Interconnect Energy Savings and Lower Latency Networks in Hadoop Clusters: The Missing Link

Renan Fischer e Silva, Paul M. Carpenter
Barcelona Supercomputing Center—*Centro Nacional de Supercomputación* (BSC–CNS)
Universitat Politècnica de Catalunya (UPC), Barcelona, Spain
Email: {renan.fischeresilva,paul.carpenter}@bsc.es

***Abstract*—An important challenge of modern data centres running Hadoop workloads is to minimise energy consumption, a significant proportion of which is due to the network. Significant network savings are already possible using Energy Efficient Ethernet, supported by a large number of NICs and switches, but recent work has demonstrated that the packet coalescing settings must be carefully configured to avoid a substantial loss in performance. Meanwhile, Hadoop is evolving from its original batch concept to become a more iterative type of framework. Other recent work attempts to reduce Hadoop's network latency using Explicit Congestion Notifications. Linking these studies reveals that, surprisingly, even when packet coalescing does not hurt performance, it can degrade network latency much more than previously thought. This paper is the first to analyze the impact of packet coalescing in the context of network latency. We investigate how to design and configure interconnects to provide the maximum energy savings without degrading cluster throughput performance or network latency.**

***Keywords*—IEEE 802.3az, Energy Efficiency, MapReduce, Hadoop, AQM, ECN, Packet Coalescing**

I. Introduction

An important challenge of modern data centres is to minimise energy consumption, a significant proportion of which is due to the network. Network energy savings are possible using Energy Efficient Ethernet (EEE) IEEE 802.3az, which is already supported by a large number of NICs and switches. Our previous work was the first to study the impact of Energy Efficient Ethernet on MapReduce workloads [1]. MapReduce [2] and its open-source implementation, Apache Hadoop [3], are widely used for the processing of huge data sets on large commodity clusters. Overall, we found that although substantial energy savings are available, the packet coalescing settings must be carefully configured to avoid a substantial loss in performance.

Meanwhile, network switches are steadily increasing their per-port buffer capacities. New SDRAM-based products are being launched with per-port buffer densities of up to ten times larger [4]. Large buffers increase throughput, but they can exacerbate the Bufferbloat problem [5], with network latencies reaching tens of milliseconds for certain classes of workloads. A recent paper of ours was the first to quantitatively evaluate the control of network latency in Hadoop clusters, which was done using Active Queue Management (AQM) with Explicit Congestion Notification (ECN), and had minimal impact on Hadoop batch performance [6].

This paper connects these two distinct efforts, by introducing a cluster design approach for reducing the interconnect energy consumption while also reducing network latency. As previously demonstrated, the Packet Coalescing settings must be carefully configured in order to avoid a substantial loss in performance [1]. Even so, the impact of the extra network latency incurred by packet coalescing, which increased the Bandwidth–Delay Product, had to be compensated with more buffering and TCP packets in-flight. The increase in latency was tolerable because of Hadoop's original batch-oriented design. In contrast, it is surprisingly difficult to effectively combine packet coalescing on the 10 GbE links with controlled latency, as implemented using ECN combined with AQM.

Our guidelines are especially targeted for workloads with long east–west flows inside the data centre, such as Apache Hadoop. Our findings are simple to implement and straightforward to understand. By considering our settings and configurations, vendors and cluster administrators can reduce interconnect energy consumption without adversely affecting network latency. We also wish to open discussion and promote research towards new solutions. We present experimental results in terms of interconnect energy consumption, cluster throughput and network latency. Finally, we show the impact on Hadoop job execution time.

In short, our main contributions are:

1) We analyse the impact of different buffer densities and Packet Coalescing settings on Hadoop network latency.
2) We align the Packet Coalescing technique with Active Queue Management to reduce network latency and identify how to extract the best from the combined techniques.
3) We evaluate the proposed solution in terms of interconnect energy consumption, cluster throughput and network latency, as well as its expected impact on Hadoop job execution time.
4) We provide a set of recommendations to network equipment manufacturers and cluster administrators in order to benefit from this work.

The rest of the paper is organized as follows. Section II describes our methodology and Section III presents the results. Based on these results, Section IV distills the most important recommendations. Section V compares our approach with previous work. Finally, Section VI concludes the paper.

