

pcj-blast - highly parallel similarity search implementation

Piotr Bała

bala@icm.edu.pl

ICM University of Warsaw

Marek Nowicki

faramir@mat.umk.pl

ICM University of Warsaw

Davit Bzhalava

N. Copernicus University

davit.bzhalava@ki.se

Karolinska Institutet

- Sequence alignment is essential for NGS
 - There is a number of software packages for sequence alignment based on various a similarity search methods
 - BLAST Basic Local Alignment Search Tool (1991)
 - The heuristic algorithm it uses is much faster than other approaches
 - The search time can be long (days or weeks) for large datasets
 - NCBI blast is the most widely used implementation
-
- There is strong interest in using large computer systems to run blast
 - Blast running on cloud or grid
 - Parallel versions of blast running on HPC systems

Fasta input (2 reads out of 1 milion)



>C1093377_2.0

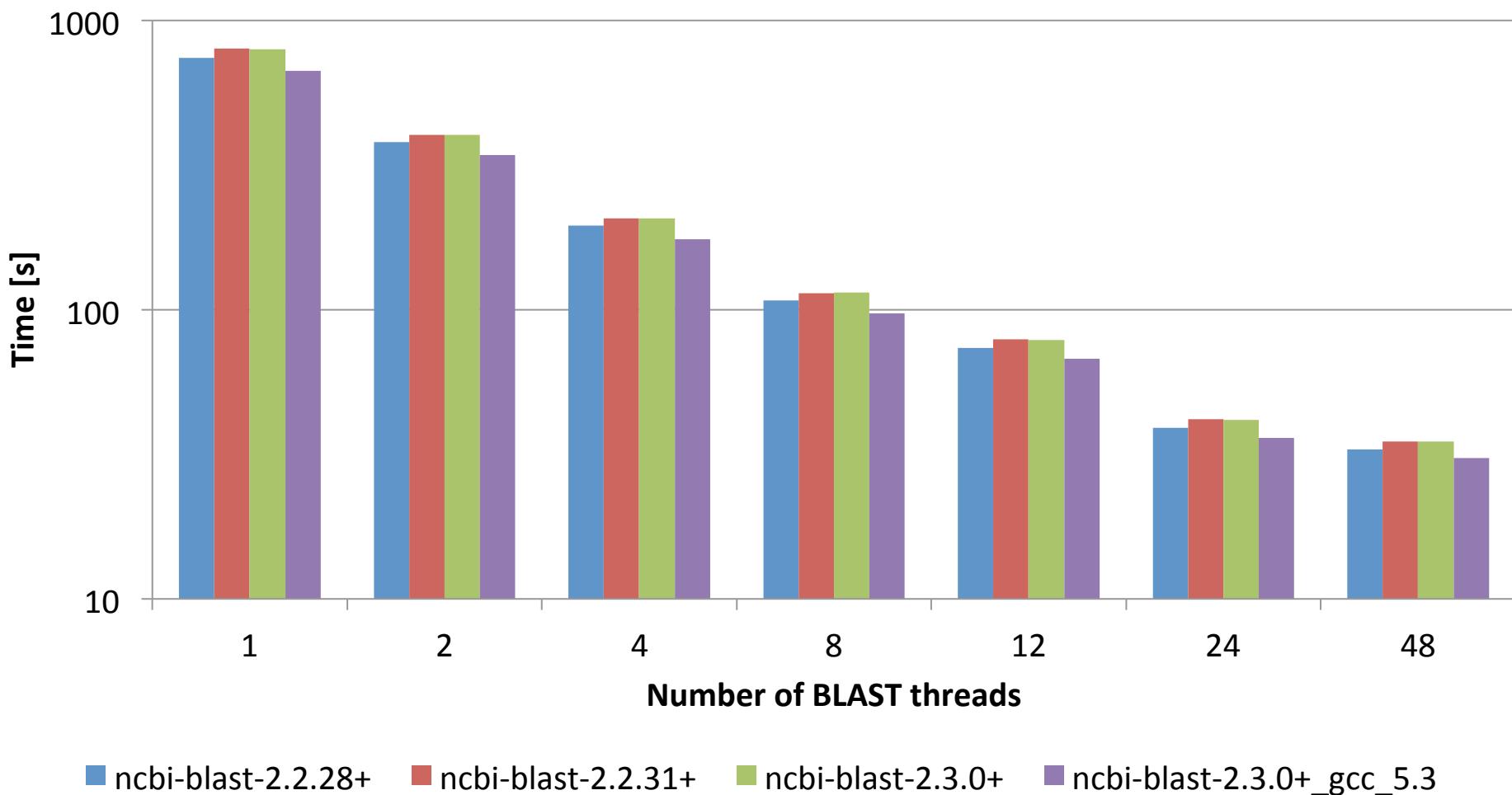
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAACACTTGGGAGGCCAAGATGGGAAGATCTTTGAGG
CCAGGCAGTCAAGACCAGTCTGGCAATATGGAGAGACCT
GTCTCTACAAAAAAATTAGCCAGACCTAGTGGCTGGCTGA
GGCAGGAGGATCATCTGAGCTCAGGAGATTGAGAT

>C1093379_2.0

AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAATTAAAGTTTCCTTATAAGGGAAATAAGCTTATTCT
TTTAGAACGAGAAACTGCAAATGGGGGTGTATGGAATTCTT
TTTTTTTTTTTGCGACAAGGTCTCACTGTGTTGCCCA
GACTGGAGTGCAGTGGTGCAATCATAGCT

