Parameterised Gradient Based Routing (PGBR) for Future Internet

Sasitharan Balasubramaniam, Dmitri Botvich, Julien Mineraud, William Donnelly
Telecommunication Software and Systems Group
Waterford Institute of Technology, Carriganore
Waterford, Ireland
{sasib, dbotvich, jmineraud, wdonnelly}@tssg.org

Abstract— **The current internet infrastructure is facing a number of limitations that is not suitable to meet the growing number of services and users. In particular, one aspect that requires enhancement is routing, where original routing concepts were designed for static traffic patterns with minimal variations and supporting mainly low throughput traffic (e.g. Data). As the number of users as well as services supporting the user grows, the current routing mechanisms will not be feasible. In this paper we present a gradient based distributed routing technique that is based on discovering routes through a gradient field created in the topology. The gradient calculation is based on weighted sum of a number of components, which modifies the gradient field as the network load changes. Simulation results have also been presented to validate the proposed routing algorithm.**

Keywords-Dynamic and Distributed Routing, Future Internet

I. INTRODUCTION

The communications network research community is currently pursuing programs to enhance the Future Internet infrastructure in order to efficiently and flexibly support end user requirements. The future will witness content services driving the user's needs, where in recent times various services have gained immense popularity, in particular multimedia services (e.g. VoIP and VoD). As new and more advanced services are developed, this will lead to more stress on the underlying communication networks, which have been designed with protocols that support primarily data traffic with very static traffic patterns. In particular, one aspect of the internet that will need to be enhanced is routing (for both intra and inter domain). A number of criterias will need to be addressed to support new routing mechanisms for Future Internet, which includes (i) **scalability** [16] – where future communication systems will witness large number of nodes (this includes the number of AS as well as nodes within each AS [16]), (ii) **adaptability** [16] – where network configurations can be adaptively tuned to environmental changes, and (iii) **efficiency** [16] – where networks should be able to adaptively change their configurations while minimising disruptions to existing and future traffic flows.

Based on the criteria described above, the current intra-domain routing protocols are faced with a number of limitations. Future communication systems must cope with varying environmental changes, which maybe due to (i) user's frequently changing service requirements in the network (content as well as network services), (ii) topology changes, or (iii) changes of preferences by the operators with respect to resource utilisation. In the case of (i), it is envisioned that greater flexibility will be created for users, where users will be able to select and change network service preferences to suit their content's Quality of Service (QoS) requirement. The flexibility will allow users to select suitable network service which will maximize their QoS for their application, at the right cost. In the case of (ii), topologies may change due to long term of short term failures (transient failures), where short term failures may be due to planned maintenance or deployments of new protocols [1].

In this paper, we present a new routing mechanism known as Parameterised Gradient Based Routing (PGBR). PGBR is a distributed routing technique for core networks that relies purely on local interactions between network nodes. The proposed routing algorithm is based on creation of chemical fields within the network, where the chemical field will create a gradient that attracts route discovery from a specific source to destination. Each route discovery is flow-based, and is discovered on a hop-by-hop basis, where each hop will select the link with the highest gradient attraction. The advantage of the proposed mechanism is the ability to be able to change priorities of different parameters with the gradient equation to suit the topology size or requirements of traffic type. In order to validate our proposed routing algorithm, a number of simulation tests have been performed and compared with existing routing techniques.

The paper is organized as follows: Section 2 presents the related work for route management. Section 3 presents the PGBR routing mechanisms. Section 4 presents our simulation work to validate our solution. Section 5 presents the role PGBR can play in the Future Internet, and lastly section 6 presents the Conclusion.

II. RELATED WORK

In recent years, research in network routing has gained phenomenal momentum. Current routing techniques uses IGP routing protocols such as OSPF [8] [10] in IP based networks or Spanning Tree in the case of Metro Ethernet. The mechanism is based on a partial centralised solution, where each node is required to maintain a central view of the entire topology. Routing protocols can be subdivided into static and dynamic routing protocols, where dynamic routing supports dynamic changes within the network [9]. Although traditionally, OSPF has been categorised as a static routing protocol, in recent years a lot of focus has been made to make the protocol more dynamic [4] [2]. A number of investigation have looked at creating more dynamic OSPF through manipulation of link weights to determine optimal route that can support various QoS constraints [11] [12]. Iwata and Fujita [7] proposed a hierarchical multi-layer QoS routing system in order to support dynamic SLA management. The solution accepts dynamic SLA changes from the users, which is submitted to a policy server of the core network to perform the routing, which is based on QoS-Enabled OSPF. Applegate and Cohen [13] calculated the routes using OSPF with minimal knowledge of the traffic demand. Although a lot of initiatives have been extended to OSPF, the protocol is still not ideal for fluctuating traffic environments. In the event of traffic changes, notifications are broadcasted to other nodes to trigger global route re-calculations (e.g. shortest path). The proposed solution is time consuming, and is not suitable for dynamic traffic, which can lead to route instability or lengthy re-routing process in the event of network failures.

