

# Metrics and Algorithms for Scheduling of Data Dissemination in Mesh Units Assisted Vehicular Networks\*

Zhongyi LIU, Bin LIU, Wei YAN

School of EECS, Peking University, Beijing, China

E-mail: {lzy, liubin, yanwei}@net.pku.edu.cn

Received March 19, 2009; revised May 8, 2009; accepted May 12, 2009

## Abstract

Data dissemination is an important application in vehicular networks. We observe that messages in vehicular networks are usually subject to both time and space constraints, and therefore should be disseminated during a specified duration and within a specific coverage. Since vehicles are moving in and out of a region, dissemination of a message should be repeated to achieve reliability. However, the reliable dissemination for some messages might be at the cost of unreliable or even no chance of dissemination for other messages, which raises tradeoffs between reliability and fairness. In this paper, we study the scheduling of data dissemination in vehicular networks with mesh infrastructure. Firstly, we propose performance metrics for both reliability and fairness. Factors on both the time and space dimensions are incorporated in the reliability metric and the fairness in both network-wide and Mesh Roadside Unit-wise (MRU-wise) senses are considered in the fairness metric. Secondly, we propose several scheduling algorithms: one reliability-oriented algorithm, one fairness-oriented algorithm and three hybrid schemes. Finally, we perform extensive evaluation work to quantitatively analyze different scheduling algorithms. Our evaluation results show that 1) hybrid schemes outperform reliability-oriented and fairness-oriented algorithms in the sense of overall efficiency and 2) different algorithms have quite different characteristics on reliability and fairness.

**Keywords:** Data Dissemination, Vehicular Networks, Scheduling, Mesh Backhaul

## 1. Introduction

Recently, vehicular networks with the assistance of roadside units (RSUs) have received considerable attention [1–5]. RSUs are useful in many different scenarios, such as Internet access “on the go”, collecting of sensed data from the sensors on vehicles, buffering data at hotspots, etc. However, we propose vehicular networks with mesh backhaul. As wireless mesh networks have the potential advantage of easy deployment, self-configurability and large coverage [6], mesh routers are adequate to act as RSUs, which we call MRUs (Mesh Roadside Units).

In vehicular networks, data are often subject to some type of time constraints and space constraints. For example, congestion information is meaningless for vehicles 10 miles away and might become invalid after two hours.

Other types of messages can include the following: “Road maintenance work will be performed from 3:00pm to 4:00pm at the Lincoln Street”, “Traffic control will be enforced from 10:00am to 10:30am near the railway station”. Messages can also be generated by the transportation monitoring system, such as “The Lincoln Street is often in congestion from 5:00pm to 6:00pm”. This kind of messages should be disseminated during a specified *duration* and within a specific *coverage*. As vehicles are constantly moving in and out of a region, dissemination of a message should be repeated in the specified duration to achieve reliability, namely to ensure that at all times all the vehicles in a region are notified. However, the reliable dissemination of some messages might be at the cost of the unreliable or even no chance of dissemination of other ones, which raises tradeoffs between *reliability* and *fairness*. In this scenario, reliability has the meaning in both time and space dimensions. The time dimension depicts the reliability achieved by a specific MRU in its scheduling process while the space

\*This work is co-supported by National Key Basic Research Program of China (No. 2009CB320504 and No. 2007CB310902), State key lab. of virtual reality technology and systems and Peking University-Morgan Stanley Research Fund.

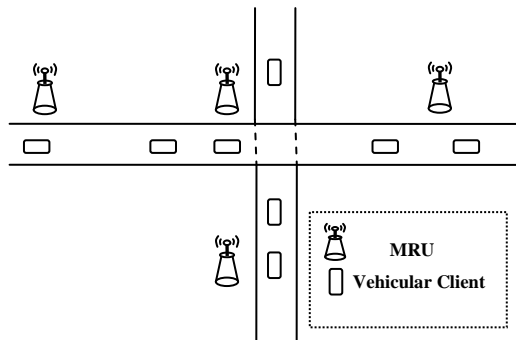
dimension describes whether messages are disseminated at all the MRUs within their requested coverage. Similarly, fairness is significant in both the network-wise sense and the MRU-wise sense. In this paper, we first propose metrics for reliability and fairness and then develop and evaluate five scheduling algorithms quantitatively. To the best of our knowledge, this is the first paper on scheduling mechanisms in this scenario to address the reliability and fairness issues.

The rest of this paper is arranged as follows: the second part presents our system model, including the network architecture and performance metrics. Five different scheduling algorithms are proposed in Section 3 and evaluation results are shown in Section 4. Related work is reviewed in Section 5, and in the last section, we conclude this paper.

