

CNCA Centro Nacional
de Computação
Avançada

DEUCALION

Novidades no Deucalion

B. Malaca (CNCA)
Outubro 2025



Fundação
para a Ciência
e a Tecnologia



Universidade do Minho



O que mudou?

Otimização de software

Simulador quântico

Melhorias infraestrutura

E mais!



O Deucalion em números

Partição ARM

1632 nós

Processador Fujitsu A64FX
(48 cores)

32 GB HBM2 RAM (50%
mais rápida)

Ainda com disponibilidade



Partições x86

500 nodes + 33 GPU nodes

2 x AMD EPYC 7742 por nó
(128 cores)

256 GB RAM

**Com taxas de utilização
mais altas**

Partição ARM está no top 300 dos supercomputadores do mundo e recentemente passou a barreira dos 4 PFlops*

*Trabalho de Miguel Peixoto, Bernardo Malaca e Fujitsu

O que mudou?

Otimização de software

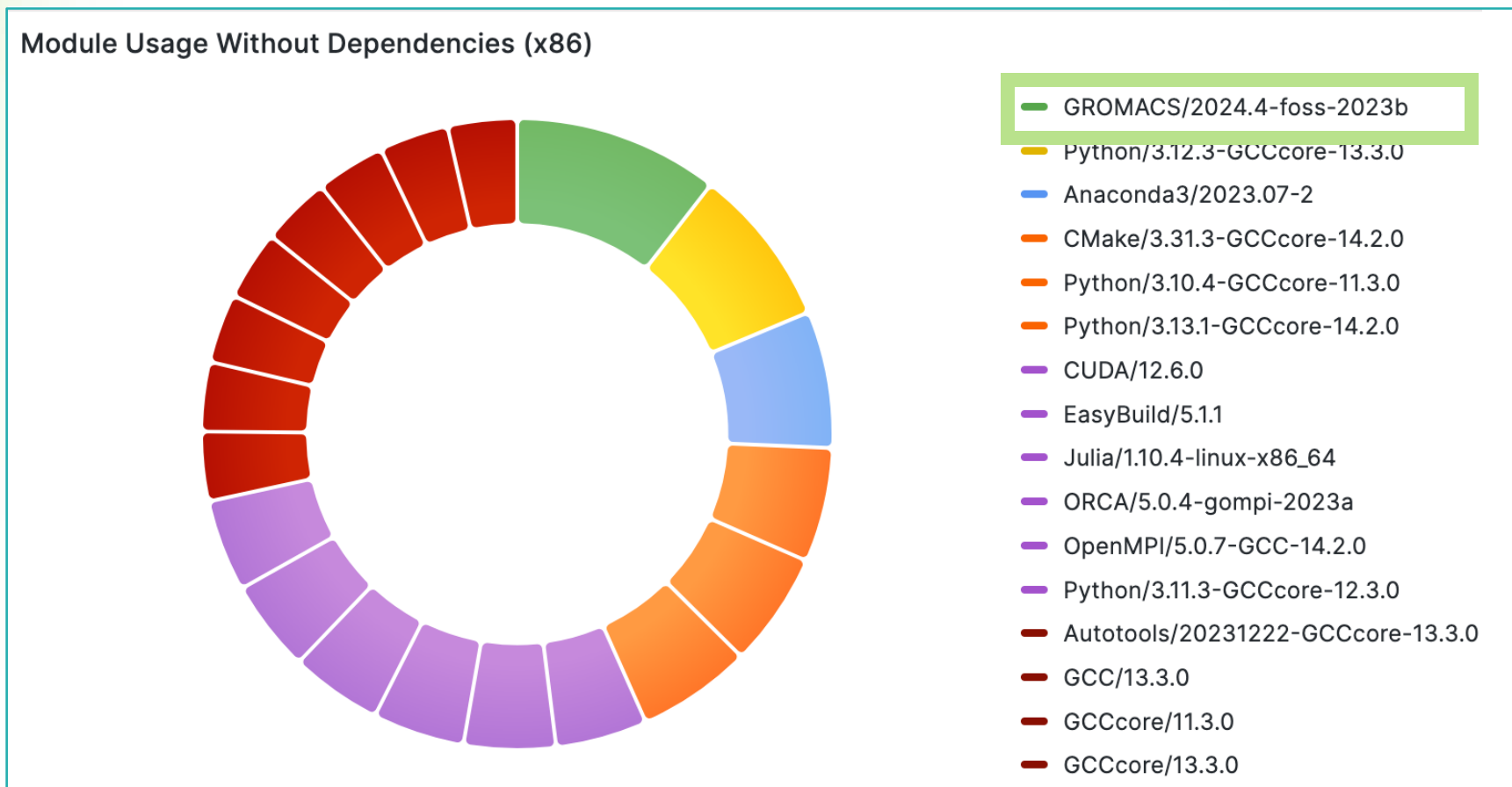
Simulador quântico

Melhorias infraestrutura

E mais!



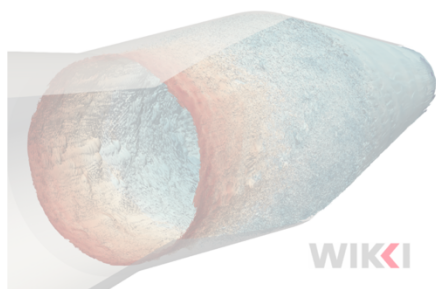
Aplicações mais utilizadas (x86)



Comparison with x86

All comparisons are made with the same number of cores. Even though memory access is faster with the A64FX, **the clock speed of the x86 is 70% faster.**

OpenFOAM (CFD)

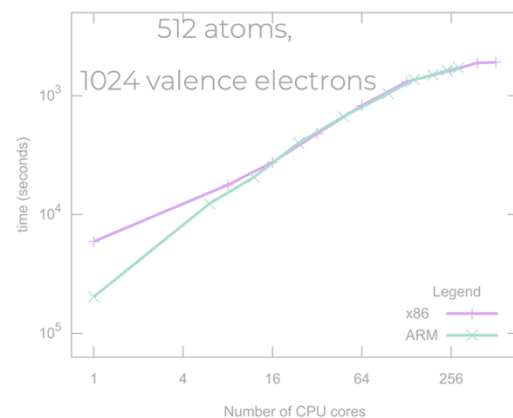


WIKI

X86 is 1.4x faster than ARM (both are not vectorized)