A number of hop-by-hop distributed routing algorithms have also been proposed [5] [6]. Base et al [17] proposed a gradient hop-by-hop routing paradigm using potentials (e.g. forwarding packets in the direction of maximum force). There are a number of limitations with the potential based routing. Firstly, the potential value on each node is the weighted sum of traffic load on the node as well as shortest path to the destination. However, the dynamic change of the potential value of each node is reliant on the load in the node itself and does not provide an accurate picture of congestion in the links (e.g. a node with 6 links should not be considered in the same priority of a node with 3 links if both normalized load value are the same). Secondly, the authors mention that the routing technique is not flexible in discovering alternative routes when the sending rates are high compared to link capacities. Bio-inspired techniques have also been applied to routing, where Leibnitz et al [3] proposed a biologically inspired technique towards self-adaptive routing for overlay networks based on attractor-selection techniques.

III. PARAMETRISED GRADIENT BASED ROUTING

Fig. 1 illustrates the proposed PGBR routing mechanisms.

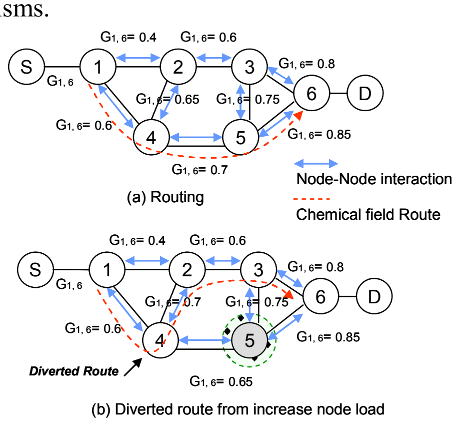


Figure 1. Illustrative example of PGBR routing

TABLE I TABLE OF NOTATIONS

Symbol	Definition
$G_{n,n\to j}$	Gradient value of each link on a node, that is used for route discovery (*n – node; n, j ∈ links, where n ≠ j*)
H_k	Hop count value (k ∈ final value, source value)
$h_{j(s,d)}$	Normalized hop count value
α, β, γ	Weights for $G_{n,i\to j}$ calculation
Φ_n	Load calculation for node *n*
$p_{dC,(S\to D)}$	Path discovery packet between source *s* and destination *d*
$l_{n,i\to j}$	Link spare capacity between node *n* and *j*
FN	Forbidden nodes list of $p_{dC,(S\to D)}$
PL_T	Path length threshold

As mentioned in the introduction, the distributed routing algorithm is based on local interactions between each node and its neighbour. The route automatically discovers the path by attracting to the highest gradient, and avoiding nodes with high load by diverting towards parts of the networks that is lightly loaded. Fig. 1 illustrates an example of PGBR, where a route along path 1 – 4 – 5 – 6 is established for S, D = (1→6). At node 4, the gradient value of node 4 to 2 ($G_{1,6,4\to 2} = 0.65$) is lower than gradient value from node 4 to 5 ($G_{1,6,4\to 5} = 0.65$). However, as the load on node 5 (showed from the shaded circle) increases, the route will automatically divert by selecting link from 4 to 2 (as $G_{1,6,4\to 2} = 0.7$ in comparison to $G_{1,6,4\to 5} = 0.65$). Based on this process, automated route discovery can be performed in a distributed fashion, and at the same time support efficient load balancing mechanism [14]. We will first describe the parameters involved in the route gradient calculation, and this will be followed by the route discovery and routing algorithm.

A. Parameter definition

The gradient search performed during the route discovery is based on a hop-by-hop technique, which is by evaluating the gradient value of the different links in each node and selecting the link with the highest gradient value. The difference between PGBR and the potential based routing proposed in [17], is the components in the gradient equation. In the case of PGBR, we also include the link load value in order to provide more fine grain flexibility when the route discovery process is trapped in congested regions of the networks. The gradient is calculated for each link on a node (e.g. a node with 5 connecting links, will have 5 gradient values).

The gradient calculation for destination *d* of node *n* for link connecting *n* to node *j* is represented in equation 1 as,

$$G_{n,d,n\to j} = \alpha\Phi_j + \beta l_{n,n\to j} + \gamma h_{j,d} \qquad (1)$$

where Φ represents the load of neighbouring node *j*, $l_{n,n\to j}$ represents the link spare capacity between node *n* and *j*, and h_j represents the normalised hop count of each node to destination *d*. The hop count value is used to determine the relative distance of each node with respect to a specific destination. Therefore, each node contains a table with the different hop count values corresponding to each destination. Initially, a hop count value (un-normalized) is transmitted from each respective destination, and the messaging process is only initiated when the topology is formed. As the hop count messages travels from node to node, the message reduces the hop count and the value is stored in each node. In the event that a number of hop count value arrives at the node, the highest value will be selected. The hop count value H is transmitted to each node starting from a set value at the destination H_s. Immediately following the stabilisation of hop count message flooding, and each node has retained the highest hop count value, normalization of the hop count is performed. This is performed through the back propagation process where the source node that receives the final hop count H_f, will transmit this value back in the opposite direction towards the destination. Each node that receives the H_f value, will normalise their own hop count, which is represented in equation 2,

$$h_{j(s,d)} = \frac{((H_s - H_f)+1)-(H_s - H_j)}{(H_s - H_f)+1} \qquad (2)$$

The first component of the gradient equation (1), is the load information (Φ) of the neighbours node. The load information is periodically transmitted from each node to the immediate neighbours. This mechanism is inspired from reaction-diffusion mechanism that creates self-organisation mechanism of cells through cell-cell interactions [15]. The load node of the neighbour node is represented in equation 3,

$$\Phi_n = \frac{\sum_{i}^{I}(l_{n,c,n\to j} - l_{n,n\to j})}{\sum_{i}^{I} l_{n,c,n\to j}} \qquad (3)$$

Equation 3 calculates the ratio of the spare capacity to the full capacity of all the immediate links of the node. Each n node contains i number of link $l_{n,n\to j}$, where the capacity of each link i of node n is represented as $l_{n,c,n\to j}$. The second component of the gradient equation 1 is the spare capacity of the immediate link to the node.