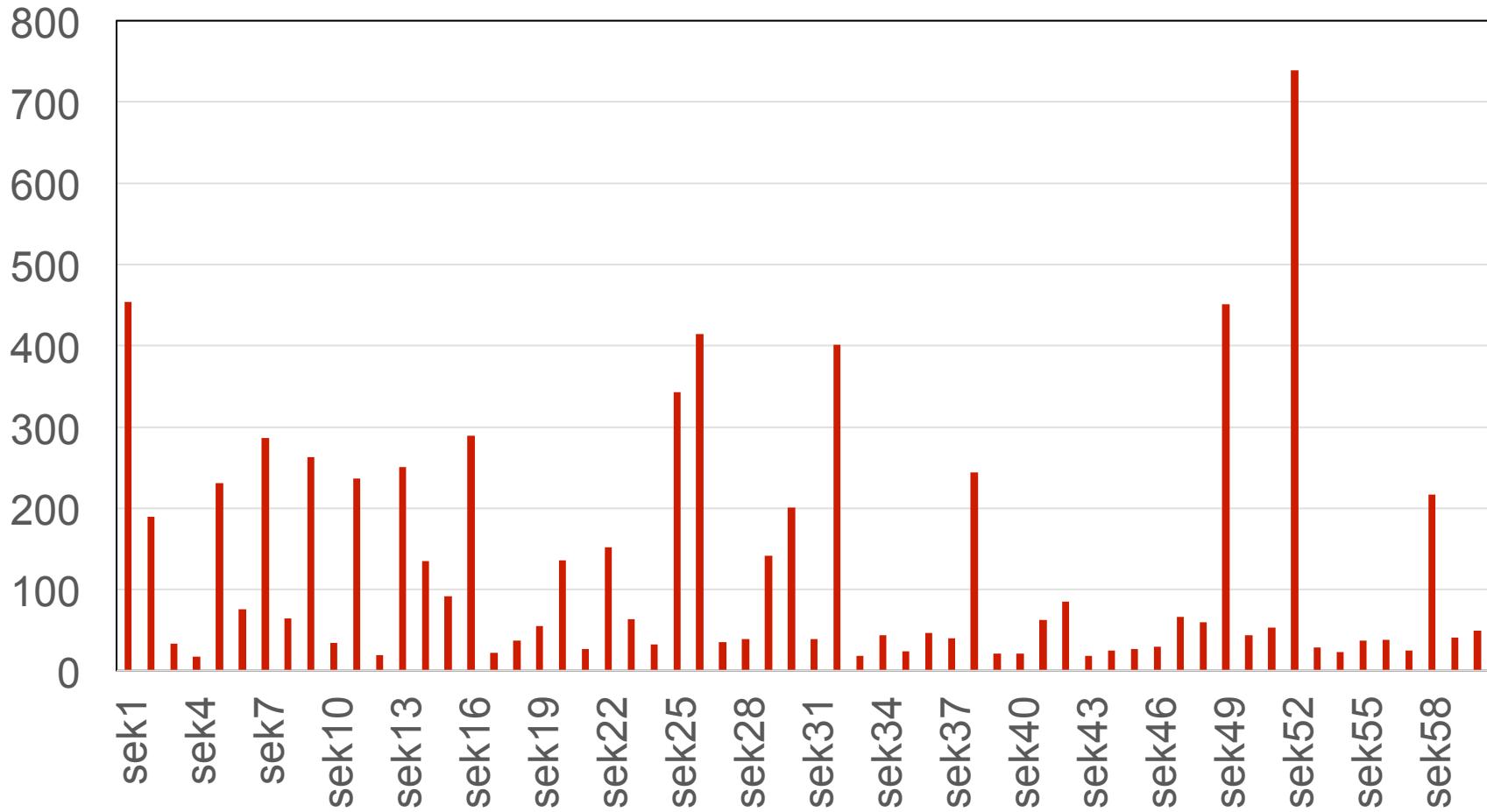
...

Time of processing 48 sequences at once using 1 instance of different version of NCBI BLAST with different number of BLAST threads



Blast – processing time of reads changes

Processing time [s]

