Ecole Doctorale STIC

Sciences & Technologies de l'Information et de la Communication

MINES ParisTech

Université Nice SOPHIA ANTIPOLIS

ED STIC - Proposition de Sujets de Thèse

pour la campagne d'Allocation de thèses 2017

Axe Sophi@Stic :	aucun|
Titre du sujet :	Joint Application and Network Optimization of Big Data Analytics
Mention de thèse :	Informatique
HDR Directeur de thèse inscrit à l'ED STIC :	Fabrice Huet

Co-encadrant de thèse éventuel :

Nom :	Lopez-Pacheco
Prénom :	Dino
Email :	dino.lopez@unice.fr
Téléphone :	

Email de contact pour ce sujet :	fabrice.huet@unice.fr
Laboratoire d'accueil :	I3S

Description du sujet :

Motivation

Batch and data stream applications are routinely used to process respectively large amounts of static data or data streams arriving at high velocity. In the case of batch
processing, the network often constitutes a bottleneck especially during the shuffle phase of MapReduce applications [3].
The network need to be taken into account in real-time stream processing applications, which are partitioned into tasks (that form a directed acyclic graph) distributed over
compute nodes [4]. To scale with demand (rate or complexity of input stream), some tasks might be duplicated, the complexity of the task varying depending on whether the
task is stateful [5] or stateless [6,7].

The networking community has also proposed some solutions to improve the performance of such applications. A first stream of work has strived to improve the bisectional bandwidth offered in data centers [8]. Some solutions have also been proposed at the transport layer, such as DTCP, which aims at alleviating the incast problem arising typically during the shuffle phases of MapReduce jobs [9] or L2DCT [12]. Last but not least, some efforts have been devoted to design schedulers that could meet the constraints of big data analytics solutions [10,11].

Objectives

In this thesis, we aim at exploring the synergy between the network and the application layer when scheduling big data analytics. Those applications feature a scheduler that steers the execution of submitted jobs. We are seeking to interconnect this scheduler with a network scheduler to :

Help the application scheduler to optimize the initial placement of computation tasks.

Make use of traffic engineering techniques to optimize criteria like the number of completed jobs.

Perform a post-mortem analysis of failed jobs to understand the root cause of the problem.

We envisage to use a rich networking toolbox to achieve the above objectives, typically we might:

Assume a Software Defined Networking (SDN) data center [13].

Use different active and passive measurement or troubleshooting tools, e.g. [20,21,22].

Use variants of TCP specifically designed for data centers [9,10].

Use some specific network schedulers to be deployed at the servers or in the network [15].

The synergy between the network and the application controllers has been explored in a recent work, [15] and we would like to further study this joint optimization problem with a richer toolbox. In particular, we believe that SDN, with its advanced traffic engineering capabilities could be a key asset to effectively control a data center.

See : http://i3s.unice.fr/~huet/docs/SDN-these.pdf

References

[3] Ahmad, Faraz, et al. "ShuffleWatcher: Shuffle-aware Scheduling in Multi-tenant MapReduce Clusters." USENIX Annual Technical Conference. 2014.

[4] Peng, Boyang, et al. "R-storm: Resource-aware scheduling in storm." Proceedings of the 16th Annual Middleware Conference. ACM, 2015.

[5] Caneill, Matthieu, et al. "Locality-Aware Routing in Stateful Streaming Applications." Middleware'16-17th International Middleware Conference. 2016.

[6] Rivetti, Nicoló, et al. "Online Scheduling for Shuffle Grouping in Distributed Stream Processing Systems Research Paper." ACM/IFIP/USENIX Middleware 2016. 2016.

[7] Schneider, Scott, et al. "Dynamic Load Balancing for Ordered Data-Parallel Regions in Distributed Streaming Systems." Proceedings of the 17th International Middleware
Conference. ACM, 2016.

[8] Singh, Arjun, et al. "Jupiter rising: A decade of clos topologies and centralized control in google's datacenter network." ACM SIGCOMM Computer Communication
Review 45.4 (2015): 183-197.

[9] Alizadeh, Mohammad, et al. "Data center tcp (dctcp)." ACM SIGCOMM computer communication review. Vol. 40. No. 4. ACM, 2010.

[10] Hong, Chi-Yao, Matthew Caesar, and P. Godfrey. "Finishing flows quickly with preemptive scheduling." ACM SIGCOMM Computer Communication Review 42.4 (2012):
127-138.

