Resource scheduling in centralized OBS-based grids

Wei Dai, Guiling Wu,* Yahong Yang, and Jianping Chen

The State Key Laboratory on Advanced Optical Communication Systems and Networks, Shanghai Jiao Tong University, 800 Dongchuan Road, Shanghai, China 200240

**Corresponding author: wuguiling@sjtu.edu.cn*

Received September 6, 2007; revised November 21, 2007; accepted November 21, 2007; published January 10, 2008 (Doc. ID 87275)

A centralized optical burst switching (OBS)-based grid architecture is proposed. Three scheduling strategies, for the resources on the grid level, the network level, and the OBS network level, respectively, are presented and analyzed. The overall grid performance and the network performance under different scheduling strategies are analyzed and compared by simulation.

OCIS codes: 060.4259, 060.2330, 060.6719.

1. Introduction

Recently, much attention has been paid to the grid with the motivation not to merely transmit data as fast and as reliable as possible, but also to integrate all the available resources on the network to meet the rapidly growing application demands. There also has been significant evolution on the optical infrastructure technologies such as DWDM (dense-wavelength-division multiplexing) and fast optical switching. A photonic grid, which integrates optical network resources as well, seems technically matured. An interactive improvement and/or enhancement of the performance on both the optical side and the grid side can be expected [1].

A photonic grid needs an efficient optical network infrastructure. Optical circuit switching (OCS) and optical burst switching (OBS) are two of the promising infrastructures at present [2]. In comparison with OCS, OBS networks are more flexible for high-burst-data services. Its smaller granularity makes it easier for the photonic grid to schedule its resources. Simeonidou *et al.* [3] and De Leenheer *et al.* [4] have proposed a grid-over-OBS architecture (GoOBS). The distributed architecture is composed of a grid edge device, an intelligent OBS router, a grid user network interface (GUNI), and a grid resource network interface (GRNI). An anycast algorithm has been developed for this architecture, which forwards the bursts according to the resource information of the grid and the job requirement contained in the burst header packet (BHP) [5]. Also proposed is another OBS-based grid architecture [6], which consists of traditional OBS routers and active OBS routers that can perform grid function as well. Besides, a manycast algorithm for scheduling multiple host resources in OBS-based grid computing was recently reported [7]. Scheduling is a key issue in the OBS-based grid. It is relevant to the adopted OBS-based grid architecture.

In this paper, a centralized OBS-based grid is proposed. The corresponding resource scheduling strategies in the centralized OBS-based grid is studied on three levels, i.e., the grid level, the network level, and the OBS network level. The overall grid performance and the network performance of the three scheduling strategies are analyzed and compared for different application numbers, access bandwidth, terminal configurations, etc. The paper is organized as follows. In Section 2, the proposed architecture of a centralized OBS-based grid is given. In Section 3, three scheduling strategies for the centralized OBS-based grid are described in detail. In Section 4, the simulation results for different scheduling strategies and configurations are presented and analyzed. Section 5 is the conclusion.

2. Centralized OBS-Based Grid

The proposed centralized OBS-based grid is shown in Fig. 1. It consists of one grid scheduler node (GSN), traditional OBS core nodes, and grid edge nodes. The grid edge

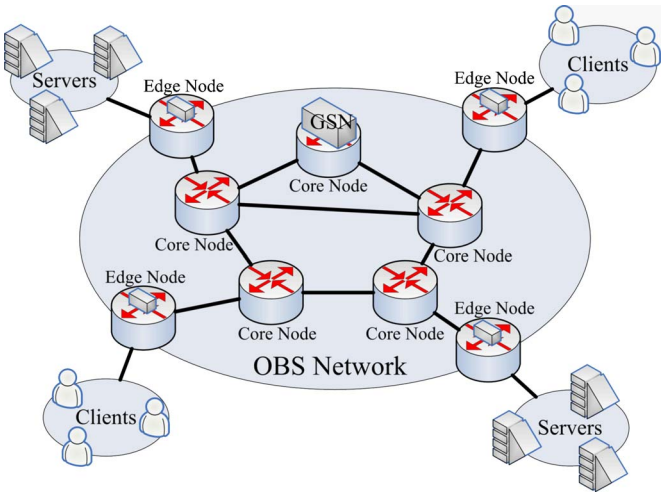

Fig. 1. Centralized OBS-based grid.

