# Fish Eye OLSR Scaling Properties

Cedric Adjih, Emmanuel Baccelli, Thomas Heide Clausen, Philippe Jacquet, and Georgios Rodolakis

*Abstract:* **Scalability is one of the toughest challenges in ad hoc networking. Recent work outlines theoretical bounds on how well routing protocols could scale in this environment. However, none of the popular routing solutions really scales to large networks, by coming close enough to these bounds. In this paper, we study the case of link state routing and OLSR, one of the strongest candidates for standardization. We analyze how these bounds are not reached in this case, and we study how much the scalability is enhanced with the use of Fish eye techniques in addition to the link state routing framework. We show that with this enhancement, the theoretical scalability bounds are reached.**

*Index Terms:* **Ad hoc, mobile, network, routing, scalability.**

## I. INTRODUCTION

In their famous paper, Gupta and Kumar [1] have shown via information theory that when the size $N$ of the network increases (with randomly placed nodes), the optimal neighbourhood size is $O(\log N)$, which leads to the maximum network capacity per node being $O(1/\sqrt{N \log N})$. This in turn leads to a neighbourhood radius that shrinks in $1/\sqrt{N \log N}$, which yields a network diameter in hop number being $O(\sqrt{N \log N})$.

However, if we drop the requirement for the network to be connected, and just require the existence of a giant component, we can actually drop the $\log N$ factor in these formulas. Indeed, the condition to have a giant component is that the average neighbourhood size is greater than 1, and we can therefore consider in our study that it is no longer $O(\log N)$, but rather $O(1)$. In this case, the maximum capacity per node is then $O(1/\sqrt{N})$. Note that Gupta and Kumar have also shown that when the nodes are optimally placed, the giant component is actually the whole network.

This property means that when the network size increases, the neighbourhood size must be kept *at least* constant, and well above 1, in order to have an operational network. But as noticed by Gupta and Kumar, this neighbourhood size essentially depends on the amount of traffic generated by each node: The larger the generated traffic is, the smaller the neighbour size will be. When too much transmissions occur, packet collisions prevent longer range links from providing satisfactory neighbour link. This fact imposes the $O(1/\sqrt{N})$ bound they found for the optimal bandwidth assigned per node.

In fact, part of the traffic generated by each node is the control

Manuscript received July 30, 2004.

C. Adjih, P. Jacquet, and G. Rodolakis are with the HIPERCOM Project, INRIA Rocquencourt BP 105, 78153 Le Chesnay Cedex, France, email: {Cedric.Adjih, Philippe.Jacquet, Georges.Rodolakis}@inria.fr.

E. Baccelli is with the Hitachi Ltd. He is also part of the INRIA HIPERCOM Project, email: Emmanuel.Baccelli@inria.fr.

T. H. Clausen is with the Project PCRI Pôle Commun de Recherche en Informatique du plateau de Saclay, CNRS, École Polytechnique, INRIA, Université Paris-Sud, France, email: Thomas.Clausen@polytechnique.fr.

traffic due to the routing protocol in use. Therefore, the larger this control traffic is, the smaller the actual neighbour size will be. There is a need to have an efficient control over the routing protocol overhead in order to avoid that the network capacity degrades or even collapses because of too much control traffic. The aim of this paper is to define the condition a routing protocol must satisfy in order to fit the Gupta and Kumar optimal scaling property.

The above theoretical results need to be compared with the reality when using existing routing protocols. For example, link state routing protocols do not exactly satisfy the scaling properties outlined by Gupta and Kumar. In reality, the average neighbourhood size tends to slowly decrease as the network size increases. This is due to the fact that the control traffic consists here in topology information generated and relayed by each node in the network, while the amount of this information tends to increase linearly with the size of the network. This obviously puts an upper bound on the maximal size of the network, above which the amount of control traffic generated by topology updates purely and simply prevents the network from being formed and connected. In fact, this limitation is common to every flat routing protocol, where all nodes have the same role and are put on the same level of information importance, i.e., every node is supposed to know the same amount of information about its direct neighbours as about any node in the network, however remote it may be.

One way to work around this problem is to establish a hierarchical protocol that takes advantage of the scaling properties of node clustering and super-clustering. This technique greatly reduces the transit of topology information between clusters. However, complexity remains in adequately distinguishing and forming different clusters. This is especially difficult in an inherently decentralized and mobile environment like ad hoc networks.

An alternate solution was proposed by Gerla *et al.* in [2], who introduced the concept of Fish eye routing. Contrary to the hierarchical approach, the Fish eye technique is totally decentralized. Essentially, it consists in reporting remote nodes information less frequently than nearby nodes information: The further away a node is, the less frequently information about it will be reported. The idea is that, in order to route data to a remote destination, what a node really needs is just a "general direction" in which the data is to be sent, while totally accurate routing information is superfluous at that point. And as the data approaches the destination, the available routing information becomes increasingly more accurate, finally enabling it to be delivered correctly.

Aside from being much less complex than the hierarchical approach, another advantage of the Fish eye technique is that network-wide, the weight of the control traffic generated by a node decreases as a function of the distance from this node. Therefore, if the employed Fish eye technique uses an appro-

priate function of the distance from the node to decrease the frequency of topology updates, we can get the control traffic (generated or relayed by each node) to converge to a finite upper bound, even when the network size grows infinitely. The control traffic density remains $O(N)$ per area unit and the neighbourhood radius decreases in $1/\sqrt{N}$, enabling the routing to scale for arbitrary large networks if the parameters are appropriately tuned to keep the average neighbour size greater than 1.

This paper is organized as follows. The next section introduces some basic elements in order to model ad hoc networks: Slotted time, propagation model, fading model, uniform density of transmitters dispatched on an infinite plane, etc. Then the following section will extend this model to networks of finite size: $N$ nodes uniformly distributed on a finite portion of plane. We will apply this model to study link state routing in the context of ad hoc networking, and focus in particular on OSPF (open shortest path first [3]) and OLSR (optimized link state routing [4]), two link state routing protocols.