[11] Zats, David, et al. "DeTail: reducing the flow completion time tail in datacenter networks." ACM SIGCOMM Computer Communication Review 42.4 (2012): 139-150.

[12] Munir, Ali, et al. "Minimizing flow completion times in data centers." INFOCOM, 2013 Proceedings IEEE. IEEE, 2013.

[13] Kreutz, Diego, et al. "Software-defined networking: A comprehensive survey." Proceedings of the IEEE 103.1 (2015): 14-76.

[14] Jain, Sushant, et al. "B4: Experience with a globally-deployed software defined WAN." ACM SIGCOMM Computer Communication Review 43.4 (2013): 3-14.

[15] Ali Munir, Ting He, Ramya Raghavendra, Franck Le, and Alex X. Liu. “Network Scheduling Aware Task Placement in Datacenters”. In Proceedings of the 12th
International on Conference on emerging Networking EXperiments and Technologies (CoNEXT). 2016.

[16] Zhang, Hong, et al. "CODA: Toward automatically identifying and scheduling coflows in the dark." Proceedings of the 2016 conference on ACM SIGCOMM 2016
Conference. ACM, 2016.

[17] Chen, Li, et al. "Scheduling Mix-flows in Commodity Datacenters with Karuna." Proceedings of the 2016 conference on ACM SIGCOMM 2016 Conference. ACM, 2016.

[19] Wierman, Adam, and Misja Nuyens. "Scheduling despite inexact job-size information." ACM SIGMETRICS Performance Evaluation Review. Vol. 36. No. 1. ACM, 2008.

[20] Chen, Ang, et al. "The good, the bad, and the differences: Better network diagnostics with differential provenance." Proceedings of the 2016 conference on ACM

SIGCOMM 2016 Conference. ACM, 2016.

[21] Ciraci, Selim, et al. "Taking the Blame Game out of Data Centers Operations with NetPoirot." Proceedings of the 2016 conference on ACM SIGCOMM 2016 Conference. ACM, 2016.

[22] Dovrolis, Constantinos, Parameswaran Ramanathan, and David Moore. "Packet-dispersion techniques and a capacity-estimation methodology." IEEE/ACM Transactions On Networking 12.6 (2004): 963-977.

[23] Harchol-Balter, Mor, et al. "SRPT scheduling for web servers." Workshop on Job Scheduling Strategies for Parallel Processing. Springer Berlin Heidelberg, 2001.

URL : http://i3s.unice.fr/~huet/docs/SDN-these.pdf

English version:

Motivation

Batch and data stream applications are routinely used to process respectively large amounts of static data or data streams arriving at high velocity. In the case of batch processing, the network often constitutes a bottleneck especially during the shuffle phase of MapReduce applications [3].

The network need to be taken into account in real-time stream processing applications, which are partitioned into tasks (that form a directed acyclic graph) distributed over compute nodes [4]. To scale with demand (rate or complexity of input stream), some tasks might be duplicated, the complexity of the task varying depending on whether the task is stateful [5] or stateless [6,7].

The networking community has also proposed some solutions to improve the performance of such applications. A first stream of work has strived to improve the bisectional bandwidth offered in data centers [8]. Some solutions have also been proposed at the transport layer, such as DTCP, which aims at alleviating the incast problem arising typically during the shuffle phases of MapReduce jobs [9] or L2DCT [12]. Last but not least, some efforts have been devoted to design schedulers that could meet the constraints of big data analytics solutions [10,11].

Objectives

In this thesis, we aim at exploring the synergy between the network and the application layer when scheduling big data analytics. Those applications feature a scheduler that steers the execution of submitted jobs. We are seeking to interconnect this scheduler with a network scheduler to :

Help the application scheduler to optimize the initial placement of computation tasks.

Make use of traffic engineering techniques to optimize criteria like the number of completed jobs.

Perform a post-mortem analysis of failed jobs to understand the root cause of the problem.

We envisage to use a rich networking toolbox to achieve the above objectives, typically we

might:
Assume a Software Defined Networking (SDN) data center [13].
Use different active and passive measurement or troubleshooting tools, e.g. [20,21,22].
Use variants of TCP specifically designed for data centers [9,10].
Use some specific network schedulers to be deployed at the servers or in the network [15].