## 2. System Model

### 2.1. Network Architecture

We assume a vehicular network with mesh backhaul, as shown in Figure 1. Mesh Roadside Units (MRUs) are placed at roadside locations to receive messages from cars nearby or to disseminate information to vehicles in its vicinity. Since wireless mesh network has the potential advantage of easy deployment and self-configurable, we assume that MRUs are connected via wireless links. MRUs can have larger coverage than vehicular clients, so that they can serve more vehicular clients at the same time and disseminate messages efficiently. When a vehicle needs to send messages to an MRU, it can first select a nearby MRU and then looks for relays to forward information to the designated MRU. However, the mechanism with which vehicles send messages to MRUs is out of the scope of this paper. We assume perfect message transmission from vehicular clients to MRUs in this work. We focus on the scheduling for data dissemination in this type of vehicular networks, which will be explained in detail in later sections.



**Figure 1. Network architecture.** We assume a mesh backhaul assisted architecture. Mesh roadside units are assumed to be well connected.

### 2.2. Performance Metrics

In our message dissemination model, each message is coupled with a  $\langle \text{start-time, end-time, } x, y, \text{radius} \rangle$  tuple, in which “start-time” and “end-time” indicate the instants when message dissemination should begin and terminate;  $\langle x, y \rangle$  and “radius” specify the center and radius of the dissemination area. With “x”, “y”, “radius” and some geographical information, the MRU which receives the message dissemination request can easily obtain the destination MRUs which locate in the destination area. With this message dissemination model, we figure out two important factors which determine the efficiency of message dissemination:

- **Reliability.** Reliability describes the quality of service for the selected messages. In this case, reliability covers two different dimensions. On one hand, in the time dimension, a message should be given as much dissemination time as possible in the [start-time, end-time] duration. On the other hand, in the space dimension, message dissemination should occur in an area as large as possible within the requested  $\langle x, y, \text{radius} \rangle$  coverage.
- **Fairness.** To achieve better reliability for the a selected message to disseminate, more time as well as MRUs should be allocated to it, which may in turn decrease the quality of service for the other messages. Therefore, fairness should be taken into account to enhance the efficiency of message dissemination.

We now present the metrics for both reliability and fairness. Our metric for reliability, RM (reliability metric) is defined as

$$RM = \alpha * TRM + (1 - \alpha) * SRM \quad (0 \leq \alpha \leq 1)$$

where TRM and SRM stand for Time Reliability Metric and Space Reliability Metric, respectively. The formulas for TRM and SRM are as follows.

$$TRM = \frac{\sum_{m @ MRU_i} TR(m, i)^\kappa}{N_{MRU}^i}$$

$$SRM = \frac{\sum SR(m)^\kappa}{N_{messages}}$$

where  $N_{MRU}$  is the total number of MRUs in the network and  $N_R^i$  is the number of messages to disseminate (or received) at the  $i$ th MRU.  $TR(m, i)$  indicates the *time ratio* for message  $m$  at the  $i$ th MRU and  $SR(m)$  is the *space ratio* for message  $m$ . The exponent  $\kappa \geq 1$  is used to specify the reliability level, whose effects will be explained later in this section.  $TR(m, i)$  and  $SR(m)$

are in turn defined as  $TR(m, i) = \frac{D\_time(m, i)}{Duration(m)}$  and

$SR(m) = \frac{N_{D\_MRU}^m}{N_{C\_MRU}^m}$ , respectively, namely  $TR(m, i)$  is the

ratio between dissemination time allocated to message  $m$  at the  $i$ th MRU and the requested duration of message  $m$ , and  $SR(m)$  is the ratio between number of MRUs which provide dissemination service for  $m$  and the overall number of MRUs within the requested  $\langle x, y, radius \rangle$  coverage. It is clear that our metric of reliability incorporates the reliability factors in both the time dimension and the space dimension.

The selection of  $\kappa$  is critical to achieve different reliability levels. Take the definition of TRM as an example. We assume the dissemination cycle of an MRU is  $\Delta$ , which we call it a *slot*. Assume there are two messages  $m_1, m_2$  to disseminate, both with the same duration of 2 slots and the same coverage. If  $\kappa = 1$ , then the scheduling sequences  $[m_1, m_2]$  (the first slot for  $m_1$  and the second slot for  $m_2$ ) and  $[m_1, m_1]$  will result in the same TRM. This is because in the first sequence

$$TR(m_1) + TR(m_2) = \frac{1}{2} + \frac{1}{2} = 1$$

And in the second sequence

$$TR(m_1) + TR(m_2) = 1 + 0 = 1$$

However, if we choose  $\kappa = 2$ , then the TRM of the two sequences will be different. For the first one,

$$TR(m_1) + TR(m_2) = \left(\frac{1}{2}\right)^2 + \left(\frac{1}{2}\right)^2 = \frac{1}{2}$$

And for the second one

$$TR(m_1) + TR(m_2) = (1)^2 + (0)^2 = 1$$

Therefore, the sequence with *less* messages will achieve higher reliability. The value of  $\kappa$  for TRM and SRM can be different. However, in this paper, we assume the same reliability level is used for both TRM and SRM.