The approach taken by link state protocols is that of a distributed database describing the network, which is replicated and maintained throughout the routing domain. More precisely each node starts by describing it's local environment, i.e., the state of the links with its immediate neighbours. This neighbour sensing is done via the periodic sending/receiving of simple "Hello" packets. At the same time, each node floods topology descriptions (LSA packets in OSPF or TC packets in OLSR) to all the other routing nodes in the network—not only immediate neighbours this time. These longer range packets contribute pieces to a database which therefore (i) contains the descriptions of all the nodes in the network, and (ii) is present and the same in each node. This link state database is kept up-to-date in all nodes by the same flooding mechanism, and such periodically as well as occasionally in case of change in some node's neighbourhood. Each node then possesses enough information at any time to build a view of the entire network and to compute the shortest paths to any other node (with the help of a Djikstra-like algorithm).

We show that these protocols actually don't fulfill the scaling properties outlined by Gupta and Kumar. However, in the last section, we study the scaling properties of OLSR and OSPF enhanced with the Fish eye technique, and we show that the enhanced protocols fulfill the theoretical scaling properties.

Note that introducing Fish eye features in OLSR is immediate, by playing on TTL and V-Time parameters in topology update packets (TC packets), as described in [5]. Introducing Fish eye features in the OSPF framework is a little less straightforward since LSAs do not feature TTL inside their format. Nevertheless, playing on age fields should essentially do the same job.

## II. MODELING AD HOC NETWORKS

In this section, we will describe how we model the different aspects of ad hoc networks.

### A. Propagation Model

We consider the following model: Time is slotted and the mobile nodes are all synchronized, i.e., transmissions occur at the beginning of slots and according to an ALOHA-like protocol (i.e., nodes select at random their transmission slots). We consider an area of arbitrary size $\mathcal{A}$ (we will ignore border effect). $N$ transmitters are uniformly distributed. We call $\lambda$ the density of transmitters per slot and per area unit, and $f$ the rate of packet transmissions per slot and per node. In this model, we will assume that the distribution of active transmitters per slot and area unit is a Poisson process.

In order to justify this assumption, note that we have a uniform distribution of nodes and that nodes use an ALOHA-like multiple access scheme. Therefore, the number and positions of transmitters at beginning of slots vary with time and changes from slot to slot like a random process. The resulting distribution of transmitters should therefore be exactly identified as a Bernoulli distribution over a uniform distribution. However, these kind of distributions are known to quickly converge to a Poisson distribution as soon as $N \rightarrow \infty$ and $fN/\mathcal{A} \rightarrow \lambda$. Thus we decided to directly work with this approximation which turns out to be very accurate in practice.

Let X be a node at a random position. We will again ignore border effects and assume that all nodes transmit at the same nominal power. The reception signal at distance $r$ is then $P(r) = r^{-\alpha}$ with $\alpha > 2$. Typically $\alpha = 2.5$. Notice that the expression of quantity $P(r)$ does not involve any fading factor. Fading is an alteration of the signal which is due to factors other than the distance (obstacles, co-interferences with echos, and so on). Fading is generally modeled via the introduction of a non-zero factor that varies randomly with time and node location. We will address the fading issue more thoroughly in Section IV.

Let $W$ be the signal intensity received by node X at a random slot. The quantity $W$ is then a random variable since the number and location of transmitters is random and vary with the slot. Let $w(x)$ be its density function. If we consider $\mathcal{A}$ to be infinite, we can use [6] where it is shown that the Laplace transformation of $w(x)$, $\tilde{w}(\theta) = \int w(x)e^{-x\theta}dx$ satisfies the identity (still with no fading):

$$\tilde{w}(\theta) = \exp\left(2\pi\lambda \int_0^\infty (e^{-\theta r^{-\alpha}} - 1)r dr\right). \quad (1)$$

Then, using standard algebra we get

$$\tilde{w}(\theta) = \exp\left(-\lambda\pi\Gamma\left(1 - \frac{2}{\alpha}\right)\theta^{2/\alpha}\right). \quad (2)$$
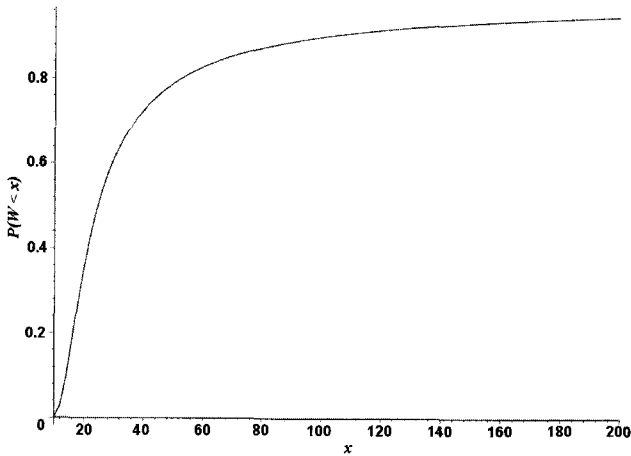
Note that if instead of an area, the node location map was a line (for instance a sequence of mobiles nodes on a road) we would then have

$$\tilde{w}(\theta) = \exp\left(-\lambda\Gamma\left(1 - \frac{1}{\alpha}\right)\theta^{1/\alpha}\right). \quad (3)$$

And similarly, if the location map was a volume (for instance a network formed by aircrafts), we would instead have

$$\tilde{w}(\theta) = \exp\left(-\frac{4}{3}\lambda\pi\Gamma\left(1 - \frac{3}{\alpha}\right)\theta^{3/\alpha}\right). \quad (4)$$

In the following, we will restrict ourselves to the case where nodes are located on a 2D map.

Fig. 1. Quantity $P(W < x)$ versus $x$ for $\alpha = 2.5$, no fading.
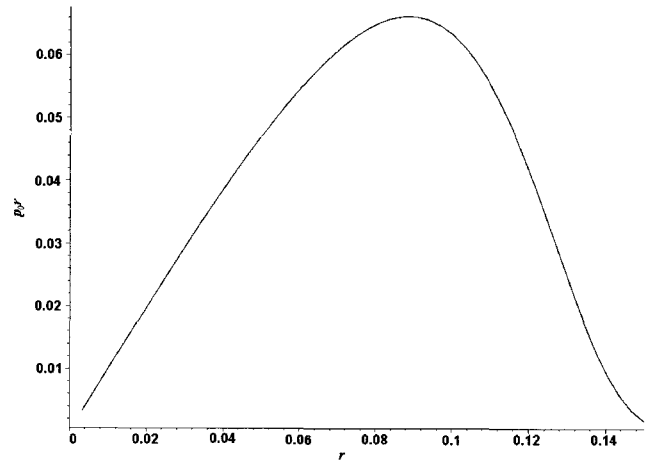


Fig. 2. Quantity $p_0 r$ versus $r$ for $\alpha = 2.5$, no fading.