Based on the three components of the gradient equation, each node will calculate the gradient value for each respective destination. As each route discovery packet migrates from node to node, the node will evaluate the packet destination address, and select the highest neighbouring gradient value to forward the packet. In effect, this process creates a gradient field that dynamically changes with respect to the load changes within the network.

B. Routing algorithm

Our routing algorithm is illustrated in Fig. 2.

```
1for all n,m∈ N
2  destination diffuse H_s to all nodes
3    each n deducts H_s and transmits to m
4      if source receives H_s
5        back-propagate final H_f value
6        calculate normalized h_n,d value
7for node n
8  if detect current load > Threshold
9      calculate
10     emit   to neighbours
11for node n
12  if receive   from neighbour m
13     calculate G_n for all links
14  if link load change
15     calculate G_n for all links
16for discovery packet p_d,(S • D)
17  n receives p_d
18  n determines address d for p_d
19  n selects highest G to forward p_d
20  p_d records n as FN
21if p_d arrives back to n (n∈ FN)
22  p_d backtrack one hop to node e
23  e selects second G to forward p_d
24  if p_d > PL_T
25     route discovery is null
26if p_d is successful
27     route stream
```

Figure 2. Algorithm for de-centralise routing

Lines 1 – 6 of the algorithm describes the propagation of the hop count and calculation of the normalized hop count value when the topology is formed. Lines 7 – 10 describes the mechanism of transmitting the node load information to the neighbours based on a comparison to a threshold value, where lines 11 - 15 demonstrates how the gradient is then calculated on node n from this value for each link. Line 16 – 20 illustrates the mechanism of route discovery from a specific source to destination ($S \to D$). When node n receives the discovery packet p_d from its neighbour (line 17), n evaluates the address of packet p_d and selects the link with the highest gradient $G_{n,n\to j}$ to forward the packet (in this case node j). As described earlier, the packet transmission will be performed hop-by-hop, until p_d reaches the destination D. Once p_d has reached the destination, the packet will return to the source, where the streaming begins (line 26 - 27). However, as Iwata and Fujita [7] have evaluated, the hop-by-hop solution for packet transmission can lead to unnecessary loops (e.g. packet comes back to a node that was along the path). Our algorithm, however, considers this drawback and incorporates mechanism to discover alternate paths in such event (line 21 – 23). This process is illustrated in Fig. 3.

As the p_d migrates from node to node during the discovery process, the packet records all the traversed nodes and stores this in a list known as the Forbidden Node (*FN*) list. In the event that the discovery packet approaches a node in *FN*, the discovery packet will backtrack one hop and evaluate the next highest gradient. In the event that this leads to another loop, the p_d will back track two hops and select the second highest gradient link. However, this process has a limitation, where the route discovery will terminate if the number of nodes along a path is higher than the threshold PL_T(line 24 – 25). An example of backtracking is illustrated in Fig. 3, where initially the route traverses through nodes 1 – 4 – 2 – 1, and discovers node 1 in *FN*. Therefore, the p_d backtracks by one hop and select the next highest gradient, which is link between node 2 and 6. The final path is, therefore, 1 – 4 – 2 – 6 – 7. The algorithm was extended from the our previous distributed routing algorithms [18], where we didn't incorporated route discovery, loop elimination process or backtracking. Since our previous algorithm did not incorporate these functionalities, the packet streams would get caught in loops leading to performance degradation. However, through the packet discovery process, this gives greater flexibility for manipulating and modifying paths before the streaming is performed.

