Comparing Burstable and On-Demand AWS EC2 Instances using NAS Parallel Benchmarks

Gian Carlo F. Ferrari[1], **Vanderlei Filho**[1], **Márcio Castro**[1]

[1]Federal University of Santa Catarina (UFSC)
Florianópolis/SC

gian.cff@grad.ufsc.br, vanderlei.filho@proton.me,
marcio.castro@ufsc.br

Abstract. *Amazon Web Services (AWS) offers burstable instances, which have a baseline CPU performance with the ability to burst above the baseline as long as is required. In this paper, we evaluate the performance and monetary costs of NAS Parallel Benchmarks running on AWS burstable and standard virtual machines. Our results show that burstable instances can perform similarly with lower or similar monetary costs for some parallel applications.*

1. Introduction

Amazon Web Services (AWS) offers different instance configurations and pricing models. Pricing models include *On-demand (but non-burstable)* and *spot*. *Non-burstable instances* allow users to pay for compute capacity by the hour or second with no long-term commitments and fixed price with availability guaranteed by AWS. On the other hand, *spot instances* are spare unused AWS instances that cost considerably less and run on the same type of infrastructure with identical performance and characteristics. However, AWS can revoke spot instances from users on short notice when capacity needs to be reclaimed for on-demand instances.

Another model are *Burstable instances*, which provide a baseline level of CPU utilization with the ability to burst CPU utilization above the baseline level. This ensures that users pay only for baseline CPU plus any additional burst CPU usage resulting in lower compute costs. The baseline utilization and ability to burst are governed by CPU *credits*: CPU usage lower than baseline accumulates credits and usage above consumes credits. There are currently two *credit specifications*: *standard*, which throttles the CPU back to baseline if there are no credits available and *unlimited*, which accumulates a credit debt instead, and if the CPU usage is bursting without spare credits for a 24-hour period, a flat rate is charged on top of the on-demand hourly price.

In this paper, we evaluate the use of AWS burstable instances for running High Performance Computing (HPC) workloads from the NAS Parallel Benchmarks (NPB), both in terms of performance and monetary costs. We compare the results achieved with burstable instances and non-burstable ones. We also evaluate both standard and unlimited credit specifications of burstable instances to observe how they would affect performance and cost when running the benchmarks.

2. Infrastructure and Applications

In this section, we first give details about the AWS instances used in the experimental evaluation. Then, we briefly discuss the parallel workloads employed in this research.

Table 1. Overview of the instance configurations.

Hardware Description	Non-burstable	Burstable
Intel Xeon Platinum 8259CL @ 3.1 GHz	M5	T3
AMD EPYC 7571 @ 2.5 GHz	M5a	T3a
AWS Graviton2 Processor with 64-bit Arm Neoverse cores	M6g	T4g

2.1. AWS Instance Types

To make a fair comparison, we selected AWS instance types that have the same hardware configuration for both non-burstable (M5, M5a and M6g) and burstable (T3, T3a and T4g) pricing models. All instances were of size *2xlarge*, which means they have 8 vCPUs (4 physical threads + hyperthreading) and 32 GB RAM. This instance size was chosen as a middle ground between accommodating benchmark requirements (for the size of the benchmark chosen, which is class C)[1], simulating a large work load, and running in an adequate amount of time. Table 1 presents an overview of the configurations[2] used in this study. Although M5 allows for use of AVX-512, this requires flags to be set at compile-time or the addition of special AVX instructions in the source code. To make a fair comparison with its burstable counterpart (T3), which does not offer AVX-512, we did not compile NPB with AVX-512 support.

2.2. Parallel Workloads

In order to evaluate AWS burstable and non-burstable instances with a diverse range of HPC workloads, we carried out experiments with all the eight original kernels available from the Message Passing Interface (MPI) version of NPB: IS, EP, CG, LU, SP, MG, BT and FT (more details about these kernels can be found in [Bailey et al. 1994]). NPB was compiled via GCC v11.4.1 with Open MPI v5.1.0a1 and v4.1.2 on x86-based (T3 and M5) and ARM-based (M6 and T4) instances, respectively. For each AWS instance type (including two credit specifications of burstable instances), we carried out five runs of each kernel with four MPI processes bound to the available physical cores of a single AWS instance. We employed the HPC@Cloud toolkit [Munhoz and Castro 2023] to build the infrastructure on AWS, and implemented a custom-made bash script to automate the execution of the kernels as well as the data gathering (execution time).

