On Piggybacking in Vehicular Networks

Sanjit Kaul, Roy Yates and Marco Gruteser
WINLAB, Rutgers University, NJ, U.S.A
{sanjit,ryates,gruteser}@winlab.rutgers.edu

***Abstract*—This work is motivated by network applications that require nodes to disseminate their state to others. In particular, vehicular nodes will host applications that periodically disseminate time-critical state across the network to help improve on-road safety. In this work, we want to minimize the average *age* of state information that a node observes from any other node in networks with hundreds to thousands of nodes. We explore the benefits, vis-a-vis reducing age, of a multi-hop wireless network over a fully-connected one, for a physical network of on-road vehicles, by allowing nodes to piggyback other nodes' states. We show that for a large road network and a chosen schedule, there exists an optimal fraction of connected neighbor nodes, which, for a fixed signal-to-noise ratio between most distant nodes, is invariant to the size of the network. Via simulation we confirm that significant reductions in age are obtained via piggybacking for network sizes of interest.**

I. Introduction

Our work is motivated by applications of safety messaging in vehicular networks. Most safety applications, proposed in DSRC [1], involve periodic broadcast of their state information, for example location, possibly as often as 10 times a second, to other vehicles that may be hundreds of meters away.

Prior work has looked at techniques like repetition [2]–[4] and congestion control [5], [6] to improve broadcast reliability of state information in such networks, which use carrier sense multiple access (CSMA) mechanisms specified by DSRC. We explore the state dissemination problem in large networks at a more fundamental level. We capture the stringent delay requirements of these applications using the *system age* metric, Equation (1), which is the average age (over time) of any node's state at any other node. We then ask what wireless network connectivity would minimize the *system age* for given a physical road network of vehicles. Given that safety applications require all cars *in the region of interest* to know each others state information, the chosen network connectivity must lead to a connected graph, that is it must ensure that there exists a path between any two vehicles in the network. Also, nodes may have to piggyback/relay other nodes' states.

While almost all work on safety messaging assumes a CSMA based access control, we assume that all nodes are scheduled in a round robin manner. It allows interference free scheduling of nodes, eliminates penalties suffered due to the randomized mode of transmitter selection in CSMA, and makes analysis more tractable. This helps expose the mechanisms that lead to accumulation of system age over a chosen network connectivity. This allows us to arrive at theoretical insights, to the best of our knowledge, hitherto unpublished in the area of safety messaging.

The paper organization and its contributions are as follows. Sections II summarizes related works. Section III describes the system model. Section IV derives the system age metric for a connected graph, which is followed by Section V that motivates piggybacking. In Section VI, for a heuristic based round robin schedule over a multi-lane road network, we show that there exists an optimally connected (multi-hop) graph, whose connectivity is *independent* of the size of the network. Also that the system age grows as the number of nodes in a large network, which is the same rate of growth for a fully-connected (single-hop) network. Section VII shows the significant reductions in the achieved system age as a result of the above and the larger achievable wireless link rates by a node's transmission in a multi-hop network. We end with a summary of our work in Section VIII.

II. Related Work

The DSRC proposes a CSMA based medium access control (MAC) defined in the 802.11p standard. A large body of work, for example [5], [6], has looked at congestion control mechanisms that use power control, to alleviate the deterioration of CSMA performance in large networks. Improving broadcast reliability via repetition has been explored in [2]–[4], where techniques like network coding and piggybacking are used to reduce the additional bandwidth incurred. Interference free scheduling of adjacent cells of vehicles using existing cellular infrastructure is proposed in [7] to perform alert dissemination within bounded delay. In [8] the authors evaluate algorithms that reduce the mean dissemination delay, which is the average time it takes for a given warning message to be received by all nodes. Last but not the least, we earlier used the system age metric in a CSMA setting [9].

