, if message transmission takes 0.5 ms, $r_j(t) = 0.004$ corresponds to eight messages per second (msg/sec). Here, we assume that each user can accurately measure

$$r_C(t) = \sum_{j=1}^{K} r_j(t), \tag{1}$$

the fraction of the capacity of the network allocated in aggregate to all users and that it can do so without knowing or estimating neighborhood size K. Note that if K were reliably known at each node, a fair division of the channel resource would be trivial, i.e., each node taking a $1/K$ share. We discuss in Section III-D the option of estimating $r_C(t)$ based on the IEEE 802.11 clear channel assessment (CCA) function [3]. While $r_C(t)$ in general varies with measurement location, here, we make the simplifying assumption that all vehicles measure the same value at a given time t. The resulting analysis provides useful insights for behavior in more complicated situations, which are briefly discussed in Section III-D. While LIMERIC does not explicitly use K, the analyses in this paper treat K as fixed. This allows us to explore the critical steady-state characteristics of the algorithm, including stability, convergence, and fairness. In practice, of course, LIMERIC will operate in environments where K changes with time. As with all adaptive algorithms, LIMERIC will track these changes if adaptive parameters α and β are chosen to be sufficiently aggressive. We comment further on this in Sections III-A and IV. LIMERIC uses a parameter, i.e., r_g, that, to a first-order approximation, is the goal for total rate allocated to the nodes. Our analysis will admit a more precise understanding of the role of r_g. In this paper, we assume that it is desired to share the total rate equally among the nodes. Variations of LIMERIC involving priority systems or weighted fair sharing are possible [28] but are outside the scope of this paper.

With the added information available in $r_C(t)$, each user can adjust her own rate linearly with respect to the error in the total rate being used on the network, i.e., with respect to $e(t) = r_g - r_C(t)$. Compared with binary control, such a linear scheme has an advantage in that it can converge to fair rates with no limit cycles and that such convergence can be analyzed using tools from linear systems theory. Since each user updates her rate episodically, we consider this system in discrete time, and we initially consider the synchronous update case where each user updates her rate once per discrete-time step, all users update at the same time instants, and the effect of all rate adjustments appears in the measurement of the total channel rate at the next discrete-time step.

Let each user update her rate by

$$r_j(t) = (1 - \alpha) r_j(t-1) + \beta \left(r_g - r_C(t-1) \right) \tag{2}$$

where $r_C(t)$ is given by (1), $0 < \alpha < 1$, $\beta > 0$.

In this equation, the fundamental role of parameter α is as an exponential forgetting factor that promotes fair convergence. The message rate offset is linear in the total rate error $e(t)$, which is scaled by adaptive gain factor β. As shown in the succeeding analysis, α and β jointly trade off convergence speed with algorithm convergence. To a first-order approximation, for small positive values of each, higher values promote faster convergence, and lower values improve stability robustness. These parameters also determine the precise convergence point for a given r_g and K. In the presence of measurement noise, as analyzed in Section V, α and β also impact the size of the induced steady-state message rate variation. For most of these metrics, β is the more critical of the two parameters.

An important quality of the synchronous case is that the $r_C(t)$ measurements and the individual rate updates are each performed at the same time instants by all users. This is in contrast to what we call the sequential case (analyzed in Section III-C) where user 2's update is based on a measured total rate that includes an intermediate update in user 1's rate, and user 3 includes both user 1 and user 2's intermediate updated rates in her calculation and so on.

The system defined by (1) and (2) can be analyzed by forming a vector

$$\vec{r}(t) = [\, r_1(t) \quad r_2(t) \quad \ldots \quad r_K(t) \,]^T \tag{3}$$

where T indicates the matrix transpose operator. With this, (1) and (2) become

$$\vec{r}(t) = A\vec{r}(t-1) + b r_g \tag{4}$$

where

$$A=\begin{bmatrix} 1-\alpha-\beta & -\beta & -\beta & \dots & -\beta \\ -\beta & 1-\alpha-\beta & -\beta & \dots & -\beta \\ -\beta & -\beta & 1-\alpha-\beta & \dots & -\beta \\ . & . & . & \dots & . \\ . & . & . & \dots & . \\ -\beta & -\beta & -\beta & \dots & 1-\alpha-\beta \end{bmatrix}$$

$$b=\begin{bmatrix} \beta \\ \beta \\ \beta \\ . \\ . \\ \beta \end{bmatrix}. \tag{5}$$

Matrix A has one eigenvalue at $z = 1 - \alpha - K\beta$ and $(K - 1)$ eigenvalues at $z = 1 - \alpha$. A set of unnormalized eigenvectors are

$$\begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ . \\ . \end{bmatrix}, \begin{bmatrix} 1 \\ -1 \\ 0 \\ 0 \\ 0 \\ . \\ . \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ -2 \\ 0 \\ 0 \\ . \\ . \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 1 \\ -3 \\ 0 \\ . \\ . \end{bmatrix}, \dots .$$

With the added condition

$$1 - \alpha - K\beta > -1 \quad \text{or} \quad \alpha + K\beta < 2 \tag{6}$$

the system represents an asymptotically stable linear discrete-time system and, thus, converges to a single fixed point, which is given by

$$r_j = \frac{\beta r_g}{(\alpha + K\beta)}, \qquad j = 1, 2, \dots, K. \tag{7}$$

Convergence to fairness, i.e., of rates r_m and r_n to each other, is controlled by eigenvalues $1 - \alpha$ and is assured independent of K. The speed of this convergence to fairness is a discrete exponential $(1 - \alpha)^t$ from any initial condition of unfairness. This eigenstructure and (2) imply that for any two users

$$r_m(t) - r_n(t) = (1 - \alpha)\left(r_m(t - 1) - r_n(t - 1)\right) \tag{8}$$

or

$$r_m(t) - r_n(t) = (1 - \alpha)^t \left(r_m(0) - r_n(0)\right). \tag{9}$$

Now, consider the dynamics of the total rate on the channel, which is controlled by the eigenvalue at $z = 1 - \alpha - K\beta$. Summing across the individual r_j in vector (4), we get

$$r_C(t) = (1 - \alpha - K\beta) r_C(t - 1) + K\beta\, r_g. \tag{10}$$

This difference equation can be solved to get

$$r_C(t) = r_C + (1 - \alpha - K\beta)^t (r(0) - r_C) \tag{11}$$

where the steady-state total rate r_C is given by

$$r_C = \frac{K\beta r_g}{\alpha + K\beta}. \tag{12}$$

Comparing (7) and (12), it is evident that the fairness property ensures that each r_j converges to r_C/K.

To maintain convergence of the overall total rate, β must be chosen so that the convergence condition of inequality (6) is met for the largest envisioned value of K. Section III-A discusses detailed implications for the choice of parameters α, β, and r_g. However, here, we briefly note two approaches to maintaining good convergence behavior. The first involves adapting β in a slower acting loop compared with r_j adaptation. When $r_C(t)$ is significantly less than r_g, β can be increased. When a node sees a large and oscillating $[r_g - r_C(t)]$, β should be decreased.

A second method of ensuring a well-behaved algorithm that is independent of K is to apply a maximum magnitude saturating nonlinearity to the adjustment term $\beta(r_g - r_C(t - 1))$ in (2). Then, when K is too large for the chosen static β, the system will converge to a limit cycle behavior and react similar to a system with binary feedback, but with the advantage that it can return to a response with no limit cycles when either K or β decreases to satisfy inequality (6). These two methods, slowly adapting β and saturating nonlinearity, appear to be complementary. In this paper, we employ the analyzable static β and explore the use of the gain saturation approach in Section III-B. The approach with slowly adapting β is not studied in this paper and is left as future work.

III. Discussions

A. Selection of System Parameters

This section presents suggestions on how adaptation parameters α and β should be set, as well as the nominal goal rate r_g. This paper only considers the case where all three of these parameters are constants; however, the work could be extended to allow for dynamic parameters.

These parameters, along with the number of vehicles K, determine the following algorithm behaviors:

- convergence condition—inequality (6);
- convergence equilibrium point—(12);
- convergence speed—(9) and (11).

Ensuring convergent behavior is the highest priority. Convergence of the base linear algorithm requires $\alpha + K\beta < 2$. β has a more important role in this expression than α, since it needs to counter the effect of K. Therefore, α is selected more for its impact on equilibrium and convergence speed, as will be discussed. Given that β is chosen to be constant, it is tempting to consider whether there exists a largest K, i.e., $K_{\max}$, and to set β proportional to $1/K_{\max}$. However, choosing β to be very small has negative consequences for the other behaviors, as will be explained. Hence, in this paper, β is set to ensure convergence of the base algorithm over a broad range of neighborhood sizes and to allow for a small probability that inequality (6) is not satisfied. Section III-B provides a means for ensuring appropriate behavior for that low-probability case.

The choice of a value for α in the range 0–1 involves a tradeoff that can best be seen in (9) and (12). From (9), it is seen that the speed of convergence to fairness entirely depends on α and that larger α corresponds to faster convergence. From (12), we see that the aggregate rate equilibrium point, i.e., r_C, depends on unknown K. In general, it is desired that r_C varies as little as possible with K. Choosing α small, i.e., closer to 0, promotes that objective. To achieve a good tradeoff between faster convergence to fairness and lower variations in r_C, α is set to 0.1 in this paper in all of the examples.

Returning to the selection of β, we see in (12) that the goal of small r_C variation is consistent with higher values of β, within the constraints of inequality (6). Furthermore, from (11), we see that the aggregate rate converges with time constant $1-\alpha-K\beta$. For a given α and K, as β increases from 0 toward $(1-\alpha)/K$, the time constant is positive with decreasing magnitude, leading to faster convergence. As β increases beyond $(1-\alpha)/K$, the time constant becomes negative, and the magnitude grows. For values of the time constant between 0 and 1, the system monotonically converges in a noiseless environment. For values between 0 and -1, the system converges but in an oscillatory way. For values less than -1, convergence of the base algorithm is not assured. The β value should be small enough to ensure convergence for realistic values of K and large enough so that the variation in r_C is acceptably low. In [8], it is shown that when $K = 150$ vehicles communicate V2V safety messages at 10 Hz and 20 dBm, the system can be said to be moderately congested, with channel activity measuring between 60% and 70%. For smaller values of K, the IEEE 802.11 frame collision rate is modest, on the order of 20% or less. For larger values of K, the frame collision rate increases much more rapidly. In the remainder of this paper, we evaluate LIMERIC for the case $\beta = 1/150$; as shown below, this achieves a good balance between the competing objectives outlined here. While the specific values that we have chosen for α and β provide a good basis for performance evaluation, we do not assert that they are optimal. We identify improved methods for selecting parameter values as a valuable area for future research. For the parameter assignment ($\alpha = 0.1,\ \beta = 1/150$), the system behavior of LIMERIC is as follows.

