On-demand flow regulated routing for ad hoc wireless networks

Ming-Shen Jian*,†, Peng-Long Wu and Chung-Nan Lee

Department of Computer Science and Engineering, National Sun Yat-Sen University, Kaohsiung, Taiwan

Summary

In this paper, we present an on-demand flow regulated routing algorithm (OFRA) for ad hoc wireless networks. The OFRA consists of two parts: an intermediate node load evaluation process and a routing path selection process. The intermediate node load evaluation process evaluates the load efficiency of the intermediate nodes according to bandwidth, data packets and computing capability. The routing path selection process selects the routing path with lower flow and fewer intermediate nodes. The OFRA can prevent intermediate nodes to be overcrowded and distribute traffic load over routing paths more evenly. The simulation result shows that the percentage of blocked routing paths is reduced and the total flow is more balanced and distributed. Copyright © 2006 John Wiley & Sons, Ltd.

KEY WORDS: traffic load, flow regulated, routing path selection, ad hoc network, wireless

1. Introduction

Currently, wireless network needs a fixed station as an intermediate node to transmit messages. Mobile devices like mobile phone, PDA and notebook, communicate to each other or browse the internet through base stations with a wired network. Under the network model, mobile devices only connect to network and make communication if they are in the range of power of base stations, which limits the scope and methods of communications. To overcome the limitations of fixed communication networks, an ad hoc wireless network is proposed, whereby mobile devices can establish a local wireless network without a base station and communicate to other devices via intermediate nodes that are also mobile devices.

In addition, the heterogeneous wireless network (HWN) [24,25] that is the hybrid wireless network is proposed. It combines the popular 3G cellular network with the ad hoc network. To make use of the advantage in such HWN, it still needs an efficient routing selection algorithm without additional hardware cost.

Since there is no fixed base station in this network, the communication links (or routing paths) are established under a dynamic situation. Each device needs to find a routing path to its destination node and sends messages only after a routing path is established. To find the destination node, the source node broadcasts the routing path build request to its neighborhood nodes. These nodes become intermediate nodes and broadcast one by one until there are no more intermediate nodes or the destination node is found.

*Correspondence to: Ming-Shen Jian, Department of Computer Science and Engineering, National Sun Yat-Sen University, Kaohsiung, Taiwan.
†E-mail: jianms@mail.cse.nsysu.edu.tw
Copyright © 2006 John Wiley & Sons, Ltd.
In an ad hoc wireless network, mobile devices are dynamic and the states of each device like scope of power, bandwidth and computing capability, movement and directions may not be the same. In other words, many factors may affect the stability of routing paths. If a node moves outside the scope of power of its neighboring nodes, the routing path via that node will be disconnected. A break in the routing path between the source node and the destination node can reduce transmission efficiency.

Hence, how to maintain the information about neighboring nodes is a significant issue. To solve this problem, Perkins and Bhagwat proposed destination sequenced distance vector routing (DSDV) [1] to make every mobile device maintain a routing table periodically. Royer and Toh [2] chose a shortest routing path between a source node and a destination node. Some routing strategies have been proposed based on the idea of table driven such as Clusterhead gateway switch routing (CCSR) [3,4], wireless routing protocol (WRP) [5]. The table-driven algorithms [1,3–5] need to get the information about all the nodes near the source node first. Each node holds a table to record the distance, location and other needed information of its neighboring nodes. When a node requests a routing path to a destination node, this source node builds a routing path according to its information table. Because the information table is renewed periodically, the routing path established according to this table may be blocked or does not exist during this period; since the state of an ad hoc wireless network changes constantly and the mobile devices are dynamic.

To reflect the dynamic behavior of an ad hoc network, some algorithms [7–12,14,15] use a source-initiated on-demand routing algorithm. This type of routing algorithm establishes routing paths without maintaining a table. In this type, all the possible routing paths from the source node to the destination node are examined. Then the destination node chooses the shortest path or finds a path with more stable signal nodes as the communication routing path between source node and destination node. However, it is important to consider more parameters in selecting routing path [6]. In References [15–17], parameters such as time complexity, communication complexity, multicast capability, message or data packet size, bandwidth, computing efficiency and power consumption are taken into account.

