Deflection Routing in Anycast-based OBS Grids

(Invited Paper)

Marc De Leenheer, Jens Buysse, Chris Develder, Bart Dhoedt, Piet Demeester
Dept. of Information Technology, Ghent University - IBBT, Belgium
Email: marc.deleenheer@intec.ugent.be

***Abstract*—Deflection routing is a much-studied contention resolution technique in the context of Optical Burst/Packet Switching networks, as it promises to improve burst blocking performance and may reduce or even eliminate buffer requirements. An OBS-based Grid is frequently based on anycast routing, which holds even greater potential to use the deflection technique for succesful delivery of Grid jobs. As such, we propose an extension to deflection routing whereby jobs are rescheduled, to improve job blocking probabilities under various traffic parameters. We present a case study and perform simulation analysis to demonstrate the effectiveness of our proposal.**

I. Introduction

Grid computing aims to offer a unified interface to access various resources such as computational clusters, data storage sites and scientific instruments. In general, these resources are heterogeneous in nature, are distributed on a global scale and have differing access policies. The main driver to deploy Grid networks are the highly challenging applications which emerge mainly from large-scale, collaborative experiments and the eScience field. Since the datasets involved in such applications pose a challenge to the transport network, photonic networks appear to be the most suitable solution. In particular, Wavelength Division Multiplexing (WDM) allows simultaneous access to multiple wavelengths on a single strand of fiber, and each wavelength offers data rates of 40 Gbps and more. Additionally, optical cross connects (OXC) make it possible to switch these wavelengths over multiple network hops (generally referred to as a lightpath), without costly O/E/O conversions. In this way, an Optical Circuit Switched (OCS) network is created, which can make efficient use of available bandwidth as long as data traffic between end nodes remains high.

However, efficiency drops rapidly in case bandwidth requirements of individual end users decrease [1]. This is frequently the case for applications geared towards enterprise and consumer markets [2]. A potential solution is Optical Burst Switching [3] to access bandwidth on a sub-wavelength scale; as such statistical multiplexing of several data transfers (called bursts) is possible on a single wavelength. This approach could prove essential in the realization of true global-scale Grid computing, where a very diverse set of applications is supported on a single, common data plane.

The initial proposal for OBS has quickly gained attention in research communities and has delivered a number of theoretical studies on performance evaluation and fairness [4]–[6]. The first accurate model to evaluate the blocking behaviour of OBS networks, appeared in [7]. These works focused almost exclusively on the OBS technology in itself, without incorporating Grid-related concepts. In contrast, several papers have addressed the role of introducing network awareness in Grid scheduling algorithms [8], [9]. These clearly demonstrated the need for an integrated approach (i.e. network and end resources) for optimal job scheduling in Grid networks, even though no specific attention was given to Grids based on an OBS network.

These considerations have lead to the introduction of *anycast routing* in OBS-based Grids [10], [11], as these alleviate users from the challenging problem of finding suitable network and Grid resources for a given task. By using an anycast address, service providers can offer a generic interface to end users for a wide range of services and applications. Moreover, advanced scheduling and traffic engineering can be incorporated on a global scale, and as such desirable features such as load balancing or congestion control can be implemented.

In this paper, we extend the well-known contention resolution technique referred to as *deflection routing* in the context of an anycast-based OBS Grid network. Since the general idea of anycast routing is that the destination is not fixed, we can redirect a contending job burst towards the most suitable resources in the network.

The remainder of this paper is structured as follows. In Section II, we discuss the general concept of deflection routing, and present its applicability in the context of OBS-based Grid networks. Then, we present simulation results in Section III, in which we demonstrate that random deflection algorithm combined with rescheduling of jobs can indeed improve the blocking performance of an OBS-based Grid. Finally, our conclusions are formulated in Section IV.

II. Deflection Routing

A. Contention Resolution

Recall that OBS does not reserve the complete path before data transmission; instead a header (BCP or Burst Control Packet) is sent out-of-band to reserve bandwidth on all intermediate links. The header is closely followed by the actual data burst; the only requirement on the offset time between header and burst is that the burst can not arrive at a node before the header[1]. The fact that bandwidth reservations are performed during data transfer, implies that on arrival at a certain router,

[1]Note that the header is processed at each routing node, which requires a small but non-negligable timeframe.

