



Memory Affinity in Multi-threading The Bowtie2 Case Study



Claudia Misale, Marco Aldinucci
University of Torino, Italy

Massimo Torquati
University of Pisa, Italy

Memory bound problem

Synchronisation
efficiency problem for
very fine grain tasks
($<1\text{ms}$)

Load balancing problem
for coarse grain tasks

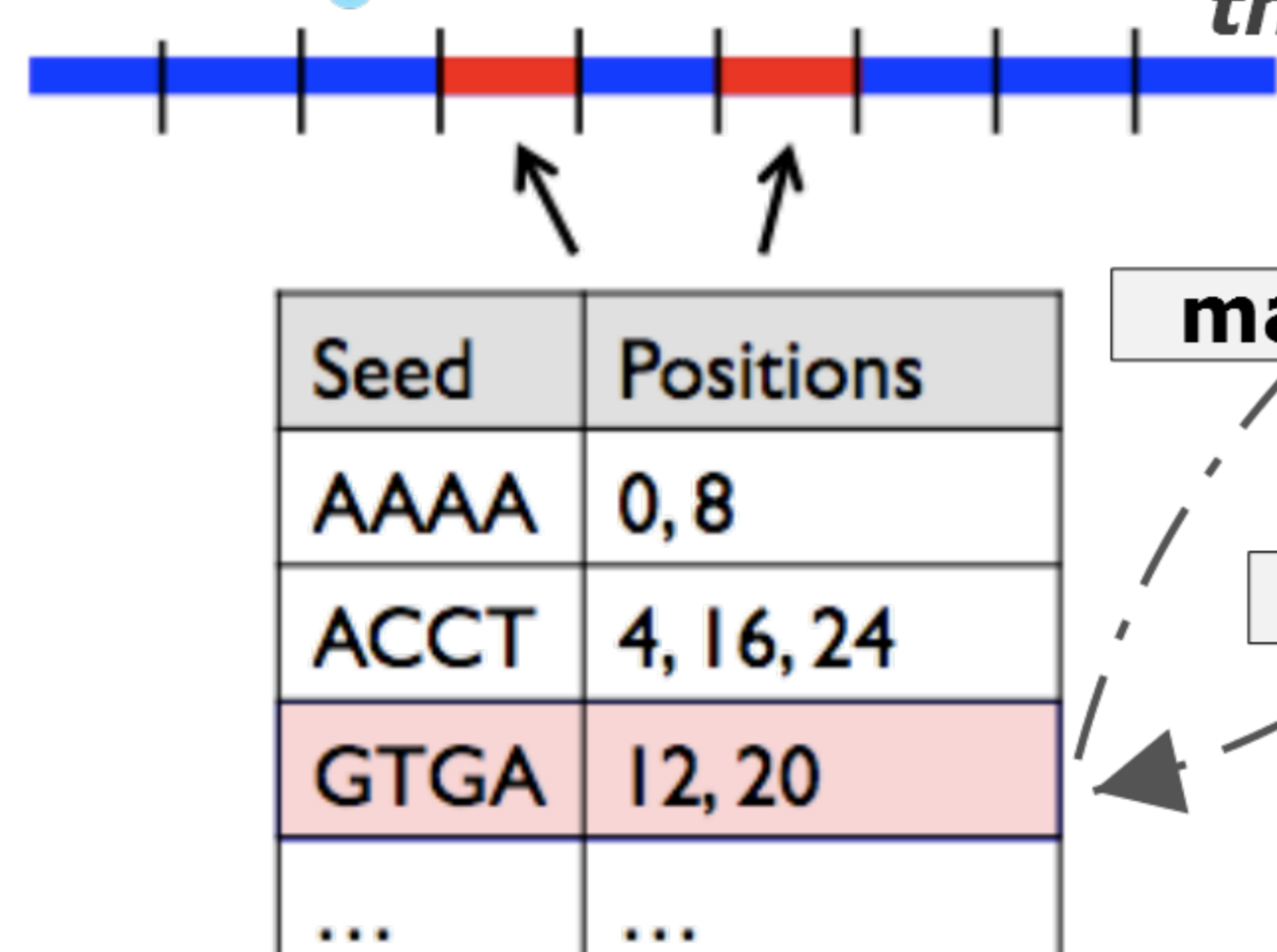
Strongly lock-based due
to shared data

Input Dataset as Shared Data

Mutex access via
spin-lock

Indexed Genome

Reference Genome
global to all
threads



match

GTGA

GTGA
AAAA
TCCG
ACTC
GTTC

W1 W2 Wn-1 Wn

GTGA Pos: 12,20
AAAA Pos: 0,8

Shared Output File

Mutex access via
spin-lock

Bowtie2:
a fast Alignment
tool for
Bioinformatics

Bowtie2

Locks
removed

Memory affinity for
private data

Tasks balanced with
memory affinity