Our metric for fairness, FM (fairness metric) is defined as

$$FM = \beta * \frac{N_D}{N_R} + (1 - \beta) * \frac{\sum_{MRU_i} N_D^i}{N_{MRU}} (0 \leq \beta \leq 1)$$

In which  $N_D$  and  $N_R$  stand for the number of disseminated messages and the number of dissemination requests for the network while  $N_D^i$  and  $N_R^i$  stand for the number of disseminated messages and the number of dissemination requests at the  $i$ th MRU. It should be clear that our fairness metric combines fairness factors in both the network-wise sense and the MRU-wise sense. However, currently our fairness metric only reflects whether a message is given the opportunity to be disseminated, without regarding whether different messages are given

the same level of opportunities. We leave this topic as our future work.

Also, we can combine the two metrics together. Therefore the combined metric, CM, can be defined as

$$CM = \gamma * RM + (1 - \gamma) * FM \quad (0 \leq \gamma \leq 1)$$

### 3. Scheduling Algorithms

Given the system model described above, we developed several scheduling algorithms, which exhibit different characteristics of reliability and fairness. As been stated before, we assume the dissemination cycle of an MRU is  $\Delta$  and call it a slot. A given duration between start-time and end-time can be transformed into the equivalent representation with the ordinal number of dissemination cycles. The task of scheduling algorithms is to determine the message to disseminate in the future  $W$  dissemination cycles, here we call  $W$  the *schedule window*. We further assume that any MRU knows locations of all the MRUs in the network, so that a given geographical coverage can be mapped into an equivalent representation with a list of MRUs. In the following sections, the coverage of a message means the number of MRUs in the geographical area. All the scheduling algorithms have a time complexity of  $O(W * n)$ , where  $W$  is the size of schedule window and  $n$  is the number of messages to disseminate.

#### 3.1. MQIF-Maximum Quality Increment First

Our first scheduling algorithm, Maximum Quality Increment First (MQIF) scheduling, serves first the messages which would bring the maximum quality of service (namely reliability) increment. Our approach is to first calculate the expected increment in TRM and then estimate the expected increment in SRM assuming allocation the current cycle to a message. Afterwards, the two increments are combined. The detail of MQIF is shown in Figure 2.

For each slot in the future schedule window, MQIF compares the expected quality increments of all the messages whose duration covers that slot and selects the one with the maximum quality increment. The precise calculation of expected increment of RM (denoted as QI) assuming the allocation of a slot to a message is impossible at runtime in a distributed manner. Therefore, we use the scheme shown in Figure 3 to estimate the value of QI.

The expected quality increment (QI) can be obtained from the expected increment of TRM (denoted as TQI) and that of SRM (denoted as SQI). TQI can be easily got from local information. However, SQI is dependent on the scheduling results of other MRUs in the coverage of the given message. Since we don't assume one MRU knows the scheduling status of other MRUs, we estimate

```

MQIF_Schedule()
1. selected_messages ← []
2. for i ← 1 to schedule_window do
3.   max_QI ← 0
4.   for j ← 1 to number_messages do
5.     if (msg[j].start_time - current_slot ≤ i)
        AND (msg[j].end_time - current_slot ≥ i) then
6.       QI ← calc_QI(msg[j])
7.       if QI > max_QI then
8.         max_QI = QI
9.         selected_messages[i]=msg[j]
10.      end if
11.    end if
12.  end for
13.  if selected_messages[i] ≠ null then
14.    selected_messages[i].selection_count ++
15.  end if
16. end for

```

**Figure 2.** Maximum quality increment first (MQIF) scheduling.

```

Calc_QI(msg)
1.  $TQI \leftarrow \left( \frac{msg.selection\_count + 1}{msg.duration} \right)^{\kappa} - \left( \frac{msg.selection\_count}{msg.duration} \right)^{\kappa}$ 
2.  $r \leftarrow random(0, msg.coverage)$ 
3. if msg.selection_count = 0 then //estimate increment of SRM
4.    $SQI \leftarrow \left( \frac{r + 1}{msg.coverage} \right)^{\kappa} - \left( \frac{r}{msg.coverage} \right)^{\kappa}$ 
5. else
6.    $SQI \leftarrow 0$ 
7. end if
8.  $QI \leftarrow \alpha * TQI + (1 - \alpha) * SQI$ 
9. return QI

```

**Figure 3.** Calculating the expected increment of reliability metric.

the current number of MRUs who have already scheduled the given message by generating a random number in the range 0 to  $msg.coverage$ .

```

LSF_Schedule()
1. selected_messages ← []
2. for i ← 1 to schedule_window do
3.   min_selection_count ← INFINITY
4.   for j ← 1 to number_messages do
5.     if msg[j].start_time - current_slot ≤ i
        AND msg[j].end_time - current_slot ≥ i then
6.       selection_count ← msg[j].selection_count
7.       if selection_count < min_selection_count then
8.         min_selection_count = selection_count
9.         selected_messages[i]=msg[j]
10.      end if
11.    end if
12.  end for
13.  if selected_messages[i] ≠ null then
14.    selected_messages[i].selection_count ++
15.  end if
16. end for

```

**Figure 4.** Least selected first (LSF) scheduling.

### 3.2. LSF-Least Selected First

The second scheduling approach, Least Selected First (LSF) scheduling, tries to schedule into the schedule window as many messages as possible. The general idea is that if a message had the least opportunity to be served before, it will be given the highest priority this time. LSF is given in Figure 4.