node has a grid agent, which collects job requests from grid clients connected to it, generates request packets, and sends it to the GSN through burst control channels. The GSN maintains information such as the resource of each server, the load of each server, the network topology, the burst loss ratio (BLR) of each link, the transmission rate of each path, and so on. All this information is updated dynamically by corresponding schemes. For example, the resource of each server can be obtained and updated by the typical registration process in the grid. The network topology can be obtained and updated by routing protocol. The BLR of each link and the transmission rate of each path can be periodically collected by the node connected to the link and sent to the GSN through the control channel. A scheduler module in the GSN is responsible for the task scheduling, that is, when the GSN receives grid job requests, it carries out a resource assignment for each job request according to a certain scheduling strategy. After completing the scheduling, the GSN sends back a response packet to the client through burst control channels, which includes the destination of the remote servers, the network path, etc. After receiving the response packet, the grid client begins to send application data using allocated resources and adopting traditional OBS protocols. The remote servers begin to dispose relevant applications after receiving the data.

The proposed centralized OBS-based grid can be easily implemented based on the traditional OBS network and the grid, since it uses burst control channels to transmit job scheduling information, and the centralized grid scheduling strategy is relatively simple and mature. On the other hand, the centralized grid scheduling strategy can achieve optimal resource assignment by using global information.

3. Scheduling Strategies of the Centralized OBS Grid

In the centralized OBS grid architecture, the scheduling strategy can be considered from the grid level, the network level, and the OBS level (see Fig. 2). The grid level scheduling strategy considers both network resources and resources at each remote server, such as the computational resource and the storage resource. The network level scheduling strategy considers both OBS network resources and access network

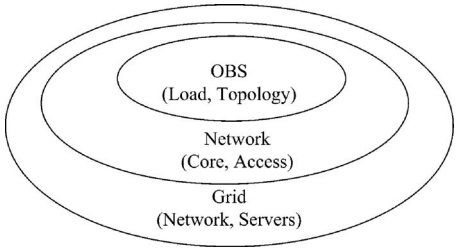

Fig. 2. Three levels in scheduling consideration.

resources. The OBS-level scheduling strategy considers only the OBS network resources such as bandwidth, links, ports, etc.

To optimize the overall grid performance, certain trade-offs in the scheduling process are necessary among various factors such as the remote server capability, the access bandwidth, the OBS path bandwidth, the transmission delay, etc. The best one in each of these aspects may not necessarily be the best choice for the overall performance from the aspect of the grid application level. For example, the best remote server resource assignment would often impose a less-balanced traffic load on the network, which would degrade the overall grid performance. Scheduling in these three levels should adequately represent the relationship between these factors and the overall grid performance.

3.A. OBS-Level Scheduling Strategy

The first step of this strategy is to find all available remote servers according to the application request, and then select the destination from all available remote servers by choosing the OBS path from the edge node that connects the source client to the edge node that connects the remote servers. The OBS path can be selected according to different criterions such as reliability or shortest path. After the OBS path is determined, the server with the least computational load connected to the destined edge node is chosen.

For the reliability criterion, the OBS path is selected according to the overall BLR of paths between two edges. The overall BLR of an OBS path can be calculated from the BLR of each link in the path:

$$\mathrm{BLR}_{\mathrm{overall}} = 1 - \prod_{i=1}^{N} (1 - \mathrm{BLR}_i),$$

where N is the link number of the OBS path, and BLR_i is the burst loss ratio of link i in the OBS path. The BLR of each link can be obtained by the node connected to the link and is periodically sent to the scheduler in GSN through the control channel. As for the shortest-path criterion, the link cost functions can be hops of the path from edge to edge, or the distance of the path.

