
# an HPC framework for large scale graph exploration and inference (MLHPCS)

Marc Josep, Xavier Teruel, Victor Giménez, et al.

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

# Outline

Introduction

Related work and context

Analysis, design and implementation

Experimental results

Conclusions

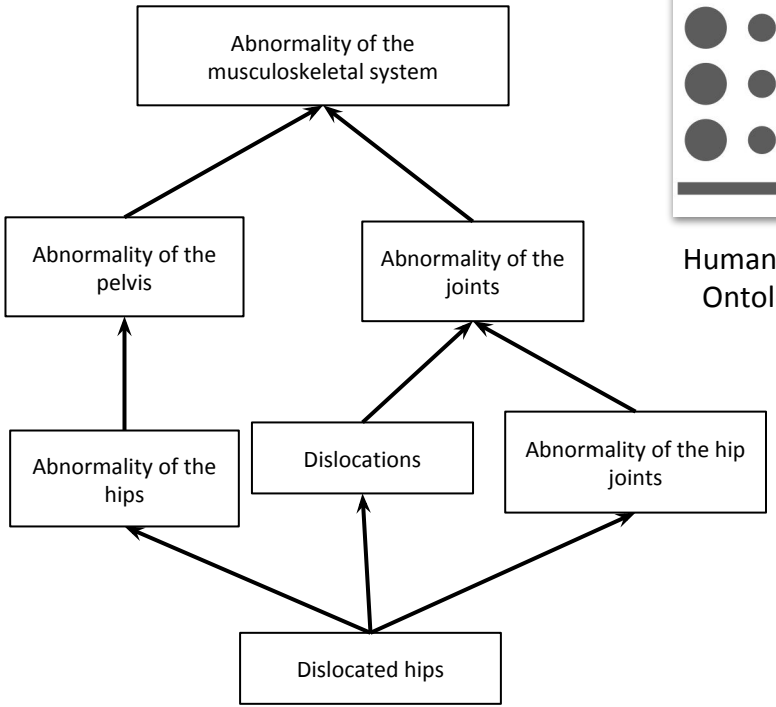
Future work

# Introduction

**Ontologies** are widely used in Biomedicine

- Data integration & interoperability
- Interpretation of high-throughput experiments and clinical information
- E.g., HPO describes phenome abnormalities (symptoms)

**HPC and AI solutions** are needed to traverse and model the interconnectivity of **multiple ontologies**.





Human Phenotype Ontology (HPO)

# Related work and context

Open Biomedical and Biological Ontologies (OBO) library

- best practices
- curated corpora of ontologies

Phenotypic and genotypic relationships studies → Identify molecular drivers underlying human diseases

Genome-wide association studies (GWAS)