## B. Neighbour Model

A node is considered to be neighbour with another node if the probability of successfully receiving Hellos from each other is greater than a certain threshold $p_0$. For example, we can take $p_0 = 1/3$. This can be achieved by keeping track of the Hello receival success rate per neighbour, as it is done in the "advanced neighbour sensing" of OLSR [4].

We will assume that a packet can be successfully decoded if its signal-over-noise ratio is greater than a given threshold $K$. Typically $K = 10$. Therefore, a node will correctly receive a packet from another node at distance $r$ with probability $P(W < r^{-\alpha}/K)$. Since Hello packets are never retransmitted, the Hello success rate from a node at distance $r$ is exactly $P(W < r^{-\alpha}/K)$. Therefore, nodes at distance $r$ are neighbours as long as $P(W < r^{-\alpha}/K) > p_0$. This is equivalent to $r < r(\lambda)$, where $r(\lambda)$ is the critical radius such that $\int_0^{r(\lambda)^{-\alpha}/K} w(x)dx = p_0$. In fact quantity $\lambda$ is a parameter which is easy to handle since by simple algebra it comes that $r(\lambda) = \lambda^{-1/2} r(1)$ (see Appendix). The surface covered by the radius $r(\lambda)$ is then the neighbourhood area $\sigma(\lambda) = \frac{\sigma(1)}{\lambda}$.

We will now compute $\sigma(1)$. We remind that factor $\lambda$ is now omitted ($\lambda = 1$). For simplification purposes, we set $C = \pi\Gamma(1 - \frac{2}{\alpha})$ and $\gamma = \frac{2}{\alpha}$. By application of the reverse Laplace transformation we get

$$P(W < x) = \frac{1}{2i\pi} \int_{-i\infty}^{+i\infty} \frac{\tilde{w}(\theta)}{\theta} e^{\theta x} d\theta. \tag{5}$$

Expanding $\tilde{w}(\theta) = \sum_n \frac{(-C)^n}{n!} \theta^{n\gamma}$, it becomes

$$P(W < x) = \frac{1}{2i\pi} \sum_n \frac{(-C)^n}{n!} \int_{-i\infty}^{+i\infty} \theta^{n\gamma-1} e^{\theta x} d\theta. \tag{6}$$

Then by bending the integration path towards the negative axis we get

$$\frac{1}{2i\pi} \int_{-i\infty}^{+i\infty} \theta^{n\gamma-1} e^{\theta x} d\theta = \frac{\sin(\pi n\gamma)}{\pi} \int_0^\infty \theta^{n\gamma-1} e^{-\theta x} d\theta$$

$$= \frac{\sin(\pi n\gamma)}{\pi} \Gamma(n\gamma) x^{-n\gamma}.$$

Fig. 1 shows the plot of $P(W < x)$ versus $x$ for $\alpha = 2.5$ and $\lambda = 1$. Let $x_0$ denote the value such that $P(W < x_0) = p_0$, therefore $r(1) = (x_0 K)^{-1/\alpha}$.

Notice that if $p_0 = 1/3$, then $x_0 \approx 20$, $r(1) = (x_0 K)^{-1/\alpha} \approx 0.12$. And then that $\sigma(1) = \pi r(1)^2 \approx 0.045$.

## C. Optimizing the Neighbourhood

In this section, we estimate the best threshold on $p_0$ to consider a neighbour node to really be in the neighbourhood. The objective is to minimize the number of retransmissions of a packet when routed to its destination. By retransmission we mean the retransmission due to multihoping as well as the retransmissions due to packet collisions. We assume that each slot is used by unicast packets (re)transmitted *à la* ALOHA until they are correctly received by the next node.

Therefore, we want to optimize the neighbourhood by excluding from it "bad" neighbour nodes that feature a too low probability of successful one hop packet transmission. They might be too far or behind an obstacle: In any case, the link is not reliable enough and the number of retransmissions needed for a correct reception is not worthy the hop distance. In other words, we want the best possible ratio of hop distance over number of retransmissions.

For this end we tune the parameter $p_0$. The optimal value does not depend on $\lambda$ as we see below. If the probability of successful transmission is $p_0$ then the average number of retransmission for one hop is $\frac{1}{p_0}$. And thus we have to optimize the quantity $p_0 r(\lambda)$, i.e., $rP(W < r^{-\alpha}/K)$. All computations done (see Fig. 2), we get $\sqrt{\lambda} r(\lambda) \approx 0.089$ and we see that the optimum $p_0 \approx 0.75$. So roughly, if a node logically excludes from its neighbourhood any neighbour from which it successfully receives less than 75% of the Hellos actually sent by this neighbour, we ensure a simple optimization of the overall number of retransmissions, on a network-wide level.

## D. Modelization of Fading

The propagation of radio waves in presence of random obstacles experiences random fading. Usually, modelization of fading consists in the introduction of a random factor $F$ modeling

signal attenuation at distance $r$: $r^{-\alpha}$. For example, $\log F$ is uniform on $[-v, v]$. In this case we, have a new expression of $\tilde{w}(\theta)$:

$$\tilde{w}(\theta) = \exp(-\pi\lambda\Gamma(1 - \frac{2}{\alpha})\phi(-\frac{2}{\alpha})\theta^{2/\alpha}), \qquad (7)$$

with $\phi(s) = E(F^{-s})$, the Dirichlet transformation of the fading. When fading is uniform on $[-v, v]$ we have $\phi(s) = \frac{\sinh(sv)}{sv}$. For any given real number $x$ we also have $P(W < xF)$ equaling

$$\sum_n \frac{(-CF(-\gamma))^n \sin(\pi n\gamma)}{\pi n!}\Gamma(n\gamma)\phi(n\gamma)x^{-n\gamma}, \qquad (8)$$

which helps the computation of $\sigma(1)$ with fading.