Gabriel Marcos Magalhães, UMinho

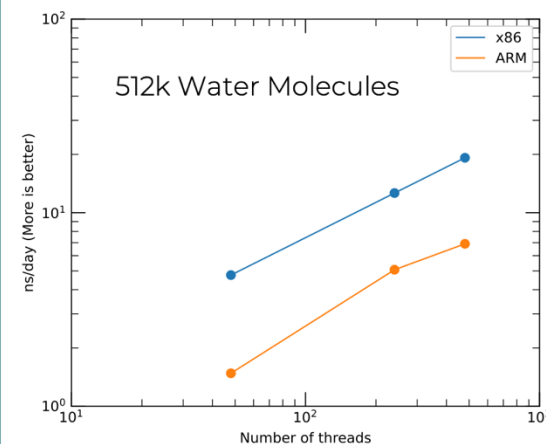
VASP (DFT)



ARM and x86 have the same performance

José Coutinho, Universidade Aveiro

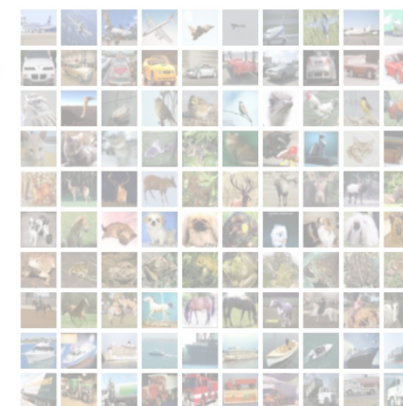
GROMACS (MD)



X86 is 2.4x faster than ARM

Pytorch (AI)

airplane
automobile
bird
cat
deer
dog
frog
horse
ship
truck

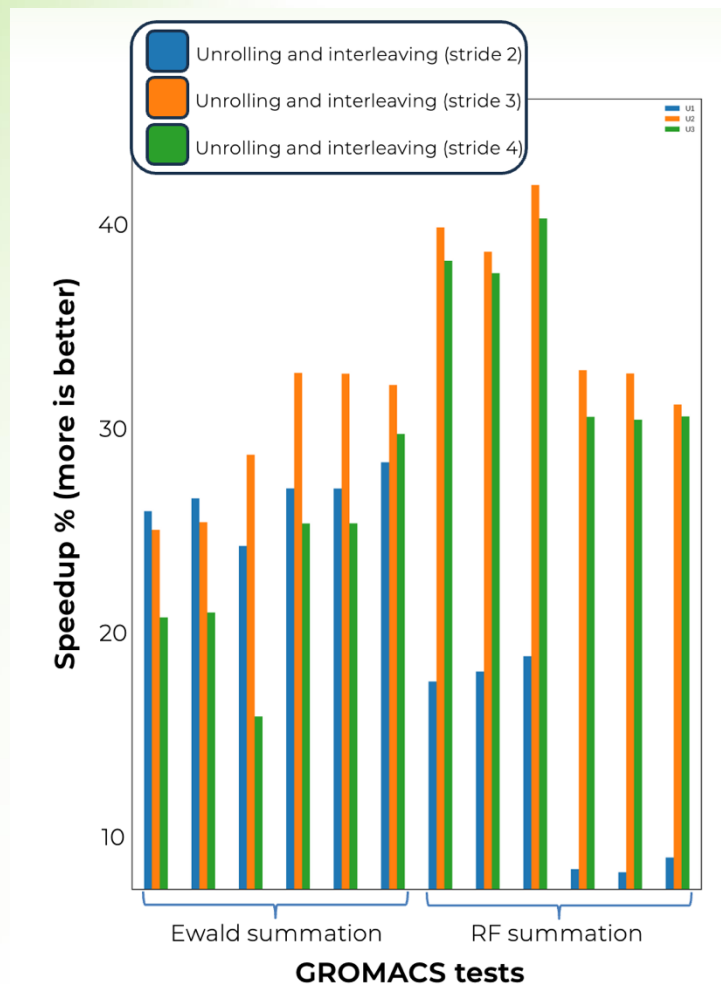


ARM 2-10x faster than x86. 1 GPU is equivalent to 15-20 ARM nodes



We routinely see **2-3x slower per-core performance on the ARM nodes**, with better performances in memory-bound, optimized codes (particularly in AI)

Otimização do GROMACS



Passos

Contacto com o investigador

Pacote de software estava preparado para um distribuidor diferente.

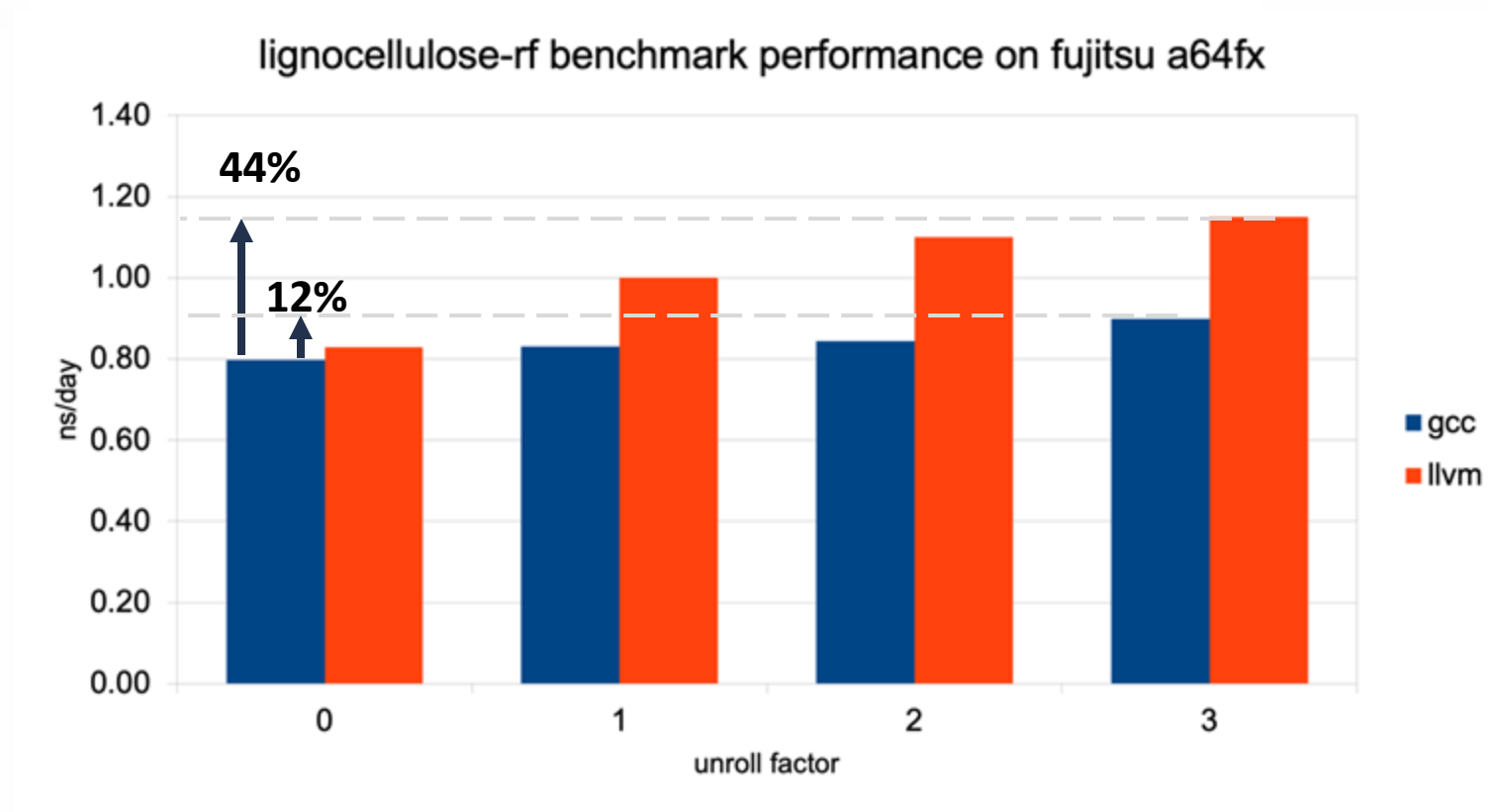
Criação de configuração adequada para o sistema de módulos do Deucalion

