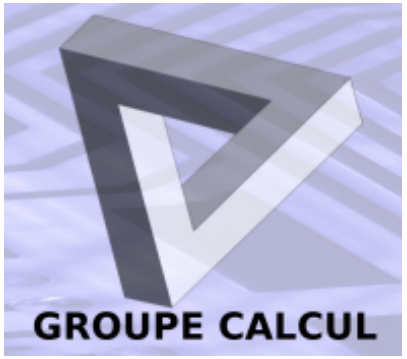
Hybrid parallel computing applied to DNA processing

Vincent Lanore

Laboratoire de Biométrie et Biologie Évolutive
Under the direction of Vincent Miele

September 6, 2011



Outline

Genomics and New Generation Sequencing

Genomics

Genomics: studying the *Genome* of organisms.
Sequencing: extracting DNA text representation from cells.

New Generation Sequencing (NGS)

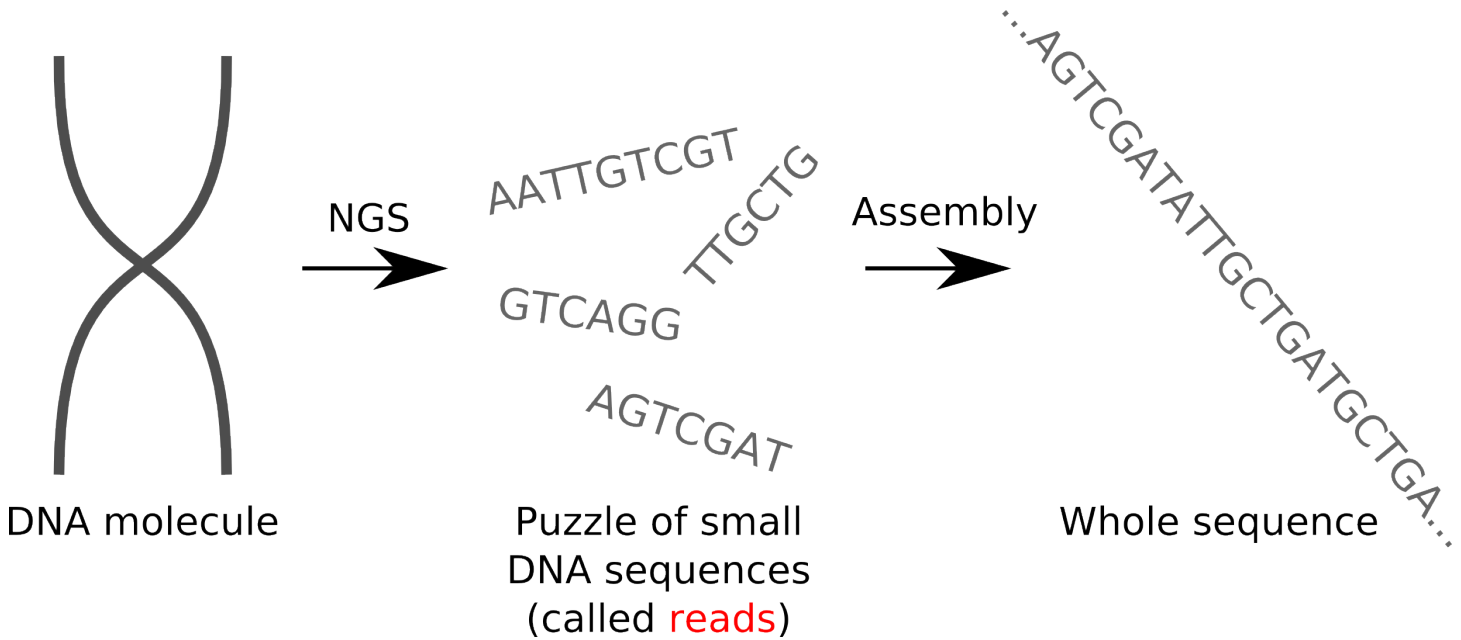
A recent breakthrough.
Faster sequencing. Produces huge amounts of raw data
(up to several gigabytes of text per use)

NGS-produced data must be processed
using text processing algorithms.
Amount of data $\Longrightarrow$ Long processing time with existing
sequential algorithms.

