Accelerating 3-way Epistasis Detection with CPU+GPU processing

Ricardo Nobre
ricardo.nobre@inesc-id.pt

Aleksandar Ilic
aleksandar.ilic@inesc-id.pt

Sergio Santander-Jiménez
sergio.jimenez@inesc-id.pt

Leonel Sousa
leonel.sousa@inesc-id.pt

inesc id lisboa 20 YEARS
DEFINING TECHNOLOGY

Impact of discovering genotype / phenotype associations

genotype
(collection of genes)

phenotype
(observable characteristics)

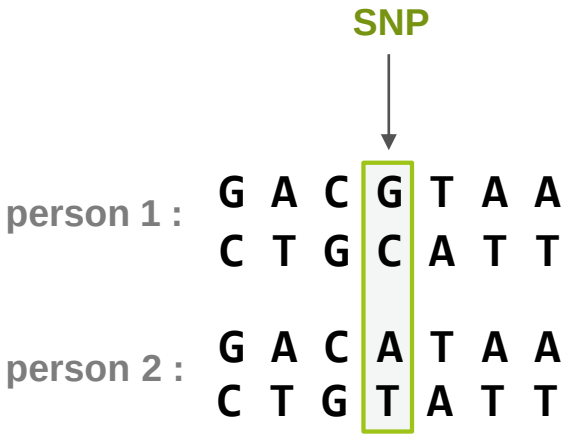
identify risk factors

predict
- **drug response**
- **viral infection response**

personalize treatment

drug development

Single Nucleotide Polymorphism (SNP)



at specific DNA position
(> 1% population)

major allele (*A*) — highest freq.

minor allele (*a*) — lowest freq.

AA → **homozygous major**

aa → **homozygous minor**

aA → **heterozygous**

Genome-Wide Association Study (GWAS)

data for large number of samples (for different SNPs)

cases

have trait

controls

do not have trait

SNPs most correlated w/ trait under study

test for difference in genotype frequency

(e.g., K2 bayesian score, mutual information)

Complex traits result from high-order interactions

epistasis

interactions between k SNPs

$$C(m,k) = \frac{m!}{k!\,(m-k)!}$$

10K SNPs

$\sim 50\text{M}\ (k=2)$

$\mathbf{\sim 167G\ (k=3)}$

more genotypes (3^k) → higher complexity

$9\ (k=2)$ $\qquad$ $\mathbf{27\ (k=3)}$

Problem is extremely data-parallel

For each set of SNPs

1. Count genotype frequencies
2. Score combinations of SNPs
3. Reduce scores / identify solution

multiple sets of SNPs to evaluate

(same calculations per set)

suits parallel architectures

Related Work

Approaches differ in relation to the targeted architectures and devices

multicore CPUs[1,2] **GPUs**[2,3,4,5] **FPGAs**[4] **other accelerators**[5] (e.g., Xeon Phi)

some target multiple classes of devices and/or clusters

high-order exhaustive searches are rarely tackled

[1] J. C. Kässens, UPC++ for bioinformatics: A case study using genome-wide association studies. CLUSTER, 2014

