EDITOR: Ewa Deelman, deelman@isi.edu

Pushing the Cloud Limits in Support of IceCube Science

Igor Sfiligoi, *University of California San Diego, La Jolla, CA, 92071, USA*
David Schultz, *University of Wisconsin–Madison, Madison, WI, 53706, USA*
Frank Würthwein, *University of California San Diego*
Benedikt Riedel, *University of Wisconsin–Madison*

Scientific high throughput computing needs are growing dramatically with time and public Clouds have become an attractive option for occasional bursts, due to their ability to be provisioned with minimal advance notice. The available capacity of both compute and networking is however not well understood. This article presents the results of several production runs of the IceCube collaboration that temporarily expanded its batch system environment with GPU-providing compute instances from the three major Cloud providers, namely Amazon Web Services, Microsoft Azure, and the Google Cloud Platform. The aim of these Cloud bursts was to push the limits of Cloud compute, with a particular emphasis on GPU-providing instances. On the compute side, we showed that it is possible to reach peaks of over 380 fp32 PFLOPS using all available GPU-providing instance types and integrate over 1 fp32 EFLOP hour in a single workday by using only the most cost-effective ones. On the network side, we showed intra-Cloud network throughputs of over 1 Tbps, and 100 Gbps throughputs toward on-prem storage both using shared peering arrangements and dedicated network links.

Cloud computing has become mainstream in many commercial environments, but it is still marginal in scientific high throughput computing (HTC). There are obviously many reasons for this situation, but one important aspect is the lack of understanding of the available compute and networking capacity in commercial Clouds. While there have been some recent large scientific computing runs in the Clouds,[1] none attempted to maximize the provisioned amount of resources across all the major Cloud providers concurrently, namely Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform (GCP).

We decided to focus on GPU resources, due to both their high performance and their relative scarcity in the scientific on-prem compute environments. And since large-scale HTC is expensive, both on-prem and in the Clouds, we used a production scientific workload for most of the exploration work to maximize return on investment. The selected workload was IceCube's photon propagation simulation, used for its detector simulation, both for scientific reasons (high impact science) and their experience with GPU-based HTC.[2–4]

We initially explored the available compute capabilities of the Cloud providers, by picking the most compute-intensive subset of IceCube's workload and keeping most of the network traffic inside the Cloud networking domain. This work has been widely disseminated over the past year,[5,6] but we provide a summary of the procedure and the results in the next section for completeness.

However, most scientific applications are data-driven, so our recent activity has focused on characterizing the available Cloud networking, with an emphasis on cost-effective dedicated network links.

1089-7801 © 2021 IEEE
Digital Object Identifier 10.1109/MIC.2020.3045209
Date of current version 3 February 2021.

We selected a much more data-intensive subset of IceCube's workload and moved the data synchronously to and from on-prem storage systems, as is the norm when using on-prem compute resources. The procedure and results are presented in the second half of this article.

AVAILABLE COMPUTE RESOURCES

Cloud providers like to publicize the elastic nature of their Cloud infrastructure, with an implied suggestion that they can accommodate an infinite amount of computing work. This is impossible, so we set out to measure just how large the available capacity is. We were particularly interested in GPU-providing instance types, due to their high performance, and were happy to use resources located anywhere on the planet.

As mentioned in the introduction, we used IceCube's production workload during this exploration step, although we did limit ourselves to only the most compute-intensive subset of it. IceCube's production setup uses HTCondor[7] as the batch system, with the compute resources coming partially from local on-prem infrastructure and partially from remote systems, dynamically provisioned through the Open Science Grid (OSG).[8,9] Extending the provisioning to Cloud resources was thus just a minor operational change; we opted to host a separate batch queue mostly due to the order of magnitude higher scale.

IceCube normally does not run on Cloud resources, so we did not have an existing provisioning infrastructure in place. Given the exploratory nature of the exercises, we thus provisioned the resources directly using the native Cloud mechanisms. After creating the base virtual machine (VM) images using the standard OSG-provided worker node software, the actual large-scale provisioning was delegated to native group provisioning mechanisms, namely Fleets on AWS, VM scale sets (VMSS) on Azure, and Instance Groups on GCP. Note that while the three Cloud providers use different implementations, the operational semantics is quite similar among the three. It should also be noted that each region in each Cloud provider is essentially independent, so we had to set up and operate this infrastructure in 28 independent environments.