3.B. Network-Level Scheduling Strategy

The network-level scheduling strategy takes the state of the entire network resource into account. A possible implementation scheme is as follows. The first step is to find all available servers. The second step is to find the bottleneck by comparing the transmission rate of the shortest OBS path with that of the access path, which is the path from the host node to the relevant edge node inside the access network, for each of the available servers. If the bottleneck is in the access path, the server and the grid edge nodes at the access path with the maximum transmission rate would be selected, and then the corresponding shortest OBS path connected to the grid edge nodes is chosen. If the bottleneck is in the OBS network, the OBS path with the maximum transmission rate is selected first, and then the server with the maximum access rate among all available servers connected to the relevant OBS edge is selected. The average transmission rate of a path can be obtained by monitoring the transmitted data size in a previous period at the destination node of the path.

3.C. Grid-Level Scheduling Strategy

In this scheduling strategy, both network resources and computational resources at the servers are taken into account. Here we take total delay as the evaluation factor for the grid. The first step is to find all available servers according to the request. Then estimate the total delay for each available server destination, which include the overall transmission delay in the network and the processing delay at the relative server. The overall transmission delay is calculated by the required job size and the corresponding average transmission rate of the path. The average transmission rate of a path is the average throughput of the corresponding path [8], which can be obtained at the destination node of the path by monitoring the transmitted data size in a previous period. It is periodically sent to the GSN through the control channel. Finally, the server with the least estimated total delay is assigned to the request.

4. Simulation Results and Discussion

The performance of the above three scheduling strategies is evaluated and compared by simulation. The network topology used in simulation, as shown in Fig. 3, is similar to the Shanghai Education and Research Network (SHERNET). It consists of 12 edge nodes, 12 core nodes, and 48 host nodes (32 client hosts and 16 server hosts) located at different campuses. The core node of Shanghai Jiao Tong University (SJTU) is selected as the central scheduling site, since the core node at SJTU is the key central node for SHERNET. All hosts connecting to the four major nodes (SJTU, U_4, U_5, and U_8) serve as the grid resource hosts (remote servers), while the rest of the hosts function as the grid client hosts. The computational capacity of each grid resource host is 10^9 instructions per seconds. A transmission-control-protocol (TCP) based interactive application is adopted in the client hosts, where the first job request is generated and sent to the grid OBS network using the TCP protocol at the beginning. A new job request is sent with an exponentially distributed interval time after receiving the processed results of the former job. The mean interval time after receiving the reply is set to be 30 ms, which is estimated according to the typical arrival interval of the job for the interactive application [9] and the configuration in our simulation scene. The data size and the required calculated amount of each job are also exponentially distributed. The average data size is 2 Mbytes.

Figure 4 shows the average processing time at the remote servers, the throughput, and the average overall delay as a function of the application number for OBS-level scheduling strategy with different OBS path selection criteria when the access bandwidth is set to be 2.5 Gbit/s. The link cost function of the shortest-path criterion is hops of the path from edge to edge in our simulation. The average overall delay is the average time needed to complete a job from the start time of sending the request to the end time of receiving the processing results, which is used to evaluate the performance of the entire grid to a certain extent. Two average required calculated amounts for each application, namely, 10 MI (10^6 instructions) and 40 MI, are adopted, where 10 MI represent the case where the processing time at the server is much shorter than the data transmission delay, and 40 MI represent the case where the processing time at the server is much larger than the data transmission delay. We find that for the average required calculated amount of 10 MI, the shortest-path criterion performs slightly better than the reliability criterion when the application number is relatively low. This is because the network under the low application number has a very low BLR, thus the reliability selection based on the BLR has less effect. The shortest-path criterion, however, can save network resources and propagation time. As the application number increases, BLR increases and becomes a significant factor. So the reliability criterion will get higher throughput, since it takes BLR into account and avoids the congestion path [see Fig. 4(b)]. At the same time, the average job processing time of the reliability criterion at the remote servers is also reduced [see Fig. 4(a)] by avoiding the assignment of jobs to some fixed remote servers connected to the shortest path. However, the reliability criterion will again become less effective when the