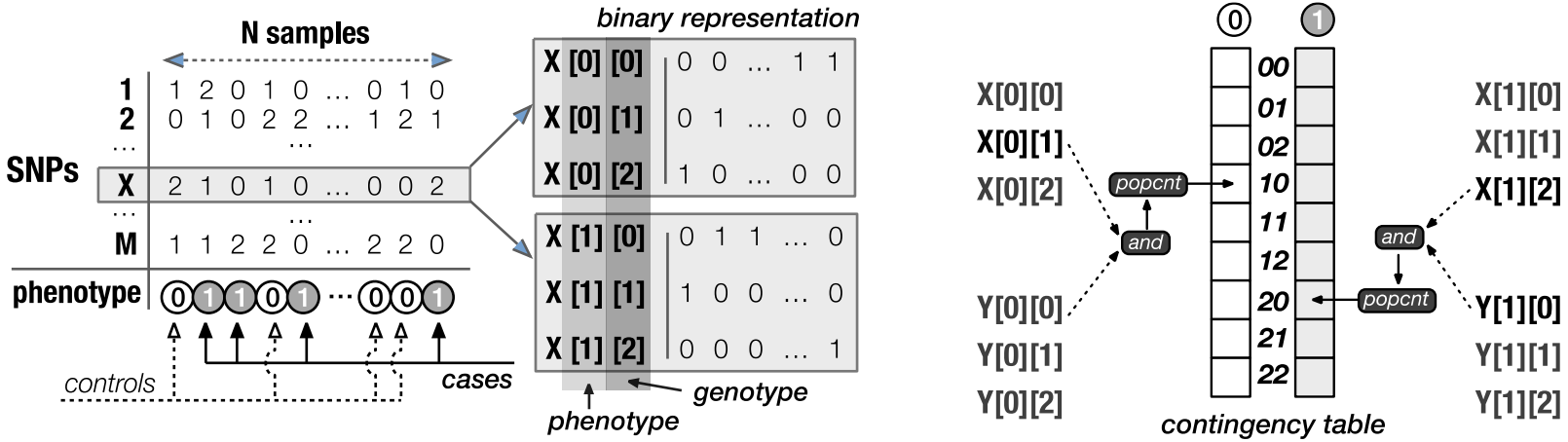
[2] C. Ponte-Fernández, Fast search of third-order epistatic interactions on CPU and GPU clusters. IJHPCA, 2020

[3] W. Joubert, Attacking the opioid epidemic: determining the epistatic and pleiotropic genetic architectures for chronic pain and opioid addiction. SC, 2018

[4] L. Wienbrandt, 1000x faster than PLINK: Combined FPGA and GPU accelerators for logistic regression-based detection of epistasis. J COMPUT SCI-NETH, 2019

[5] J. González-Domínguez, Parallel Pairwise Epistasis Detection on Heterogeneous Computing Architectures. TPDS, 2015

Contingency table w/ binary AND and POPC



more efficient than indexing calculation w/ original dataset

AND and POPC instructions process multiple samples per genotype (e.g., CPU[1], GPU[2])

[1] X. Wan, BOOST: A fast approach to detecting gene-gene interactions in genome-wide case-control studies. AJHG, 2010
[2] C. Ponte-Fernández, Fast search of third-order epistatic interactions on CPU and GPU clusters. IJHPCA, 2020

Overview of presented work

Novel proposal and evaluation

1. Efficient GPU+CPU exhaustive high-order epistasis detection
2. Analytical and experimental analysis on different systems
3. Comparison with recent related art targeting CUDA cores

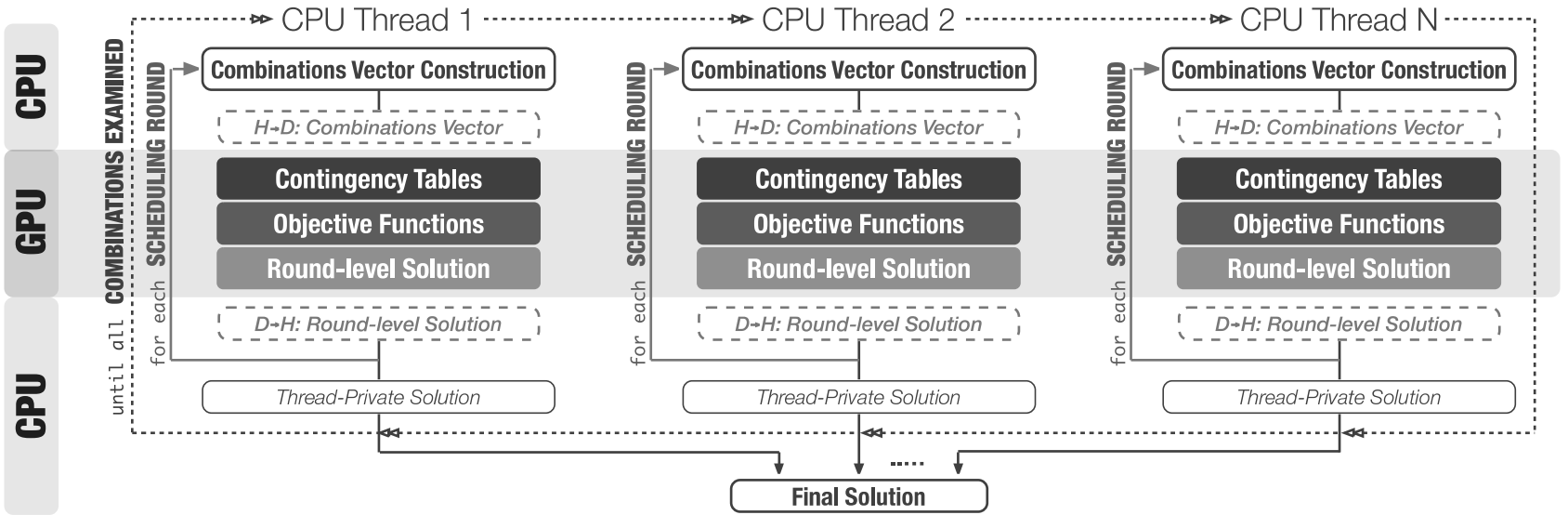
All-to-all combinations processed w/ joint collaborative action of CPU and GPU devices