The first Cloud burst[5] was executed in November 2019, using all available GPU-providing instance types from the three Cloud providers and using a mix of on-demand, spot, and preemptible pricing, as recommended by the respective capacity planning teams. We were able to reach a peak of about 52k GPUs in about 2 hours (see Figure 1), using eight different GPU types for an equivalent theoretical compute throughput of about 380 fp32 PFLOPS (see Figure 2).

A follow-on Cloud burst[6] was executed in February 2020, with a longer sustained plateau and using only the most cost-effective Cloud instances in spot mode, which kept the Cloud costs at under $60,000 (note: We are not authorized to disclose the cost of the first Cloud burst). The peak value reached was of course lower, about 180 fp32 PFLOPS (see Figure 3), but we still integrated approximately 1 fp32 EFLOP hour. Looking at the science output, we produced about 50% more files than in the first run. In the process, we also measured the amount of preemption incurred using spot instances, which was under 10% even at such high scales (see Figure 4).

Between the two Cloud bursts, we demonstrated that large scale HTC in Clouds is possible, and can greatly benefit compute-intensive science computing, when there is a need for additional resources not available on-prem.

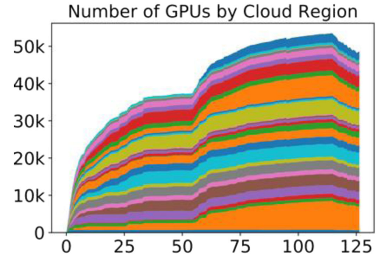

FIGURE 1. Number of GPU instances over time (in minutes) during the first Cloud burst ramp-up period. (Used, with permission, from.[5])

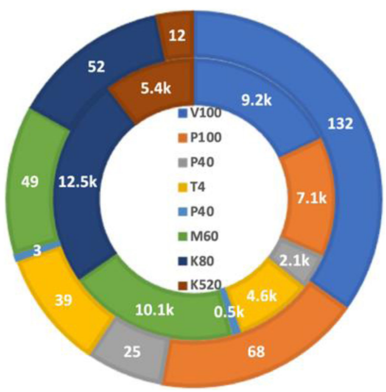

FIGURE 2. GPU composition in the first Cloud burst at peak. The inner circle shows the number of instances, the outer circle the fp32 PFLOPS contribution. (Used, with permission, from.[5])

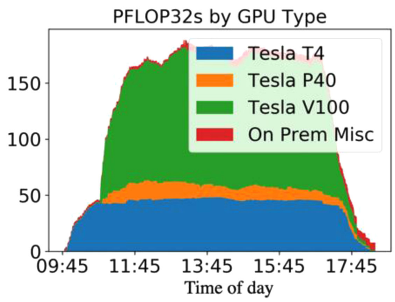

FIGURE 3. Provisioned Cloud resources in the second Cloud burst, alongside on-prem resources. (Used, with permission, from.[6])

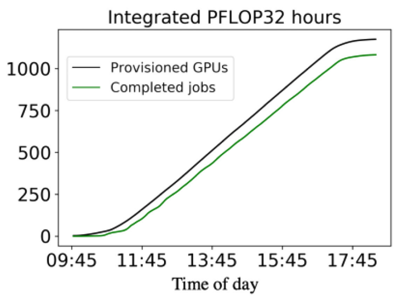

FIGURE 4. Difference between provisioned GPUs in spot mode and jobs that ran to completion during the second Cloud burst. (Used, with permission, from.[6])

AVAILABLE NETWORK RESOURCES

Many scientific computing problems are data-driven, which implies that one needs excellent network performance in order to make full use of the compute hardware. In preparation for the first Cloud burst,[10] we verified that networking inside Cloud provider's infrastructure was more than adequate, measuring in excess of 1 Tbps in a single region (see Figure 5) and at least 200 Gbps between major regions. We now wanted to demonstrate that the same was possible when accessing data in on-prem storage from the Clouds.

