

Approximate Pattern Matching

Paolo Calcagni

Lino Moises
Mediavilla Ponce

Parallelization Strategies

The background of the slide is a dark blue gradient. It is decorated with an abstract pattern of geometric elements: small squares in teal, pink, and orange, and thin white vertical lines of varying lengths. These elements are scattered across the slide, creating a modern, tech-oriented aesthetic.

Patterns Over Ranks

1

Levels of parallelization

Master distributes a pattern to each

1. MPI_RANK

2. OPEN_MP

Within each rank, threads collaborate

If there is one, each rank uses it side by side with

3. GPU



Database Over Ranks

2

Levels of parallelization

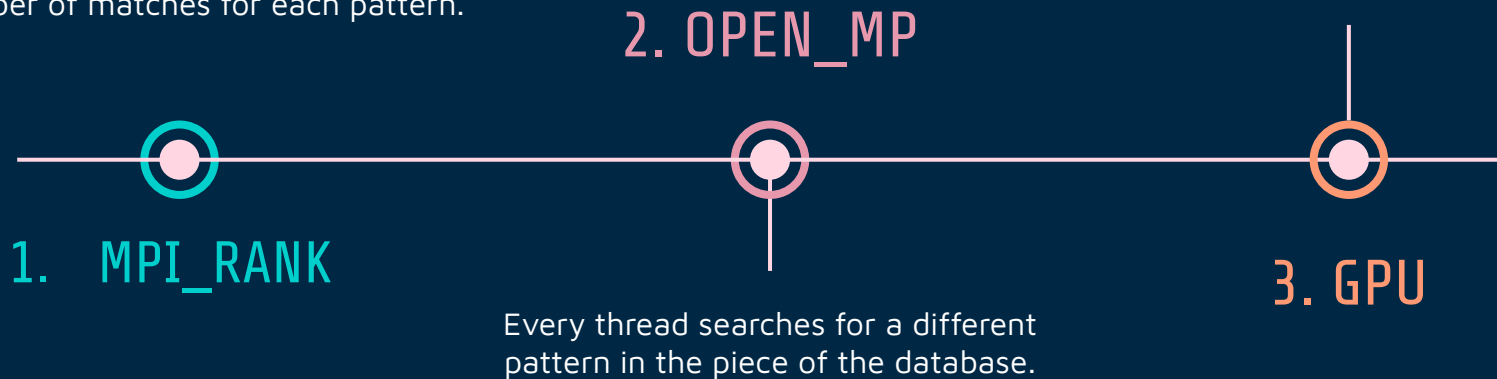
Rank 0

Assigns a piece of database to each rank.
For every pattern waits the answers of ranks.

Other Ranks

Reads the piece of database and calculates the number of matches for each pattern.

If a GPU exists and the number of patterns is greater than one, the GPU will be used to further parallelize the searching of different patterns.



Cost Model

The background of the slide is a dark blue field decorated with an abstract pattern of geometric elements. It includes numerous small squares in white, light blue, and light orange, some of which are solid while others are hollow. Thin, light-colored vertical lines of varying lengths are scattered across the slide, creating a modern, minimalist aesthetic.

1 Pattern

The background of the slide is a dark blue field decorated with an abstract pattern of geometric elements. It includes numerous small squares in solid colors (pink, teal, orange) and as thin white outlines. Additionally, there are several thin, vertical white lines of varying lengths scattered across the composition, creating a modern, minimalist aesthetic.

Advantages & Disadvantages

Patterns Over Ranks

- - Only 1 MPI Rank is exploited
- All Open MP threads exploited

Database Over Ranks

- Only 1 Open MP thread is exploited
- All MPI Ranks exploited

Thread loss concept

$$(Active\ MPI\ Ranks - 1) * OMP_Threads$$

$$Active\ MPI\ Ranks * (OMP_Threads - 1)$$

The background is a dark blue field decorated with a sparse, abstract pattern. It features several thin, vertical white lines of varying lengths. Scattered throughout are small squares in three colors: pink, teal, and orange. Some squares are solid, while others are outlined in white. The overall aesthetic is minimalist and modern.