Teste em GCC+LLVM

Reprodução em máquinas Nvidia grace de nova geração

*Gilles Gouaillardet, https://www.hpci-office.jp/documents/meeting_A64FX/220727/GROMACS_A64fx.pdf

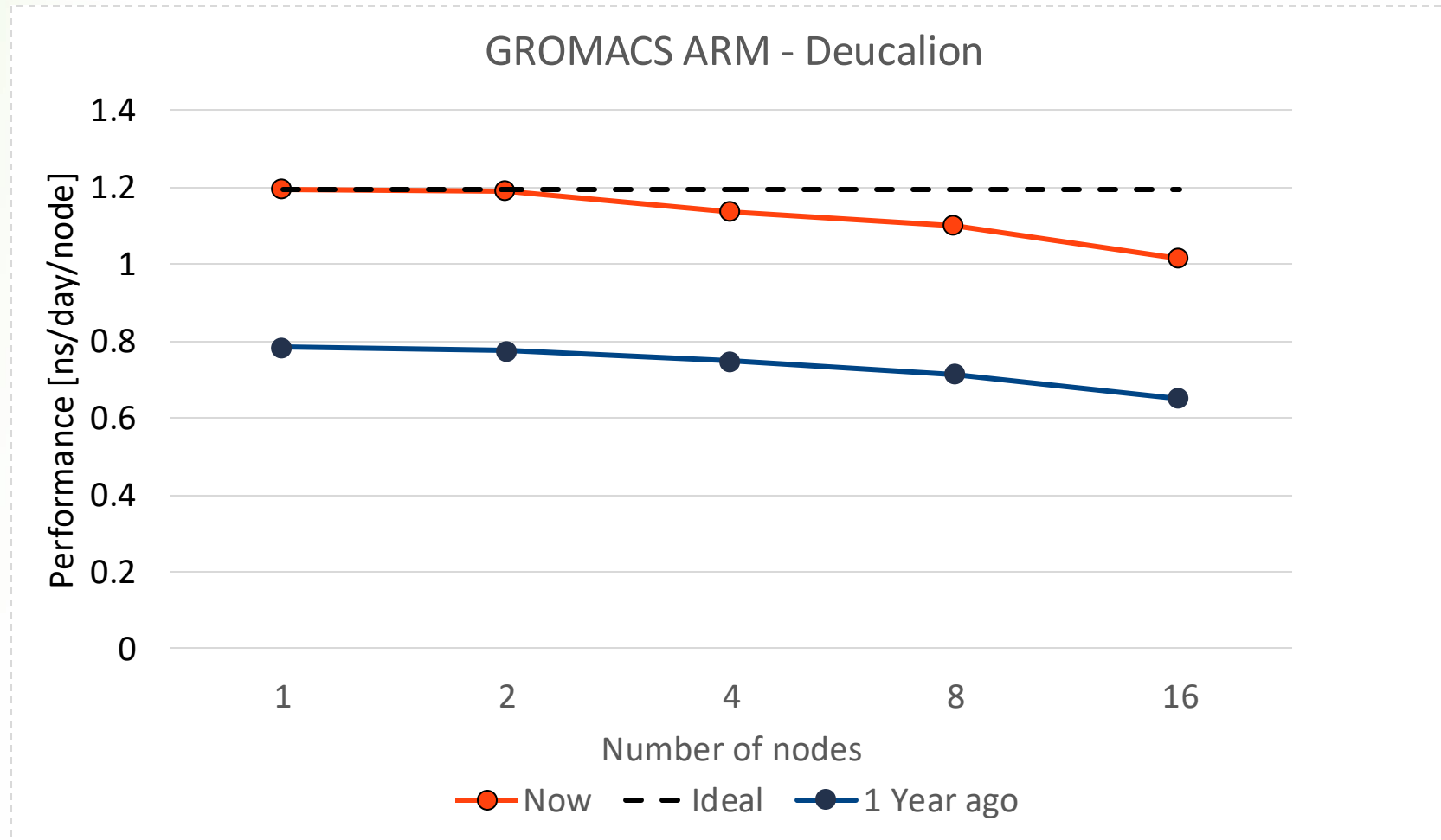
Resultados A64FX (48 cores)



Speedup 50% face à versão não otimizada (apenas com LLVM!)

