Benchmark of GPU-accelerated bioinformatics methods for processing raw RNA-seq data

Etienne BARDET[1], Pauline BAZELLE[1], Nicolas WIART[2], Christophe BATTAIL[1] and KATY consortium[3]

[1] Laboratoire Biologie et Biotechnologies pour la Santé, IRIG, UMR 1292 INSERM-CEA-UGA, Univ. Grenoble Alpes, 38000 Grenoble, France.
[2] Centre National de Recherche en Génomique Humaine, CEA, Université Paris-Saclay, 91057 Evry, France.
[3] https://katy-project.eu, European Unions's Horizon 2020 research and innovation programme, Grant agreement No 101017453

contact: christophe.battail@cea.fr

Abstract

The emergence of **personalized medicine** requires being able to produce and **process huge amounts of biological data** generated from a patients' biological samples, in a **quick manner and at a reasonable cost**.

While **modern sequencing technologies** have keep up with these need, and are now able to **produce large amount of data** in record time, bioinformatics tools still have to make this transformation.

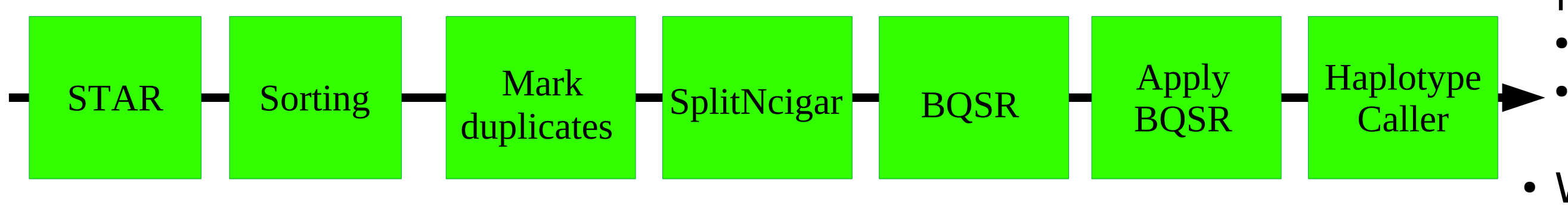
Indeed, most bioinformatics methods focus more on **the accuracy of their results** than on the **speed of their execution**. This creates a situation where bioinformatics analysis can **create a bootlneck.** To remedy that problem, we have to look at ways to **speed up the analysis.**

NVIDIA, one of the world's largest GPU manufacturer recently released the 3rd version of it's **Clara Parabricks suite**, which **accelerates populars bioinformatics tools** by allowing them to **use GPU** for theirs calculations. However, Parabricks has only been **independently benchmarked** on it's ability to handle **genomic data** [1] and not RNAseq data. We thus propose to **benchmark parabricks on RNAseq data**.

