Using the Latest Cloud Technology to Accelerate LS-DYNA®

Rodney Mach
TotalCAE

Abstract

This paper will focus on the latest technologies available on cloud to accelerate LS-DYNA using the major IaaS vendors including Amazon AWS and Microsoft Azure, and how they impact the total simulation job cost of a LS-DYNA benchmark model.

Background

Customers today have many options to access a wide variety of compute technologies to accelerate LS-DYNA. This paper will highlight some of the latest HPC technologies on the major IaaS vendors, including AWS and Azure, and look at the benefits of these technologies for reducing the run time of LS-DYNA.

Today's modern HPC cloud systems have the same performance as on-premise clusters since they are built with the same technologies, such as HDR InfiniBand with SR-IOV, AMD EPYC or Intel Xeon Gold Processors. Figure 1. shows a photo of an Azure compute node. Cloud systems essentially are compute power you can rent in another data center on-demand to enhance your on-premise compute capability.

Figure 1. Azure HBv2 Hardware

A notable difference between on-premise HPC and on-demand cloud is that you get billed for the usage of the cloud machines, and it is easy to change hardware. Many clients are interested in getting an individual job result back in a reasonable time frame, while minimizing simulation costs, when using on-demand cloud resources.

With on-premise HPC, a common metric is "jobs per day". On the cloud, there is not a fixed amount of compute resources so in theory one can run an unlimited number of jobs per day. This makes the new limiting factor your simulation budget for renting cloud compute resources. It becomes important to look at the optimal job cost to maximize the amount of simulation workload that can be performed per day in your budget envelope.

This paper will analyze the job cost of the classic LS-DYNA car2car benchmark on both Azure and AWS for common HPC instance types as an illustrative methodology.

Benchmark Systems

The following cloud systems were used for benchmarking. The On-demand pricing is standard list pricing for the regions used in the benchmark. All systems had Hyperthreading disabled.

Table 1. Benchmark System Specifications

Instance Type	Real CPUs Used	Processor	Memory (GB)	Interconnect	IaaS Vendor	On-demand Hourly Cost
Standard_HB120rs_v2	120	AMD EPYC 7V12 64-Core Processor 2.45 GHz	480	HDR IB	Azure	$3.60
HC44rs	44	Intel Xeon Platinum 8168 processor 2.70 GHz	352	HDR IB	Azure	$3.168
c5n.18xlarge	36	Intel Xeon Platinum 8124M 3.50 GHz (All Core Turbo)	192	AWS EFA	AWS	$4.018
c5.18xlarge	36	Intel Xeon Platinum 8124M 3.50 GHz (All Core Turbo)	144	25 Gbps	AWS	$3.18

Model Selection

The car2car benchmark from http://www.topcrunch.org/ uses a NCAC minivan model created by Dr. Makino and analyses a head-on collision between 2 vehicles with around 2,500,000 nodes and elements. This model was chosen as it has been widely benchmarked, even though it is smaller by todays standards.