Our work is not the first to look at piggybacking in vehicular networks. In [2], the authors use piggybacking as a form of repetition coding to improve broadcast reliability, which they define as packet reception rates. The work in [10] is probably the closest to ours in that it asks whether multi-hop broadcasting is better than single-hop (fully-connected graph) or vice-versa. The authors conclude that single-hop broadcasting is better when it comes to the age of received beacon information, however. Our claim is that a multi-hop network with piggybacking can in fact lead to a much smaller *system age* than a fully connected network of on-road vehicles. The different conclusions may be explained by the authors creating a multi-hop network by reducing the transmit power. In our work, we fix the transmit power, and increase the rate (bits/sec/Hz) to create a multi-hop network.

III. System Model

We consider a slotted transmission system with slot length τ. Let $\Delta_{ij}(t)$, denoted as Δ_{ij}^t for notational brevity, be the age of node i's information at node j at the end of slot t. Measured in slots, i.e., the age is the number of slots old the state information of node i is at node j. Thus, $\Delta_{ii}^t = 0$. Let the length of transmission of node i be l_i slots, assuming that it ends at t, for any node j that decodes the transmission, $\Delta_{ij}^t = l_i$. After k slots without another transmission by i, $\Delta_{ij}^{t+k} = k + l_i$. Let $\Delta_{uv} = E[\Delta_{uv}^t]$ denote the average age of node u's information at node v. There are a total of N nodes and for each node u, the other $N-1$ nodes in $\{v : v \neq u\}$ have old information of node u, which leads to a total of $\mathcal{N} = N(N-1)$ unique age terms. The *system age* is obtained by averaging over the $\mathcal{N}$ age terms, and is defined by

$$\Delta = \frac{1}{\mathcal{N}} \sum_{u} \sum_{v \neq u} \Delta_{uv}. \tag{1}$$

We want to minimize Δ.

Piggybacking: A packet transmitted by any node i consists of at least its own state information and a header, which includes all overheads such as for authentication, link and physical layer headers. Define $\tau_s = s\tau$ and $\tau_h = h\tau$, that is a node's state information occupies s slots while the header occupies h slots. It is often the case that $\tau_s \lll \tau_h$. For example, the IEEE 1609.2 standard uses the ECDSA algorithm for supporting authentication. The payload (containing state information) size of 53 bytes together with a certificate and a signature could lead to a packet size of up to 262 bytes [11].

When a node piggybacks the state of $k-1 \geq 1$ other nodes in its own transmission, the header still occupies h slots and the packet contains k nodes' state information. The total transmission time of the packet will be $h + ks$ slots. Thus, piggybacking may provide timely updates but comes at the expense of additional slots used for the packet transmission.

IV. Modeling the Network as a graph

Consider a network of nodes described by its connectivity graph $G = (V, E)$, where the set V is the set of all nodes in the network and the set E consists of edges between nodes. The set of *all possible directed edges* between nodes in G is denoted by $\overrightarrow{E}$. We assume that G is a connected graph, that is there exists a path between any two nodes in the graph. Further, only nodes connected by an edge can successfully decode each others transmissions. The implication is that the system employs coding such that reliable communication has a sharp threshold behavior; nodes within communication range of a transmitter decode packets perfectly while nodes beyond range discard the transmission. The number of nodes in the network is $N = |V|$.

Let P denote the set of unordered pairs of nodes in the network and $\overrightarrow{P}$ be the set of ordered pairs of nodes. A pair of nodes may or may not be connected. We denote an ordered pair of nodes by $[u, v]$ and an unordered pair by (u, v). We have $[u, v] \in \overrightarrow{P}$ and $(u, v) \in P$. If the pair is connected we use the notation $[i_k, j_k]$ and (i_k, j_k) for ordered and unordered respectively. The ordered and connected pair $[i_k, j_k] \in \overrightarrow{E}$ and also denotes the directed edge from $i_k \to j_k$, while the pair $(i_k, j_k) \in E$ and denotes the undirected edge between them. The subscript k is to index the pair and will be dropped when referring to one or any such pair. The ordered pair $[u, v]$ will be connected by a path consisting of a sequence of directed edges $([i_1 = u, j_1], [i_2 = j_1, j_2], \ldots, [i_k = j_{k-1}, j_k = v])$. When it is convenient we will denote such a path by $(u, i_2, \ldots, i_k, v)$.