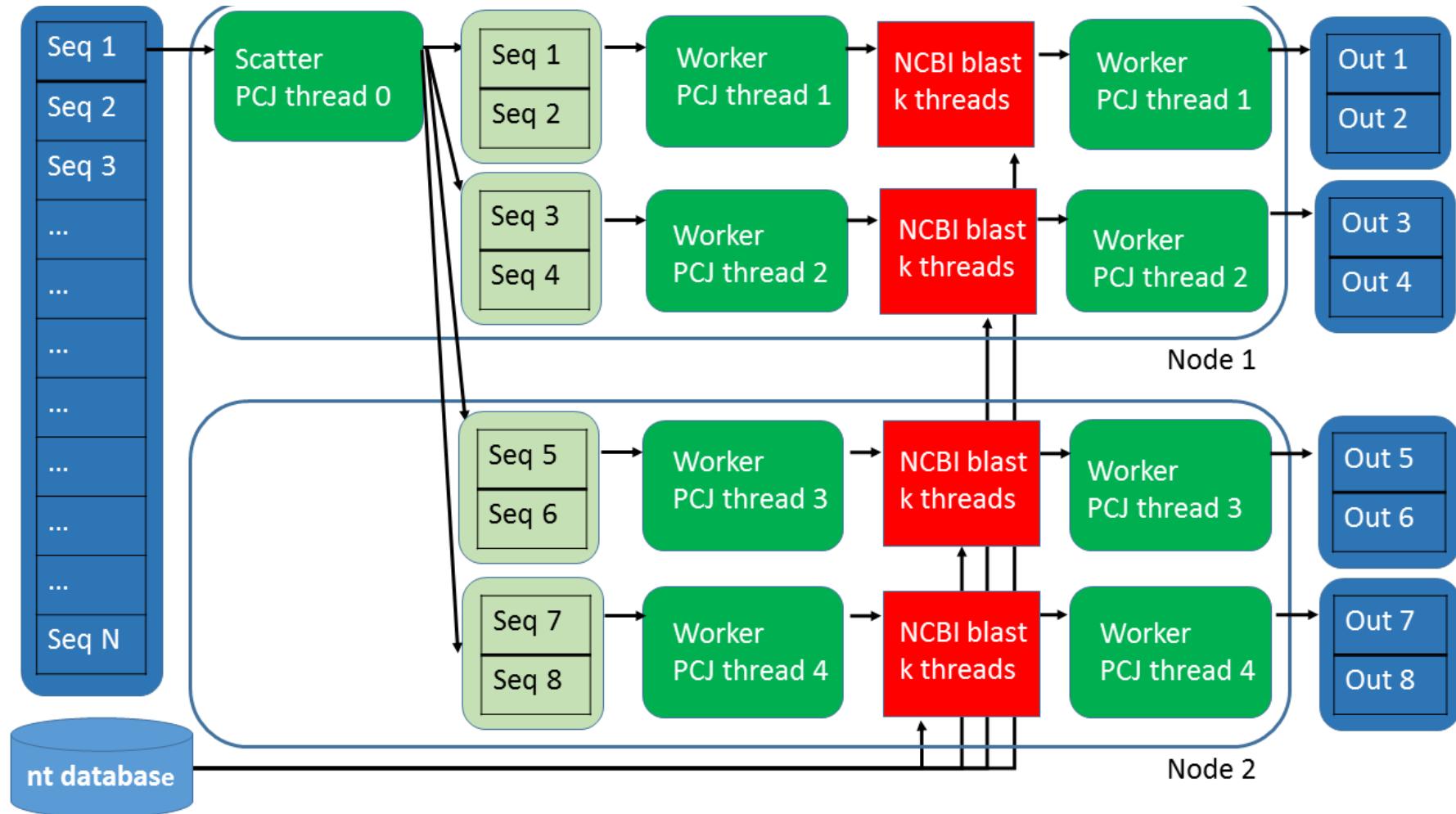


Parallel blast implementations

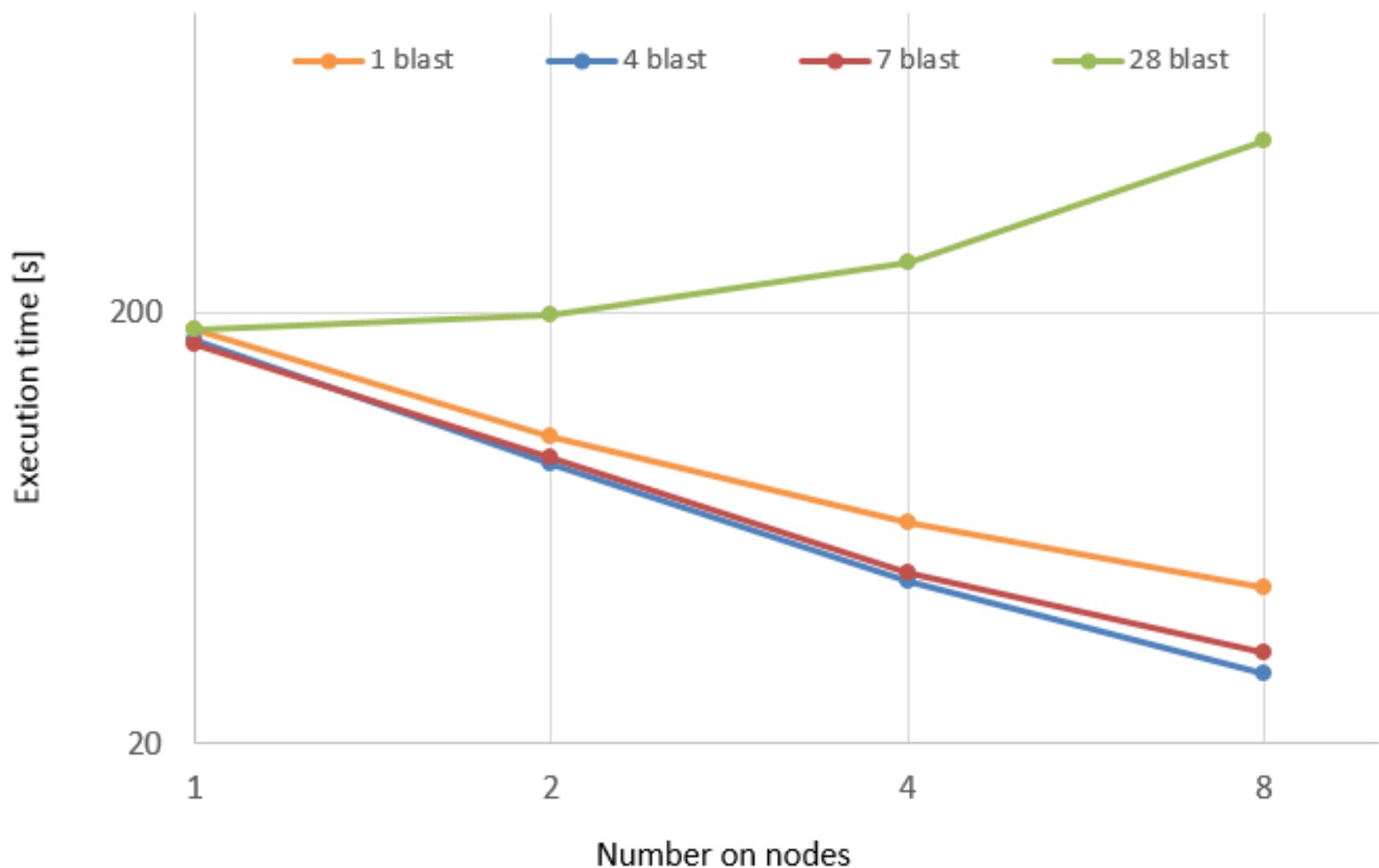
- mpiBLAST – uses old version (2012) of blast
- parallel_blast.py – limited to single node, regular split of input
- Paracel BLAST (based on NCBI blast 2.2.26) – costly!
- The BlueGene implementation – limited to IBM BGQ systems
- pioBLAST parallel version which uses MPIIO – memory problems while reference database is large
- ...