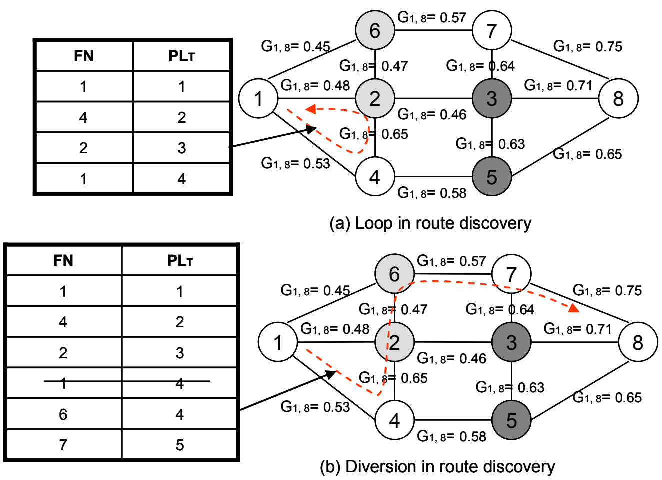


Figure 3. Loop elimination routing

C. Parameters re-configuration

As described earlier, an advantage of the PGBR routing algorithm is the ability for us to reconfigure the parameters to suit the management requirements of the network. An example of different routing paths resulting from different parameter configuration is illustrated in Fig. 4. Fig. 4(a) shows the configuration of PGBR: α = 0.2, β = 0.2, γ = 0.6, which led to a route that is shortest path but traverses through slightly congested parts of the network. However, in the case of Fig. 4 (b), for configuration of PGBR: α = 0.2, β = 0.4, γ = 0.4, the route diverts to lesser congested parts of the network. The reason that PGBR: α = 0.2, β = 0.2, γ = 0.6 took the shortest path route is because the highest weight was set for γ, which represents the hop count, while for PGBR: α = 0.2, β = 0.4, γ = 0.4, equal weight was given to shortest path as well as link load. Therefore, the route discovery tended to move towards the lighter parts of the network with higher link capacities.

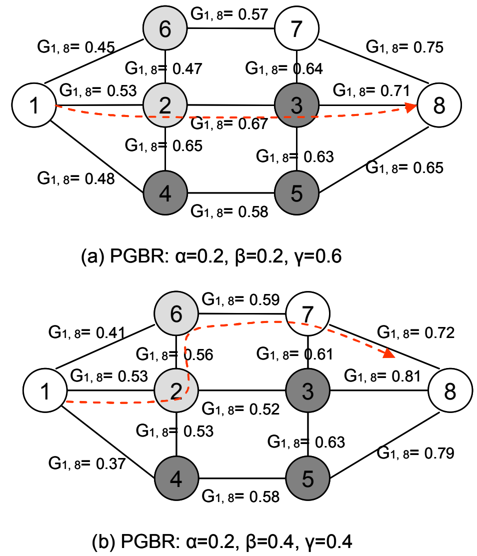


Figure 4. PGBR Routing with different parameters

By discovering different routes for different parameters, the PGBR routing algorithm has the potential of discovering routes based on service QoS requirements. Therefore, the paths for different streams can be set to different regions of the network depending on the congestion and loads of the network as well as service QoS requirements (this will be better elaborated in Section V, when we describe the role of PGBR from the perspective of Future Internet).

IV. SIMULATION

We have performed simulation work to evaluate the performance evaluation of the PGBR routing technique. The simulation work is separated into two sections, which includes Part A and B. For both parts, the parameters for the simulation are presented in Table I and II. The α, β, and γ parameters are selected for different simulation cases (for Part A we have a fixed set, while for Part B we have 3 different sets).

TABLE I. TOPOLOGY PARAMETERS

No. of nodes	**Connectivity**	**Av. Source – destination no.**	**Av. link capacity**
50	0.036	276	100 Mbps
100	0.0198	595	100 Mbps
150	0.0132	703	100 Mbps
200	0.010	1081	100 Mbps

TABLE II. TRAFFIC TYPE PARAMETERS

Traffic Type	**Distribution for Service time (average - seconds)**	**Distribution for Flow Quantity (average - Kbps)**
HTTP	Uniform (2-8)	Uniform (20 - 60)
VoIP	Uniform (60-180)	Uniform (54-74)
VoD	Uniform (200-400)	Uniform (200-400)