DNA assembly: a puzzle game

DNA sequence

A DNA sequence is a string on alphabet A, T, G, C.
A single letter from a DNA sequence is called a base.



Breaking reads into k-mers: De Bruijn graphs

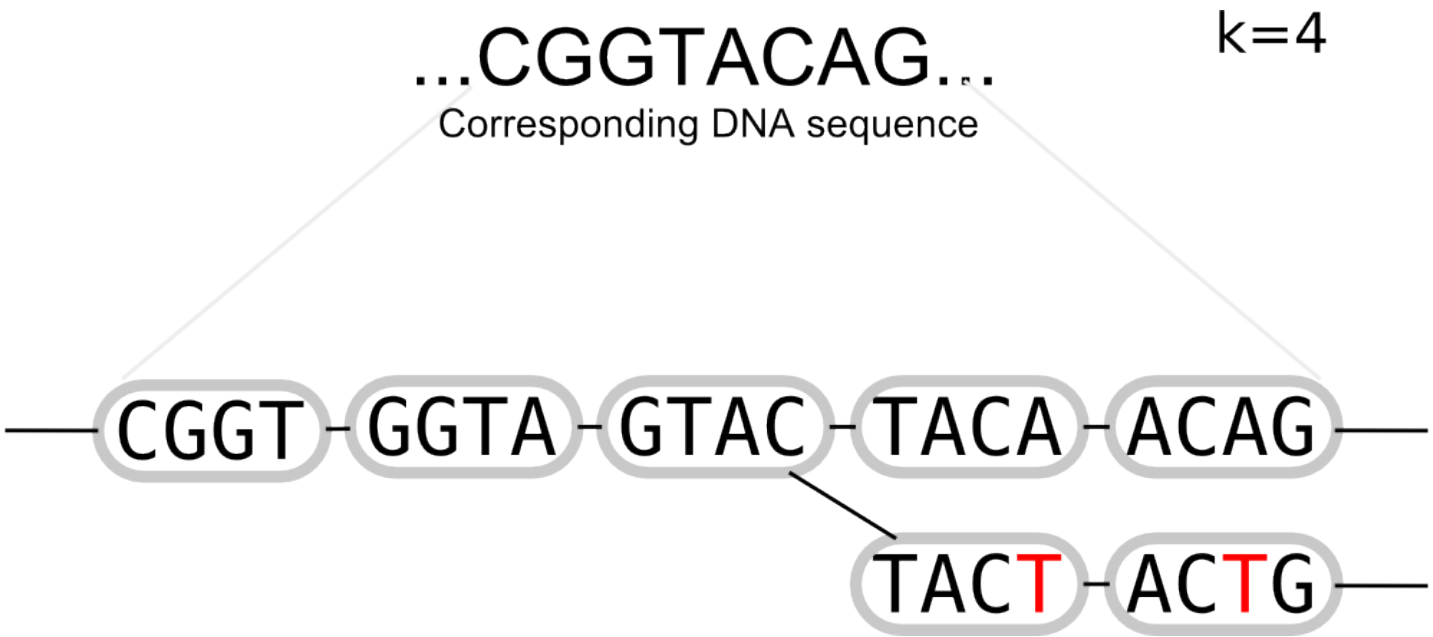
k-mers

A k-mer is a substring of length k of a read.
All the k-mers of a set of reads form its *set of k-mers*.

De Bruijn graphs

For a given k, the De Bruijn graph $\mathcal{G}$ of a set of reads $\mathcal{R}$ is a graph (V, E) such that E is the set of k-mers of $\mathcal{R}$ and such that $(u, v) \in V$ if and only if u and v overlap by $k - 1$ bases.

An example De Bruijn graph



SNPs: a genomics problem

The *genome* of an individual is made of *chromosomes*.
A chromosome $\rightarrow$ a DNA sequence
chromosomes are often *paired*.