- biological complexity
- lack of consensus
- susceptibility

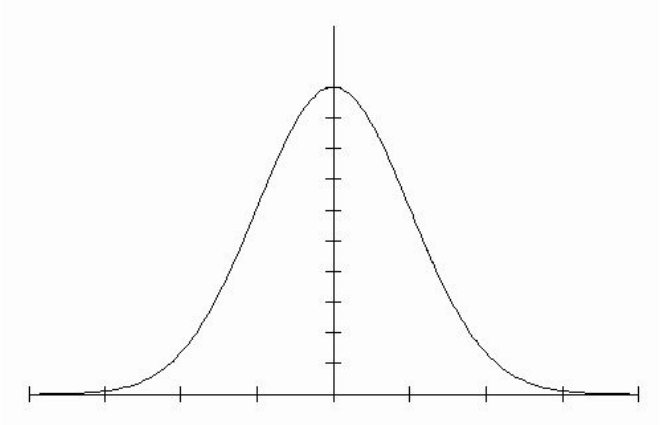
} **Limiting factors**

**HPC:** Message Passing Interface (*MPI*), Open Multi-Processing (*OpenMP*), and OpenMP SuperScalar (*OmpSs*)

# Analysis and design (methods)

**Probability**, assuming a gaussian distribution,...

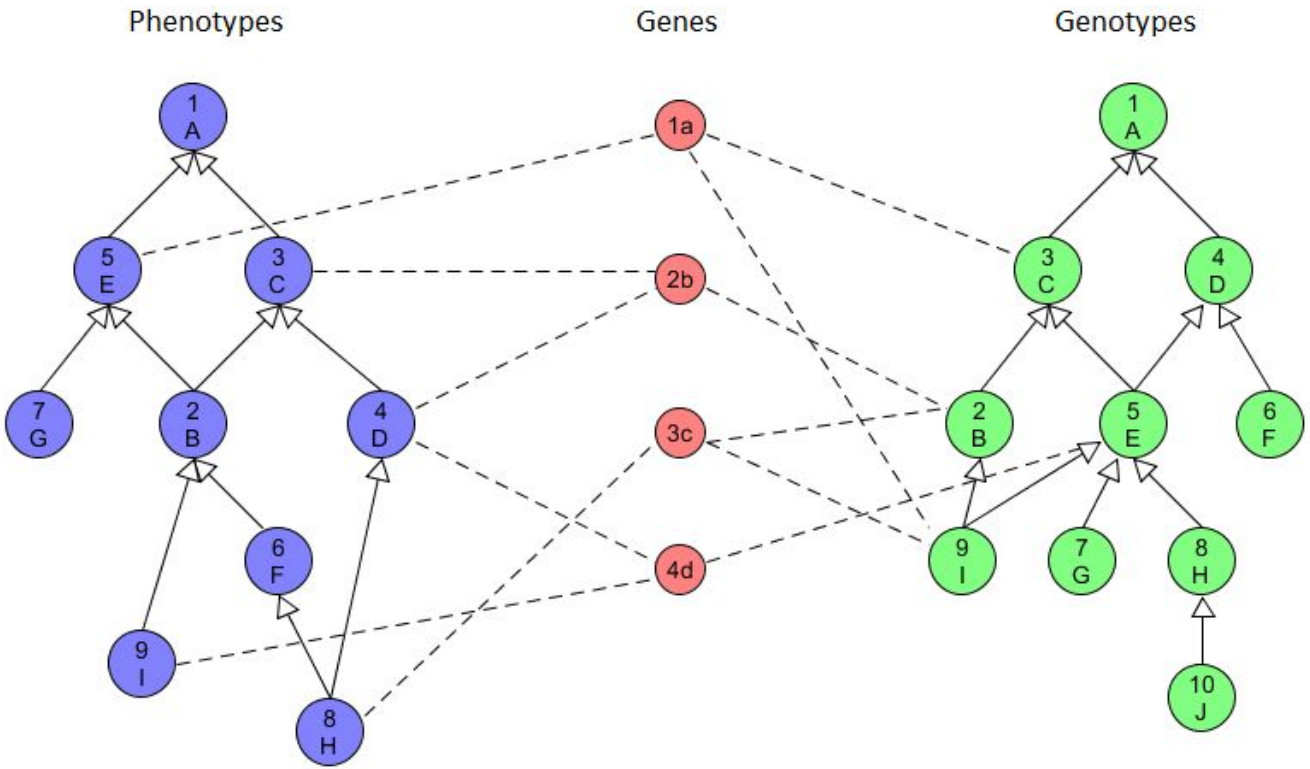
$$P(c|\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2}$$