3. Experimental Results

Figure 1 shows the average execution time (in seconds) of each NPB kernel obtained on AWS burstable and non-burstable instances. Burstable instances are shown with their credit specification variant next to their names ("S" for *standard* and "U" for *unlimited*). The results indicate that, for the most part, burstable instances running on unlimited credit specification can perform very similarly to their non-burstable counterparts. Attention is drawn to EP, where the performance across both specifications of credit and non-burstable is very similar, with a modest deviation. EP is described in [Bailey et al. 1994] as, amongst other things, having no significant inter-processor communication. Thus it seems that performance-wise, a CPU-bound workload can run similarly in all instances.

[1]A quick overview of benchmark classes can be seen at https://www.nas.nasa.gov/software/npb.html, and more detail about their sizes at https://www.nas.nasa.gov/software/npb_problem_sizes.html.

[2]More details can be found at https://aws.amazon.com/pt/ec2/instance-types/.

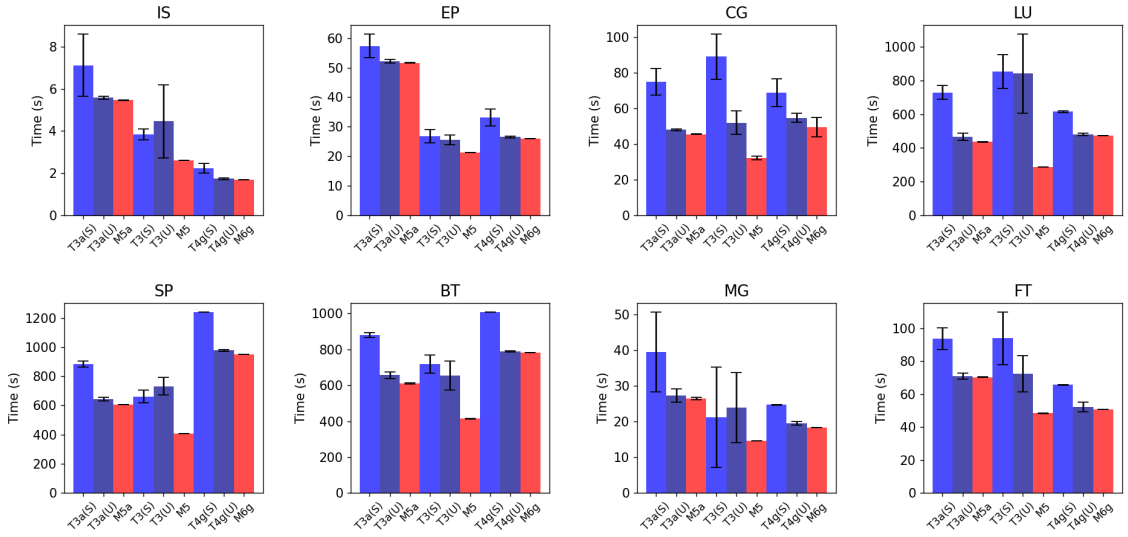

Figure 1. Execution time (in seconds) comparison of all NPB kernels running on AWS burstable and non-burstable instances.

Instance	M5	T3	M5a	T3a	M6g	T4g
Hourly price (USD)	0.384	0.3328	0.344	0.3008	0.308	0.2688

Table 2. Instance hourly prices for size *2xlarge*.

In contrast, MG is described as testing short and long distance communication, while also having greater deviation for T3a(S), T3(S) and T3(U) run times. Both CG, FT and IS also test communication, with varying degrees. In these kernels, a greater deviation can also be observed on some of the instances. Thus it appears that communication-bound tasks (or even ones with balanced CPU and communication usage as the IS description suggests) would have less predictable running times in burstable instances.

To properly compute the approximate monetary cost of running each kernel on each AWS instance, we converted the hourly price of each AWS instance, obtained from AWS, to USD cents per second. Then, we multiplied it by the average execution time (in seconds) of each NPB kernel. Figure 2 presents these results. The standard deviation was calculated by computing the price for each kernel execution.

The monetary cost of EP suggests that CPU-bound tasks cost around the same price for both burstable and non-burstable, without much variation. Communication-bound tasks (represented by the CG, MG and slightly by the IS benchmarks), however, maintain their higher deviation and seem to be able to go either way regarding price. Thus we can not predictably conclude that, despite their lower hourly prices, the running of these types of tasks in standard specification burstable instances would be cheaper.