TABLE I
SIMULATED ENVIRONMENT

Category	Parameter	Value
Simulated hardware		
System	Number nodes	80
	Number racks	2
Node	CPU	Intel Xeon 2.5 GHz L5420
	Number cores	2
	Number processors	2
Network	Each node	*1GbE:* 1 —
	Each leaf switch	*1GbE: 40 10GbE:* 1
	Each spine switch	— *10GbE:* 1
Buffers	Shallow buffer per-port	200 packets - approx. 100 KB per port
	Deep buffer per-port	2000 packets - approx.1 MB per port
Link power	10GbE	2.5 W
RED settings	Min. and Max. Thresholds	125 - 375
TCP buffer	Max. packet per connection	Unlimited

II. METHODOLOGY

This section describes the experimental methodology for this paper based on our recent work [6], [7] and using the NS–2 packet-level network simulator [8]. The topology selected for this work was the leaf–spine architecture [9], which is generally recommended for Hadoop, as seen in various references for cluster design [10]–[12]. We provide results for two different buffer sizes: shallow and deep buffer switches. Table I also shows the configuration of the simulated workload using the MapReduce simulator MRPerf [13]. A single Terasort job is configured to sort 4.9 GB (random elements). Terasort is a popular batch benchmark commonly used to measure MapReduce performance on a Hadoop cluster [14].

We assume the sleep and wake timings given in Table II. The *ideal* case uses Energy Efficient Ethernet, but the sleep and wake transitions were considered to be both instantaneous and zero energy, providing an "ideal" point of comparison. In this case, the link is optimally controlled by simply entering low power mode as soon as it becomes inactive, providing perfect energy proportionality without affecting runtime. This result gives a lower bound on energy consumption. Finally, we also considered different values for packet coalescing. The Ethernet specs used in this work are given in Table III.

We considered RED as the selected AQM to mark packets with ECN feature configured on TCP senders. Finally, the four performance metrics considered are: the *interconnect energy consumption* which is the energy consumed by 10GbE links, the *runtime* which is the total time needed to finish the Terasort workload, which is inversely proportional to the effective throughput of the cluster; the *average throughput* per node and the *average end-to-end latency* per packet.

TABLE II
EEE WAKE AND SLEEP OPERATIONS

Speed	Min. T_w (µs)	Min. T_s (µs)
1000Base-T	16.5	182
10GBase-T	4.48	2.88
Ideal	0	0

TABLE III
ETHERNET SPECS

Label	Packet Coalescing settings	
	Holding time	Trigger
legacy eth	*No Energy Efficient Ethernet*	
ideal	*No overhead from sleep and wake operations*	
eee	*Energy Efficient Ethernet - no Packet Coalescing*	
12us10	12 µs	10 packets
120us100	120 µs	100 packets
500us500	500 µs	500 packets
1ms1000	1 ms	1000 packets

III. RESULTS

A. Buffer density and Packet Coalescing on Hadoop

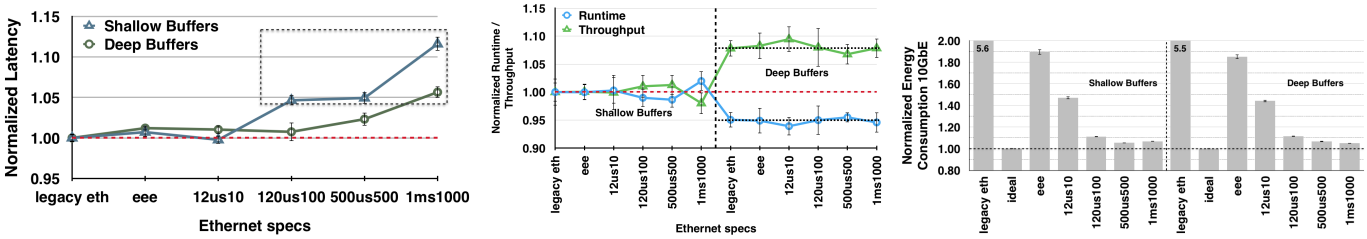
We start the analysis of our results by discussing Figure 1a. The dashed area shows the extra latency introduced by Packet Coalescing, which for shallow buffers translates into an additional latency of 12% while for deep buffers the extra latency is about 6%. Since the latency found on deep buffers is much higher, the extra latency incurred by Packet Coalescing accounts for a lower (relative) impact on the normalized numbers.

