QoS Routing with Bandwidth and Hop-Count Consideration: A Performance Perspective

B.Peng, A.H.Kemp and S.Boussakta
School of Electronic and Electrical Engineering, University of Leeds, LS2 9JT, UK
Email: {eenbp, a.h.kemp, s.boussakta}@leeds.ac.uk

***Abstract*— QoS Routing has been studied to provide evidence that it can increase network utilization compared to routing that is insensitive to QoS traffic requirements. However, because of its complexity, QoS routing is still the missing piece in a fully-fledged QoS architecture for the Internet. This paper exposes in detail the relationship between the routing algorithm and network condition (traffic load and the scale of the network topology) in different scenarios. We demonstrate that network size affects the performance, and give evaluation of different locations of source-destination pairs. We also point out the importance of using length of route as a metric to evaluate algorithms. This has seldom been used in previous simulation studies. This work confirms and extends earlier studies, and offers new insights for designing efficient QoS routing algorithms for future large-scale networks.**

***Index Terms*— Quality of service, routing algorithm, hop-count, bandwidth**

I. Introduction

The next generation Internet is expected to be a high-speed, high diversity, heterogeneous network, combining wired and wireless links together to delivery numerous different applications for end users. The boom of applications drives the increasing requirement of providing Quality of Service (QoS). The definition of Quality of Service from the International Telecommunication Union (ITU) is [1]:

> "Quality of Service–the collective effect of service performance which determines the degree of satisfaction of a user of the service."

The performance can be given as a set of particular constraints, such as certain amount of available bandwidth or end-to-end delay bound. The Internet Engineering Task Force (IETF) has attempted to introduce QoS in the Internet by a series of architectures such as Integrated Service (Intserv) [2], Differentiated Services (Differv) [3] and Multiprotocol Label Switching (MPLS) [4]. Among these architectures, extensive effort has been carried out on each layer of the Internet protocol stack, in particular the MAC layer, transport lay and application layer. However, most current QoS mechanisms are based on the current QoS-unaware routing. From this perspective, *QoS Routing* is the missing piece in a fully-fledged QoS architecture for the Internet [5].

This paper is based on "Impact of network conditions on QoS routing algorithms," by B.Peng, A.H.Kemp and S.Boussakta,which appeared in the Proceedings of the 3rd IEEE Consumer Communications and Networking Conference, Las Vegas, NV, USA, Jan. 2006. © 2006 IEEE.

QoS routing is a routing scheme to find a path in the network for each traffic flow able to guarantee quality parameters (e.g. bandwidth and/or delay), it has been widely accepted as an efficient way to provide the Internet with guaranteed QoS [6]–[11]. In MPLS architecture, QoS routing can be supported well between an ingress-egress node pair which is called a Label Switching Path (LSP). There are two goals for QoS routing to achieve:

1) To select routes able to meet the particular QoS requirements.
2) To provide efficient utilization of the network.

A well-known QoS routing theorem is that computing optimal routes subject to constraints of two or more additive and/or multiplicative metrics is NP-complete [9]. However, in many practical cases, bandwidth and hop-count are the only two metrics considered which relate to the data rate and delay requirements. Consequently, many QoS routing algorithms are proposed with bandwidth and hop count considerations. Three of the most popular QoS routing algorithms with bandwidth and hop count considerations are Shortest-Widest Path (SWP) [9], Widest-Shortest Path (WSP) [10] and Shortest-Distance Path (SDP) [12]. Previous simulation studies of these algorithms can be found in [8]–[11]. However, most studies actually use so called *pruning* to remove the links whose available bandwidth are lower than the requested bandwidth before calculating the explicit path. In general, pruning improves the effectiveness of QoS routing under small to moderate load by allowing connections to consider nonminimal routes. However, pruning degrades the performance under heavy load condition since nonminimal routes consume extra link capacity at the expense of other connections. In this paper, we expand upon the work in the following four dimensions:

1) All the three QoS routing algorithms are studied in two different scenarios which differentiate the impact of the traffic flow on the algorithms.
2) The algorithms with and without pruning are both considered and compared.
3) Two fixed source and destination pairs are used in both with and without background traffic conditions. Overall results from random source-destination pairs are also presented.
4) Average length of route is used as one of our metrics to evaluate different routing algorithms operating on different topologies and different traffic loads

which reflects not only the resource consumed by a connection, but also the end-to-end delay along a path.

We try to shed some light on the relationship among the routing algorithm, traffic load and topology scale and obtain detailed conclusions as reference for the future design of routing algorithms. This study is concerned with the evaluation of WSP, SWP and SDP under several different network conditions which have not been previously studied. The remainder of this paper is organized as follows. In Section II, some background knowledge and related work will be given. In Section III, we describe the three QoS routing algorithms which are evaluated in this paper. Section IV contains the detailed simulation model and the results from two Scenarios. Conclusions are provided in Section V.

II. RELATED WORK

A. Why these link metrics

Some commonly used metrics in the networking field to characterize the QoS requirements of applications are as follows:

- **Bandwidth**: When we mention the bandwidth of a link, we refer to the available bandwidth on that link. For a link to be able to accept a new flow with a certain requirement of bandwidth, that much bandwidth must be available on the link. So it is the current unused bandwidth resource which is available to a new flow.
- **Hop-count** can be used as the path cost of networks. A path with minimal hop-count is preferred because it conserves network resource as well as the most convenient indicator of path delay.
- **Delay**: There are several types of delay which packets suffers from when they travel from one node to another node along a path in networks. The most important of these delays are: *processing delay, queuing delay, transmission delay* and *propagation delay*. Processing delay is the time required to process the arrived packets in a node. For example, the time to examine the packet's header and determine the next node to send. Queuing delay is the time a packet experience at a queue as it waits to be transmitted onto the link. It can vary from zero to very long depending on how many packets in the queue are waiting to be transmitted. Transmission delay is the time required to transmit all of the packet bits into the link. For example, if the length of a packet is L Mbit and the rate of the link (aka access rate) is R (10/100 Mbps for Ethernet link as an example), the transmission delay is L/R. Hence, transmission delay is actually determined by link bandwidth. Propagation delay is the time required to propagate a bit from the beginning to the end of a link. It is the distance between two nodes divided by the propagation speed which is a little less than the speed of light. Therefore, the total delay is the sum of processing delay, queuing delay, transmission delay and propagation delay, namely:

$$d_{total} = d_{proc} + d_{queue} + d_{trans} + d_{prop} \quad (1)$$

- **Jitter** refers to the variation in the delay of the packets and is because of routers' internal queues behavior in certain circumstances, routing changes, etc.