Dynamic source routing [7,8,12], proposed by Broch et al., is an algorithm that searches for a routing path without preparing a table in advance. For this, a source node broadcasts a routing path required message without getting neighborhood information first. Both routing metrics still use the shortest routing path [2]. Dube proposed signal stability routing [9] in order to find a more stable routing path with the shortest path between source and destination nodes. Ad hoc on-demand distance vector routing [10,12,14] and associativity based routing [11] both follow the idea of the source-initiated on-demand routing. These algorithms maintain no tables but need more time than the table-driven algorithm to discover a routing path.

If the algorithm selects a routing path with the shortest distance, different routing paths may pass through the same intermediate nodes, so these nodes easily become bottleneck nodes or high risk block nodes [16] and may be overcrowded with the traffic. When a high risk block node leaves the scope of power (or communication range), all the routing paths are interrupted and needed to be rebuilt. Therefore, we emphasize on balancing the load of intermediate nodes when selecting a routing path.

If there are many routing paths established through the same intermediate node, it may lead to a higher risk for these routing paths to be blocked. Therefore, it is important to distribute the risk over different intermediate nodes. To distribute the traffic load, some load balance algorithms for wireless networks and multiple routing paths are proposed. In Reference [23] Yanmaz and Tonguz proposed an algorithm to dynamically balance the load, but the parameters considered are only bandwidth and the size of data packet. The algorithm in Reference [18] uses only round trip time (RTT), as criterion. But RTT is not the only parameter that affects the traffic load and cannot reflect the real routing state. The algorithms in References [17–21] proposed multiple routing paths, but the traffic is not diversified into multiple routes at the same time. In other words, they will select one main routing connection to transmit data. SBPMR [22] is proposed to consider time to live (TTL) and the number of intermediate hops.

In this paper, we focus on finding a routing path with flow-regulated in order to distribute traffic load on less busy intermediate nodes and select the one or better routing path with more resources or parameters considered. The remainder of the paper is organized as follows. Section 2 describes the proposed on-demand flow regulated routing algorithm (OFRA). Section 3 presents the simulation and results.
Discussion is given in Section 4. Finally, conclusions are drawn in the last section.

2. The Proposed Algorithm

In this section, we first give an overview of the proposed algorithm. Then we describe the routing discovered algorithm based on flow-regulated and the principle of message and node setting.

2.1. Overview of the Proposed Algorithm

In an ad hoc wireless network, if there are more nodes with high block risk, it means the traffic load is more centralized and more interrupted routing paths may occur. Hence, it is important to reduce the number of nodes with high block risk. Figure 1 shows the routing path discovered to be used by most existing algorithms. Suppose that a source node, S1, establishes a routing path, which is the shortest path between itself and the destination node. When node S2 wants to build the other routing path to communicate with the node Des, nodes A and B may be chosen again for the new routing path between the node S2 and the node Des due to the shortest distance property. Once nodes A and B leave, the total routing path blocked rate will be high. Suppose that each communication (S1 to Des and S2 to Des) consumes the same resource, R, for each node and each path, then the total consumption of resource for nodes A and B in these two routes is 4R. We define the degree of the flow traffic load as the total resource consumption divided by the number of intermediate nodes. Then the degree of traffic load in Figure 1 is 4R/2. When node A or node B leaves, both routing paths are interrupted.

To reduce the risk of all routing path blocked at the same time, we need to distribute the routing paths into uncrowded nodes. By restricting the use of the flow and the number of the nodes, one can lower the risk. Figure 2 shows the different result of routing paths.