## III. OSPF AND OLSR SCALABILITY

Gupta and Kumar have shown in [1] that when the size of the network $N$ increases, the neighbourhood size is $O(\log N)$ and the number of hops increases at least in $\sqrt{N}/\log N$. This means that the average neighbourhood size tends to be constant when the network size increases. Our model in the previous section confirms this property since it states that when the nodes are distributed over an infinite plan, the average traffic generated inside the neighbourhood radius is equal to $\lambda\sigma(\lambda) = \sigma(1)$, a constant that we determined.

The neighbourhood size depends on the traffic control generated by each node: The bigger is the amount of control traffic, the smaller is the neighbourhood size. Therefore, performance may vary with the use of different protocols, yielding different control traffic patterns. In this section, we therefore study more precisely the scaling properties of two link state protocols: OSPF [3] and OLSR [4].

### A. Network Topology Model

We will consider that the network is uniformly distributed with density $\nu$ over an area of finite size $\mathcal{A}$. The total number of nodes in the network is $N = \nu\mathcal{A}$. If $\lambda$ is the traffic density in the network, then the average number of neighbours per node is $M = \sigma(\lambda)\nu = \sigma(1)\frac{\nu}{\lambda}$.

### B. General Control Traffic Model

The aim here is to derive the traffic density generated by the protocol control packets. Generally, there are two sources of control traffic: Neighbour sensing on one hand and topology discovery on the other hand.

Neighbour sensing is the same for all link state protocols: It consists in each node periodically transmitting a Hello message containing the list of neighbours heard by the node. By comparing their lists the nodes can determine the set of neighbours with which they have symmetric links. Let $h$ be the rate at which nodes refresh their neighbour information base and let $B$ be the maximum number of node identifiers that a slot can contain. For a network with the capacity of Wifi (1–10 Mbps) we have $B = 100$ and 1,000 slots per second. For instance, an OLSR node generates Hellos every 2 sec, i.e., $h = 1/2000$. If the neighbour list exceeds $B$ then the node generates several Hellos

per update period and distributes the neighbour list among these several Hellos. The node must generate $\lceil \frac{M}{B} \rceil$ Hellos per Hello period. Therefore, the Hellos lead to a traffic density of $h\nu\lceil\frac{M}{B}\rceil$. Omitting fractional part, we get

$$\lambda = h\nu\frac{M}{B}, \qquad (9)$$

if the Hellos is the only source of control traffic. Since $M = \sigma(1)\frac{\nu}{\lambda}$, we get

$$\frac{\sigma(1)}{M} = h\frac{M}{B}. \qquad (10)$$

In fact this is only an upper bound because the network size might be smaller than $\sigma(1)$. Therefore, taking into account only the Hello control traffic, the maximum manageable neighbourhood size is $\sqrt{B\sigma(1)/h} \approx 71$. This applies to both OLSR and OSPF as well as to any other protocol that uses such Hellos.

Topology discovery varies with each protocol. With OSPF, each node periodically broadcasts its list of adjacent links in an LSA (link state advertisement) message, and nodes re-broadcast in turn the LSA towards their neighbours. In OLSR, on the other hand, the nodes periodically broadcast TC (topology control) messages containing only a subset of their adjacent links—the MPR (multipoint relay) selector links. Moreover, only a subset of the neighbours (the MPR nodes) re-broacast the TC messages. However we will assume that in both protocols the topology discovery update period is the same, in order to compare two protocols with the same agility to adapt their topology to mobility. For instance, OLSR's TC rate per node is $\tau = 1/5000$.

### C. OSPF Specific Model

In this section, we will work on modeling the overhead induced by OSPF. The idea is to express $\lambda$ only in function of the protocol overhead. We consider no other traffic than the signalling protocol. In OSPF, a node periodically:

1. Transmits Hellos with rate $h$. A Hello contains the list of all neighbour identifiers (if the list is too long, it will take several packets on several slots).
2. Transmits LSAs with rate $\tau$. An LSA contains the list of all adjacent links.
3. Retransmits received LSAs with a large jitter to all neighbours separately (one copy per neighbour).

Therefore, the traffic density satisfies the following identity:

$$\lambda = h\nu\lceil\frac{M}{B}\rceil + \tau\nu NM\lceil\frac{M}{B}\rceil. \qquad (11)$$

In the following, we drop the ceil factor.

$$\lambda = h\nu\frac{M}{B} + \tau\nu N\frac{M^2}{B}. \qquad (12)$$

Using $M = \sigma(1)\frac{\nu}{\lambda}$ we have the identity:

$$\frac{\sigma(1)B}{M} = hM + \tau NM^2. \qquad (13)$$

This outlines a direct relation between the total size of the network $N$ and the average neighbourhood size $M$. Notice that when $N$ increases, $M$ decreases. This corresponds to the fact

that as more and more nodes are concentrated in a single radio range, interferences and collisions make more and more links perform too badly to be considered valid. Therefore, more and more nodes that are theoretically directly reachable (because physically within radio range) are not considered neighbours, and hence, $M$ decreases. The absolute minimum for $M$ is 1, below which the network does not have a significant connected component. If a single fully connected network is wished for, this threshold is raised to $M = \log N$ (see [1]).

Furthermore, the limit $M = 1$ yields a maximum network size of

$$N_{\max} = (\sigma(1)B - h)\frac{1}{\tau}, \tag{14}$$

which gives $N_{\max} = 25,000$.

On the other hand, when the network size decreases, it reaches a level where $N = M$. Below this level the network is only one hop (full meshed), and the control traffic does not saturate the neighbourhood. This corresponds to the maximum manageable neighbourhood size. From (13), we get that the maximum manageable neighbourhood size for OSPF is $N = 11$. Having an average neighbourhood size as big as possible is important in that it reduces the average number of hops needed to go from a given source to a given destination. This way the amount of retransmissions network-wide (hence the overhead) is reduced.

### D. OSPF-B

In this section, we propose an adaptation of OSPF which aims at reducing the overhead. OSPF-B slightly differs from OSPF with the fact that the nodes broadcast the LSA only once, instead of duplicated in several copies to each neighbour.