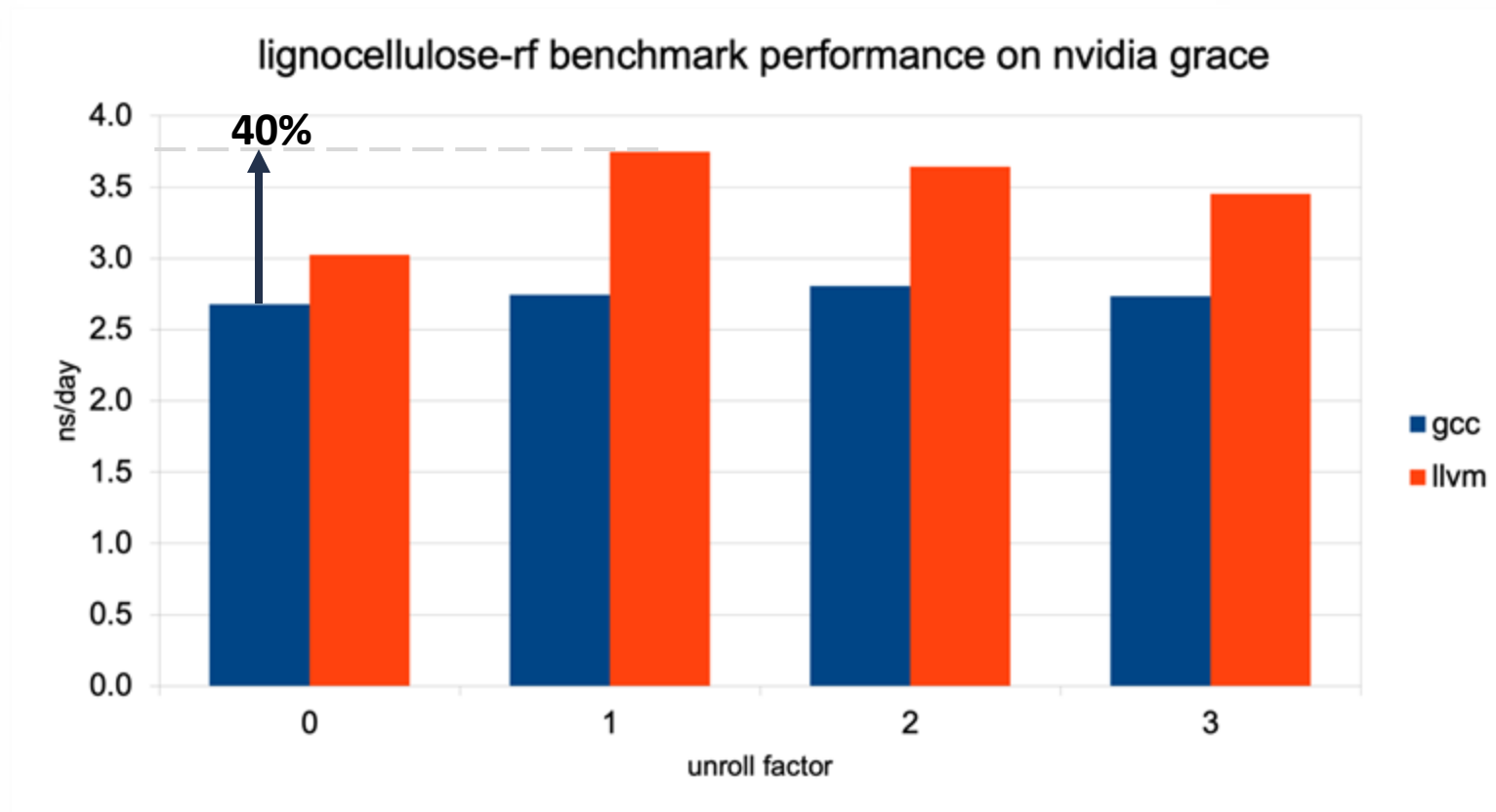
*Obtidos com Miguel Dias Costa

Testes de escalabilidade



Diferença de performance mantém-se em multi-nó

Resultados Nvidia Grace (72 cores)



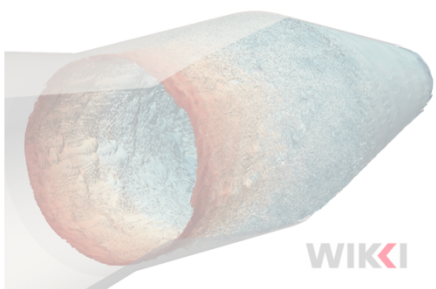
Volta-se a observar um fenómeno muito semelhante mesmo num chip completamente diferente

*Obtidos com Miguel Dias Costa

Comparison with x86

All comparisons are made with the same number of cores. Even though memory access is faster with the A64FX, **the clock speed of the x86 is 70% faster.**

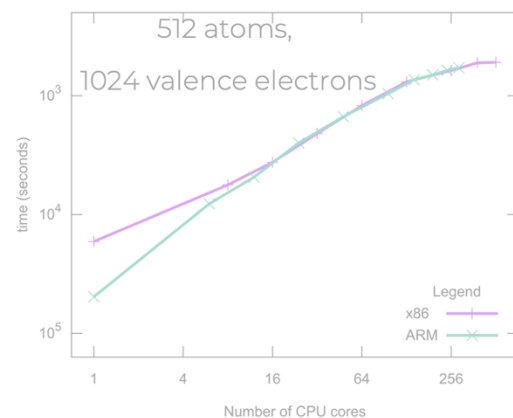
OpenFOAM (CFD)



X86 is 1.4x faster than ARM (both are not vectorized)

Gabriel Marcos Magalhães, UMinho

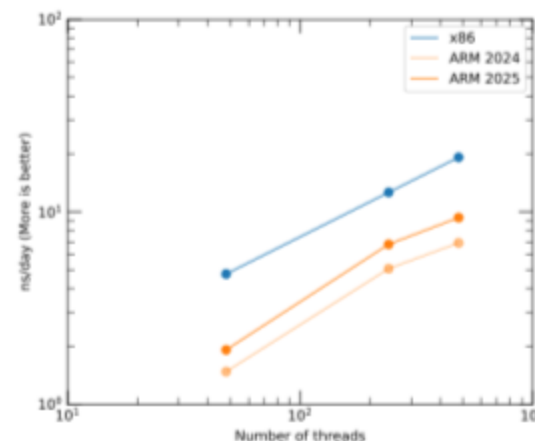
VASP (DFT)



ARM and x86 have the same performance

José Coutinho, Universidade Aveiro

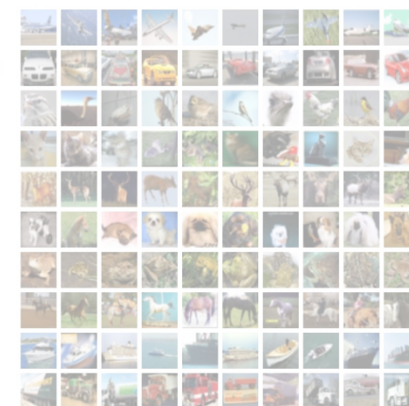
GROMACS (MD)