From theory, it has been established that routing with multiple constrains is NP-complete [9] (e.g. constrained by above metrics), and hence currently unsolvable. However, pragmatically, the solution is simpler because actually only bandwidth and hop-count metrics need to be considered. This is explained in [13] as follows:

- "Although applications may care about delay and jitter bounds, few applications cannot tolerate occasional violation of such constraints. Therefore, there is no obvious need for routing flows with strict delay and jitter constraints. Besides, since delay and jitter parameters of a flow can be estimated by the allocated bandwidth and the hop count of the route [14], delay and jitter constraints can be mapped to bandwidth and hop-count constraints if needed.
- Many real-time applications will require a certain amount of bandwidth. The bandwidth metric is therefore useful. The hop count metric of a route is important because the more hops a flow traverses, the more resources it consumes."

Indeed, today, end-to-end delay is usually dominated by access rate (bandwidth) and router hops rather than by queuing delays in the routers [15].

B. Resource conservation and load distribution

As suggested in [6], in the QoS routing field, there are two contradictory approaches: either *Resource Conservation* (RC) or *Load Distribution* (LD).

Resource Conservation: The RC approach is recommended by a large community of researchers [7], [16]–[19]. Usually, routing algorithms achieve the RC by selecting the path with the minimum hop count (aka the shortest path). An example of the RC approach is the WSP algorithm, which selects a path with the highest available bandwidth from the set of shortest paths. The advantage of the RC approach is that by only using the shortest paths, it conserves network resources and allows networks to accept additional future requests. However, these characteristics lead to one major drawback; only using the links belonging to the shortest paths will lead to congestion on these shortest paths, while the longer paths may not have been used at all (when pruning is disabled). This makes the traffic of the network uneven and the resource not to be used efficiently.

Load Distribution: The LD approach is also widely considered [20], [21]. It is often achieved by using the lightest loaded path (aka the widest path). An example of the LD approach is the SWP algorithm, which selects a path with minimum hop-count from the set of widest

paths. The advantage is intuitively obvious: it can avoid the congestion on the shortest paths, and distribute the load over the whole network, in this case, the network resource can also be fully utilized. However, this approach also has its disadvantage; using a longer path for a request consumes extra resource and increases the blocking probability of future requests.

C. Pruning

There is an important technique to deal with link-constrained metrics which is commonly called *pruning*. The link-constrained metric of a path, bandwidth for example, is determined by the bottleneck link on that path. Hence, a link whose bandwidth is less than the requested bandwidth must not be included in any feasible path. Pruning simply removes these links from the network topology. By this method, pruning can guarantee that all paths on the remaining topology will meet the link-constraint (e.g. sufficient bandwidth), therefore, pruning provides a simple method to guarantee a bandwidth constraint in routing algorithms. In an N-node network with E links, pruning has $O(E)$ computational complexity and produces a sparser graph consisting entirely of feasible links [7].

III. QoS Routing Algorithms with Bandwidth and Hop-Count Consideration

Widest-Shortest Path (WSP) [10] and Shortest-Widest Path (SWP) [9] are two relative simple QoS routing algorithms which consider the bandwidth and hop-count metrics. They belong to sequential double optimization problem which means the two optimal metric will be reduced into two single metrics, where the second optimization would be processed only if there are multiple optimal paths according to the first metric.

Widest-Shortest Path: Consider a network $G(V, E)$, two metrics $w_1(v_i, v_j)$ and $w_2(v_i, v_j)$ for each link $(v_i, v_j) \in E$. Let w_1 and w_2 denote hop-count and bandwidth respectively. This algorithm is to find a path P from a source node s to a destination node t that minimize $w_1(P)$. If there are multiple paths with the same $w_1(P)$, then choose the one that maximizes $w_2(P)$.

WSP chooses a path with the minimum hop-count first, and if there are more than one such path, the one with the maximum bandwidth is chosen. Preferring the shorter path will minimize the consumption of network resource, while preferring the widest path will balance network load as well as maximize the chance to meet the required bandwidth in case of inaccurate network state information. When people refer to the WSP algorithm, in most cases, they admitted the WSP will be used among the feasible paths (i.e. the paths with sufficient bandwidth) , e.g. [8]. Commonly, pruning is used to remove all the infeasible links, and then WSP will be used on this remaining topology. However, there is significant difference between the WSP-with-pruning and WSP-without-pruning.

Basically, pruning provides a simple way to guarantee the bandwidth constraint in routing algorithms. WSP-with-pruning theoretically belongs to the link-constrained path-optimization routing problem. The second optimization (bandwidth) will be used only if there are several paths with minimal hop-count. It finds a path with minimum hops among those having sufficient bandwidth, and the one with the maximal available bandwidth is selected when there are multiple shortest paths. In general, pruning improves the effectiveness of QoS routing under small to moderate load by allowing connections to consider nonminimal routes. However, when state information is imprecise, it was observed in [7] that the source may incorrectly prune a feasible link. Even with accurate link state information, pruning degrades performance under heavy load since nonminimal routes consume extra link capacity at the expense of other connections. While WSP-without-pruning, obviously, does not prune any link in the topology, so it will always use the shortest paths. By limitation to use only the shortest path, WSP-without-pruning can guarantee that all the traffic flows are delivered by the "cheapest" way. However, the main drawback of this approach is that it only uses the links belonging to the shortest paths which make it unable to benefit from existing longer but lightly load paths.

Shortest-Widest Path: Consider a network $G(V, E)$, two metrics $w_1(v_i, v_j)$ and $w_2(v_i, v_j)$ for each link $(v_i, v_j) \in E$. Let w_1 and w_2 denote hop-count and bandwidth respectively. This algorithm is to find a path P from a source node s to a destination node t that maximize $w_2(P)$. If there are multiple path with the same $w_2(P)$, then choose the one that minimizes $w_1(P)$.