We are interested in round robin schedules over nodes in V that minimize the system age. The nodes may choose to piggyback the information of zero or more other nodes to achieve the minimum. They, however, always send their own information during their transmission turn. For any node k, let $0 \leq p_k < N$ be the number of other nodes' information it piggybacks. Therefore, the length of transmission of node k is $l_k = h + (p_k + 1)s$ slots. The total number of slots in the round robin schedule is therefore given by

$$L = \sum_{k \in V} l_k. \tag{2}$$

Deciding on a schedule K for the nodes involves choosing the ordering of the nodes and the information that each node piggybacks.

Consider the pair of nodes i and j such that $(i, j) \in E$. Let node i end its packet transmission at the end of slot t. The transmission is decoded by j at the end of the slot. The age of i's information at j at the end of slot t, Δ_{ij}^t, is the length of the packet transmission by i. Thus $\Delta_{ij}^t = l_i$. Further, $\Delta_{ij}^{t+1} = l_i + 1$ and $\Delta_{ij}^{t+s} = l_i + s$, where $s = 2, \ldots, L-1$. The next transmission of i ends in slot $t + L$, which is a round robin cycle after i's previous transmission. We have $\Delta_{ij}^{t+L} = l_i$, which marks the start of a repeat of the above sequence of ages. Thus, the age, averaged over a round robin cycle, is

$$\Delta_{ij} = E[\Delta_{ij}^t] = \frac{1}{L} \sum_{s=0}^{L-1} \Delta_{ij}^{t+s} = \Delta_R + l_i, \tag{3}$$

where

$$\Delta_R = \frac{L-1}{2}. \tag{4}$$

The age Δ_{ij} has two components, l_i, which we will call the *transmission delay* and Δ_R, which we will call the *round robin delay*. While l_i depends on the length of i's transmission, and therefore on the number of other nodes' state information i piggybacks, Δ_R is a consequence of the round robin nature of the scheduling, which restricts a node to transmitting once every L slots. Clearly, both l_i and Δ_R depend on the chosen schedule.

We define ρ_{uv} as the *schedule delay* between any two nodes u and v. Let node u begin its transmission in slot t_u. Let the next scheduled transmission of node v start in slot t_v. Then, $\rho_{uv} = t_v - t_u$. Also, let d_{uv} *be the age of node* u*'s information when it is forwarded by node* v (age at the start of node v's transmission). We refer to d_{uv} as the *relay age* of node u by

v. Note that the relay age $d_{ij} = \rho_{ij}$ for the connected node pair $(i,j) \in E$.

Now consider the node pair u and v, where $(u,v) \notin E$. In this case when v transmits, it may not have received the latest update from u. Thus $d_{uv} \geq \rho_{uv}$. Let P_{uv} be the set of directed paths starting at u and ending at v in the graph G. The paths P_{uv} are those that *can* relay[1] u's information to v. Let one such path be $p_{uv} \in P_{uv}$. The path $p_{uv} = (u, i_2, \dots, i_k, v)$. The length of the path is $|p_{uv}| = k$. Consider the flow of u's information along the path p_{uv} to v. The flow starts with u's transmission, followed by the transmissions of the nodes i_2,i_3,…,i_k, all of whom by assumption will piggyback u's information. The information of node u, before it is received by node v, suffers delays that accrue over the length of transmissions of nodes u and i_n, where $1 < n \leq k$, and the scheduling delays between them. Since $[i_k, v] \in \overrightarrow{E}$, v receives what u had transmitted at the end of the transmission by i_k. It is worthy of note that the delays do not include the scheduling delay between i_k and v. Also that the scheduling delay between i_{n-1} and i_n, $1 \leq n \leq k$, includes the length of transmission l_{n-1} of node $n-1$. The sum of these scheduling delays is d_{ui_k}, which is the age of u's information that is relayed by i_k. Next we will express the age of u's information received by v as a sum of d_{ui_k} and the length of transmission of node i_k.