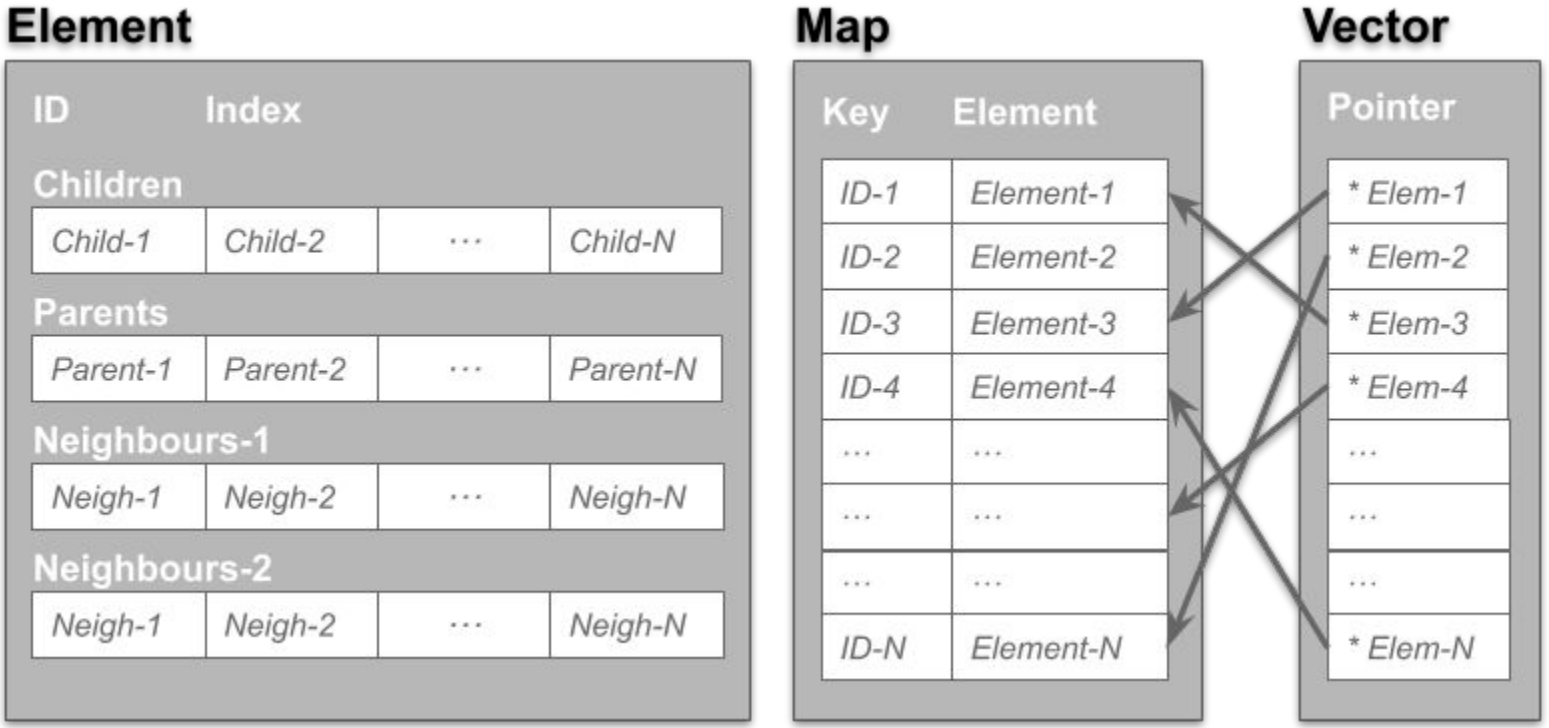
… and computing the **odds**

$$Odds(connected|Counts) = \prod_{pt \in pathtypes} \frac{P(Count(pt)|\mu_{con}(pt), \sigma_{con}(pt))}{P(Count(pt)|\mu_{discon}(pt), \sigma_{discon}(pt))}$$

