Bandwidth as a Metric for Candidate Path Selection: An Approach to Localized Routing Algorithms in Communication Networks

Turki A. Alghamdi
Department of Computing, School of Computing, Informatics and Media
University of Bradford
Bradford, UK
t.a.alghamdi@bradford.ac.uk

M. E. Woodward
Department of Computing, School of Computing, Informatics and Media
University of Bradford
Bradford, UK
m.e.woodward@bradford.ac.uk

Abstract—Most of the Quality of Service routing (QoS) routing mechanisms involve periodic exchange of global state information which causes communication overheads. Localized routing is a method to avoid this problem. The network in this technique is inferred by the source nodes using statistics which are collected locally. Three new localized routing algorithms called Highest Minimum Bandwidth routing (HMB), Best Reserved Bandwidth routing (BRB), and Highest Link Average Bandwidth History (HLABH) are introduced in this paper. The new algorithms are compared under different traffic loads and network topologies to the existing localized Credit Based Routing (CBR) and the global WSP routing algorithm, our algorithms show better performance without undue increase in complexity. The selection of disjoint paths and recalculation of the set of candidate paths were used in the presented algorithms. This resulted in a positive effect of dynamic path selection method on the performance of localized routing algorithms.

Keywords- Algorithms; Localized Routing; Quality of Service

I. INTRODUCTION

Connectivity is one of the key priorities of routing in the existing internet, which relies on the “best-effort” principle. Most routing protocols work with a single metric, such as hop count or administrative weight. OSPF [1] is a well known example of such protocols and uses the shortest path routing paradigm. The main drawback of using such protocols is a lack of resources which can not satisfy the requirements of IP telephony or video-on-demand applications. Unbalancing in the network is regarded as a major problem of using the shortest path for best effort service. This happens due to the frequent use of the links assigned to the shortest path and retains most of the network loads. Thus, Quality-of-Service (QoS) routing [2, 3] is proposed as an essential mechanism. The procedure for selecting the paths in QoS routing is based on the availability of the resources which can satisfy the QoS requirements. Thus, the QoS state is the key factor in the QoS routing as the path will be selected if it has sufficient resources to satisfy the QoS requirements. Therefore, the disadvantage of unbalancing the network, which appears when using the shortest paths in the best effort mechanism, will be reduced by using the QoS routing due to the knowledge of QoS state.

Periodic exchanging of global QoS state information and keeping it in a database of network routers is one of the problems in the global state routing. Several routing schemes [3-6] that have been introduced in the QoS routing need this exchange of the link state information among the routers in the whole network. This difficulty leads to flapping of routers and high communication overheads. Localized QoS routing is considered an alternative method to global state routing which provides better performance and reduces the above mentioned problems. Flow blocking statistics, that are collected locally and provided to the network QoS state by the source nodes, minimize the communication overhead and eliminate the need for the routers to update and keep a database of QoS state[7].

Candidate path selection plays a part in localized QoS routing. Various methods of selecting the preferred path have been proposed in [8, 9]. A set of candidate paths between sources and destinations are chosen based on different schemes, e. g. a well known algorithm of the shortest path Dijkstra’s, and the flow is routed along these paths. Effective methods have been applied in our work to improve the selection of candidate paths. Highest Minimum Bandwidth routing (HMB), Best Reserved Bandwidth routing (BRB) and Highest Link Average Bandwidth History (HLABH) are new algorithms that are also presented in this paper. To further illustrate the effectiveness of our proposed algorithms, we conduct some simulation experiments.

The simulation results demonstrate that our proposed algorithms perform well and in a number of cases actually produce significantly better results, even when prior approaches fail to do so. In a previous study, the global state algorithm Widest-Shortest-Path (WSP) [6] and the Proportional Sticky Routing (PSR) localized algorithm [7] were compared to the Credit Based Routing (CBR) algorithm [10]. The latter performed better than WSP and PSR. For this reason, our proposed algorithms are compared to CBR only.
The reminder of the paper is organized as follows. The next section presents the most recent localized algorithms and summarizes the problems associated with these algorithms. In section 3, we provide the objectives of the proposed algorithms and describe the functionality of these mechanisms. Section 4 introduces the environment and characteristics of the simulation and briefly reviews the description of the network topologies used in our experiments. Results of our simulation are shown and discussed in section 5. Finally, section 6 draws the conclusions and lists possible future work.