The main difference between Figures 1 and 2 is that S1 and S2 establish different routing paths through different nodes, which means that the flow is distributed. Here, if either node A or node B leaves, only the routing path established by S1 and Des will be affected, while the routing path through nodes C and D is still maintained. When the traffic load is evaluated for Figure 2, the degree of the flow traffic load is 4R/4 = R. Compared with the flow traffic load in Figure 1, the load in Figure 2 is only half of that in Figure 1. Hence one can reduce the risk of all routing path blockage and make the flow in ad hoc network more balanced.

2.2. Intermediate Node Load Evaluation Process

The intermediate node load evaluation process consists of two parts: source message requirement setting and the node load ability estimation. Source message requirement records the message type and the resource required to establish a routing path. Every node registers and adds the information needed before sending the message package. The other nodes retransmit the message package after using the function of node load ability estimation.

2.2.1. Source message requirement setting

Because of the different types of multimedia data used in 3G wireless communication networks, different requirements of communication services for different data are required. For example, multimedia data normally are limited to shorter delays and require larger bandwidth, but mail data can endure longer delays for transmission. Several parameters (e.g. priority, bandwidth etc.) should be considered for
different requirement of messages or communication traffics to get appropriate treatment, since different requirements need different QoS. Thus, parameters of diverse requirements are given by a user defined profile.

Although there are many types of messages and different parameters for different messages, here we assume that there are two types of messages. One type is the message with the smaller bandwidth and a larger quantity of data, the other is the message with a larger bandwidth and a smaller quantity of data. These factors, given on demand, are the length of the message and the bandwidth required by the message. What types of messages the nodes send are randomly chosen. The message length is the size of the packet. The longer the message, the more computing efficiency is consumed.

In this paper, the routing path format required message package is defined in Figure 3. The message package contains the source node ID, destination node ID, number of hops, node’s ID through the path, bandwidth required, message length and flow load ability (flow load ability). Number of hops means that the number of nodes the requirement message has passed. The maximum number of hops is 10, so a message is discarded after it has passed over 10 nodes. Each node that is not the destination node adds its ID into the intermediate nodes field of the message. Before a node adds its ID, this node checks the bandwidth required and calculates computing efficiency and flow load ability. Only when the node load evaluate process is satisfied, a node adds its ID. After calculating the flow load ability, the intermediate node rewrites the flow load ability value if its current flow load ability value is smaller than the flow load ability value in the message. Then, this intermediate node broadcasts this message again.

### 2.2.2. Node load ability estimation

In addition to the types of messages, we consider the ability of nodes. Practically there are many kinds and properties of mobile devices, but in this paper, we emphasize computing or operation efficiency and bandwidth supported by each node. The bandwidth supported by nodes is decided according to the computing efficiency of each mobile device. The devices with more computing efficiency have greater probability to support more bandwidth. The parameters used in the evaluation function are summarized in Table I.

The state of resource usage for a node can be calculated using the intermediate node load evaluation process, including bandwidth calculation function, computing efficiency function and available flow load ability function as follows:

The bandwidth calculation function is defined as;

$$B_{EI}^{(l)} = B_{T}^{(l)} - \left( B_{U}^{(l)} + B_{M}^{(l)} \right)$$ (1)

The computing efficiency function, $E_I$, which considers the computing capability, $C_I$, and the size of the data packet from source node, $S_M^{(x)}$, is defined as;

$$E_I = \frac{1}{n} \sum_{x=1}^{n} T_I^{(x)} = \frac{1}{n} \sum_{x=1}^{n} \frac{S_M^{(x)}}{C_I}$$ (2)

where $T_I$, $S_M^{(x)}$ and $C_I$ are defined in Table I. Then, the available flow load ability of node $I$ can be estimated as $l(I)$ and defined as;

$$l(I) = \frac{B_{EI}^{(l)}}{B_{EI}^{(l)} E_I}$$ (3-a)

When a node starts the node load ability estimation process, it checks the parameter fields, such as the number of hops, bandwidth required, message length and flow load ability. The node then checks the value of number of hops field. If the value equals or is