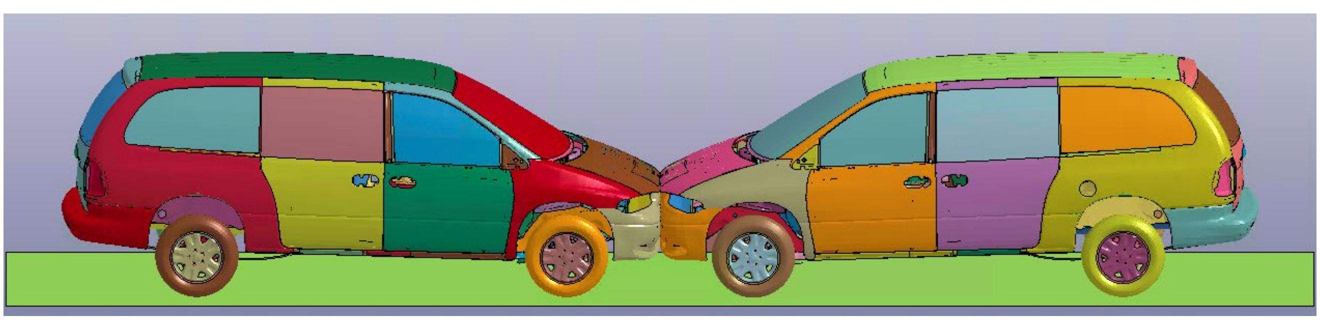

Figure 2. car2car Benchmark Model

Per Job Costs with On-demand Cloud

Most cloud vendors have many costs that can factor into a LS-DYNA job cost such as storage and network egress charges. For the purpose of this job cost analysis, we will analyze the two main cost drivers which are 1) the on-demand compute costs, and 2) the LS-DYNA on-demand software licensing costs. Most other ancillary costs are negligible compared to the compute and licensing cost.

LS-DYNA (and most CAE software today) is licensed by core, with additional cores getting relatively cheaper as you utilize more cores. On-demand compute hardware from Azure and AWS are billed by the hour for the whole compute node, with compute nodes having varying amounts of cores depending on the instance type. Note that using partial cores on a node still results in the whole node being billed, IaaS vendors do not bill for partial cores used.

Job Cost of car2car on Azure

Microsoft Azure is one of the major IaaS providers for HPC, and has two modern HPC instances for running LS-DYNA:

1. HBv2 - These nodes have non-blocking Mellanox HDR InfiniBand, with 120 AMD EPYC 7V12 (Rome) processors cores at 2.45 GHz, 4 GB of RAM per CPU core, and no simultaneous hyperthreading.

2. HC44rs – These nodes have non-blocking Mellanox EDR InfiniBand with 44 Intel Xeon Platinum 8168 processor cores at 2.70 GHz, 8 GB of RAM per CPU core, and no hyperthreading.

These machines must be used in an Azure Virtual Machine Scale Set for the InfiniBand communication support.

Azure testing was done with LS-DYNA MPP R11.1.0 Single Precision. AVX2 binaries were used on AMD HBv2 with Platform MPI, and AVX-512 used on Intel Xeon based HC44rs with Intel MPI 2018. (NOTE: AMD EPYC does not support AVX-512).

It is recommended to use the Azure HPC CentOS 7 images as the basis for your system, as these are preconfigured and tested to deliver maximum performance, scalability, and consistency and include all required drivers.

AMD EPYC has a unique NUMA memory architecture, and some tunings or optimizations required including using Platform MPI options "-cpu_bind=rank,v -e MPI_FLUSH_FCACHE=1,full".

As can be seen from the following elapsed time graphs, the HBv2 instances require more processor cores per job to achieve similar runtimes to the HC44rs instances on car2car.