- Parallel version developed using PCJ (Parallel Computing in Java library developed at ICM)
- Java code is responsible for reading fasta input and execute NCBI-blast on set of sequences (reads)
- The NCBI blast (currently 2.2.28) is executed with the `-num_threads` option
 - For 28 core nodes: 4 PCJ threads with `-num_threads=7`
- The solution can be executed on any multinode system with Java 8 installed
- The I/O performance is important
 - thousands of blast instances reads blast database (52GB) concurrently
 - lustre filesystem performs worse than nfs

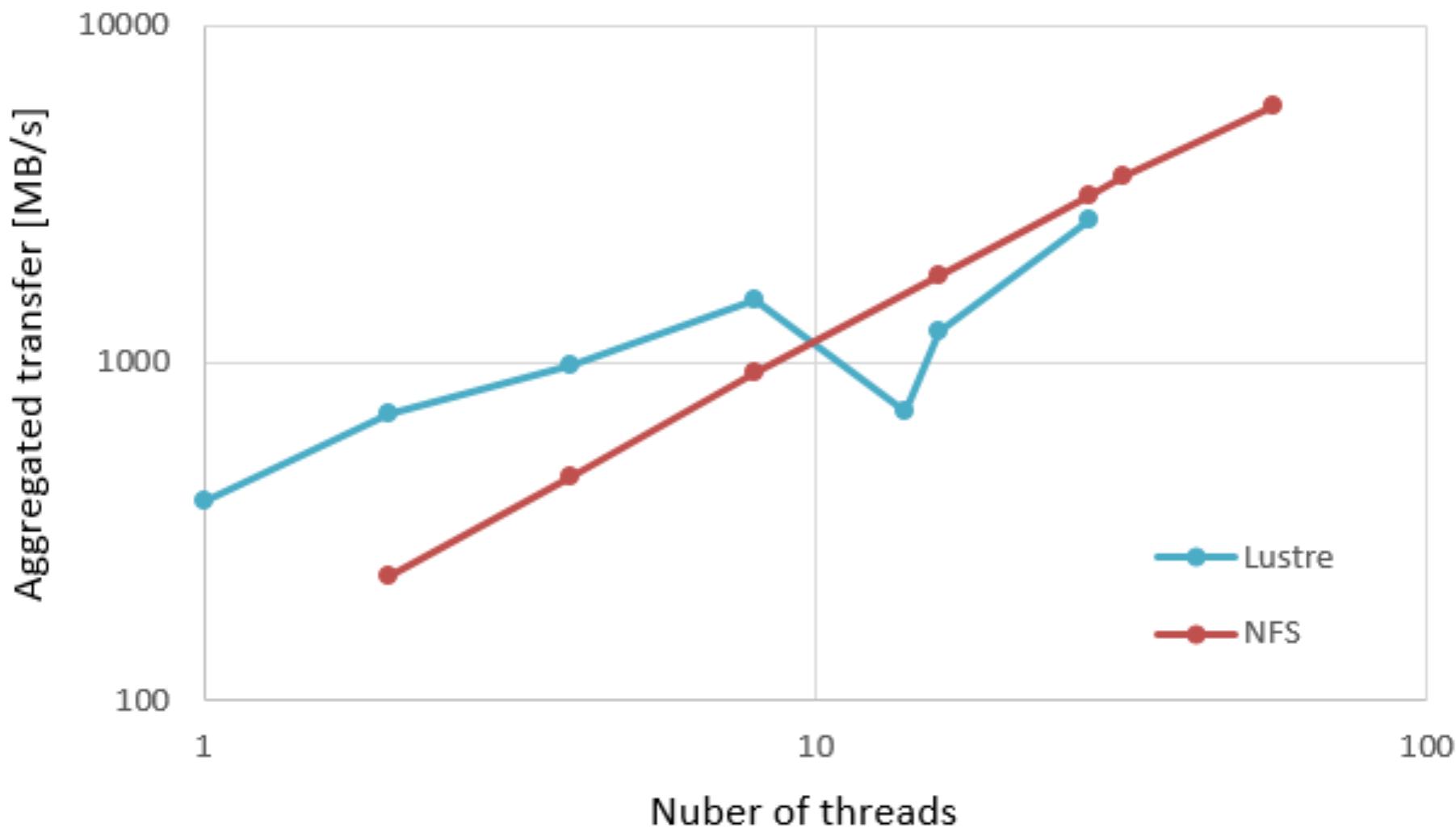
- Reads input sequence (fasta) line by line
- Blocks of lines (reads) are used as input for blast query
- Pcj-blast can start hundreds of blast instances
- NCBI blast is used
 - pcj-blast not bounded to particular version
- One or more NCBI blast instances are run on the hardware node
- Output from each node (text or XML) is gathered
- Output is postprocessed and is stored in a single file
 - Postprocessing is parallelized
- Pcj-blast can be run on HPC systems, clusters
 - Requires Java 8 and NCBI blast installed