- From inequality (6), convergence is guaranteed for $K < 285$.
- From (12), for $K < 285$, the aggregate rate converges to

$$r_C = \frac{r_g}{1+\frac{15}{K}}. \tag{13}$$

- From (9), fairness time constant is 0.9.
- From (11), for $K < 285$, the system convergence time constant is $|0.9-(K/150)|$, i.e., for $K < 270$, it is never slower than a system with constant 0.9, and it is fastest when K is approximately 135. In simulations shown below, using these parameters, convergence from any initial condition is generally achieved in 20–40 iterations of LIMERIC. In practice, we use an update interval of 200 ms; hence, convergence is on the order of 4–8 s. Small-scale tracking, with incremental changes to K, is much faster because the difference in equilibrium points [see (12)] is quite small when K changes by a few vehicles. If faster convergence were desired, the best way would be to increase α, as seen in (11).

Finally, from (13), it is apparent how to choose goal rate r_g to achieve a desired equilibrium r_C. Ideally, this equation would be independent of the unknown K; however, there remains some dependence. For large values of K, $r_C \approx r_g$. For smaller values of K, r_C converges to a smaller fraction of r_g.

This is not a concern for V2V safety communication; however, since it is generally recognized that there is a small diminishing return for any marginal increase in the vehicle message rate above approximately 10 msg/sec. V2V safety messages are typically on the order of 3 kb; hence, there is room for approximately 2000 msg/sec in a channel using a 6-Mb/s data rate. Imagine that r_g is set to 0.6, which is equivalent to about 1200 msg/sec. From (13), if K is small, e.g., 15, equilibrium r_C is only $0.5^* r_g$ or 0.3. This corresponds to an aggregate rate of about 600 msg/sec. From (9), these are shared fairly among the 15 vehicles, i.e., the individual vehicle message rate in steady state, $r_j, = r_C/K, \forall j = 1,\ldots,15$. Hence, although r_C is only half of r_g, it still allocates approximately 40 msg/sec/vehicle, far in excess of the maximum "useful" rate.

In practice, it is likely that each vehicle's rate will be capped, e.g., at 10 msg/sec (or equivalently, 0.005 of the channel's 2000-msg/sec capacity), external to the adaptive algorithm. In that case, if r_g is set to 0.6, then using (13), the aggregate steady-state offered message rate is

$$r_C = K * \min\left[0.005, \frac{0.6}{K+15}\right]. \tag{14}$$

For values of $K < 105$, the per-vehicle maximum rate applies. When $K > 105$, the adaptive equilibrium is such that each node converges to a rate below its cap. In that case, r_C is within 87.5% of r_g.

Note that we do not impose a positive minimum rate on $r_j(t)$. However, if required for the functioning of a particular safety application, a minimum rate can be applied after LIMERIC adaptation, i.e., (2). In this paper, our goal is to allocate fair rates to all participating vehicles while keeping the channel load below the specified threshold, to achieve a low PER and high throughput. It is possible that some safety applications might require a positive minimum rate; this is beyond the scope of our paper. By having a positive minimum rate, for scenarios with high vehicle density, the channel load might not be maintained below the threshold. This would lead to a high PER, and throughput will significantly degrade [12].

The MATLAB plots are presented to demonstrate the behaviors determined by the analysis in Section II. Note that while the analysis uses normalized message rates, where 1.0 is the maximum number of messages the channel can support, the simulation results in this paper show unnormalized message rates in units of messages per second. This is a more natural way of viewing safety message behavior. While the analysis does not rely on a minimum or a maximum rate constraint, the simulations limit the rate at each node to a range of 0–10 msg/sec (except that no maximum rate is employed in Figs. 12 and 13 in Section V). We consistently set the values

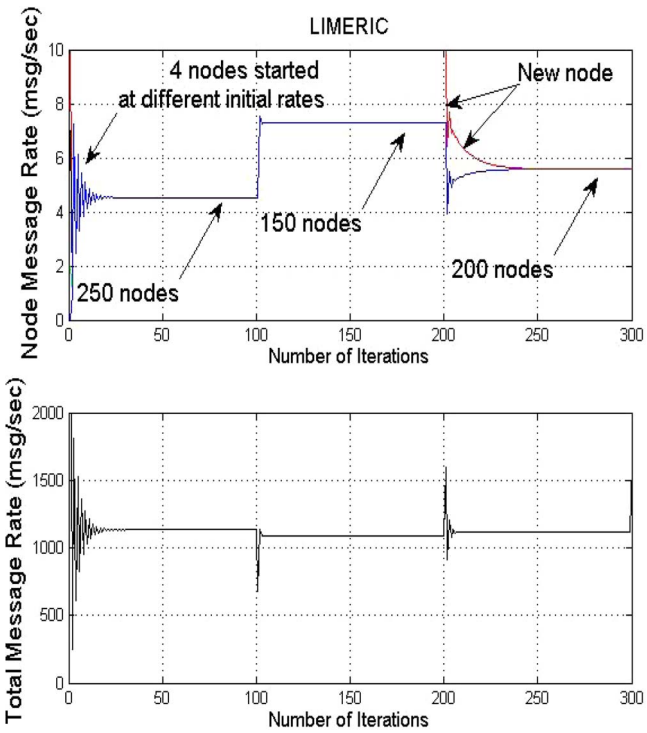


Fig. 1. (a) Individual node message rate while using LIMERIC algorithm. Total number of nodes = 250, 150, 200; inequality (6) satisfied in all three cases. (Blue curves) Nodes present from Iteration No. 0. (Red curves) Nodes introduced at Iteration No. 200. (b) Total message rate for all nodes while using LIMERIC algorithm.

of α to 0.1, β to 1/150, and r_g to 0.6, unless otherwise noted. This r_g is equivalent to about 1200 msg/sec (see Fig. 6).

Fig. 1 shows the convergence behavior as LIMERIC is applied first to a group of 250 nodes; then as 100 of those nodes are removed, leaving LIMERIC operating on a set of 150 nodes; and finally, as 50 nodes are added and LIMERIC operates on a set of 200 nodes. While the neighborhood size would normally change much more incrementally, these large changes are used to illustrate important aspects of algorithm behavior. Fig. 1(a) shows the individual message rate of selected nodes, and Fig. 1(b) shows the aggregate message rate of all nodes active at a given iteration (250, then 150, then 200). During the first 100 iterations, Fig. 1(a) plots the message rate of four selected nodes, with respective initial message rates 10, 7, 2, and 0 msg/sec. The 246 nodes whose individual rates are not plotted are initialized to 10 msg/sec. Note that after the first few iterations, the rates of those four nodes have converged to each other, such that the curves are no longer distinguishable, and they remain indistinguishable through the remainder of the 300 iterations. This demonstrates the fairness properties of LIMERIC, as previously developed. After approximately 20 iterations, the individual and aggregate message rates have reached steady-state values of approximately 4.5 and 1130 msg/sec, as predicted by (7) and (12). When 100 nodes are removed at iteration 100, the individual message rates of the four selected nodes quickly reconverge to a new steady state of approximately 7.3 msg/sec [see (7)]. The aggregate message rate initially dips, due to the removed nodes, and the returns to approximately the same steady-state value, showing that

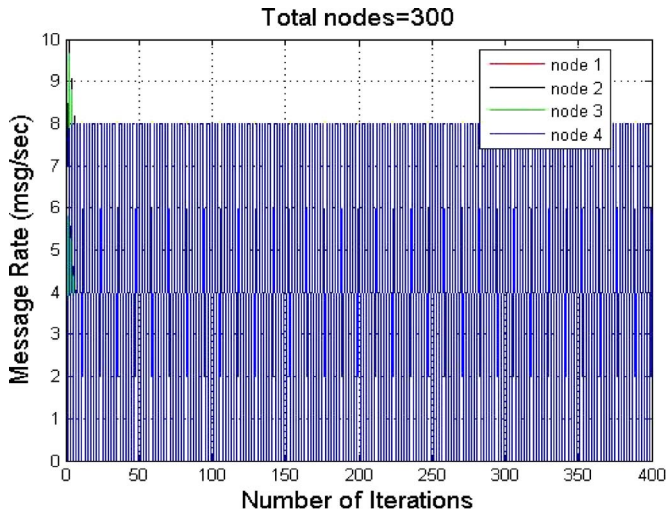


Fig. 2. LIMERIC for total $nodes = 300$; inequality (6) not satisfied.

LIMERIC controls the aggregate message rate [see (12)] as the number of nodes changes. At iteration 200, 50 new nodes are introduced with an initial rate of 10 msg/sec. Fig. 1(a) shows the convergence of one representative node among this subset of 50 (red curve). The original 150 nodes, which are represented by the four blue curves, quickly reconverge to a new steady state of approximately 5.6 msg/sec [see (7)], and are joined there by the message rate of the newly introduced node (red curve). The aggregate message rate in Fig. 1(b) has a step increase with the introduction of the new nodes but then quickly reconverges to approximately the same total that was achieved with 250 and 150 nodes [see (12)]. Again, this demonstrates that LIMERIC controls the aggregate message rate (channel load), independent of the number of nodes contributing.

In Fig. 2, we change the total number of nodes to 300; hence now, inequality (6) is not satisfied. For this scenario, as predicted by the theory, the base linear algorithm does not converge. The constraints added to the simulation function keep the rates bounded but with large oscillations. In the following section, we demonstrate a simple practical modification called gain saturation that for very large K achieves limit cycle convergence similar to most binary control algorithms, allowing the improved behavior of LIMERIC when K satisfies inequality (6).