Multiple Patterns

Patterns over ranks

If the number of patterns is equal to the actual MPI ranks,
there is no round-robin.

All the hardware is being used. Every MPI rank has a pattern.
All the ranks will finish at the same time, after one iteration.

If the number of patterns is greater than the actual,
MPI ranks round robin takes place.

If the execution time of an iteration of round-robin is a divider
of the time slice, all the hardware is being used.
Otherwise, the use of the hardware is not optimized.

If the number of patterns is less than the actual MPI ranks,
some MPI ranks are not used.
The use of the hardware is not optimized.

Database over ranks

If the number of patterns is equal to the number of threads,
there is no round-robin.
All the hardware is being used.

If the number of patterns is greater than the number of
threads, there is round-robin.

If the execution time of an iteration is a divider of the time
slice, all the hardware is being used.
Otherwise, the use of hardware is not optimized.

If the number of patterns is smaller than the threads,
some threads are not used.
The use of the hardware is not optimized.

```
dimensionOfIterationPatternsOverRanks = ActiveMPIRanks/patterns  
dimensionOfIterationDatabaseOverRanks = ThreadsPerRank/patterns
```

□

To calculate

HardwareOptimizationPatternsOverRanks and
HardwareOptimizationDatabaseOverPatterns:

```
while(x < 1){
```

```
  x = x * 2
```

```
}
```

```
ratioHardwareOptimizationApproachChosen = x - 1;
```

□

```
if(ratioPatterns == 0 && ratioDatabase == 0){
```

```
  // Both of the approaches use the hardware at its maximum  
  // capacity. We could choose the approach randomly or in a  
  // predetermined way (as we did in the code).
```

```
} else{
```

```
  // Otherwise, we calculate the minimum between ratioPattern and  
  // ratioDatabase and we choose the approach which optimizes better  
  // the use of the hardware.
```

```
}
```

The background is a dark blue field decorated with various geometric elements. It includes several thin white vertical lines of varying lengths. Scattered throughout are small squares in three colors: light blue, pink, and orange. Some of these squares are solid, while others are hollow outlines. The overall aesthetic is modern and minimalist.

Experiment Results

Example 1. Let's assume we have 4 MPIranks, 3 threads, 6 patterns and a database with 100 characters.

$$\textit{dimensionOfIterationPatterns} = \textit{ActiveMPIRanks}/\textit{patterns} = 4/6 = 0.6$$

$$\textit{dimensionOfIterationDatabase} = \textit{ThreadsPerRank}/\textit{patterns} = 3/6 = 0.5$$

$$\textit{ratioPatterns} = 0.2$$

$$\textit{ratioDatabase} = 0$$

Execution time of Patterns over Ranks: 54 seconds

Execution time of Database over Ranks: 41 seconds

Example 2. Let's assume we have 4 MPIranks, 5 threads, 6 patterns and a database with 100 characters.

$$\textit{dimensionOfIteratioPatterns} = \textit{ActiveMPIRanks}/\textit{patterns} = 4/6 = 0.6$$

$$\textit{dimensionOfIterationDatabase} = \textit{ThreadsPerRank}/\textit{patterns} = 5/6 = 0.8$$

