Modeling novel back-pressure mechanisms for a 100 Gb/s switch

Fagertun, Anna Manolova; Ruepp, Sarah Renée

Published in:
Proceedings of OPNETWORK2012

Publication date:
2012

Modeling novel back-pressure mechanisms for a 100 Gb/s switch

Anna Manolova Fagertun, Sarah Ruepp
DTU Fotonik, Department of Photonics Engineering
2800 Kongens Lyngby, Denmark
E-mail: anva@fotonik.dtu.dk

Abstract
In this work we evaluate the performance of novel back-pressure mechanisms in a Clos-based 100 Gb/s switch system via OPNET modeler simulations. The effectiveness of the mechanisms under different switch configurations, as well as under different traffic patterns, is presented. Our results indicate that the proposed back-pressure techniques can effectively reduce the requirements for buffer space in the different stages of the Clos switch.

Introduction
In the recent years, the emergence of 100 Gb/s switching technology has attracted the attention of all players in the telecommunication industry. The growing need for support of highly-demanding multimedia services all the way to the end-user stretches the capabilities of the existing access, metro and core networks and necessitates new technologies for high-capacity data delivery. The first standards for 100 Gb/s interfaces and network elements are currently being tested and deployed in the production networks of many network operators.

Operating flexibly and efficiently high-port, high-speed data switches with 100 Gb/s speeds per port poses immense challenges for the switch architecture in many aspects such as scheduling, traffic management, memory organization, switch fabric architecture, space and power consumption etc. [1]. The aggregated switch speed can easily amount to several Terabits per second and standard single-stage switches cannot handle effectively and in a scalable manner such loads. Thus, multi-stage switch architectures have been proposed as a promising solution for the next-generation 100 Gb/s switch fabrics. In particular, Clos-based switches have been intensely evaluated for scalability and efficiency under diverse scenarios (including 100 Gb/s applications) [2-4]. This switch architecture provides a unique path through the switch for every crossing data flow and is thus a good candidate for building a scalable and flexible switching node.

In this work, we focus on Gigabit Ethernet (GE) switching as a solution for the metro/access segment of the network. Under this application scenario, it is custom to apply traffic shaping and policing at the edges of the network for access control and traffic flow management, i.e., it can be assumed that some level of aggregation of the traffic flow has been achieved and the traffic pattern is not very bursty. Nevertheless, traffic irregularities are still possible and thus, the issue of handling temporal burstiness in the traffic flow is important for providing high quality of service and for supporting optimal network operation and utilization. Different flow control mechanisms can be applied in a network: between network nodes and within a network node. Short-term traffic irregularities are more effectively handled within the switching node. Two types of flow control can be applied in a node: internal (link-level), between the individual modules of a switch; and an end-to-end, between the input and the output traffic managers [5]. The simplest flow-control is the so-called Back Pressure (BP) where a downstream queue sends a one-bit signal to an upstream queue indicating overflow (bit 1) or operational (bit 0) condition [5-7]. Credit-based flow control is a more sophisticated mechanism where the downstream queue grants credits to the upstream queue, indicating how much traffic it can send [8, 9]. Though very highly-efficient, such scheme can suffer scalability issues in terabit/s operational conditions.

In this paper we focus on two aspects of a Clos-based 100 Gb/s switch node: the efficiency of different internal switch architectures and the efficiency of novel BP mechanisms for buffer-flow control. In the first part, we look into the internal organization of the Clos-based switch and focus on the imposed requirements for the buffer space in order to achieve non-blocking operation under 0.95% input load. In the second part, we present several back-pressure mechanisms for temporal flow-control under short-term bursty conditions and evaluate their efficiency in reducing the amount of needed buffer space for two of the presented internal organizations. Initial evaluation of the efficiency of the BP schemes has been done in [10] for a small 9x9 switch.

Clos-based switch fabric – architecture and flow control
Clos-based switches have 3 stages (see Fig. 1), which can be either buffered or bufferless. A bufferless stage is simply a space-switch which forwards a cell/packet directly from its input to its output within one time-slot of the operation of the switch. Depending on the combination of buffered and buffereless stages, several different types of switches are possible [10]. In this work we use a Space-Memory-Memory (SMM) architecture [2], i.e., the Input Module (IM) is a space switch, whereas the Central Module (CM) and the Output Module (OM) are buffers, which can be organized in different ways. For our work, we employ Virtual Output Queuing (VOQ) as an effective solution which avoids Head of Line blocking and provides high throughput. The IM connection matrix can be organized in several ways [4], but in our work we adopt a static configuration for simplicity. The second stage of the switch is referred to as *distribution stage*, and the third stage as *aggregation stage* in this work. Initial performance investigation of the outlined architecture is presented in [3].