II. RELATED WORK

A considerable amount of literature has been published on QoS routing [2, 11]. The concept of processing local information has been used in many different mechanisms of telephone technology and was applied long before its implementation in computer networks [12]. Both PSR [7] and CBR [10] represented the idea of localized routing in computer networks and they are considered as relevant work to our algorithms.

The superior performance of CBR compared to PSR has been shown [7, 10, 13]. For this reason we used CBR as a standard to compare our mechanisms, whereas WSP was a relevant global algorithm.

A. Proportional Sticky Routing

Nelakuditi presented the first Localized Proportional Sticky Routing algorithm (PSR) [7]. Presumably, the main principle of the PSR mechanism suggests that the route-level statistics, i.e. number of flows blocked, is the only QoS state data obtained by the source. The existing statistics are the basis for the algorithm to proportionally allocate the load from a source to a destination amongst numerous paths according to their flow blocking probability. Therefore, the PSR algorithm needs every node to sustain a predefined set of candidate paths \( R \) to each destination. Supposedly, routing algorithms structured to choose the short path show better performance over the algorithms that are not burdened with the path length [5, 14].

Two types of paths are differentiated by PSR. These are minhop paths \( R_{\text{min}} \) (the shorter paths) and alternative paths (paths with longer length) \( R_{\text{alt}} \), where \( R = R_{\text{min}} \cup R_{\text{alt}} \). Thus, the algorithm selects minhop paths and consequently decreases the so-called ‘knock-on’ cascade effect arising from alternative path utilisation by some sources. It involves other sources, where minhop paths share links with alternative paths, and prompts them to use these alternative paths as their minhop paths [7, 12].

The PSR algorithm encompasses two steps: proportional flow routing and calculation of flow proportions. Proportional flow routing is a cyclic process, where each cycle is of a certain length and presented by a selection of the set of eligible paths \( R_{\text{elg}} \), followed by directing the flows along the preferred paths. A prearranged proportion \( \alpha \), serves to choose a path \( r \) with programmed frequency.

At the start, all the candidate paths are eligible and assigned a maximum permissible flow blocking parameter \( \gamma_r \). It verifies the permitted number of blocked flows directed along the path before it is converted designated as ineligible. For each minhop path, \( \gamma_r \) is set to \( y \), a configurable system parameter. For each alternative path, the value of \( \gamma_r \) is dynamically set between \( 1 \) and \( y \). A new cycle is reset with \( R_{\text{elg}} = R \), and when the old one is finished \( R_{\text{elg}} \) becomes empty. An observation period includes the number of cycles \( n \), and is terminated by a new flow proportion for each path \( r \in R \), \( \alpha_r \), which is calculated based on its observed blocking probability \( b_r \).

After the completion of each observation period, the PSR mechanism adjusts the minhop paths flow proportions to equalise with their blocking probability \((\alpha_r, b_r)\). For alternative paths, flow proportions are regulated by the minimum blocking probability among the minhop paths, \( b^* \), i.e., for each \( \gamma \in \mathbb{R}^n \), if \( b_r < \Psi b^* \), \( \gamma_r = \min \left( \gamma_r + 1, \gamma \right) \). If \( b_r > b^* \), \( \gamma_r = \max \left( \gamma_r - 1, \gamma \right) \), where \( \Psi \) is a configurable factor to minimise the ‘knock-on’ phenomenon under system overloads.

