Dynamic Load-Balanced Manycasting Over Optical Burst-Switched (OBS) Networks

Rajesh R.C. Bikram and Vinod M. Vokkarane
[*] *Department of Computer and Information Science, University of Massachusetts, Dartmouth, MA 02747, USA*
E-mail: {rrc,vvokkarane}@umassd.edu

Abstract

Data-loss in optical-burst-switched networks primarily occur due to contention of bursts at the core nodes. We propose two dynamic congestion-based load-balanced manycast-routing techniques, namely load-balanced SPT and load-balanced DM that significantly improve blocking-probability and end-to-end-delay. [1]

OCIS Codes: 060.4259 Networks, packet-switched; 060.4255 Networks, multicast

I. INTRODUCTION

Increasing amount of data use in the Internet cloud explicitly hikes the demand for more bandwidth. Optical fiber has the capacity to accommodate this demand for bandwidth. Optical burst switching (OBS) uses fiber optics to transfer bits over a long haul network. Several next-generation distributed applications require a single source to communicate with multiple destinations, such as grid computing and storage area networks. Traditionally applications implement multicasting to transmit data from a source to multiple destinations. The fundamental issue with multicasting is that the destination set is fixed. As the network becomes congested, traditional multicasting algorithms cannot guarantee that the information reaches all the desired destinations. One variation to multicasting that could help, is to dynamically pick the destinations from a larger candidate set of destinations depending on the status of the network. This technique is commonly referred to as *manycasting* and the problem is defined as follows: given a network $G(V, E)$, with V nodes and E edges, edge cost function is given by $g : E \rightarrow R^{+}$, an integer k, a Source s, and the subset of candidate destinations $D_c \subseteq V$, $|D_c| = m \geq k$, where $|D_c|$ is the cardinality of the set D_c. Note that if $k = 1$, one destination is chosen from the set D_c and this is called *anycasting*.

In an OBS network, multiple packets destined to the same egress node are assembled together in to a single data burst at the ingress node. Control information for this data burst is transmitted ahead on separate channel and is called *burst header packet (BHP)*. BHPs are processed electronically at each intermediate node to reserve network resources before the data burst arrives at the node. After a certain offset, time data burst is transmitted all-optically through the network. Data loss in OBS network primarily occur due to burst contentions. Burst contentions in OBS networks occur due to burstiness of IP traffic and the lack of optical buffering. Contention occurs when multiple bursts contend for the same outgoing port at the same time. Many schemes have been proposed to resolve burst contentions, such as wavelength conversion, optical buffering, and deflection routing [2]. All contention resolution techniques react after the contention has already occurred. An alternative approach is to avoid contentions using contention avoidance techniques.

Load-balanced routing is an approach to implement contention avoidance in OBS networks [4]. Load-balanced routing involves two stages, *route calculation* and *route selection*. Both route calculation and route selection can be implemented in a static or a dynamic manner. In this paper, we adopt load-balanced routing using dynamic route-calculation approach as proposed in [4]. At every τ units, the network load, ρ, is calculated for every link in the network. Let τ_s and τ_d be the duration of successful burst arrivals and dropped burst arrivals during the time interval τ, respectively. The offered load on each outgoing link is expressed as the duration of all arriving bursts over the interval τ, given by, $L_{i,j} = \frac{\tau_s + \tau_d}{\tau}$. Once the load information is calculated, a new route tree is created for each manycast request.

The two manycast routing algorithm, shortest path tree (SPT) and dynamic membership (DM) are purposed in [1] that minimize data contention loss in OBS networks. These algorithms generate route trees to send bursts from the source to a set of destinations. The generated route tree does not consider dynamic behavior of network, such as congestion. To overcome this problem, we extend our previous work on manycasting to provide dynamic congestion-based load-balanced manycast routing. Given a manycast request, we implement manycast routing algorithms that create load-balanced manycast route trees. The rest of the paper is organized as follows: Section II discusses issues related to supporting manycasting over OBS networks. In Section III we describe the proposed congestion-based load-balanced manycasting algorithms over OBS. Extensive simulation results are presented in Section IV, where we compare the average blocking probability and the average end-to-end delay of different algorithms with and without load-balanced manycast routing. Finally, Section V concludes the paper.