B. Gain Saturation to Improve Behavior for Very Dense Environments

The convergence condition for LIMERIC is shown in inequality (6), as a function of α, β, and K. Since K is unknown and has no hard upper bound, it is not possible to choose static values of α and β for which convergence is assured. One strategy for dealing with this problem is to adjust α and/or β dynamically with changing K to keep inequality (6) satisfied. This technique is promising but is not studied in this paper and is left for future study. The strategy adopted in this paper is to introduce constraints to adjustments in the basic update (2). Recall that LIMERIC neither knows nor attempts to estimate

K. This section describes this gain saturation approach and illustrates the resulting behavior.

We modify the LIMERIC update (2) as follows:

$$r_j(t) = (1-\alpha)r_j(t-1) + \text{sign}\,(r_g - r_C(t-1)) * \min \times [X, \beta * abs(r_g - r_C(t-1))]\,. \quad (15)$$

In other words, if the magnitude of the update offset, i.e., $\beta|r_g - r_C(t-1)|$, exceeds threshold X, the update is limited to $\pm X$, with the sign reflecting the sign of the error term $(r_g - r_C(t-1))$. Under conditions that would cause inequality (6) not to be satisfied, the system with saturation nonlinearities converges to limit cycle behavior. That is, each vehicle's message rate oscillates with a fixed amplitude around a point. The trapping behavior within the limit cycle and the size of the limit cycles can be established by using describing function analysis, which is a common control theory technique [19]. The logic of the theory is that when the error signal, i.e., $\beta|r_g - r_C(t-1)|$, is large, the saturation value produces the equivalent of a small gain, i.e., $X/\beta|r_g - r_C(t-1)|$, and the system tends to converge to smaller error signals. However, if the linear gain β of the saturation nonlinearity is too large for the given value of K, the system will tend to diverge from small error signals. The limit cycle results when these two effects balance. The behavior of this system in limit cycle is similar to the behavior of an adaptive system with binary feedback, e.g., TCP. If K is small enough so that the gain in the linear region satisfies inequality (6), the system converges as if the saturation were not present. Therefore, adding the saturation nonlinearity provides insurance in the unlikely case that inequality (6) is not satisfied; however, it does not interfere with the convergence to the point shown in (7) when inequality (6) is satisfied.

In the prior section, it was noted that, in practice, a V2V safety system will likely impose a maximum message rate per node. Including gain saturation logic in the LIMERIC update imposes X/α as a maximum steady-state fraction of channel capacity that a node can use (corresponding to a 2000*X/α-msg/sec rate). If a particular $r_j(t)$ takes on a larger value, e.g., $(X+\varepsilon)/\alpha$, then after the next update, it will necessarily contract to a value no higher than $(1-\alpha)(X+\varepsilon)/\alpha + X = [X + \varepsilon^*(1-\alpha)]/\alpha < r_j(t)$. The rate will continue to contract in subsequent updates until it is no greater than X/α. Setting the saturation gain X too small can impose an unwanted rate limit and can also slow down convergence. The value of X is also proportional to the limit cycle peaks when gain saturation is active; hence, it should also not be set too large. In this paper, when gain saturation is used, X is set to 0.005 (corresponding to 1 msg/sec). For $\alpha = 0.1$, this creates a maximum per-node channel usage of 0.05 (corresponding to 10 msg/sec), which is consistent with common industry assumptions about maximum message rates.

Note that while the positive and negative gain saturation values have equal magnitude in (15), it is also possible to use asymmetric saturation. Such an adjustment alters the precise form of the limit cycle and the average message rate in steady state.

Fig. 3 is a MATLAB plot of the convergence trajectory for nodes = 300, 1000 under a LIMERIC configuration (four selected nodes with initial message rates of 10, 7, 2, and 0 msg/sec are shown) that is identical to that in Fig. 2, except that the update includes gain saturation, i.e., it uses (15) rather than (2). Comparing Figs. 2 and 3 (for 300 nodes), the trapping effect of gain saturation is evident; instead of diverging, the trajectories converge to a limit cycle. Further, in Fig. 3, we also show the LIMERIC results with gain saturation for 1000 nodes. This is an extreme scenario for which the network is heavily congested. We observe from the plot that even for this extreme scenario, LIMERIC remains stable and enters into a tolerable limit cycle.

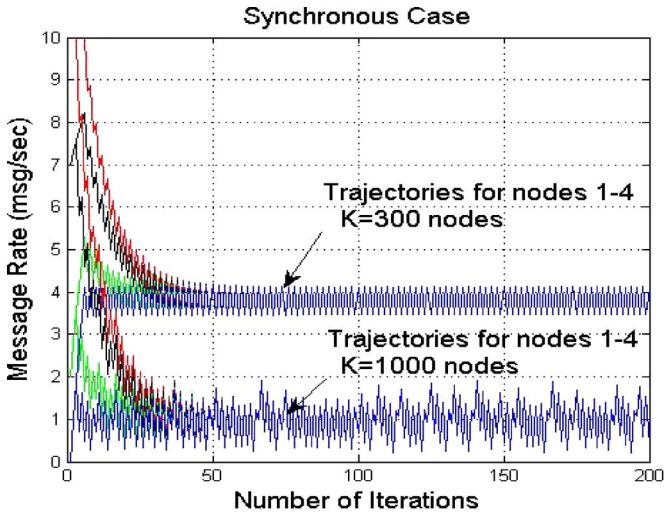


Fig. 3. LIMERIC for total nodes = 300 and 1000 with gain saturation; inequality (6) not satisfied.

For most of the rest of this paper, we are interested in the behavior of the algorithm in cases where the analysis applies, and thus, we pick scenarios where inequality (6) applies, and no gain saturation is used.

C. Asynchronous Case

So far, we have assumed that all nodes participating in congestion control are synchronized; however, in practice, it may be preferred to implement the system without synchronization. Nodes might perform message rate updates asynchronously, i.e., they will not update at the same time instants. Here, we analyze the system when the updates by the nodes are sequentially performed within each time step.

The mechanism that can cause convergence problems when the algorithm is synchronously implemented gives reason to believe that convergence may occur more easily when an algorithm is asynchronously implemented. In the synchronous case at each update instant, all K nodes move their rates in the same direction. The analysis then depends on the potentially large gain $K\beta$ appearing in (11) and the system's eigenvalue as $(1-\alpha-K\beta)$. If the synchronicity is broken up, in each period, some nodes will increase rate while others decrease rate, making the effective gain smaller and convergence more likely for larger numbers of nodes.

A scenario that can be realized more easily than synchronous updates is that all vehicles use the same update period but that each vehicle's specific update time within the period repeats from one time to another. In this special case of asynchronous

updates, the vehicles perform updates in a sequential pattern across the period, and that pattern is repeated in each period. One can assign vehicle index 1 to the first vehicle to update in the period, index 2 to the next vehicle, and so on. When vehicle 2 performs its update, the aggregate rate it uses includes vehicle 1's most recent update.

We now consider the sequential update algorithm with two nodes. The update equations are

$$\begin{aligned} r_1(t) &= (1-\alpha)r_1(t-1) + \beta\,(r_g - r_1(t-1) - r_2(t-1)) \\ r_2(t) &= (1-\alpha)r_2(t-1) + \beta\,(r_g - r_1(t) - r_2(t-1))\,. \end{aligned} \tag{16}$$

The key here is the time indexing on rate r_1 in the update of r_2. Manipulating the second equation in (16) gives us a state-space representation in the rates

$$\begin{aligned} r_2(t) &= (1-\alpha)r_2(t-1) + \beta\,(r_g - (r_1(t) - r_1(t-1)) \\ &\qquad - r_1(t-1) - r_2(t-1)) \\ &= (1-\alpha)r_2(t-1) + \beta\,(r_g - r_1(t-1) - r_2(t-1)) \\ &\quad + \alpha\times\beta\times r_1(t-1) - \beta^2\,(r_g - r_1(t-1) - r_2(t-1)) \\ &= (1-\alpha-\beta+\beta^2)r_2(t-1) - (\beta - \beta^2 - \alpha\beta) \\ &\quad \times r_1(t-1) + (\beta-\beta^2)r_g. \end{aligned} \tag{17}$$

It is more convenient to redefine the states with

$$x_j = \sum_{k=1}^{j} r_k. \tag{18}$$

Now, we have

$$\begin{aligned} x_1(t) &= (1-\alpha)x_1(t-1) + \beta\,(r_g - x_2(t-1)) \\ x_2(t) &= x_1(t) + r_2(t) \\ &= (1-\alpha)x_1(t-1) + \beta\,(r_g - x_2(t-1)) \\ &\quad + (1-\alpha)\,(x_2(t-1) - x_1(t-1)) + \beta\,(r_g - x_2(t-1)) \\ &\quad + \alpha\beta x_1(t-1) - \beta^2\,(r_g - x_2(t-1)) \\ &= \alpha\beta x_1(t-1) + (1-\alpha-\beta-\beta(1-\beta))\,x_2(t-1) \\ &\quad + (\beta - \beta(1-\beta))\,r_g. \end{aligned}$$

Continuing in this vein for larger numbers of nodes, a pattern emerges for the partial sums state evolution, i.e.,

$$\vec{x}(t) = G\vec{x}(t-1) + hr_g\,.$$

The following code fills the state transition matrix and the input vector for any number of nodes, i.e., K:

```
g = 1 − a;
d = 1 − b;
G = zeros(K, K);
h = zeros(K, 1);
for jj = 1 : K − 1
G(jj, jj) = g;
G(jj + 1, jj) = a*b;
for ii = jj + 2 : K
G(ii, jj) = d*G(ii − 1, jj);
end
end
G(1, K) = −b;
h(1, 1) = b;
for ii = 2 : K − 1
G(ii, K) = −b + d*G(ii − 1, K);
h(ii, 1) = b + d*h(ii − 1, 1);
end
G(K, K) = g − b + d*G(K − 1, K);
h(K, 1) = b + d*h(K − 1, 1);
```

We provide simple proofs for the following five properties.

Prop. 1: When $\beta = 0$, the system is stable, but all rates asymptotically go to zero as no error information is used.

Proof: When $\beta = 0$, $G = (1-\alpha)I$, where I is the identity matrix, and h is an all-zero vector.

Prop. 2: For $0 < \alpha < 1$ and small enough β, matrix G is the state transition matrix of a stable system.

Proof: The eigenvalues of G are continuous functions of β. This, combined with property 1, gives property 2.

Prop. 3: When $\alpha + \beta = 1$, G is singular, and there is at least one eigenvalue at zero.

Proof: When $\alpha + \beta = 1$, the last row of G is $1-\beta$ times the penultimate row of G.

Prop. 4: If G is the state transition matrix of a stable system, the steady-state values of $\vec{x}(t)$ (denoted by $\vec{x_{ss}}(t)$) are given by

$$\vec{x_{ss}} = (I-G)^{-1}\vec{h}\,r_g\,.$$

Proof: Simply set $\vec{x}(t) = \vec{x}(t-1) = \vec{x}_{ss}$ in

$$\vec{x}(t) = G\vec{x}(t-1) + \vec{h}\,r_g\,.$$

Prop. 5: If the system is stable, all rates converge to the same value, i.e.,

$$r_{ss} = \frac{\beta r_g}{\alpha + K\beta}\,.$$

Proof: Let $\vec{x}_{ss} = r_{ss}[1, 2, \ldots, K]$, then each line of $(I - G)\vec{x}_{ss} = \vec{h}r_g$ is $(\alpha + K\beta)r_{ss} = \beta r_g$.