The synergy between the network and the application controllers has been explored in a recent work, [15] and we would like to further study this joint optimization problem
with a richer toolbox. In particular, we believe that SDN, with its advanced traffic engineering capabilities could be a key asset to effectively control a data center.

See : http://i3s.unice.fr/~huet/docs/SDN-these.pdf
References

[3] Ahmad, Faraz, et al. "ShuffleWatcher: Shuffle-aware Scheduling in Multi-tenant MapReduce Clusters." USENIX Annual Technical Conference. 2014.

[4] Peng, Boyang, et al. "R-storm: Resource-aware scheduling in storm." Proceedings of the 16th Annual Middleware Conference. ACM, 2015.

[5] Caneill, Matthieu, et al. "Locality-Aware Routing in Stateful Streaming Applications." Middleware'16-17th International Middleware Conference. 2016.

[6] Rivetti, Nicoló, et al. "Online Scheduling for Shuffle Grouping in Distributed Stream Processing Systems Research Paper." ACM/IFIP/USENIX Middleware 2016. 2016.

[7] Schneider, Scott, et al. "Dynamic Load Balancing for Ordered Data-Parallel Regions in Distributed Streaming Systems." Proceedings of the 17th International Middleware
Conference. ACM, 2016.

[8] Singh, Arjun, et al. "Jupiter rising: A decade of clos topologies and centralized control in google's datacenter network." ACM SIGCOMM Computer Communication
Review 45.4 (2015): 183-197.

[9] Alizadeh, Mohammad, et al. "Data center tcp (dctcp)." ACM SIGCOMM computer communication review. Vol. 40. No. 4. ACM, 2010.

[10] Hong, Chi-Yao, Matthew Caesar, and P. Godfrey. "Finishing flows quickly with preemptive scheduling." ACM SIGCOMM Computer Communication Review 42.4 (2012):
127-138.

[11] Zats, David, et al. "DeTail: reducing the flow completion time tail in datacenter networks." ACM SIGCOMM Computer Communication Review 42.4 (2012): 139-150.

[12] Munir, Ali, et al. "Minimizing flow completion times in data centers." INFOCOM, 2013 Proceedings IEEE. IEEE, 2013.

[13] Kreutz, Diego, et al. "Software-defined networking: A comprehensive survey." Proceedings of the IEEE 103.1 (2015): 14-76.

[14] Jain, Sushant, et al. "B4: Experience with a globally-deployed software defined WAN." ACM SIGCOMM Computer Communication Review 43.4 (2013): 3-14.

[15] Ali Munir, Ting He, Ramya Raghavendra, Franck Le, and Alex X. Liu. “Network Scheduling Aware Task Placement in Datacenters”. In Proceedings of the 12th International on Conference on emerging Networking EXperiments and Technologies (CoNEXT). 2016.

[16] Zhang, Hong, et al. "CODA: Toward automatically identifying and scheduling coflows in the dark." Proceedings of the 2016 conference on ACM SIGCOMM 2016 Conference. ACM, 2016.

[17] Chen, Li, et al. "Scheduling Mix-flows in Commodity Datacenters with Karuna." Proceedings of the 2016 conference on ACM SIGCOMM 2016 Conference. ACM, 2016.

[19] Wierman, Adam, and Misja Nuyens. "Scheduling despite inexact job-size information." ACM SIGMETRICS Performance Evaluation Review. Vol. 36. No. 1. ACM, 2008.

[20] Chen, Ang, et al. "The good, the bad, and the differences: Better network diagnostics with differential provenance." Proceedings of the 2016 conference on ACM SIGCOMM 2016 Conference. ACM, 2016.

[21] Ciraci, Selim, et al. "Taking the Blame Game out of Data Centers Operations with NetPoirot." Proceedings of the 2016 conference on ACM SIGCOMM 2016 Conference. ACM, 2016.

[22] Dovrolis, Constantinos, Parameswaran Ramanathan, and David Moore. "Packet-dispersion techniques and a capacity-estimation methodology." IEEE/ACM Transactions On Networking 12.6 (2004): 963-977.

[23] Harchol-Balter, Mor, et al. "SRPT scheduling for web servers." Workshop on Job Scheduling Strategies for Parallel Processing. Springer Berlin Heidelberg, 2001.

URL : http://i3s.unice.fr/~huet/docs/SDN-these.pdf