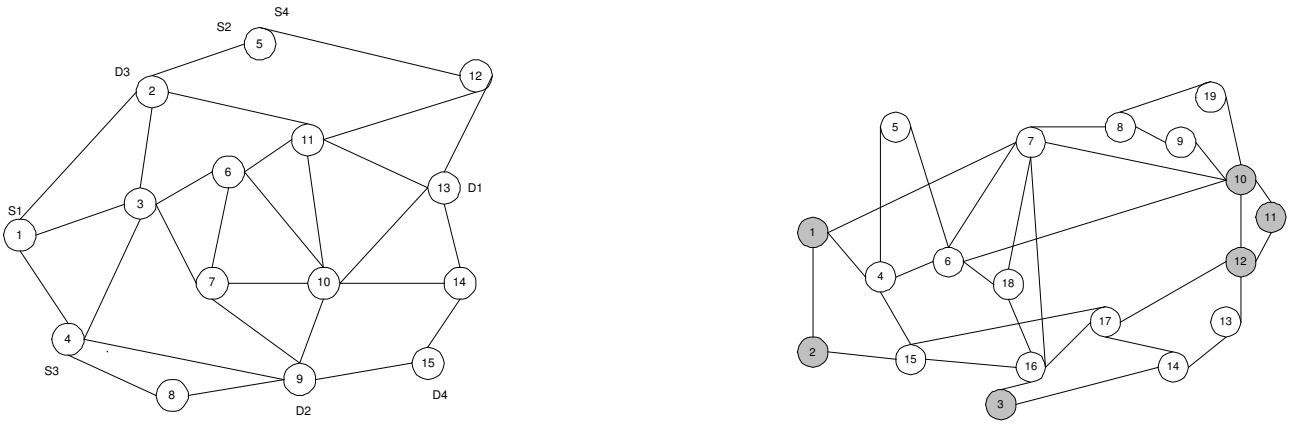
SWP chooses a path with the maximum bandwidth first. If there are more than one such path, the one with the minimum hop-count is chosen. Preferring the widest path can balance network load and therefore achieves the LD goal. Like WSP, SWP can also be used with or without pruning. SWP-with-pruning belongs to the link-constrained link-optimization routing. The second optimization (hop-count) will be used only if there are several paths with maximal bandwidth. It finds a path with maximum bandwidth among those which have sufficient bandwidth, and the one with the minimal hops is selected when there are multiple widest paths. It shares the similar problems with WSP-with-pruning but the impact of pruning in SWP is not as significant as in WSP because of the sequence of optimizing metrics.

Shortest-Distance Path: Consider a network $G(V, E)$, two metrics $w_1(v_i, v_j)$ and $w_2(v_i, v_j)$ for each link $(v_i, v_j) \in E$. Let w_1 and w_2 denote hop-count and bandwidth respectively. Let

$$dist(v_i, v_j) = \frac{1}{w_2(v_i, v_j)} \quad (2)$$

find a path P from a source node s to a destination node t that minimizes

$$dist(P) = \sum_{i=1}^{n-1} \frac{1}{w_2(v_i, v_{i+1})} \quad (3)$$



(a) Random Topology (b) ISP Topology

Figure 1: Network topologies used in simulations

The definition of SDP reflects that it prefers the lighter load path while also takes the hop-count into consideration. Previous simulation studies [8], [12] have shown that this algorithm performs consistently well in most situations.

In addition, the standard SDP algorithm can be extended to a family of polynomial link costs

$$dist(P) \quad = \quad \sum_{i=1}^{n-1} \frac{1}{w_2^n(v_i, v_{i+1})} \tag{4}$$

By changing n, it can cover the spectrum between shortest ($n = 0$) and widest ($n \to \infty$) path algorithms [12]. This is an interesting character because basically, it can be adjusted from favouring the shortest to widest path. Moreover, if this changing can be triggered by a change of network conditions, we expect this will make the algorithm more robust in different conditions. In [12], the authors have tried three cases corresponding to $n = 0.5, 1$ and 2 in max-min fair share networks.

IV. Evaluations of the routing algorithms

In this section, we present a detailed simulation study about the QoS routing algorithms discussed above. Several previous studies [6]–[8], [12] have addressed this problem by different methods and point of views. However, since the performance of the routing algorithm is effected by many factors, our work will investigate some of the "untouched" problems and studies them in more representative range of conditions and scenarios in order to shed some light on the relationship among the routing algorithm, traffic load and topology scale and to obtain more detailed conclusions as a reference for the future design of routing algorithms. Moreover, we use the average length of routes as one of our metrics for evaluations, which has seldom been used in the previous simulation studies.

A. Simulation Metrics

The metrics used to measure the performance of the algorithms are the call blocking rate (CBR), bandwidth blocking rate (BBR) and average length of routes (ALR) .

- **call blocking rate** is defined as:

$$CBR \quad = \quad \frac{number\ of\ rejected\ requests}{number\ of\ requests\ arriving}$$

- **bandwidth blocking rate** is defined in [8] as:

$$BBR \quad = \quad \frac{\sum_{i \in BG\_blk} bandwidth(i)}{\sum_{i \in BG} bandwidth(i)}$$

where BG_blk is the set of blocked sessions and BG is the set of requested sessions, $bandwidth(i)$ is the requested bandwidth of session i.

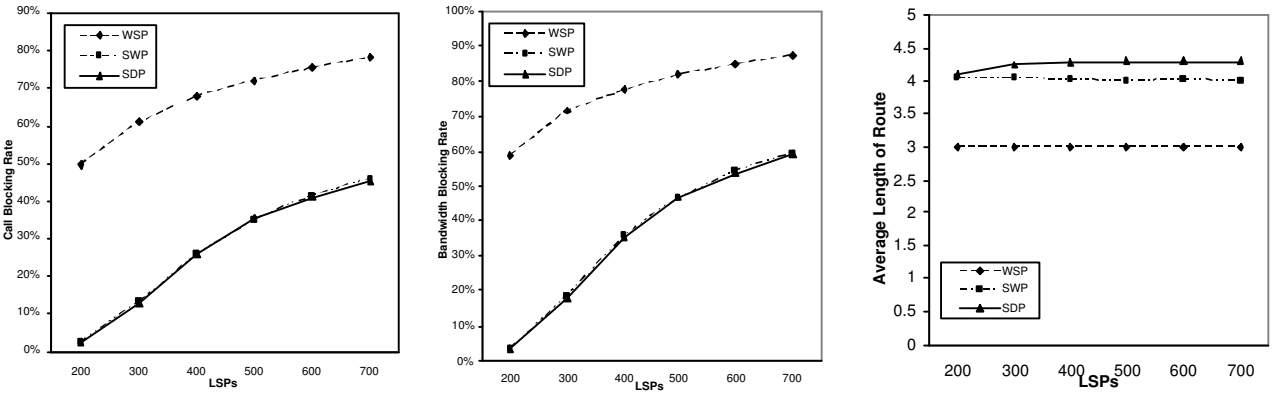
- **average length of routes** is defined as:

$$ALR \quad = \quad \frac{\sum_{i \in ACP} hopcount(i)}{ACP}$$

where ACP is the set of accepted sessions, and $hopcount(i)$ is the number of hops travelled by the accepted sessions.