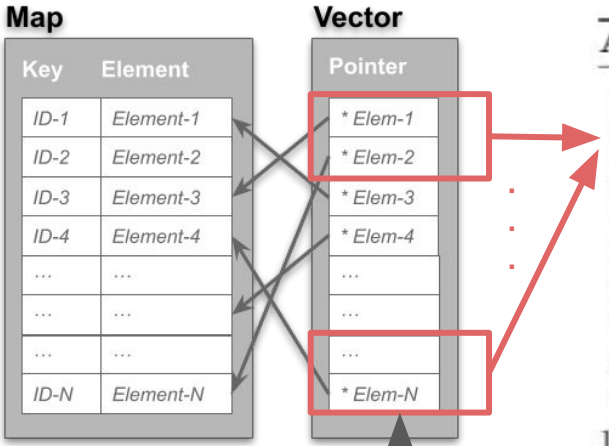
# Implementation: the graph



# Implementation: data structures



# Implementation: MPI Implementation (baseline)

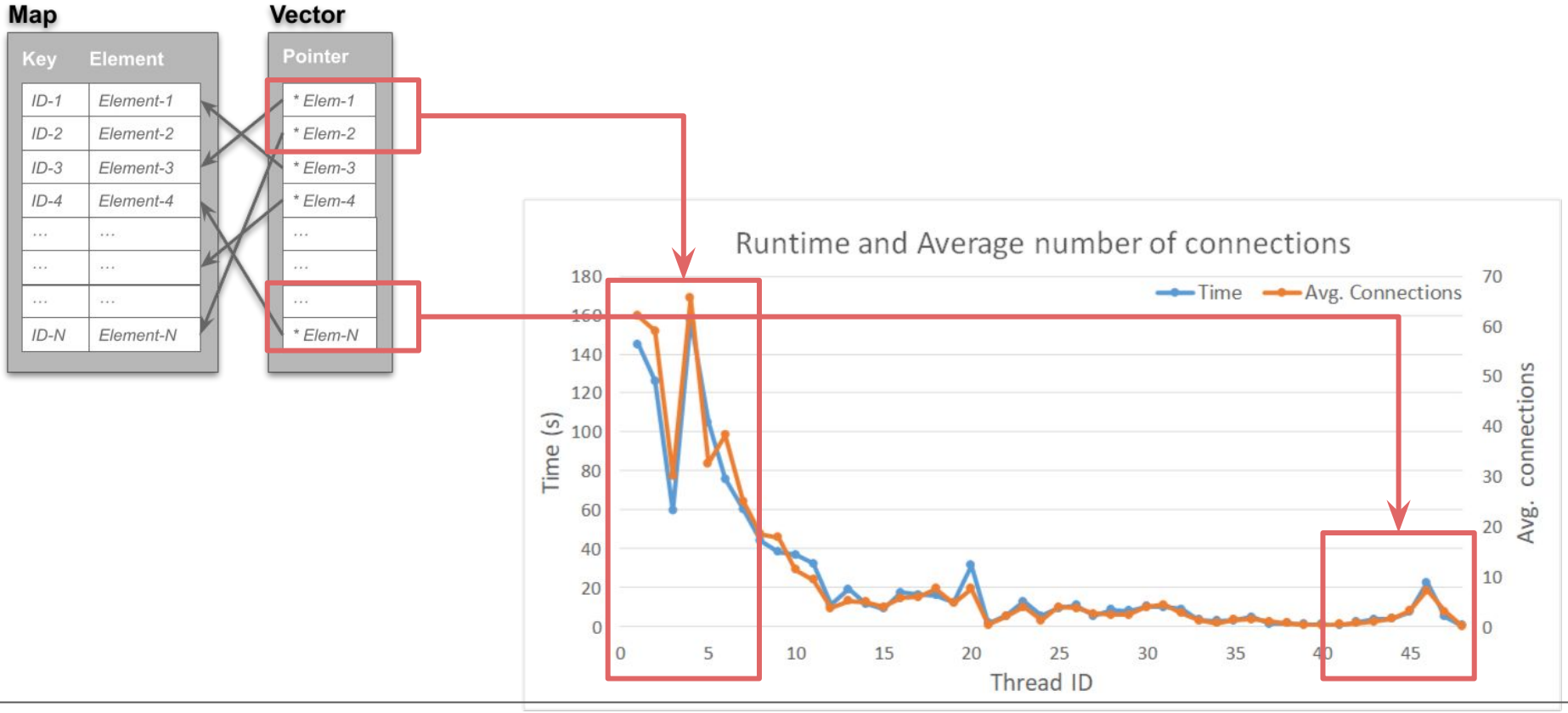



**Algorithm 2** MPI parallelization with communication

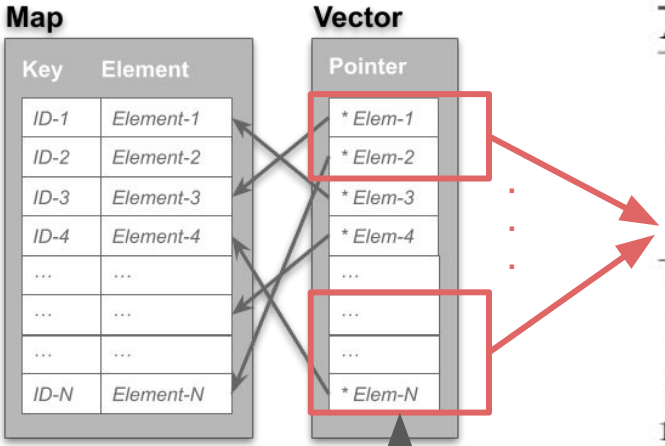
```
1: chSize ← Ontology 1 size / num Ranks
2: (start, end) ← (my Rank ∗ chSize, my Rank ∗ chSize + chSize)
3: (nPaths, nPairs) ← (0, 0)
4: for i ← start, end do
5:     for j ← 0, Ontology 2 size do
6:         nPaths ← nPaths+ Search Paths(Ontology 1(i), Ontology 2(j), path type)
7:         nPairs ← nPairs + 1
8: MPI_AllReduce (nPaths, nPairs)
9: (average, st dev) ← (nPaths/nPairs, Calculate Local St Dev)
10: MPI_Reduce (st dev)
11: if my Rank = 0 then
12:     st dev ← Calculate Overall St Dev
```