For each slot in the future schedule window, LSF compares the selection count of all the messages whose duration covers that slot and allocate the slot to the message with the minimum selection count.

### 3.3. Hybrid Schemes

Since MQIF tends to achieve high reliability and LSF tends to achieve good fairness, we can combine the two strategies to make tradeoffs between the two metrics. We figure out two approaches to do this:

- Add a certain condition to MQIF or LSF. We call the resulting algorithm Conditional-MQIF or Conditional-LSF. In Conditional-MQIF, MQIF strategy is applied only when the given condition is met; otherwise the LSF strategy is adopted. Different conditions can result in different tradeoffs between reliability and fairness; therefore this approach can be adapted to different application scenarios easily.

Combine the two strategies by simply combining the selection metrics of the two. Although it is less tunable than the former one, it may achieve better overall efficiency.

```

Conditional_MQIF_Schedule()
1. selected_messages ← []
2. for i ← 1 to schedule_window do
3.   max_QI ← 0, min_QI ← INFINITY
4.   min_selection_count ← INFINITY
5.   for j ← 1 to number_messages do
6.     if msg[j].start_time - current_slot ≤ i
       AND msg[j].end_time - current_slot ≥ i then
7.       QI ← calc_QI(msg[j])
8.       selection_count ← msg[j].selection_count
9.       if QI > max_QI then
10.        max_QI = QI
11.        MQIF_msg ← msg[j]
12.       end if
13.       if QI < min_QI then
14.        min_QI = QI
15.       end if
16.       if selection_count < min_selection_count then
17.        min_selection_count = selection_count
18.        LSF_msg ← msg[j]
19.       end if
20.     end if
21.   end for
22. //determine which strategy to use
23.   if  $\frac{\max\_QI}{\min\_QI} \geq MQIF\_THRESHOLD$  then
24.     selected_messages[i] = MQIF_msg
25.   else
26.     selected_messages[i] = LSF_msg
27.   end if
28.   if selected_messages[i] ≠ null then
29.     selected_messages[i].selection_count ++
30.   end if
31. end for

```

Figure 5. Conditional-MQIF.

### 3.3.1. Conditional-MQIF

The first approach to combine MQIF and LSF is to add a threshold to MQIF or LSF. In Conditional-MQIF, MQIF strategy is applied only when the ratio between the maximum and minimum QI exceeds a specified *MQIF\_THRESHOLD*. If the condition is not met, LSF is applied. Similarly, we can also combine MQIF and LSF using Conditional-LSF. In Conditional-LSF, LSF is used only when the difference between the maximum and minimum selection count exceeds a predefined *LSF\_THRESHOLD*. We show the detail of conditional-MQIF in Figure 5. By varying the *MQIF\_THRESHOLD* or *LSF\_THRESHOLD*, we can control the proportion of opportunities for applying MQIF or LSF strategy; therefore the algorithm can be adapted to various application demands. For example, small *MQIF\_THRESHOLD* values tend to achieve better reliability thus adequate for reliability-sensitive scenarios.

### 3.3.2. MQILSF-Maximum Quality Increment Least Selected First

Since MQIF tends to select messages with small coverage and duration, while LSF favors messages with small selection count, we can simply incorporate selection count into the quality increment (QI). This strategy is similar to MQIF, except that in MQILSF the Quality Increment (QI) is redefined as

$$QI = \frac{\alpha * TQI + (1 - \alpha) * SQI}{msg[j].selection\_count + 1}$$

Note that we use  $msg[j].selection\_count + 1$  instead of  $msg[j].selection\_count$  because the initial values of selection counts are all 0.

## 4. Performance Evaluation

We developed a discrete event simulator in Java. It takes as input an XML configuration file and a scenario file, runs the designated scheduling algorithms and writes the scheduling results to trace files.

### 4.1. Simulation Setup

We extract a 2500m\*2500m network scenario from a realistic geographical map of the TianAnMen district of Beijing, whose e-map is shown in Figure 6 [13]. We assume MRUs are evenly distributed with a distance of 300m along the roads. Therefore over 50 MRUs are deployed. The communication range of an MRU is assumed to be 350m.

The *message generation interval* and *message generation probability* indicate how often events are generated.