The following two properties have not been proven; however, experience with computing the eigenvalues of G for a large number of trials with various values of the parameters, particularly K, gives us cause to conjecture.

Conjecture 1: When $\alpha + \beta = 2$, all K eigenvalues have magnitude 1.

Conjecture 2: For $\alpha + \beta < 2$, matrix G is the state transition matrix of a stable system.

Notice that the stability boundary conjectured differs from the stability boundary for the synchronized algorithm by missing K, i.e., being independent of the unknown number of nodes. Fig. 4 is a plot of the magnitudes of the 800 eigenvalues of G when $K = 800$; $\alpha = 0.1$; $\beta = 1.8$. The magnitudes of the eigenvalues approaches but does not reach 1.

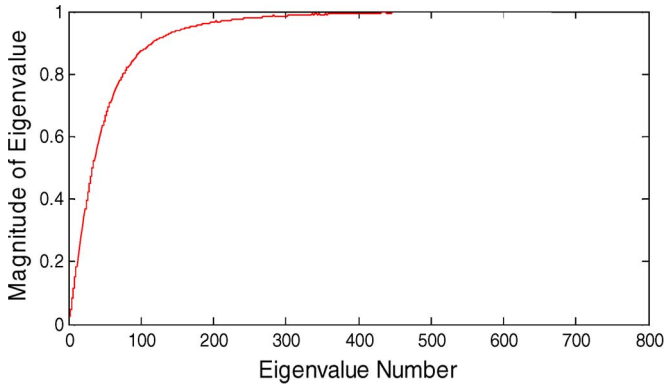


Fig. 4. Magnitude of the eigenvalues for the sequential update case with 800 nodes.

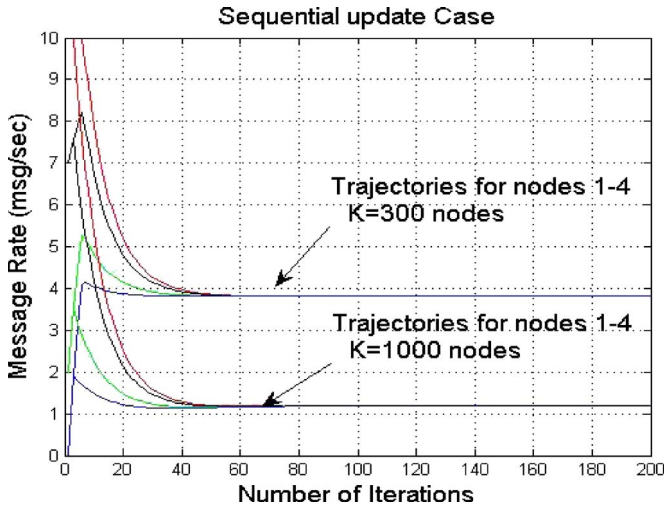


Fig. 5. LIMERIC for total nodes $=$ 300 and 1000 for sequential update case.

Fig. 5 shows the more robust convergence that LIMERIC experiences under the sequential update regime for K equals 300 and 1000 (four selected nodes with initial message rates of 10, 7, 2, and 0 msg/sec are shown). Using the same parameters as the configuration in Figs. 2 and 3, i.e., with inequality (6) not satisfied, it shows a convergence trajectory for a subset of nodes. To explore the convergence properties of this case, the updates do not use gain saturation, i.e., they follow (2), not (15). While this scenario produces nonconvergent behavior in the synchronous update case in Fig. 2, for sequential updates, the algorithm converges. Gain saturation was introduced in Section III-B to avoid divergent behavior for any K. By contrast, the enhanced convergence properties observed for sequential updates are expected to have a limit. Hence, gain saturation should still be applied in practice.

D. Implementation in a Distributed System (Using CBF)

The analysis in Section II utilizes $r_C(t)$, which is the sum of the individual vehicles' current message rates $r_j(t)$. The earlier analysis assumes that $r_C(t)$ is available at each vehicle. This could be accomplished, with nontrivial complexity, by having each node include its current rate $r_j(t)$ in its safety message

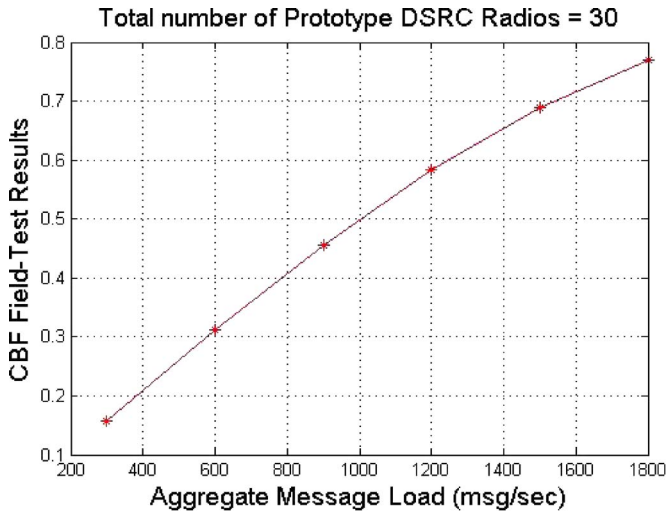


Fig. 6. CBF results using field test with actual DSRC radios.

broadcasts, although the unreliable channel would make the aggregate rate thus obtained a noisy estimate of the exact $r_C(t)$. A simpler approach takes advantage of a readily available measure from which $r_C(t)$ can be derived. The IEEE 802.11-2012 standard [3] defines the *CCA* function to determine at any given time whether the channel is classified as idle or busy. This CCA function is measured in every compliant IEEE 802.11 chip. The idle or busy state can be tracked over an interval of time to determine the ratio of time that the channel is busy. In this paper, the interval measure is referred to as the *channel busy fraction (CBF)*, and it is a function of time t. CBF is also referred to in other documents as the channel busy ratio or the CCA busy fraction.

A well-understood property of the CSMA mechanism used in MAC protocols such as IEEE 802.11 is that there is a one-to-one relationship between the "offered load" of a set of nodes and the resulting CBF [22]. Offered load is the aggregate quantity of data transmitted on the channel. In the V2V safety communication regime, where each message is broadcast, and there are no acknowledgment frames or retransmissions, offered load is equivalent to $r_C(t)$. Therefore, there exists a one-to-one function that maps from $CBF(t)$ to $r_C(t)$.

The precise mapping function depends on a number of things, including the number of transmitters, the distribution of packet lengths, and the distribution of packet transmission times. In the case of V2V safety communication, the sources are nearly periodic, and the packet lengths are approximately constant. As shown in [23], under these conditions and if the number of vehicles K is at least 30, the mapping function is relatively insensitive to K, i.e., the aggregate offered load is more important than the number of vehicles it is divided among. Fig. 6 shows obtained CBF field-test results versus aggregate message rate load. For example, a measured CBF of 0.58 corresponds approximately to 30 nodes sending 40 msg/sec or 1200 aggregate msg/sec. In practice, the mapping function can be modeled as accurately as is desired. NS-2 simulation results presented in Section IV illustrate the effect of implementing LIMERIC by using measured $CBF(t)$ as an input, rather than measuring $r_C(t)$ directly.

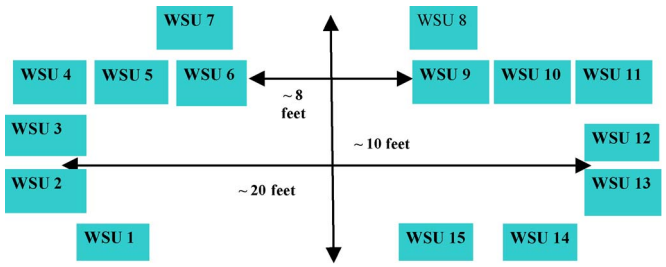


Fig. 7. Physical WSU layout in the testbed; each WSU has two DSRC radios as in [23].

Although global fairness for a larger network is out of the scope of this paper, here, we provide a brief discussion of this issue. Some research is reported in [11] demonstrating global fairness in a larger network that uses a binary-feedback-based message rate control. Global fairness means that nodes contributing to congestion at a given point within their respective interference ranges participate fairly in controlling that congestion, even if not all of the nodes are within one-hop communication range of each other (i.e., some are hidden). In [11], global fairness is achieved via a multihop exchange of additional protocol information among the vehicles. The authors have observed similar fairness when LIMERIC is coupled with this type of information exchange, for example, each node's measured CBF [29], although the results are not reported here. We plan to study this aspect of LIMERIC in more detail as future work.

IV. Numerical Results

Here, we present numerical results for LIMERIC using the NS-2 simulator for synchronous updates for more realistic scenarios. NS-2 implementation is done for the synchronous update case where all nodes measure the CBF at the end of a monitoring period and then update their message rate. The monitoring period in the simulations is set to be 200 ms. The parameters used for DSRC simulations are as follows: transmit power $= 20$ dBm, enhanced distributed channel access parameters AIFSN $= 6$, and $\mathrm{CW}_{\min} = 7$ (AC0 from IEEE 1609.4-2006), over the air message size $= 378$ bytes, data rate $= 6$ Mb/s, channel bandwidth $= 10$ MHz, carrier-sense threshold $= -92$ dBm, and reception threshold $= -92$ dBm. The noise floor was set to be -99 dBm, and the preamble capture has been enabled with a capture threshold set to 4 dB. Notice that a 6-Mb/s channel has capacity for approximately two thousand 378-byte messages. The nodes are placed such that there are few hidden nodes, using the exact topography and transmit attenuation values reported in [8]. The topography was constructed by placing 15 DSRC dual-radio DENSO wireless safety unit (WSU) systems close to each other (see Fig. 7). Details of the NS-2 implementation are discussed in [23]. The target CBF is set to 0.6. For LIMERIC, the values of α, β, and X are again set to 0.1, 1/150, and 1 Hz, respectively. The maximum and minimum message rates for a node are set to 10 and 0 msg/sec, respectively. We present CBF and message rate results for varying number of nodes from 90 to 300.

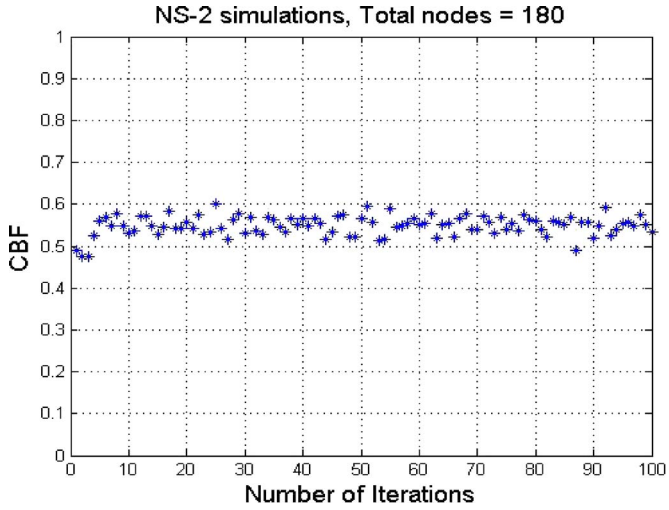


Fig. 8. CBF plot for LIMERIC in NS-2 with total nodes $= 180$.