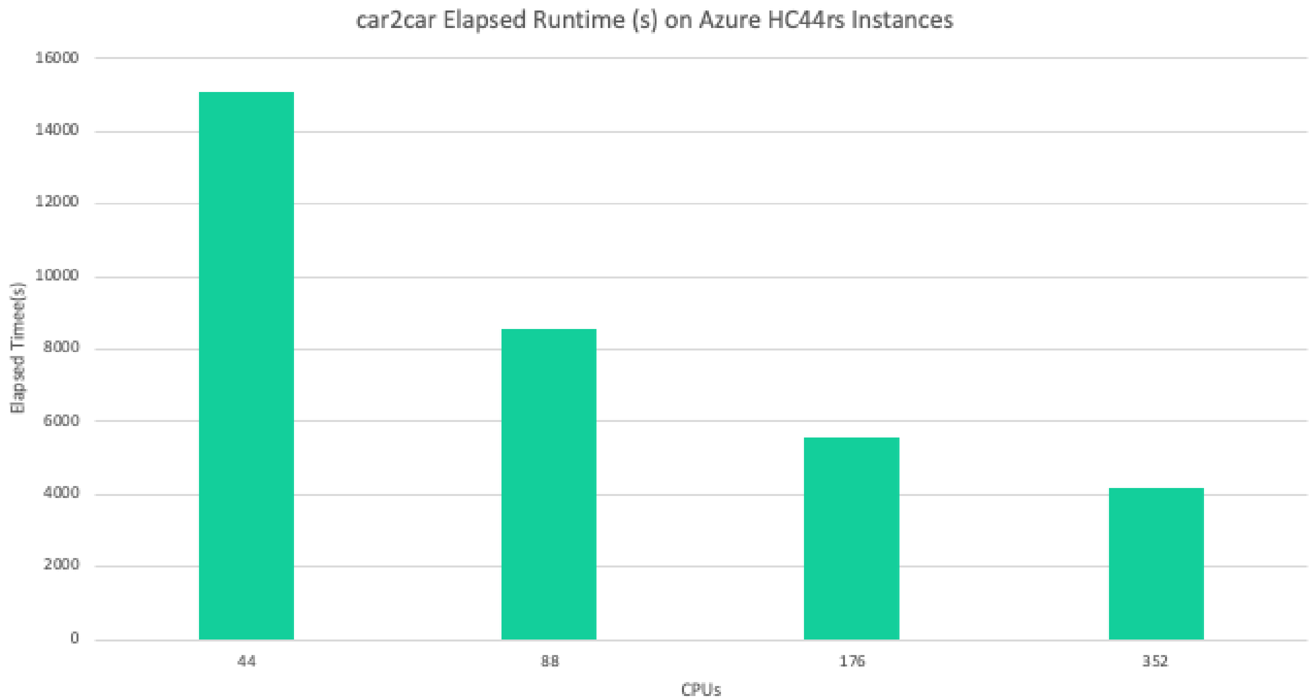

Figure 3. Elapsed Time car2car on HC44rs

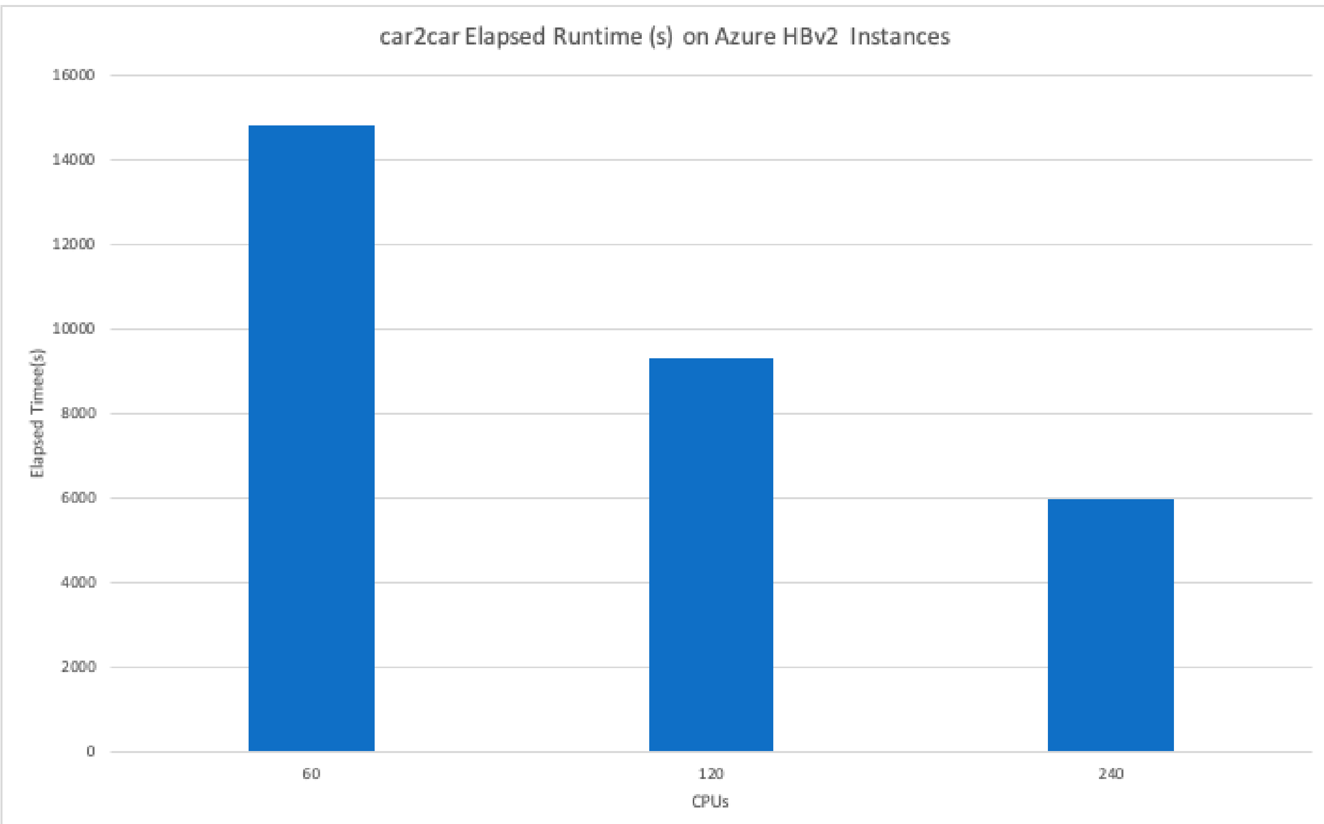

Figure 4. Elapsed Time car2car HBv2

From a pure per-core hour price, the price of AMD HBv2 instances are .03 cents per core hour, when compared with the Intel HC44rs instances at .07 cents per core hour (more than double the price). However, when looking at the actual application job cost below in Figure 5 and 6, the job cost is within 2% for cost and runtime of 240 cores of HBv2 and 175 cores of HC44rs on car2car.

We have found that the HC44rs Intel series have the most flexibility in terms of LS-DYNA versions, LS-DYNA MPI choices, and overall job stability.