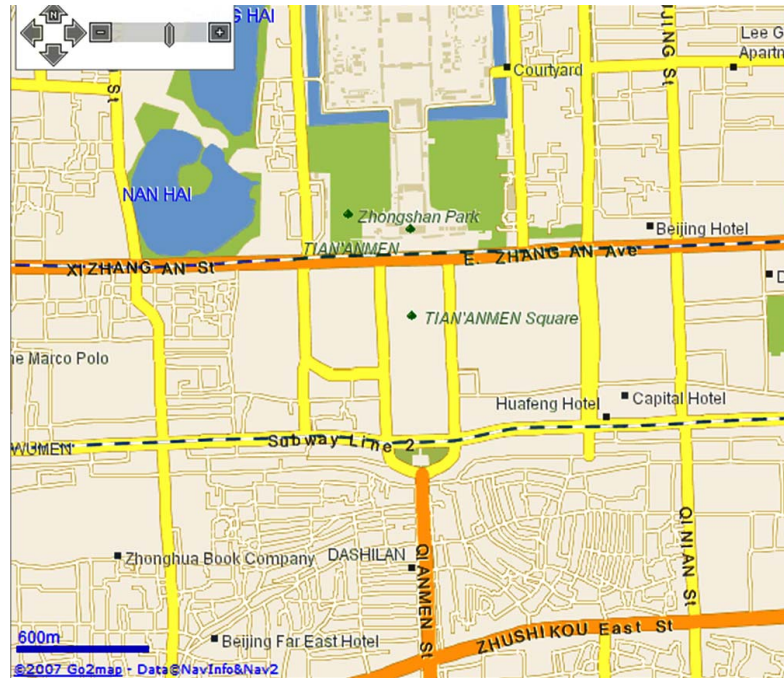


Figure 6. Simulated scenario.

Table 1. Simulation setup.

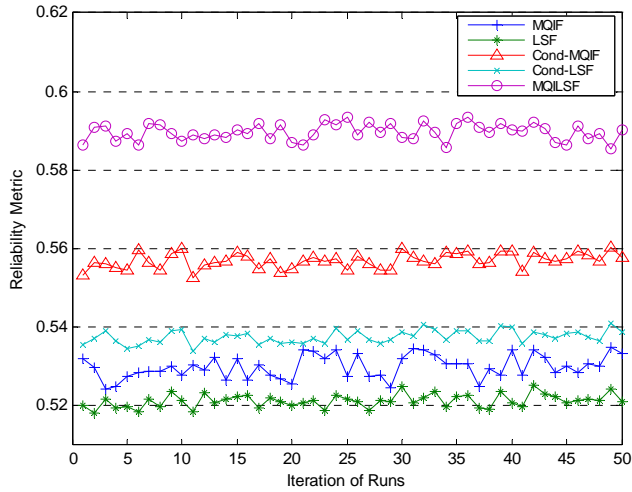
| Parameter                             | Value      |
|---------------------------------------|------------|
| Simulation time                       | 6000s      |
| Dissemination Cycle of MRUs           | 1s         |
| Duration of messages                  | 5s~300s    |
| Coverage of messages                  | 600m~1500m |
| Message Generation Interval           | 30s        |
| Message Generation Probability        | 0.15       |
| Schedule Interval                     | 5s         |
| Reliability Level $\kappa$            | 2          |
| Reliability Metric Parameter $\alpha$ | 0.5        |
| Fairness Metric Parameter $\beta$     | 0.5        |
| Combined Metric Parameter $\gamma$    | 0.5        |
| Threshold in Conditional-MQIF         | 15         |
| Threshold in Conditional-LSF          | 15         |

The simulation parameters are shown in Table 1. In our experiments, an opportunity is given to an MRU every 30 seconds and the MRU generates a message with a probability of 0.15. Durations of messages are in the range [5s, 300s] and the coverage of messages are in the range [600m, 1500m], namely about 2~5 hops. The reliability level  $\kappa$  is set to 2 in Subsection 4.2 and Subsection 4.4. Threshold values for Conditional-MQIF and Conditional-LSF are all set to 15 in Subsection 4.2 and 4.3.

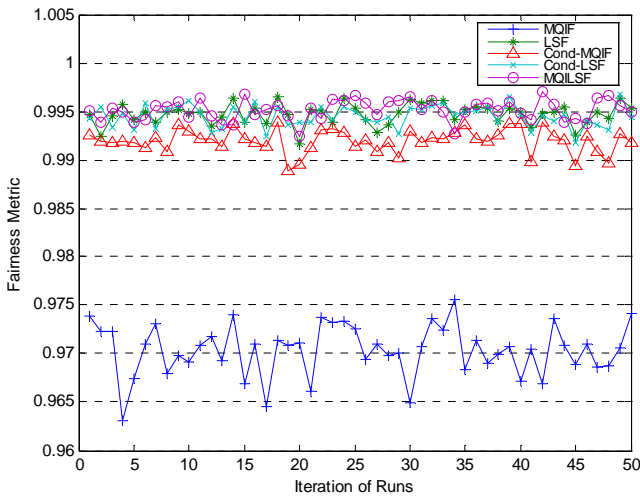
## 4.2. Comparison of Different Schemes

The Reliability Metric (RM), Fairness Metric (FM) and Combined Metric (CM) achieved by different scheduling algorithms are shown in Figure 7(a), Figure 7(b) and Figure 7(c), respectively.