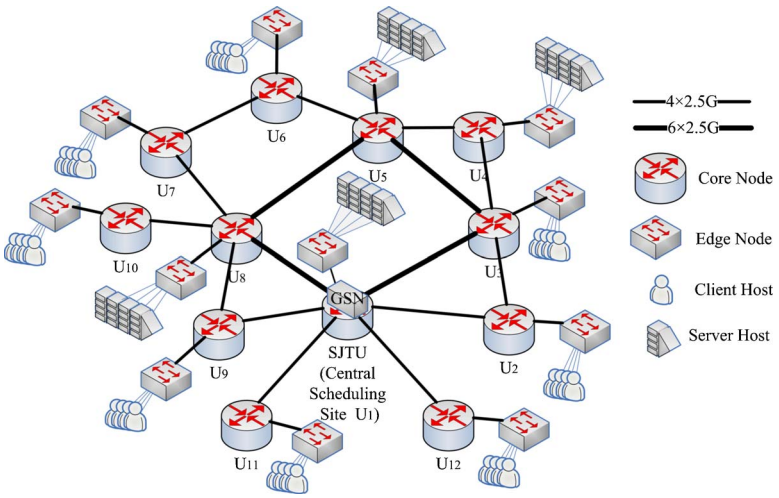

Fig. 3. Simulated grid OBS network.

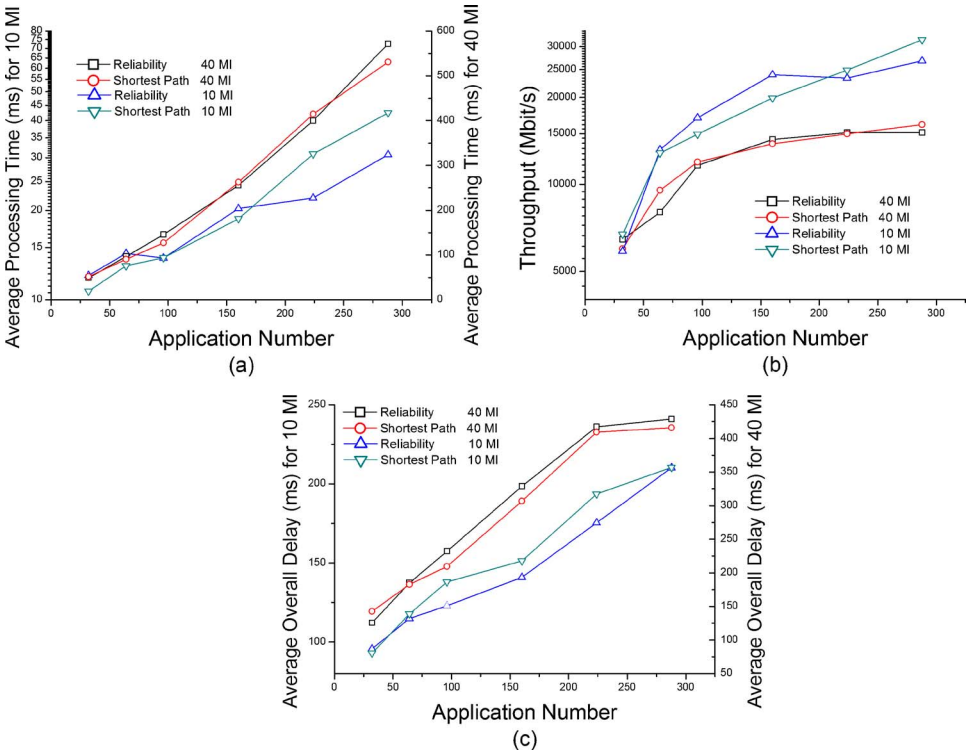

Fig. 4. Effect of application number on the performance of (a) average processing time at remote servers, (b) throughput, and (c) average overall delay.

application number becomes very large. It makes no sense to seek a less-blocked path when all the paths in the OBS networks are severely blocked. On the other hand, the shortest-path criterion becomes a wiser selection, since it minimizes the entire network traffic load by minimizing the propagation delay. In short, the reliability criterion generally performs better than the shortest-path criterion when the load is moderate [see Fig. 4(c)].

From Fig. 4, we find that average required calculated amount has an obvious effect on the performances. This is reasonable since a shorter processing time means the job can be processed and replied to faster in the server, which will lead to more job requests and a heavier traffic load related to the number of job requests. We find that the network throughput and the average processing time of the two path selection criteria are very close to each other when the average required calculated amount is 40 MI. The reason is that the bottleneck of the grid lies in computational resources (i.e., in the remote servers) in this case, and the effect of path selection criterions no longer becomes determinant. The average overall delay of the shortest path criterion when the average required calculated amount is 40 MI is lower than that of the reliability criterion [Fig. 4(c)], since the propagation delay of the shortest path is less and the effect of the load balance is unimportant for the network with low throughput.