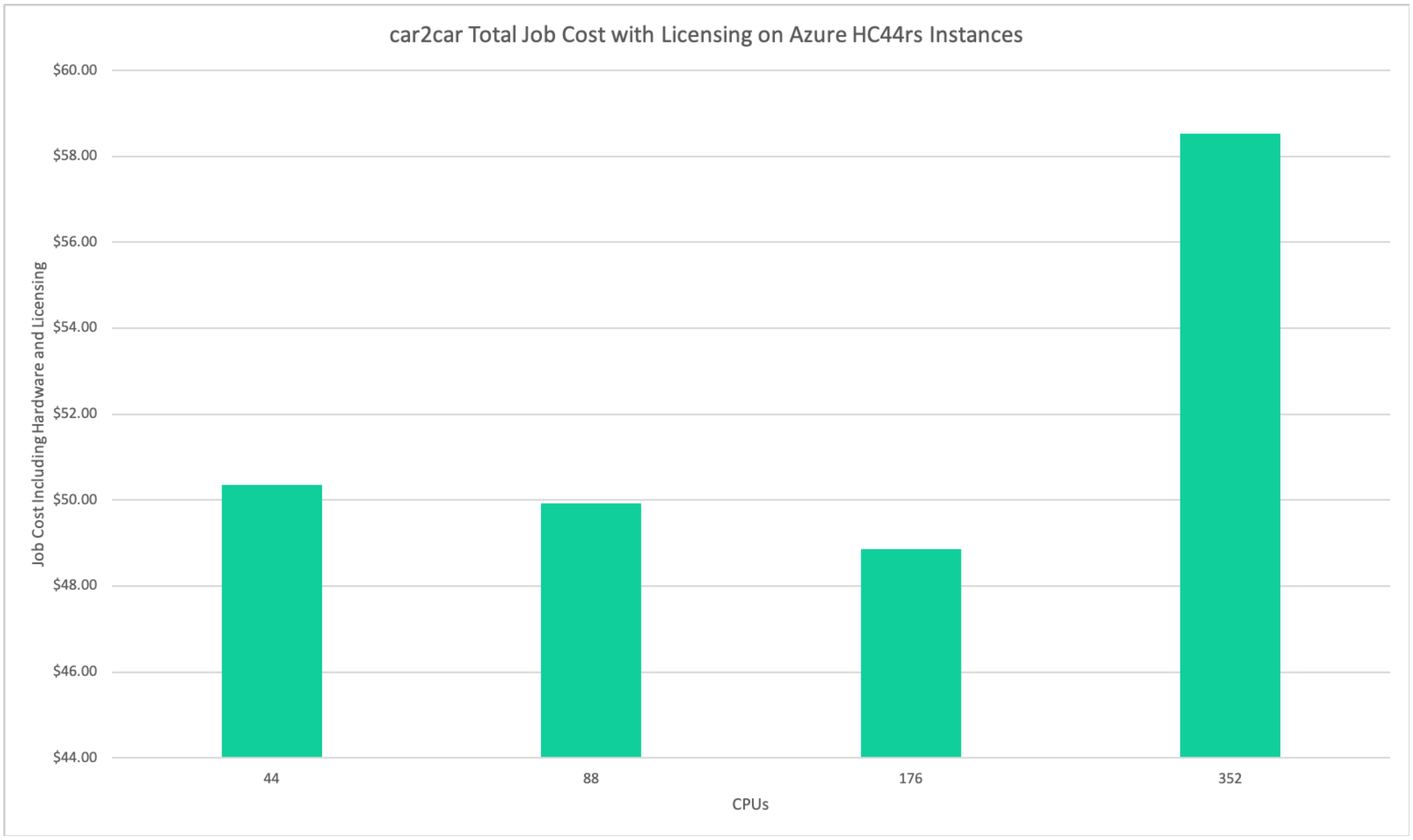

Figure 5. Job cost HC44rs (Compute+License) with car2car

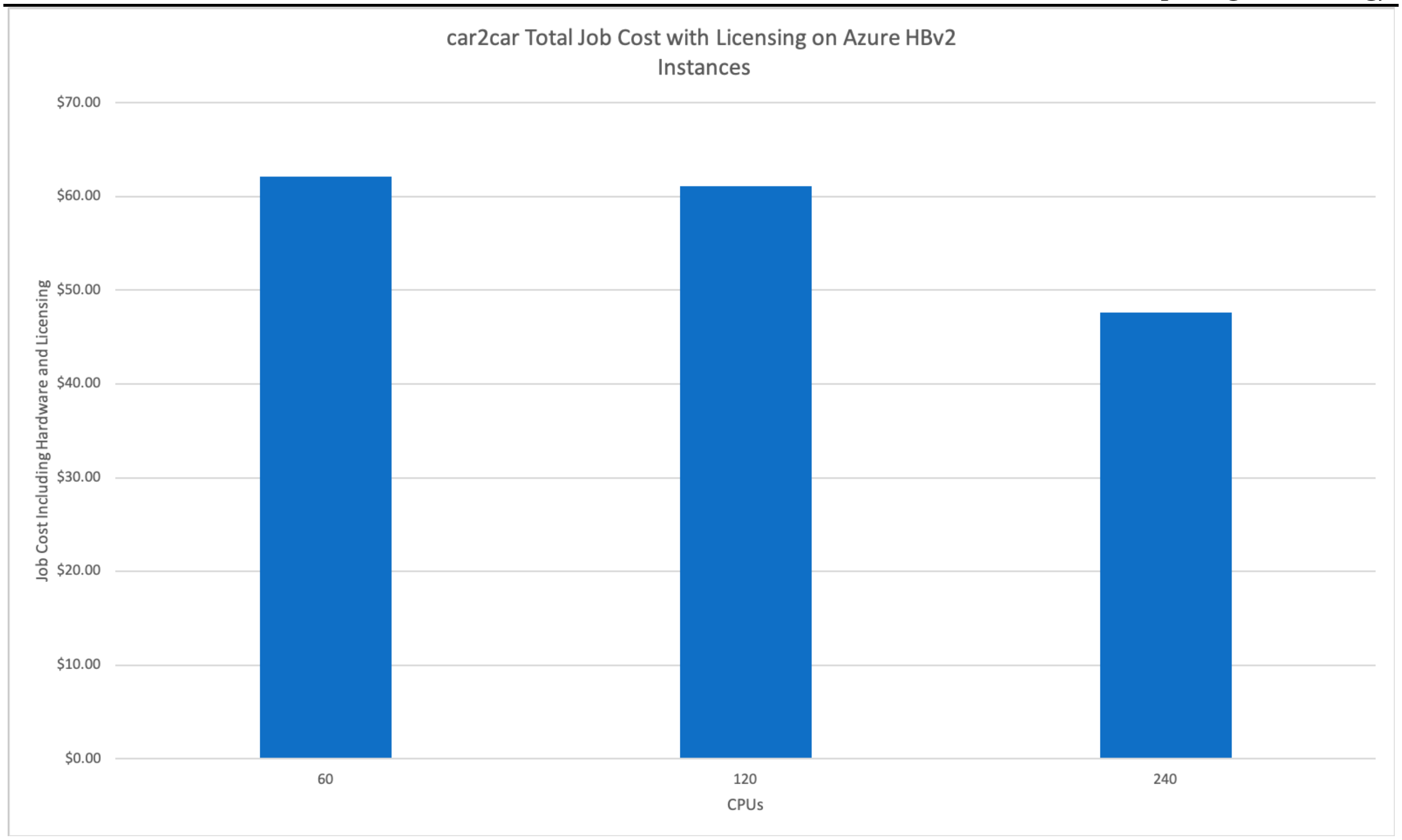

Figure 6. Job Cost HBv2 (Compute+License) with car2car

Job Cost of Car2Car on AWS

Amazon AWS is the market leader for cloud infrastructure, and AWS has many HPC instances capable of running LS-DYNA. The main two price/performance instances for LS-DYNA are:

1. c5n.18xlarge - These nodes feature Amazon Elastic Fabric Adapter (EFA) with 100 Gbps of network bandwidth with consistent low latency communication between instances, 36 real Intel Xeon Platinum processors sustained all core turbo of 3.50 GHz, 5 GB of RAM per real CPU core, and hyperthreading was configured to be disabled (enabled by default).

2. c5.18xlarge– These nodes have 25 Gbps of network bandwidth, 36 real Intel Xeon Platinum processors at sustained all core turbo of 3.50 GHz, 4 GB of RAM per real CPU core, and hyperthreading was configured to be disabled (enabled by default).

AWS testing was done with LS-DYNA MPP R10.2.0 Single Precision. AVX2 binaries were used with Intel MPI 2019U6

The c5n.18xlarge instance was configured to utilize an Elastic Fabric Adapter (EFA) which is a network device to accelerate tightly coupled distributed HPC applications such as LS-DYNA. An EFA device can be added to supported AWS instance types at no extra charge and enables applications such as LS-DYNA to scale to larger core counts. EFA speeds up jobs by reducing communication latency and overhead. All instances had Hyperthreading disabled and were part of the same cluster placement group.

The graph below shows the benefits of utilizing EFA with c5n.18xlarge on scaling. The addition of the EFA adapter allows LS-DYNA to continue to scale Car2Car up to 586 cores, while without EFA, using the standard AWS ENA adapter, LS-DYNA MPI collection communication overhead starts to dominate resulting in a slow down after 144 cores.