In the presented switch architecture applying a standard back-pressure flow control between the CMs and the IMs is impossible, since the IMs are simple space-switches without buffers, i.e., they cannot stop sending traffic in case the downstream buffer gets overloaded. Thus, novel methods for deflecting the traffic flow, which is overloading the buffers in a given CM, must be employed. Standard BP schemes work on a queue-to-queue control principle, whereas our novel BP mechanisms operate on a queue-to-module principle. In particular we propose to use a 1-bit BP signal to enforce connection matrix change in the IM, i.e., when a CM detects a buffer overflow it sends a 1-bit signal to the IM which causes the

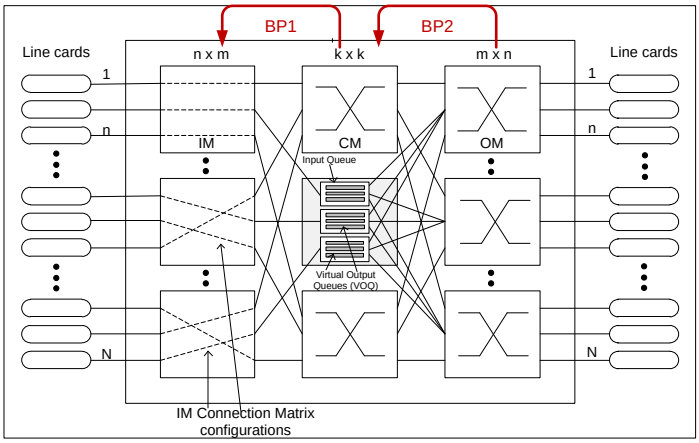

Fig. 1 A 3-stage Clos-based switch with VOQ organization of CMs and OMs and static IM connection matrix configuration. BP1 and BP2 are backpressure signals from CM to IM and from OM to CM respectively.

overflow, effectively enforcing it to change its connection matrix and thus, re-direct the flow which causes overload. The work in [10] presents initial performance evaluation of the following novel BP schemes (referred to as BP1 type in Fig.1):

- **BP_coarse**: a scheme in which a CM detects a total buffer overflow (i.e., the sum of the sizes of all Input queues crosses a threshold) and sends a 1-bit backpressure signal to all IMs connected to it, enforcing them to change their configuration, following a one-time round-robin principle: i.e. a connection i → j is changed to connection i→ j + 1 mod N, where N is the number of outputs from the IM.
- **BP_fine**: a scheme in which a CM detects an overflow on a particular Input queue and sends a 1-bit backpressure signal to the corresponding IM which causes the overflow in order to change its configuration following a one-time round-robin principle.
- **BP_fine_load_deflefct**: a scheme similar to **BP_fine**, but instead of following a one-time round-robin principle for changing the configuration, the IM redirects the flow from its least-loaded input towards the CM which sent the BP signal.
- **BP_fine_load_balance**: a scheme similar to **BP_fine**, but instead of following a one-time round-robin principle for changing the configuration, the IM continuously changes its configuration every time-slot until it receives a 1-bit BP signal from the same CM, indicating that the overloaded Input queue is operational again.

There are two control mechanisms for the proposed BP schemes – timer-based and threshold-based. A BP signal is activated after the length of an Input queue (or the total length of all Input queues as in **BP_coarse**) in a CM crosses a threshold. After an IM receives a BP signal, it is locked for receiving any other BP signals from other CMs. This is needed in order the effect of the BP to stabilize. If a time-based control is employed, an IM unlocks for other BP signals after a predefined period of time called Back-off Time (used for the analysis in [10]). On the other hand, if a threshold-based control is employed, the CM which activated the BP and locked the IM for other BP signals must send a release BP signal, indicating that the IM can receive and react to other BP signals. The threshold values, under which the activation and release BP signals are sent, are an important operational parameter which determines the effectiveness of the applied BP scheme.

A standard back pressure can be employed between the OMs and the CMs since both are buffer-based modules. This BP mechanism (referred to as BP2 on Fig.1) can also have a time-based or a threshold-based control similar to the schemes described earlier. A time-based control is simpler to implement, since no control/check is needed for detecting if a buffer is loaded low enough, but the efficiency of the BP mechanism will depend on the combined effect of the employed timer and the traffic load/pattern in the system. Setting up a suitable timer value is tightly bound to the specific traffic pattern and load in the system. This results in a unique timer value for every simulation run, which is not an optimal operational procedure for performance evaluation via simulations. Thus, a threshold-based control is used for the BP2 mechanism as well.