Available sources dispute the benefit of the PSR algorithm. Even though the application of PSR improves routing, this particular technique employs Erlang’s loss equation for calculating flow proportions, based on the steady-state blocking probability distribution [10]. It leads to certain complications in case of non-Poisson or bursty traffic. Besides, as the number of the candidate paths decrements with each cycle, it becomes necessary to normalise flow proportions calculated at the beginning of the cycle as soon as a path becomes ineligible. Due to the fact that flow distribution may not recall the prearranged proportion, the path which last happens to be ineligible is inclined to receive a higher number of flows than others.

B. Credit Based Routing

The CBR is the most relevant work to our algorithms. Its superior performance compared to PSR has already been shown in available sources [7, 10, 13]. For this reason we will use CBR as a standard to compare our mechanisms. This algorithm works according to the following scenario.

Each source node needs to set a number of candidate path sets \( R \), based on minimum hop path \( R_{\text{min}} \) and an alternative path \( R_{\text{alt}} \) which refers to the paths with minimum hop and minimum hop plus one respectively, where \( R = R_{\text{min}} \cup R_{\text{alt}} \). CBR assigns a maximum credit for each path \( P.credits = max.credits \), and selects the path with the highest credit from both sets to direct the flow. The crediting scheme works as follows. A test message is sent along the chosen path and each node in this link behaves as a router to test the outgoing link, if it has sufficient residual bandwidth for the flow. If the residual bandwidth is not adequate to satisfy the QoS, a failure message is sent back informing the source about the path failure and the path credit is decremented by its blocking probability \((1)\). In the successful case the bandwidth of the message will be reserved from the residual
bandwidth. Then message is forwarded till it reaches its destination and the path credit is increased by (1 - the path blocking probability). The CBR algorithm uses both $\Phi$ and $\max_credits$ as system parameters, where $\Phi$ controls the usage of alternative paths and where $\max_credits$ verifies the maximum credit for each path.

$$P.credits = P.credits - P.BlockingProbability$$  \hspace{1cm} (1)

CBR routes the flow based on a crediting scheme that rewards a successful path and penalizes the failed ones. Blocking probability statistics are calculated in CBR within predefined period (20 flows). The credit is given for each path based on the average number of rewarded and penalized points during these periods.

In spite of CBR performing at a more enhanced level than PSR, the metric employed for path selection relies on a crediting scheme and therefore does not show the quality of the path. It has to be estimated on the basis of the QoS characteristic, e.g. bandwidth or delay. In addition, it is not a logical approach to use manipulation by blocking probability as a gradient variable.

III. THE PROPOSED ALGORITHMS

This paper presents new localized routing algorithms and the following section illustrates the concept of each algorithm and identifies the key determinants of each algorithm to choose the optimum path from the candidate path set.

A. Hypothesis

The new algorithms are source routing algorithms, where the source node takes the routing decision. Multi-Protocol Label Switch (MPLS) [15] is an example of a protocol used to set up one or multiple paths between each pair of source and destination nodes. Our QoS algorithms assume that the network is enhanced by signalling and resource reservation mechanisms to make the path for the new flow. Flows are routed along the explicit routed paths, which will be referred to as candidate paths in the following text. This process starts at the source node by sending a setup message along the selected path. Each node in a path acts as a router to test if the following link has sufficient residual bandwidth for the flow. In the event that the residual bandwidth is not sufficient and does not satisfy the QoS requirements, a failure message is sent back to the source informing of the failure of this path. In the successful case, the bandwidth of the message will be reserved from the residual bandwidth and message will be forwarded till it reaches its destination. CBR and other localized routing PSR use the statistic of blocking probability as a factor in selecting the routing paths. Apart from their principles of function, our proposed algorithms use residual bandwidth as the direct QoS guideline to select routing paths. Description of the methods of how the algorithm chooses the set of candidate paths and the mechanisms of how each algorithm works, are shown in next section.

Figure 1. The pseudo code for the HLABH algorithm.