Material and methods

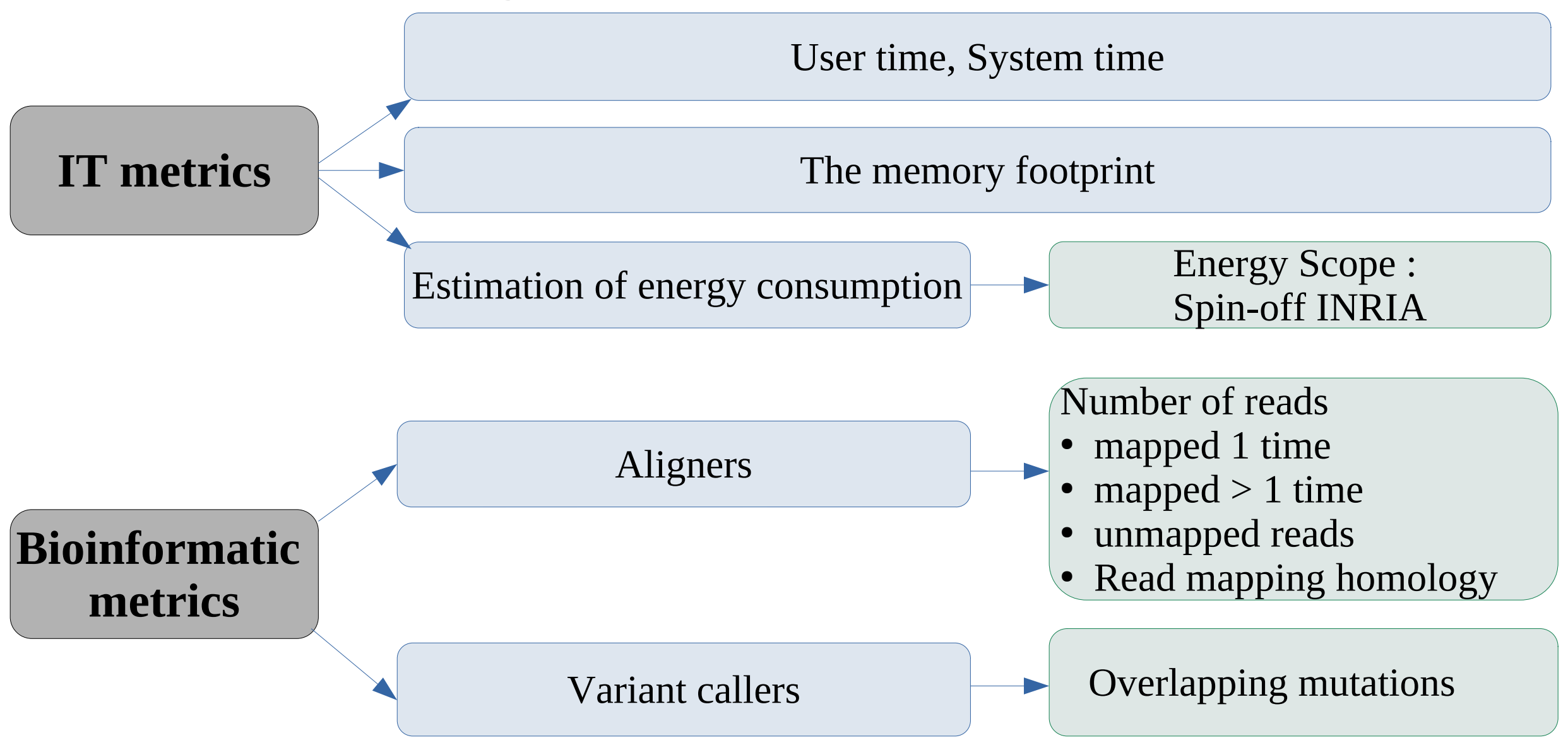
WE BENCHMARKED:

- The whole Parabricks **Built in RNA Pipeline** [2]



- The Individual **read aligners** and **variant callers** implemented in Parabricks

We used the **following metrics:**



TECHNICAL DETAILS:

- We ran tools and pipelines on the following configurations:
  - **16** and **32 CPU 2.30GHz**
  - **4 NVIDIA Volt GPU**
- We used the University Grenoble Alpes **GRICAD** cluster.
- Individual tools were installed and executed via their **official docker image** (if available) and **custom made singularity images**.
- The **reference CPU pipeline** that we used to benchmark the RNA GPU pipeline was **developed in-house using the Common Workflow Language (CWL) and executed by cwlTool**.
- Tests were performed using **single-end RNAseq data** of kidney cancer samples [2].