<table>
<thead>
<tr>
<th>Source node ID</th>
<th>Destination node ID</th>
<th>Number of hops</th>
<th>Intermediate nodes</th>
<th>Bandwidth required</th>
<th>Message length</th>
<th>Flow load ability</th>
</tr>
</thead>
</table>

**Fig. 3. Message package type.**
greater than 10, the node drops the message. When a node receives the message for the first time, it starts the node load ability estimation process to check the bandwidth required, message length and flow load ability. First, if the bandwidth supported by this node is less than the requirement of this message, the node drops the message. Second, the intermediate node calculates the computing efficiency according to Equation (2). If the computing efficiency of this intermediate node cannot satisfy the requirement of this message, the intermediate node drops the requirement message. In other words, the number or size of data packets transmitted and the computing capability are both the parameters considered.

The less power consumption and less path centralization can also make the load of an intermediate node reduced. As mentioned above, more parameters considered are important. In order to take the power consumption and path centralization into consideration, we extend Equation (3-a) as follows:

$$l(I) = \frac{B_f(I)}{B_f(I)E_l} \times \frac{1}{P(I)} \times \frac{1}{F(I)}$$  \hspace{1cm} (3-b)

where $P(I)$ indicates the current power consumption of intermediate node $I$ and $F(I)$ means the total number of routing paths through this intermediate node $I$.

In other words, only when the available flow load ability of the intermediate node can satisfy all the requests, the node passes on the message.

Suppose that $f$ indicates the resource considered. Then, the resource that an intermediate node supports can be denoted as $f_{sup}$ and the resource required by source node can be denoted as $f_{req}$. Then the check function in a node can be defined as:

$$F_k = \begin{cases} 
1 & \text{if } f_{sup} \text{ satisfy } f_{req} \\
0 & \text{otherwise} 
\end{cases} \quad k = 1, 2, 3, \ldots K_{parameter}$$  \hspace{1cm} (4)

where $K_{parameter}$ is the total number of parameters considered.

And the result of all resource checks, $\varepsilon$, can be expressed as:

$$\varepsilon = \prod_{k=1}^{K_{parameter}} F_k$$  \hspace{1cm} (5)

When $\varepsilon$ equals one, this node can satisfy the QoS requirements of the message from the source node.

After checking the parameters registered in the message, the node checks the flow load ability, then calculates the flow load ability and compares it with the value of the flow load ability field in the message. If the current value of flow load ability is smaller than the value registered in the message, the node replaces the value by the current value. If the node receives the message more than once, it verifies that the value of flow load ability of the current routing path is larger than the message that has passed the node before without checking other parameters. If the current value is larger, the node broadcasts the message to its neighborhood. Otherwise, the node drops the message.

After the node meets the requirement, it joins as an intermediate node. It adds its information into this requirement message and reserves the resource for this established path. Then it passes the message to its neighboring nodes. Then, the node waits for acknowledgement from the destination node within a valid period. If there is no acknowledgement from the destination node after a valid period, it releases the reserved resource.

2.3. The Routing Path Selection Process

If the information about the destination node of this requirement message is the same as the node itself, it checks the information about nodes it has passed. Because the destination node may receive more than one routing path established requirement from the same source node through different routing paths, it will choose the one with the lowest load as the routing path and the one with the second lowest load path as the backup path. Then, it returns an acknowledgement message including the chosen path to the source node and all intermediate nodes. After the source node receives the message, the connection between two nodes is established.