**Even number of elements distributed among MPI ranks**

# Implementation: the load imbalance problem



# Implementation: MPI Implementation (balancing)

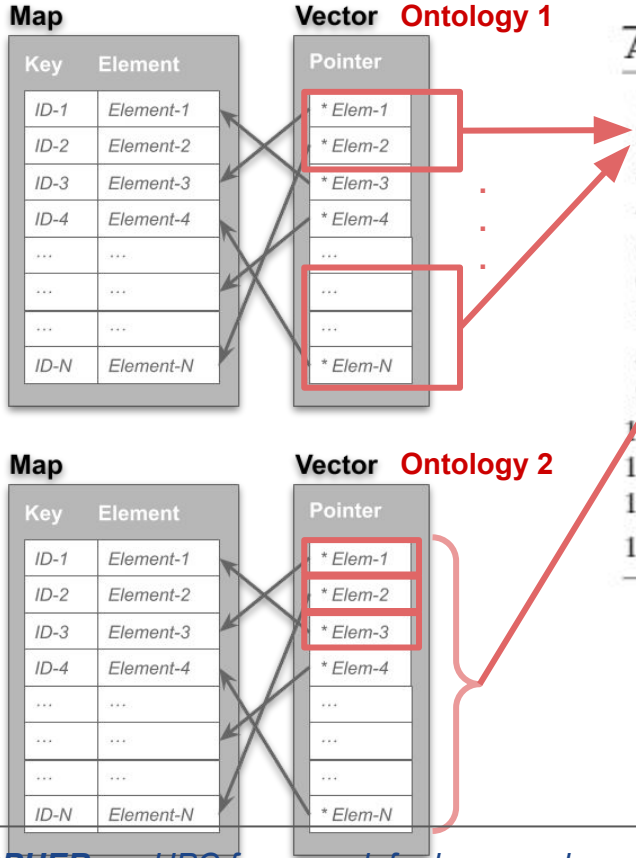



**Algorithm 3** MPI parallelization with new work distribution

```
1: total work ← 0
2: for i ← 0, Ontology 1 size do
3:     total work ← total work + connections(i)
4: chSize ← total work / num Ranks
5: (start, end) ← (begin(my Rank, chSize), end(my Rank, chSize))
6: (nPaths, nPairs) ← (0, 0)
7: for i ← start, end do
8:     for j ← 0, Ontology 2 size do
9:         nPaths ← nPaths+ Search Paths(Ontology 1(i), Ontology 2(j), path type)
10:        nPairs ← nPairs + 1
11: MPI_AllReduce (nPaths, nPairs)
12: (average, st dev) ← (nPaths/nPairs, Calculate Local St Dev)
13: MPI_Reduce (st dev)
14: if my Rank = 0 then
15:     st dev ← Calculate Overall St Dev
```