X86 é 1.6x – 2.1x mais rápido

Pytorch (AI)

airplane
automobile
bird
cat
deer
dog
frog
horse
ship
truck

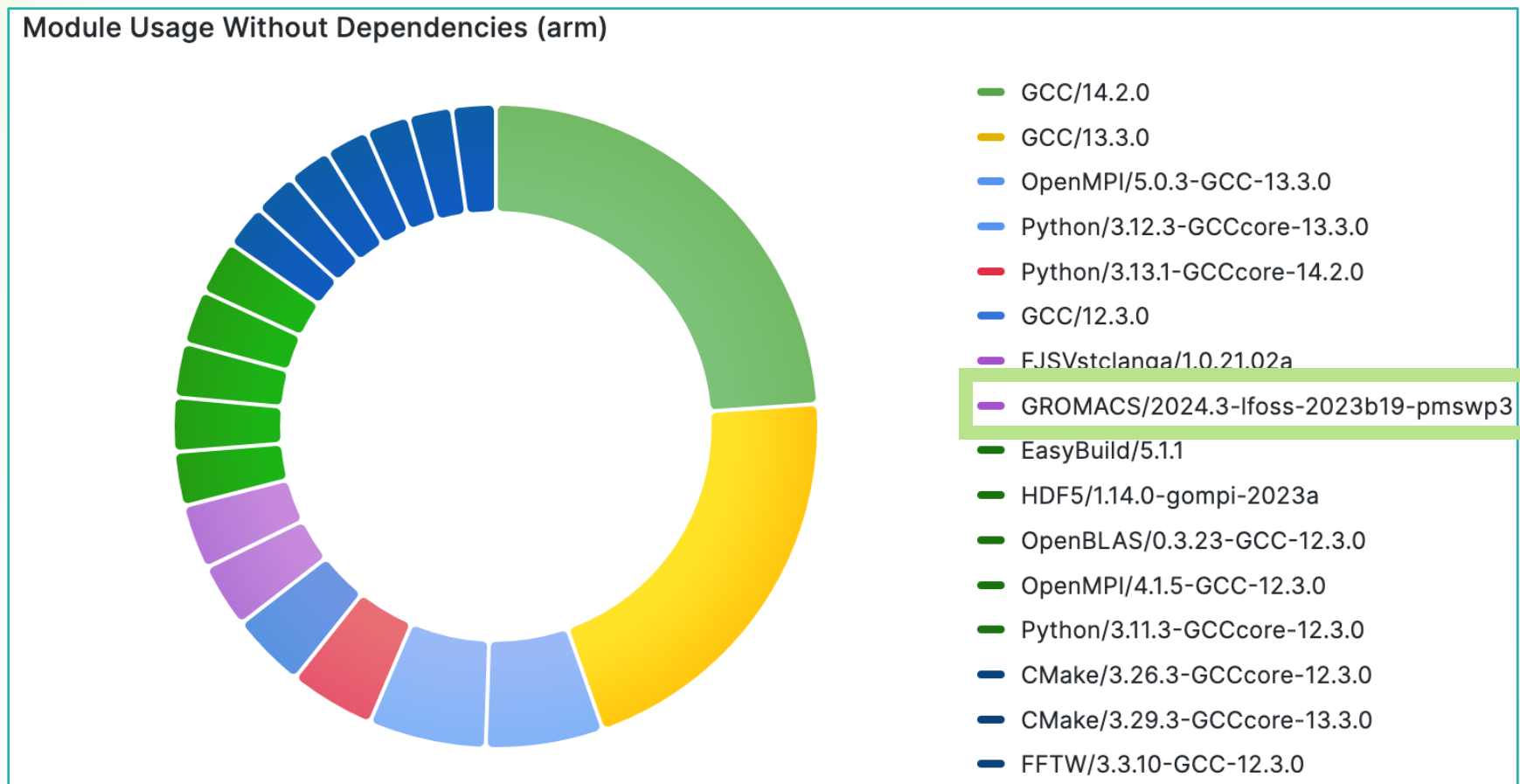


ARM 2-10x faster than x86. 1 GPU is equivalent to 15-20 ARM nodes



We routinely see **2-3x slower per-core performance on the ARM nodes**, with better performances in memory-bound, optimized codes (particularly in AI)

Novo modulo já disseminado



Versão disponibilizada para todos os utilizadores (com bom feedback)

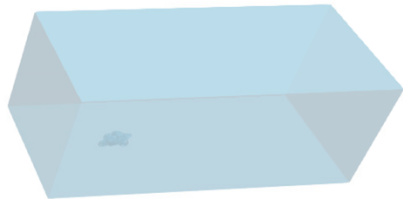
Também há outras maneiras de melhorar performance

GCC or Fujitsu? It depends

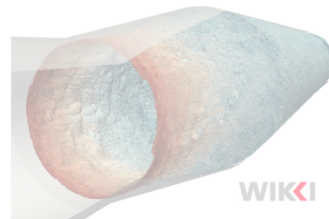
A lot of effort has been put into open-source compilers (e.g., GNU's gcc) so they can match well with proprietary compilers (Fujitsu's fcc)

OpenFOAM

Small motorbike



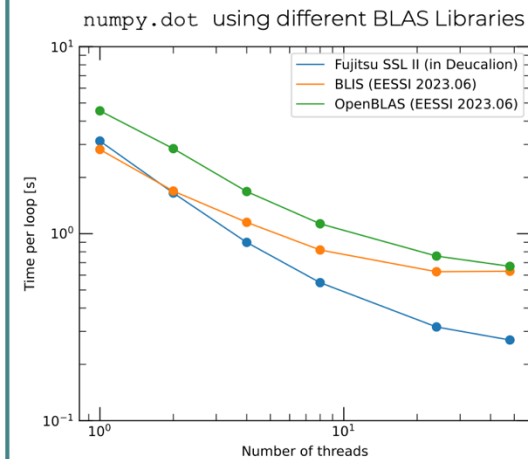
Conical Diffuser



GCC grasps **97%** of the performance of Fujitsu's FCC

Gabriel Marcos Magalhães, OpenFOAM Iberia (2024)

BLAS Libraries



Fujitsu Optimized BLAS still outperforms **(150% speedup)** other libraries

On a64fx, single-threaded BLIS outperforms OpenBLAS and is even faster than Fujitsu's!

Miguel Dias Costa (2024)