scheduling policy

**Bowtie2 +
FastFlow +
Pinning**

Threads pinned on cores

Interleaving of shared data

**Bowtie2 +
FastFlow +
Pinning +
Interleaving**

Output file

GTGA Pos: 12,20
AAAA Pos: 0,8

Input Dataset

GTGA
AAAA
TCCG
ACTC
GTTC

Load Balancing

Incoming and
outcoming tasks
are pushed in
lock-less (CAS-free)
FIFO buffers

Affinity Scheduling Policy

Tasks are allocated in
the master to match
NUMA node of the
selected worker

W1 W2 Wn-1 Wn

Memory node 1

Memory node k

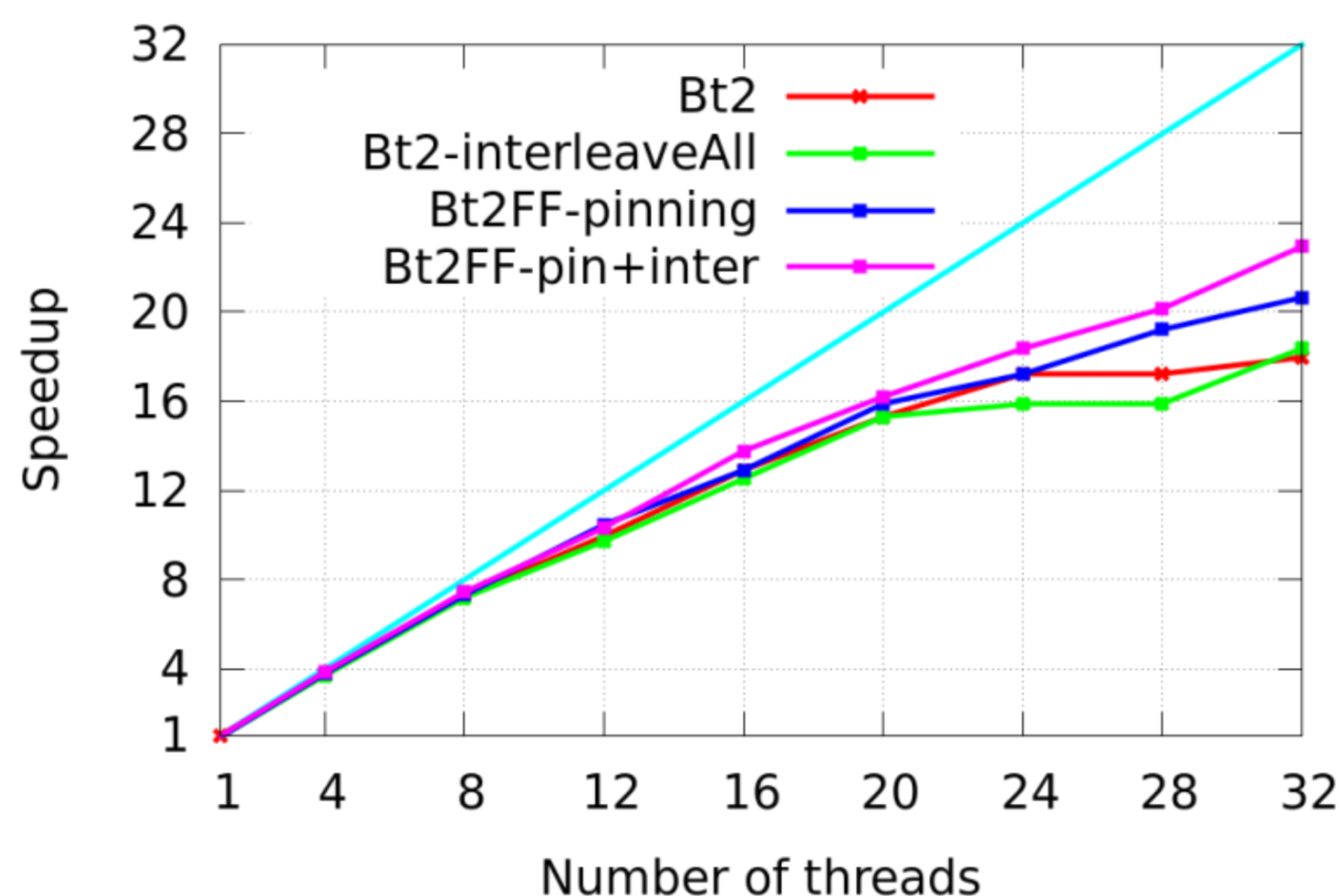
Alignment against the Genome

The Genome can be
allocated interleaved
on all NUMA nodes of
the shared memory

Pinned Threads

- Get tasks from the Master thread
- Do the alignment
- Send back results

3M reads, readlen 8 and 68 (SRR078586)



10M reads, readlen 20 (SRR072996)