B. Description

The mechanisms presented in this paper differ from PSR and CBR, whose routing decisions are motivated by flow statistics of path blocking probability. Our algorithms utilise the residual bandwidth as the main metric in order to direct the flows. In HLABH, each link in the path is given an interval (sliding window) in order to compute the average residual bandwidth for that particular link. The best choice path is assigned on the basis of the window results. The path which generates the highest link average residual bandwidth becomes the candidate path to direct the arriving flow. HLABH performs the screening of residual bandwidth across the network and constantly maintains each link’s average residual bandwidth. By taking into account the average residual bandwidth for each link it indicates the viability of the path. The pseudo code for the HLABH algorithm is presented in Fig. (1).

The principle of localized QoS routing mechanisms is that each pair of source and destinations needs a pre-established set of candidate paths $R$. In HLABH the average residual bandwidth link is the key determinant which is assigned to every path $P$ in the candidate path set. A comparison is performed by the setup or the test message over the path links to obtain the lowest residual bandwidth link history for that path (line 6). These selected links are compared with each other in order to obtain the path that has the highest average bandwidth link history (lines 7). The flow is routed along the selected path (line 9).

The HMB algorithm selects the highest minimum residual bandwidth among the candidate path set. The HMB scenario starts when a setup message travels from source to destination along the outgoing links in the path. All the outgoing links in the single path are compared to locate the link with the minimum residual bandwidth. Each selected link refers to a path in the candidate path set. The HMB algorithm selects the best path by selecting the link that has the largest residual bandwidth among the selected links with the minimum residual bandwidth. Flows will be routed through the selected path. The pseudo code for this algorithm is shown in Fig. (2).

In the HMB algorithm a predefined set of candidate paths $R$ are required by each source and destination pair. A variable $\text{P.MinResidualBW}$ stores the minimum residual
bandwidth link in each path $P$ (line 1). Every path $P$ is associated with this variable. First, path ($P_0$) in the candidate path set is assumed to have the minimum bandwidth link (line 2). The comparison (in lines 3-4) selects the highest minimum bandwidth between all the selected links in the candidate paths and flows are routed along the selected path (line 5). Importantly, preference has to be given to the link with the minimum residual bandwidth among the selected paths. It ensures that the selected path can encompass the flow.

The final technique is the BRB algorithm, which is quite similar to the HMB concept. The defining difference between them, however, is that in BRB the bandwidth is reserved for all the links in the path at an early stage and the comparison occurs on the basis of the residual bandwidth after the reservation. In other words, BRB reserves the whole path for the current flow and the next flow deals with the residual bandwidth after the reservation in any link in the path, whereas HMB reserves the bandwidth link by link. The BRB pseudo code is represented in Fig. (3).

In the BRB algorithm, all the links in the selected path are reserved for the current flow if the links satisfy the QoS requirements (line 3). If the link fails to satisfy the QoS requirements, bandwidth will be returned back to the previous links (line 5). Subsequently, the same steps of the HMB algorithm will take part at following stage. We assume that the first path in the candidate path set is selected as the one which has the minimum residual bandwidth link (line 10). The selected path will then be compared to the rest to obtain the path which has the highest minimum residual bandwidth link.

CBR routes the flow based on a crediting scheme that rewards a successful path and penalizes the failed. Furthermore, CBR credits the whole path as one block, whereas our three algorithms choose the beneficial path on the basis of the best link from compared ones. We use a different technique in all three algorithms: bandwidth and residual bandwidth in each link play a fundamental role.

IV. METHODOLOGY

In this section we describe the simulation characteristics and the network topologies used in our experiments.

A. Simulation Environment

Our Localized QoS routing algorithms were programmed to operate at flow level. Resource reservation, admission control, and selecting of the preferred path are based on the algorithms (HMB, HLABH, BRB and CBR), and simulated using the discrete-event simulator OMNeT++ [16].