Results: Read aligners

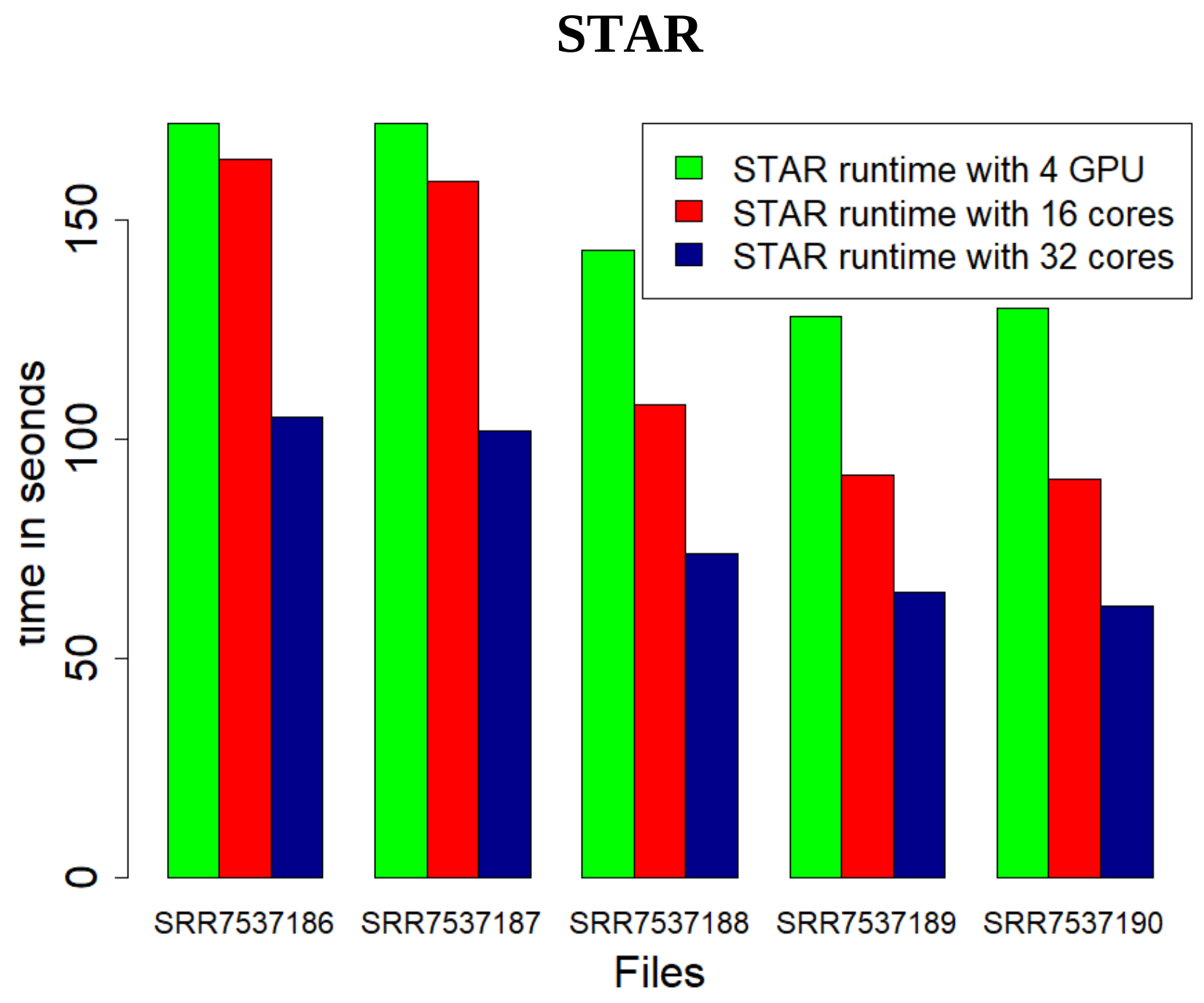


Figure 1: Comparison of STAR's execution times.

These preliminary results surprisingly showed that Parabrick implementation of STAR took a bit more time to align the same FASTQ read file than the CPU counterpart.