Node i_2 is scheduled ρ_{ui_2} slots after u, i_3 is scheduled $\rho_{i_2 i_3}$ slots after i_2 and so on. Let $\tilde{p}_{uv}$ be the subpath from $u = i_1$ to i_k, which lies within p_{uv}. Thus

$$\tilde{p}_{uv} = p_{uv} - [i_k, v] = (u, i_2, \dots, i_k). \tag{5}$$

Also, as discussed earlier, $d_{ui_k} = \sum_{[i,j] \in \tilde{p}_{uv}} \rho_{ij}$. Further, to emphasize that i_k is a function of u and v and a certain path p_{uv} between them, define $\tilde{d}^{p}_{uv} = d_{ui_k}$. Similarly, define $\tilde{l}^{p}_{uv} = l_{i_k}$. The number of slots before $v = j_k$ receives u's transmission, which is also the age of u's information when received by v, is then $\tilde{d}^{p}_{uv} + \tilde{l}^{p}_{uv}$.

The age of information of node u at node v, however, depends on the minimum delay path *for a given schedule* K. Let $p^*_{uv} \in P_{uv}$ be the minimum delay path. Once v has received u's information via p^*_{uv}, the same information received later via other paths in P_{uv} is ignored by v, as it does not update the information of node u at node v.

Let's say that node v receives u's state update at the end of slot t. Thus, $\Delta^t_{uv} = \tilde{d}^{p^*}_{uv} + \tilde{l}^{p^*}_{uv}$. Further, $\Delta^{t+l}_{uv} = \tilde{d}^{p^*}_{uv} + \tilde{l}^{p^*}_{uv} + l,\ l \leq L-1$. Finally, after an L slot round robin cycle, $\Delta^{t+L}_{uv} = \tilde{d}^{p^*}_{uv} + \tilde{l}^{p^*}_{uv}$. Thus the average age of u's information at v is

$$\Delta_{uv} = \frac{1}{L} \sum_{l=0}^{N-1} \Delta^{t+l}_{uv} = \Delta_R + \tilde{d}^{p^*}_{uv} + \tilde{l}^{p^*}_{uv}. \tag{6}$$

The age Δ_{uv} is specific to schedule K as it is a function of the path p^*_{uv}. Further, since for a connected node pair i and j, $\rho_{ij} = d_{ij}$, we can rewrite $\tilde{d}^{p^*}_{uv}$ in terms of relayed ages giving

$$\tilde{d}^{p^*}_{uv} = \sum_{[i,j] \in \tilde{p}^*_{uv}} \rho_{ij} = \sum_{[i,j] \in \tilde{p}^*_{uv}} d_{ij}. \tag{7}$$

Rewriting $\tilde{d}^{p^*}_{uv}$ in Equation (6), we obtain

$$\Delta_{uv} = \Delta_R + \sum_{[i,j] \in \tilde{p}^*_{uv}} d_{ij} + \tilde{l}^{p^*}_{uv}. \tag{8}$$

The first term is the round robin delay Δ_R, the second term is an accumulation of scheduling delays and the last term is the transmission delay on the last hop of the path, at the end of which v receives u's information. Comparing the delays with those incurred between a connected pair i and j, we see that the *round robin delay* is suffered by u and v too. Also the delay $\tilde{l}^{p^*}_{uv}$ incurred over the last hop corresponds to the *transmission delay* incurred between a pair of nodes i and j over the only transmission, the one by i, as $\tilde{l}^{p^*}_{ij} = l_i$. Lastly, the schedule delays suffered between the pair i and j are 0, as $\tilde{d}^{p^*}_{ij} = 0$.

