Manycast Multiple QoS Constraints Based Routing Algorithms Over Optical Burst Switched Networks

Suganthi.S, Reethi.S

Department of Electronics and Communication Engineering, Anna University
Sri Venkateswara College of Engineering, Chennai, Tamil Nadu, India

Abstract— Distributed applications such as video conferencing require data to be transmitted to a group of destinations from a single source. Such applications can be implemented using multicasting. However in multicasting, if any one of the fixed members in the group can not satisfy the service requirement of the application, the multicast request is said to be blocked. This drawback of multicasting has paved way for the communication paradigm called manycasting, where in, the destination can join or leave the group dynamically, depending on whether it satisfies the service requirement or not. Manycasting over Optical Burst Switched (OBS) networks is Quality of Service (QoS) constraints based. These multiple constraints are in the form of noise factor, propagation delay and reliability of the link. Due to the consideration of multiple QoS constraints, there is a possibility of the request to get blocked. The algorithms for reducing the request blocking are called Multiconstrained Manycast (MCM) algorithms. In MCM-Nearest Destinations (MCM-ND), the burst data is transmitted from the source to the minimum nearest destinations required for the manycast request. The destinations are selected among the candidate destinations on the basis of their shortest paths from the source, provided they satisfy the QoS constraints. However, if any of these primary destinations fail to satisfy the QoS constraints, the request gets blocked. On the other hand, in improved-MCM (I-MCM), all the candidate destinations that satisfy the QoS constraints are available, out of which the minimum required destinations are chosen using genetic algorithm. Our simulation results show that, in most scenarios I-MCM performs better than MCM-ND in terms of lower average request blocking, thus making it useful for data as well as real time service applications.

Keywords— manycast, optical burst-switched networks (OBS), quality of service (QoS), QoS routing.

I. INTRODUCTION

There has been recent emergence of many distributed applications, such as video conferencing, telematics, grid computing, storage area networks (SANs), and distributed content distribution networks (CDNs), which require large amounts of bandwidths and an effectual communication between single source and a set of destinations. Providing connections based on QoS to these applications is an important issue. QoS may include delays incurred during transmission, reliability, and signal degradation. These distributed applications require a single source to communicate with a group of destinations. Traditionally, such applications are implemented using multicast communication. A typical multicast session requires creating the shortest-path tree to a fixed number of destinations.

The fundamental issue in multicasting data to a fixed set of destinations is receiver blocking. If one of the destinations is not reachable, the entire multicast request (say, grid task request) may fail. A useful variation is to dynamically select destinations depending on the status of the network. Hence, in distributed applications, the first step is to identify potential candidate destinations and then select the required number. This dynamic approach is called manycasting.

Manycast, also called as quorumcast, is a generalization of multicast. In manycasting, the group of destinations that receive the message are to be selected rather than being given. With the advent of many Internet-based distributed applications, there is a continued demand for a high capacity transport network. Networks based on wavelength division multiplexing (WDM) are deployed to tackle the exponential growth in the present Internet traffic. WDM network include optical circuit switching (OCS), optical packet switching (OPS), and optical burst switching (OBS). One of the primary issues with OCS is that the link bandwidth is not utilized efficiently in the presence of bursty traffic. On the other hand, many technological limitations have to be overcome for OPS to be commercially viable. OBS networks overcome the technological constraints imposed by OPS and the bandwidth inefficiency of OCS networks. In OBS the user data is transmitted all-optically as bursts with the help of an electronic control plane.

Manycasting over optical burst-switched networks (OBS) is based on multiple quality of service (QoS) constraints. These multiple constraints can be in the form of physical layer impairments, transmission delay and reliability of the link. Each application requires its own QoS threshold attributes. Destinations qualify only if they satisfy the required QoS constraints set up by the application. Due to multiple constraints, burst blocking could be high.