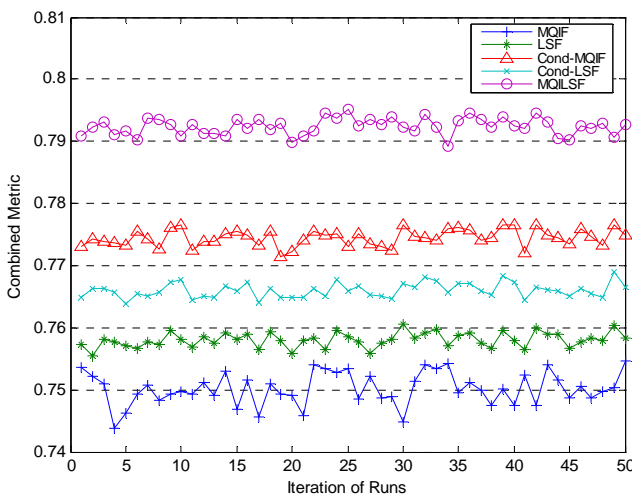
It is not hard to understand that the reliability metric of LSF and the fairness metric of MQIF are the worst among all the algorithms. However, it is interesting that



(a)

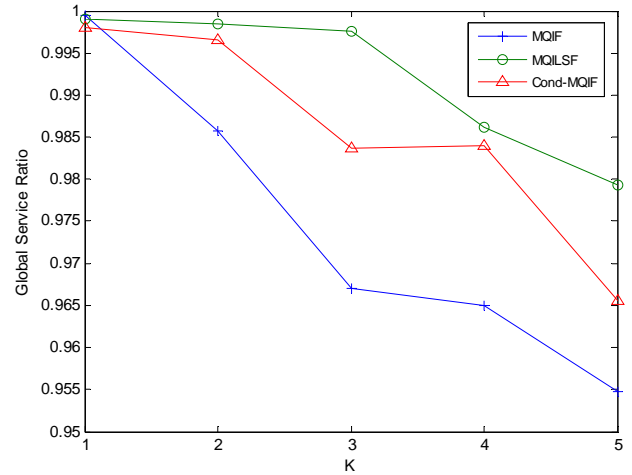


(b)

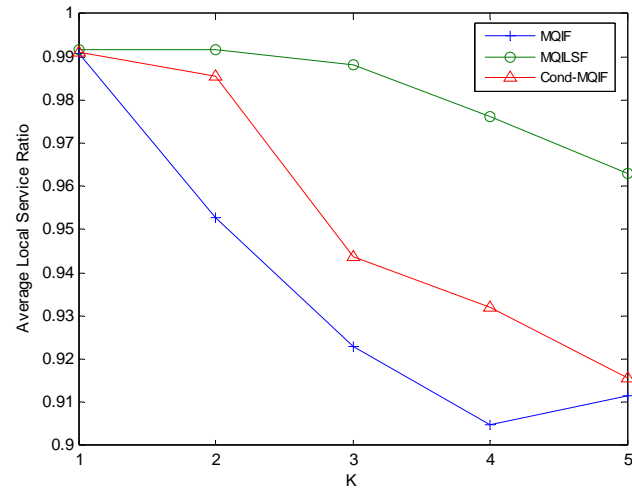


(c)

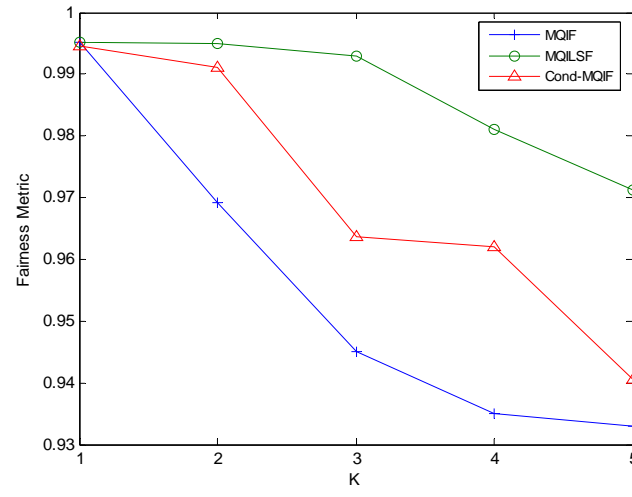
Figure 7. a) Comparison of reliability metric, b) Comparison of fairness metric, c) Comparison of combined metric.



(a)



(b)



(c)

Figure 8. a) Effects of  $\kappa$  on global service ratio, b) Effects of  $\kappa$  on average local service ratio, c) Effects of  $\kappa$  on fairness metric.

MQIF does not achieve the best reliability. We attribute this to the fact that MQIF is a greedy approach but its decisions are not made based on deterministic information. Hybrid algorithms achieve better reliability and fairness, therefore better combined metric. For example, the Reliability Metric of Conditional-LSF is about 7% higher than that of LSF and they achieve about the same level of fairness metric; the Reliability Metric and the Combined Metric of MQILSF are about 11% and 7.5% higher than those of MQIF, respectively.