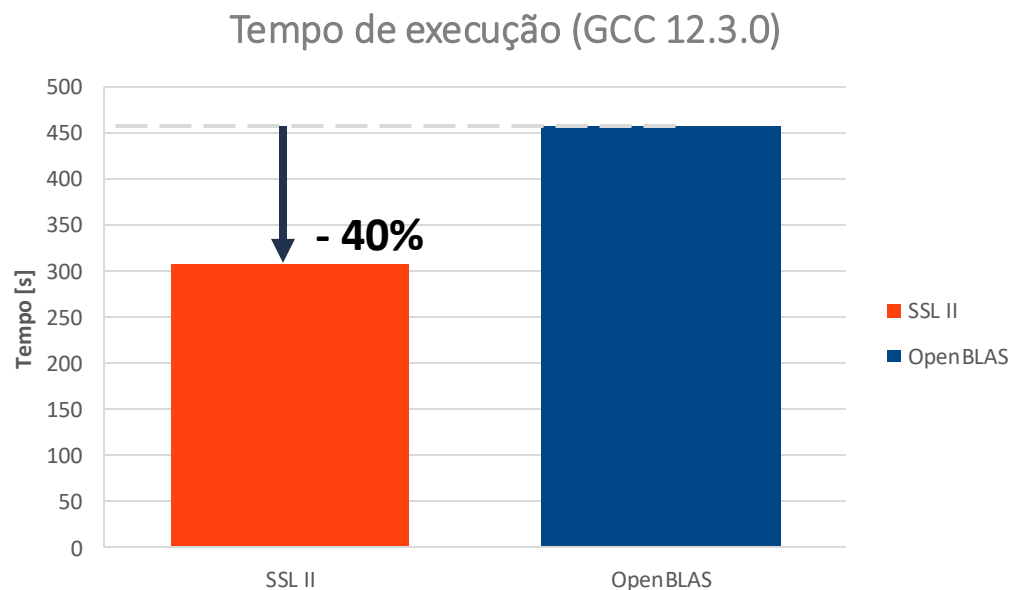
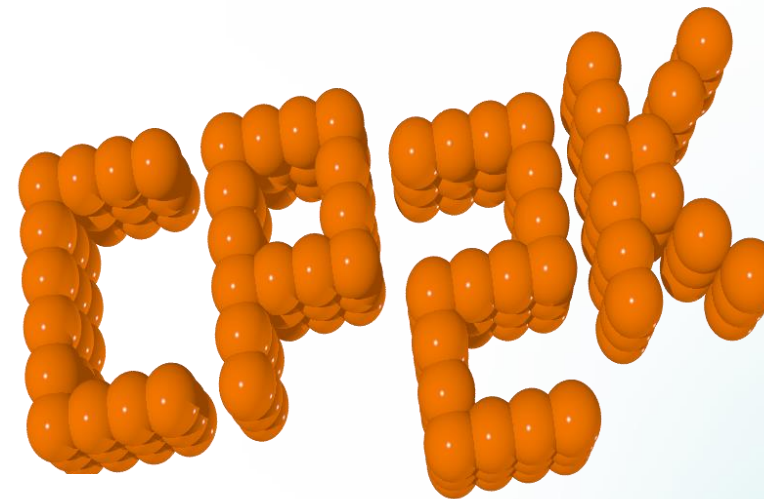
If modules, there is not a significant performance increase. We maintain and support software stacks that benefit from using Fujitsu's toolchain



Existe diferença em aplicações científicas

CP2K é um Código para química quântica e física do estado sólido

Benchmark utilizado era essencialmente composto por operações matriciais



Comparações com outros centros

CPU – Energy usage in [kJ]

GROMACS

	LUMI	MeluXina	Vega	Karolina	MN5	Leonardo	Deucalion	Discoverer
1	360,16	394,38	766,96	334,05	398,12	339,05	357,69	371,30
2	372,47	372,89	758,62	333,90	437,86	395,10	368,07	372,64
4	391,61	326,97	657,33	352,31	487,24	447,42	401,19	383,56
8	414,44	387,20	657,51	379,58	570,45	563,39	456,98	440,65
16	491,29	463,77	403,18	433,31	863,00	932,68	568,48	544,05

CP2K

	LUMI	MeluXina	Vega	Karolina	MN5	Leonardo	Deucalion	Discoverer
1	294,92	249,08	329,14	216,57	265,07	321,14	213,75	291,05
2	330,93	294,27	367,40	250,65	304,10	416,23	232,90	359,03
4	287,19	324,76	447,95	269,45	338,92	572,71	265,42	421,65
8	473,51	408,21	559,47	488,66	330,81	740,46	317,17	571,28
16	716,28	653,16	770,77	588,12	388,89	1492,97	493,50	794,34

Identificar pontos fracos e melhorias para o futuro

Outras lições

Nem todos os softwares têm melhoria imediata na passagem de GCC para LLVM, mas vale a pena experimentar (cuidado com a versão 20 do LLVM!)

No que toca a operações matriciais, a biblioteca SSL II da Fujitsu ainda é a melhor e nota-se mesmo em aplicações científicas

Aquilo que se verifica no A64FX pode ser benéfico noutros chips ARM mesmo que o fabricante seja diferente

É possível trazer os utilizadores para a partição ARM oferecendo propostas diferenciadoras...

O que mudou?

Otimização de software

Simulador quântico

Melhorias infraestrutura

E mais!



Simulador quântico

Quantum Com

How to simulate quantum computers

States (or bits) can be represented as turned on (1) or turned off (0)

Classical N-bit systems can represent one out of 2^N possible states

Quantum systems with N qubits can represent these states as a superposition

To which state we can associate a complex number that determines the probability of measuring that state

<https://rnca.fccn.pt/acesso/laboratorios/quantumvlab/>

B. Malaca, QuantumvLab Webinar, 18 June 2025



EDUCALION

RAM

2 bytes

MB

GB

EDUCALION

Bernardo Malaca

<https://github.com/andre-sequeira10/HPC-Quantum/>

Ecossistema de simulação

Qulacs



Otimizado pela Fujitsu para correr nos A64FX da partição ARM

API para Python e C++ (embora a investigação em computação quântica seja feita a 99% Python)

Qiskit



O gold standard da simulação (utilizado por mais grupos)

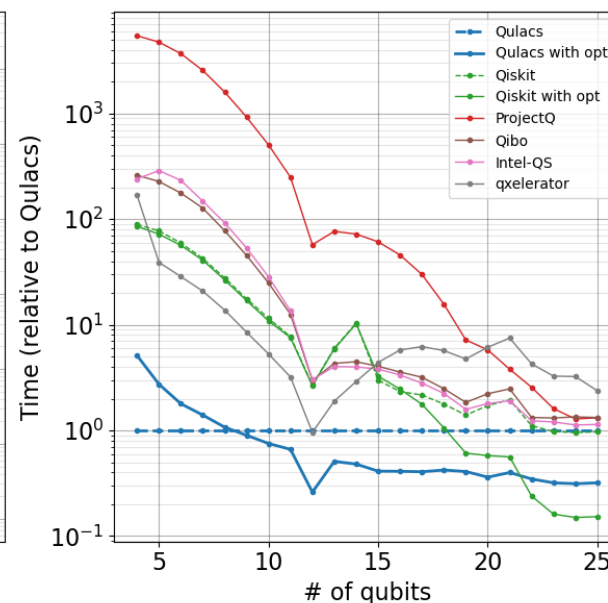
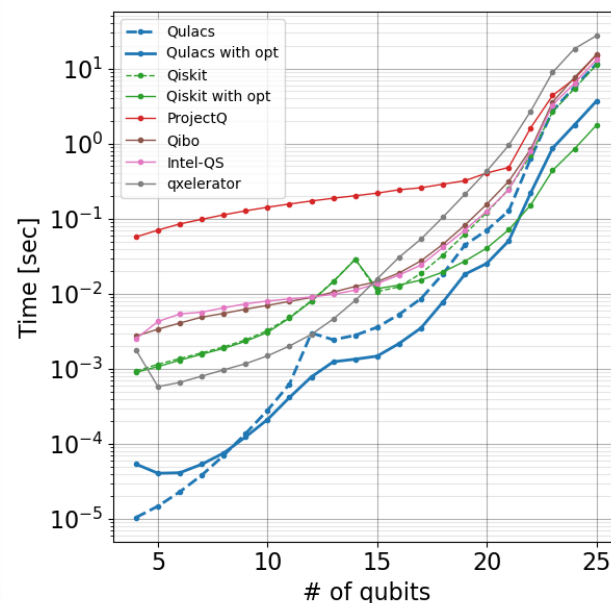
Performance semelhante ao qulacs em x86, inferior em ARM.