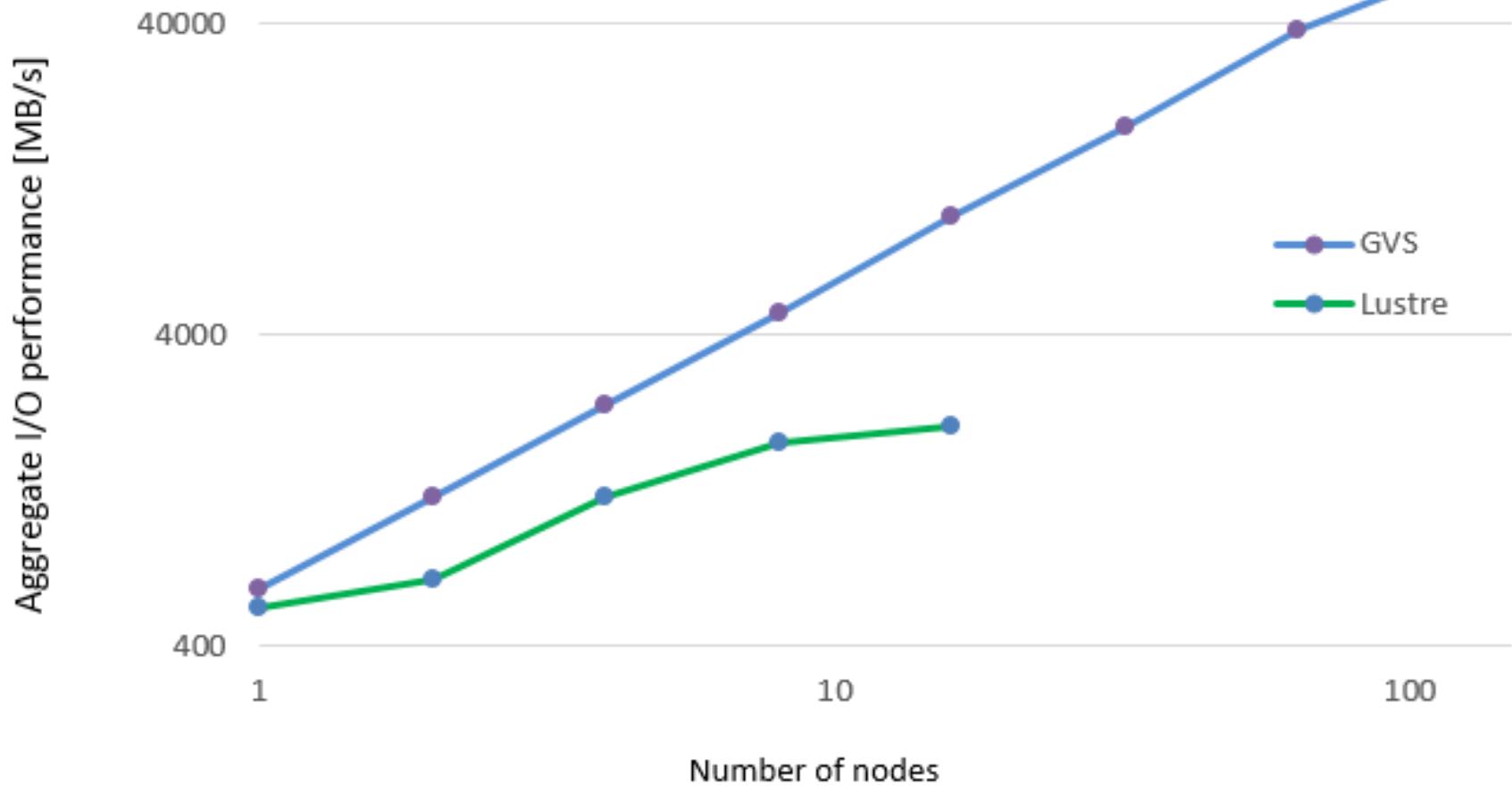
Multiple NCBI blast running on single node (X86 cluster)



I/O performance (x86 single file read)



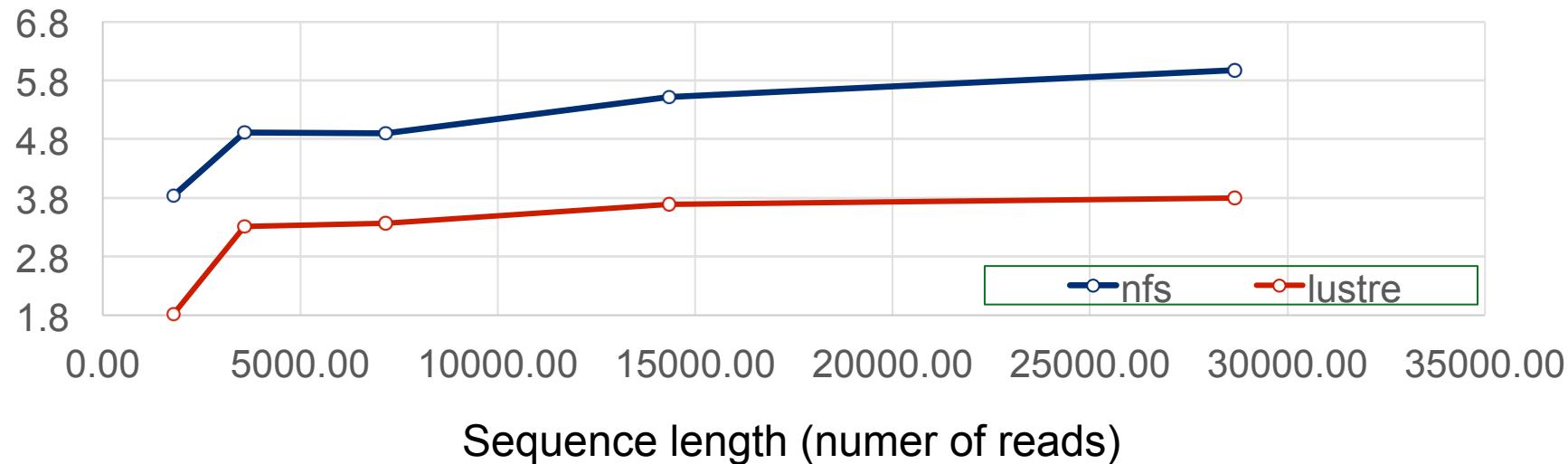
I/O performance (XC40 single file read)