Optimization over a Graph: The system age (1) is a function of the schedule K and graph G. We denote it as $\Delta(G, K)$. Using Equations (1), (3) and (8),

$$\Delta(G, K) = \Delta_R + \Delta_S + \Delta_L, \tag{9}$$

where

$$\Delta_S = \frac{1}{\mathcal{N}} \sum_{[u,v] \in \overrightarrow{P}} \sum_{[i,j] \in \tilde{p}^*_{uv}} d_{ij} = \frac{1}{\mathcal{N}} \sum_{[u,v] \in \overrightarrow{P} \setminus \overrightarrow{E}} \sum_{[i,j] \in \tilde{p}^*_{uv}} d_{ij}, \tag{10}$$

$$\Delta_L = \frac{1}{\mathcal{N}} \sum_{[u,v] \in \overrightarrow{P}} \tilde{l}^{p^*}_{uv} = \frac{1}{\mathcal{N}} \sum_{[u,v] \in \overrightarrow{P} \setminus \overrightarrow{E}} \tilde{l}^{p^*}_{uv} + \frac{1}{\mathcal{N}} \sum_{[i,j] \in \overrightarrow{E}} l_i, \tag{11}$$

and Δ_R is given by (4). System age, in Equation (9), is contributed to by the following interrelated quantities, the *round robin delay* Δ_R, the *average scheduling delay* Δ_S suffered by information, and the average last hop *transmission delay* Δ_L. None of the quantities are schedule independent. We want to find the schedule K that minimizes $\Delta(G, K)$ for a given G.

V. Fully-Connected Network vs. Piggybacking

Assume that all vehicles transmit at a fixed transmit power. If all vehicles in our network choose a rate C such that two vehicles that have the smallest point-to-point link SNR are connected, then the network is a fully connected (single-hop) network. Choice of a larger rate ($> C$) will lead to a network in which not all nodes are connected to each other. In such a multi-hop network, some nodes will have to piggyback other nodes' states, so that all nodes receive all other nodes' state, which will lead to larger packet transmission sizes. However, a larger rate will imply a smaller transmission time than in the fully-connected case, for a given packet size. Piggybacking is beneficial when improvements in rate outdo the increases in packet size that piggybacking may cause.

Next we ask, when for a given transmit power and physical network of nodes does transmitting at a rate which leads to a multi-hop network lead to a smaller system age? For

[1]Whether a path relays or not, in a given schedule, also depends on the piggybacking of information by the nodes in the path.

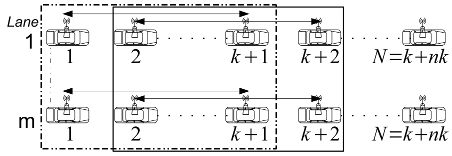

Fig. 1: A k-connected m-lane network. Each rectangle encloses a group of nodes that are connected to each other.

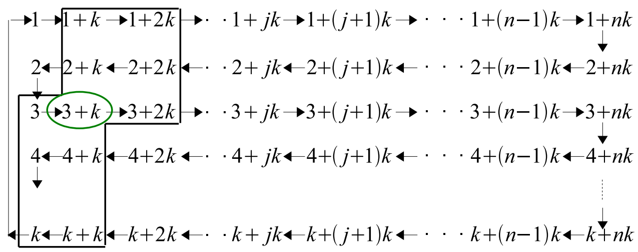

Fig. 2: Forward-and-back heuristic schedule for a k-connected single lane network. Nodes that receive when $3+k$ transmits are boxed.

exploratory purposes and given that the motivating application is vehicular, we first look at a single lane road network.

VI. A Single Lane Network

Figure 1 shows an example of a m-lane road network, each lane consisting of N cars, indexed 1 to N. A car u is connected to cars v in all m lanes, such that $\max(1, u-k) \le v \le \min(u+k, N)$, where the connectivity $1 \le k \le N$. If $k = N$, the network is fully connected and cars can transmit their state to all other cars. When $k < N$, a car's state may need to be piggybacked by other cars to reach all cars in the network. Without loss of generality, we assume that k is even and $N = k + nk$, where n is an integer, $n = \lceil (N/k - 1) \rceil \ge 0$.