Traffic models

The initial investigations presented in [3] specify several types of traffic patterns and load distributions. Our work focuses on two specific problems, for which two types of traffic models are needed. For the investigation related to the effect of the internal configuration of the switch on the needed buffer space we employ a *Uniform* traffic distribution (i.e., every input port targets randomly every output port), whereas for the investigation related to the efficiency of the backpressure mechanisms on the needed buffer space we employ *Unbalanced* traffic distribution [3] for introducing short-term burstiness in the flow. In particular, at random intervals of time a source employs an *Unbalanced* traffic distribution, targeting only one destination with 75% of its traffic in order to simulate temporal oversubscription condition. The duration of this *Unbalanced* traffic distribution is an important performance parameter, which influences highly the throughput and the buffer space requirements of the switch.

For both cases we employ Bernoulli traffic generation, which is the standard input traffic type for benchmarking of switch performance according to [11].

OPNET model design and implementation

A switch node model is presented on Fig. 2. Depending on the capacity of the node and the desired number of input/output ports, it is possible to have many different configurations. An input line card (see Fig. 1) is represented by a combination of a source module and an IPP module (see Fig. 2), where the IPP module performs Ethernet frame segmentation into cells of fixed size for forwarding through the system. An output line card is represented by a combination of an OPP module and a sink module, where the OPP module performs the frame reassembly. Since the general model of the Clos-based switch has been already presented in [3], here we present only the enhancements, related to the implementation of the backpressure mechanisms and the specific traffic patterns, needed for our analysis.

Traffic source module

The process model for the traffic source is presented on Fig. 3. For the analysis, related to the performance evaluation of the BP schemes, all sources start with generating *Uniformly* distributed

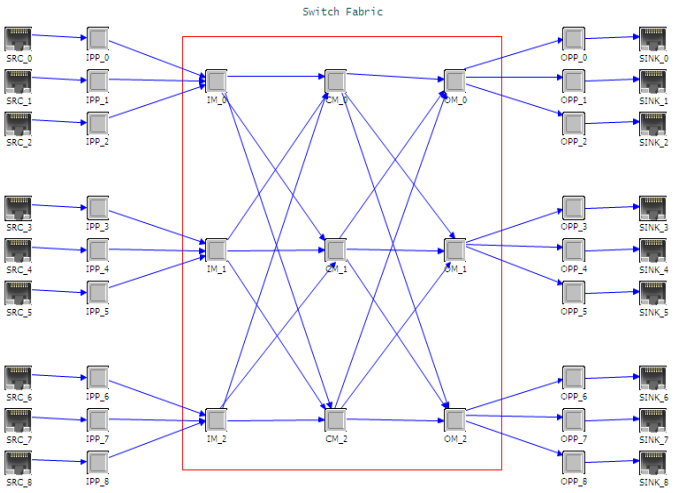

Fig. 2 An example of a 9x9 Clos-based switch fabric.

traffic following a *Bernoulli* generation process. At random periods of time a source independently changes its traffic distribution pattern to *Unbalanced* for an exponentially distributed interval of time with mean value given as an input parameter. For this period of time the source targets 75% of its traffic towards one particular output line card, effectively simulating oversubscription. At the end of the period, the source starts generating uniformly distributed traffic again. This process repeats randomly throughout the simulation. The operation of each source is controlled via the model parameters presented in Tab. 1. The main operational parameters are the *Chance of distribution change* and the *Change duration* parameters which specify the temporal oversubscription duration and intensity.

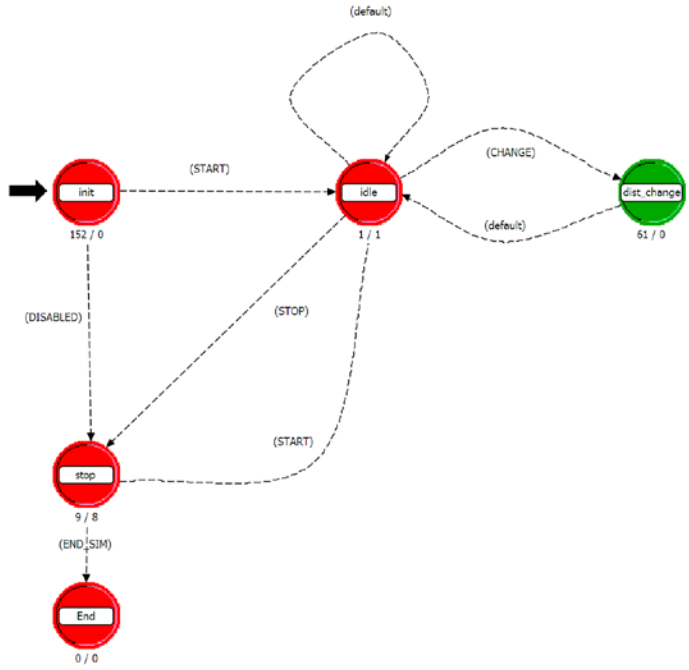

Fig. 3 Source process model.

For the analysis related to the evaluation of the internal switch configuration each source employs uniformly distributed traffic following a Bernoulli generation process for the Ethernet frames for the entire simulation duration.