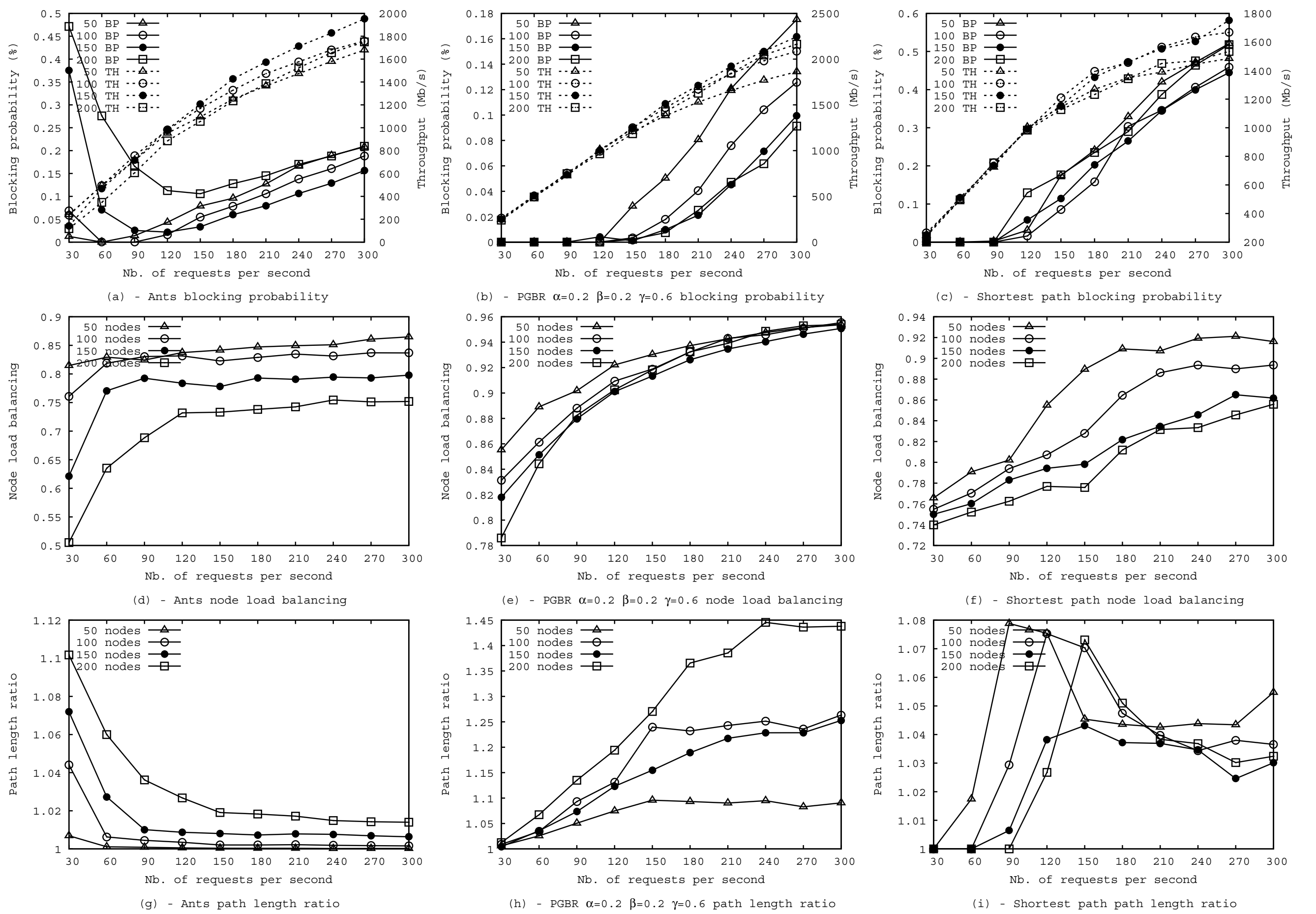


Figure 5. Performance Evaluation between ANTS, Shortest Path and PGBR

Table I presents the four topologies used in our simulation, including the connectivity, average number of source and destination pairs and average link capacities for each topology. Table II illustrates the types of traffic used in the simulation, including their service time and flow quantity. Our simulation is compared with shortest path (SP) algorithm as well as the ANTS distributed routing algorithm. The ANTS algorithm uses swarm intelligence to find the best route, and is also based on gradient based routing where the gradients are set through pheromone trails. The parameters used for the ANTS algorithm is shown in Table III.

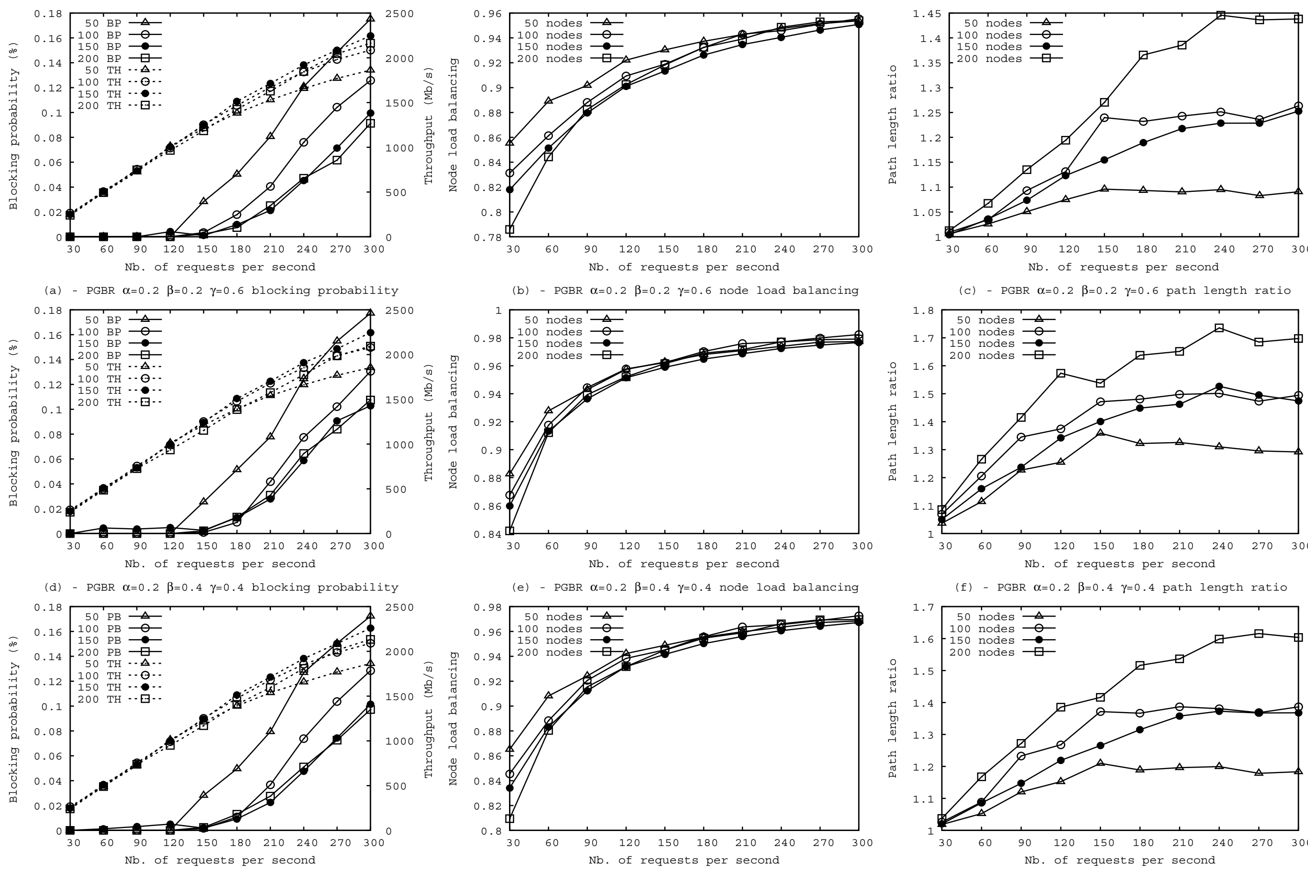
TABLE III. PARAMETERS FOR ANTS ALGORITHM