### 4.3. Effects of $\kappa$

The different values of  $\kappa$  indicate different reliability levels. Although we cannot analyze the effects of  $\kappa$  on reliability by directly comparing the values the reliability metric under different models, we can do analysis by comparing the number of messages disseminated globally and locally. The Global Service Ratio (GSR), which is defined as

$$GSR = \frac{N_D}{N_R},$$

reflects the service ratio of the overall network, where  $N_D$  and  $N_R$  stand for the number of message dissemination requests received and the number of disseminated messages of the network, respectively. Note that GSR is actually the first part of the fairness metric. The Average Local Service Ratio (Average-LSR), which is defined as

$$\text{Average-LSR} = \frac{\sum_{MRU_i} \frac{N_D^i}{N_R^i}}{N_{MRU}},$$

reflects the average of the local service ratio of all MRUs in the network. Note that Average-LSR is actually the second part of the fairness metric. We also study the net effect of  $\kappa$  on the fairness metric.

As shown in Figures 8(a), (b) and (c), in MQIF, Conditional-MQIF and MQILSF, the Global Service Ratio, the Average Local Service Ratio and the Fairness Metric all decrease as  $\kappa$  increases.

Specially, the effect of  $\kappa$  on the Average Local Service Ratio is stronger than on the Global Service Ratio and the Fairness Metric, and moreover, MQIF is extremely sensitive to the value of  $\kappa$  while MQILSF is the least sensitive. This may indicate that MQILSF has the advantage of increasing reliability without degrading fairness very much.

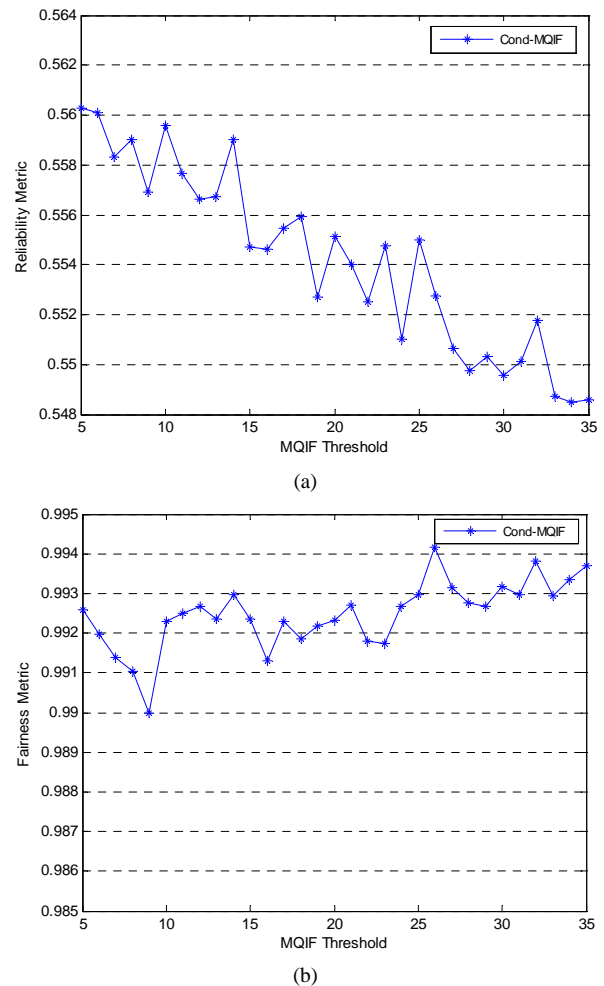
### 4.4. Effects of Threshold Values

In Conditional-MQIF and Conditional-LSF, the threshold values are critical on the reliability and fairness metric achieved.

As shown in Figure 9(a) and Figure 9(b), in Conditional-MQIF, the reliability metric decreases as the threshold value increases while the fairness metric increases as the threshold value increases. This can be attributed to the fact that the larger the threshold, the less opportunities MQIF strategy is adopted while the more opportunities are given to the LSF strategy. Similar trends are also observed in Conditional-LSF. In Conditional-LSF, as the threshold increases, more opportunities are given to the MQIF strategy, which results in better reliability metric and smaller fairness metrics. In our simulated scenario, the best threshold for Conditional-MQIF is 14 or 15, while the best for Conditional-LSF is any number in [15,18].

## 5. Related Work

Although a lot of work has been done to develop vehicular networks with infrastructure [1–5], they are usually restricted to one-hop communication between vehicular clients and roadside units. However, we propose using



**Figure 9. a) Effects of threshold values on reliability metric, b) Effects of threshold values on fairness metric.**



wireless mesh routes as the backhaul of the network, which has the potential advantage of easy deployment, self-configurable and scalability.

Scheduling for data access in vehicular networks is studied in [5]. However, our work is different from [5] because

- The work by [5] only studies scheduling for data access within one hop. In contrast, we focus on the multi-hop case, which is realistic for data dissemination in vehicular networks.
- The work by [5] does optimization for scheduling of upload/download data access. However, we consider the scheduling for data dissemination.
- The matter of fairness is not taken into account by [5]. We figure out that in data dissemination in our scenario, reliability and fairness should both be studied so that dissemination efficiency can be enhanced.

To the best of our knowledge, this is the first paper to address the reliability (in both the time dimension and the space dimension) and fairness issues in scheduling of data dissemination in the vehicular networks with mesh infrastructure.