Figure 5 shows the dependence of the finished job number, the average overall delay, the throughput, and BLR on the access bandwidth when the application number is 160. The average required calculated amount of each job is set to be 20 MI, and the reliability criterion is adopted for the OBS-level scheduling strategy. From Fig. 5, we find that an optimal access bandwidth exists (1.2 Gbit/s in our simulated environment) under the three scheduling strategies, where the throughput and the finished job number are the maximum and the average overall delay is the shortest. The reason is explained as follows. When the access bandwidth is very low, the traffic is blocked on the access link, and the overall network throughput is limited. As the access bandwidth increases, the overall network performance improves. However, when the access bandwidth exceeds the optimal value, the traffic from the hosts is sent to the OBS network at such a high rate through the access path that it will increase the bursty of input traffic of the OBS network. The increase of the traffic bursty will cause more burst loss, and the overall network performance is degraded.

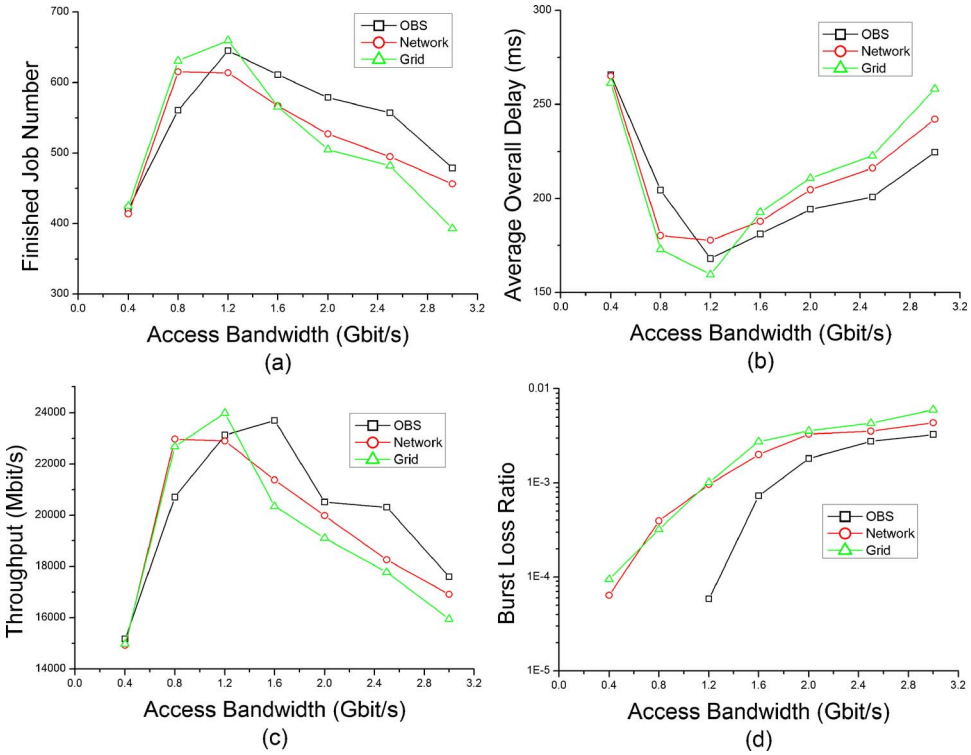

Fig. 5. Dependence of (a) finished job number, (b) average overall delay, (c) throughput, and (d) BLR on access bandwidth.

The optimal access network bandwidth depends on many factors, such as the network topology, the imposed traffic characteristics, the bandwidth of the OBS network, and so on.