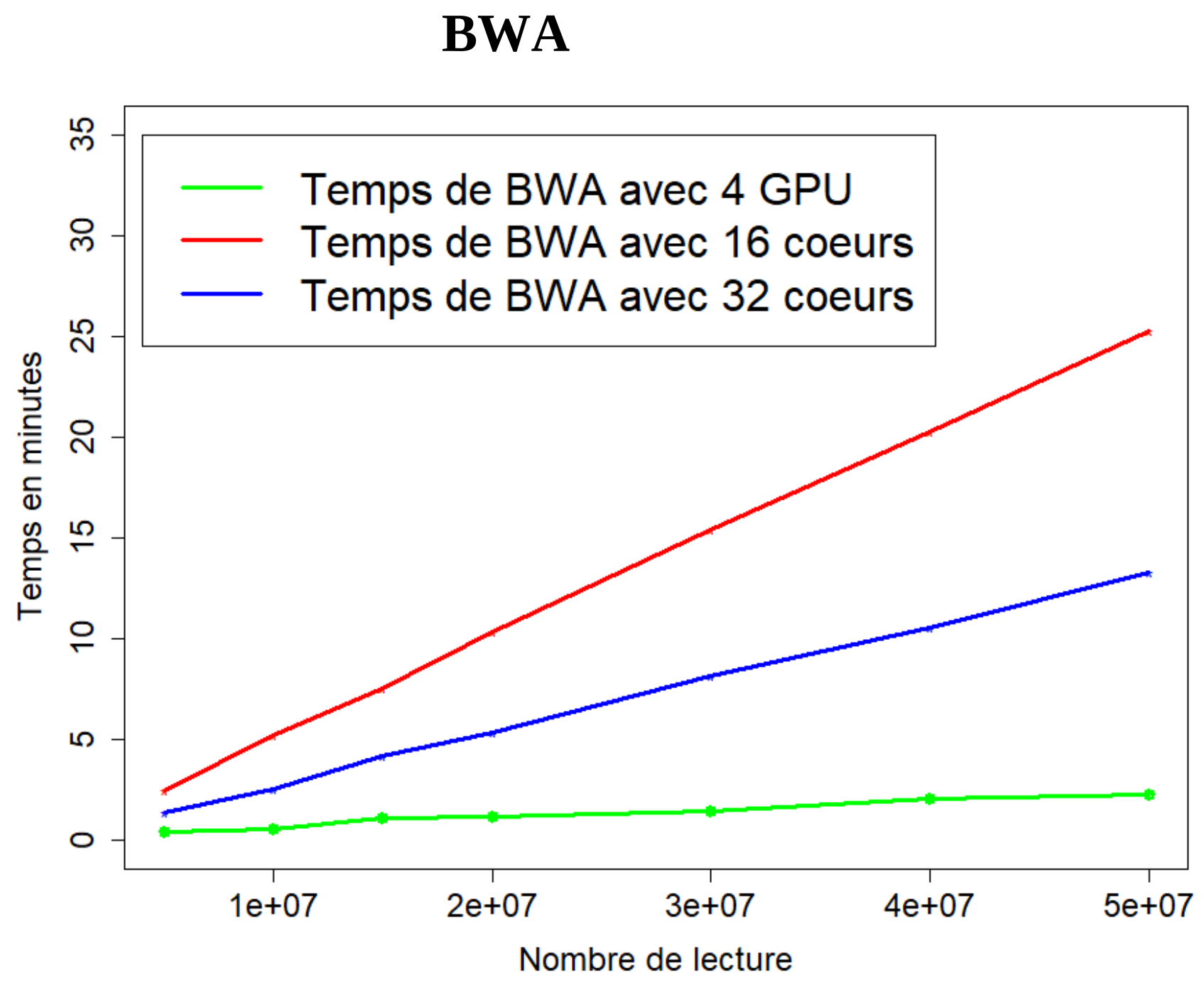


Figure 2: Comparison of BWA's execution times.

We found big improvement for BWA's runtime; alignments never takes more than 3 minutes.