In this paper, to reduce burst blocking, algorithms are proposed, that provide manycast QoS based routing to selected destinations over OBS networks. The approach incorporates multiple constraints related to different services. The proposed methods are service- centric and completely decentralized, as here only the local-network state information is made use of. The rest of the paper is organized as follows: we first discuss related work in this topic in Section I-A. In Section II, problem formulation for ordering valid destinations based on service constraints is discussed. In Section III, the proposed algorithms are explained with suitable examples.
Performance evaluation is discussed in Section IV. Finally, Section V concludes the paper.

A. Related Work

Manycasting work has been first reported independently by [1] and [3] as the quorumcast problem and the k-Steiner tree problem. The manycast problem is suggested to be NP-hard in [3]. Supporting manycasting over OBS networks is necessary for bandwidth-efficient manycasting [4]. Apart from building minimum cost tree that spans from source to manycast group members, the necessity for QoS routing has been discussed by [2]. This paper discusses the quality of a tree in terms of source-destination end to end delay constraints imposed by applications that use the tree. As the delay-constrained quorumcast routing problem is NP-complete, an proficient heuristic QoS routing algorithm has been proposed in [2] with cost of the quorumcast tree close to that of optimal routing tree.

Apart from supporting manycasting over optical networks, we also need to offer QoS in OBS networks. This is because QoS provisioning methods in IP will not apply to the optical counterpart, as there is no store-and-forward model [5]. Such mechanisms for QoS provisioning in IP over OBS networks must consider the physical characteristics and limitations of the optical domain. Physical characteristics of the optical domain include optical-signal degradation, propagation delay incurred from source to destination, and link reliability from calamitous effects. As the optical signal traverses in the transparent optical network, with the absence of electrical regenerators there will be significant loss of power due to many impairments. These impairments can be attenuation loss, multiplexer/demultiplexer loss, optical-cross-connect switch loss, and split loss (for multicast capable switches) [6]. ASE noise present in the EDFAs decreases the optical-signal-to-noise ratio (OSNR). Decrease in OSNR increases bit error rate (BER) of the signal. Hence, the signal is said to be lost if BER is more than the required threshold [7] and [8]. Challenges and requirements for introducing impairment-awareness into the management and the control planes in WDM networks have been discussed in [9]. For the first time, impairment-awareness for implementing manycasting over OBS networks has been addressed in [6]. This paper discusses the importance of physical layer awareness and determines the loss due to burst contentions and high BER. Another important QoS parameter is the reliability of the links along the end-to-end path between the source and the destination. A novel dynamic least cost multicast routing protocol using hybrid genetic algorithm (GA) has been discussed in [10]. This paper defines as to how GA is hybridized by incorporating the heuristic local search function and by using improved genetic operators.

II. PROBLEM FORMULATION

In this section, we explain the mathematical framework for Manycasting that is multiconstrained. The work focuses on selecting the best possible destinations that can meet the service demands effectively. Destinations chosen must be able to provide quality of service attributes. A destination is said to qualify as the member of quorum pool if it satisfies the service requirements of the application. The proposed methods are based on distributed routing, where each node individually maintains the network state information and executes the algorithms. Algorithms implemented in the centralized way, may fail due to a single failure and resulting in poor performance.

A. Notations

\[(o, D_o, d_o)\] is a manycast request where \(o\) is the source node, \(D_o\) is the destination set, elements of which are probable candidates for the particular service request, and \(d_o\) is the minimum number of destinations that are required to participate in the manycast session for the job to be successfully completed. Manycast session is also denoted by \(g/d_o\). Manycasting can be considered as the dynamic version of the multicast, where in the members can leave and new members can join, so that minimum required destinations will always participate in the session. A destination is said to qualify as the member of quorum pool if it satisfies the service requirements of the application, such as BER, reliability, and delay.

B. Service Attributes

The service attributes considered here are the noise factor \((\alpha_o)\), reliability factor \((\beta_0)\) and the end-to-end propagation delay \((\kappa_o)\) for the Link \(j\), respectively.