Interestingly, despite lower hourly prices (see table 2)[3], some of the costs of running the benchmarks are higher in burstable instances than on non-burstable. The most apparent cases of running on burstables being cheaper than on non-burstable instances are with the unlimited specification, suggesting that throttling the CPU during runtime leads to less predictable and sometimes more expensive costs.

[3] As seen on https://aws.amazon.com/pt/ec2/pricing/on-demand/.

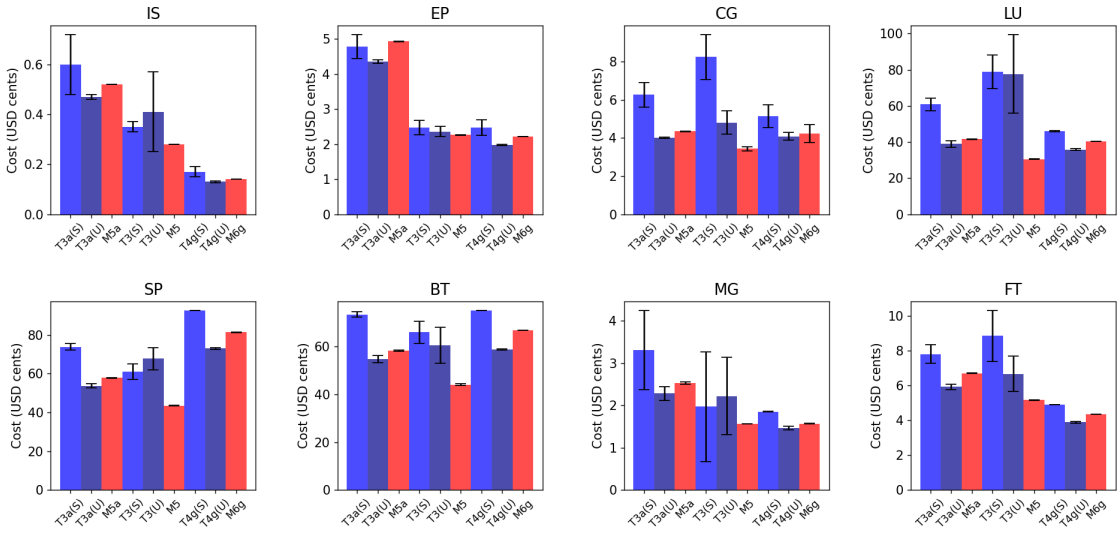

Figure 2. Approximate monetary costs (in USD cents) for running all NPB kernels on AWS burstable and non-burstable instances.

4. Conclusions

We presented an initial study of the performance and monetary costs of running HPC workloads on AWS burstable instances in comparison to non-burstable instances. We carried out experiments with all kernels available from the MPI-based version of NPB on three different burstable and non-burstable AWS instances. Our results showed that for single instances, communication overhead ends up making both runtime and cost more unpredictable for burstable instances, while CPU-bound ones seem to be more predictable present similar costs.

As future works, we intend to extend the performance and monetary costs analysis by using clusters of two or more instances built on top of AWS. We believe that these scenarios will be more interesting to the burstable pricing model, since the communication overhead will be long enough to allow burstable instances to accumulate credits that will be used for compute-intensive phases of the kernels. The running of more demanding versions of these benchmarks would also allow us to observe what the 24-hour flat rate mentioned earlier would do to the prices of unlimited specification burstable instances in comparison to their non-burstable counterparts.

Acknowledgements

This work was partially funded by the National Council for Scientific and Technological Development (CNPq) and Amazon Web Services (AWS) through the CNPq/AWS Call No 64/2022 – Cloud Credits for Research.

References

Bailey, D., Barszcz, E., Barton, J., Browning, D., Carter, R., Dagum, L., Fatoohi, R., Fineberg, S., Frederickson, P., Lasinski, T., Schreiber, R., Simon, H., Venkatakrishnan, V., and Weeratunga, S. (1994). The nas parallel benchmarks. Technical report, NASA.

Munhoz, V. and Castro, M. (2023). Enabling the Execution of HPC Applications on Public Clouds with HPC@Cloud Toolkit. *Concurrency and Computation: Practice and Experience*, pages 1–19.