Digital Object Identifier: 10.4108/ICST.BROADNETS2009.7831
http://dx.doi.org/10.4108/ICST.BROADNETS2009.7831

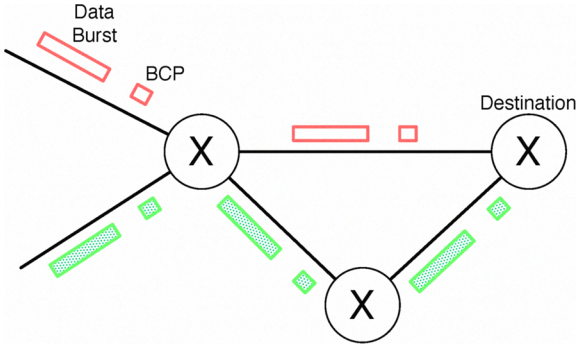

Fig. 1. Burst contention on the top link: deflection allows the bottom burst to reach its destination over an alternate path.

the requested link may already be in use by another burst. In this case, *contention* has occurred, and normally the burst would be dropped. However, several contention resolution techniques have been proposed to ensure the contending bursts can still be saved [12]. These techniques take place in the wavelength domain (wavelength conversion), time domain (buffering), or space domain (deflection routing). In this paper, we consider both the wavelength and time domain solutions as impractical due to their high costs and technological issues. However, deflection routing is a fairly straightforward technique that forwards the contending burst to another output link (thus changing the original route) as shown in Figure 1. Even though deflection routing is known to create unstable network behaviour under certain circumstances[2] [13], [14], the technique does have a number of attractive properties. These include improved burst blocking performance, and reduction or even elimination of buffering requirements (the latter is mostly relevant for packet-based networks).

B. Anycast-based Grids

We can exploit this mechanism even further in the context of an anycast-routed Grid network. This scenario assumes a single Grid job is embedded in one optical burst, and jobs can be processed at one of many resource sites. As such, depending on network and/or resource state, we can deflect jobs towards sections of the network where resource capacity is most likely to be found. The fundamental issue of a burst deflection algorithm is thus determining where to deflect the burst to [10].

Random deflection: In this case, upon contention, the burst is deflected to an available randomly selected egress port. This scheme is similar to the hot potato protocol in the sense that the node forwards the burst to the first available channel on any randomly selected egress port.

Network-based deflection: The port selection at a node is based on some network-related parameters. These include, for instance, the port's blocking probability, whether the port is on an alternative shortest path to the original destination, etc. The examples cited are all based on information which can be assumed to be locally available at the router.

[2]due to the increased network load that is generated

Grid-based deflection: In this case, the node examines all Grid resources throughout the network. Then, the node decides which egress port should be selected in order to forward the contending burst. Typical parameters include the current job load, processing speed, memory capacity, etc. Furthermore, the weight can be shifted in favor of ports that provide e.g. alternative shortest paths to the original destination node, thus including network-related parameters as well. Observe that this approach requires that each network node has up-to-date knowledge of the state of all resources. This can only be guaranteed in case accurate resource states are disseminated throughout the network. As discussed in [2], using the job response burst is one potential way to update the resource state information.

C. Rescheduling

In this paper, we propose to introduce rescheduling when using deflection routing in an anycast-based Grid network. In a Grid scenario, the essential insight is that the succesful processing of a job is more important than the precise location where this processing takes place. Although a job is assigned a destination at its origin, we can still change this destination whenever necessary. Since deflection implies that congestion has occurred, this seems an opportune time to reschedule the job, i.e. to assign a new destination to the burst. This idea is reflected in Algorithm 1.