A large amount of work has been performed on packet scheduling of MAC layer in wireless networks. The work by [7,8] tried to providing packet-level quality of service by packet scheduling. The main goals of [7,8] is to achieve fairness and maximum channel utilization. The work by [9] proposed OSMA, a packet scheduling approach in MAC layer to enhance throughput by choosing a receiver with good channel condition. However, none of them address the time and space constraints in the scenario of data dissemination in vehicular networks.

## 6. Conclusions and Future Work

As messages in vehicular networks are usually subject to space and time constraints, tradeoffs must be made between reliability and fairness for message dissemination algorithms. We propose the performance metrics for reliability and fairness in the scenario of scheduling for message dissemination in vehicular networks with mesh infrastructure. Five different scheduling algorithms are developed and evaluated quantitatively. We concluded that

- Although a greedy approach is adopted, the reliability-oriented algorithm, MQIF, does not achieve the best reliability. We attribute this to the fact that MQIF makes its greedy decisions based-on non-deterministic information.
- The fairness-oriented algorithm, LSF, achieves the best fairness metric as well as the worst reliability metric.
- The hybrid scheme, MQILSF, achieves the best reliability and combined metric and its fairness metric is nearly the same as LSF.

- The other two hybrid schemes, Conditional-MQIF and Conditional-LSF, are not as good as MQILSF. However, the idea of combining different algorithms by adding a certain condition to one algorithm can be helpful in other research fields, because it is easy to be adapted to different application scenarios.

Our evaluation on the reliability level parameter  $\kappa$  of the reliability metric show that different values of  $\kappa$  means different reliability levels. Therefore, for scenarios requiring different reliability levels, different values for  $\kappa$  should be used.

However, our current metric for fairness is not perfect; for example, it does not incorporate the relative dissemination time between different messages. On the other hand, different messages may have different priorities (indicating different level of importance or urgency), which is not considered in this paper. Furthermore, dynamic traffic densities may be useful for scheduling algorithms. For example, if the current traffic density is low, diversity of messages or fairness might be favored. Therefore, we plan to develop priority and traffic density aware scheduling schemes in the future.

## 7. References

- [1] V. Bychkovsky, B. Hull, *et al.*, "A measurement study of vehicular internet access using in situ wi-fi networks," In Proceedings of the 12th Annual International Conference on Mobile Computing and Networking (MOBICOM'06), pp. 50–61, 2006.
- [2] D. Hadaller, S. Keshav, T. brecht, *et al.*, "Vehicular opportunistic communication under the microscope," In Proceedings of the 5th International Conference on Mobile Systems, Applications, and Services (MobiSys'07), 2007.
- [3] B. Hull, V. Bychkovsky, Y. Zhang, *et al.*, "Cartel: A distributed mobile sensor computing system," In Proceedings of the 4th International Conference on Embedded Networked Sensor Systems (SenSys'06), pp. 125–138, 2006.
- [4] V. Navda, A. P. Subramanian, *et al.*, "MobiSteer: Using steerable beam directional antenna for vehicular network access," In Proceedings of the 5th International Conference on Mobile Systems, Applications, and Services (MobiSys'07), 2007.
- [5] Y. Zhang, J. Zhao, and G. H. Cao, "On scheduling vehicle-roadside data access," In Proceedings of the Fourth ACM International Workshop on Vehicular Ad Hoc Networks (VANET'07), pp. 9–18, 2007.
- [6] I. F. Akyildiz, X. Wang, *et al.*, "Wireless mesh networks: A survey," In Computer Networks, Vol. 47, No. 4, pp. 445–487, 2005.
- [7] H. Y. Luo, *et al.*, "A new model for packet scheduling in multihop wireless networks," In Proceedings of the 6th

- Annual International Conference on Mobile Computing and Networking (MOBICOM'00), pp. 76–86, 2000.
- [8] H. Y. Luo, *et al.*, “A packet scheduling approach to Qos support in multihop wireless networks,” In *Mobile Networks and Applications*, Vol. 4, pp. 193–206, 2004.
- [9] J. F. Wang, *et al.*, “Opportunistic packet scheduling and media access control for wireless LANs and multi-hop ad hoc networks,” In *IEEE Wireless Communications and Networking Conference*, (WCNC'04), pp. 1234–1239, 2004.
- [10] Y. Ding, *et al.*, “A static-node assisted adaptive routing protocol in vehicular networks,” In *Proceedings of the Fourth ACM International Workshop on Vehicular Ad Hoc Networks (VANET'07)*, pp. 59–68, 2007.
- [11] I. Leontiadis, *et al.*, “Opportunistic spatio-temporal dissemination system for vehicular networks,” In *Proceedings of the 1st International Mobisys Workshop on Mobile Opportunistic Networking (MobiOpp'07)*, pp. 39–46, 2007.
- [12] P. V. Kanodia, *et al.*, “Distributed multi-hop scheduling and medium access with delay and throughput constraints,” In *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking (MOBICOM'01)*, pp. 200–209, 2001.
- [13] E-map of Beijing (English Version), <http://en.beijing2008.cn/06/78/emap.shtml>.