Parameter	Description
Port rate	Specifies the input data rate of the line-card
Number of sub-sources	Specifies how many sub-flows a line card will simulate
Active source	Specifies if the line-card is active for the simulation run
Packet arrival process	Specifies the traffic type: Bernoulli, Poisson, ON/OFF Pareto
Traffic distribution	Specifies the distribution: Balanced or Unbalanced
Variable packet length	Boolian parameter, specifying if fixed or variable sized Ethernet frames are used
Chance of distribution change	Specifies what is the probability that a line-card will perform Unbalanced traffic distribution
Change duration	Specifies the duration of the Unbalanced traffic distribution (in number of time-slots)

Tab. 1 Source model parameters

Input, Central and Output Modules

The IMs, the CMs and the OMs have a similar process model, presented on Fig. 4. The enhancement from the models presented in [3] is the back-pressure mechanisms, which are implemented via remote interrupts (OM to CM and CM to IM). A predefined amount of cells in every given Input buffer (for both the OMs and the CMs) is given as a simulation parameter and out of it the threshold values for activating and deactivating the BP mechanisms are calculated. The condition for activating a BP mechanism is evaluated at the entrance in the *idle* state and a remote interrupt (representing the back-pressure 1-bit signal) for the upstream module is generated if needed. If a module receives a remote interrupt, it enters in the corresponding *remote-int* state and performs all needed reconfiguration procedures which follow based on the applied BP mechanism. The model parameters presented in Tab. 2 control the application and configuration of the BP mechanisms (separate parameters for BP1 type and BP2 type).

Parameter	Description
Status	Specifies if a BP mechanism is active or not for the simulation run
Type	Specifies a sub-type of the BP mechanism
Upper bound	Specifies at threshold for activating a BP mechanism (in % of the VOQ limit)
Lower bound	Specifies the threshold for de-activating a BP mechanism (in % of the VOQ limit)
VOQ limit	Specifies the threshold from which the boundaries for activating and de-activating BP are calculated

Tab. 2 Back-pressure model parameters (same format for BP1 and BP2).

Simulation setup and results

As indicated earlier, there are two separate analyses we perform in this work. First, we evaluate the differences in the amount of required buffer spaces for the CMs, the OMs and the entire switch when different internal architectures for the Clos-based switch are used. Such analysis is important since Clos-based architectures are very flexible and scalable. Thus, a proper

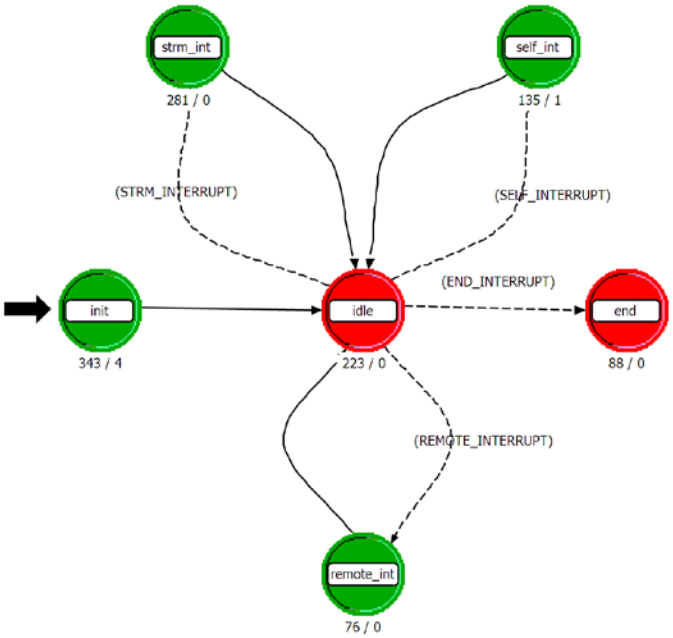

Fig. 4 IM, CM and OM process models.

configuration must be identified so that the switch can be built with as less components as possible, resulting in smaller sizes, lower power consumption and better utilization of the switch. Second, the performance of the described back-pressure mechanisms, in terms of needed buffer space for the CM and the OM modules, is investigated under different amount of over-subscribing users for two of the investigated switch architectures. In both cases the switch splits incoming Ethernet frames into 512-bit cells, which are switched through the system and all links in the system run at 100 Gb/s speed, i.e, there is no speed-up in the system.