PennyLane



Capacidade de utilizar backend próprio, mas também qulacs ou qiskit.

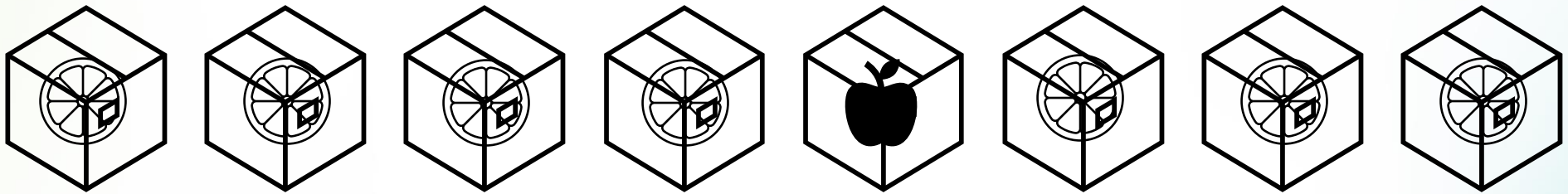
Integração com outros pacotes Python para aprendizagem automática (PyTorch, JAX, etc.)



Qulacs ARM: 40 qubits
com 10 gates de 20 qubits
cada:

7 minutos! (sampling leva
1 segundo)

Exemplo: algoritmo de Grover



ARM apresenta melhor performance

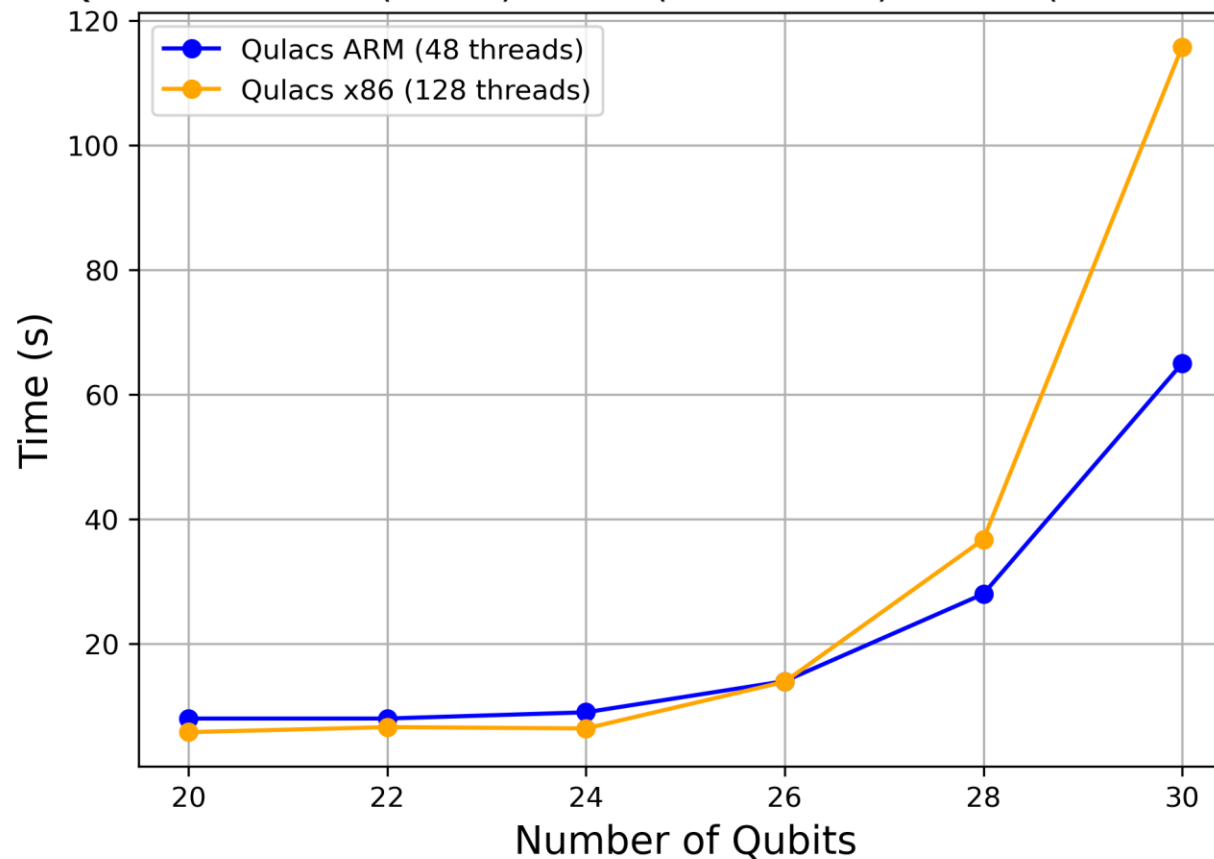
**Módulo de qulacs disponível para todos os
utilizadores:**

```
module load qulacs
```

**Limites são dependentes da memória de
cada máquina**

**Todos os módulos disponibilizados
permitem a utilização de várias máquinas
em simultâneo**

Qulacs - Grover (10 its) - ARM (48 threads) vs x86 (128 threads)