**Different number of elements, according with the number of connections, distributed among MPI ranks**

# Implementation: OpenMP/OmpSs implementation




**Algorithm 4** OmpSs parallelization with reduction

```
1: (nPaths, nPairs) ← (0, 0)
2: for i ← start, end do
3:     for j ← 0, Ontology 2 size do
4:         # pragma omp task reduction(+, nPaths)
5:         nPaths ← nPaths+ Search Paths(Ontology 1(i), Ontology 2(j), path type)
6:         nPairs ← nPairs + 1
7: # pragma omp taskwait / barrier
8: MPI All Reduce (num paths, num pairs)
9: (average, st dev) ← (nPaths/nPairs, 0)
10: # pragma omp parallel for reduction(+, st dev)
11: for i ← 0, start − end do
12:     st dev ← st dev + (nPaths(i) − average)^2
13: MPI Reduce (st dev)
```

# Experimental results: model use case

## Use case

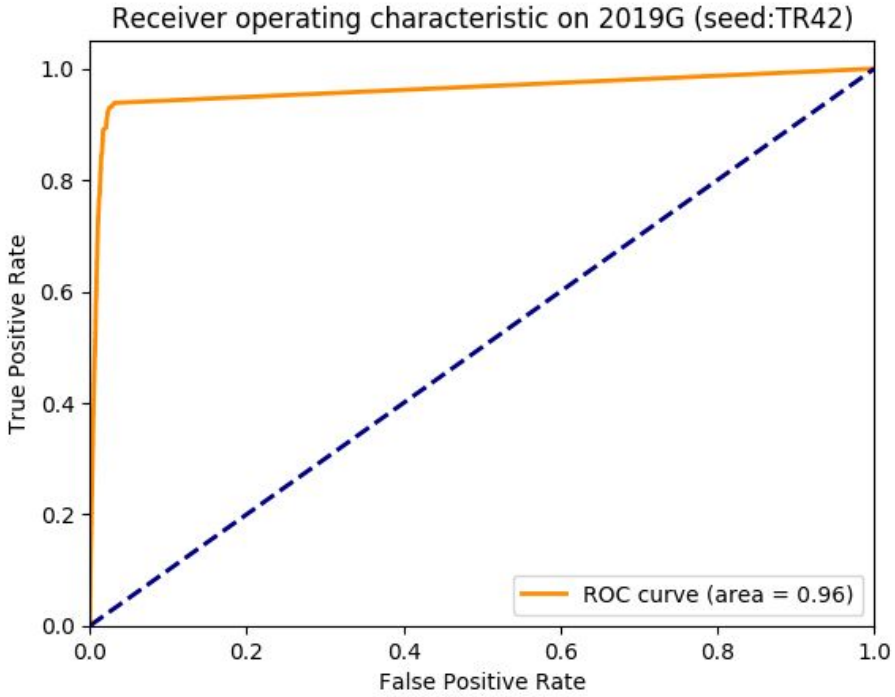
- **Ontologies:** GO and HPO, only human, version: January 2019.
- **Path size:** all types of paths with size of 4 or 5 elements.
- **Pairs nature:** from phenotypes to genotypes.
- **Samples:** 85,750 randomly sampled pairs of both types.
- **Direct edge removal:** yes