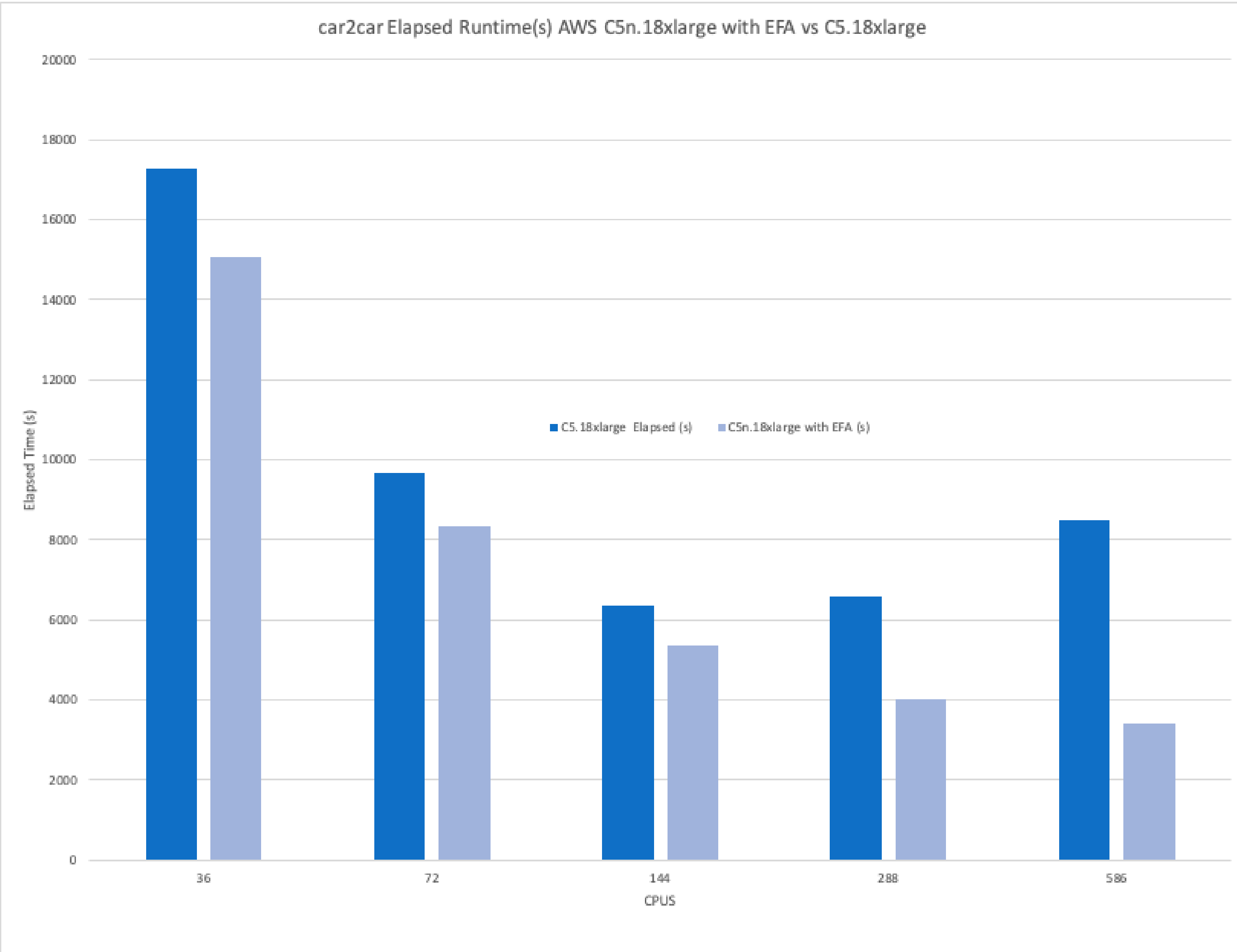

Figure 7. Elapsed Time Savings with AWS EFA on car2car

While there is no additional charge for the EFA adapter, the c5n.18xlarge instance that it requires, is more per hour than c5.18xlarge, but the overall job cost is still lower across the board due to the reduced runtimes EFA enables on car2car. See Figure 8.