- CPUs generate
- GPUs evaluate

High performance across multiple GPU architectures

- adaptive workload distribution
- multi-GPU support

GPU+CPU overview



$$\text{num. rounds} = \left\lceil \frac{\frac{m!}{3!(m-3)!}}{s} \right\rceil$$

$s \longrightarrow$ num. combinations per round

CPU as generator of combinations of SNPs

Exhaustive epistasis detection requires evaluating all-to-all combinations

(permutations ≠ combinations)

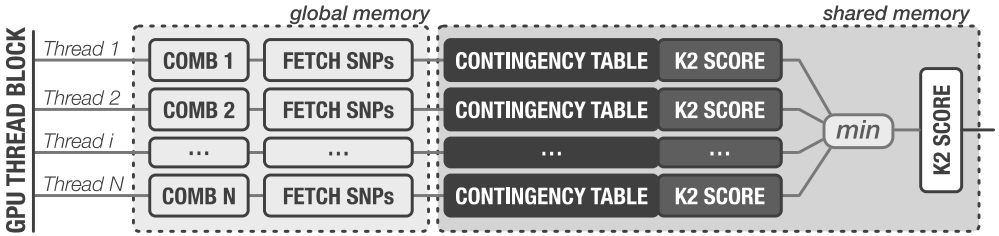
Each round (on a given CPU thread) starts with the generation of the first combination to be processed on that round

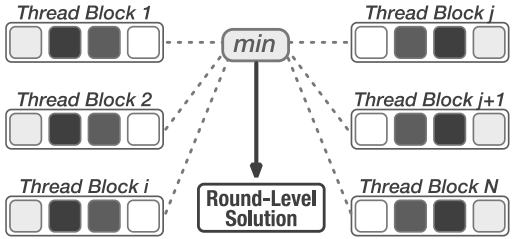
$comb()$ maps to lookup table

Sequentially generate next combinations untill s (chunk size)

Data: m, l
Result: c
$r = l;$
$c[0] = -1;$
for $i = 0;\ i < 2;\ i = i + 1$ **do**
  while $r > 0$ **do**
    $c[i] = c[i] + 1;$
    $d = comb(m - (c[i] + 1), 2 - i);$
    $r = r - d;$
  end
  $r = r + d;$
  $c[i + 1] = c[i];$
end
$c[2] = c[2] + r;$

GPU as evaluator of combinations of SNPs





read / write memory accesses in coalesced patterns

contiguous threads fetch contiguous SNP data
{32,41,1854} → {32,41,1855}

(global memory)

contingency table updates (filling) and reads (K2 score) from different threads map to different memory slots

(shared memory)

score reduction and identification of triplet w/ best score

between threads in thread block

(shared memory)

between thread blocks

(global memory)

K2 Bayesian score

r_{ij} → number of samples where trait takes j_{th} state and SNPs take the i_{th} genotype

$$K2 = \sum_{i=1}^{I}\left(\sum_{b=1}^{r_i+1}(\log b) - \sum_{j=1}^{J}\left(\sum_{d=1}^{r_{ij}}(\log d)\right)\right)$$