From Fig. 5, we also see that the OBS strategy has the best performances and the performance of the grid strategy deteriorates faster than that of the two other strategies after the optimal point. This is because the burst loss becomes the major reason affecting the performance after the optimal point, and its effect increases along with the increase of the access bandwidth. In this case, the OBS strategy performs best, since it always selects the OBS path with the least BLR; thus it is less sensitive to the burst loss caused by the increase of the access bandwidth. On the other hand, the grid strategy select paths according to the overall delay, which is decided not only by the overall transmission delay (which includes the access network and the OBS network) but also the processing delay at the server. The OBS path with the higher BLR may be selected by the grid strategy. So the performance of the grid strategy will degrade sharply accordingly with the increase of the access bandwidth when the burst loss becomes the determinant factor. The network strategy has performance between the OBS strategy and the grid strategy, since it considers the overall transmission delay without the processing delay.

Figure 6 shows the average overall delay, the BLR, the throughput, and the average processing time at the remote servers as a function of the application number under the three level strategies. The average required calculated amount of each application is set to be 20 MI, and the access bandwidth is 2.5 Gbit/s. From the results we can see that the average overall delay, the BLR, the throughput, and the average processing time all increase with the application number for the three strategies. This is reasonable and easy to understand, since the number of requested jobs increases with the application number. Figure 6(a) shows that the average overall delay of the three strategies is similar. The reason is as follows. When the application number is small, the transmission delay to the different server destinations and the processing time of the jobs under all three strategies are close to those of the best situation. As the application number increases, the transmission delay of the grid strategy will become longer than that of the network strategy and the OBS strategy. However, the processing time of the grid strategy will become smaller than that of the two other strategies at the same time, as shown in Fig. 6(d), since the computational load balance is

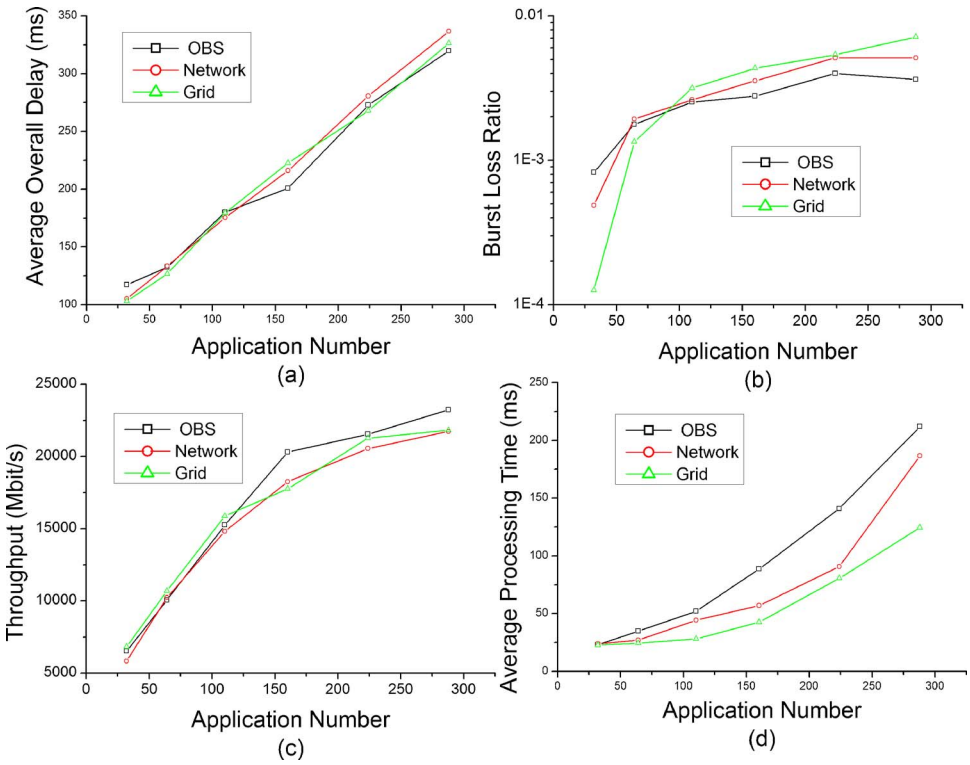

Fig. 6. Dependence of (a) average overall delay, (b) BLR (c) throughput, and (d) average processing time at the remote servers on the application number.

improved under the grid strategies. As for the OBS strategy, its average processing time at the remote server is worse than that of the two other strategies as shown in Fig. 6(d), while its transmission delay is remarkably shorter by selecting the reliable path, i.e., least burst loss [see Fig. 6(b)]. As a result, the average overall delay of the three strategies is also close to each other at a larger application number, since the difference of the transmission delay between the different strategies and that of the corresponding processing times are counteracted by each other.