The routing path selection process is to find a routing path to achieve the best load balance. Using the same parameters as defined in Table I, the routing path selection process is described as follows. Let $R$ be a set of available routing paths given the source node, the destination node and all other nodes:

$$R = \{R_1, R_2, \ldots, R_j\}$$  \hspace{1cm} (6)

For an arbitrary path, the $j$th routing path $R_j$ consists of $t_{i_1}^1, t_{i_2}^1, \ldots, t_{i_j}^1$ intermediate nodes is defined as follows:

$$R_j = \{t_{i_1}^1, t_{i_2}^1, \ldots, t_{i_j}^1 \mid 1 \leq k_j \leq 10\}$$  \hspace{1cm} (7)
The set of loads in the $j$th routing path can be calculated using the node load evaluation function proposed previously for each intermediate node.

$$L(R_j) = \min\{l(I_{j1}), l(I_{j2}), \ldots, l(I_{jk}) \mid 1 \leq k_j \leq 10\} \quad (8)$$

Then the load of available routing paths from the source node to the destination node can be written as:

$$L(R) = \{L(R_1), L(R_2), \ldots, L(R_j)\} \quad (9)$$

The routing selection function can be defined as:

$$S(R) = \max\{L(R_j)/k_j \mid \text{for all } j \text{ that } 1 \leq j \leq \text{No of all available routing paths}\} \quad (10)$$

Then the source node selects the routing path corresponding to $S(R)$ to establish communication with destination node. When the connection between the source node and the destination node is established, each intermediate node refreshes its parameters. It refreshes them again when the connection is discarded.

Figure 4 shows an example of the routing path discovered according to the OFRA. Suppose that S1 sends the routing path established requirement message to node Des at $t_0$, and S2 sends the same type of routing path established requirement message to node Des at $t_1$. Define the bandwidth required as BR and the message length as L. Nodes A, B, C and S2 have the same conditions and nodes E, F and D have the other same conditions. Define nodes A, B and C as the nodes with 112 K bandwidth and 100 MHz computing capability; and nodes E, F and D as the nodes with 10 M bandwidth and 500 MHz computing capability.

When S1 sends the requirement message at $t_0$, the routing paths to the Des can be found as S1-A-B-Des, S1-A-S2-C-D-Des and S1-A-S2-E-F-D-Des. The load, $L(R)$, of these three paths between S1 and Des is the same and is:

$$\frac{112k - BR}{112k} \div \frac{L}{100 \text{ MHz}} \quad (11)$$

But considering the flow-regulated algorithm, we propose the available flow load ability of the first routing path is:

$$\frac{112k - BR}{112k} \div \frac{L}{100 \text{ MHz}} \div 2 \quad (12)$$

It is larger than the second and the third routing paths. After establishing a routing path between S1 and Des, S2 wants to build a routing path to Des at $t_1$. Also, there are three routing paths, S2-A-B-Des, S2-C-D-Des and S2-E-F-D-Des that can be found between S2 and Des. But the flow load ability now are different. The flow load ability of the routing path S2-A-B-Des is:

$$\frac{112k - BR - BR}{112k} \div \left(\frac{1}{2} \times \left(\frac{L}{100 \text{ MHz}} + \frac{L}{100 \text{ MHz}}\right)\right)$$

$$= \frac{112k - 2BR}{112k} \div \frac{L}{100 \text{ MHz}} \quad (13)$$

In addition, we calculate the second and the third routing paths. Then we can get three different values of these three routing paths. The first is:

$$\frac{112k - 2BR}{112k} \div \frac{L}{100 \text{ MHz}} \div 2 \quad (14)$$

The second routing path S2-C-D-Des is:

$$\frac{112k - BR}{112k} \div \frac{L}{100 \text{ MHz}} \div 2 \quad (15)$$

and the third routing path S2-E-F-D-Des is:

$$\frac{10M - BR}{10M} \div \frac{L}{500 \text{ MHz}} \div 3 \quad (16)$$

The available flow load ability of the third routing path is about $10/3$ times than that of the first and second routing path, when BR is very small.
3. Simulation

To simulate a real world case, the computing capability of the mobile devices are ranged from 500 MHz to 5 MHz. Two types of messages listed in Table II are used to simulate the voice and data messages respectively [1,15]. Different types of messages have different requirements for bandwidth. During the routing path established period (or called required TTL), the source node can send a Type 1 message with length from 5 to 15 KB or send a Type 2 message with length from 30 to 90 KB.