Switch architecture analysis

A switch has *k* input modules of size *n* x *m*, *m* central modules of size *k* x *k*, and *k* output modules of size *m* x *n* (see Fig. 1). In total the switch has *N* inputs and *N* outputs, where $N = n \times k$. Given these parameters, a switch can be denoted as *C(n, m, k)*. In the first part of our analysis we evaluate the performance of three different configurations for an *N*=32 port switch (maximum input traffic of 3.2 Tb/s per). In particular we use three different configurations as follows:

- **Switch type 1:** C(8,8,4) where 8 line cards are connected to one IM/OM, there are 8 CMs and 4 IMs/OMs.
- **Switch type 2:** C(4,4,8) where 4 line cards are connected to one IM/OM, there are 4 CMs and 8 IMs/OMs.
- **Switch type 3:** C(4,8,8) where 4 line cards are connected to one IM/OM, there are 8 CMs and 8 IMs/OMs.

The performance metric for this analysis is the maximum needed amount of buffer space per module (CM and OM) and for the entire switch under 95% load in the system, (i.e., each line card generates 95 Gb/s), uniform traffic distribution and Bernoulli traffic generation. Ten independent simulations with different random seeds are performed and 95% confidence intervals are presented.

Tab. 3 presents the maximum amount of needed buffer space per central module and per output module. For **Switch type 3**, the maximum amount of total buffer space per CM is very small, i.e., a CM chip can be of a smaller size. The required maximum amount of buffer space per OM on the other hand is quite high, indicating that bigger OM chips are needed. The same trade-off can be seen for **Switch type 1** as well. In order to compare fairly all three architectures, the maximum total buffer size per stage (i.e., for all CMs and all OMs) and the maximum needed total buffer size per switch (assuming every module needs the indicated maximum buffer size) are presented on Fig. 5. Two things draw attention. First, the more memory is needed for the CMs, the less memory is required for the OMs and vice versa. This indicates that the different architectures simply put the bottleneck at different places of the switch. Second, on average all three switch types require roughly the same amount of total memory for buffers: between 25000 and 30000 cells. For our next analysis we look at the two architectures which require the lowest amount of memory, namely **Switch type 1** and **Switch type 3**. These two also represent two cases at opposite sides of the spectrum: **Switch type 1** requires nearly the same amount of memory for its CMs and its OMs, whereas **Switch type 3** requires a lot of memory for OMs and very little for CMs, i.e., the distribution part of the switch (CMs) is small, whereas the aggregation part (OMs) is large, which effectively places the switch closer to an output buffered architecture.

Switch type	Required memory per CM [in cells]	Required memory per OM [in cells]
Type 1: C(8,8,4)	1748,1 ± 106,22	3097,9 ± 175,91
Type 2: C(4,4,8)	2684,8 ± 114,49	2390,8 ± 191,26
Type 3: C(4,8,8)	424,2 ± 168,9	2937,1 ± 134,55

Tab. 3 Maximum needed total buffer space (in cells) per CM and OM in all switch types.

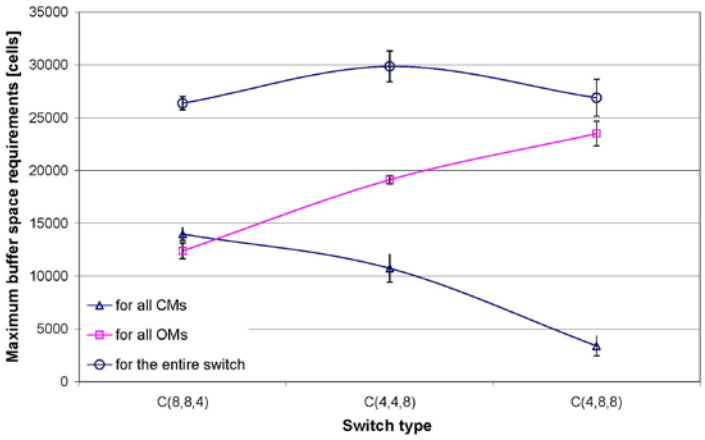

Fig. 5 Maximum needed buffer space (in number of cells) for all modules (per type) and for the entire switch, for 3 different switch configurations.

Back-pressure mechanisms performance

The second part of our analysis is focused on the performance of the proposed backpressure mechanisms. The performance measure is the maximum needed amount of buffer space per input queue in a CM/OM for operation without losses, i.e., we employ endless buffers and observe their maximum length

throughout the simulation. Additionally, we examine the amount of overhead needed for the mechanisms to operate (in number of 1-bit signals sent for activation and de-activation of BP). Three scenarios are compared for **Switch type 1** and **Switch type 3**:

- **Case 1**: no BP mechanisms applied
- **Case 2**: only BP2 mechanism applied (i.e., there is a BP signal from OM input queue to the corresponding overloading CM)
- **Case 3**: BP2 and BP1 mechanisms applied simultaneously. In this case we have 3 sub-cases, applying three of the BP1 mechanisms described earlier:
    - Case 3_1: **BP2** + **BP_fine**
    - Case 3_2: **BP2** + **BP_fine_load_deflect**
    - Case 3_3: **BP2** + **BP_fine_load_balance**

For this analysis, temporary unbalance in the traffic distribution is needed in order to simulate short-term disturbance in the traffic flow (for example oversubscription or improper traffic shaping/policing at the edge of a network). The time-duration during which a source is employing unbalanced traffic distribution is set to 5000 time-slots[1]. Each source has a certain probability of employing unbalanced distribution, which is given as an input parameter (*Chance of distribution change* from Tab.1). Five different values of the parameter are tested: from 0.1 to 0.5, i.e., between 10% and 50% of all sources introduce traffic imbalance. The switch operates at 85% capacity (i.e., every source generates 85 Gb/s traffic).