Figure 1b presents the execution time and throughput results normalized to the shallow buffer baseline. We verify that more buffer density translates to higher throughput which translates to a faster runtime. We also verify that Packet Coalescing increases variability, but overall, the extra latency can be compensated by more buffering and packets in flight. Therefore, the gains obtained from deep buffers where maintained, even with the more aggressive setting for Packet Coalescing.

Finally, we analyse Figure 1c. The values are normalized to the ideal energy consumption, which means zero energy for sleep and wake operations. On our benchmark we verify that the 10GbE links consumed more than five times the energy consumption per NIC. We zoomed-in the bars to obtain a clearer comparison for the other settings. Energy Efficient Ethernet is able to significantly reduce the energy consumption but it is still almost 80% from ideal. It was therefore possible to obtain considerable gains with Packet Coalescing. The best gains were obtained using 1ms1000 with deep buffers, reaching near only 5% more energy than the ideal model, while 120us100 was near 10% from that and 500us500 in between these two settings.

We move on with the next set of experiments, which consist of: enabling ECN on the TCP end-points, enabling ECN's marking feature on each RED egress buffer, and using the configuration described in Section II. We expect to not only reduce network latency but also maintain cluster throughput and specially maximize the energy savings for the 10GbE links, obtained with Packet Coalescing.

A



(a) Network latency (b) Runtime and throughput (c) Energy consumption of 10GbE

Fig. 1. Packet coalescing impact considering different buffer sizes

B. Combining Packet Coalescing with ECN/AQM/RED

Figure 2 brings the detailed results considering our four metrics described on Section II. Starting by Hadoop performance, we can see that the two more agressive settings decreased cluster throughput, which also impacted on a larger execution time. While 500us500 impacted on approximately 25% performance degradation, the more aggressive Packet Coalescing 1ms1000 inflicted a 50% performance loss.

Analyzing the energy consumption, we also see that the two more aggressive Packet Coalescing settings no longer provide the best energy savings. The increase on execution time was responsible for losing all the greatness on energy savings we verified when Packet Coalescing is used stand-alone.

Finally, we verified an overall reduction on network latency as expected. Considering the Packet Coalescing settings, the network latency suffered an increment of almost 50% for shallow buffers when using the more aggressive Packet Coalescing settings. For deep buffers the extra latency was responsible for a smaller increment of 10%. Still, when combining our metrics together, we can no longer verify any benefit of utilizing 1ms1000 or even 500us500. We included a star to highlight the best combination which includes latency compared to the baseline, energy near 10% the ideal model and finally no loss on performance and cluster throughput. Considering 120us100 packets, we demonstrate it is possible to achieve a much lower network latency while still maintaining the interconnect energy savings obtained by utilizing Packet Coalescing.

IV. Discussion and Recommendations

The results presented in this paper show that Hadoop clusters can significantly benefit from packet coalescing combined with proactive congestion control mechanisms. *The results presented here are not exclusive to Hadoop, but are expected to be reproduced on other types of workload that present the following three characteristics*: East–west traffic patterns and long-lived TCP flows with bursty communication; TCP flows configured to use ECN, either as TCP–ECN or DCTCP, and switches configured to mark packets; NICs and switches that implement Energy Efficient Ethernet with the option to coalesce packets. We now distill our most important recommendations.

a) Recommendations for equipment vendors: Due to the potential energy savings, equipment vendors should consider implementing Packet Coalescing in their NICs and switches. It is important, however, to offer some reconfigurability, since depending on the workload, more aggressive settings may be desired while for other classes of workloads, less aggressive settings may already provide good energy savings. Therefore we strongly recommend that Packet Coalescing should offer some flexibility for its configuration.

b) Recommendations for network administrators: Energy Efficient Ethernet NICs do not currently offer the possibility to adjust the configuration of the Packet Coalescing settings. We argue that EEE NICs should in future offer such flexibility. If this does finally happen, we recommend this work as a guideline to obtain maximum energy savings without degrading Hadoop performance or network latency. For batch workloads where latency is not a concern, we recommend the more aggressive settings which have its extra latency compensated with more buffering and packets in-flight. If reducing network latency is the major concern, we recommend the utilization of some congestion control mechanism as ECN or DCTCP without discarding the utilization of Packet Coalescing. We demonstrated it is feasible and possible to combine both techniques with no loss on the four metrics considered on this work.