In this case the equation (13) should be rewritten in

$$\frac{\sigma(1)B}{M} = hM + \tau NM. \tag{15}$$

It then comes that in the case of OSPF-B we get a maximum manageable neighbourhood size of $N = 22$.

### E. OLSR Specific Model

In this section, we will work on modeling the overhead induced by OLSR. With this protocol, a node periodically:

1. Transmits TCs with rate $\tau$. A TC contains the list of neighbours having selected the node as MPR (its MPR selectors).
2. Retransmits received TCs only once (and with large jitter), and such only when the node has been selected as MPR by the neighbour from which it first received the TC.

Let $M_r$ be the average number of MPRs selected by a node with neighbourhood size $M$. Since the network is modeled as a disk unit graph, it comes from [7] that $M_r \leq (9\pi^2 M)^{1/3}$. Simulations show that $M_r \sim \beta M^{1/3}$ when $M \to \infty$ with $\beta \approx 5$ (see Fig. 3). Simulations were performed up to $M = 6,000,000$.

In [8], it is proven that an MPR flooding costs on average $M_r N/M$ retransmissions. Therefore, we get the following traffic density identity:

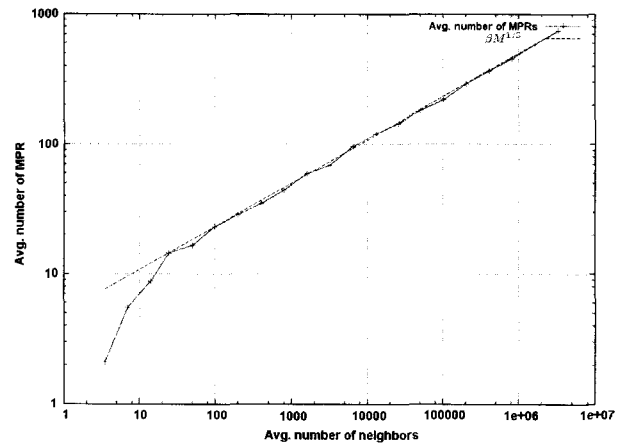$$\frac{\sigma(1)B}{M} = hM + \tau(M_r)^2 \frac{N}{M}. \tag{16}$$



Fig. 3. Average MPR set of a node versus neighbourhood size.

It then comes that the maximum manageable neighbourhood size for OLSR is with $N = 35$. Also note that this identity and the connectivity limit of $M = 1$ (which in turn implies that $M_r = 1$) gives the same maximum network size for OLSR as with OSPF, that is $N_{\max} = 25,000$.

### F. F-OLSR

In this section, we introduce a slight modification of OLSR called F-OLSR, for full optimized link state routing. In F-OLSR the TCs contain the list of all the adjacent links, and not just MPRs. Therefore, every node has the knowledge of the complete link state of the network instead of its restriction to MPR links. The TCs are still forwarded via MPR nodes. The identity for F-OLSR is then

$$\frac{\sigma(1)B}{M} = hM + \tau M_r N. \tag{17}$$

It then comes that the maximum manageable neighbourhood size for F-OLSR is at $N = 27$.

### G. Comparisons between the Protocols

In Fig. 4, we show the respective neighbourhood size versus network size for the two versions of OSPF. With Fig. 5, we show the respective neighbourhood size versus network size for the two versions of OLSR. And finally, Fig. 6 compares the network diameter as a function of the network size (number of nodes) in the case of OLSR and OSPF. The number of hops is estimated as the square root of the ratio network size over neighbourhood size.

Basically, what we can conclude from this analysis is that optimized link state (OLSR) shows here much better performance than classical link state (OSPF). However, as the network size increases, both types of approaches feature slowly decreasing (towards 0) neighbourhood size. This fails to reach the Gupta and Kumar scalability: If the network size grows to be too big, it will break down by not being able to create significant connectivity.
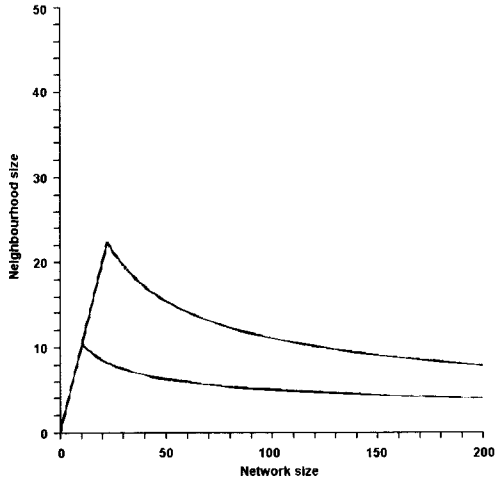
Fig. 4. Neighbourhood size versus the network size, $\alpha = 2.5$, no fading, respectively for OSPF (bottom) and OSPF-B (top).
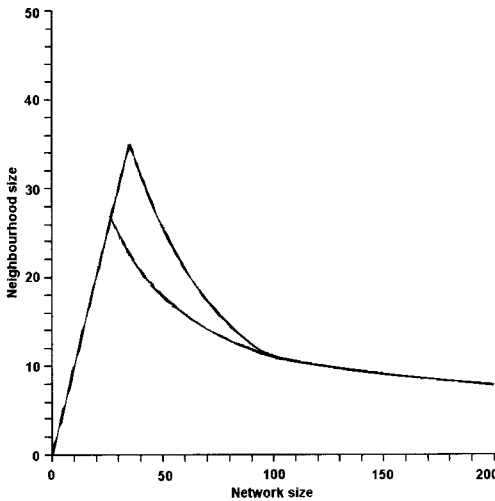


Fig. 6. Hop number estimated diameter of the network versus network size, $\alpha = 2.5$, no fading, respectively for OLSR (bottom) and OSPF (top).



Fig. 5. Neighbourhood size versus the network size, $\alpha = 2.5$, no fading, respectively for F-OLSR (bottom) and OLSR (top).