Parameters	Value
Interval creation ants	100μs
Source Selection	Random (uniform)
Destination Selection	Random (uniform)
Maximum age of the ant	30 hops

We use an ANTS version which modifies the pheromone table using age and delay based on link load. This solution tries to find the less loaded paths which is the same approach to the PGBR solution. The modification of the pheromone table and the delaying technique are based on a specific ANTS algorithm [19].

Part A

The initial comparison between PGBR, ANTS and SP is based on $\alpha = 0.2$, $\beta = 0.2$, and $\gamma = 0.6$. Figure 5 (a), (b), and (c) presents the blocking probability as well as throughput comparison between ANTS, PGBR, and SP. As shown in Fig 5(c), the worse performance is demonstrated by the SP algorithm with the highest blocking probability for all topology sizes (the changes between each topology size is not too different). The comparison between the ANTS and PGBR, shows that PGBR has a greater variation between 50 and 200 nodes. For all topology sizes, the PGBR outperforms the ANTS algorithm for blocking probability (in particular this is evident at high load of 300 requests/second - 50 nodes: PGBR - 0.175, ANTS – 0.22; 100 nodes: PGBR – 0.13, ANTS – 0.18; 150 nodes: PGBR – 0.1, ANTS – 0.15; 200 nodes: PGBR – 0.12, ANTS - 0.22). This is also reflected in the throughput, where there is larger number of flows admitted (e.g. at 300 requests/s for 200 nodes, PGBR is able to accept nearly 2.3Mbps of flows in comparison to 2.05Mbps of flows for ANTS). The other main disadvantage for ANTS is the amount of time required for convergence. As shown in Fig. 5 (a), the initial blocking probability for all



(a) - PGBR α=0.2 β=0.2 γ=0.6 blocking probability
(b) - PGBR α=0.2 β=0.2 γ=0.6 node load balancing
(c) - PGBR α=0.2 β=0.2 γ=0.6 path length ratio
(d) - PGBR α=0.2 β=0.4 γ=0.4 blocking probability
(e) - PGBR α=0.2 β=0.4 γ=0.4 node load balancing
(f) - PGBR α=0.2 β=0.4 γ=0.4 path length ratio
(g) - PGBR α=0.2 β=0.3 γ=0.5 blocking probability
(h) - PGBR α=0.2 β=0.3 γ=0.5 node load balancing
(i) - PGBR α=0.2 β=0.3 γ=0.5 path length ratio

Figure 6. Performance Evaluation of PGBR with different parameters

topologies is very high before it reduces and converges (in particular for 200 nodes, the blocking probability is highest). The reason that PGBR is able to perform better blocking probability is due to the reason that PGBR is able to better load balance the network, and therefore accept more streams. Therefore, as the topology gets larger, the PGBR is able to have a larger variation and space of movement during route discovery (hence the reason for the large variation between the different topology sizes). For small topology size, the PGBR takes into consideration the link load, and is able to discover routes in congested parts of the network, which increases the chance of route discovery. As shown in the results as the topology gets larger, the difference between the blocking probabilities gets larger. Fig. 5 (d), (e) and (f) compares the load balancing tests between the different algorithms. The computation for the load balancing is based on determining the ratio between the average of loads in each node against the average load of the network (therefore, for optimum load balancing, the value should be close to 1). As shown in Fig. 5 (f) there is greater variation in load balancing for SP, where the best performance is for 50 node topology and has the closest performance to PGBR for 50 nodes (0.89 to 0.955). However, the largest variation is found in both SP and ANTS as the topology size increases. In comparison to PGBR, the load balancing is far more effective and balances the network irrespective of the topology size, where there are very little variations. The path length ratio is calculated by determining the ratio of the discovered path over the shortest hop count path length (for unloaded network) between the source and destination. As shown in Fig 5 (g) (h) and (i) the path length ratio for PGBR performs the worst (for number of requests higher than 150 requests/s, the path length ratio is nearly 1.5 times for 200 node topology). Therefore, the performance worsens as the topology size increases. The reason for this is because of the load balancing effect that PGBR has over the other algorithms, leading to longer routes. However, this results in admittance of higher number of flows in comparison to the other algorithms. The ANTS algorithm has higher path lengths at the beginning due to the route discovery process, before the algorithm converges to stable path length. Based on this result, it is clear that ANTS will not react well when the traffic pattern changes, where a considerable time is required for path discovery convergence which leads to higher blocking probability.