B. Simulation Characteristics

There are some assumptions applied on our work to simplify the simulation. In all networks, all simulated links are assumed to be bidirectional and of the same capacity with $C$ units of bandwidth in each direction ($C = 150$ Mbps). The flow bandwidths are distributed uniformly within a range $[0.1, 2$ MB$]$, whereas, the arrival rate of flows reaches each source according to a Poisson process with rate $\lambda$. Nodes in the networks can be either sources or destinations. The flow service time is exponentially distributed with mean $1/\mu$. A random source node is chosen, and the destination node is selected randomly from the rest of the nodes. Following [5, 17], the offered load of the network $\rho = \lambda N bh / \mu C$, where $N$ is the number of nodes in the network, $b$ is the mean bandwidth per flow, $h$ is the average hop count per flow, $\lambda$ is the number of links in the network. Various ranges of loads are used in our simulation to evaluate the performance of the new algorithms. These values of loads chosen are based on the network characteristics, where some networks need to apply a heavy load to evaluate the algorithm performance, and this explains the reason for choosing different loads values in the represented networks. All paths between each source-destination pair having a maximum length, at most, of one hop more than the minimum number of hops, are chosen as the candidate paths [7]. In this paper we also introduce some methods of candidate path selection to improve the CBR. The same methods are applied on our offered schemes. Selection of the disjoint paths between each pair of source and destination shows progress in reducing the blocking probability. Recalculation is the second method which aims to determine the amount of blocking in each path between each pair of source and destination nodes in the network, and
subsequently modify the set of candidate paths by replacing the path of higher blocking by the lower blocking path. Although the recalculation method is applied, we keep the same number of candidate paths in the set. In our experiments, after the first 100,000 flows the recalculation method is applied and this reduces the blocking probability and produces better performance. The results were obtained from a simulation of 2,000,000 flow arrivals with the first 200,000 flow arrivals used as a run-up period to stabilize the algorithms.

C. Network Topologies

The underlying network topology might affect the functionality of the routing algorithms. Therefore, we simulate different topologies of both regular and random networks and run our three algorithms over these types of networks. In the selected random topologies the Doar-Leslie Model [18] plays the main part of identifying the probability of the node degree between any pair of source and destination. Waxman random graph scheme [19] has been modified by adding a scaling factor to produce the Doar-Leslie Model. The ISP topology is one of the regular backbone networks used in our study and has previously been used in various studies [4], [20]. Therefore, all networks were implemented in OMNet ++ combined with C++. Table 1 shows the characteristics of the topologies used.

D. Performance Metric

The performance is estimated by determining the overall flow blocking probability. It is calculated as the ratio of the number of flows blocked $B$ and the total number of flows that arrived at the network $T$ as in equation (2).

$$\text{Flow blocking probability} = \frac{|B|}{|T|}$$

V. Simulation Results

In this section we first describe the path selection methods followed by the evaluation of the performance of the HMB, HLABH and the BRB algorithms. We weigh them against the CBR scheme. The comparison method considers the QoS blocking probabilities, and the fact that the algorithm with lower blocking probability achieves better performance. Complexity also plays part in the performance evaluation.

A. Path Selection Methods

We introduce two methods of candidate path selection to improve our offered mechanisms. Selection of the disjoint paths between each pair of source and destination shows progress in reducing the blocking probability. Recalculation of the set of candidate paths is the other method which modifies the set of candidate paths by replacing the path of higher blocking by the lower blocking ones after calculating the amount of blocking in all possible paths between each pair of source and destination nodes in the network during predefined intervals. The recalculation method was still combined with the same number of candidate paths in the set. The selection of disjoint paths and recalculation of the set of candidate paths decrease the blocking probability in our new algorithms and the existing CBR algorithm. Although the use of disjoint paths and recalculation of the set of candidate paths dramatically decreased the flow blocking probability for all network topologies, the best improvement of using these methods observed in RAND80 network. The limitation in blocking probability possibly occurs, because a random network is likely to give a better chance of selecting disjoint paths in the candidate path set than the more restrictive topologies of the regular networks, such as ISP. Moreover, the RAND80 network has the highest average node degree compared to other random network in our simulation model (RAND45) that gives RAND80 more options to benefit from the disjoint path method.