Single Nucleotide Polymorphism (SNP)

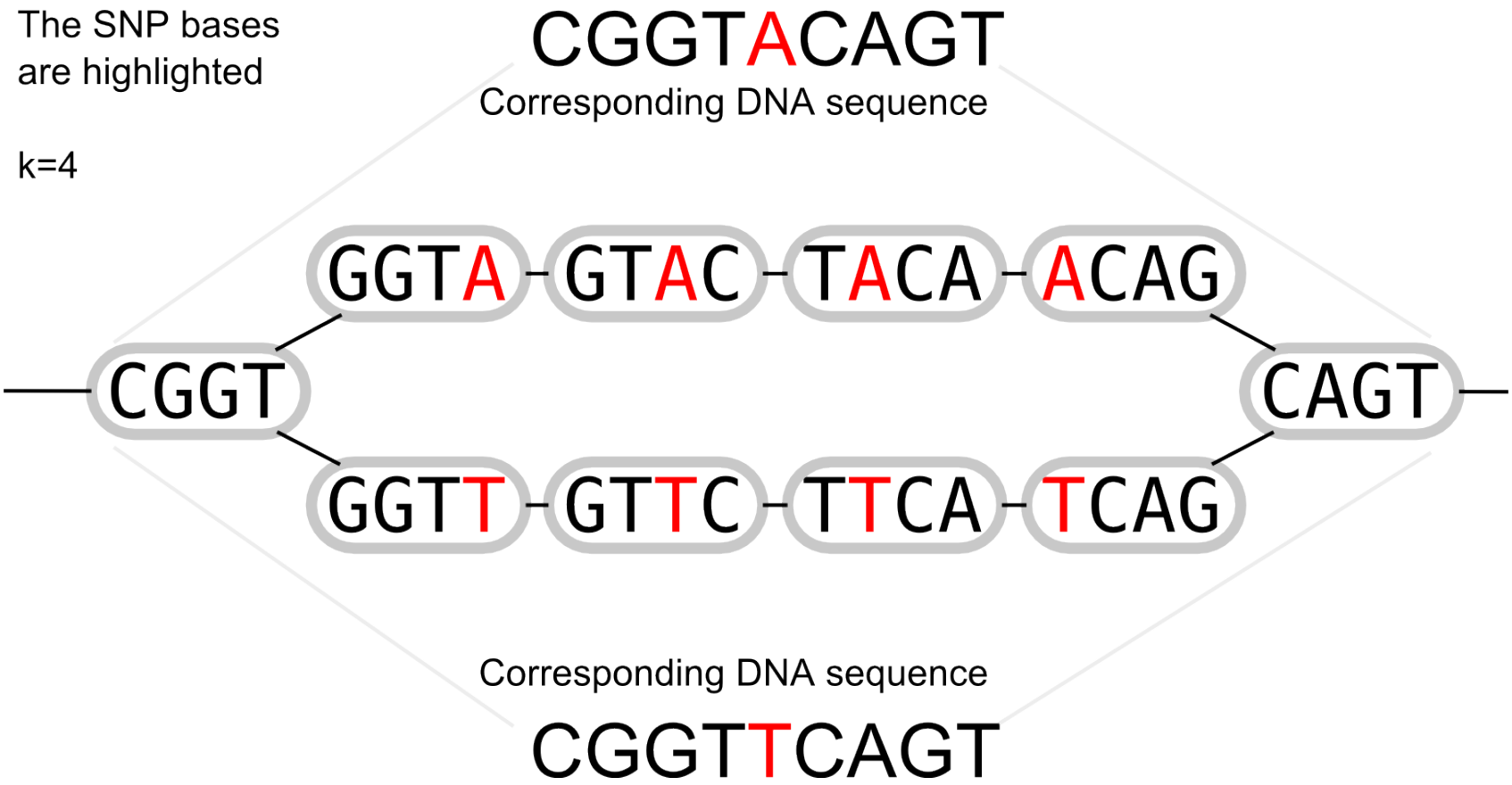
A Single Nucleotide Polymorphism or SNP is a variation of a single base (also called a *nucleotide*) between two paired chromosomes or between two individuals of the same species.

Goal of the internship
$\rightarrow$ design a SNP-finding algorithm based on De Bruijn graphs

SNPs in De Bruijn graphs

The SNP bases are highlighted

k=4



Hybrid parallel programming

Hybrid parallel programming

Mixing shared memory and message passing in a parallel algorithm.