Part B

The simulation tests for this section will compare the different parameters of γ, β, and α for the PGBR algorithm. The three variations includes: α = 0.2, β = 0.2, γ = 0.6; α = 0.2, β = 0.4, γ = 0.4; and α = 0.2, β = 0.3, γ = 0.5.

For the 50 and 100 node topology, the overall performance of blocking probability was very similar for different loads for different parameter configurations. This means that flexibility is available for route discovery in the event that shortest path (e.g. γ is given priority) is required, to allow short diversions around congested parts of the topology. As the load increases, the routes will slowly increase in size. However, once the topology increased in size (150 and 200 nodes), the configurations of $\alpha = 0.2$, $\beta = 0.2$, $\gamma = 0.6$ gave the best blocking probability performance and this was followed closely by the performance of $\alpha = 0.2$, $\beta = 0.3$, $\gamma = 0.5$. The main reason behind this result is because larger topologies will lead to higher diversity of route discovery. Therefore, with the same traffic load for a larger topology, the best configuration is by prioritising the path lengths.

For the network load balancing tests, all the topologies exhibited similar behaviour for each configuration. The difference between different topologies was dependent on the parameter configuration. As the γ value decreased, we can see a more balanced network (e.g. γ and β given equal priority). The load balancing is slightly less in the case of PGBR: $\alpha = 0.2$, $\beta = 0.2$, $\gamma = 0.6$ and $\alpha = 0.2$, $\beta = 0.3$, $\gamma = 0.5$, since priority is given to shortest path leading to the topology being largely congested in certain parts. The changes are also reflected in the average path lengths, where at $\alpha = 0.2$, $\beta = 0.2$, $\gamma = 0.6$ we note a shortest path length average for all loads (the smaller the topology the smaller the average path length). A slightly longer path length can be seen once the γ priority is reduced ($\alpha = 0.2$, $\beta = 0.3$, $\gamma = 0.5$), where the longest path length is found for equal priority of hop count and link load – $\alpha = 0.2$, $\beta = 0.4$, $\gamma = 0.4$.

V. PGBR AND FUTURE INTERNET

In this section, we will describe how the Future Internet will evolve, and how PGBR can play a contributing role. Fig. 7 illustrates an example of how the Future Internet will evolve towards a service based architecture. This will be realized through a variety of services that are created to meet end user's needs. These services will encompass various types of applications on various different devices (e.g. ranging from live TV on mobile device to HDTV streaming to set top boxes in homes). While the diversity of services are increased, a big challenge lies in the ability to efficiently delivering these services through the communication networks. The diversity of services will also lead to very dynamic traffic behaviour, which is not suitable with current routing approaches that will require frequent global recalculation each time the traffic demand changes (changing traffic pattern may be due to user's changing services, each leading to different demand traffic). Therefore, dynamic and adaptive routing will be required at the underlying infrastructure networks.

This is where the PGBR routing technique can add major benefits, where the routing can be performed irrespective of the traffic demand. At the same time, the flexible control of parameters can allow routes to be discovered with respect to different application QoS requirements. This was described earlier, and we saw the effects of path discovered with different parameter combinations (for example for standard definition VoD, we can have shortest path (high γ)), while for VoIP we can have a slightly lower γ, and for data we can have low γ and β value and high α value. The high α value will lead to increased load balancing priority, which will allow the data streams to use the outer parts of the network while the center parts are more focused on multimedia streaming. Although in this paper we have projected static configuration of the parameters for all streams, the parameter configuration can also be adaptively change when the traffic demand for specific traffic type changes (e.g. HDTV traffic increases and provides higher revenue).

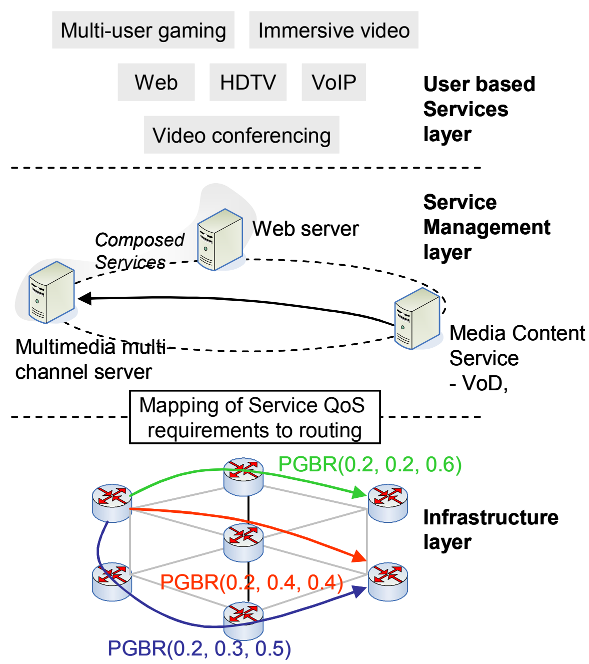


Figure 7. Proposed architecture

VI. CONCLUSION

In this paper, we have proposed the PGBR routing algorithm, which is a distributed gradient based routing protocol. The PGBR routing technique discovers the route by traversing through the network hop-by-hop till it reaches the destination. The chemical gradient field produced in the network is based on local node to node interaction, which dynamically changes as the load in the network changes. Based on the load in the network, the route discovery will avoid parts of the network that is highly congested and route through lightly congested parts of the network, leading to a well balanced network. At the same time, the PGBR routing algorithm is also very suitable for Future Internet architectures that will require highly dynamic routing mechanism to support diverse service types as well as dynamic traffic patterns. The flexibility in the parameter selection, can also allow each route to be discovered with respect to specific service QoS requirements.