The threshold values for activating and deactivating the BP mechanisms are 10% and 30% of the predefined VOQ limit respectively (see Tab.2), i.e., when the length of a queue is 10% lower than the VOQ limit, a BP signal is activated; when the length of the queue is 30% lower, the BP signal is de-activated. The VOQ limits are different for the different types of switch architectures and depend on the observed amount of needed maximum buffer space per input queue without BP employed. Tab. 4 presents the used values for the BP operation.

Switch type	Parameter	Value
Type 1	Upper bound	10 %
	Lower bound	30%
	VOQ limit	BP1: 4096 BP2: 4096
Type 3	Upper bound	10%
	Lower bound	30%
	VOQ limit	BP1: 2048 BP2: 4096

Tab. 4 Back-pressure configuration parameters.

Fig. 6 and Fig. 7 present the amount of maximum needed buffer space per input queue at any CM for **Switch type 1** and for **Switch type 3** respectively. The first thing to note is that **Switch type 1** requires almost double the amount of buffer space per Input queue in any CM. This is due to the connectivity between the modules and the internal memory organization of the buffers (VOQ is used for the CMs, see Fig. 1), where in **Switch type 1**

[1] This value is chosen based on numerous simulation trials, where the time needed for a simulation run, the memory consumption of the simulation and the effect of the value on the operation of the switch have been balanced.

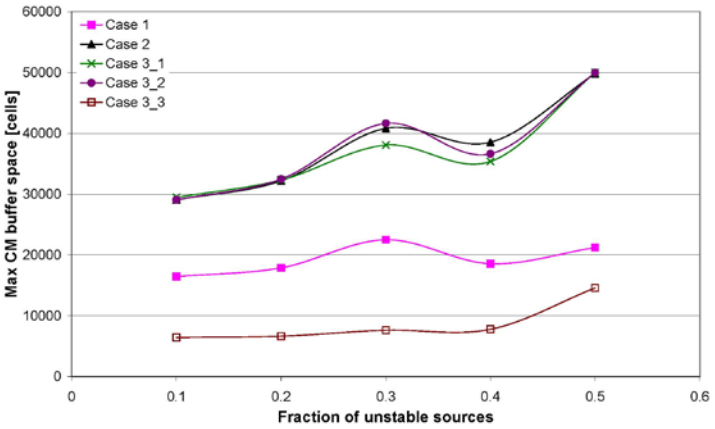

Fig. 6 Maximum needed buffer space per Input queue in a CM for Switch type 1.

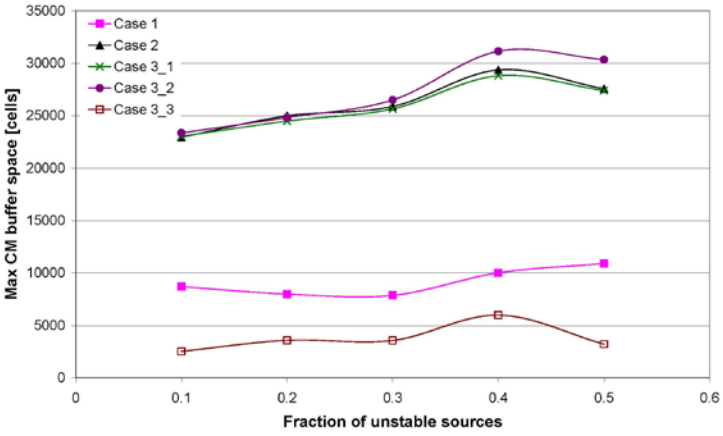

Fig. 7 Maximum needed buffer space per Input queue in a CM for Switch type 3.