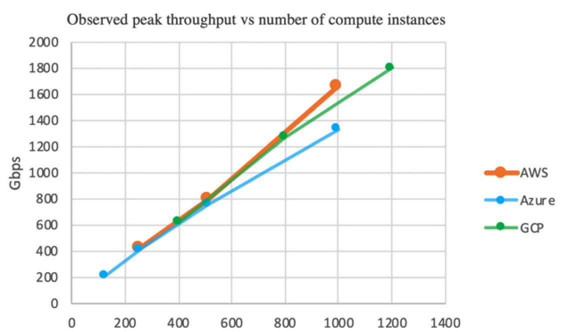

FIGURE 5. Peak throughput observed in a Cloud region while downloading from a local object storage instance. (Used, with permission, from.[10])

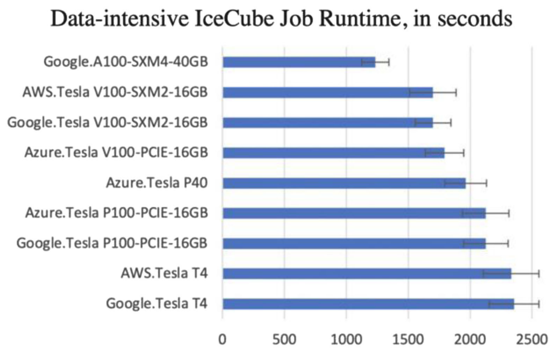

FIGURE 6. Average runtime of the data intensive IceCube photon propagation simulation jobs, per instance type. The output file size was on average 2.3 GB in all cases.

IceCube's main storage system is located at the University of Wisconsin–Madison (UW). The storage system is configured as a distributed Lustre filesystem, with several gateway nodes for wide area network (WAN) connectivity. UW is connected to the Cloud through a 100 Gpbs research WAN link, while the theoretical throughput of the storage system is significantly higher than that, making networking alone the bottleneck.

In order to evaluate the feasibility of real data-intensive IceCube photon simulation workloads, we picked the appropriate subset in its production queues and measured the compute runtimes and data sizes of a modest sample on Cloud resources. We observed that runtimes varied between approximately 20 and 45 minutes, depending on GPU used (see Figure 6), with an average input of 300 MB and output of 2.3 GB, which yields an average network throughput of about 10 Mbps per GPU. We would thus need approximately 10,000 GPUs to reach a sustained 100 Gbps network flow, which seemed achievable.

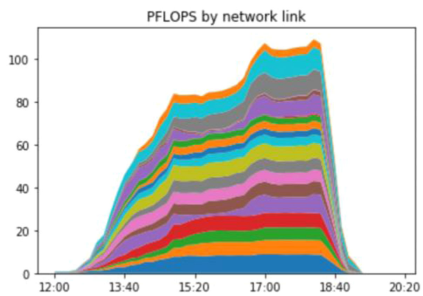

FIGURE 7. Provisioned Cloud resources in the data-intensive Cloud burst.

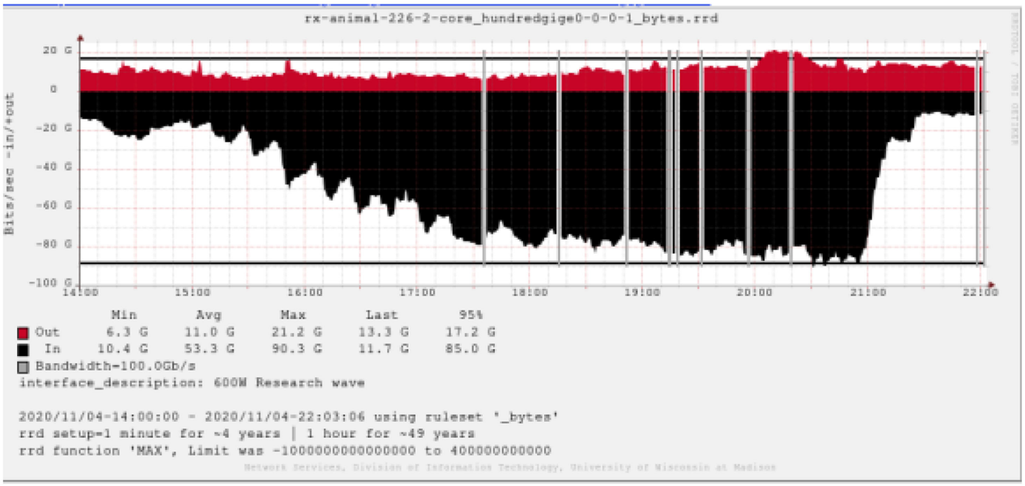

FIGURE 8. Screenshot of the UW research WAN link monitoring Web page.