Noise factor \((\alpha_o)\) is defined as ratio of input optical signal to noise ratio \((\text{OSNR}_{in})\) and output optical signal to noise ratio \((\text{OSNR}_{out})\). Here OSNR is defined as the ratio of the average signal power received at a node to the average ASE noise power at that node. The overall noise factor of a burst that has traversed \(H\) hops is given by

\[
\alpha_H = \prod_{k=1}^{H} \alpha_k
\]

where in the above equation, the product is performed \(H\) times, starting from the initial link.

Reliability is another factor considered for providing services. The reliability factor \(\beta\) of a link, is defined, such that \(0 \leq \beta \leq 1\), it indicates the percentage of reliability for a particular link. The reliability prediction method involves the calculation of down times contributed to all building blocks required to establish end-to-end network path.

The end-to-end reliability for the path traversing \(H\)-hops is

\[
\beta_H = \prod_{k=1}^{H} \beta_k
\]

The last attribute that is considered as an important service parameter for distributed applications is propagation delay. If \(\kappa\) is the propagation delay of a link, then end-to-end delay for \(H\) hops, is given by

\[
\kappa_H = \sum_{k=1}^{H} \kappa_k
\]
For each service there exists a threshold parameter (or constraint) that is defined as $T^{(\theta_p)}$ and is given by

$$T^{(\theta_p)} = \left(\begin{array}{c}
\frac{\alpha_{\max}}{eta_{\min}} \\
\frac{\beta_{\max}}{\delta_{\max}}
\end{array}\right)$$

(4)

For the successful establishment of QoS-based Manycast session, the chosen destinations must satisfy the service requirements as defined in (4).

C. Path Information Vector

The service attributes can be used to maintain the local network information and by properly comparing these vectors, the destinations can be chosen. Comparison of multi dimension metrics can be done using the notion of lattices. Lattices are explained using the ordering, which has the properties of reflexivity, antisymmetry, and transitivity.

The path information vector from source $o$ to destination $d$, is given by

$$\Omega_{<o,d>} = \left(\begin{array}{c}
d-1 \\
\sum_{k=0}^{d-1} \alpha_k \\
\sum_{k=0}^{d-1} \beta_k \\
\sum_{k=0}^{d-1} \kappa_k
\end{array}\right)$$

(5)

Thus, for the successful transmission of data between the source and the destination, the path information vector must satisfy the threshold set and it is denoted in the following way.

$$\Omega_{<o,d>} \leq T^{(\theta_p)}$$

(6)

Here, in every case, the noise factor and the end-to-end propagation delay should be less than their pertaining threshold values. However, the reliability should be greater than its pertaining threshold value. If these conditions are satisfied, then the threshold conditions are said to be satisfied and the transmission of data is feasible.

III. MULTICONSTRANDED MANYCAST (MCM) ALGORITHMS

In this section, we explain the proposed MCM algorithms with the help of illustrative examples. We propose two algorithms, MCM-nearest destinations (MCM-ND) and Improved-MCM (I- MCM) for evaluating the performance of manycasting with QoS constraints.