## IV. SCALING PROPERTIES OF OSPF AND OLSR ENHANCED WITH FISH EYE STRATEGY

With OSPF and OLSR as well as with any other flat routing protocol, the neighbourhood size tends to slowly decrease towards zero as the network size increases. Therefore, they do not achieve the Gupta and Kumar scaling properties. This is due to the fact that the topology information that each node in the network has to (re)transmit tends to increase linearly with the size of the network. This in turn yields an upper bound on the maximal size of the network, which we have computed to be of about 25,000 nodes for OSPF as well as for OLSR.

However, when $N$ is well below this limit of $N_{\max} = 25,000$ the two routing protocols have their neighbourhood size almost constant as $N$ increases and thus the number of hops increases in $J\sqrt{N}$. The constant $J$ depends on the nature of the routing protocol and can vary greatly. We analyzed the impact of the routing protocol on the value of this constant: We have shown that it changes quite a bit between pure link state (OSPF) and optimized link state (OLSR). In particular, we have shown that
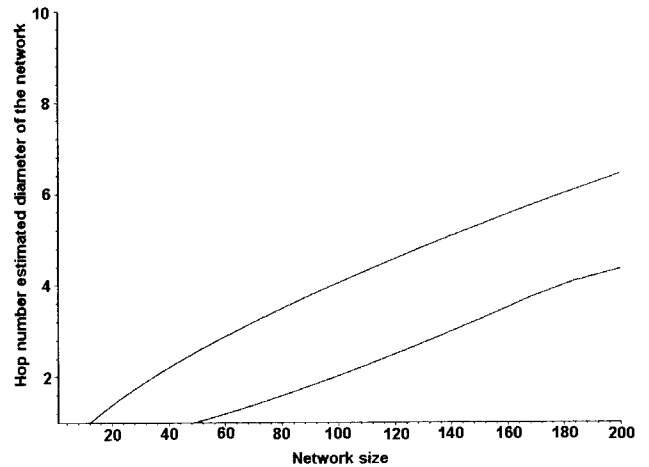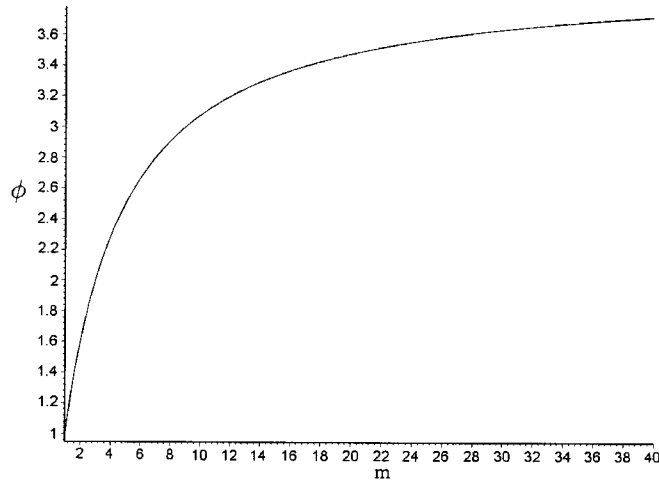
as the network size increases, the maximum manageable neighbour size is respectively of 11 nodes with OSPF, while it is of 35 with OLSR. Note that as the maximum manageable neighbour size decreases, the average number of hops (and hence retransmissions) between any random source and destination increases, therefore augmenting the overall traffic overhead.

However, both OLSR and OSPF just need minor modifications in order to reach the Gupta and Kumar scalability. In this section we describe the "Fish eye" strategy [2] that can easily be inserted inside both OSPF and OLSR frameworks. With this strategy the overall incompressible overhead induced by periodical topology updating tends to be constant instead of linearly increasing with the network size. Of course, this doesn't come without a cost, i.e., less accurate information about the link status of remote nodes. However, this cost is not expensive: It does not degrade the delivery reliability and it does not introduce additional overhead in form of longer paths (see [5]).

The principle of Fish eye strategy is that TC (or LSA) information from remote nodes are less frequently received, and the more remote, the less frequent. For example, inside the OLSR framework, nodes send TC packets with variable TTL count and V-Time. The TTL limit is the maximum number of hops a packet can be relayed before being discarded and the V-Time is the maximum time for which the information carried by this packet is considered valid. A node transmitting a packet with low TTL value ensures that the packet will be forwarded only inside the vicinity of this node, and not further. Conversely, a large TTL value (the maximum value is 255) ensures that the packet will be forwarded in the entire network.

Each node uses a decreasing function $f(D) \leq 1$ to determine the fraction of the TCs (or LSAs) which are generated with a TTL larger than $D$ ($D$ is an integer indicating the number of hops away that the TC may reach). When no Fish eye strategy is employed, $f(D) = 1$ for any value of $D$. We can assume that $\sum_{D=1}^{\infty} Df(D) < \infty$. This is indeed always the case, since $f(D) = 0$ for all $D \geq 255$. Of course, information that is received less frequently should not age as rapidly as frequently received information. This can be achieved by adequately tuning

Fig. 7. Example of function $\phi$ used for Fish eye strategy.



Fig. 8. Neighbourhood size versus the network size, $\alpha = 2.5$, no fading, respectively for OLSR (bottom) and OLSR with Fish eye (top).

the Age field in the LSAs (for OSPF) or the V-Time field in the TC packets (for OLSR).

Let us consider a node at the center of a circular network: $N$ nodes uniformly dispatched on a disk. $M$ is the average number of neighbour of the central node. In this case, the central node has $3M$ two hop neighbours, and $(D^2 - (D - 1)^2)M$ $D$-hop neighbours, for $D \leq \lfloor \sqrt{N/M} \rfloor$ (it comes that $2\sqrt{N/M}$ is the diameter of the network).

### A. Fish Eye Enhanced OSPF

Let us now consider the OSPF protocol enhanced with Fish eye strategy. The frequency of LSAs received by the central node from $D$-hop neighbours is $f(D)\tau$. Therefore, the frequency at which the central node relays LSAs is

$$\tau M \sum_{D=1}^{\lfloor \sqrt{N/M} \rfloor} (D^2 - (D - 1)^2)f(D).$$

We will call $\phi(x) = \sum_{D=1}^{\lfloor \sqrt{x} \rfloor} (D^2 - (D - 1)^2)f(D)$. It then comes that the control traffic of the central node equals to $h\frac{M}{B} + \tau\phi(\frac{N}{M})\frac{M^3}{B}$ and we get the following general identity:

$$\frac{\sigma(1)B}{M} = hM + \tau\phi(\frac{N}{M})M^3 . \qquad (18)$$

When the networks grows and $N \to \infty$ with $\phi(\infty) = 4$ we get an average neighbourhood size converging towards $M \to 7.5$.