Algorithm 1 Rescheduling algorithm for incoming burst

```
portList = interfaceList(currentRouter)
portList ← portList \ {incomingPort}
while portList ≠ ∅ do
  port ← route(burstDestination)
  if isAvailable(port) then
    send()
  else
    portList ← portList \ {port}
    burstDestination = reschedule()
  end if
end while
drop()
```

We will illustrate in the following paragraphs that even the most basic deflection algorithm, i.e. random, can indeed improve performance of an OBS-based Grid. Furthermore, since our interest lies in providing anycast routing services, we also evaluate the case where jobs are given a new destination after deflection.

III. Simulation Analysis

In this section, we present simulation results obtained by implementing a random deflection algorithm. We consider the basic European network (Figure 2), with all links carrying $W = 20$ wavelengths operating at bandwidth B, and implemented Horizon [15] as wavelength reservation algorithm. Furthermore, 5 fixed nodes are chosen to act as resource (Athens, Warsaw, Milan, Paris and Munich), and 5 fixed

Digital Object Identifier: 10.4108/ICST.BROADNETS2009.7831
http://dx.doi.org/10.4108/ICST.BROADNETS2009.7831

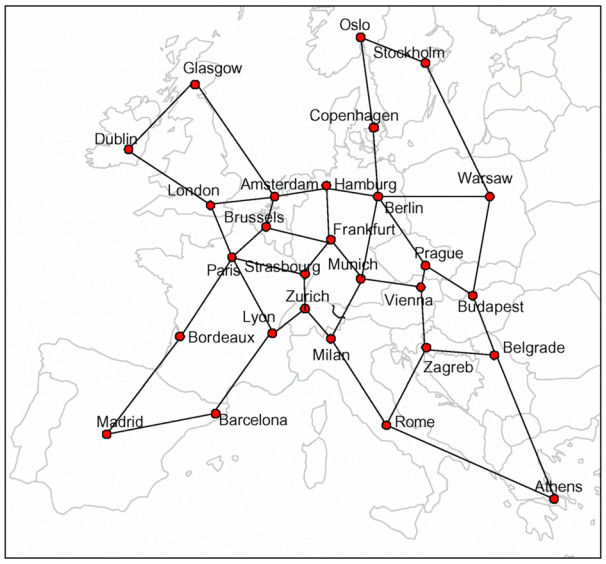

Fig. 2. Simulation topology: basic European network (28 nodes, 41 bidirectional links)

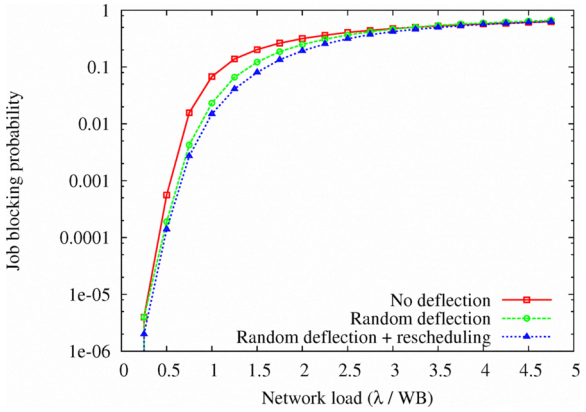

Fig. 3. Deflection routing reduces job blocking probability, with further improvements through rescheduling (20% average generated resource load)

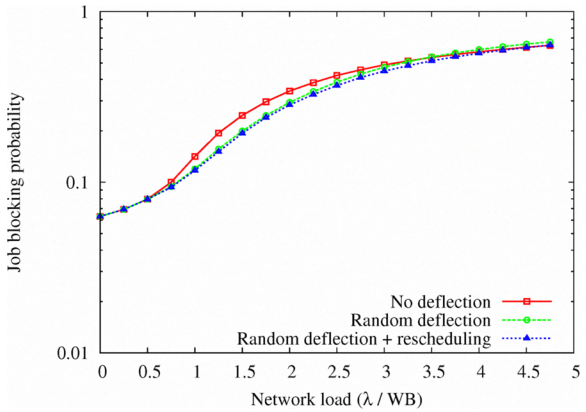

Fig. 4. Deflection routing causes insignificant decrease in job blocking probability (80% average generated resource load)

nodes are chosen as client (Glasgow, Oslo, Zagreb, Barcelona and Brussels). Each resource can process at most 20 jobs in parallel, and Poisson job arrivals are generated at each client (average arrival rate λ). Jobs have exponentially distributed data size and processing times, each with a varying average to establish the resource and network load. Three client nodes have a node degree of 2, while the remaining two have degree 3, which implies the client links experience 100% load when $\frac{\lambda}{WB} = 3$ (see Figures 3, 4 and 5). The burst offset times are a factor 1000 smaller than the actual data burst's transmission time.