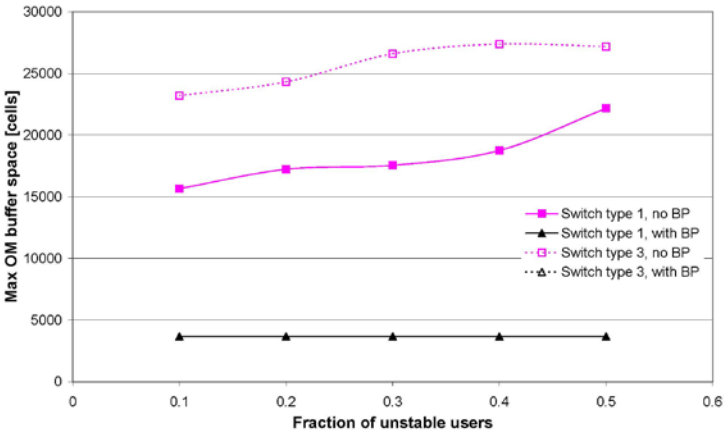

Fig. 8 Maximum needed buffer space per Input queue in an OM for both switch types with and without BP.

there are 4 IMs connected to every CM, whereas in **Switch type 3** there are 8 IMs connected to every CM. Since in both cases the throughput of the system is the same (same load and no losses) then the required queue length per input queue in a given CM will be less, when we have more input queues, i.e., when we apply **Switch type 3**. Furthermore, it is interesting to observe that applying BP2 in fact increases the required amount of buffer

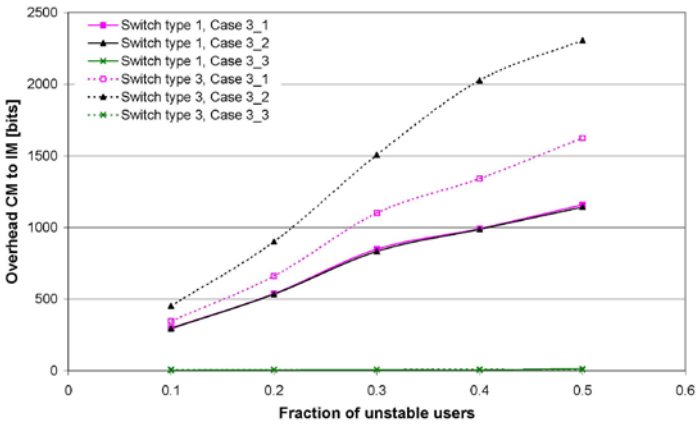

Fig. 9 Overhead for control of BP1 schemes (CM to IM).

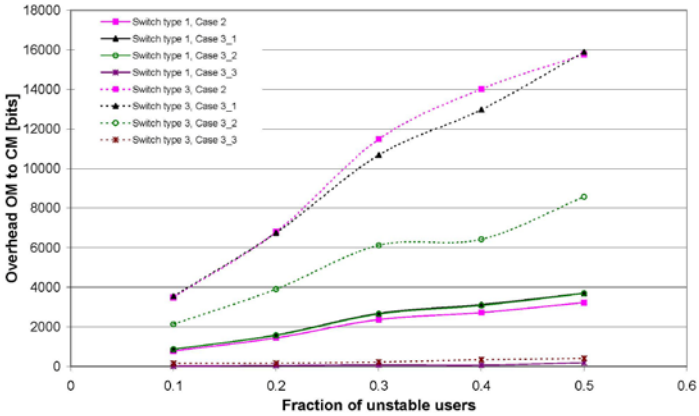

Fig. 10 Overhead for control of BP2 scheme (OM to CM).

space. This comes from the fact that when a BP2 is employed, the queues in the CMs stop sending traffic to the overloaded OMs which results in accumulating traffic in the CMs. Thus, applying solely a standard BP mechanism is not sufficient, and is even harmful, for the operation of a Clos-based SMM switch. Adding the BP1 mechanisms has variable success. Only the combination of BP2 and **BP_fine_load_balance** (Case 3_3) results in a clear advantage with respect to lowering the amount of required maximum buffer space in the input queues in the CMs, compared to the case where no BP is employed at all. It must be noted, that the achieved improvement is tightly related to the applied VOQ limit and threshold boundaries.

Fig. 8 presents the result for the required maximum amount of buffer space per input queue at any OM for both switch types and all back-pressure cases. Since all cases of employing back-pressure use BP2, it is clear that the maximum needed buffer space will be regulated strictly by the VOQ limit and the upper bound (see Tab.4). When no back-pressure is applied, **Switch type 1** needs lower maximum buffer space per input queue than **Switch type 3**. This result follows the observation from Fig. 5, where the total amount of buffer space per OM in **Switch type 1** was lower than the one for **Switch type 3**. This result indicates that not only the total amount of needed buffer space per module but also the maximum amount of buffer space per input queue follow the same trend: when a lot of buffer space is needed in the distribution part of the switch, less buffer space is needed in the aggregation part, and vice versa. When back-pressure is applied, the maximum allowed buffer length will be 10% lower than the VOQ limit, which is set the same for both switches: 4096 cells, and results in the obtained value of 3687 cells.

For all three results, it is clear that the more unstable users we have in the system, the higher the required maximum buffer space will be on average in order to accommodate the traffic bursts.