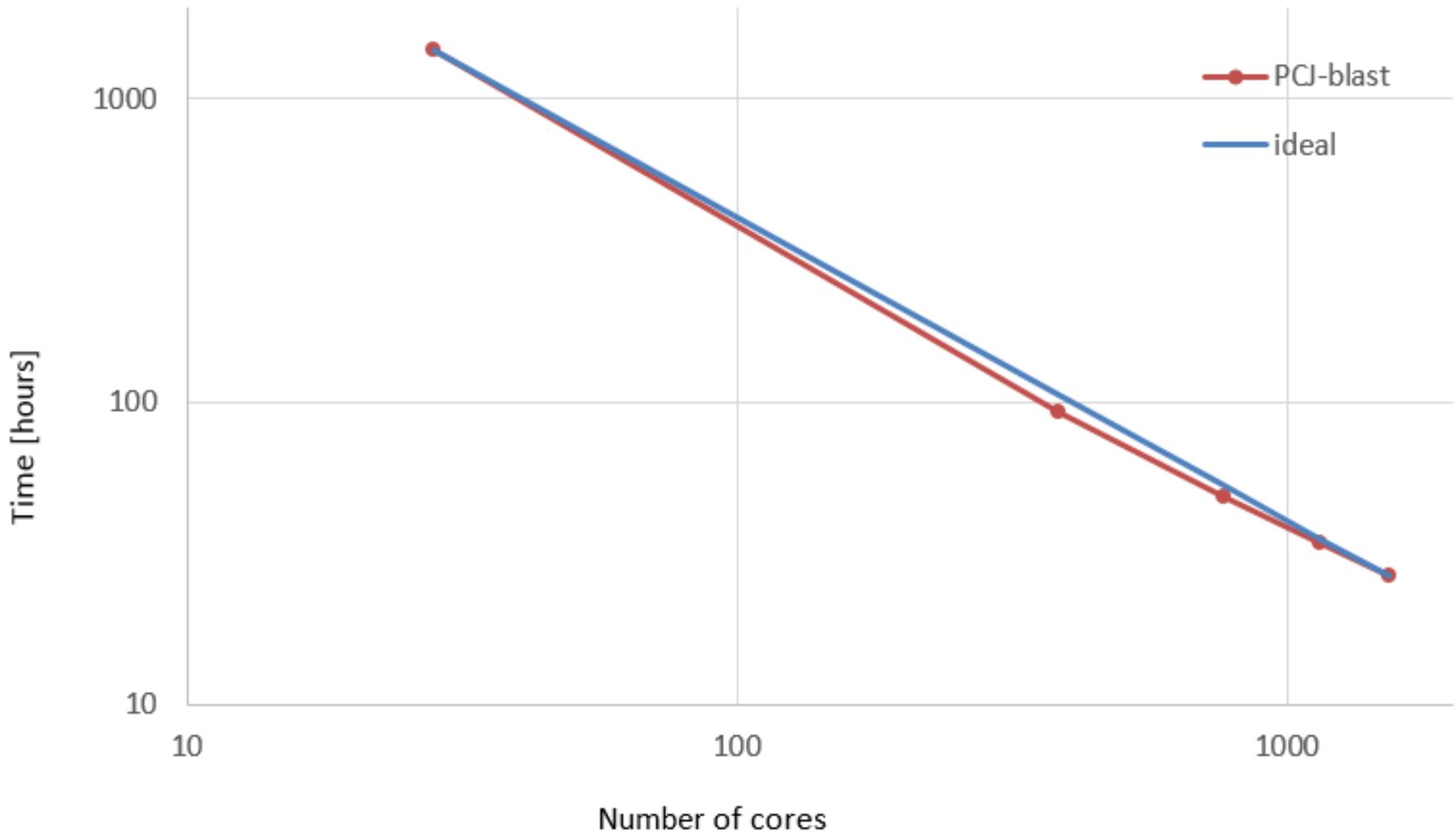
Blast (with PCJ) performance (x86 cluster)

| nodes | cores | Time [hours] | Time [days] | 1 mln reads |
|-------|-------|--------------|-------------|-------------|
| 1 | 28 | 1600.0 | 67.0 | estimated |
| 32 | 768 | 49.9 | 2.0 | |
| 48 | 1152 | 34.1 | 1.4 | |
| 64 | 1536 | 26.5 | 1.1 | |