Simulation works have also been presented to illustrate the benefits of the PGBR algorithm over the ANTS distributed routing algorithm as well as the shortest path algorithm. The simulations have shown how the PGBR outperforms other routing techniques for four different topology sizes and different network load. Simulation work has also been conducted on different parameters of the PGBR routing algorithm and how this affects different topology size and load in the network.

ACKNOWLEDGMENT

This work has received support from the Higher Education Authority in Ireland under the PRTLI Cycle 4 programme, in the project *Serving Society: Management of Future Communications Networks and Services*.

Special thanks to Dave Malone from the Hamilton Institute, National University of Ireland, Maynooth, for his valuable feedback.

REFERENCES

[1] Y. Wang, E. Keller, B. Biskeborn, J. Van der Merwe, J. Rexford, “Virtual Routers on the Move: Live Router Migration as a Network-Management Primitive”, in Proceedings of ACM SIGCOMM 2008, WA, USA, August 2008.

[2] K. Xi, H. J. Chao, “IP Fast Rerouting for Single-Link/Node Failure Recovery”, in Proceedings of 4th International Conference on Broadband Communications, Networks, and Systems (IEEE BROADNETS 2007), Raleigh, North Carolina, USA, September 2007.

[3] K. Leibnitz, N. Wakamiya, M. Murata, “Biologically inspired Self-Adaptive Multi-path routing in overlay networks”, Communications of the ACM, vol. 49, no.3, March 2006.

[4] Y-W. Chen, R-H. Hwang, “QoS Routing Algorithms for Multiple Traffic Classes”, IEEE International Conference on Communications (ICC), 2002.

[5] G. Apostolopoulos et al, “Implementation and performance measurements of QoS routing extensions to OSPF”, in Proceedings of Infocomm 99, April 1999.

[6] S. Nelakuditi et al, “Quality-of-Service routing withouth global information exchange”, in Proceedings of IEEE IWQoS Workshop 99, March 1999.

[7] A. Iwata, N. Fujita, “A Hierarchical Multilayer QoS Routing System with Dynamic SLA Management”, IEEE Journal on Selected Areas in Communication, vol. 18, no. 12, December 2000.

[8] J. Moy, “OSPF: Anatomy of an Internet Routing Protocol. Addison-Wesley, MA, 1998.

[9] S. Chen, K. Nahrstedt, “An overview of Quality-of-Service routing for the next generation high speed networks: problems and solutions”, IEEE Network, 12(6), 1998.

[10] J. Moy, “OSPF version 2”, RFC 2328, April 1998.

[11] B. Fortz, N. Thorup, “Internet traffic engineering by optimizing OSPF weights”, in Proceedings of IEEE INFOCOM, 2000.

[12] B. Fortz, M. Thorup, “Optimizing OSPF/IS-IS Weights in a Changing World”, IEEE Journal on Selected Areas in Communications, 20 (4), 2002.

[13] D. Applegate, E. Cohen, “Making Routing Robust to Changing Traffic Demands: Algorithms and Evaluation”, IEEE/ACM Transactions on Networking, Vol. 14, no. 6, December 2006.

[14] K. Gopalan, T. Chiueh, Y.-J. Lin, “Load Balancing Routing with Bandwidth-Delay Guarantees”, IEEE Communications Magazine, June 2004.

[15] A. M. Turing, “The chemical basis of morphogenesis”, Philos. Trans. R. Society. London, 1952, 237. pp. 37-72

[16] J. Rexford, “Rethinking Internet Routing”, in Proceedings of Symposium on Theory of Computing Conference (STOC’08), Victoria, Canada, May 2008.

[17] A. Basu, A. Lin, S. Ramanathan, „Routing using Potentials: A Dynamic Traffic-aware Routing algorithms“, in Proceedings of SIGCOMM 2003, Karlsruhe, Germany, August 2003.

[18] S. Balasubramaniam, D. Botvich, W. Donnelly, M. Ó Foghlú, J. Strassner, “Bio-inspired Framework for Autonomic Communication Systems”, (Editors Falko Dressler and Iacopo Carreras) in “Advances in Biologically Inspired Information Systems: Models, Methods, and Tools”, Studies in Computational Intelligence, Springer Verlag, 2007.

[19] R. Schoonderwoerd, J. L. Bruten, O. E. Holland, L. J. M. Rothkrantz, “Ant-based load balancing in telecommunications networks”, Adaptive Behaviour, vol. 5 (2), 1996, pp. 169 – 207.