In order to compare the performance of the proposed algorithm with other existing algorithms, we conduct simulations with the number of nodes which are 50, 100 and 500. Nodes are placed randomly within a fixed-size $S \times S$ area, where the length is in terms of unit-length, ranging from 50, 100, 500 to 1000. To simulate different node density, we define six simulation cases, as listed in Table III, to evaluate the algorithms. Nodes are moved in random directions [13,15,16] inside $S \times S$ area with 2 and 4 unit-lengths per second. The total simulation time is about 600 s, and the maximum number of hops is 10. Each mobile device has the same scope of power (or communication range) and is set to 10. According to the different types of devices and two types of messages, they have different bandwidth and different computing efficiency. All simulation parameters and values are listed in Table III.

The computing capability of all the nodes are divided into five levels, 5 MHz, 25 MHz, 100 MHz, 200 MHz, and 500 MHz. Poison distribution is used to select, which level the node is. Most of devices in the ad hoc network are handheld with computing capability from 25 MHz to 500 MHz. According to computing capability levels, we also define bandwidth from 10 Mbps to 56 Kbps because devices with less computing capability usually support less bandwidth.

When a node sends a message, it also sends the information about the requirements to the intermediate nodes. The requirement state of the message is determined as follows. First, according to the behavior of a mobile device user, the message may be voice or data. For this, we use a random function to decide the message type that indicates the total length or total size of this message. Second, the source node can randomly request the requirement of computing capability. Third, the bandwidth should be considered since nodes without enough bandwidth cannot support a routing path. Based on Equation (3-b), the value of power consumption and the path centralization are as low as possible. Since they are not determined by the source node, source node gives no requirement about these two parameters.

Here, we propose two parameters to indicate the performance of the algorithm. They are the traffic load rate, $\lambda$, and the block rate value, $\beta$.

The traffic load rate, $\lambda$, is used to evaluate the distribution state and is defined as follows:

$$\lambda = \frac{\sum \text{consumption of resources}}{\text{devices}_{\text{used}}} \tag{17}$$

where devices$_{\text{used}}$ is the total number of the intermediate nodes in all the routing paths established during the period of the simulation, and ‘$\sum \text{consumption of resources}$’ indicates the total consumption of resources in all the intermediate nodes used by all the routing paths established during the simulation period. If the value of $\lambda$, which indicates the traffic load rate (or flow load rate) is small, then it indicates that the network is more balanced and distributed in the flow regulation. Moreover, it means that these nodes can provide more resources and serve more communication with different requirements. In other words, if the value of $\lambda$ is small, it indicates that the routing path selection algorithm can distribute the routing paths through different intermediate node more efficiently.
Finally, the required time to live (RTTL) of the routing path is considered. If a routing path cannot be maintained during RTTL, we say this routing path is blocked. In this paper, we also define the block rate $\beta$ as:

$$\beta = \frac{\text{number of block routing paths}}{\text{total number of routing paths built}}$$

(18)

In order to evaluate the performance of the proposed flow-regulated routing algorithm, we compare it with three other existing routing algorithms: the shortest path algorithm, the location stability algorithm and the signal strength stability path algorithm. The conditions during the simulation are all the same. The shortest path algorithm finds the routing path by searching for the shortest path between the source node and the destination node with the lowest number of intermediate nodes. The location stability path algorithm chooses the intermediate nodes so that the moving speed of the nodes is small at the time. The signal strength stability path algorithm detects the strength of the signal sent by other nodes and chooses the nodes with a suitable distance based on a profile from the source node [9].