Results: Variant callers

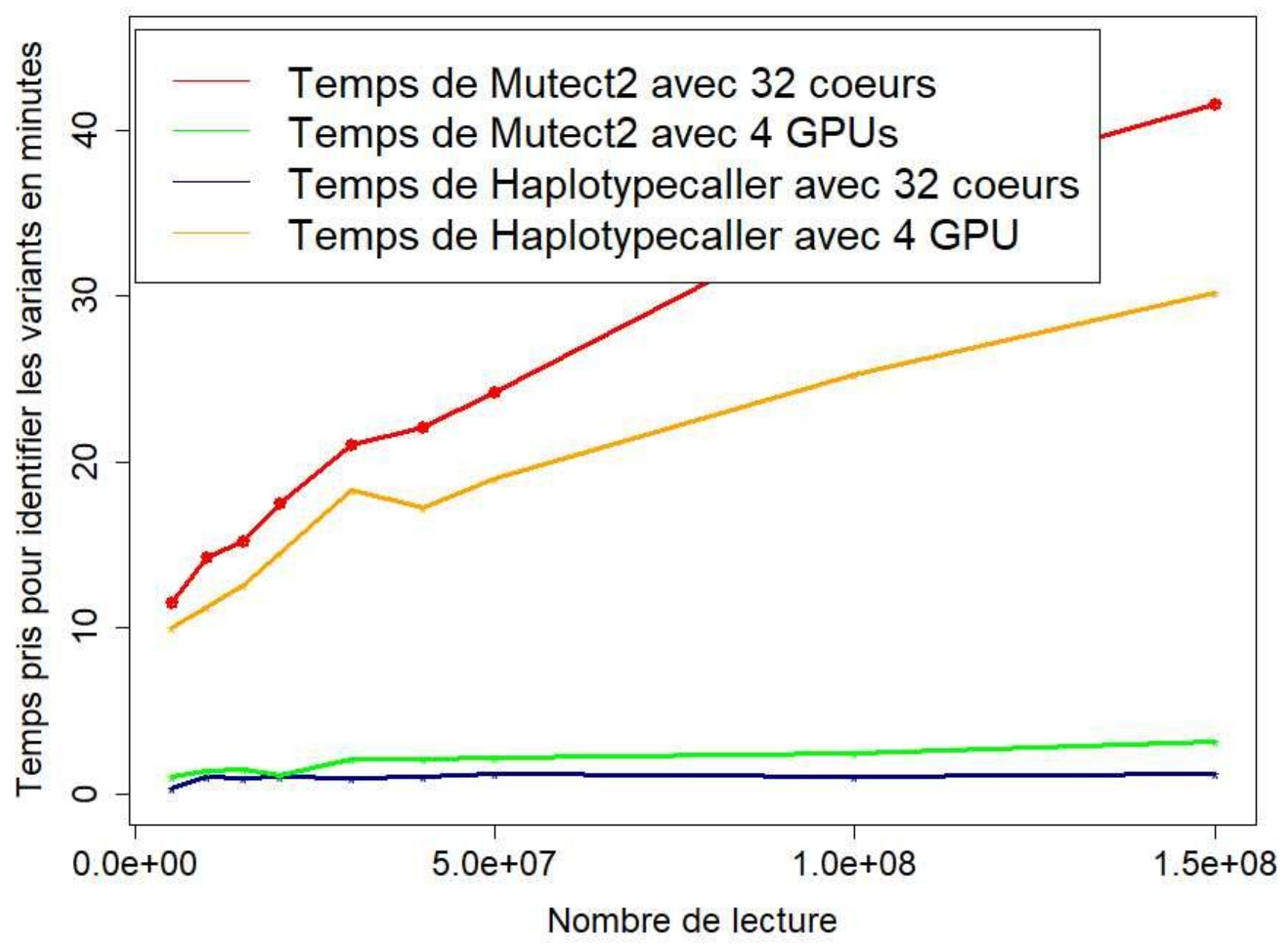


Figure 3: Comparison of Mutect2 and Haplotypecaller's execution times.

We observed, at minimum, a 15 fold reduction in execution time.