Jobs are scheduled at the client side according to a weighted function which incorporates the amount of free capacity (the number of non-occupied job slots), i.e. the probability that resource r is chosen is $P_r = \frac{C_r}{\sum_u C_u}$, with C_r representing the number of free slots at resource r. Shortest path routing is used at all times, and in case of contention the job is dropped when deflection is not enabled. The implemented deflection algorithm randomly selects an alternate output port, and is evaluated both with and without rescheduling. In case of rescheduling, a new resource is chosen according to the same weighted function used at the client. We focus on two performance metrics as the network load varies: the job blocking probability and average job hop count of jobs that were succesfully executed.

Figure 3 shows the blocking probability of the different deflection mechanisms for a 20% resource load. Our results indicate that deflection routing can indeed improve job acceptance, by using alternate network paths to reach the assigned resource. It is also clear from the figure that after deflection, rescheduling can further increase the Grid's performance. Note that high network loads limit the number of available ports when contention occurs, ultimately resulting in identical performance for all approaches. However, as shown in Figure 4, the advantages of deflection routing are much smaller when high resource loads (80%) are generated, which implies that deflection should mainly be used to overcome network-related blocking events. The minimal effect of rescheduling, shown in the same graph, further supports this conclusion.

A drawback of deflection is that it increases the average hop count (and thus, although not shown, the network utilization as well), as shown in Figure 5. Note that the average hop count is calculated from succesfully completed jobs only; blocked jobs are not taken into consideration. This result is identical for both 20% and 80% generated resource loads, as the network forms the main bottleneck in this scenario. Observe also that rescheduling after deflection can reduce the higher hopcount.

As mentioned earlier, the performance results shown are for the most basic deflection technique (random); we have studied more advanced deflection algorithms in [10], but do not repeat the results here. Irrespective of the exact deflection algorithm, it is important to remark the restricted environment in which these algorithms are executed. Processing time is limited because the offset time must be respected, and memory capacity is constrained to maintain router scalability. As such, practical deflection schemes should remain relatively simple

Digital Object Identifier: 10.4108/ICST.BROADNETS2009.7831
http://dx.doi.org/10.4108/ICST.BROADNETS2009.7831

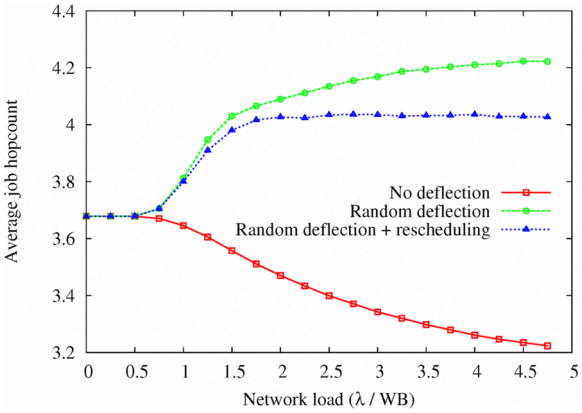

Fig. 5. Deflection routing increases the average hop count

in terms of both computational and memory requirements.

IV. Conclusions

We have demonstrated the potential performance improvements of an OBS-based Grid, by implementing deflection mechanisms with and without rescheduling. Even though the evaluated algorithms are fairly simple (random deflection, destination assignment based on highest free capacity), results indicate that this approach is both valid and practical. We also demonstrated that deflection, when used in an OBS-based Grid scenario, should mainly be used to overcome network-related blocking events, instead of blocking caused by the lack of resources.

Acknowledgment

The work described in this paper was carried out with the support of the BONE-project (Building the Future Optical Network in Europe), a Network of Excellence funded by the European Commission through the 7th ICT-Framework Programme, as well as the IST Phosphorus-project. J. Buysse is funded by the IWT through a Ph.D. grant, and C. Develder is supported by the FWO through a post-doc grant.