The case $m = 1$ is that of a *single lane network*. Our chosen *forward-and-back* heuristic schedule for such a k-connected network of cars is illustrated in Figure 2. The construction of the schedule is motivated by the heuristic that a given car's state must take the smallest possible number of hops through the network to be received by all cars in the network. Not all cars' states will satisfy the above requirement, however. Also, a node always piggybacks another node's state (as known to it) if any nodes connected to it have older state information about the other node.

The schedule progresses along the arrows in Figure 2. Consider the node $3+k$. Its transmission is received by all nodes boxed in the figure, a total of $2k$ nodes including node $3+2k$, which follows it in the schedule. The subset of nodes $\{3+2k+1, \ldots, 3+3k\}$ is connected to $3+2k$ but not $3+k$ and so does not receive $3+k$'s transmission. Node $3+2k$'s transmission will therefore include its own state and also piggyback node $3+k$'s state. It will also piggyback, amongst others, the state of node $2+3k$, since $2+3k$'s state is newer at $3+2k$ than at one or more nodes connected to $3+2k$.

Accounting, as above, for the number of states p_u a node $u = e + jk$ piggybacks, the length of transmission l_u, defined in Section IV, can be obtained. Define $x = k/N$, where $0 < x \le 1$ is the *normalized connectivity* w.r.t. the fully connected network ($k = N => x = 1$). Let $s = \beta h$, where $\beta \ge 0$ is the *ratio of the size of state information and the header*. For a fixed x, using Equations (4), (10), and (11), as N becomes large, Δ can be approximated by

$$\Delta = hN\left(\frac{1}{2} + \frac{5\beta}{2} + \frac{(1-x)^2\beta}{x} - \frac{3x\beta}{2}\right). \tag{12}$$

The age is in slots, where a slot length was defined to be τ seconds in Section III. Let q be the SNR of the wireless link between the farthest nodes 1 and N. While we assume that the SNR is time-invariant, we will later show simulation results over a wide range of SNR. Assuming a path loss exponent of γ, the SNR between any two nodes k-hops away, for example nodes i and $i+k$, is $q_k = q((N-1)/k)^\gamma \approx qx^\gamma$. The capacity C_k of the link between two such nodes is $0.5\log_2(1+q_k) = 0.5\log_2(1+qx^\gamma)$ bits/sec/Hz. We make a simplifying assumption (relaxed in simulations) that all the nodes in the k-connected network transmit with rate C_k. The age in seconds is given by $\Delta_\tau = \Delta\tau = \Delta/C_k$. Equation (12) suggests that the x that minimizes age Δ_τ, that is *the optimal normalized connectivity* $x = x^*$ for the single lane network and the chosen schedule is *independent* of N. Also, it can be shown that increasing β for a given γ and q, increases x^*. As β increases the optimal connectivity moves towards a fully-connected one. Vehicular networks can benefit from piggybacking as state dissemination typically involves a small β, for example $\beta \approx 0.25$ in Section III. The optimal x^* also increases when q is increased for a given γ and β. Lastly, for a given β and q, increasing γ reduces x^*.

Scaling of age: We will show that for the chosen schedule and a fixed x, the age increases as $O(N)$, which is also the case for a fully-connected network. The maximum length of any node's transmission can be shown to be $2n+1$ slots, which is $O(n)$. Also, for any two nodes u and v, the age of u's information at v is the sum of relay delays along the fastest path between u and v and the length l_v of v's transmission. From Figure 2, the maximum number of hops, and thus transmissions, any node's information needs to take before it is received by all other nodes is O(n). Node $u = 3 + (n/2)k$ and $v = 3$ is an example of a worst case pair. Node u's state is forwarded along the path

$$\begin{aligned} p_{uv} = (&u, 3 + ((n/2)+1)k, \ldots, 3+nk, \\ &4 + ((n/2)-1)k, \ldots, 4, v). \end{aligned}$$