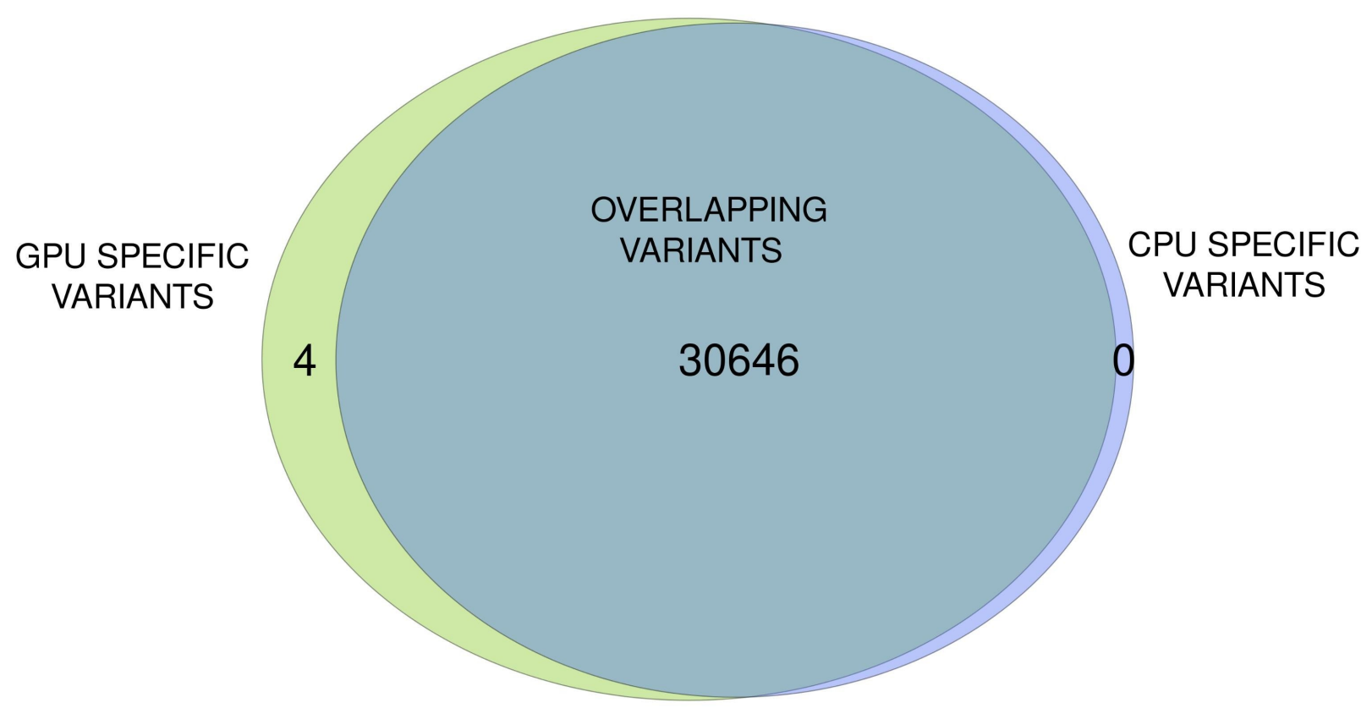


Figure 4: Comparison of the variants called by Haplotypecaller.

Almost no differences were found between variants called by CPU and GPU versions of Haplotypecaller.