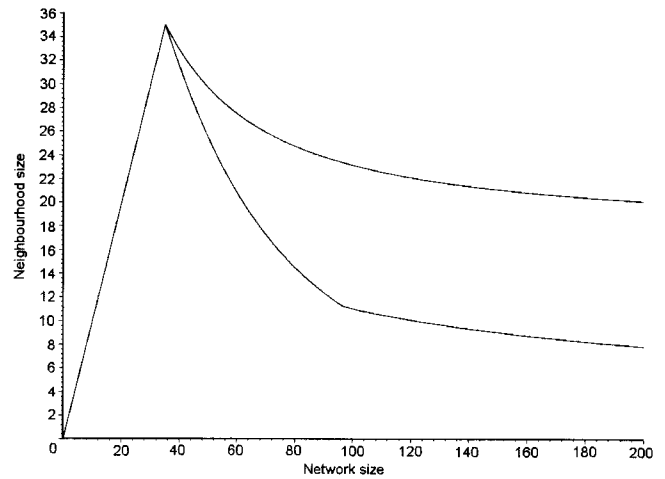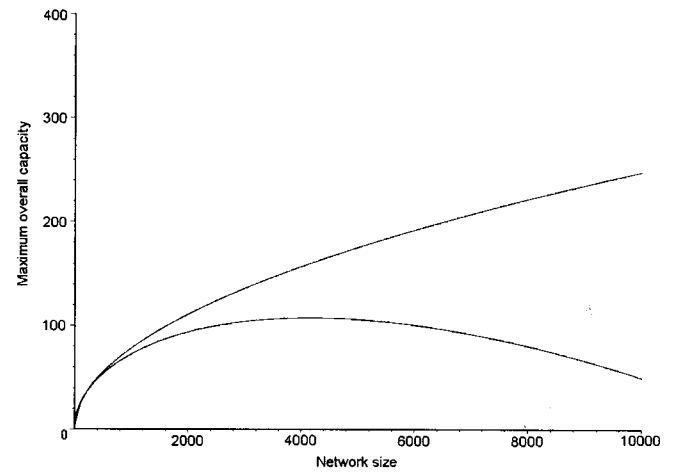### B. Fish Eye Enhanced OLSR

In the case of OLSR the identity 18 becomes

$$\frac{\sigma(1)B}{M} = hM + \tau\phi(\frac{N}{M})(M_r)^2. \qquad (19)$$

When $N \to \infty$ with $\phi(\infty) = 4$ we get an average neighbourhood size converging towards $M \to 18$. That is: Three times better than Fish eye enhanced OSPF.

Fig. 7 shows an example for function $\phi$: $\phi(x) = \frac{4x}{3+x}$. Fig. 8 shows the neighbour size evolution with respect to this function $\phi$ and compares it to basic OLSR.



Fig. 9. Maximum overall capacity versus the network size, $\alpha = 2.5$, no fading, respectively for OLSR (bottom) and OLSR with Fish eye (top).

### C. Useful Capacity

In this section, we estimate the useful capacity with the OLSR protocol. We denote $\rho$ the average quantity of data traffic generated by each node. We assume that on average, the network diameter in number of hops is $\ell\sqrt{N/M}$, where $\ell$ denotes a linear factor that depends on the actual shape of the network area $\mathcal{A}$. Therefore, each packet must be retransmitted $\frac{\ell\sqrt{N/M}}{p_0}$ times, which leads to an average traffic density (including control traffic and retransmissions) of:

$$\lambda = h\frac{M}{B} + \tau(M_r)^2\frac{N}{MB} + \frac{\rho\ell}{p_0}\sqrt{N/M}. \qquad (20)$$

Therefore, using the identity $\lambda = \frac{\sigma(1)}{M}$ we get an expression of $\rho$ as a function of $N$ and $M$. Clearly, for a given fixed $N$, $\rho$ is maximized when $M$ is minimized, the minimal value being $M = 1$. This yields Fig. 9, which displays the overall maximum capacity $N\rho$ versus network size for basic OLSR and for Fisheye OLSR (we took $\frac{\ell}{p_0} = 1$). Notice that basic OLSR with default tuning collapses at $N > 12000$, while Fisheye OLSR features an overall capacity that keeps growing in $\sqrt{N}$.

## V. CONCLUSIONS

With the help of a simple interference model, we have evaluated and compared the scalability of classical and optimized link state routing. We have shown that the the nature of the routing algorithm in use impacts essentially on the maximum manageable neighbourhood size, via the control traffic it induces. We have modeled this overhead and we have shown how it varies from one protocol to another. The maximum neighbourhood size is limited to 11 neighbours if OSPF is used, while with the same update rate parameters, the OLSR neighbour size can reach up to 35 nodes—noting that the maximum neighbourhood size is anyways limited to 71 nodes due to neighbour sensing control traffic on its own. Having a greater neighbourhood size actually reduces the overhead network-wide, by reducing the number of needed retransmissions on paths through the network.

We have also shown how both routing protocols (OLSR and OSPF) fail to scale to large networks. In fact, none of the popular ad hoc routing solutions ([4], [9], [10], etc.) really scales. There is a limit to the number of nodes in the network above which there is no significant connected component, due to incompressible topology update control traffic. We have computed this limit to be 25,000 nodes for both OLSR and OSPF. However both protocols feature practical scalability issues well within this theoretical limit. We have also shown that OSPF performs quite poorly compared to OLSR. This is not real surprise as OSPF was not designed for ad hoc environments, contrary to OLSR. These results also conform with simulations carried out independently from our work.

Finally, a simple and practical way to enable ad hoc routing to scale for larger networks has been described. We have shown how link state routing can attain the famous theoretical scaling bounds outlined by Gupta and Kumar: With the enhancement of Fish eye strategies. Such techniques can be very simply incorporated into the OLSR framework (or the OSPF protocol), and we have outlined how. Nevertheless, we have found that Fish eye enhanced OLSR still clearly outperforms Fish eye enhanced OSPF.