Figure 5 shows the block percentage of routing path. If a routing path is interrupted during the RTTL period, the source node broadcasts the routing path required message and packages message again to find a new routing path. After broadcasting, this routing path is considered blocked if the path cannot be established again. Suppose that the probability for a node not to move away, is $P_m$. The block probability of both routing paths in Figure 1 is

$$1 - (P_m)^2$$

(19)

The block probability of both routing paths in Figure 2 can be calculated as:

$$[1 - (P_m)^2][1 - (P_m)^2]$$

(20)

The block probability of the flow-regulated algorithm proposed here is smaller than the algorithms using the shortest path. Every time a node moves away, the fewer the routing paths through that node are, the less chance the routing paths are blocked. In Figure 5, we can find that the routing path block percentage of the flow-regulated routing algorithm is about 21–74% lower than the routing path block percentage of algorithms based on the shortest path selection. For the worst case (cases), it is at least 1% lower than other competing algorithms.

The efficiency or performance of algorithms is shown in Figure 6, where the efficiency ratio is defined as:

$$\text{Efficiency} = \frac{\lambda_{\text{the competing algorithm}}}{\lambda_{\text{the proposed algorithm}}}$$

(21)

Comparing the results for different cases of resources in an ad hoc network, the proposed algorithm can save a large amount of resources. In most cases, the flow-regulated routing algorithm is 2–7 times better than other algorithms, and in some cases, it is 90 times better than other algorithms, which do not
consider the consumption of resources and the flow regulation. In these six cases, we can calculate the density of the node communication coverage in the simulation area and are \( \frac{5 \times 10^{-1}}{C2} \), \( \frac{4}{C0} \), \( \frac{1}{C2} \), \( \frac{4 \times 10^{-2}}{C0} \), \( \frac{2 \times 10^{-1}}{C0} \), and \( \frac{5 \times 10^{-2}}{C0} \). If the density is large, it means that the source node may connect to the destination node directly and there is no need to employ any intermediate node. Therefore, if the routing paths need to employ more intermediate nodes, the performance of our algorithm can be better.

Figure 7 shows the average number of intermediate nodes used in different cases and different algorithms. The number of nodes used is about the same among different algorithms in most cases.

Figure 8 shows that the more the factors are considered, the lower the block rate is. The OFRA\(_a\) indicates that the routing selection algorithm is based on Equation (3-a) and OFRA\(_b\) is based on Equation (3-b). The OFRA\(_b\) considers two more parameters such as power consumption and path centralization, than that of OFRA\(_a\). The value of routing path block percentage \( \beta \) of OFRA\(_b\) is 1–21% lower than that of OFRA\(_a\). That means, to consider more factors in selecting the best routing path is needed and important.

4. Discussion

Based on the simulation results, the algorithm proposed here can reduce the block rate and distribute the traffic load. It is difficult to ensure the state of
each intermediate node, and to forecast the state or adopt the suitable state of the node. Hence, to adopt a scheme that can distribute the traffic load is important.

If an intermediate node serves too many routing paths, it has a high risk of blocking routing paths. In other words, if the node is out of communication with other nodes, all the routing paths served by the node will be blocked. Because nodes may change direction or speed in the next period and their future state cannot be forecast, the algorithm based on on-demand conditions becomes less efficient. When a routing path is established using the shortest path, it may repeatedly use the same intermediate nodes. As a result, the routing paths may repeatedly be centralized through certain intermediate nodes, so that the traffic load is also centralized. The consumption of resources used in these intermediate nodes becomes large, which means that resources are not used equitably and fairly. Hence a blockage can easily occur, because the node may move away. On the contrary, the routing path block percentage and traffic load are much lower by using the flow-regulated routing algorithm.

The flow-regulated routing algorithm can perform even well when multiple routing paths to the same destination node are available. If there is only one routing path discovered to the destination node, the flow-regulated routing algorithm will degenerate to the algorithms that select the shortest path. In Figure 9, we can find that the number of routing paths built (or established) becomes small with the same number of nodes when simulation area increases. When length $S$ is 500, there are fewer than 100 routing paths established. Although the number of routing paths established is small, there may be more than one routing path discovered during each routing path search process.