II. MANYCASTING SERVICE

A manycast request is denoted by (s, D_c, k). Each burst is comprised of a single manycast request from Source S to k destinations out of m ($|D_c| = m$) possible candidate destinations. There is no guarantee that exactly k destinations receive the burst, due to burst loss that occurs due to burst contention. In general, most multicasting solution approaches are largely applicable to manycasting. Networks that can support optical multicast can also support optical manycasting. Thus, manycasting can be implemented using multicast-capable optical cross-connects (MC-OXC) [5]. In order to route the manycast request (burst), shortest-path tree (SPT) can be computed, as given below:

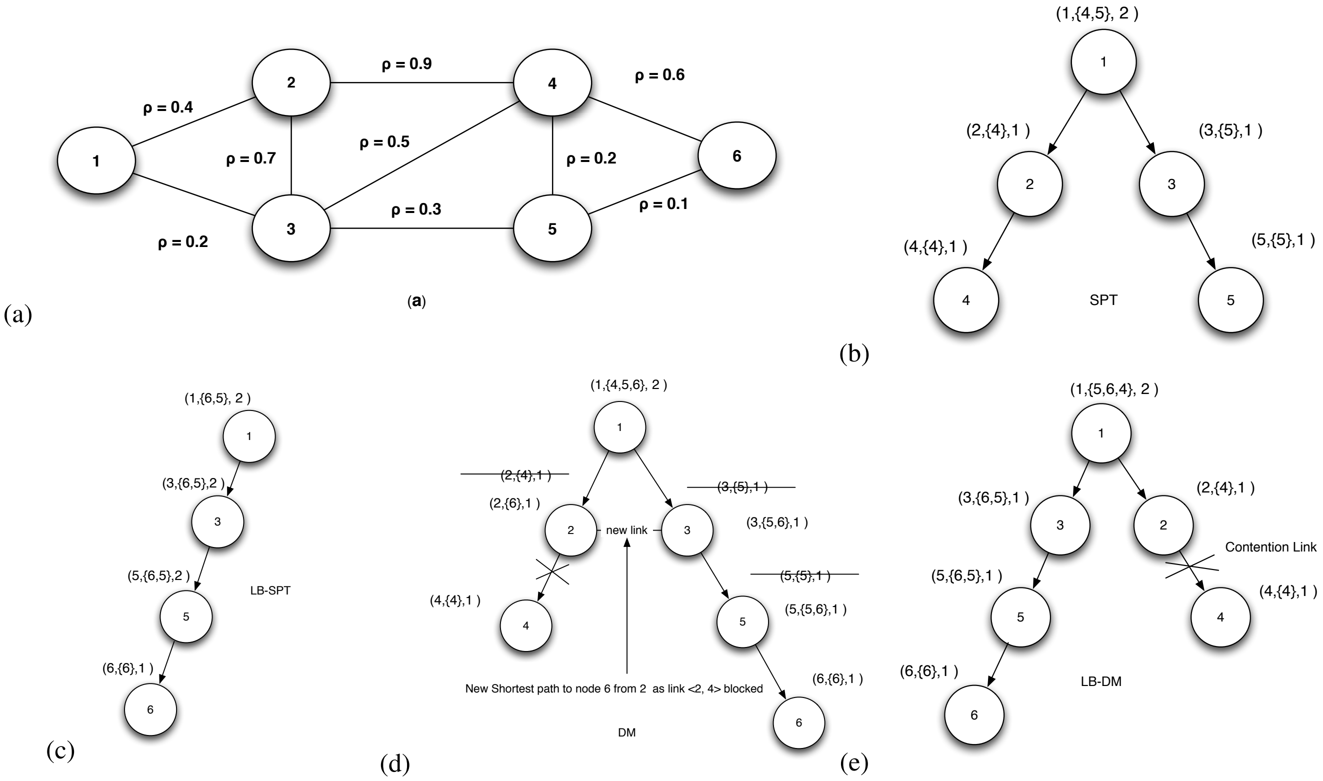
- *Step 1*: Find the shortest path from Source s to all the destinations in D_c. Let $D_c = \{d_1, d_2, \ldots, d_{|D_c|=m}\}$ and the minimum hop-distance from s to d_i, where $1 \leq i \leq m$ is $H^{(s)} = \{h_1, h_2, \ldots, h_m\}$.