Fig. 4 shows the impact of selection of disjoint paths and recalculation of the candidate path set on two of the presented algorithms (HLABH and BRB) and the existing localized algorithm CBR. Joining both methods decreases the flow blocking probability in HLABH, HMB and CBR algorithms. In other words, applying these methods in any localized algorithm leads to reduction in the flow blocking probability. HMB (not shown in Fig. 4) performs same as BRB when using both disjoint path selection method and recalculation of the candidate path set.

![Figure 4. Impact of disjoint paths and recalculation the candidate path set of different algorithms on RAND80 network.](image-url)
B. Blocking Probability

Fig. 5 represents the HMB, HLABH and BRB algorithms supplemented with the disjoint and recalculation methods. The figure evaluates the output of the three new algorithms against the current one (CBR) and the global algorithm (WSP) by calculating the flow blocking probability which is graphed versus a range of load states in several network topologies. The use of HMB notably decreased flow blocking probability in both regular and random topology networks. There is no significant difference between the performance of both HMB and BRB algorithms in all types of networks used in our experiments. Although HLABH achieved higher blocking probability of both HMB and BRB, it performed better than the CBR and WSP algorithms and gave low blocking in all networks. Comparing the overall performance of proposed algorithms showed their more successful function in all types of networks considered, with the HMB and BRB being the best of all.

Two factors usually affect any routing mechanism. These factors are (1) the main plan of the routing algorithm which is global or localized algorithm and (2) the path selection method of the algorithm. In case of the WSP algorithm, the path is chosen according to the updated global state of information. The performance of WSP deteriorates, if the updates do not keep up with the traffic fluctuations in the network. Thus, the WSP selection method always opts for the most suitable path proceeding from present global state and follows it, even if the path becomes exhausted till the latest update comes. Consequently, WSP performs as the worse scheme among all the algorithms in the selected topologies.

Our three algorithms and CBR share the approach of localized routing which does not need any network update but they differ in the path selection way. The preferred path for the CBR algorithm is the one with the maximum credits until it rejects the flows. The credits are decremented after every flow passed, which sanctions the update of the path crediting. Once a rejection has occurred, an alternative, higher credited path is chosen. The best of our three algorithms, HMB, selects the most feasible path based on the source view which gets the clear observation of the network. The main features of the new algorithms contributing to their performances estimated by the flow blocking probabilities and shown in Fig. 5. A similar pattern is followed in all network topologies. The WSP algorithm with the update interval 30 performs the worst because of its extended update interval in the global state. The CBR technique already acts superior to the WSP (30) due to the alternative path selection implemented in its principle of routing. Nonetheless, the flow blocking probability used to credit the alternative path in the CBR still is not as efficient as bandwidth metric chosen to qualify the path in our suggested algorithms. Thus, the graphs, illustrating the performance of our suggested algorithms, show superior results in all tested types of networks under the presented range of loads.

C. Impact of Network topology

Any routing algorithm performance is radically conditioned by the diverse types of network topologies. Therefore, there is a need to compare the performance of our proposed algorithms with both CBR and WSP under different network topologies. Fig. 5 shows the effect of the network types mentioned above in Table 1 on the
performance of the algorithms evaluated by the level of flow blocking probability. The figure shows that the superior performance of the algorithms is not significantly affected by the type of network topology.

D. Impact of Bursty Traffic

As presented in [8, 21] we now analyse the performance of proposed algorithms HLABH, HMB and CBR under bursty settings. The BRB algorithm data is not presented, as its performance coincides with HMB. The experiment is carried out using various flow lengths according to a Weibull distribution with shape parameters 0.4 and 0.7. The burstiness is increased by a small shape value. Fig. 6 shows the flow blocking probability versus the offered load with different shape values for the random topology network RAND80. The amplified burstiness in the arrival process causes an increased blocking probability over the range of loads used. Although there is no major impact of burstiness in the RAND80 topology, other sets of the experiments (data not presented here) demonstrate that burstiness has considerable effect on the performance of our algorithms and CBR in other network topologies.