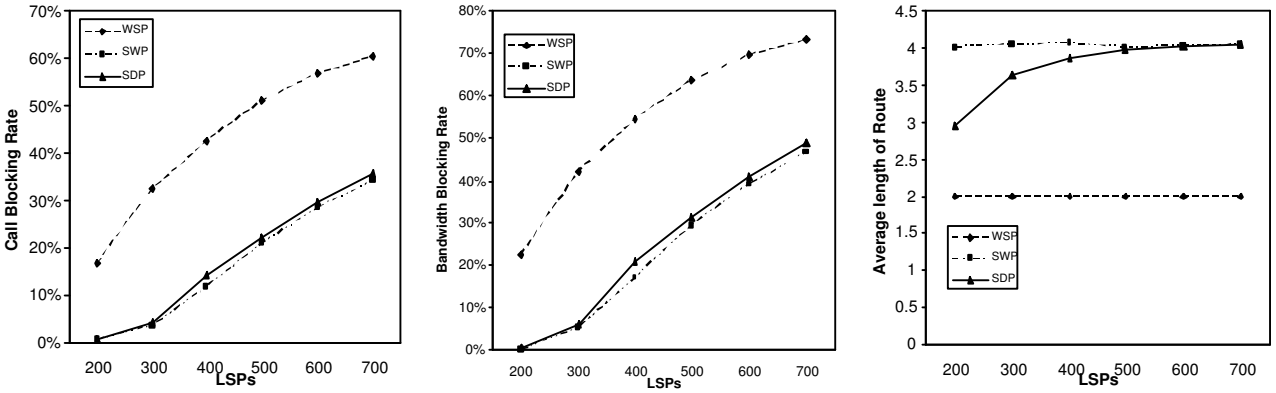
B. Simulation model

Two topologies used in our simulation are shown in Figure 1, one is a random topology borrowed from [11], the other is an ISP [22] topology which is widely used in simulation studies. All links are symmetric and have the same bandwidth of $6Mb/s$. The 4 fixed source and destination pairs (S, D) are indicated in Figure 1(a). The requested LSPs arrive following a Poisson distribution with rate λ where the requested bandwidth arriving is uniformly distributed from $64kb/s$ to $1Mb/s$, with mean value $\overline{b} = 0.532Mb/s$. Call holding time is exponentially distributed with a mean $l = 20s$. We change the network load by increasing the number of requested LSP. We assume guaranteed sessions are the only traffic type in the network. We do not consider inaccurate information due to delay, which means the topology information maintained at each node is accurate and up to date. In addition, the topology remains fixed throughout each simulation experiment, i.e. we do not model the effects of link failures. All simulations run for $100s$ and the results are the average of repeating each simulation 50 times.



(a) Call Blocking Rate (b) Bandwidth Blocking Rate (c) Average Length of Route

Figure 2: From Source 1 to Destination 1



(a) Call Blocking Rate (b) Bandwidth Blocking Rate (c) Average Length of Route

Figure 3: From Source 3 to Destination 3

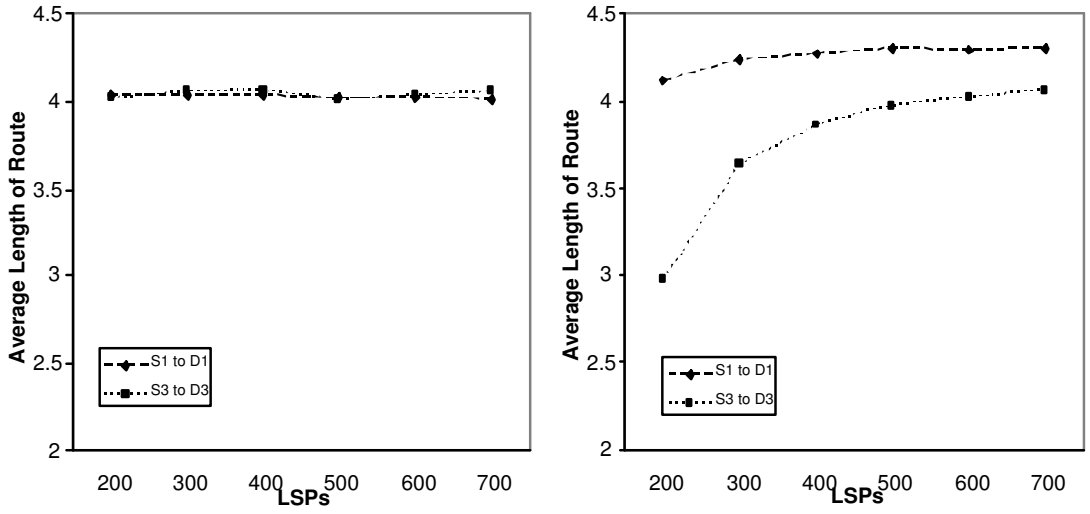
C. Scenario 1

In this section, we study the situation of transmitting between the single source-destination pairs in a random topology [23]. All the LSP requests are sent only from one source and received by another destination. Two pairs considered here are: S1-D1 and S3-D3. In Figure 1(a), S1-D1 is located each side of the whole network and it reflects the situation of transmitting packet through the entire network. The S3-D3 shows the situation of short distance transmission. Although this kind of only single source-destination transmission does not typically happen in practise, this particular environment is suitable for reflecting the properties of routing algorithms because traffic generated by other sources in networks reduces the performance impacted originally due to the algorithms themselves.

As shown in Figure 2 and Figure 3, when the network load is increased by increasing the number of LSPs, both CBR and BBR will increase. WSP consistently performs worse than SWP and SDP according to the blocking rate because it only uses the shortest path between source and destination pair even if it has been heavily congested. Moreover, from Figure 1(a) we can see that for S1-D1 pair, there is only one shortest path which follows nodes 1-2-11-13, however, for S3-D3 pair, two shortest paths (4-1-2 and 4-3-2) can be used. Therefore, comparing the CBR and BBR in Figure 2 and Figure 3, we can see that for the S3-D3 blocking is much lower than for the S1-D1 pair (more than a 50% reduction). This suggests that WSP is better suited for the topologies whose number of min-hop paths is high (or simply saying that the connectivity degree is high). A more detailed study of the influence of topology can be found in [6].

As shown in Figure 3, when transmitting between S3 and D3, the blocking rate of SWP is consistently lower than SDP. While in Figure 2, SWP and SDP's performance are virtually the same. This suggests that the distance (in terms of average hop-count) of source and destination pairs do affect the performance of routing algorithms. SWP always prefers wider path first, which especially benefits the performance for transmitting over a short distance according to its lower blocking rate. The reason is that for a short distance transmit, RC is less crucial than long distance transmission.