5. Conclusion

In this paper, a centralized OBS-based grid is proposed, which carries out centralized resource scheduling at a GSN and utilizes the existing control channels in OBS to transmit grid scheduling information. It is relatively easy to implement in the current OBS network, and it can achieve optimal resource assignment. The scheduling strategies in the centralized OBS based grid are studied from the grid level, the network level, and the OBS network level, respectively. Two path selection criterion for the OBS-level scheduling strategy are proposed and analyzed by simulation. The effect of the access bandwidth, the processing time, and the application number on the performance of the entire grid and the network under the three scheduling strategies are also studied and compared. The results show that the reliability criterion for OBS scheduling strategy performs better than the shortest criteria when the application number is moderate and the required calculated amount is relatively low. There is an optimal access bandwidth corresponding to certain configurations of the OBS-based grid for the three scheduling strategies. OBS strategy has the best network performance in BLRs whereas the grid strategy has the shortest average processing time.

Acknowledgment

The paper is partially supported by National Science Foundation of China (NSFC) (ID90704002), 863 project (ID2006AA01Z242 and 2007AA01Z275), Dawn Program for Excellent Scholars by the Shanghai Municipal Education Commission, and the Key Disciplinary Development Program of Shanghai (T0102).

References

1. D. Simeonidou and R. Nejabati (ed.), B. St. Arnaud, M. Beck, P. Clarke, D. B. Hoang, D. Hutchison, G. Karmous-Edwards, T. Lavian, J. Leigh, J. Mambretti, V. Sander, J. Strand, and F. Travostino, "Optical network infrastructure for grid," Grid Forum Draft I.036, http://www.ogf.org/documents.GFD.36.pdf.
2. C. Qiao and M. Yoo, "Optical burst switching (OBS)—a new paradigm for an optical internet," J. High Speed Networks **8**, 69–84 (1999).
3. D. Simeonidou, R. Nejabati, M. J. O'Mahony, A. Tzanakaki, and I. Tomkos, "An optical network infrastructure suitable for global grid computing," presented at the TERENA Networking Conference 2004, session "Grid," Rhodes, Greece, 7–10 June 2004.
4. M. De Leenheer, E. Van Breusegem, P. Thysebaert, B. Volckaert, SiF. De Turck, B. Dhoedt, P. Demeester, D. Simeonidou, M. J. O'Mahoney, R. Nejabati, A. Tzanakaki, and I. Tomkos, "An OBS based grid architecture," in the *IEEE Global Telecommunication Conference (GLOBECOM) Workshop on High-Performance Global Grid Networks* (IEEE, 2004), pp. 390–394.
5. M. De Leenheer, F. Farahmand, K. Lu, T. Zhang, P. Thysebaert, B. Volckaert, F. De Turck, B. Dhoedt, P. Demeester, and J. P. Jue, "Anycast algorithms supporting optical burst switched grid networks," in *Proceedings of International Conference on Networking and Services (ICNS 2006)* (IEEE, 2006), pp. 63–63.
6. D. Simeonidou, R. Nejabati, G. Zervas, D. Klonidis, A. Tzanakaki, and M. J. O'Mahony, "Dynamic optical network architectures and technologies for existing and emerging grid services," J. Lightwave Technol. **23**, 3347–3357 (2005).
7. Q. She, X. Huang, N. Kannasoot, Q. Zhang, and J. P. Jue, "Multi-resource manycast over optical burst switched networks," in *IEEE International Conference on Computer Communications and Networks (ICCCN 2007)* (IEEE, 2007), pp. 222–227.
8. J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP Reno performance: a simple model and its empirical validation," IEEE/ACM Trans. Netw. **8**, 133–145 (2000).
9. E. Weigle and W. Feng, "A case for TCP Vegas in high-performance computational grids," in *9th IEEE International Symposium on High-Performance Distributed Computing (HPDC '01)* (IEEE, 2001), pp. 158–167.