We demonstrated a 100-Gbps file transfer between UW and the Cloud providers in February 2020, with short data transfer bursts using simple test instances.[11] Similar tests also demonstrated 100-Gbps file transfer capabilities to other on-prem systems but also confirmed the high cost of egress network traffic using the standard peering routes, at over $80/TB. That would make standard egress costs significantly higher than compute costs for T4-providing instances during a production run, at approximately $0.18 vs $0.11 per job in pre-emptible mode.

To keep egress network costs reasonably low, we thus decided to provision dedicated links for the data-intensive Cloud burst; Cloud providers charge significantly lower per-TB cost on dedicated links, for a fixed per hour fee. In the USA, the Internet2's Cloud Connect service[12] acts as a network provider for all three major Cloud providers, namely AWS, Azure, and GCP, with a fixed set of physical links in place for routing toward supported research networks. This allows various academic institutions to dynamically provision logically dedicated network links, with the associated reduced costs, without the need to change the physical infrastructure. The process does however still need the involvement of on-prem network engineers.

We executed the data-intensive Cloud burst in November 2020, by first provisioning 22 dedicated network links and subsequently provisioning approximately 100 fp32 PFLOPS of compute from the Cloud providers (see Figure 7). The run lasted about 6 hours, during which we integrated about 220 PFLOP hours of compute and produced 130 TB of data. The total network cost for the day was approximately $6000, which is about half of what we would have paid if we went the normal routing path.

On the networking side, we reached approximately 100 Gbps between the Clouds and on-prem storage, with about 80 Gbps going to UW and the rest to storage provisioned at University of California San Diego. At peak, the UW research WAN link was over 90% full (see Figure 8); while we did not use all of the bandwidth, we were responsible for the vast majority of it.

The main challenge in this last exercise was appropriately spreading the load over the 22 provisioned links, most of which were 5 Gbps, with five 10 Gbps and six 2 Gbps links. This was particularly challenging due to the spiky nature of the IceCube workload, where the whole output is uploaded to on-prem storage immediately after the compute is finished. Given the unpredictable nature of spot Cloud resource availability, we had to ramp up slowly to randomize as much as possible the upload times. This strategy proved successful, as seen in relatively smooth bandwidth use in the second part of the run.

CONCLUSION

By executing three independent production Cloud bursts in support of IceCube science mission, we demonstrated that it is possible to provision large amounts of compute capabilities from the commercial Cloud providers in rapid and/or cost-effective manner. While running compute-intensive high throughput workloads is certainly easier, data-intensive workloads are also feasible and can be executed in a cost-effective way with some additional setup.

While Cloud resources are not infinitely elastic, we were able to provision over 375 fp32 PFLOPS of GPU compute. Cloud computing is also not free, but we showed that $60,000 can buy 1 fp32 EFLOP hour of useful compute. Finally, data-intensive applications can easily scale to 100 Gbps data transfer rates, but egress-focused applications should consider dedicated links to minimize the network-related costs.

Our work shows that Cloud computing can be appealing for scientific endeavors that have tight deadlines, as it allows for much quicker time to solution, albeit likely at a higher cost than spreading the compute over longer periods of time on on-prem equipment.

ACKNOWLEDGMENTS

This work was supported in part by the US National Science Foundation (NSF) under Grant OAC-1941481, Grant MPS-1148698, Grant OAC-1841530, Grant OAC-1826967, and Grant OPP-1600823. We thank M. Hare, V. Brik, S. Barnet, T. Hutton, M. Zekauskas, and J. Hicks for their support.