A. Multiconstrained Manycast- Nearest Destinations (MCM-ND) Algorithm

In this section, we explain the MCM-ND algorithm. The pseudocode for this is given in Algorithm 1. When a new burst arrives in the network, it is assigned a unique burst ID, $id$. A Burst Header Packet (BHP) is created for this burst, where $v = o$ with the fields destination set $D_o$, quorum pool $d_o$, threshold parameters $T^{(\theta_p)}$ for the service $\theta_p$. This is indicated in line 1, in the algorithm. Destinations in $D_o$ are sorted on the basis of shortest path using $SORT.SP(D_o)$. Next-hop nodes from $o$ to $d'_o$ are calculated and added to the set $N$. The loop in lines 7–10 selects the first $d$ destinations from $D'_o$ and next-hop nodes are added to $N$ for each destination $d'_o$ using line 9. The path information vector is calculated using path algebra explained in Section II-C. If QoS constraints is satisfied for link between $o$ and $n_i$, then the link $<o,n_i>$ qualifies the QoS threshold attributes for the service $\theta_p$. The set of all destinations that use node $n_i$ as the intermediate node is given by the set $DEST[n_i]$. The new destination set is given by $D'_o$ as shown in line 14. The BHP at node $n_i$ is updated with the new values as given by line 15. If the condition given in line 13 were false, then the manycast request is said to be blocked, as the minimum number of members in the pool are less than the required. We refer to this blocking as QoS Blocking. This algorithm repeats until the destinations are covered for a burst. Fig. 2 shows the shortest-path tree for the given manycast request of the NSF network in Fig. 1, with links shown in bold dotted lines for a 7/4 destination group configuration. Here out of 7 candidate destinations, minimum of 4 destinations are required for carrying out the Manycast request successfully. Let the service threshold be $T^{(\theta_p)} = [6, 0.6, 20ms]$. In order to guarantee QoS for the service, our aim is to identify destinations that have overall noise factor $\alpha \leq 6$, reliability $\beta \geq 0.6$ and the propagation delay $\kappa \leq 20$ ms. When the burst enters the network at source (node 0) burst head field values are updated with the initial values. Using $SORT.SP$ the new destination set is given by $D'_o$. In MCM-ND we select the first $d_0$ from $D'_o$. The path information vector is computed as in line 12. Computation of noise factor is done using the parameter values given in [6]. Noise factor, reliability and end to end propagation delay are calculated using the (1), (2) and (3). In the above illustration, out of destinations 4, 6, 13, 11, 10, 9 and 12, destinations 4, 6, 13 and 11 are the minimum required destinations to carry out the manycast request considered. As all the four destinations satisfy the QoS constraints, the manycast request is successful. We thus see that the manycast session is successful for the given service if the QoS threshold conditions for the service are met by all the required destinations. The same manycast request can be blocked for different service threshold conditions. If at least one of the destinations is not reachable through the next-hop node, due to insufficient QoS, the entire manycast request is said to be blocked. The same procedure can be carried out for 3/2 destination group configuration.

Algorithm 1 Multiconstrained Manycast-Nearest Destination (MCM-ND)

1: Gather $(o, D_o, d_o, T^{(\theta_p)})$
2: if $v \in D_o$, then
3: $D_v \leftarrow D_v \setminus \{v\}$
4: $d_v \leftarrow d_v - 1$
5: else
1: Gather candidate destination group configuration (3/2 or 7/4).
2: Gather the (o, D_o, d_o, T^(θ_p)).
3: for i ← 1 to D_o do
4: Procedure.QoS( );
5: end for
6: if counter < d_o then
7: DELETE.BURST
8: end if
9: if counter = = d_o then
10: Procedure.Transmit( );
11: end if
12: if counter > d_o then
13: Procedure.Genetic_algorithm( );
14: end if
15: Procedure.QoS( ) {
16: for j ← 1 to D_o do
17: for i ← 1 to |N| do
18: Ω_{v_o,n_i} ← Ω_{v_o-1,n_i} o Ω_{v_o,n_i};
19: if Ω_{v_o,n_i} ≼ T^(θ_p) then
20: D_o ← DEST[n_i];
21: UPDATE.BHP[ n_i ];
22: else
23: DELETE.BURST
24: end if
25: end for
26: if Ω_{D_o,n_i} ≼ T^(θ_p) then
27: dest_{counter} ← dest_{counter +1};
28: end if
29: end for
30: Procedure.Transmit( ) {
31: Transmit data to destinations dest
32: Procedure.Genetic_algorithm( ) {
33: for j ← 1 to counter do
34: for i ← 1 to |N - 1| do
35: dis_i ← DISTANCE[ n_i , n_{i+1} ];
36: end for
37: dis_i ← dis_{i-1}
38: end for
39: Procedure.Fitness_cal( );
40: Procedure.Selection( );
41: Procedure.Crossover ( );
42: offspring_1, offspring_2 ← MUTATE [offspring_1 , offspring_2];
43: Procedure.Reinsertion( );
44: Procedure.Transmission( );
45: Procedure.Fitness_call( ) {
46: for j ← 1 to counter do
47: fitness_i ← 1/dis_i ;
48: end for
49: Procedure.Transmit( );
50: sum ← 0;
51: for k ← 1 to counter do
52: sum ← sum + fitness_k ;
53: end for
54: for i ← 1 to counter do
55: procedure Transmit() {
56: Transmit data to destinations from the next-hop node, defined by Procedure.Genetic_algorithm(), in order to carry out the manycast request successfully.