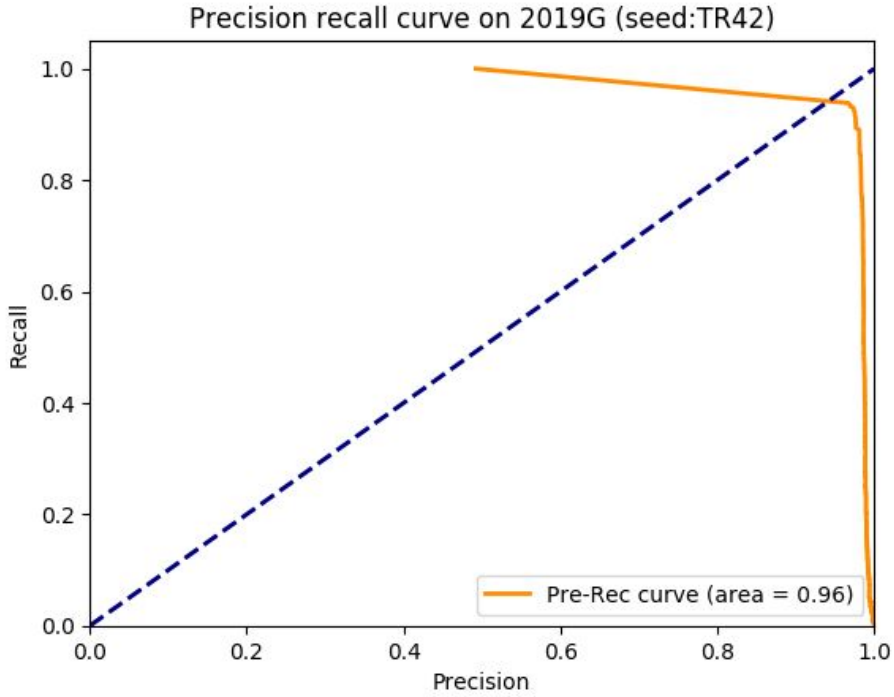
## Methods:

- Receiver Operating Characteristic (ROC)
- Precision-Recall (PR) curves

**Algorithm:** *Each path type possible of up to length 5*

# Experimental results: model validation



# Experimental results: performance use case

**Environment -** MareNostrum IV cluster, located at BSC, each node:

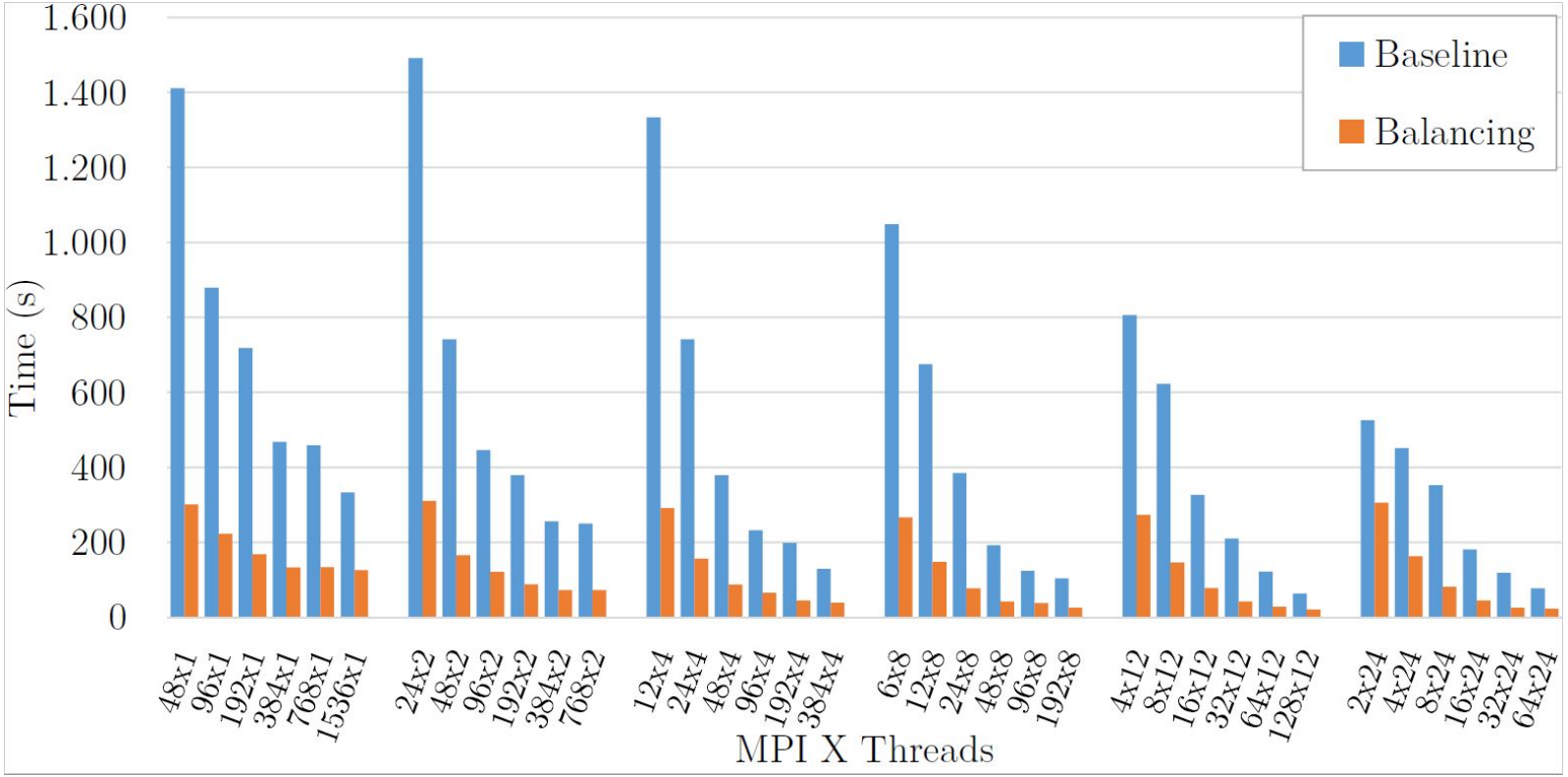
- 2 Intel Xeon Platinum 8160, running at 2.1 GHz
- 48 cores (i.e., 24 per processor) and 33 MB L3 Cache
- 2 NUMA sockets (i.e., 1 socket per processor), 192GB per socket