The corresponding sequence of relay delays is $\{1(n/2 \text{ times}), n, 1(n/2 \text{ times})\}$, which is a total of $O(n)$ relay delays. Therefore the sum of the relay delays is of the order of $O(n)$ packet transmissions, each of which is $O(n)$, as discussed above. Thus from (10) and (11), Δ_S is $O(n^2)$ and Δ_L is $O(n)$, as Δ_L is at worst a sum of $\mathcal{N}$ transmission lengths normalized by $\mathcal{N}$ and Δ_S is at worst a sum over $O(n)$ relay delays for each of $\mathcal{N}$ pairs of nodes u and v, normalized by $\mathcal{N}$. The last component of the system age is $\Delta_R \approx L/2$, where L is the sum of transmission lengths of all nodes and thus is $O(Nn)$.

If we *fix* n, which is the same as fixing x as $n \approx 1/x$, and increase N, for large enough N, the system age Δ can be approximated by just its round robin component Δ_R. The $O(N)$ scaling of age is the best achievable for a round-robin schedule as it is the scaling obtained for a fully-connected network. The $O(N)$ scaling together with choosing an optimal x leads to smaller system age than a fully-connected network for a range of SNR q, as we show later.

Generalization to multi-lane: Consider a multi-lane road network, which is $m > 1$ in Figure 1. It has a total of mN cars. Further assume the schedule described next. It is a concatenation of the schedule in Figure 2 carried out for lanes $1, 2, \ldots, m$ one after the other. We argue that adding m lanes to the road network, for a given k and hence $x = k/N$ (note N, *not* mN), leads to a similar Δ_S and Δ_L as for $m = 1$ and increases Δ_R by m times. The reason is that even with additional lanes, the state information of a node u takes the same number of transmissions to reach all nodes in the network (now Nm). The fastest relay paths are still the same. Also, nodes piggyback the same information as they did for the single lane case. Δ_S, even for the m-lane network, scales as $O(n^2)$, while Δ_L as $O(n)$.

If we have a single lane network with system age $\Delta_1 \approx \Delta_R$ (large N), for the same k, the multi-lane network will have a system age $\Delta_m = m\Delta_R$. Also, the x^* is the same for all $m \geq 1$.

VII. Simulation Results

We carry out MATLAB simulations for different sized road networks ($N = 50, 100$, $m = 1, 4$) assuming a pathloss propagation model with $\gamma = 3$. Reductions of up to 70% in system age for large road networks may be achieved by piggybacking. We also verify (not shown for lack of space) the notion of an optimal connectivity (see Section VI).

Unlike in analysis where nodes transmit at a common rate C, in simulations a node u transmits at a rate determined by the SNR of the link to the node farthest from it. If the farthest node is at a distance t units, then u transmits at a rate $C_u = \log_2(1 + q((N-1)/t)^{\gamma})$bits/sec/Hz. This allows some nodes to transmit at rates $> C$ for connectivity $k > N/2$ and thus yields the smallest achievable system age for a given q. In practice nodes use a common rate when broadcasting, which will lead to larger reductions in system age than shown next.

Figure 3 shows (all solid lines) the system ages obtained for $N = 100$ nodes, $m = 4$ lanes, when using the schedule from Section VI. The ratio of size of state information to header is $\beta = 0.25$. Each line corresponds to a different q and plots ages obtained for different k. The right most $k = 100$ corresponds to a fully-connected network. The obtained ages are clearly smaller for $k \approx 20$ than $k = 100$. Specifically, the reduction in achieved age (as a percentage of system age for the fully-connected case) is $(66.75, 54.91, 42.35, 30.25, 19.6)\%$ for $q = (-3, 0, 3, 6, 9)$dB respectively. Similar improvements were obtained for a network with $N = 50$ and $m = 4$ and selected q. This corroborates our earlier observation that for the chosen schedule, and a fixed x, the round robin delay Δ_R