## APPENDIX

**Factor $\lambda$ in $r(\lambda)$.** By definition $\int_0^{r(\lambda)^{-\alpha}/K} w(x)dx = p_0$. Using the reverse Laplace transformation we have $w(x) = \frac{1}{2i\pi}\int_{-i\infty}^{+i\infty} \tilde{w}(\theta)e^{\theta x}d\theta$. Inserting this expression in the first equation and commuting integral signs, since $\int_0^{r(\lambda)^{-\alpha}/K} e^{\theta x}dx = \frac{e^{\theta r(\lambda)^{-\alpha}/K}-1}{\theta}$, yields:

$$\frac{1}{2i\pi}\int_{-i\infty}^{+i\infty} \frac{e^{\theta r(\lambda)^{-\alpha}/K}-1}{\theta}\tilde{w}(\theta)d\theta = p_0.$$

The change of variable $\lambda^{\alpha/2}\theta = \theta'$ makes $\lambda$ disappear from the $\tilde{w}(\theta)$ expression:

$$\frac{1}{2i\pi}\int_{-i\infty}^{+i\infty} \frac{e^{\theta'(r(\lambda)\sqrt{\lambda})^{-\alpha}/K}-1}{\theta'}\tilde{w}(\lambda^{\alpha/2}\theta')d\theta' = p_0.$$

Since $\tilde{w}(\lambda^{\alpha/2}\theta')$ is independent from $\lambda$ and $r(\lambda)$ appears multiplied by $\sqrt{\lambda}$, we get that $r(\lambda)$ is simply proportional to $1/\sqrt{\lambda}$: $r(\lambda) = r(1)/\sqrt{\lambda}$.

## REFERENCES

[1]   P. Gupta and P. R. Kumar. "Capacity of wireless networks," *Technical report*, University of Illinois, Urbana-Champaign, available at http://citeseer.nj.nec.com/gupta99capacity.html, 1999.

[2]   M. Gerla, X. Hong, and G. Pei, "Fisheye state routing protocol (FSR) for ad hoc networks," *Internet Draft (expired)*, available at http://www.watersprings.org/pub/id/draft-ietf-manet-fsr-03.txt, 2002.

[3]   J. Moy, "OSPF version 2," *RFC 2328*, available at http://ietf.org/rfc/rfc2328.txt, 1998.

[4]   T. Clausen *et al.*, "Optimized link state routing protocol," *RFC 3626*, available at http://ietf.org/rfc/rfc3626.txt, 2003.

[5]   T. Clausen, "Combining temporal and spatial partial topology for MANET routing - merging OLSR and FSR," in *Proc. WPMC 2003*, 2003.

[6]   P. Jacquet, "Elément de théorie analytique de l'information, modélisation et évaluation de performances," *INRIA Research Report RR-3505*, available at http://www.inria.fr/rrrt/rr-3505.html, 1998.

[7]   P. Jacquet *et al.*, "Performance evaluation of multipoint relaying in mobile ad hoc networks," in *Proc. Networking 2002*, Pisa, 2002.

[8]   C. Adjih, P. Jacquet, and L. Viennot, "Computing connected dominating set with multipoint relays," *INRIA Research Report RR-4597*, 2002.

[9]   D. Johnson, D. Maltz, and Y. Hu, "The dynamic source routing protocol (DSR)," *Internet Draft (IETF work in progress)*, available at http://www.ietf.org/internet-drafts/draft-ietf-manet-dsr-09.txt, 2003.

[10]  C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc on-demand distance vector (AODV) routing," *RFC 3561*, available at http://ietf.org/rfc/rfc3561.txt, 2003.
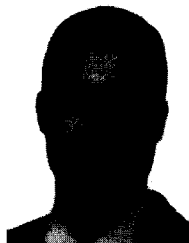
**Cedric Adjih** worked on his Ph.D. diploma at INRIA in France, and obtained it in 2001 from Versailles University (France). He is currently a researcher at INRIA, in the Hipercom team. His research interests include mobile ad hoc networking, at protocol level, from performance evaluation, design, algorithm, implementation, and measurement point of view; along with security, peer-to-peer systems, and generally networking architectures.

**Emmanuel Baccelli** received his Masters of Science degree from EURECOM and from INT in France, in 2000. He obtained the same year his D.E.A. degree in Computer Networks from Sophia-Antipolis University, France. After working at AT&T Labs in Florham Park, New Jersey, he was involved with Metro-Optix Inc. in the Silicon Valley, as development engineer. Since November 2002, he joined INRIA (Hipercom Team) as Ph.D. candidate. At the same time, since April 2003, he also joined Hitachi Ltd. as research engineer. His main research interests involve mobility, ad hoc networks, and design and analysis of network protocols and algorithms.
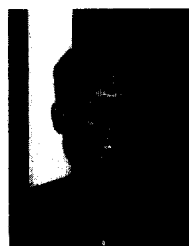
**Thomas Heide Clausen** is a graduate from Aalborg University, Denmark. He spend a number of years working in the Hipercom team at INRIA, developing and refining routing protocols for mobile ad hoc networks. He currently holds a position as assistant professor at Ecole Polytechnique, France, where he continues his research in the fields concerning mobility in next generation Internet.

**Philippe Jacquet** graduated from Ecole Polytechnique, Paris, France in 1981 and from Ecole Nationale des Mines, Paris, France in 1984. He received his Ph.D. degree from Paris Sud University, Paris, France in 1989. Since 1998, he is the scientific leader of the Hipercom Project at INRIA, France. His research interests involve information theory, probability theory, quantum telecommunication, evaluation of performance, algorithm and protocol design for mobile, and wireless and ad hoc networks.

**Georgios Rodolakis** received the Diploma degree in electrical engineering from Aristotle University, Thessaloniki, Greece in 2002, and the D.E.A. degree in Computer Science (algorithmics) from Ecole Polytechnique, France in 2003. Since October 2003 he is a Ph.D. student in project Hipercom, INRIA Rocquencourt and Ecole Polytechnique. His main research interests are in the areas of mobile ad hoc networks and design & analysis of algorithms.