V. Related Work

Energy Efficient Ethernet (EEE): Our previous work was the first to study the impact of Energy Efficient Ethernet, including Packet Coalescing, on MapReduce workloads [1]. Overall, we found that Packet Coalescing offers substantial energy savings, of 20% to 60% beyond that of standard EEE, the packet coalescing settings must be carefully configured to avoid a substantial loss in performance [1].

Latency control: A recent paper of ours [6] compared ECN and DCTCP [15] performance and showed a performance degradation of about 20%, with respect to the baseline, which used deep buffer switches. In that study ECN was considered to achieve a lower latency than DCTCP, showing that in a congested environment with long-lived TCP flows, both TCP combined with ECN and DCTCP can achieve similar throughputs, but the more aggressive cut in the congestion window in the case of ECN leads to a lower-latency solution. For such a reason, on this work we considered only ECN with the AQM settings from RED described in Section II.

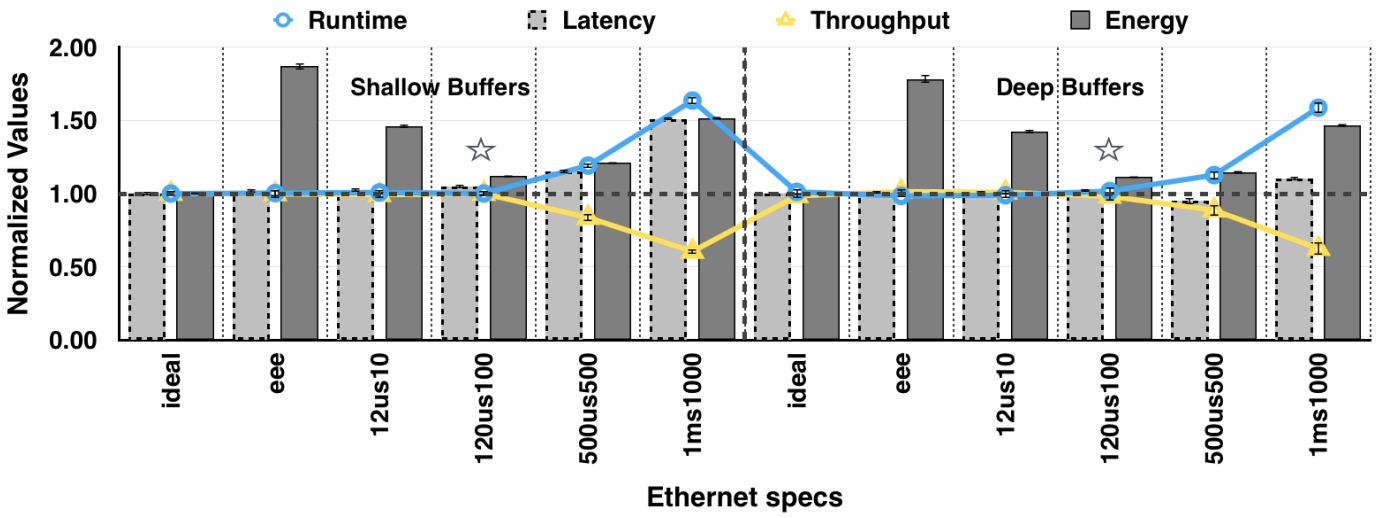


Fig. 2. Runtime, Latency, Throughput and Energy values for Packet Coalescing combined with RED and ECN

VI. Conclusions

This paper has presented a novel analysis of the impact of Energy Efficient Ethernet (EEE) and Packet Coalescing on network latency for Hadoop Clusters. Combining Packet Coalescing with ECN plus AQM, which is already found on network switches, delivers network latencies comparable to that found for ideal on/off links, without EEE's sleep and wake overheads. We were also able to reduce the energy consumption from 10GbE links by 70%, compared to default EEE, which does not use Packet Coalescing.