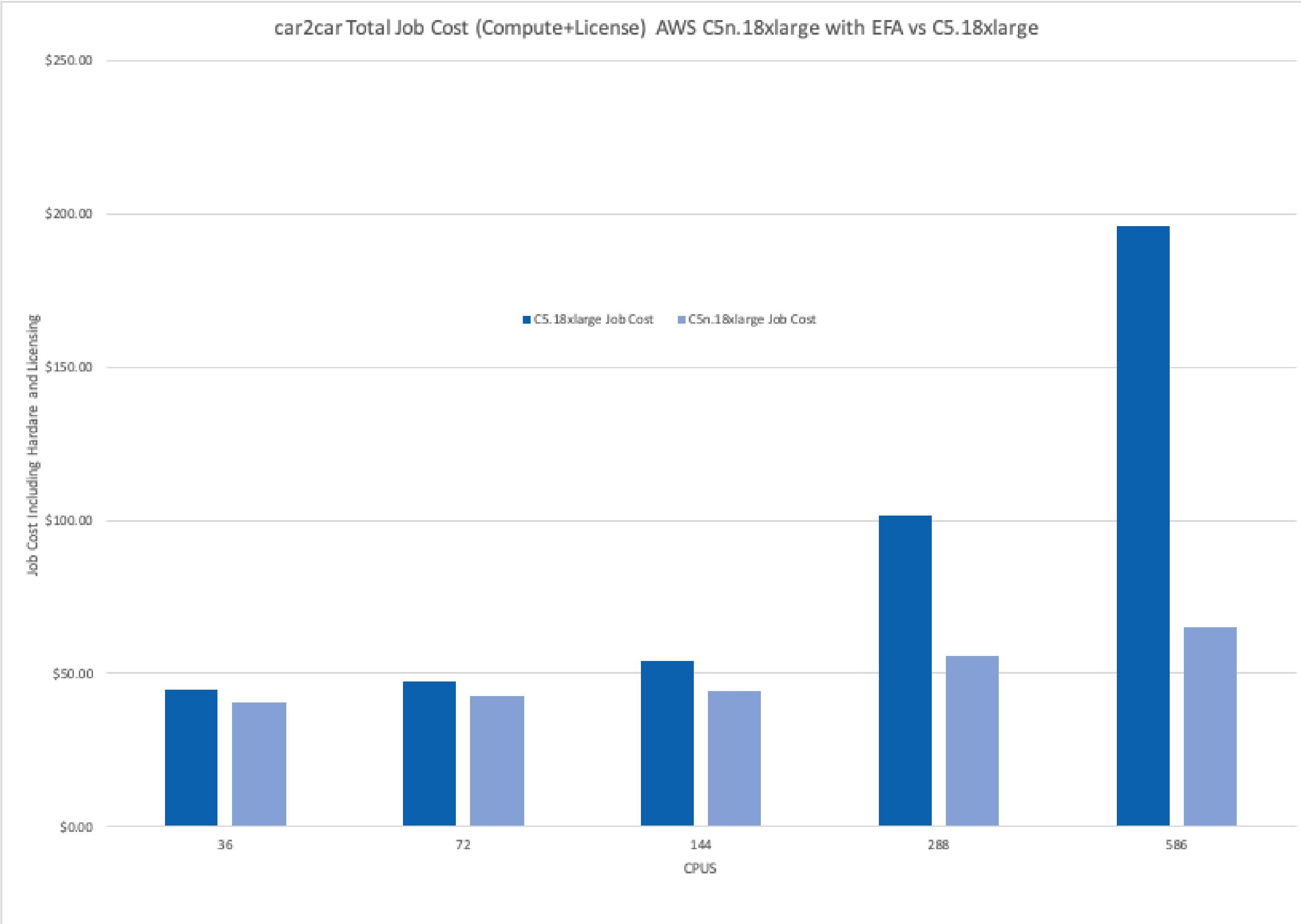

Figure 8. Job Cost Savings with EFA with car2car

Cloud File Systems for LS-DYNA

Temporary scratch I/O on the cloud can include local ephemeral SSD storage, Network File System (NFS) storage, and parallel file systems. AWS offers a fully managed parallel filesystem called Amazon FSx based on the popular open source Lustre filesystem that makes it easy to utilize Lustre without the management complexity. FSx provides the ability to distribute large files automatically and transparently on a single name space filesystem to reduce I/O wait times. TotalCAE compared the usage of FSx to the locally attached ephemeral storage of the compute instances when running LS-DYNA.

Options added to Intel MPI which has native Lustre support are "-genv I_MPI_ADJUST_BCAST=5 -genv I_MPI_EXTRA_FILESYSTEM=on -genv I_MPI_EXTRA_FILESYSTEM_LIST=lustre"

The performance of using Lustre was the same as using locally attached SSD ephemeral disks that are attached to the compute node instances. The locally attached SSD disks do not have additional charges associated with them, and thus lower the overall job cost by utilizing them.

Conclusions

Today's modern HPC cloud systems offer the same performance as on-premise HPC compute with the ability to pay per use. It is important to understand how various HPC cloud technologies affect the per-job cost of your HPC simulation with LS-DYNA, to take best of advantage of your simulation budget when using on-demand cloud resources.

We looked at the most appropriate HPC instances for LS-DYNA from both Amazon AWS and Microsoft Azure, and showed that for car2car HC44rs from Azure and c5n.18xlarge with EFA from AWS offered the best turnaround time, at the lowest job cost.

We also examined new HPC cloud technologies such as Amazon FSx, and Amazon EFA on their performance and impact to car2car simulation job cost and saw benefits to EFA on reduced run time, increased scaling, and lower job cost for LS-DYNA.