Compared to the CBR, our algorithms showed superior performance with the prevalence of HMB. Lower shape parameters, especially 0.4, worsened only the CBR performance, presented by a notable dissociation of the 0.7 shape parameter graph. This can be explained by the CBR principle of routing decisions of blocking probability elevated with burstiness. Clearly, the HLABH and HMB algorithms outperformed CBR as well as each other, and HMB happened to be the best. Although the 0.4 shape parameter insignificantly worsened the performance of them, mostly in case of HLABH, their performance was not greatly affected by burstiness of traffic, since their routing decision is taken based on the bandwidth as path selection criterion.

E. Routing Scheme Overheads

Both path selection and collection of information are the major reasons leading to overhead in the network. Selecting of a path in global QoS algorithms is based on finding the shortest path by applying some variant of Dijkstra’s algorithm or Belman-Ford’s algorithm. In localised routing algorithms, the preferred path is selected from a set of the candidate paths \( R \) by a localised mechanism. Therefore, most of the global routing algorithms take no less than \( O(N \log N + L) \) time to select the path, where \( N \) refers to the network size measured by the number of the nodes. The complexity in finding and setting up the path in localised algorithms is \( O(H) \), where \( H \) is the average length of the candidate paths in hops. This is because in setting up a path each hop of the chosen candidate path needs to be traversed by the setup message.

F. Effective Dynamic Path Selection Method

This method improves the functionality of existing localized algorithms as well as our three new algorithms. Basically, the candidate path set has a certain number of candidate paths between sources and destinations. In our dynamic method the amount of blocking in every path, between each pair of source and destination nodes is calculated. It is followed by modification of the candidate paths set by replacing the path of higher blocking with the lower blocking ones. The main difference between the recalculation method described in the earlier paper and the suggested dynamic method is that the second technique replaces the set frequently on the basis of the given system variable value, the RecalculatePathsPackets. It gives the period or the number of connection requests to start the dynamic method, whereas recalculation changes the set only once.

Although the selection of disjoint path technique develops the algorithms performance, the same method combined with the dynamic approach gives superb results and drops the blocking probability.

Fig. 7 represents the success of recalculation method and shows the impact of it and also shows the benefit of dynamic method. We run a variety of experiments under fixed load 0.2 under same network conditions. The recalculation method results stay the same, even if the system variable (number of connection requests) increases. However, it shows improved performance in case of using the fixed

![Figure 7. Impact of Bursty traffic on RAND80 topology.](image1)

![Figure 6. Dynamic method performance.](image2)
candidate path set (Fig. 4). Although the dynamic method gives fluctuating results with every single change in the value of the dynamic system variable, it still performs better than a single recalculation of the path set. In addition, combining the disjoint path with both recalculation and dynamic methods once again results in the disjoint path method giving best results.

VI. CONCLUSIONS AND FUTURE WORK

On the basis of the previous research of localized routing, we analyzed the functionality of the CBR which appeared to be the best amongst existing global and localized routing algorithms. In our study we offered two methods to improve the performance of the CBR algorithm and introduced three new localized routing algorithms, HMB, HLABH and BRB. We analyzed their performance and compared them to CBR and WSP in different network topologies. In three types of networks, ISP, RAND45 and RAND80, our algorithms consistently performed better than both CBR and the global routing algorithm WSP.

The methods offered for selecting the candidate paths, which are disjoint paths and recalculation, not only improved the function of the CBR algorithm but allowed the three proposed algorithms to perform more beneficially. We analyzed the performance of the HMB, HLABH and BRB algorithms with disjoint and recalculation path methods. The HMB algorithm generally gave the best performance and decreased blocking probability the most.

Although the CBR algorithm, based on highest path credit, was designed to select the best candidate path, it still does not reflect the quality of path like bandwidth, and delay, which both are examples of QoS constraints and mirror the path superiority to be selected or rejected. Our three algorithms in this paper act using bandwidth as QoS metric. Future work will investigate the influence of delay as a QoS metric on the performance of some new localized routing algorithms.

REFERENCES