Figure 2(c) and Figure 3(c) show that the route length depends on the particular QoS routing algorithm and



(a) SWP (b) SDP

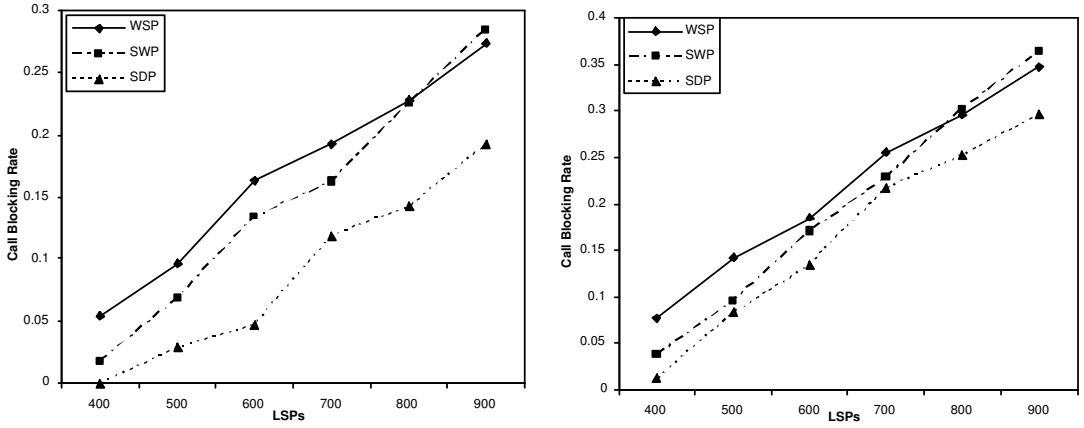
Figure 4: Route Length for different source-destination pairs

obviously the location of the source-destination pair. WSP always chooses the shortest path for S1-D1 and S3-D3, which is 3 hops and 2 hops respectively. We are particularly interested here in SWP and SDP. For SDP, on average it chooses a longer route as the traffic load becomes high. While, SWP is less sensitive to traffic load and location of source-destination pairs (a more obvious comparison is shown in Figure 4). SWP always selects the widest path no matter how many hops it has. It provides a nearly constant length of route over a wide range of network load. In Figure 2(c), the route length of SDP is continuously longer than SWP and slowly increases as network load becomes high. While in Figure 3(c), the route length of SDP is much shorter than SWP when network load is light, when network load increases, the route length of SDP increases rapidly to be almost the same as SWP when network load is high. This suggests that the different location of source-destination pairs do have an impact on SWP and SDP and lead them to have contrary performance according to the average route length. The reason is SWP only uses the bandwidth as its first consideration, while SDP balances the impact of bandwidth and hop-count with traffic load. For real time applications, in this case for example, SWP is preferred by S1-D1 and SDP is preferred by S3-D3 because a shorter route is more important for time sensitive applications. The two metrics we use here, the blocking rate (CBR and BBR) and the length of route, sometimes conflict with each other. For example, as shown in Figure 3(a) and (b), SWP outperforms SDP, i.e. has a lower blocking rate. However, in Figure 3(c), SDP outperforms SWP because of a lower route length. If they can not be achieved simultaneously, we need to balance which one is more important under which conditions. As we have mentioned, for time sensitive applications, the shorter route maybe preferred.

D. Scenario 2

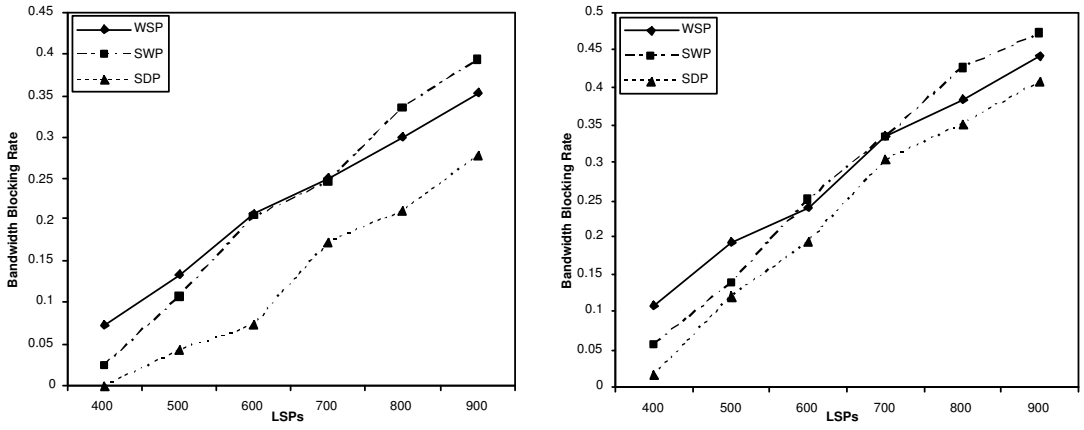
In this section, we first study the situation of transmitting between the fixed source-destination pairs with background traffic in a random topology [24]. Unlike scenario 1, this time, all nodes will randomly be selected as source and destination pairs to provide the background traffic except the 4 fixed source-destination pairs indicated in Figure 1(a). This is more practical situation because the performance will be influenced by traffic generated by all other nodes in the topology. Then, we will provide the overall performance evaluations in the ISP topology. The source will be randomly selected in node 1, 2, 3 (shaded nodes) and destination in node 10, 11, 12 (shaded nodes) in Figure 1(b).

Figure 5 and Figure 6 show how the call blocking rate and bandwidth blocking rate vary as a function of the number of requested LSPs. For the three algorithms without pruning, when the network load is increased by increasing the number of LSPs, both CBR and BBR will consistently increase. When the load is very light, SWP and SDP have similar performance with SDP slightly better, while WSP performs worst in this condition. The reason is that without pruning, only the links belonging to the shortest path can be used, even the longer links which exist are idle. WSP, which is strictly a minimum-hop path, limits the utilization of network resource in this light load condition. However, the curves of SWP and WSP cross at 800 LSPs and 700 LSPs in Figure 5 and Figure 6 respectively as the network load increases and from the intersection, WSP begins to gradually outperform SWP. This is because the preference of RC and LD is changing with the increase in load and the request of LSP with higher bandwidth requirement have a higher chance of being rejected. Consequently, the intersection in BBR is higher than in CBR. SDP consistently performs best among these three algorithms under all load conditions



(a) From Source 1 to Destination 1 (b) From Source 3 to Destination 3

Figure 5: Call blocking rate as a function of the number of LSP requests in random topology

