ConExA: ConEx-Based Load Balancing for Data Centers

Mirja Kühlewind  
ETH Zurich  
mirja.kuehlewind@tik.ee.ethz.ch  

Dirk Kutscher  
NEC Europe Labs  
dirk.kutscher@neclab.eu

ABSTRACT
In this paper we present an approach to utilize Congestion Exposure (ConEx) in data centers for multipath routing. The ConEx signaling protocol is currently under standardization in the Internet Engineering Task Force (IETF) and provides the network congestion information based on transport layer feedback such as loss or Explicit Congestion Notification (ECN). Therefore we propose the use of ECN in combination with a Data Center TCP (DCTCP)-like congestion control to obtain a more fine-grained congestion response without the injection of packet loss. Subsequently ConEx-enabled senders inform the network about the congestion encountered earlier by this flow. This information is used in the network to make multipath routing decisions. This is initial work proposing a data center load balancing architecture.

1. INTRODUCTION
Data center networks are high speed, high volume networks often deploying a clos structure providing multiple paths from one server to another. Examples are VL2 [4] and Portland [8] both utilizing Equal Cost MultiPath (ECMP) routing. Due to known limitation of ECMP for data centers recently a new approach, named CONGA (CONGestion Aware balancing) [2], was proposed. CONGA is a load balancing architecture for data centers using explicit in-band congestion signaling between a source and a destination leaf switch. The source leaf makes the load balancing decision based on rate estimation for each out-going link as well as congestion information provided by the destination leaf. Therefore a counter in each packet can be increased by each forwarding node to indicate congestion and is feed back to the source leaf by attaching additional information to reverse traffic from the destination leaf switch.

In this paper we propose ConExA, a load balancing architecture based on ConEx [3] information and utilizing a DCTCP-like congestion control. Therefore ConExA is based on the same approach than CONGA but uses standardized protocols and, moreover, avoids packet loss for congestion signaling.

As ConEx is an IP extension for congestion signaling under standardization, it is easier to deploy, induces less overhead, and does not introduce a dependency of a data center operator to a certain vendor. ConEx operates based on ECN [9], a standardized signaling protocol indicating congestion to the sender without the injection of loss. DCTCP is an ECN-based congestion control proposal that addresses the TCP incast problem in data centers and therefore maintains low latency. DCTCP requires modification to congestion control in the end host and utilizes a certain Active Queue Management (AQM) configuration that can simply be parameterized in most currently available switches.

Due to the limited resolution of ConEx and a one Round-Trip Time (RTT) time delay, the provided information is less accurate. As RTTs are small in data center, we argue that this still provides a valuable approach while avoiding packet loss and any introduction of proprietary protocols compared to CONGA.

2. BACKGROUND
ConEx [3] is a signaling protocol to expose the congestion encountered by a flow in the previous RTT to the network. ConEx is currently under standardization as IPv6 ConEx Destination Option (CDO) [5]. For TCP flows this congestion information is derived based on loss or ECN information. ECN [9] is a TCP/IP extension where network nodes can signal congestion without dropping packets. ECN was standardized in 2001 and is widely implemented but hardly used as it needs support by the sender, the receiver, and the congested network node. However, that is no problem in data centers.

Therefore e.g. DCTCP [1] proposes congestion control for data centers utilizing ECN to get an early congestion feedback without the drawback of a high loss rate. DCTCP is a combination of a certain congestion control and AQM scheme. DCTCP implements three changes: a different reaction to congestion in the sender, a specific AQM configuration based on Random Early Detection (RED) in the network node’s congestion feedback, and a more accurate ECN feedback mechanism from the receiver to the sender. With the use of ECN and DCTCP-like congestion control a more fine-grained congestion feedback can be induce into the
network without causing high loss rates or under utiliza-
tion due to a too strong response to congestion at the
sender. Note a more accurate ECN feedback, as needed
due to DCTCP, is already under standardization [6] and
therefore could easily be deployed in data center hosts.

Finally a ConEx sender sets one mark in the CDO of
an outgoing (data) segment for each incoming ECN con-
gestion feedback. Using this information, an immediate
network node can estimate the current congestion level
on a path by counting the number of marked packet-
s/bytes and compare them to the total traffic volume of
a certain outgoing interface as further described below.

3. OVERVIEW OF CONEXA

ConExA assumes a clos architecture for the data cen-
ter. Figure 1 shows a 2-stage leaf/spine fabric example
architecture. Servers are connected to the leaf switching
fabrics. All traffic is assumed to be TCP as well as be-
ing ConEx-enabled and using a DCTCP-like congestion
control based on ECN. The source leaf switch fabric
is performing the load balancing decision as explained
next. All other switching fabrics implement RED with
ECN support and are configured with a marking function
that calculates a increasing marking probability
based on the current queue length.

Similar as in our previous work [7] we propose the
use of DCTCP with minor modification on the conges-
tion control and the use of a RED configuration with
a non-smoothing marking slope. Further we envision
the use of the more accurate ECN feedback mechanism
as proposed in standardization instead of a proprietary
approach as proposed for DCTCP in [1].

Load Balancing Decision.

The source leaf switch performs the load balancing
based on a switching and a congestion information ta-
ble. The switching table maintains a hash of the 5-tuple
for each active TCP flow, the out-going port number
for this flow and a Time-To-Live (TTL) value that is
decreased over time if no new traffic for this flow is ob-
served anymore. The congestion information table is
used to estimate the congestion level for a certain out-
going port number and destination address subnet that
is connected to one destination leaf switch. For each
pair we maintain a congestion level value, the time of
the last update, the number of sent IP payload bytes
b\_sent since the last update, and the number of ConEx-
marked payload bytes b\_marked since the last update.

For each in-coming packet we first check if we already
have an entry in the switching table. If so, we add the
IP payload bytes to the corresponding b\_sent value and
also to b\_marked if marked with the ConEx E flag.

Further we check if the time since the last conges-
tion level update is larger than T\_update. If so, we recal-
culate the congestion level by dividing the total number of sent
bytes b\_sent by the number of marked bytes b\_marked
and update both the congestion level value for this port
number destination subnet pair as well as the respective
time stamp with the current time.

Otherwise if we do not have an entry for the flow of
the current packet, we look up all congestion level
values for the respective destination subnet pair as well as the respective
time stamp with the current time.

4. EVALUATION AND FUTURE WORK

As our proposal is based on similar mechanisms than
CONGA but using standardized protocols, we do not
question the implementation feasibility of ConExA. Note,
as ConEx information are available on the whole path,
ConExA could potentially also be implemented distributed
on each server when using end-to-end tunneling.

While CONGA proposes to work on a flowlet gran-
ularity and shows evidence that this is feasible, they
did not evaluate if the load balancing benefits from this
additional complexity. For simplification we currently
propose flow level granularity but plan further evalua-
tion to compare different switching granularities. Fur-
ther we propose to update the congestion level value in
fixed time intervals. Simulative evaluation is needed to
identify the the right update interval but the evalua-
tion of CONGA already indicated that this parameter
should be in a range between 100 µs and 500 µs. Note,
CONGA proposes to use a Exponential Weighted Mov-
ing Average (EWMA) for their rate estimator. We plan
further evaluation to assess if the additional complex-
ity of using EWMA provides benefits regarding the low
resolution congestion level information.

If ConExA can provide similar performance than the
proposed and already in upcoming products used CONGA,
it provides the clear advantage that no proprietary pro-
tocols are used and therefore a data center operator
does not dependent on a specific vendor. Further as
ConExA is ECN and DCTCP based, it additionally can
avoid packet loss and support low latency.
5. REFERENCES