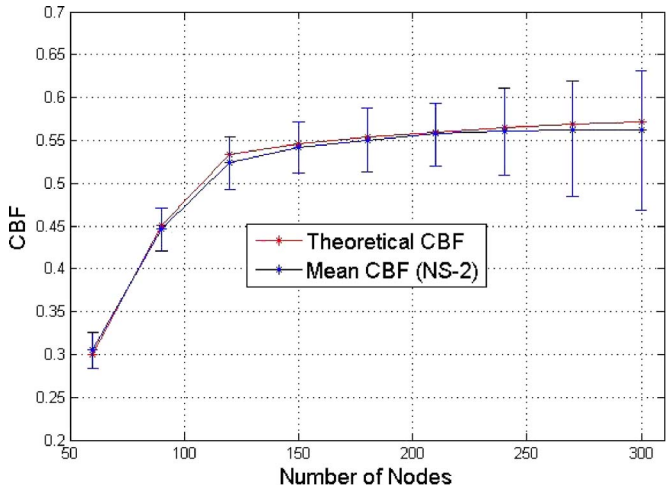


Fig. 9. Mean and 90% range CBF results using NS-2 simulator.

In Fig. 8, we plot the CBF for 180 nodes. It can be observed that CBF converges approximately to a range between 0.5 and 0.6. Further, in Fig. 9, the mean and 90% range CBF (represented by error bars) are presented to quantify the variation. It can be observed in the figure that the CBF obtained from NS-2 simulations is very close to the theoretical CBF obtained by mapping r_C computed by (14). In Figs. 10 and 11, we plot the message rate curves for $K = 180$ and 300, respectively. We have shown the message rates for two representative vehicles with different initial message rates. It can be observed from the plots that the message rates converge very close to rate r_j (scaled to messages per second) predicted by the mathematical analysis in (7). It should be noted that in NS-2, CBF is measured over a finite time, and hence, it has measurement noise that leads to oscillations in steady state. Detailed analysis is provided in Section V. This demonstrates that LIMERIC also converges in a fair and efficient manner for realistically noisy vehicular network scenarios simulated by NS-2. It is also interesting to see in Fig. 11 that for a highly loaded scenario of 300 nodes, for which inequality (6) is not satisfied, the nodes converge to message rates of approximately 4 msg/sec, albeit with a bit higher variation due to the invocation of gain

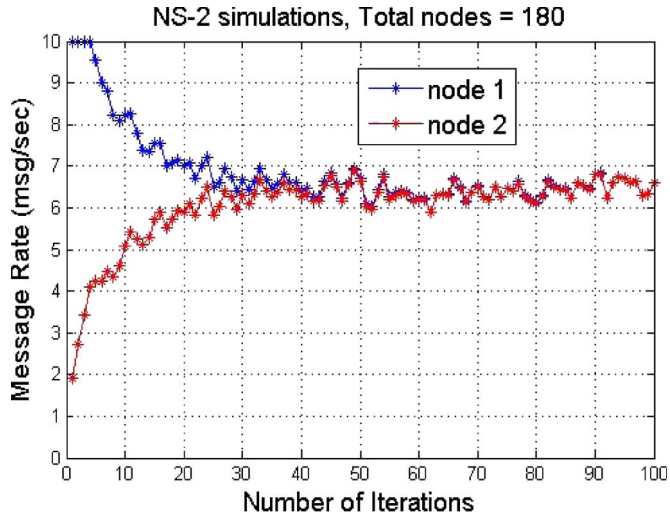


Fig. 10. Message rate plot for LIMERIC in NS-2 with total nodes = 180.

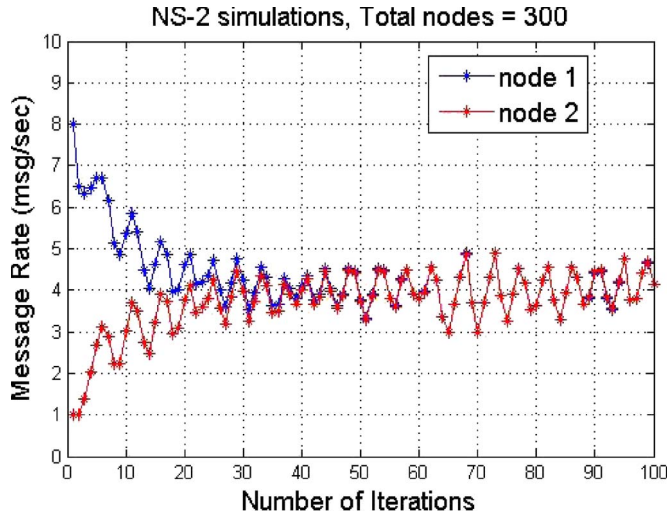


Fig. 11. Message rate plot for LIMERIC in NS-2 with total nodes = 300.

saturation. In these figures, we can also see that each rate converges to within approximately 10% of its steady-state mean after about 20 iterations, and by 40 iterations, the impact of differentiated initial conditions can no longer be observed.

V. Noisy Rate Measurements

A node will typically not measure the total rate given by (1) with perfect accuracy, whether the measurement is indirectly via CBF or through some other means. In practice, there will always be noise associated with this measurement. This section models the effect of such inaccuracies on LIMERIC with synchronous updates. A measurement inaccuracy at node j at time t can be captured by adding a zero-mean white noise sequence, i.e., $w_j(t)$, to the rate. In the first section, we assume that the noise is independent between nodes such that $E[w_j(t)w_i(t)] = 0$ for $j \neq i$. In the following section, we consider the case where all nodes see the same measurement noise at a given time. The analyses are very similar, with the independent case slightly more general.

A. Measurement Noise Independent Across Nodes

We denote the variance of the measurement noise process as σ^2. With measurement noise, (2) becomes

$$r_j(t) = (1-\alpha)r_j(t-1) + \beta\left(r_g - r_C(t-1)\right) + \beta w_j(t-1). \tag{19}$$

Define the vector

$$\vec{w}(t) = \left[\, w_1(t) \quad w_2(t) \quad \ldots \quad w_K(t) \,\right]^T. \tag{20}$$

Then, (4) becomes

$$\vec{r}(t) = A\vec{r}(t-1) + br_g + \beta I\vec{w}(t-1). \tag{21}$$

Equation (21) is a linear difference equation, and the effects of the constant input and the stochastic input may be separately considered, with the constant input affecting only the mean of the rate vector and the zero-mean stochastic input affecting only the covariance of the rate vector. Assume that inequality (6) is satisfied to produce a convergent system, then there exists a steady state, and in steady state, the rates converge to random variables with mean

$$E[r_j] = \frac{\beta r_g}{(\alpha + K\beta)}, \qquad j = 1, 2, \ldots, K, \tag{22}$$

the same as the deterministic convergence rate in Section II [see (7)]. The steady-state variance in r_j is determined by the mean square values in the following zero-mean system:

$$\vec{r}(t) = A\vec{r}(t-1) + \beta I\vec{w}(t-1). \tag{23}$$

Let

$$C(t) = E\left[\vec{r}(t)\vec{r}^T(t)\right] \tag{24}$$

be the covariance matrix for the rate vector. From (23), we have

$$C(t) = AC(t-1)A^T + \sigma^2\beta^2 I. \tag{25}$$

With parameters satisfying inequality (6), the system converges, and there is only one positive semidefinite solution to the following algebraic discrete-time Lyapunov equation for the steady-state covariance matrix C

$$C = ACA^T + \sigma^2\beta^2 I. \tag{26}$$

This equation can be solved by solving the K^2-dimensional linear system of equations, i.e.,

$$(I - A \otimes A)\,\mathrm{vec}(C) = \sigma^2\beta^2\,\mathrm{vec}\,(I) \tag{27}$$

where $\otimes$ is the matrix Kronecker product operator producing a $K^2 \times K^2$ matrix, and the operator vec(.) lines up the columns of a $K \times K$ matrix into a K^2-dimensional column vector [26], [27].

Equation (27) is a linear matrix equation with only one solution corresponding to a positive definite (symmetric) matrix C. Thus, if any such C we postulate can be verified as a solution, we have solved the equation. We use the symmetry of the problem to postulate the following properties of the

steady-state covariance: 1) all diagonal elements of C, which are the variances of the individual rates, are equal; and 2) all the nondiagonal elements of C, which are the cross covariances of the individual rates, are equal. By imposing this structure on C and writing out (27) for various K, one can discover that there are only two distinct equations within (27), i.e., one repeated K times and one repeated $K^2 - K$ times. Thus, we are left with two linear equations in two unknowns. Let $\vec{c}$ be a 2×1 vector built by stacking a diagonal element of C and an off-diagonal element of C, which are denoted by c_1 and c_2, respectively, i.e., $\vec{c} = [c_1, c_2]^T$. Then

$$Y\vec{c} = \vec{d} \quad \text{so that} \quad \vec{c} = Y^{-1}\vec{d}. \tag{28}$$

The elements of the matrix and constant vector of (28) are quite complex, making it possible but very messy to write down the solution directly. We give these elements here as follows:

$$\mathrm{Y} = \begin{bmatrix} 1-P^2-Q\beta^2 & 2\beta P - R\beta^2 \\ 2Q\beta P - QR\beta^2 & 1-P^2-(K^2-3K-3)\beta^2+2R\beta P \end{bmatrix}$$
$$\vec{d} = \begin{bmatrix} \beta^2\sigma^2 \\ 0 \end{bmatrix}$$

where $P = 1 - \alpha - \beta$, $Q = K - 1$, and $R = K - 2$.

The variance of the sum of the elements of any random vector is given by the sum of all the elements of its covariance matrix, so that the variance of the sum of rates r_C can be computed as

$$\mathrm{Var}(r_C) = K\left(c_1 + (K-1)c_2\right). \tag{29}$$

The evolution of the variance of the sum of the rates in time can be viewed more directly and simply. Consider premultiplying (23) by a row vector whose elements are all equal to 1. This produces an equation $r_C(t)$ as defined in (1). When A is premultiplied by a row vector whose elements are all equal to 1, the result is a row vector whose elements all equal the same value, i.e., $1 - \alpha - K\beta$. The resulting equation for the sum of rates is

$$r_C(t) = (1 - \alpha - K\,\beta)r_C(t-1) + \beta\, w(t-1), \tag{30}$$

where $w(t)$ is the sum of $w_j(t)$ so that it is a scalar process with zero mean and variance $K\sigma^2$.