REFERENCES

1. B. Posey *et al.*, "On-demand urgent high performance computing utilizing the google cloud platform," in *Proc. IEEE/ACM HPC for Urgent Decis. Mak. UrgentHPC*, 2019, pp. 13–23.
2. M.G. Aartsen *et al.*, "The IceCube neutrino observatory: instrumentation and online systems," *J. Instrum.* vol. 12, pp. P03012–P03012, 2017.
3. D. Chirkin, "Photon tracking with GPUs in IceCube," *Nucl. Inst. Meth. Phys. Res. Sec. A*, vol. 725, pp. 141–143, 2013.
4. E. Fajardo *et al.*, "OSG and GPUs: A tale of two use cases," in *Proc. EPJ Web Conf.*, vol. 214, 2019, Art. no. 03034.
5. I. Sfiligoi, F.Würthwein, B. Riedel, and D. Schultz, "Running a pre-exascale, geographically distributed, multi-cloud scientific simulation," in *Lecture Notes Computer Science*, Springer, Cham, 2020, vol. 12151.
6. I. Sfiligoi *et al.*, "Demonstrating a pre-exascale, cost-effective multi-cloud environment for scientific computing," in *Proc. Pract. Exp. Adv. Res. Comput.*, Jul. 2020, pp. 85–90.
7. Home page, "Computing with HTCondor," [Online]. Available: https://research.cs.wisc.edu/htcondor/
8. R. Pordes *et al.*, "The open science grid," in *Proc. J. Phys., Conf. Ser.*, 2007, Art. no. 012057.
9. Home page, "The open science grid," [Online]. Available: https://opensciencegrid.org
10. I. Sfiligoi, J. Graham, and F. Wuerthwein, "Characterizing network paths in and out of the clouds," in *Proc. EPJ Web Conf.*, vol. 245, 2020. Art. no. 07059.
11. I. Sfiligoi, "Demonstrating 100 Gbps in and out of the public Clouds," in *Proc. Pract. Exp. Adv. Res. Comput.*, Jul. 2020, pp. 495–499.
12. Internet2, "Cloud connect," [Online]. Available: https://internet2.edu/services/cloud-connect/

IGOR SFILIGOI is with the San Diego Supercomputer Center, as part of the University of California San Diego, La Jolla, CA, USA. Over the years, he has been engaged in the computing community across a wide breath of roles, participating among others as Researcher, Operator, Developer, Architect, and Manager. He has been working in both academia and industry, which helps him appreciate the variety of problems computers help to solve. He has been working with both systems that support tightly coupled and worldwide-scale pleasantly parallel codes and realizes that each approach has its own advantages and challenges. In his current role, he is putting his extensive knowledge to use across many scientific domains, spanning computational biology, fusion research, high energy physics, and astrophysics. He received the equivalent of the M.S. degree in computer science from the Universita degli Studi di Udine, Udine, Italy. He is the corresponding author of this article. Contact him at isfiligoi@sdsc.edu.

DAVID SCHULTZ is with the Wisconsin IceCube Particle Astrophysics Center, the University of Wisconsin–Madison, Madison, WI, USA. He is currently the Lead of the IceCube software infrastructure team and is deeply involved in developing and maintaining the software services for the IceCube collaboration. He received the M.S. degree in computer science from Georgia Institute of Technology, Atlanta, GA, USA. Contact him at david.schultz@icecube.wisc.edu.

FRANK WÜRTHWEIN is a Professor of Physics with the University of California San Diego, La Jolla, CA, USA. He is also high-throughput Computing Lead with the San Diego Supercomputer Center and the Executive Director of the Open Science Grid, a national cyberinfrastructure to advance the sharing of resources, software, and knowledge. His research focuses on experimental particle physics and distributed high-throughput computing. He received the Ph.D. degree in physics from Cornell University, Ithaca, NY, USA. Contact him at fkw@ucsd.edu.

BENEDIKT RIEDEL is with the Wisconsin IceCube Particle Astrophysics Center, the University of Wisconsin–Madison, Madison, WI, USA. He is the Computing Manager for both the IceCube Neutrino Observatory and Wisconsin IceCube Particle Astrophysics Center. He received the Ph.D. degree in physics from the University of Wisconsin–Madison. Contact him at benedikt.riedel@icecube.wisc.edu.