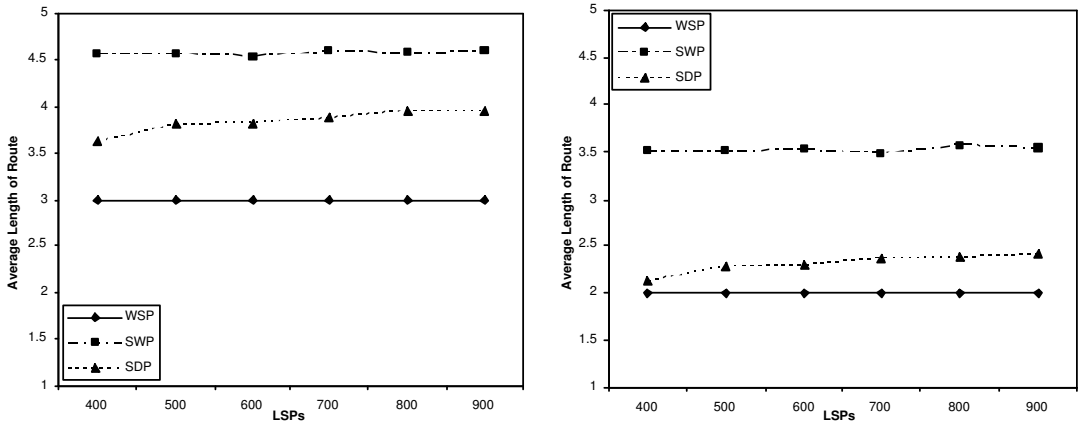


(a) From Source 1 to Destination 1 (b) From Source 3 to Destination 3

Figure 6: Bandwidth blocking rate as a function of the number of LSP requests in random topology

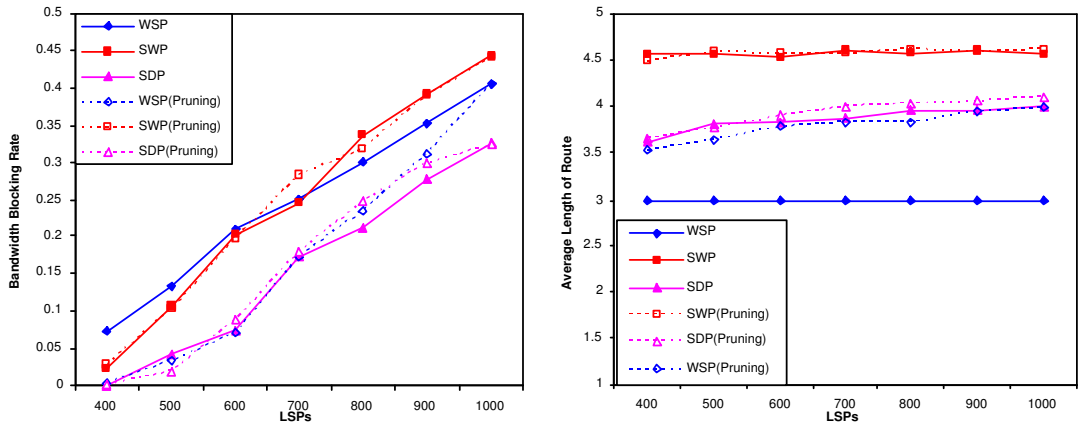
because it can dynamically balance the RC and LD in different load conditions. These results confirm that WSP prefers RC by only using the shortest path, it conserves network resources and allows networks to accept additional future requests which is especially advantageous under heavy load conditions. While the SWP prefers LD by using the lightest load path, it can avoid the congestion on the shortest paths by using longer but lighter loaded paths and distribute the load over the whole network. However, using longer paths consume extra resource and increases the blocking rate of future requests, this is especially apparent under heavy load conditions.

Figure 7 shows the average route length as the function of the number of requested LSP. Without pruning, WSP always chooses the shortest path for S1-D1 and S3-D3, which is 3 hops and 2 hops respectively. SDP can balance the hop count and traffic load and chooses a longer route as the load becomes high, while, SWP is less sensitive to traffic load. SWP always selects the widest path (according to the available bandwidth) no matter how many hops it has. It provides a nearly constant length of route over a wide range of network load for different locations of source-destination pairs. Comparing the route length of SDP, we found that in S3-D3, the route length is slightly increasing (from 2) with network load compared with the minimum route length (2 hops), while in S1-D1, even in light load conditions, the route length jumps to 3.6 hops which is higher than the minimum route length between S1 and D1. This is because there are more shorter path for S3-D3 (4-1-2 and 4-3-2) than S1-D1 (1-2-11-13). Referring back to Figure 5 and Figure 6, we can see that the difference of blocking rate between SDP and the other two algorithms is less in S3-D3 than in S1-D1. In Figure 6(a), the BBR of SDP is much lower than WSP and SWP under medium load condition (over a 10% reduction), while in Figure 6(b), the difference is smaller (less than 5%). These results reflect that the difference of location and distance of source and destination pairs do affect the performance of routing algorithms. SDP achieves even better performance for transmitting in longer distance with many hops between source and destination. Figure 8 compares the performance of the three algorithms with



(a) From Source 1 to Destination 1 (b) From Source 3 to Destination 3

Figure 7: Route length as a function of the number of LSP requests in random topology

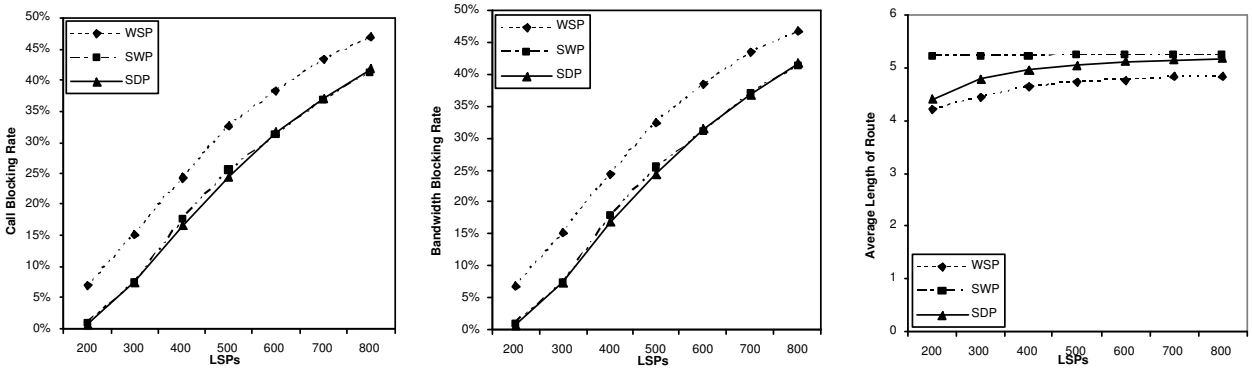


(a) Bandwidth Blocking Rate (b) Average Route Length

Figure 8: Three algorithms with/without pruning between source 1 and destination 1 in random topology