References

[1] F. Xue, S.J.B. Yoo, H. Yokoyama, and Y. Hoiuchi, *Performance Comparison of Optical Burst and Circuit Switched Networks*, Proc. of the Optical Fiber Communication Conference (OFC), Anaheim, CA, USA, March 2005

[2] M. De Leenheer, P. Thysebaert, B. Volckaert, F. De Turck, B. Dhoedt, P. Demeester, D. Simeonidou, R. Nejabati, G. Zervas, D. Klonidis, and M.J. O'Mahony, *A View on Enabling Consumer Oriented Grids through Optical Burst Switching*, IEEE Communications Magazine, 44(3):124–131, March 2006

[3] C. Qiao, and M. Yoo, *Optical Burst Switching: A New Paradigm for an Optical Internet*, Journal of High Speed Networks, 8(1):69-84, Mar 1999.

[4] Q. Zhang, V.M. Vokkarane, J.P. Jue, and B. Chen, *Absolute QoS Differentiation in Optical Burst-Switched Networks*, IEEE Journal on Selected Areas in Communications, 22(9):1781-1795, Nov 2004.

[5] M. Düser, and P. Bayvel, *Analysis of a Dynamically Wavelength-Routed Optical Burst Switched Network Architecture*, Journal of Lightwave Technology, 20(4):574-585, Apr 2002.

[6] N. Barakat, E.H. Sargent, *Separating Resource Reservations from Service Requests to Improve the Performance of Optical Burst-Switching Networks*, IEEE Journal on Selected Areas in Communications, 24(4):95–107, April 2006

[7] Z. Rosberg, H.L. Vu, M. Zukerman, J. White, *Blocking Probabilities of Optical Burst Switching Networks Based on Reduced Load Fixed Point Approximations*, Proc. 22nd Annual Joint Conference of the IEEE Computer and Communications Societies (Infocom), volume 3, pp. 2008-2018, March 2003

[8] P. Thysebaert, B. Volckaert, F. De Turck, B. Dhoedt, P. Demeester, *Network Aspects of Grid Scheduling Algorithms*, Proc. ISCA 17th International Conference on Parallel and Distributed Computing Systems (PDCS), September 2004

[9] K. Ranganathan and I. Foster, *Simulation Studies of Computation and Data Scheduling Algorithms for Data Grids*, Journal of Grid Computing, 1(1):53–62, March 2003

[10] M. De Leenheer, F. Farahmand, K. Lu, T. Zhang, P. Thysebaert, B. Volckaert, F. De Turck, B. Dhoedt, P. Demeester, J. Jue, *Anycast Algorithms Supporting Optical Burst Switched Grid Networks*, Proc. International Conference on Networking and Services (ICNS), pages 63–68, July 2006

[11] E. Zegura, M. Ammar, Z Fei, S. Bhattacharjee, *Application-Layer Anycasting: A Server Selection Architecture and Use in a Replicated Web Service*, IEEE/ACM Transactions on Networking, 8(4):455–466, August 2000

[12] A. Zalesky, H.L. Vu, Z. Rosberg, E.W.M. Wong, M. Zukerman, *OBS contention resolution performance*, Performance Evaluation, 64(4):357–373, May 2007

[13] F.P. Kelly, *Blocking Probabilities in Large Circuit-Switched Networks*, Advances in Applied Probability, volume 18, pages 473–505, 1986

[14] R.S. Krupp, *Stabilization of Alternate Routing Networks*, Proc. IEEE International Conference on Communications (ICC), volume 31, pages 1–5, June 1982

[15] J. Teng and G.N. Rouskas, *A Comparison of the JIT, JET, and Horizon Wavelength Reservation Schemes on a Single OBS Node*, Proc. 1st Workshop on Optical Burst Switching, October 2003

Digital Object Identifier: 10.4108/ICST.BROADNETS2009.7831
http://dx.doi.org/10.4108/ICST.BROADNETS2009.7831