When the simulation area is gradually larger, the traffic load rate and the number of the routing paths established will be affected. Figure 10 shows the relation between the traffic load rate and the size of area. We can find that the traffic load rate of each

Fig. 9. Relation between routing path built and the simulation area in the unit of length with node $= 100$.

Fig. 10. Relation between traffic load rate $\lambda$ and the simulation area in the unit of length with node $= 100$.
algorithm gradually decreases, when the area increases. This is because the number of the routing path established between the source node and the destination node is large with a small area but is small with a large area. The value of OFRA is lower than that of other algorithms because the other algorithms may repeatedly establish routing paths through the shortest path and intermediate nodes, so the nodes have high risk and the traffic load rate becomes large.

We summarize our major contributions as follows:

1. It is necessary to avoid overcrowded intermediate nodes and distribute load through different intermediate nodes.
2. It is important to consider more parameters when selecting routing paths in an ad hoc network.

When few parameters are considered, it will easily centralize the routing paths established through a few nodes. These few nodes become the bottleneck nodes. In this paper, we define the flow load ability which considers the size of data packet, computing capability and bandwidth to indicate the capability of an intermediate node. Hence, OFRA considers more resources than the existing algorithms to distribute the routing paths through different nodes. The routing paths use different node with more resource supported. In other words, the more parameters considered, the routing paths can be more efficiently distributed. Since the proposed algorithm selects the best routing path for a set of discovered routing paths, it can easily be extended to multiple routing paths. Another important factor to reduce the blockage is mobility that will be considered in the future study.

5. Conclusion

We have proposed an OFRA to distribute load on intermediate nodes more evenly in an ad hoc wireless network. This algorithm consists of two parts: an intermediate node load evaluation process and a routing path selection process. For most of simulation cases, the OFRA’s probability of blockage is about 1–74% lower than the competing algorithms. The traffic load rate of OFRA is 20–40% of the traffic load rate of the competing algorithms. However, for some cases, traffic load rate of OFRA is only 1–3% that of competing algorithms. Furthermore, the efficiency of OFRA can be even better when there is more than one routing path discovered between the source node and the destination node.

References


Authors’ Biographies

**Ming-Shen Jian** was born in Kaohsiung, Taiwan, in 1978. He received the B.S. degree in Electrical and Control Engineering from National Chiao Tung University (NCTU) in Hsinchu, Taiwan, in 2000. In 2001, he pursued his M.S. degree. From 2002, he skipped a grade and been a Ph.D. student at the Department of Computer Science and Engineering of the National Sun Yat-Sen University (NSYSU) in Kaohsiung, Taiwan, investigating resource management in 3G mobile communication systems. From 2001 to 2004, he also was a research and teaching assistant at NSYSU. His current research interests are in the area related to QoS and wireless heterogeneous network.

**Peng-Long Wu** was born in Kaohsiung, Taiwan. He received his B.S. degree in computer information science from Soochow University (SCU), Taipei, Taiwan, in 2001. In 2003, he received the M.S. degree in computer science and engineering from Sun Yat-Sen University (NSYSU), Kaohsiung, Taiwan. From 2003, he pursued his Ph.D. degree at NSYSU. From 2003 to 2004, he also was a research and teaching assistant at NSYSU. His research interests include wireless network and information appliances.

**Chungnan Lee** received the B.S. and M.S. degree in electrical engineering from National Cheng Kung University, Tainan, Taiwan, in 1980 and 1982, respectively, the Ph.D. degree in electrical engineering from the University of Washington, Seattle, WA, in 1992. Since 1992, he has been with National Sun Yat-Sen University, Kaohsiung, Taiwan, where he was an Associate Professor in the Department of Computer Science and Engineering from 1992 to 1999; He was Chairman of the department of Computer Science and Engineering from August, 1999 to July, 2001 and is currently a Professor. His current research interests include wireless networking, information appliances, computer graphics, multimedia systems, bioinformatics, and parallel computing.