*Resultados obtidos por André Sequeira

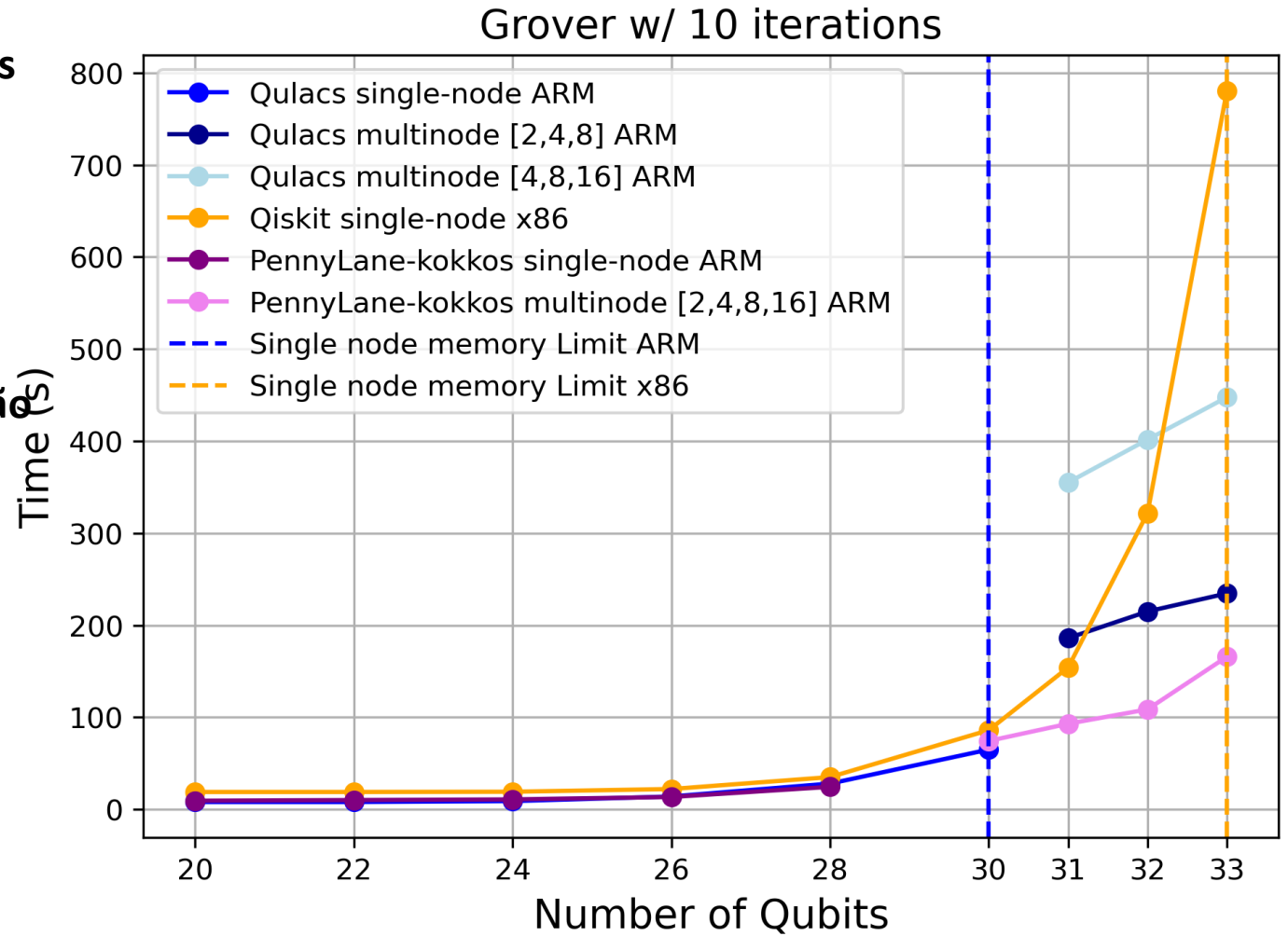
Vários códigos disponíveis no Deucalion

Módulo de qiskit disponível para todos os utilizadores:

```
module load Qiskit
```

Módulo de pennylane ainda em preparação

Qiskit aparenta ter pior performance sempre que se utiliza mais do que uma máquina



*Resultados obtidos por André Sequeira

O que mudou?

Otimização de software

Simulador quântico

Melhorias infraestrutura

E mais!



Problemas no arrefecimento

Problemas

Alterações repentinas na rede elétrica levavam os chillers a desligar-se

Esforço de toda a equipa*, fora do horário de trabalho

Falta de estabilidade

Soluções

Máxima prioridade: alocar membro da equipa (Rui Silva)

Reconfiguração por parte do fabricante

*Agradecimento especial ao Francisco Ferraz e Miguel Peixoto



O que mudou?

Otimização de software

Simulador quântico

Melhorias infraestrutura

E mais!



Dar mais controlo aos responsáveis

Como fazer para integrar um membro de um projecto que nunca utilizou uma máquina destas?

Tenho de verificar o billing todos os dias para garantir que há um certo equilíbrio entre todos os membros?

PROJECT INFO		PROJECT MEMBERS		PROJECT ALLOCATIONS		PROJECT USAGE	
Project Usage		Search				EXPORT AS	
ARM		X86		GPU			
☐	Username	Used (% of allocation)	Limit (%)	Used (% of allocation)	Limit (%)	Used (% of allocation)	Limit (%)
☐	alsousa	0	1	0	1	0	1
☐	ampereira	0	1	0	1	0	1

*Trabalho de Guilherme Fernandes e Diogo Pires

Muita

```
[testmalaca3@ln01 ~]$ billing
```

Account	Used (h)	Limit (h)	Used (%)
f202500010hpcvlabuminhoa	508	1000000	0.05
f202500010hpcvlabuminhog	0	1000	0.00

```
[testmalaca3@ln01 ~]$ billing -a f202500010hpcvlabuminhoa
```

Account	Used (h)	Limit (h)	Used (%)
f202500010hpcvlabuminhoa	508	1000000	0.05

```
[testmalaca3@ln01 ~]$ billing -u
```

User	Used (h)	Limit (h)	Used (%)
------	----------	-----------	----------

*Trabalho de Guilherme

Muitas pequenas coisas

```
[malaca@ln04 ~]$ get_energy -a f202500001hpcvlabepicurea | jq
{
  "type": "JOB",
  "data": {
    "#Jobs": 4476,
    "submit_time_first": 1746534217,
    "submit_time_last": 1760980650,
    "budgeting_levels": {
      "CPU[J]": 3316836603.575545,
      "CPU[kWh]": 921.3435009932115,
      "GPU[J]": 0,
      "GPU[kWh]": 0,
      "Node[J]": 3641647336.611277,
      "Node[kWh]": 1011.5687046142455,
      "Overall[J]": 5460536560.330105,
      "Overall[kWh]": 1516.8157112028052,
      "CO2 emissions[g]": 0
    },
  },
  (...)
}
```

*Trabalho de IT4I, Fujitsu, Miguel Peixoto e Francisco Ferraz

Muitas pequenas coisas

Billing

Comandos para aceder facilmente a informações relevantes (quantas horas tenho para gastar, quais são as minhas accounts disponíveis)

```
billing #num nó login
```

Medição energia

Comandos para aceder à energia de cada job ou da energia total de um projecto:

```
get_energy -j <jobid>  
get_energy -a <account>
```

Open OnDemand

 Jupyter Notebook Pytorch

 Jupyter Notebook Pytorch (GPU)

 mlflow

 RStudio Server

 TensorBoard

 VS Code

Permitir alocar apenas uma GPU
Containers mais rápidos

*Trabalho de Rui Silva e Bernardo Malaca

O Deucalion é dos utilizadores

Quase todas as novas ferramentas decorreram de conversas com utilizadores

Não hesitem em falar connosco se tiverem alguma sugestão!