In summary, we suggest that equipment vendors implement Packet Coalescing and also provide the ability for operators to modify the Packet Coalescing configuration settings. In turn, we suggest that network administrators use the recommendations in this paper together with knowledge of their application's network latency requirements. Doing so will provide the best possible energy savings without compromising performance or latency requirements.

VII. Acknowledgment

The research leading to these results has received funding from the European Unions Seventh Framework Programme (FP7/2007–2013) under grant agreement number 610456 (Euroserver). The research was also supported by the Ministry of Economy and Competitiveness of Spain under the contracts TIN2012-34557 and TIN2015-65316-P, Generalitat de Catalunya (contracts 2014-SGR-1051 and 2014-SGR-1272), HiPEAC-3 Network of Excellence (ICT- 287759), and the Severo Ochoa Program (SEV-2011-00067) of the Spanish Government.

References

[1] R. Fischer e Silva and P. M. Carpenter, "Exploring interconnect energy savings under East-West traffic pattern of MapReduce clusters," in *40th Annual IEEE Conference on Local Computer Networks (LCN 2015)*, Clearwater Beach, USA, Oct. 2015, pp. 10–18.

[2] J. Dean and S. Ghemawat, "MapReduce: Simplified data processing on large clusters," in *Proceedings of the 6th Conference on Symposium on Opearting Systems Design & Implementation*, ser. OSDI'04. Berkeley, CA, USA: USENIX Association, 2004, pp. 10–10. [Online]. Available: http://dl.acm.org/citation.cfm?id=1251254.1251264

[3] The Apache Software Foundation, "Apache Hadoop Project," http://hadoop.apache.org, accessed: 2016-04-20.

[4] Cisco, "Network switch impact on big data hadoop-cluster data processing: Comparing the hadoop-cluster performance with switches of differing characteristics," Tech. Rep., 2016.

[5] J. Gettys and K. Nichols, "Bufferbloat: Dark Buffers in the Internet," *Queue*, vol. 9, no. 11, pp. 40:40–40:54, Nov. 2011. [Online]. Available: http://doi.acm.org/10.1145/2063166.2071893

[6] R. F. E. Silva and P. M. Carpenter, "Controlling network latency in mixed hadoop clusters: Do we need active queue management?" in *2016 IEEE 41st Conference on Local Computer Networks (LCN)*, Nov 2016, pp. 415–423.

[7] R. F. e Silva and P. M. Carpenter, "Energy efficient ethernet on mapreduce clusters: Packet coalescing to improve 10gbe links," *IEEE/ACM Transactions on Networking*, vol. PP, no. 99, pp. 1–12, 2017.

[8] "Network Simulator NS-2," http://www.isi.edu/nsnam/ns, accessed: 2016-04-20.

[9] Cisco, "Cisco data center spine-and-leaf architecture: Design overview," Tech. Rep., 2016.

[10] A. Bechtolsheim, L. Dale, H. Holbrook, and A. Li, "Why Big Data Needs Big Buffer Switches. Arista White Paper," Tech. Rep., 2011.

[11] E. Networks, "Extreme networks: Big data a solutions guide," Tech. Rep., 2014.

[12] Cisco, "Cisco's massively scalable data center: Network fabric for warehouse scale computer," Tech. Rep.

[13] G. Wang, A. R. Butt, P. Pandey, and K. Gupta, "Using realistic simulation for performance analysis of Mapreduce setups," in *Proceedings of the 1st Workshop on Large-Scale System and Application Performance*, ser. LSAP '09. New York, NY, USA: ACM, 2009, pp. 19–26. [Online]. Available: http://doi.acm.org/10.1145/1552272.1552278

[14] R. F. e Silva and P. M. Carpenter, "High throughput and low latency on hadoop clusters using explicit congestion notification: The untold truth," in *Accepted to 2017 IEEE 19th IEEE International Conference on Cluster Computing, (CLUSTER)*, Sep 2017.

[15] M. Alizadeh, A. Greenberg, D. A. Maltz, J. Padhye, P. Patel, B. Prabhakar, S. Sengupta, and M. Sridharan, "Data center TCP (DCTCP)," in *Proceedings of the SIGCOMM 2010 Conference*, ser. SIGCOMM '10. New York, NY, USA: ACM, 2010, pp. 63–74. [Online]. Available: http://doi.acm.org/10.1145/1851182.1851192