and without pruning between S1 and D1. As we expect, pruning improves the effectiveness of the algorithm with a strong preference for minimum-hop routes, WSP for example. The reason is that by pruning the shortest path, the non-minimal routes can be used to avoid the shortest path which might be congested. Hence, in Figure 8(b), we can see that the route length of WSP with pruning is much longer than the WSP without pruning. And pruning is especially effective under light to moderate load, which has been showed in Figure 8(a). However, when increasing the load, the curve of WSP with pruning is approaching to the WSP without pruning and when the load is heavy, the WSP without pruning begins to outperform the WSP with pruning. Therefore, pruning degrades performance under heavy load condition because using nonminimal routes consumes extra link bandwidth, which consequently will increase the blocking probability of future requests, this is especially serious when network load is high. Moreover, if the link-state information is not up to date, the source may incorrectly prune a feasible link, causing a connection to follow a nonminimal route when a minimal route is available [7]. For the algorithms without strong preference for the minimum-hop path, such as SWP and SDP, we do not see much impact.

Finally, we present the study of transmitting between the random source-destination pairs using the ISP topology. All the LSP requests are chosen randomly from three source nodes and sent to three destinations randomly, so all the traffic generated by these pairs will impact each other and we are interested in the overall performance. Figure 9(a) and (b) show that WSP consistently performs the worst according to CBR and BBR among the three algorithms. This is not the case we found in the previous simulations. The main reason is that the traffic source is limited to 3 nodes this time, which reduces the impact of the load generated by other source. This has a positive effect on the WSP algorithm. SDP performs slightly better than SWP under all traffic load. As shown in Figure 9(c), WSP avoids longer routes by selecting only the shortest path. The route length of SDP slowly grows with network load, and it is constantly shorter than SWP. This suggests that although SWP and SDP have similar performance



(a) Call Blocking Rate (b) Bandwidth Blocking Rate (c) Average Length of Route

Figure 9: Random source-destination pairs in ISP topology

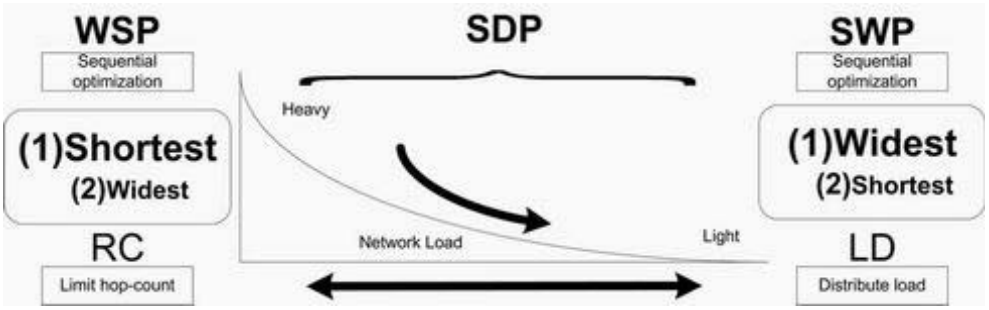


Figure 10: Relationship of three QoS routing algorithms

according to CBR and BBR, the shorter route length of SDP will benefit time sensitive applications in this condition.

V. Conclusions

The importance of the bandwidth and hop-count has driven much attention on these special yet important sub-problems in QoS routing. Several approaches have been proposed to select a path which can satisfy the particular QoS requirements of applications. The two main approaches discussed in this field are Resource Conservation (RC) or Load Distribution (LD). However, selecting a path which can utilize the network resource efficiently as well as meet the application requirements has not previously been well understood. We studied three QoS routing algorithms with bandwidth and hop-count considerations: Widest-Shortest Path algorithm, Shortest-Widest Path algorithm and Shortest-Distance Path algorithm. In general, the WSP gives high priority to limiting the hop-count. The SWP gives high priority to balancing the network load. The SDP can dynamically balance the impact of hop-count and the path load by the distance function. The relationship of these three algorithm is shown in Figure 10. Pruning is treated as an efficient method to improve the effectiveness of QoS routing under small to moderate load condition, while in heavy load condition, careful consideration is needed to balance the tradeoff between minimal and nonminimal routes.

Previous simulation studies [6], [8] have concluded generally that for QoS routing algorithms, the conflicting goals of RC and LD, can not be achieved using one of these algorithms. Under complicated network conditions, there are many factors (e.g. traffic load, network topology and scale) which will influence the final performance of the routing algorithms. More focused research work is needed to determine the relationship between these factors and routing algorithms and this detailed observation would be valuable for the future design of routing algorithms. From our simulation studies, the main observations are as follows:

- Different locations and distance of source and destination nodes will lead the routing algorithms to have different performance. This suggests an appropriate routing algorithm should be designed according to the network size. It is not practical for a single routing algorithm to suit all network scales.
- In general, pruning improves the performance of routing algorithms with shortest path preference in light and medium traffic conditions. However, it degrades the performance in heavy load conditions by using nonminimal routes.
- Although the metrics of CBR and BBR are commonly used in previous studies, the route length is another crucial metric needed to be considered for evaluating algorithms, especially for delay sensitive applications.
- RC and LD approaches seem not to be achievable

simultaneously by these threes existing algorithms, hence we need to assess which one suits which conditions. Moreover, new routing algorithms are anticipated which can balance RC and LD to network load properly without requiring additional information which would add signalling overhead.

Notably this study is related to a basic architecture without considering the inaccurate information introduced by update policy which will also affect the routing performance. Hence, future work will investigate the performance of the QoS routing algorithms under different update policies as well as different network models to understand the extent to which they influence the algorithm applicability and effectiveness.

REFERENCES

[1] "Telecommunication standardization sector of itu. itu-t recommendation e.800: Terms and definitions related to quality of service and network performance including dependability," 1994.

[2] R. Braden, D.Clark, and S. Shenker, "Integrated services in the Internet architecture: an overview," RFC 1633, June 1994.

[3] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss, "An architecture for differentiated service," RFC 2475, Dec. 1998.

[4] E.Rosen, A.Viswanathan, and R. Callon, "Multiprotocol label switching architecture," RFC 3031, Jan. 2001.