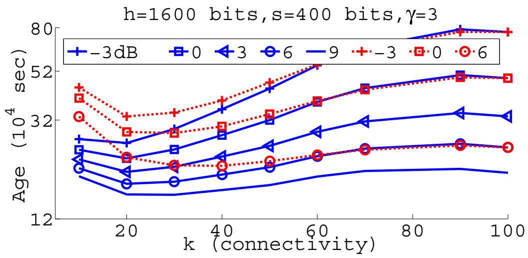

Fig. 3: 4-lane, $N = 100$ cars each lane. Each line corresponds to a fixed qdB, given by the legend. The y-axis (log-scale) shows system ages assuming 1Hz of bandwidth. An age of 12×10^4 sec corresponds to 0.012 sec in a 10MHz system.

is the most significant contributor to the system age. Also, the optimal normalized connectivity x^* is same for both networks.

The dashed lines in the figure show the system age obtained when the schedule is a randomly chosen permutation of the nodes in the network. Each age plotted is averaged over 10 random permutations. As is seen from the figure, piggybacking reduces age even when using random permutations. However, random schedules do worse than our chosen schedule.

VIII. Conclusions

An analytic formulation of the metric of system age was derived for a connected graph. The metric capture the requirements of safety messaging in vehicular networks. We define the normalized connectivity of a m-lane road network and show that its optimal value is invariant w.r.t. the number of nodes in the network, for given worst-case SNR q and path loss exponent γ. The result is shown for a chosen forward-and-back heuristic schedule. The increase in age for the schedule increases as the number of nodes for any normalized connectivity. This motivates exploring spatial reuse. Simulations show that significant reductions in age are obtained when allowing piggybacking over an optimally connected network.

References

[1] "Vehicle safety communications project task 3 final report: Identify intelligent vehicle safety applications enabled by DSRC." [Online]. Available: www.its.dot.gov/research_docs/pdf/59vehicle-safety.pdf
[2] L. Yang, J. Guo, and Y. Wu, "Piggyback cooperative repetition for reliable broadcasting of safety messages in VANETs," in *IEEE CCNC*, Las Vegas, NV, USA, 2009, pp. 1165–1169.
[3] Y. Wu, L. Yang, G. Wu, and J. Guo, "An improved coded repetition scheme for safety messaging in VANETs," in *WiCom '09.*, 2009.
[4] Q. Xu, T. Mak, J. Ko, and R. Sengupta, "Medium access control protocol design for Vehicle–Vehicle safety messages," *Vehicular Technology, IEEE Transactions on*, vol. 56, no. 2, pp. 499–518, 2007.
[5] M. Torrent-Moreno, P. Santi, and H. Hartenstein, "Distributed fair transmit power adjustment for vehicular ad hoc networks," in *IEEE SECON 2006*, Reston, VA, USA, pp. 479–488.
[6] J. Mittag, F. Schmidt-Eisenlohr, M. Killat, J. Härri, and H. Hartenstein, "Analysis and design of effective and low-overhead transmission power control for VANETs," in *ACM VANET 2008*, pp. 39–48.
[7] R. Mangharam, R. Rajkumar, and et. al., "Bounded-Latency alerts in vehicular networks," in *2007 Mobile Networking for Vehicular Environments*, 2007, pp. 55–60.
[8] A. V. Vinel, A. N. Dudin, S. D. Andreev, and F. Xia, "Performance modeling methodology of emergency dissemination algorithms for vehicular ad-hoc networks," in *CSNDSP 2010*, pp. 397–400.
[9] S. Kaul, M. Gruteser, V. Rai, and J. Kenney, "Minimizing age of information in vehicular networks," in *IEEE (SECON)*, 2011.
[10] J. Mittag, F. Thomas, J. Härri, and H. Hartenstein, "A comparison of single- and multi-hop beaconing in VANETs," in *ACM VANET 2009*.
[11] J. Petit and Z. Mammeri, "Analysis of authentication overhead in vehicular networks," in *(WMNC) 2010*, pp. 1–6.