**Algorithm 2 Improved-Multi-Constrained Manycast (I-MCM)**

---

**B. Improved- Multi-constrained Manycast (I-MCM) Algorithm**

The I-MCM is given in Algorithm 2. It contains two main procedures (1) for calculating the QoS parameters, defined as Procedure.QoS() and (2) for determining the number of destinations and the minimum destinations that can be reached from the next-hop node, defined by Procedure.Genetic_algorithm(), in order to carry out the manycast request successfully.
Instead of first sorting the candidate destinations set and then checking if they satisfy the QoS constraints, the QoS constraints are checked for each destination first, as shown in lines 16-25. The counter is incremented for each valid destination that satisfies the QoS constraints as shown in lines 26-29. If the number of destinations satisfying the QoS constraints is equal to or more than do the algorithm is proceeded with. Otherwise, the manycast application is said to be infeasible.

If the number of candidate destinations satisfying the QoS constraints is equal to the minimum number of destinations required for the manycast request to be feasible, then the burst data is transmitted to those pertaining destinations.

If the number of candidate destinations satisfying the QoS constraints is greater than the minimum number of destinations required, then apply genetic algorithm to determine the minimum required destinations to carry out the manycast request successfully.

Genetic algorithm involves the creation of population, by providing the paths from source to the valid candidate destinations, dest. These paths form the samples of the population. Determine the distance from the source to each of the destinations. Reciprocal of each distance value gives the fitness values for each of the pertaining destinations, as shown in lines 47-49. The first and the second minimum fitness values are noted down. The fitness values normalized. Two random values are generated, to be used in Roulette wheel selection.

The roulette wheel is divided into sections starting from 0.001 to 0.999, with the normalized fitness values being the boundary values. The ranges or sections into which the random values lie are checked for. Each range pertains to each of the sample of the population. The samples associated with the selected ranges are taken as parents for crossover denoted by crossover1 and crossover2. This procedure is shown through lines 57-71.

The 2-point crossover is carried out. The starting and ending point in the two parents are gathered and the nodes in between are swapped with each other among the parents.

This is followed by the mutation process. Here, say a two digit number like 23 is taken. This implies that at 2nd position, replace the node by node number 3. Thus, this gives the two mutated offsprings denoted by offspring1 and offspring2. These offsprings replace those samples of the population that have the least and second least fitness values (dest_{min.fitness1} and dest_{min.fitness2}).

Thus this procedure gives the apt valid destinations, required for the carrying out of the manycast request successfully.

We consider the same illustration as taken for the MCM-ND algorithm. Given that the candidate destinations are 4, 6, 13, 11, 10, 9 and 12. Here the destinations are checked for QoS. If destination 6 fails to satisfy the QoS constraints, the rest of the available destinations (dest), are 4, 13, 11, 10, 9, 12. Based on the distance of the destinations from the source, the destinations with the minimum fitness values are 12 and 9. The parents for crossover are 13 and 11, which are obtained using Roulette wheel selection. After the mutation procedure, the offsprings are 11 and 13. These destinations replace destinations 12 and 9. As for the 7/4 configuration, minimum four destinations are required for the manycast request to be successful. Thus, using genetic algorithm, the destinations obtained are 4, 13, 11, 10, to which transmission of data should take place for the manycast request to be feasible.