Results: RNA Pipeline

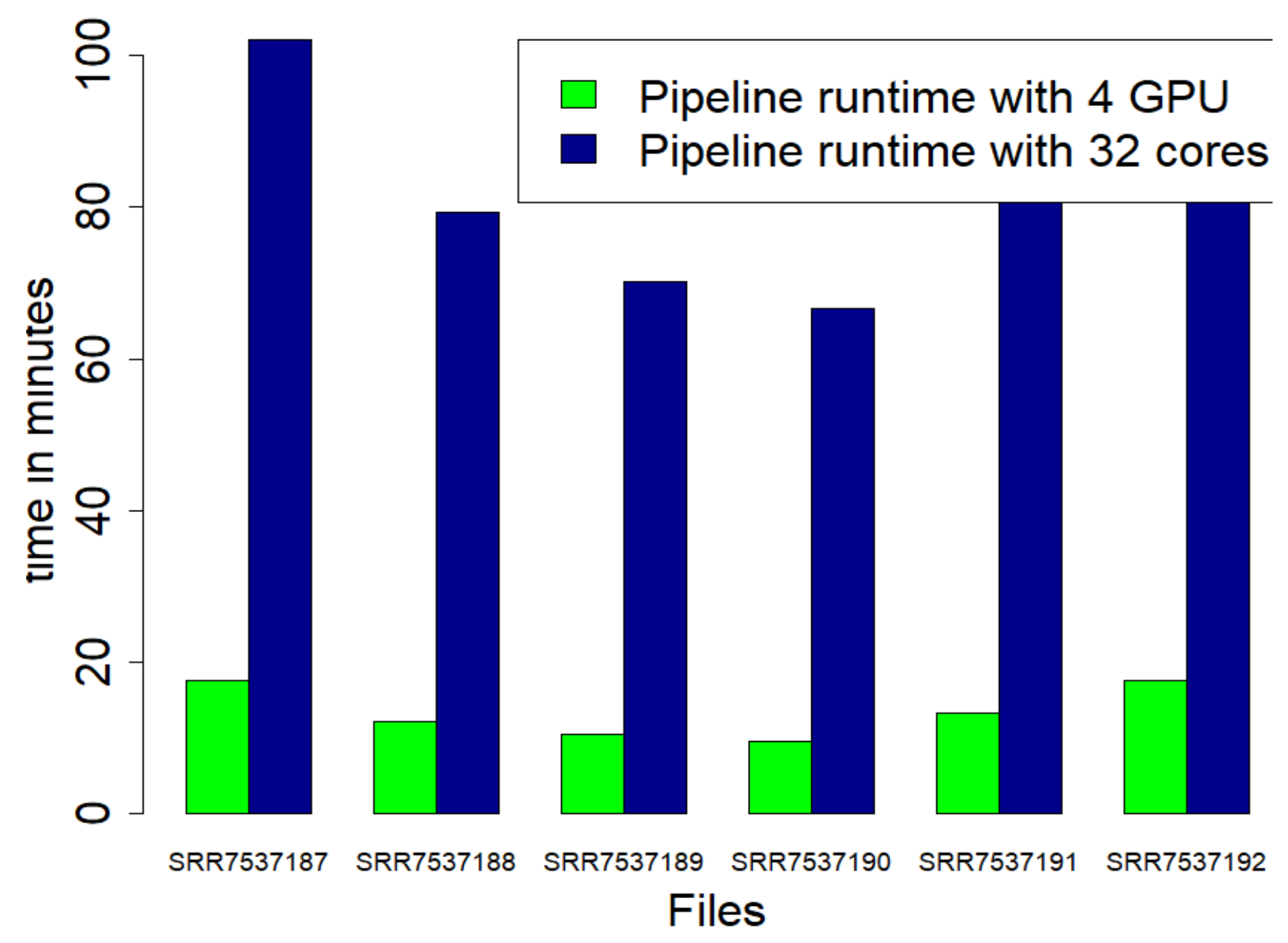


Figure 5: Execution times of the CPU and GPU versions of the RNA pipeline.

The RNA pipeline implemented by Clara Parabricks is more than 5 times faster than its CPU equivalent.