**Use case**

- **Ontologies:** GO and HPO, only human, version: January 2019.
- **Path size:** all types of paths up to a size of 5 elements.
- **Pairs nature:** from phenotypes to genotypes.
- **Samples:**100,000 randomly sampled pairs (constant seed).
- **Direct edge removal:** yes

**Algorithm:** *Number of paths for each path type*

# Experimental results: scaling factors



# Conclusions

Introduce the GOPHER framework for large graph exploration and inference

> *"estimate the likelihood that two ontology terms are associated when missing a direct connection through a co-annotated gene"*

An interdisciplinary work:

- A Biological topic;
- A Machine Learning approach;
- By means of High Performance Computing technology

Preliminary results

- Model analysis: obtaining an AUC score of 0.96 over 1.
- Performance: load imbalance problem → balancing schedule → scalability plots

# Future Work

## The HPC approach

- Study GOPHER behaviour in other architectures
- Further performance analysis (explore other metrics, trace analysis)
- Optimisation opportunities
  - Improve intra-node balance: use Dynamic Load Balancing library
  - Improve inter-node balance: use OmpSs@Cluster

## The ML approach

- Further study (and validation) of the proposed thesis

## The Biological approach

- Other actionable use cases: anticancer treatment recommendations
- Other biological ontologies: mouse and fruit fly

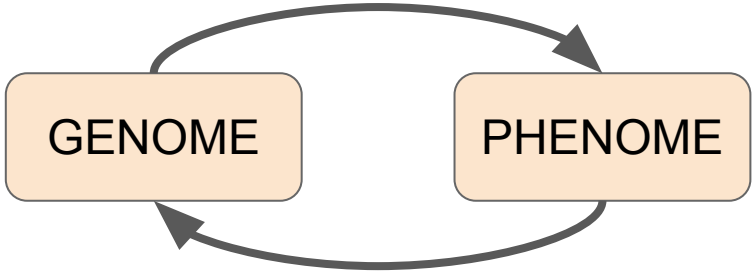

Thanks!
Further information at:
https://www.linkedin.com/in/xteruel
BSC
Barcelona
Supercomputing
Center
Centro Nacional de Supercomputación

# Related work and context

Open Biomedical and Biological Ontologies (OBO) Foundry

- best practices
- curated corpora of ontologies

www.obofoundry.org




Phenotypic and genotypic relationships studies → Identify molecular drivers underlying human diseases

**HPC:** Message Passing Interface (*MPI*), Open Multi-Processing (*OpenMP*), and OpenMP SuperScalar (*OmpSs*)