Fig. 9 and Fig. 10 present the amount of one-bit signals sent for back-pressure control throughout the entire simulation run for both switch types and all simulated back-pressure schemes. The first thing to note is that **Switch type 1** requires considerably lower amount of overhead in order to operate, which is due to the fact that **Switch type 1** has less number of IMs and OMs than **Switch type 3**. Furthermore, the combination of BP2 and **BP_fine_load_balance** (i.e., case 3_3) results in the lowest amount of overhead, for both switches types. This is due to the operation of **BP_fine_load_balance**, which performs load-balancing of the load from one IM to all CMs until the activating CM offloads enough. This requires additional complexity in the operation of the switch, since all IMs must be able to switch their configuration matrix continuously. Nevertheless, this BP1 mechanism obtains the best performance result with the least required overhead.

Conclusion

In this work we have evaluated the performance of Clos-based high-capacity switches under no-loss conditions via OPNET simulations. Two analyses are carried out. First, the effect of employing different internal switch structures on the amount of needed buffer space is evaluated for 3 different switch structures under 95% load conditions. Second, the ability of specifically designed back-pressure mechanisms to lower the requirement for maximum needed buffer space per input queue in the distribution and the aggregation part of the switch is evaluated. Under this analysis, temporary flow disturbances are introduced for activating the back-pressure mechanisms.

Two main conclusions can be drawn. First, the total amount of buffer space is roughly similar for all investigated internal structures, where if more memory is needed in the distribution part of the switch, then less memory is needed in the aggregation part, and vice versa. Second, applying solely a standard queue-to-queue back-pressure between the OMs and the CMs of a Clos-based SMM switch worsens the performance. In order to alleviate this drawback, novel back-pressure mechanisms between the CMs and the IMs are needed. The combined performance of both types of back-pressure mechanisms can indeed lower the amount of needed maximum buffer space per input queue. The extent of this improvement depends on the combined effect of the applied triggers for the activation of the back-pressure schemes and the experienced traffic unbalance.

Clos-based high-capacity switches provide modularity, extendibility and flexibility and are among the main candidates for the next generation high-capacity switch fabrics. But this provided flexibility and modularity must be carefully designed and evaluated for achieving the best possible performance under the lowest possible price.

Acknowledgment

This work has been partially supported by the Danish Advanced Technology Foundation (Højteknologifonden) through the research project "The Road to 100 Gigabit Ethernet".

References

[1] C. Hermsmeyer et al., "Towards 100G packet processing: Challenges and technologies," Bell Labs Technical Journal, vol. 14, no. 2, 2009.

[2] X. Li, Z. Zhou, and M. Hamdi, "Space-Memory-Memory Architecture for CLOS-network Packet Switches," in IEEE International Conference on Communications (ICC), 2005, pp. 1031–1035.

[3] R. Andreas, S. Ruepp, A. Manolova, M. Berger, H. Wessing, "Performance Evaluation of 100 Gigabit Ethernet Switching System," in Proc. of OPNETWORK 2011, Washington DC, USA, 2011.

[4] S. Ruepp, A. Rytlig, A. Manolova, M. Berger, H. Wessing, H. Yu, and L. Dittmann, "Performance evaluation of 100 Gigabit Ethernet switches under bursty traffic," in Proc. of 15th International Conference on Optical Network Design and Modeling (ONDM), Bologna, Italy, 2011.

[5] H. J. Chao, "Flow Control in a Multi-Plane Multi-Stage Buffered Packet Switch," in Proc. of High Performance Switching and Routing, HPSR, 2007.

[6] T. Kanazawa et al., "Input and Output Queueing Packet Switch with Backpressure Mode for Loss Sensitive Packets in Threshold Scheme," in Proc. of IEEE PACRIM, 1997, pp. 527–530.

[7] F. Chiussi et al., "Backpressure in shared-memory-based ATM switches under multiplexed bursty sources," in IEEE INFOCOM, 1996.

[8] H. T. Kung and R. Morris, "Credit-based flow control for ATM networks," IEEE Network, vol. 9, pp. 40–48, March/April 1995.

[9] R. Schoenen and A. Dahlhoff, "Closed Loop Credit-Based Flow control with Internal Backpressure in Input and Output Queued Switches," in Proc. of HPSR, 2000, pp. 195–203.

[10] A. Manolova, S. Ruepp, A. Rytlig, M. Berger, H. Wessing, and L. Dittmann, "Internal Backpressure for Terabit Switch Fabrics", in IEEE Communications Letters, Vol. 16, issue 2, pp. 265-267, 2012.

[11] I. Elhanany, "Fabric Benchmarking Traffic Models Version 1.0", Network Processing Forum, available at http://web.eecs.utk.edu/~itamar/Papers/NPF_FB.pdf