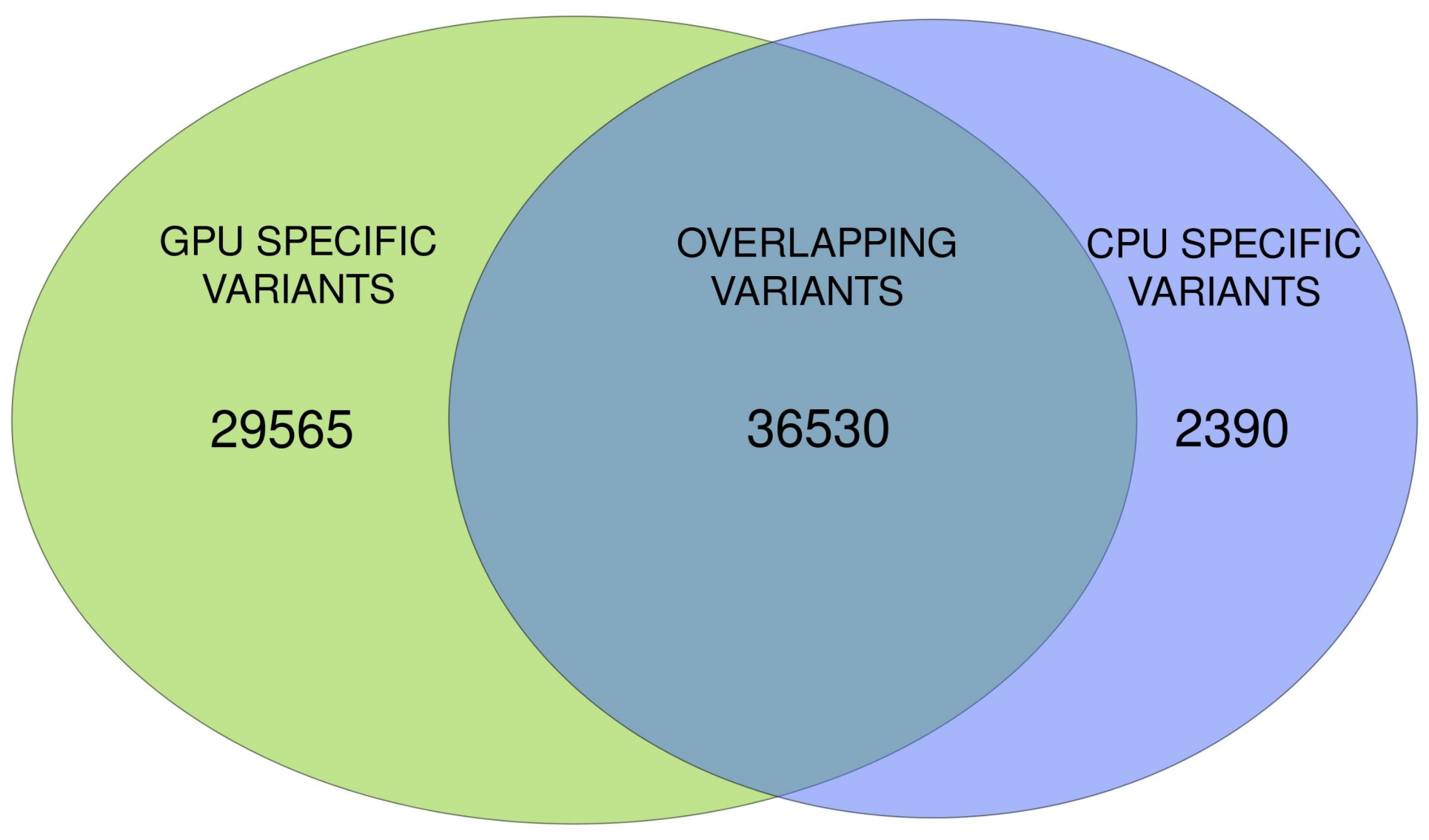


Figure 6: Comparison of the variants called by the RNA pipeline.

Surprisingly, the **GPU Pipeline** had about **60%** of it's **variant in common** with the CPU pipeline, but it **found 40% more variants**.

These results are currently investigated, and could be caused by:

- differences in **parameters** between the two versions of our RNA pîpeline
- differences between **alignments** produced by GPU and CPU versions of the STAR aligner.

Conclusion

We found that the **individual tools** implemented by NVIDIA Clara Parabricks were, for the most part, **most faster** than the original CPU implementation. The GPU implementation of the **variant callers** Mutect2 and Haplotypecaller produces almost the same results as the original CPU versions, while allowing much faster analysis.

Regarding **read aligners,** our tests seem to show that only **BWA** takes advantage from the GPU usage, while Parabricks's STAR was not faster than it's original CPU version. However, **Parabricks's STAR** is **not callable independently** and will always execute a **sorting** and a **markduplicates** steps. If we take this into account, then **Parabrick's STAR is actually faster than the CPU counterpart**.

Finally, we need to understand the differences in variant calls obtained by the GPU and CPU versions of the **full RNA pipeline**.

References

[1] Karl R. Franke and Erin L. Crowgey. Accelerating next generation sequencing data analysis: an evaluation of optimized best practices for Genome Analysis Toolkit algorithms. Genomics & Informatics, 18(1): e10, 2020

[2] Parabricks official documentation: https://docs.nvidia.com/clara/parabricks/3.8.0/index.html

[3] Kyle T.Siebenthall and Chris P,Miller. Integrated epigenomic profiling reveals endogenous retrovirus reactivation in renal cell carcinoma. EbioMedecine, 2019 Mar; 41: 427–442

The KATY project has received funding from the European Union's Horizon 2020 innovation program under grant agreement No 101017453.