$I = 3^k$ → number genotype combinations

$I = 2$ → number of phenotypic states

r_i → frequency of the i_{th} genotype

GPU accesses a lookup table

precomputed using `lgamma()` intrinsic

$\Gamma(n) = (n-1)!$

Evaluation w/ 5 CPU+GPU systems

Systems	GPU (NVIDIA) arch. | cuda | driver	CPU (Intel) #cores | freq.	DRAM #channels | freq.	Operating System	max. perf ($\times 10^9$) (triplets / sample / second)
S1	**GeForce 2070S** Turing | 10.1 | 430.40	**Xeon E3-1245 V3** 4 | 3.6GHz	16GB DDR3 dual | 2400MHz	Ubuntu 18.04	1343
S2	**Titan V** Volta | 9.2 | 396.54	**Core i9-7900X** 10 | 4.0GHz	64GB DDR4 quad | 2400MHz	CentOS 7.5	2207
S3	**Titan XP** Pascal | 10.1 | 418.56	**Core i7-4770K** 4 | 3.5GHz	32GB DDR3 dual | 1333MHz	CentOS 7.6	1800
S4	**Titan X** Maxwell 2.0 | 8.0 | 375.26	**Core i7-5960X** 8 | 3.3GHz	32GB DDR4 quad | 2133MHz	Fedora 21	991
S5	2×**GeForce 980** Maxwell 2.0 | 8.0 | 410.48	**Core i7-6700K** 4 | 4.2GHz	32GB DDR4 dual | 2133MHz	CentOS 7.3	1476

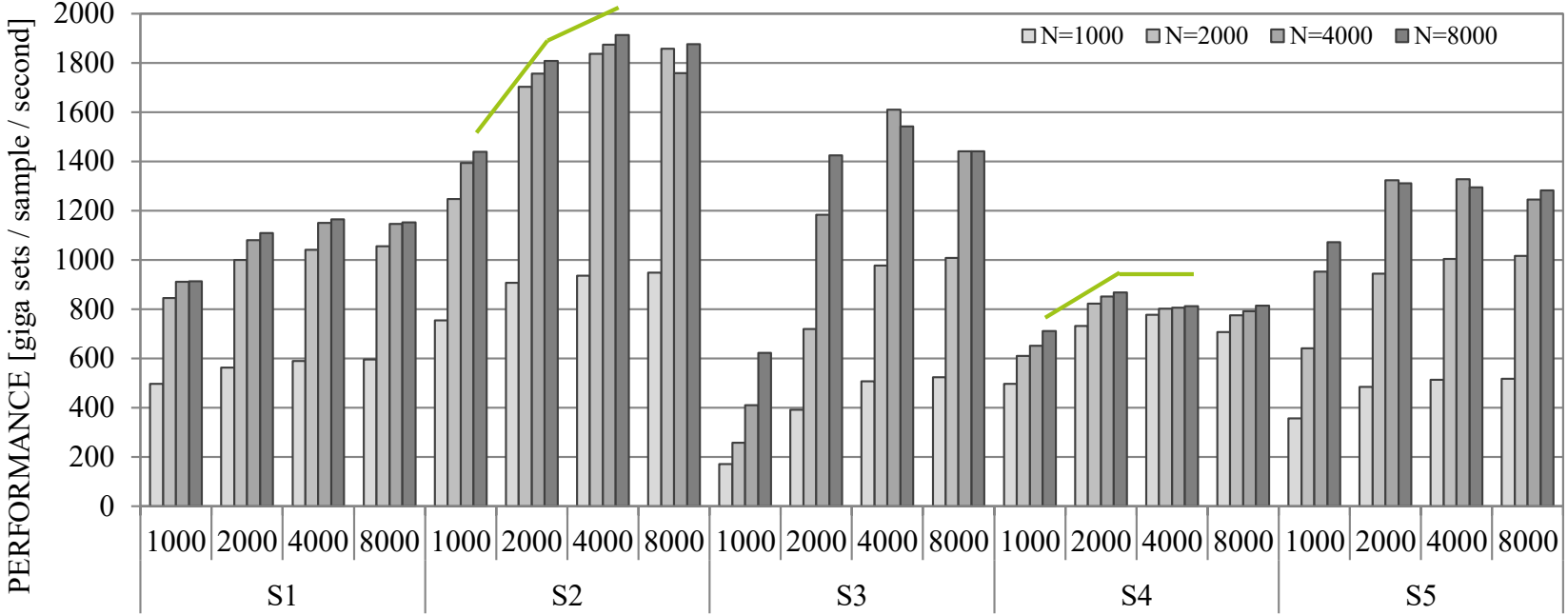
Max. theoretical perf. calculated based on POPC throughput and GPU clocks