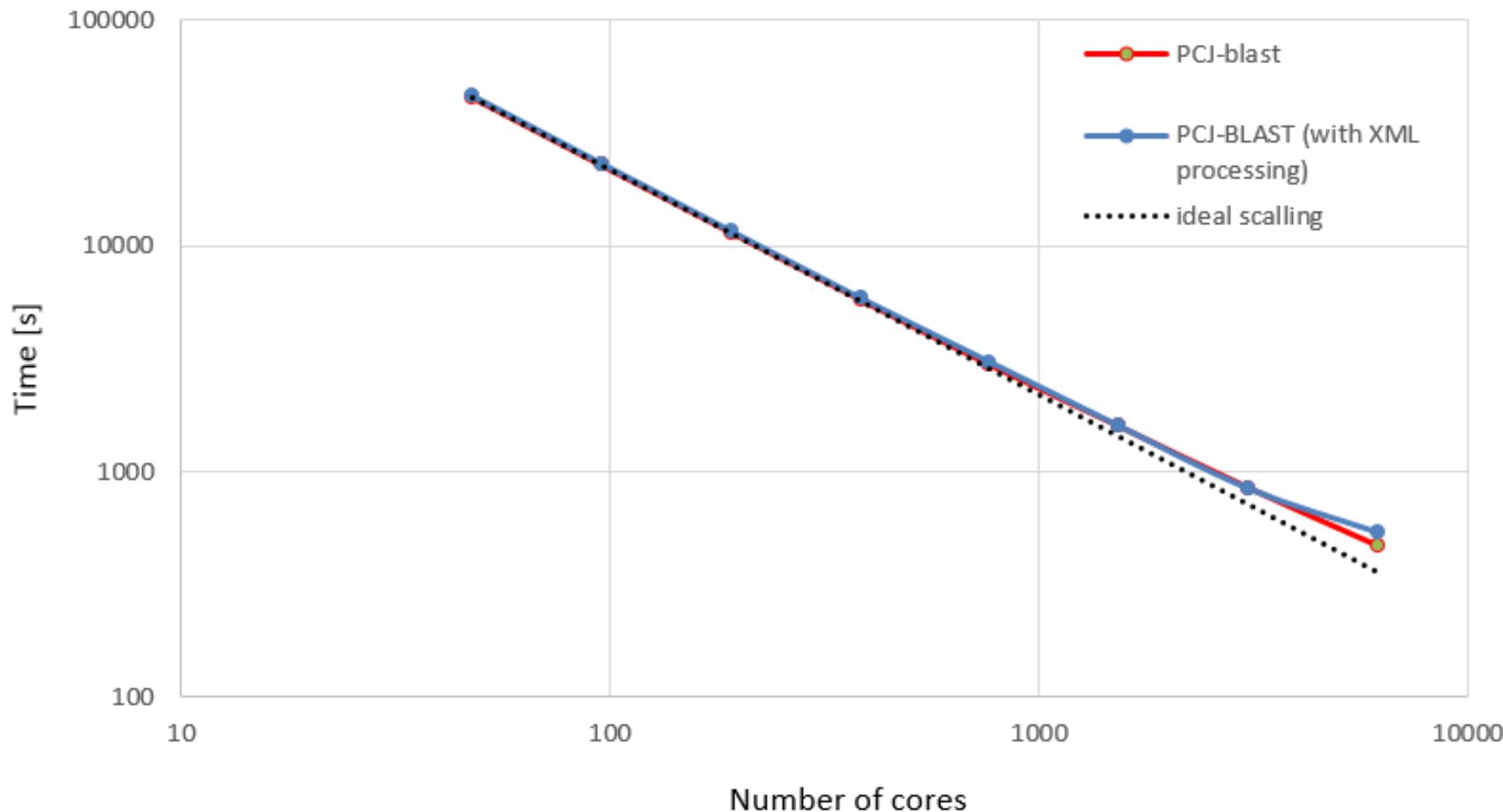
Performance [reads/s] @32 nodes



Pcj-blast scalability (x86 cluster)



pcj-blast – scalability (Cray XC40)



Workflow development

- Complicated workflow has been developed by Davit
- The bottleneck is sequence alignment with blast
 - typical data to process consists of ca. 1 mln reads
 - Input size ca. 300MB, output size (XML file) ca. 1TB
 - Blast outputs is than processed by python scripts and R

- Applications

| Application | ACK | ICM | PCSS | TASK | WCSS |
|---------------------------|-----|-----|------|------|------|
| AutoDock | X | X | | X | X |
| AutoDock Vina | X | X | | | |
| Blast | X | X | | | |
| Blast+ | X | X | | | |
| BWA | X | X | | | |
| ClustalW2 | | X | | | X |
| Custal Omega | X | X | | X | X |
| GATK | | X | | | |
| MrBayes | | X | | | |
| PicardTools | X | X | | | |
| R | X | X | X | X | X |
| GS Data Analysis Software | | X | | | |
| Rosetta | X | X | | | X |

UNICORE Rich Client

File Edit Window Help

UNICORE

FLXWorkflowBlast.... Generic Script GSMapperSFFFRL1 GSAssembler MyBlast >_2

Job name: MyBlast

Select application: BLAST+

Select version: 2.2.28

Command line arguments:

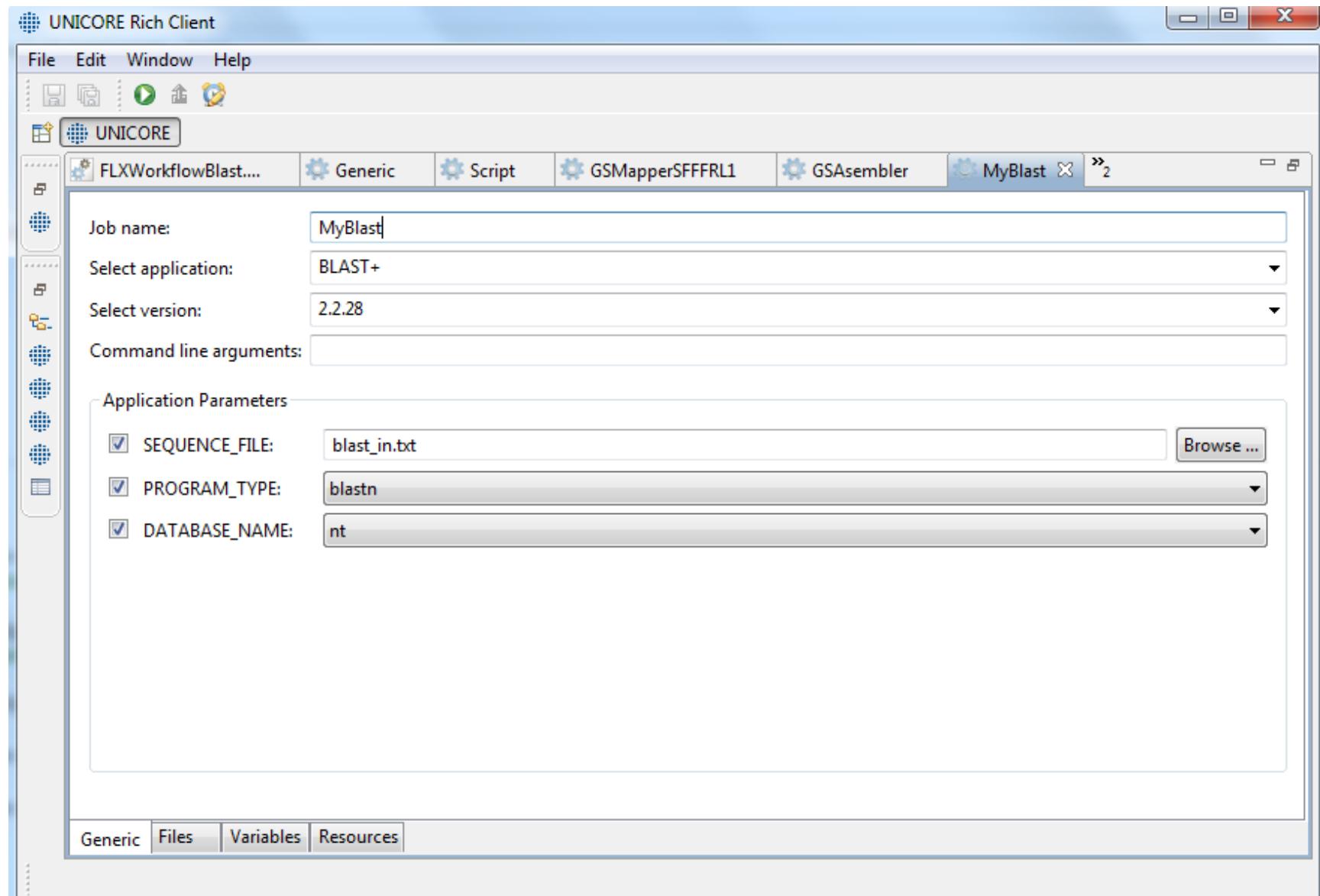
Application Parameters

SEQUENCE_FILE: blast_in.txt

PROGRAM_TYPE: blastn

DATABASE_NAME: nt

Generic Files Variables Resources



[Home](#)[Create Job](#)[My Jobs](#)[My Sites](#)[Angio Merge](#)[Sinus MED](#)

Job Computation

Alignment x [Application](#) [Input Files](#) [Output Files](#) [Resources](#)Job name: *Tag: Select application: Select version: Command line arguments:

Input parameters

 PROGRAM_TYPE: DATABASE_NAME: BLAST_PLUS_ARGUMENTS: SEQUENCE_FILE: * t d p x

Home

Create Job

My Jobs

My Sites

Angio Merge

Sinus MED

Alignment workspace

Select/Unselect all 

Path:  Show hidden files

| Name | Ext |
|--------------------------|-------|
| file.fasta | fasta |
| stderr | |
| stdout | |
| TSI_script_file_32625 | |
| UNICORE_SCRIPT_EXIT_CODE | |

https://unicore-...y/14/stdout.txt X +

BLASTN 2.2.28+

Reference: Zheng Zhang, Scott Schwartz, Lukas W Miller (2000), "A greedy algorithm for aligning Comput Biol 2000; 7(1-2):203-14.

Database: nt
20,485,291 sequences; 51,580,221,569

Query= C1093377_2.0

Length=200

Score E
Sequences producing significant alignments:
(Bits) Value

gi|2578066|emb|Z98036.1| Human DNA sequence f
333 3e-88
gi|195963538|gb|AC217054.3| MACACA MULATTA BA
110 6e-21
gi|29366933|gb|AC009054.8| Homo sapiens chromo
106 8e-20
gi|15145261|gb|AC073413.6| Homo sapiens BAC c
99.0 1e-17
gi|148727464|gb|AC198790.3| Pongo abelii BAC c
95.3 2e-16

pcj-blast - highly parallel similarity search implementation

PCJ: pcj.icm.edu.pl

Piotr Bała

bala@icm.edu.pl

ICM University of Warsaw

Marek Nowicki

faramir@mat.umk.pl

ICM University of Warsaw

Davit Bzhalava

N. Copernicus University

davit.bzhalava@ki.se

Karolinska Institutet