[5] X. Masip-Bruin, M. Yannuzzi, J. Domingo-Pascual, A. Fonte, M. Curado, E. Monteiro, F. Kuipers, P. V. Mieghem, S. Avallone, G. Ventre, P. Aranda-Gutie, M. Hollickf, R. Steinmetzf, L. Iannoneg, and K. Salamatian, "Research challenges in qos routing," *Computer Communications*, vol. 29, no. 5, pp. 563–581, 2006.

[6] K.Kowalik and M.Collier, "Should QoS routing algorithms prefer shortest paths?" in *Proc. IEEE International Conference on Communications (ICC '03)*, May 2003, pp. 213–217.

[7] A.Shaikh, J.Rexford, and K.G.Shin, "Evaluating the impact of stale link state on quality-of-service routing," vol. 9, pp. 162 – 176, Apr. 2001.

[8] Q.Ma and P.Steenkiste, "On path selection for traffic with bandwidth guarantees," in *Proc. IEEE International Conference on Network Protocols*, Atlanta, Georgia, Oct. 1997, pp. 191 – 202.

[9] Z.Wang and J.Crowcroft, "Quality-of-service routing for supporting multimedia applications," vol. 14, pp. 1288–1234, Sept. 1996.

[10] R.Guerin, A.Orda, and D.Williams, "QoS routing mechanisms and OSPF extensions," in *Proc. IEEE GLOBECOM*, Phoenix, AZ, Nov. 1997, pp. 1903–1908.

[11] M.Kodialam and T. Lakshman, "Minimum interference routing with applications to MPLS traffic engineering," in *Proc. IEEE INFOCOM*, Mar. 2000, pp. 884–893.

[12] Q.Ma, P.Steenkiste, and H.Zhang, "Routing high-bandwidth traffic in max-min fair share networks," in *Proc. ACM SIGCOMM*, Stanford, CA, Aug. 1996, pp. 206–217.

[13] X.Xiao and L.M.Ni, "Internet QoS: a big picture," vol. 13, pp. 8–18, Mar.-Apr 1999.

[14] A. K. J. Parekh, "A generalized processor sharing approach to flow control in integrated services networks," Ph.D. dissertation, MASSACHUSETTS INSTITUTE OF TECHNOLOGY, Feb. 1992.

[15] J. F. Kurose and K. W. Ross, *Computer networking :a top-down approach featuring the Internet*. Boston ; London: Pearson/Addison Wesley, 2005.

[16] G.Apostolopoulos, R.Guerin, and S.Kamat, "Implementation and performance measurements of QoS routing extensions to OSPF," in *Proc. IEEE INFOCOM*, Mar. 1999, pp. 680–688.

[17] Q.Ma and P.Steenkiste, "Quality-of-service routing for traffic with performance guarantees," in *Proc. IFIP International Workshop on Quality of Service*, New York, May 1997, pp. 115–126.

[18] C. Pornavalai, G. Chakraborty, and N. Shiratori, "QoS based routing algorithm in integrated services packet networks," in *Proc. IEEE International Conference on Network Protocols*, Atlanta, Georgia, Oct. 1997, pp. 167–175.

[19] I. Matta and A. Shanka, "Dynamic routing of real-time virtual circuits," in *Proc. IEEE International Conference on Network Protocols*, Washington, DC, USA, 1996, pp. 132–139.

[20] B. Awerbuch, Y. Azar, S. Plotkin, and O. Waarts, "Throughput-competitive on-line routing," in *Proc. 34th Annual Symposium on Foundations of Computer Science*, Nov. 1993, pp. 32–40.

[21] A. Kamath, O. Palmon, and S. Plotkin, "Routing and admission control in general topology networks with poisson arrivals," in *SODA '96: Proceedings of the seventh annual ACM-SIAM symposium on Discrete algorithms*, Philadelphia, PA, USA, 1996, pp. 269–278.

[22] G. Apostopoulos, R. Guerin, S. Kamat, A. Orda, and S. K. Tripathi, "Intradomain QoS routing in IP networks: A feasibility and cost/benefit analysis," vol. 13, pp. 42–54, Sept.-Oct 1999.

[23] B.Peng and A.H.Kemp, "Matching QoS routing algorithms with network conditions," in *Proc. London Communications Symposium*, London, UK, Sept. 2005, pp. 53–56.

[24] B.Peng, A.H.Kemp, and S.Boussakta, "Impact of network conditions on QoS routing algorithms," in *Proc.* IEEE *Consumer Communications and Networking Conference*, Las Vegas, NV, USA, Jan. 2006.

Bo Peng received a B.Eng from Xidian University, Xi'an, China, in 2002. He is currently a research student in School of Electronic and Electrical Engineering, University of Leeds, UK. His research interests are in networking area with focus on routing, quality of service, routing and positioning in mobile ad hoc network, network simulation and performance analysis.

Andrew H. Kemp received a BSc from the University of York, UK, in 1984 and PhD from the University of Hull, UK, in 1991. His doctoral studies investigated the use of complementary sequences in multi-functional architectures for use in CDMA systems. He spent several years working in Libya and South Africa assisting in seismic exploration and worked at the University of Bradford as a research assistant investigating the use of Blum, Blum and Shub sequences for cryptographically secure 3rd generation systems. More recently he helped develop wireless fieldbus systems for industrial sites and has been lecturing at the University of Leeds, UK in communications for the last 6 years. Andrew has over 30 scientific journal and conference papers and a book

chapter published. His research interests are in multi-path propagation studies to assist system development and wireless broadband connection to computer networks incorporating quality of service provision.

Said Boussakta received the Ingenieur dEtat degree in electronic engineering from Ecole Nationale Polytechnque of Algiers (ENPA), Algiers, Algeria, in 1985 and the Ph.D. degree in electrical engineering in the area of signal and image processing from the University of Newcastle upon Tyne, Newcastle upon Tyne, U.K., in 1990. From 1990 to 1996, he was with the University of Newcastle upon Tyne as Senior Research Associate in digital signal and image processing. From 1996 to 2000, he was with the University of Teesside, Teesside, U.K., as Senior Lecturer in communications. He is currently a Senior Lecturer in communications at the School of Electronic and Electrical Engineering, University of Leeds, Leeds, U.K., where he is lecturing in modern communications networks, communications systems, and signal processing. His research interests are in the areas of digital communications, coding and cryptography, digital signal/image processing, and fast DSP algorithms and transforms. He has authored and co-authored a number of publications. Dr. Boussakta is a member of IEEE Signal Processing, Communications, and Computer Societies and of the IEE.