---

[1] This work was supported in part by the National Science Foundation (NSF) under grant CNS-0626798.

- *Step 2*: All the destinations in D_c are sorted in the non-decreasing order according of the hop-distance from Source s to the destinations. Let D'_c be the new set in this order, given by $\{d'_1, d'_2, \ldots, d'_m\}$.
- *Step 3*: Select the first k destinations from D'_c.

In dynamic membership (DM) algorithm, a designated set of k destinations is tentatively set up at the source node. Instead of discarding the remaining $(|D_c|)$ - k) destinations, we evenly distribute the remaining destinations into all child branches at the source node. If any designated destination is blocked at an intermediate node, we send the burst to some of these extra destinations such that the total number of destinations that actually receive the burst is still no less than k. Therefore, in DM the designated set of k destinations may change dynamically along the route tree according to the status of the network.



(a) (b) (c) (d) (e)

Fig. 1. (a) Network Topology Graph. (b) Shortest Path Tree (SPT) (based on hop-distance). (c) Load-balanced Shortest Path Tree (LB-SPT). (d) Dynamic Membership (DM) route tree. (e) Load-balanced Dynamic Membership (LB-DM) route tree.

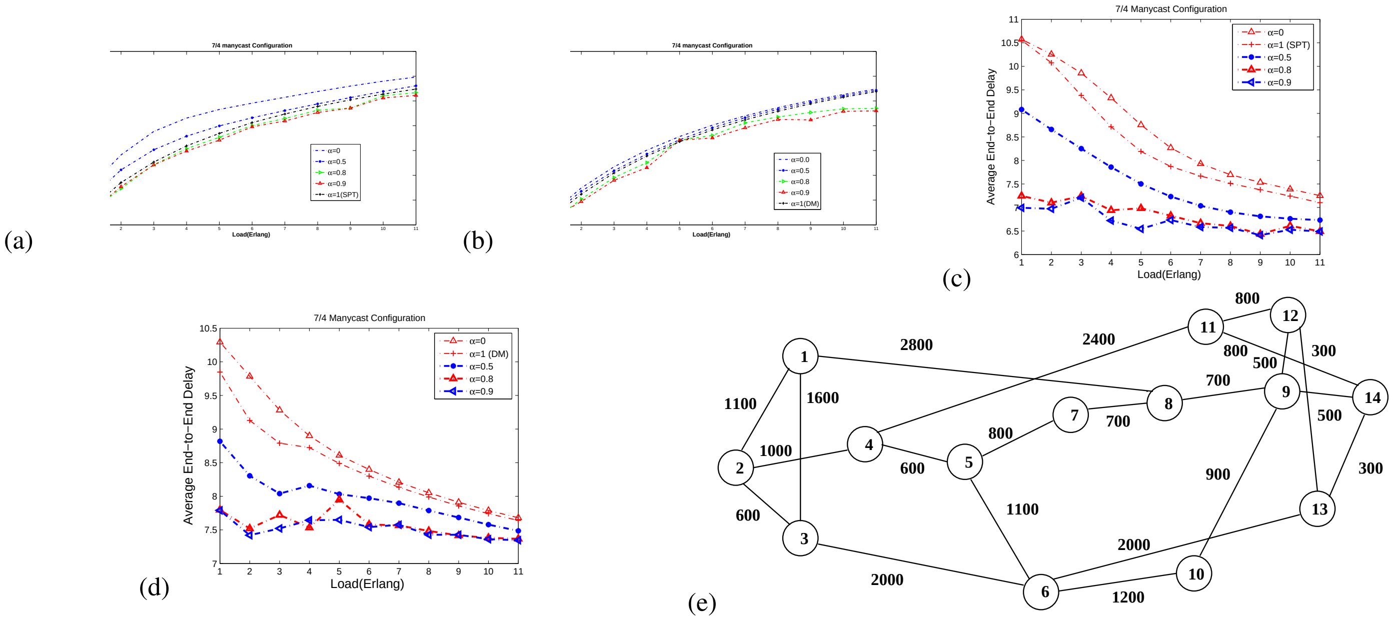
III. DYNAMIC CONGESTION-BASED LOAD-BALANCED MANYCAST ROUTING ALGORITHMS

In order to implement the proposed techniques the core nodes measure the offered load on each outgoing link in the network, $\rho(i,j)$. Load is expressed as the total burst arrival duration over the interval τ, which is $\rho(i,j) = \tau_{success} + \tau_{drop}/\tau$. Load status of all the links is updated periodically every τ units. A least-congested manycast route tree is calculated for each manycast request based on dynamic congestion information. The weight, $w_{i,j}$, is based on congestion as well as hop distance: $W_{i,j} = \alpha h_{i,j} + (1-\alpha)\rho_{i,j}$, where $h_{i,j}$ is hop-count and $\rho_{i,j}$ is the offered load on Link i,j.

All these algorithms create a dynamic route tree for each manycast request. Before sending the manycast burst to the candidate destinations, a least-congested routing path is created for every candidate destination of manycast request. The destination paths are placed in non-descending order of their cumulative path weights, $W_{s,d} = \sum_{L_{i,j} \in R_{s,d}} W_{i,j}$, where $R_{s,d}$ is the route from Source s to Destination d.