To find the variance of the sum of the rates, we square (30), take expectations, and apply the steady-state condition that the variance of the sum of the rates approaches a constant, i.e.,

$$\begin{aligned} \mathrm{Var}[r] &= \frac{\beta^2}{1-(1-\alpha-K\,\beta)^2}\,K\sigma^2 \\ &= \frac{\beta^2}{(\alpha+K\,\beta)(2-\alpha-K\,\beta)}\,K\,\sigma^2. \end{aligned} \tag{31}$$

This equation has the expected property that, as the parameters move toward the convergence boundary of inequality (6), the variance of the sum of the rates becomes unbounded. The equation is not valid when inequality (6) is not satisfied. As an example, with $K = 4$, $\alpha = 0.1$, $\beta = 0.2$, and $\sigma^2 = 1$, (28) produces

$$c = \begin{bmatrix} 0.167995746943115 \\ -0.042530568846358 \end{bmatrix}.$$

With this, the variance of the sum from (29) is $\mathrm{Var}[r_C] = 0.161616$, which matches the result of (31).

Notice that the rates are negatively correlated. When one rate is high, presumably from a recent negative noise sample, the other rates tend to be low from having reacted to the high rate in the previous time step. Because of this, the variance of the total rate is much lower than K times the variances of an individual rate.

B. Rate Measurement Noise Is the Same for Each Node

It may be a better model that many or all nodes see the same noise sample at each given time. The convergence condition and the mean value analysis is unaffected by this change in how the noise enters the system. If all the nodes see the same noise sample, (23) is replaced by

$$\vec{r}(t) = A\vec{r}(t-1) + \beta\vec{u}w_1(t-1), \tag{32}$$

where $\vec{u}$ is a vector of all 1's, and $w_1(t-1)$ is a scalar white noise sequence with variance σ^2. Equation (25) is now replaced by

$$C(t) = AC(t-1)A^T + \sigma^2\beta^2 U, \tag{33}$$

where U is a $K \times K$ matrix of all 1's. Equation (26) is similarly changed, and (27) becomes

$$(I - A \otimes A)\mathrm{vec}(C) = \sigma^2\beta^2\vec{u}_{K^2}, \tag{34}$$

where $\vec{u}_{k^2}$ is a vector of all 1's with length K^2.

Now, either from symmetry in the problem or from the structure of (34), we can postulate that all the elements of C are identical, reducing (34) to one equation repeated K^2 times. With some algebraic effort, this equation can be reduced to

$$\begin{aligned} \left[1-(1-\alpha-Kp)^2\right]\mathrm{Var}[r_i] &= \sigma^2 \\ \mathrm{Var}[r_i] &= \frac{\beta^2\sigma^2}{1-(1-\alpha-K\beta)^2}. \end{aligned} \tag{35}$$

Since all elements of C are identical, we have $\mathrm{Var}[r_i] = \mathrm{Cov}(r_i, r_j)$. Using (28), the variance of the sum of the rates is given by

$$\mathrm{Var}[r_C] = \frac{K^2\beta^2\sigma^2}{1-(1-\alpha-K\beta)^2}. \tag{36}$$

Since all nodes react the same way to a given noise sample, instead of adjusting to each other's noise samples, the variance of the sum of the rates is K^2 times as large as the case when the nodes see independent noise.

If we were to use the evolution of the variance of the sum of the rates directly as we did to develop (30), we would arrive at

$$r_C(t) = (1 - \alpha - K\beta)r(t-1) + \beta K w_1(t-1) \tag{37}$$

where $w_1(t-1)$ is the same scalar white noise sequence of (32), with variance σ^2. K appears here since we are dealing with the sum of K identical noise samples, not the sum of K independent noise samples.

TABLE I
FIXED-RATE RESULTS TO OBTAIN MEASUREMENT NOISE σ^2

Number of Nodes	Mean Message Rate (Hz)	Mean CBF	Experimental CBF variance (σ^2)
120	9.19	.538	$2.305*10^{-4}$
150	7.58	.550	$2.551*10^{-4}$
180	6.45	.557	$2.640*10^{-4}$
210	5.61	.563	$3.016*10^{-4}$

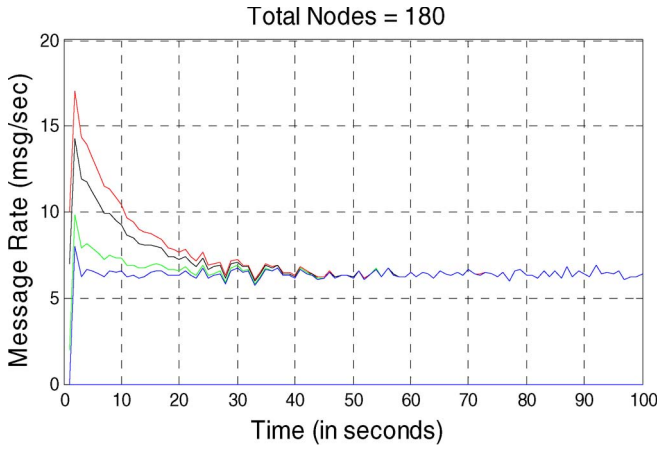


Fig. 12. MATLAB simulation with all 180 nodes seeing the same noise.

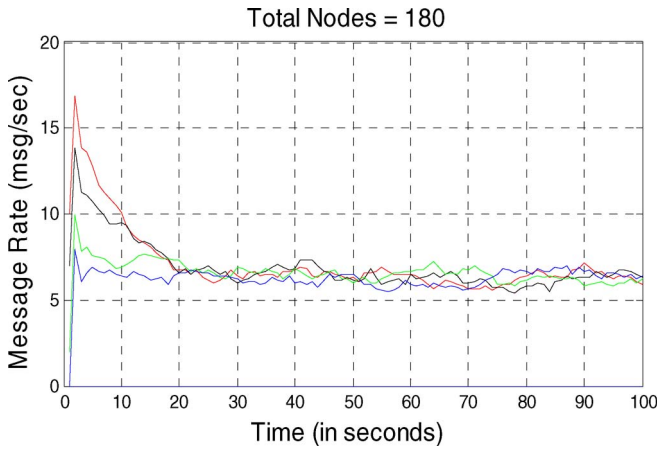


Fig. 13. MATLAB simulation with each of 180 nodes seeing independent noise samples.

Solving for the variance of r_C using (37) results in (35) as it must.

MATLAB simulations of the difference equations are now presented for two different noisy situations, using the following parameters: $\alpha = 0.1$; $K = 180$; $\beta = 1/150$; $\sigma^2 = 2.6^*10^{-4}$; and $r_g = 0.6$. The variance σ^2 was determined by equivalent NS-2 simulations (see Table I). Note that in Figs. 12 and 13, the individual node message rates are shown in units of msg/sec, i.e., they are not normalized.

In the first case, all nodes see that same noise sample at each time instant so that (35) and (36) attain. This case most closely models what is seen in the noise that occurs in the NS-2 simulator.

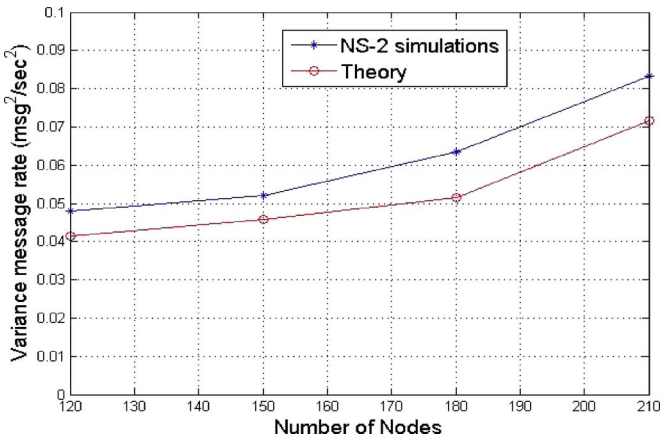


Fig. 14. Variance in message rate for LIMERIC implementation.

Fig. 12 shows how, in the first hundred seconds of simulation, i.e., (5 updates/sec), four nodes starting with different rates converge and then run in lockstep, including the reacting identically to the noise. The measured steady-state covariance matrix has the correct structure, i.e., all elements are identical. The mean message rate of each node converges to 6.4 msg/sec with variance 0.05. The mean total *normalized* rate r_C converges to 0.55 with variance $4 * 10^{-4}$. All of these results match the theoretical values well.

In the second case, the noise samples are independent from one node to another. Fig. 13 shows that, in this case, the nodes converge to the same mean and variance, but they are no longer in lockstep. The mean and variance of each node's rate is essentially the same as the previous case, as is the mean total normalized rate. However, as predicted, the variance of the total normalized rate is much smaller, measuring $2 * 10^{-6}$ as the theory predicts. This is due to the negative covariance between different nodes.

In a realistic LIMERIC implementation, similar to that simulated in Section IV, perhaps the largest source of measurement noise is the short CBF measurement interval. Since all nodes measure CBF over the same interval, the dependent noise analysis of this section is most relevant. Fig. 14 shows a comparison between the variance in total rate measured in NS-2 and predicted in theory. In this case, the variance is shown unnormalized, i.e., expressed in units of $(\text{msg/sec})^2$, and the given theoretical equations are adjusted accordingly. The simulated total rate variance (blue curve) is obtained via NS-2 simulations for various K. To compute the theoretical variance (red curve) using (36), we need an estimate of the input noise variance σ^2 caused by CBF measurement. We obtained that through a series of fixed rate simulations, i.e., LIMERIC with adaptation disabled. In each case, the message rate of each node was fixed to the mean value measured in a separate simulation when LIMERIC was allowed to adapt for that K.

Table I shows the fixed rate and measured CBF variance for each K. Note that the CBF variance is relatively insensitive to K, since mean CBF is almost constant due to LIMERIC. Using the σ^2 values from Table I, we compute the theoretical total rate variance from (36), as shown in the red curve in Fig. 14. The comparison between theoretical and simulated total rate variance is fairly close, with simulated values consistently a

bit higher, probably due to an additional σ^2 noise component not captured in the experimental CBF variance. Note also that these variance values are quite low, indicating that the α and β parameter choices are effective for this range of K.