each POPC processes 32 bits

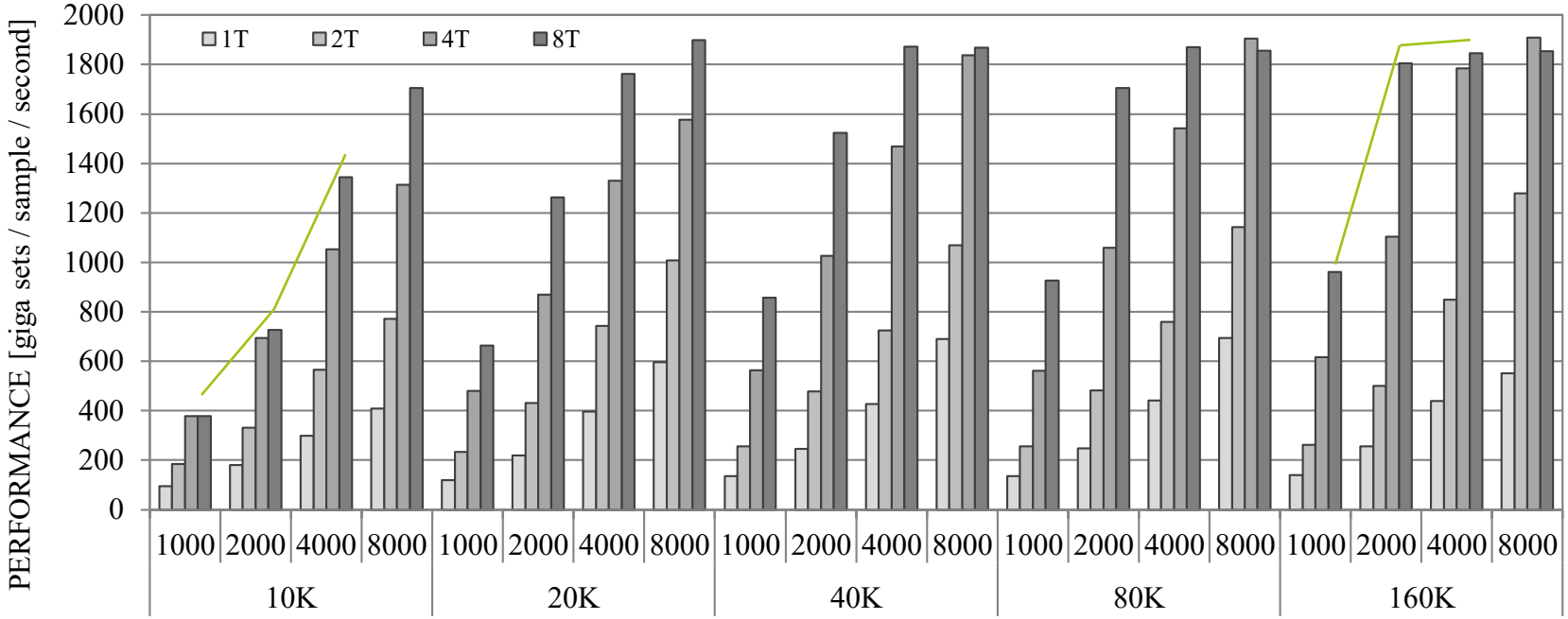
27 genotypes / triplet of SNPs

1.185 × rate of POPC instructions / second

Performance on different CPU+GPU systems

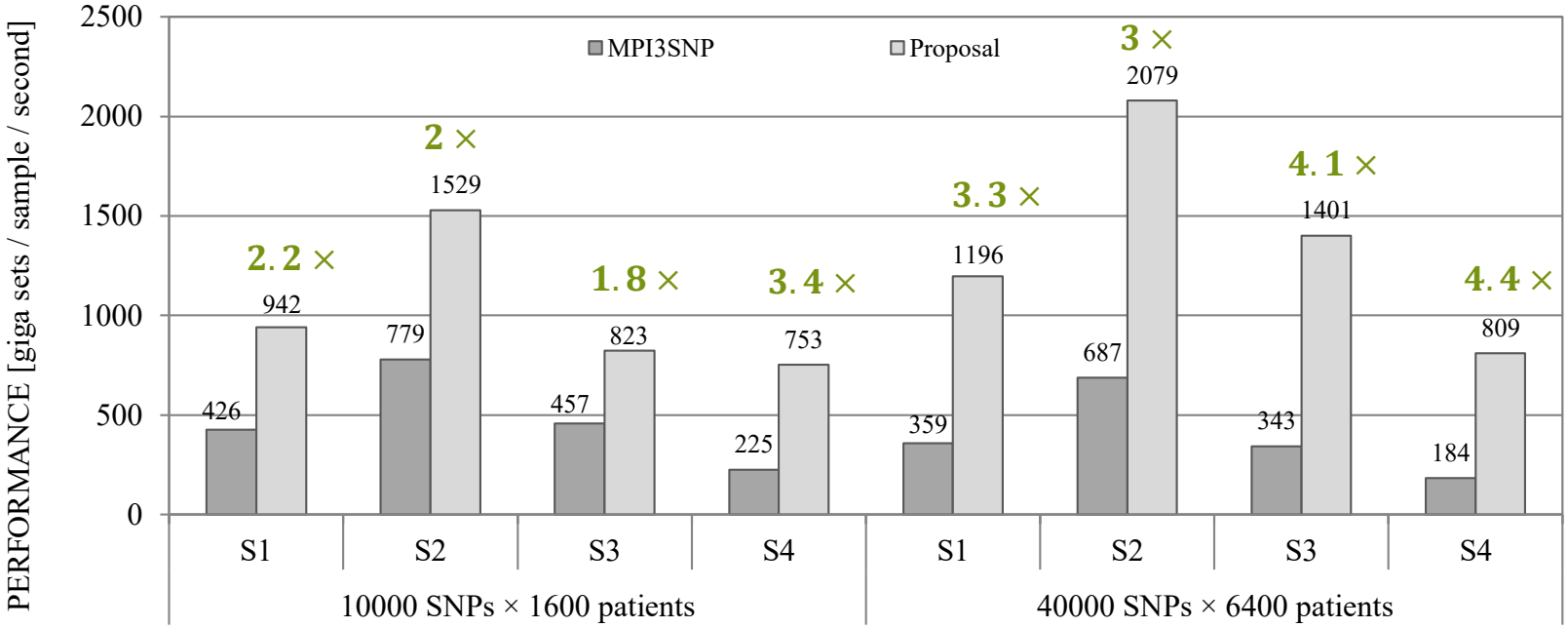


Effect of chunk size and number of CPU threads



4K SNPs on S2

Proposal vs. MPI3SNP[1]



[1] C. Ponte-Fernández, Fast search of third-order epistatic interactions on CPU and GPU clusters. IJHPCA, 2020

Datasets from: https://github.com/chponte/mpi3snp/wiki/Sample-files

Conclusions

(on 5 CPU+GPU systems)

close to max. performance of native POPC inst.

(evaluated on 4 GPU architectures)

high degree of compatibility

(~3 × vs. MPI3SNP)

faster than contemporary related art targeting CUDA cores

Ongoing work

Add support for GPUs from other vendors and multiple nodes

Generalize to higher order (4-way searches)

Thank you!

inesc id lisboa 20 YEARS

DEFINING TECHNOLOGY

FCT Fundação para a Ciência e a Tecnologia

EUROPEAN UNION
European Regional Development Fund

Supported by national funds through FCT, under project UIDB/50021/2020 and Grant SFRH/BPD/119220/2016, and the ERDF, under project LISBOA-01-0145-FEDER-031901 (PTDC/CCI-COM/31901/2017, HiPErBio).