We now describe the proposed contention-avoidance techniques using dynamic congestion-based load-balanced manycast routing algorithms. Load-balanced SPT (LB-SPT) algorithm is based on the SPT. After every fixed τ interval, the weights of all the links are updated. Using the updated weights, LB-SPT creates a new route tree for every arriving manycast request using the new ordered destination set, that is ordered based on cumulative path weights to each candidate destination. Load-balanced DM (LB-DM) algorithm uses the LB-SPT. In addition, LB-DM dynamically recovers from blocked destinations by recomputing new least-congested paths from the contention node to the next destination in the ordered set.

Consider the following illustrative example. Fig. 1(a) shows the network topology, with each Link i,j labelled with a congestion (offered load) value, $\rho(i,j)$. Based on the weight function for each destinations, we create an ordered destination set for a given manycast request. After calculating the weight values for each destination set, a route tree is created for all four algorithms: SPT, LB-SPT, DM, and LB-DM. For a manycast request, $(1, \{4,5,6\}, 2)$ originates from the Node 1 and is intended to reach two destinations out of the candidate destination, $\{4,5,6\}$. In SPT, we first calculate the weight function for each destinations of manycast request. The calculated weight values are used to order the destination set. For example, the weight values for the $R_{1,4}$ need two hops to reach to destination and so on. In SPT, the source selects destinations Node 4



(a) (b) (c) (d) (e)

Fig. 2. Average loss values based on different load and α values for (a) LB-SPT and (b) LB-DM. Average end-to-end delay for different load and α values for (c) LB-SPT and (d) LB-DM. (e) NSF simulation network.

and Node 5. If Link $(3, 5)$ is busy, then the burst reaches only Node 4 and the manycast request is blocked. For the LB-SPT, we first calculate the path weights from the source to each candidate destination. These weight values are used to order the candidate destination set. For example for the Path $1, 4$, the weight is given by $W_{1,4} = W_{1,2} + W_{2,4}$. If we set $\alpha = 0.5$, $W_{1,4} = 1.85$. Similarly, the weight for the Path $1, 5$ is $W_{1,5} = W_{1,3} + W_{3,5} = 1.25$, and the weight for Path $1, 6$ is $W_{1,6} = W_{1,3} + W_{3,5} + W_{5,6} = 1.80$. Based on the weight values to each destination from the source, the destination set are ordered as $(1, \{5, 6, 4\}, 2)$. The LB-SPT manycast request tree, $(1, \{5, 6\}, 2)$ is created as in Fig. 1(c).