VI. Analysis With Delay

Next, consider the effect of d units of delay in each node's total channel rate measurement $r_C(t)$. Delay in the rate measurement can occur for a variety of reasons. For example, [11] presents a message dissemination protocol whose purpose is to achieve wide-area fairness. This protocol can be applied to LIMERIC, with some modification to employ the channel load information obtained from neighboring nodes (not within the scope of this paper). The message dissemination process necessarily involves delay. In addition, the CBF measurement process itself can be viewed as incurring delay. The change in rate at a given sender is not immediately observable at a receiver, but rather is only observable when the sender transmits messages at the new rate. To expose the underlying convergence characteristics of the algorithm, the analysis here assumes that gain saturation is disabled. Independent of the delay, gain saturation will guarantee convergence to at least a limit cycle, as shown in simulation results below.

With a delayed total rate, (2) becomes

$$r_j(t) = (1-\alpha) r_j(t-1) + \beta \left(r_g - r_C(t-d)\right), \qquad (38)$$

where $r_C(t)$ is given by (1), $0 < \alpha < 1$, $\beta > 0$. The analysis up to this point assumed $d = 1$. Since all nodes have the same delay, the difference between the rates of any two nodes remains the same, and fairness is preserved. Thus

$$r_m(t) - r_n(t) = (1-\alpha)\left(r_m(t-1) - r_n(t-1)\right) \qquad (39)$$

or

$$r_m(t) - r_n(t) = (1-\alpha)^t \left(r_m(0) - r_n(0)\right). \qquad (40)$$

However, now, the update of the overall rate $r_C(t)$ evolves according to

$$r_C(t) = (1-\alpha) r_C(t-1) - K\beta r_C(t-d) + K\beta r_g. \qquad (41)$$

Taking Z-transforms

$$\left(1 - (1-\alpha)z^{-1} + K\beta z^{-d}\right) r_C(z^{-1}) = K\beta r_g(z^{-1}). \qquad (42)$$

The convergence of rates to each other is still given by (9), but the convergence of the total rate depends on the roots of the following equation:

$$z^d - (1-\alpha) z^{d-1} + K\beta = 0. \qquad (43)$$

The total channel rate converges to a steady-state value if the roots of (43) all remain within the unit circle. The convergence region in β of this system depends not only on K but also on the delay, i.e., d. Taking $\alpha = 0.1$, we can compute the following limits for convergence that now replace inequality (6), using the $\beta > 0$ assumption.

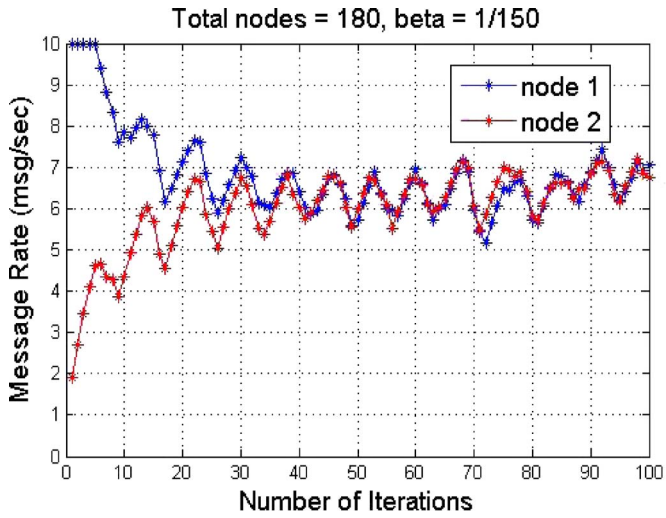


Fig. 15. Message rate plot for LIMERIC with one extra monitoring period delay ($d = 2$).

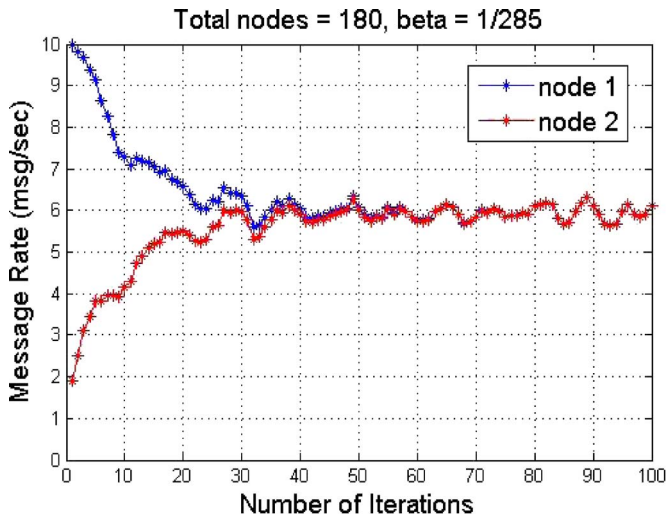


Fig. 16. Message rate plot for LIMERIC with one extra monitoring period delay ($d = 2$) and smaller β.

For $d = 1$, we need $K\beta < 1.9$. (This is the same as the previously derived case in Section II.)
For $d = 2$, we need $K\beta < 1.0$.
For $d = 3$, we need $K\beta < 0.64$.
For $d = 4$, we need $K\beta < 0.48$, and so on.

If the system converges, the Section II analysis leading to the steady-state convergence results in (7) and (12) still holds, independent of delay. Thus, as the delay increases, adaptation gain β must be kept smaller to ensure convergence; however, if convergence is obtained, the steady-state rates are the same regardless of delay. For any given adaptation gain, the convergence of the overall rate will be slower as the delay is increased since the roots of (43) will be closer to the unit circle.

Figs. 15 and 16 show NS-2 simulation results for the case that the CBF measured in one interval is delayed one extra interval before it is used in the LIMERIC update, i.e., $d = 2$. The configuration simulated in Fig. 15 is identical to that in Fig. 10 in Section IV, except for the delay. In particular, $K = 180$, $\alpha = 0.1$, and $\beta = 1/150$. For $d = 1$, these parameters

are within the convergence constraint of inequality (6), i.e., $\alpha + K\beta = 1.3 < 1.9$. As shown in Fig. 10, the algorithm converges with small steady-state variance. However, for $d = 2$, convergence requires $\alpha + K\beta < 1.0$, and these parameters are too high. Fig. 15 shows that for the $d = 2$ case, gain saturation is invoked, leading to a steady-state limit cycle. We can regain low-variance steady-state convergence by reducing β such that $\alpha + K\beta < 1.0$. For example, if we set $\beta = 1/285$, moving the convergence boundary back to $K = 285$ (the same as in the Fig. 10 configuration), we obtain the smooth rate behavior shown in Fig. 16.

VII. Conclusion and Future Work

This paper has defined a new linear message rate congestion control algorithm, i.e., LIMERIC, which is designed for DSRC safety communication. The linear control allows LIMERIC to converge perfectly in a noiseless environment, which provides a significant improvement over the limit cycle convergence behavior inherent in traditional binary control algorithms. It also avoids fairness problems observed in some binary controls. The convergence properties are proved initially for the noiseless case. Gain saturation is introduced to the adaptation to provide robust convergence independent of number of vehicles, and results are provided to illustrate the stabilizing effect. This paper has also provided analyses of the following extended cases: sequential updates, updates with noise in the total rate measurement, and updates with delayed total rate measurements. We have also discussed practical considerations related to the determination of the total message rate via a channel busy measurement, arguing that a one-to-one mapping function is available and that approximations of that function are likely to improve convergence by underestimating the total rate in extremely high density cases. Finally, this paper has presented numerical results to illustrate the behavior of LIMERIC under a variety of challenging operating scenarios, including high density, noisy inputs, and delayed inputs. LIMERIC is easily implementable and can be an important tool in controlling channel congestion for DSRC safety and other environments.

We have identified several avenues for future work. First, in Section III-A, we provide guidance regarding the selection of values for LIMERIC parameters α and β, but further research could lead to optimization of these choices with respect to tradeoffs concerning convergence speed, stability, and the variation of equilibrium channel load as a function of K. Second, with regard to maintaining stability in the face of unknown and changing K, we believe that the approach with a separate slow adaptation loop for β is promising and should be investigated. A third important area of future work is to thoroughly investigate extensions of LIMERIC that achieve "global fairness" in large networks where vehicles sense different channel loads. These questions, which were specifically out of the scope of this paper, include the definition of global fairness, strategies for decentralized sharing of information over relevant areas, and mathematical analyses of resulting extensions to LIMERIC. A fourth avenue of future work is to consider appropriate means of providing differentiated transmission opportunities while controlling congestion, e.g., to react to differences in vehicle dynamics, channel fading, or message priority. Finally, while the version of LIMERIC presented in this paper controls congestion entirely via the message rate, we believe that this control can be augmented by appropriate invocation of other controls, principally of transmit power, and therefore, we have identified joint rate and power control as a topic for future work.

References

[1] "Dedicated short range communications report and order," US Fed. Commun. Commission, Washington, DC, USA, R&O FCC 03-324, Dec. 17, 2003.

[2] D. Jiang, V. Taliwal, A. Meier, W. Holfelder, and R. Herrtwich, "Design of 5.9 GHz DSRC-based vehicular safety communication," *IEEE Wireless Commun.*, vol. 13, no. 5, pp. 36–43, Oct. 2006.

[3] *IEEE Std. for Local and Metropolitan Area Networks: Wireless LAN MAC and PHY Specification*, IEEE Std. 802.11-2012, 2012.

[4] *Amendment to IEEE 802.11: Wireless Access in Vehicular Environments*, IEEE Std. 802.11, Jul. 2010.

[5] *IEEE Standard for Wireless Access in Vehicular Environments—Networking Services*, IEEE Std. 1609.3-2010, Dec. 2010.

[6] *IEEE Standard for Wireless Access in Vehicular Environments—Security Services*, IEEE Std. 1609.2-2013, Apr. 2013.

[7] "Vehicle Safety Communications—Applications VSC-A, Final Report," U.S. Dept. Transp., Nat. Highway Traffic Safety Admin., Washington, DC, USA, Rep. DOT HS 811 492A, Sep. 2011.

[8] A. Weinfield, "Methods to reduce DSRC channel congestion and improve V2V communication reliability," in *Proc. ITS World Congr.*, Busan, Korea, 2010, pp. 1–12.

[9] C.-L. Huang, Y. P. Fallah, R. Sengupta, and H. Krishnan, "Adaptive intervehicle communication control for cooperative safety systems," *IEEE Netw.*, vol. 24, no. 1, pp. 6–13, Jan./Feb. 2010.

[10] M. Torrent-Moreno, J. Mittag, P. Santi, and H. Hartenstein, "Vehicle-to-vehicle communication: Fair transmit power control for safety-critical information," *IEEE Trans. Veh. Technol.*, vol. 58, no. 7, pp. 3684–3703, Sep. 2009.