- Message passing $\Longrightarrow$ *overhead*
- Shared memory $\Longrightarrow$ specific architecture
- Hybrid computing:
  - has low overhead
  - works with most cluster architectures
- Well-adapted to many bioinformatics clusters (several 48-cores machines, for instance)

Outline of the algorithm

To solve the SNP detection problem using De Bruijn graphs and hybrid parallel programming we have designed the following algorithm:

1. extract the k-mers from the set of reads and distribute them among the nodes
2. compute adjacency information for each k-mer
3. look for SNPs in the built De Bruijn graph

Each step will use hybrid parallelism:
$\rightarrow$ work will be divided among message-passing processes
$\rightarrow$ each process will then divide work among threads

Distribute k-mers among the nodes

The set of reads is divided among the processes.
Each process divides its reads among its threads.
Each thread executes the following algorithm:

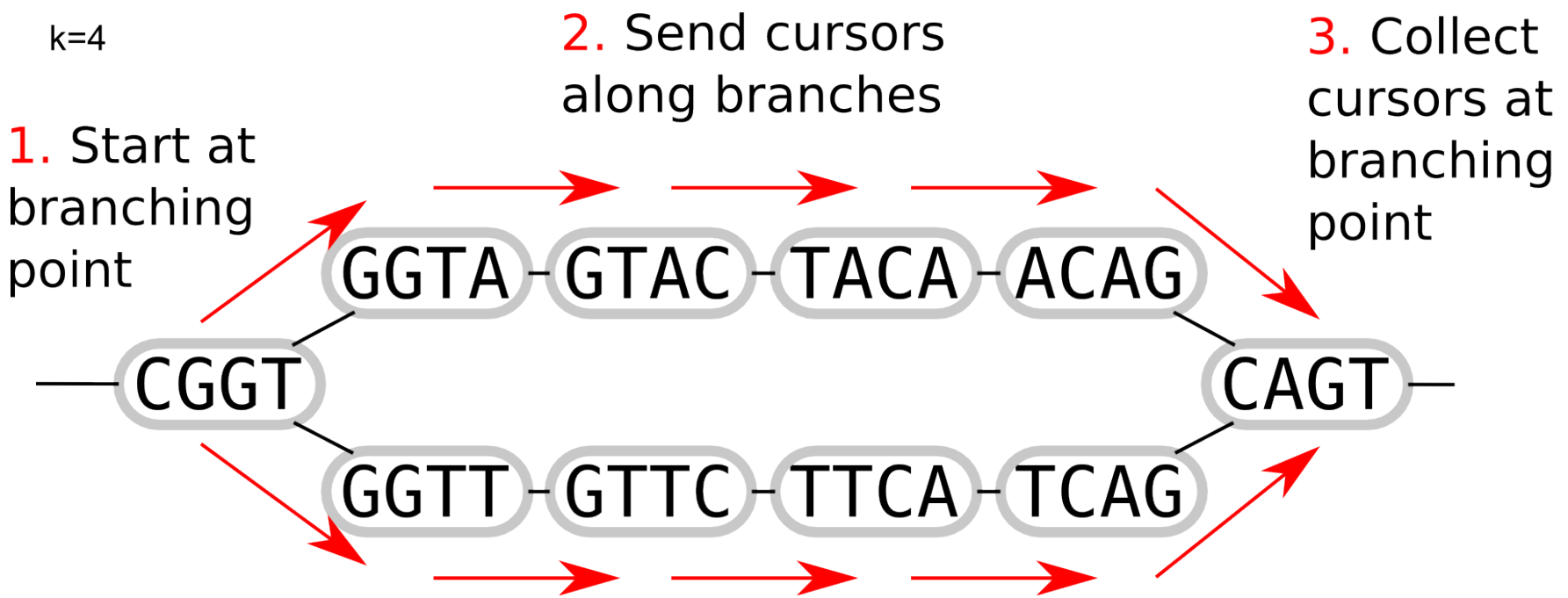
1. break each read into k-mers
2. send each k-mer to the process given by a hash function on k-mers
3. add each received k-mer to a hashtable in the shared memory of the process

Build the De Bruijn graph

The set of computed k-mers of each process is divided among its threads.
Each threads executes the following algorithm:

1. send each k-mer to the owners of its 8 possible neighbors; a property of the hash function ensures those neighbors will belong to a maximum of 2 different processes
2. for each received k-mer
   - if a neighbor u is present in the hashtable, update adjacency information for u
   - if a *branching point* is detected while updating adjacency information, add it to a branching point hashtable in shared memory

Find the SNPs



3'. Discard cursors that have done too many steps

Implementation using MPI and OpenMP

Implementation was done using:

- *C++*
- *MPI* for the message-passing part
- *OpenMP* for the shared memory part

The program was called HyBu.
First two steps of the algorithm: implemented and tested
Last step of the algorithm: work in progress

Benchmarks using the LBBE's cluster

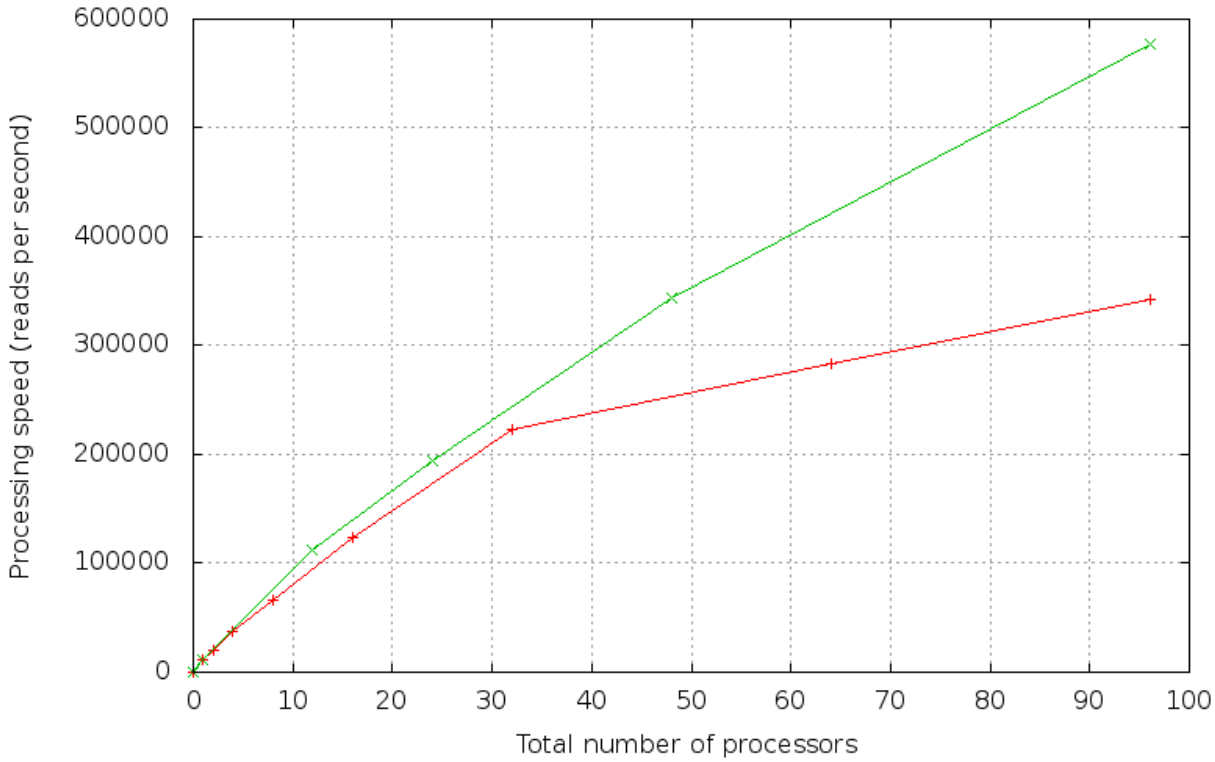


Figure: De Bruijn graph building on two 48-cores nodes.
Red curve: one process per node.
Green curve: 6 processes par node.

Conclusions

- successful design and implementation of an hybrid SNP detection algorithm
- tests on the LBBE's cluster show good scalability up to 96 processors
- gains from hybrid programming not obvious

Thank you for your attention.