We see that the manycast request that would have been blocked using MCM-ND if one of the destinations is blocked is now satisfied. I-MCM algorithm provides additional destinations from which the required minimum number of destinations can be chosen, provided they satisfy the QoS constraints.
IV. PERFORMANCE EVALUATION

As aforementioned, we have intended destinations, i.e., quorum. The candidate destination group can be small, medium or large. Two typical configurations 3/2 and 7/4 are considered for simulations. We differentiate among service requirements, i.e., different services have different constraints. Differentiated services considered for simulation are \( T^{(θ1)} = [5.7, 0.6, 20]^T \), \( T^{(θ2)} = [5.7, 0.6, 10]^T \), \( T^{(θ3)} = [4.25, 0.9, 10]^T \) and \( T^{(θ4)} = [4.25, 0.8, 10]^T \). The service \( θ_1 \) can represent data service as it has relaxed delay requirements. The service \( θ_2 \) can represent real-time service as it has stringent delay requirement. Service \( θ_3 \) and service \( θ_4 \) have high service threshold requirements. The above mentioned four threshold sets are denoted as Th-1, Th-2, Th-3, Th-4 in Figures 4, 5, 6.

Fig. 4 shows the performance of 3/2 manycast configuration for all services. Services \( θ_1 \) and \( θ_2 \) of 3/2 manycast configuration have more relaxed threshold parameters, hence, we can improve the blocking marginally using I-MCM. We observe an interesting result in Fig. 4. In the case of \( θ_4 \) for 3/2 manycast configuration, we find I-MCM to offer lower request blocking than MCM-ND significantly. This is because service \( θ_4 \) has high QoS requirement (real-time service). In an attempt to decrease the request blocking, MCM-ND selects only the nearest minimum required destinations, discarding the rest of the candidate destinations. If these nearest destinations have their service attributes exceed the threshold requirements, the manycast request fails, as there are no secondary destinations available for the replacement of the failed primary destinations. The option of checking additional destinations for QoS constraints is left unexplored.

Next we present simulation results for 7/4 manycast configuration. Fig. 5 shows the comparison of the MCM-ND and I-MCM algorithms for different set of services. The I-MCM adds or removes destinations. However, destinations that are added to the quorum pool can be, at times, at a longer distance than the blocked destination. As the result, QoS of this destination can be decreased. In spite of decrease in values, if the path-information vector is within the service threshold, the request can be satisfied. We can thus observe that I-MCM can be chosen for data service applications, where there is no specific upper bound on the propagation delay of the burst. I-MCM can also be used for real time service applications, where the destinations have the propagation delay within the limits.

Lastly, we plot a bar graph (Fig. 6) that compares the MCM-ND and I-MCM algorithms on the basis of number of requests blocked for the individual QoS constraints namely the noise factor, reliability and the propagation delay. For the noise factor and the reliability, I-MCM performs better with lower number of requests blocked as compared to MCM-ND. They behave the same way in the case of propagation delay, as the destinations selected with the help of genetic algorithm could be at longer distances from the source in the case of I-MCM.

V. CONCLUSION
In this paper, we propose algorithms to support manycast QoS-based routing over optical burst-switched networks and carry out the comparison of the proposed algorithms. Our QoS model supports certain service parameters for the transmission of optical bursts, such as physical impairments, reliability, and propagation delay. By using distributed scheduling algorithms, bursts are routed to the destinations based on the QoS conditions. We propose the multiconstrained manycast-nearest destinations and the improved-multiconstrained manycast algorithm to support manycast QoS-based routing over OBS networks. Four types of services were considered to evaluate and compare the performance of manycast QoS-based routing algorithms. They represent the service requirements of data service and real-time service. From the simulation results, it is found that in most of the scenarios, I-MCM algorithm performs better than the MCM-ND algorithm by having lower average request blocking. The dynamic membership based destination group and the utilization of genetic algorithm in I-MCM algorithm to determine the destinations required to make the manycast request feasible, decrease the average request blocking.

ACKNOWLEDGMENT

I would like to thank my supervisor S.Reethi, Department of Electronics and Communication Engineering, Sri Venkateswara College of Engineering for her support and valuable inputs.

REFERENCES