Similarly, Fig 1(d) and Fig 1(e) describe about DM and LB-DM route tree generation algorithms for manycast request $(1, \{4, 5, 6\}, 2)$ and $(1, \{5, 6, 4\}, 2)$, respectively. As in Fig. 1(d) the Link $2, 4$ is congested because of a high ρ value, in this case Node 2 has to either drop the burst if there are no more destinations to send the burst or use the weight function to find alternate shortest path to reach for remaining destinations. As in the Fig. 1(d) we have still Node 6 remaining at the destination set. Node 2 uses the Path $< 2 - 3 - 5 - 6 >$ to reach to destination Node 6. The manycast request gets updated after finding the path for each new nodes to reach to destination Node 6 from the Node 2. But in Fig 1(e) we do not have to find the alternate path to reach to destination Node 4 as we already reached two destinations Node 5 and Node 6. Route trees are dynamic and change with the network load over the next fixed interval of time τ.

IV. Simulation Results

Using discrete-event simulations we compute average blocking ratio for SPT and DM with and without dynamic congestion-based load-balanced routing over a NSF network (Fig. 2(e)). Burst arrival to network follows Poisson distribution. Packet length is of 1250B. The transmission rate of burst is 10Gb/s. We have used τ value of 10ms. A high τ value, provides less frequent congestion updates to the ingress nodes that delays the creation of new route tree and all the burst continue to follow the same congested route tree. A low τ value, increase feedback information from the core to the ingress nodes increasing control overhead. So choosing the optimal τ value is important. In the same way, when α value is low there is more delay and more loss, since the burst has to travel many hops to reach its destinations. Choosing an optimum α value gives the LB-SPT and LB-DM better loss and delay performance in the network. Fig. 2(a) clearly shows, as the α value increases with load, LB-SPT performs better than the SPT alone. Fig. 2(b) clearly shows, LB-DM performs better than DM provided that the α value is between 0.7 to 0.9. When α value is set to 1, the weight function of the load-balanced algorithms is identical to SPT and DM. Fig. 2(c) and Fig. 2(d) describes the average end-to-end delay versus load for different α values. As the α increases, the average end-to-end delay decreases. When α is low, the delay in the network is high due to longer less congested paths.

V. Conclusion

In this paper, we purposed two dynamic congestion-based load-balanced manycast routing algorithms, LB-SPT and LB-DM. The proposed algorithms reduces burst loss probability in the network by creating dynamic least-congested route trees. Choosing optimal values of α and τ is important to achieve good performance. When α is between 0.7-0.9, the LB-SPT and LB-DM outperforms DM and SPT. Through extensive simulations, we showed that congestion-based load-balanced routing algorithm outperform traditional manycast routing algorithm.

References

[1] X. Huang, Q. She, V.M. Vokkarane, and J.P. Jue, “Manycasting Over Optical Burst-Switched Networks,” *IEEE ICC 2007*.

[2] S. Yao, B. Mukherjee, et. al., ”All-Optical Packet-Switched Networks: A Study of Contention Resolution Schemes in an Irregular Mesh Network with Variable-Sized Packets,” SPIE OptiComm 2000.

[3] C. Qiao and M. Yoo, “Optical Burst Switching (OBS) - A New Paradigm for an Optical Internet,” JHSN, Jan. 1999.

[4] G. Thodime et al., “Dynamic Congestion-Based Load Balanced Routing in Optical Burst-Switched Networks,” IEEE GLOBECOM 2003.

[5] B.G. Bathula, V. M. Vokkarane, and R. R. C. Bikram, “Impairment-Aware Manycasting Over Optical Burst-Switched Networks,” *IEEE ICC 2008*.