[11] T. Tielert, D. Jiang, Q. Chen, L. Delgrossi, and H. Hartenstein, "Design methodology and evaluation of rate adaptation based congestion control for vehicle safety communications," in *Proc. IEEE VNC*, Nov. 2011, pp. 116–123.

[12] A. Weinfied, J. Kenney, and G. Bansal, "An adaptive DSRC message transmission interval control algorithm," in *Proc. ITS World Congr.*, Paper Number 1171, Oct. 2011.

[13] *Dedicated Short Range Communications (DSRC) Message Set Dictionary*, SAE Std. J2735, Nov. 2009.

[14] *Draft DSRC Message Communication Minimum Performance Requirements—Basic Safety Message for Vehicle Safety Applications*, SAE Draft Std. J2945.1, Oct. 2010, Revision 2.0, SAE International, DSRC Committee.

[15] Project Order 0004, Technical Proposal Statement of Work U.S. Dep. Transp., Nat. Hwy. Traffic Safety Admin., Cooperative Agreement DTNH22-05-01277 Interoperability Issues of Vehicle-to-Vehicle Based Safety Systems Project (V2V-Interoperability), Dec. 2009.

[16] J. Kenney, Ed., *Traffic Management Specification, Version 4.1*. Mountain View, CA, USA: ATM Forum, Mar. 1999, AF-TM 4.1, af-tm-0121.000.

[17] Recommendation I.370, ITU-T I.370 Integrated Services Digital Network (ISDN) Overall Network Aspects and Functions ISDN User-Network Interfaces, Congestion Management for the ISDN Frame Relaying Bearer Service 1991, Recommendation I.370, ITU-T I.370.

[18] S. Gorinsky and H. Vin, "Additive increase appears inferior," Dept. Comput. Sci., Univ. Texas Austin, Austin, TX, USA, Tech. Rep. 2000-18, May 2000.

[19] D. P. Atherton, *Analysis Methods, The Control Handbook*. Piscataway, NJ, USA: IEEE Press, 1996, ch. 19.

[20] M. Allman, V. Paxson, and W. Stevens, TCP Congestion Control, IETF RFC 2581, Apr. 1999.

[21] K. Ramakrishnan, S. Floyd, and D. Black, The Addition of Explicit Congestion Notification (ECN) to IP, IETF RFC 3168, Sep. 2001.

[22] D. Bertsekas and R. Gallager, *Data Networks,* 2nd ed. Englewood Cliffs, NJ, USA: Prentice-Hall, 1992.

[23] G. Bansal, J. Kenney, and A. Weinfield, "Cross-validation of DSRC radio testbed and NS-2 simulation platform for vehicular safety communications," in *Proc. 4th Int. Symp. WIVEC*, Sep. 2011, pp. 1–5.

[24] S. Subramanian, M. Werner, S. Liu, J. Jose, R. Lupoaie, and X. Wu, "Congestion control for vehicular safety: Synchronous and asynchronous MAC algorithms," in *Proc. 9th ACM Int. Workshop Veh. Inter-Netw., Syst., Appl.*, Jun. 2012, pp. 63–72.
[25] *Decentralized Congestion Control Mechanisms for Intelligent Transport Systems Operating in the 5 GHz Range: Access Layer Part*, ETSI TS 102 687 v1.1.1, Jul. 2011.
[26] P. Lancaster and M. Tismenetsky, *The Theory of Matrices,* 2nd ed. Orlando, FL, USA: Academic, 1985.
[27] G. Kitagawa, "An algorithm for solving the matrix equation $X = FxF^T + S$," *Int. J. Control*, vol. 25, no. 5, pp. 745–753, 1977.
[28] G. Bansal and J. Kenney, "Achieving weighted-fairness in message rate-based congestion control for DSRC systems," in *Proc. 5th Int. Symp. WIVEC*, Jun. 2013.
[29] G. Bansal, H. Lu, J. Kenney, and C. Poellabauer, "Error model based adaptive rate control for vehicle-to-vehicle communications," in *Proc. 10th ACM Int. Workshop VANET*, Jun. 2013, pp. 41–50.
[30] C. Lochert, B. Scheuermann, and M. Mauve, "A survey on congestion control for mobile ad-hoc networks," *Wireless Commun. Mobile Comput.*, vol. 7, no. 5, pp. 655–676, Jun. 2007.
[31] K. Chen, K. Nahrstedt, and N. Vaidya, "The utility of explicit rate-based flow control in mobile ad hoc networks," in *Proc. IEEE WCNC*, Mar. 2004, pp. 1921–1926.
[32] S. Ni, Y. Tseng, Y. Chen, and J. Sheu, "The broadcast storm problem in a mobile adhoc network," in *Proc. ACM/IEEE Int. Conf. MOBICOM*, 1999, pp. 151–162.
[33] F. Farnoud and S. Valaee, "Reliable broadcast of safety messages in vehicular ad hoc networks," in *Proc. IEEE INFOCOM*, Apr. 2009, pp. 226–234.
[34] F. Yei, R. Yim, J. Zhang, and S. Roy, "Congestion control to achieve optimal broadcast efficiency in VANETs," in *Proc. IEEE ICC*, May 2010, pp. 1–5.
[35] F. Yei, R. Yim, J. Guo, J. Zhang, and S. Roy, "Prioritized broadcast contention control in VANET," in *Proc. IEEE ICC*, May 2010, pp. 1–5.
[36] N. Nasiriani, Y. P. Fallah, and H. Krishnan, "Stability analysis of congestion control schemes in vehicular ad-hoc networks," in *Proc. CCNC 2013*, pp. 358–363.
[37] T. Tielert, D. Jiang, H. Hartenstein, and L. Delgrossi, "Joint power/rate congestion control optimizing packet reception in vehicle safety communications," in *Proc. 10th ACM Int. Workshop VANET*, Jun. 2013, pp. 51–60.
[38] Y. P. Fallah, C.-L. Huang, R. Sengupta, and H. Krishnan, "Analysis of information dissemination in vehicular ad-hoc networks with application to cooperative vehicle safety system," *IEEE Trans. Veh. Technol.*, vol. 60, no. 1, pp. 233–247, Jan. 2011.

Gaurav Bansal (M'07) received the B.Tech. degree from the Indian Institute of Technology, Kanpur, India and the Ph.D. degree from the University of British Columbia, Vancouver, BC, Canada.

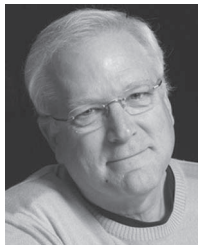
From August 2007 to July 2008, he was a Research Intern with Mercedes Benz Research and Development North America Inc., Palo Alto, CA, USA. In July 2010, he joined the Toyota InfoTechnology Center, Mountain View, CA, USA, where he is currently a Researcher with the Network Group. He represents Toyota in the Crash Avoidance Metrics Partnership Vehicle Safety Communications 3 Consortium (CAMP VSC3). His research interests include vehicular wireless communications, autonomous driving technology, and cognitive radio systems.

He was a recipient of the Alexander Graham Bell Scholarship of the Natural Science Engineering and Research Council of Canada and a best paper award at the IEEE International Symposium on Wireless Vehicular Communications (WiVeC 2013). He currently serves on the Editorial Board of the IEEE COMMUNICATIONS SURVEYS AND TUTORIALS. He was a member of the Technical Program Committee of the IEEE Vehicular Technology Conference (VTC 2013-Spring and VTC 2012-Spring).

John B. Kenney (M'83) received the B.S. degree (high honors) in electrical engineering from the University of Notre Dame, Notre Dame, IN, USA; the M.S. degree in electrical engineering from Stanford University, Stanford, CA, USA; and the Ph.D. degree in electrical engineering from the University of Notre Dame, in 1982, 1983, and 1989, respectively.

He is a National Science Foundation Graduate Fellow. In 1983, he joined the Tellabs Research Center, where he was a Senior Research Engineer until 2007. From 1989 to 2010, he was an Adjunct Assistant Professor of electrical engineering with the University of Notre Dame. From 2007 to 2010, he was a Consultant for Toyota Motor Engineering & Manufacturing North America, Inc. and Toyota InfoTechnology Center, Mountain View, CA, USA. In 2010, he joined the Toyota InfoTechnology Center, where he is a Principal Researcher with the Network Group. He represents Toyota in cooperative projects between the Vehicle Safety Communication consortium and the U.S. Department of Transportation. He also represents Toyota in dedicated short-range communications (DSRC)-related standard groups, namely, IEEE 802.11 Working Group (WG), IEEE 1609 DSRC WG, SAE DSRC Technical Committee (TC), and European Telecommunications Standards Institute TC Intelligent Transport Systems. He is the author of the "Standards and Regulations" chapter in *VANET: Vehicular Applications and Inter-Networking Technologies*, Hartenstein and Laberteaux Eds. (New York, NY, USA: Wiley, 2010). In 2011 and 2012, he was a Co-general Chair of the Eighth and Ninth ACM VANET Workshops. His current research interests include wireless congestion control, performance of vehicular networks, and spectrum-sharing technologies. His prior research interests include adaptive systems, high-speed packet-switch architectures, and packet network quality of service.

Dr. Kenney has been recognized by the IEEE Standards Association for his contributions to IEEE Standards 802.11p-2010, 1609.3-2010, 1609.4-2010, and 1609.12-2012.



Charles E. Rohrs received the B.S. degree from the University of Notre Dame, Notre Dame, IN, USA, in 1976 and the M.S. and Ph.D. degrees from the Massachusetts Institute of Technology (MIT), Cambridge, MA, USA, in 1978 and 1982, respectively.

From 1982 to 1997, he was a member of the faculty with the University of Notre Dame. From 1997 to 2000, he was a Visiting Professor with the MIT. He spent much of his career at the Tellabs Research Center, which is the research arm of Tellabs Operations, Inc.: a manufacturer of telecommunications equipment for public service network providers. From 1985 to 1995, he was the Director of Research for Tellabs as the company grew from sales in the tens of millions to over a billion dollars annually. In 1995, he became the first Tellabs Fellow. From 2001 to 2009, he was with MIT, first as a Principal Research Scientist with the Laboratory for Information and Decision Systems and then with the Digital Signal Processing Group in Research Laboratory of Electronics. He is currently working as a Consultant in mathematical modeling for control, communication, networking, and financial systems. He is the Principal of Rohrs Consulting, Inc. with whom the Toyota InfoTechnology Center is a major client. While he is best known for his early contributions to the study of robustness in adaptive control, he has also contributed work in adaptive control, adaptive signal processing, communication theory, communication networks, and switching systems. He has recently become interested in the problems of financial engineering.