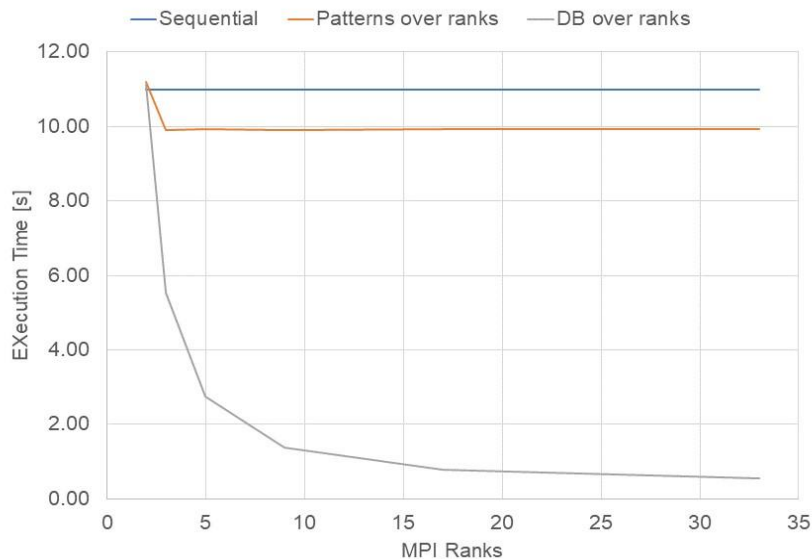
$$\textit{ratioPatterns} = 0.2$$

$$\textit{ratioDatabase} = 0.6$$

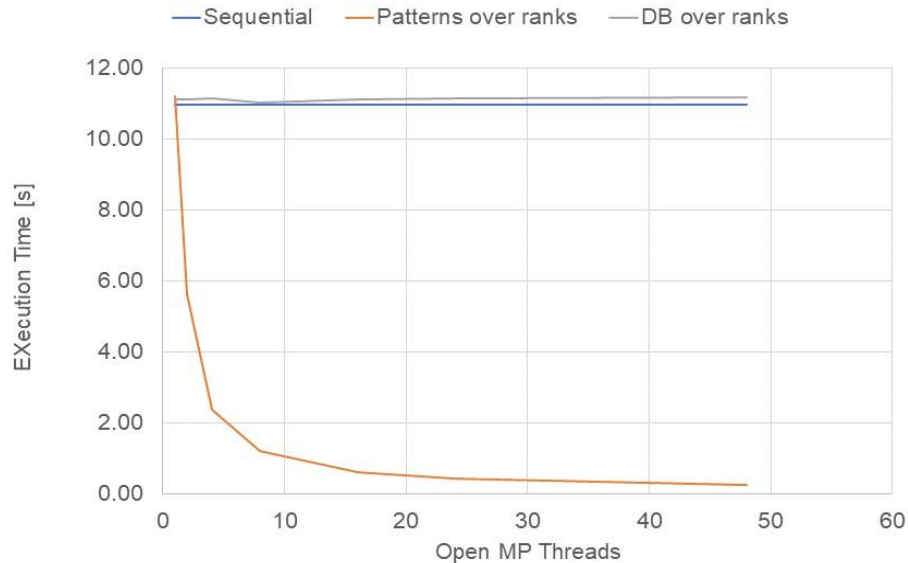
Execution time of Patterns over Ranks: 30.42 seconds

Execution time of Database over Ranks: 43.24 seconds

Strategies scaling: 1 pattern

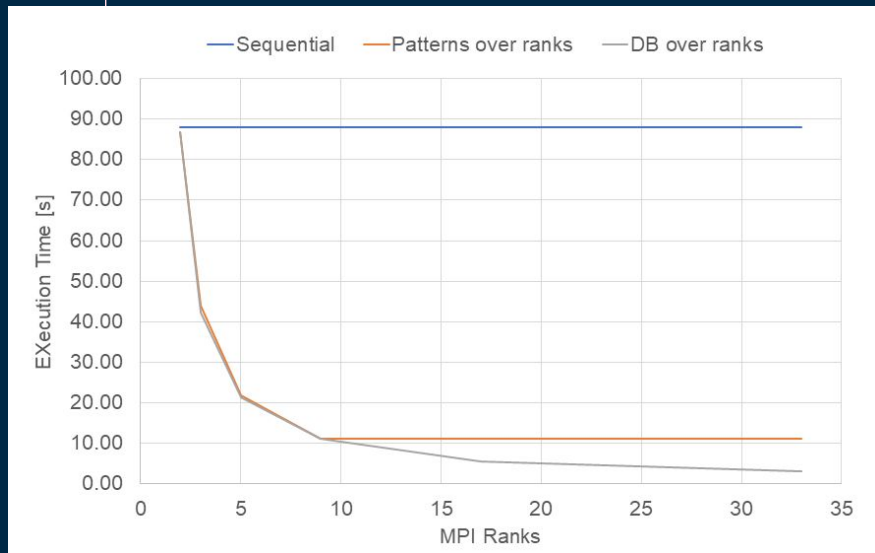


Database over ranks:
scales with ranks

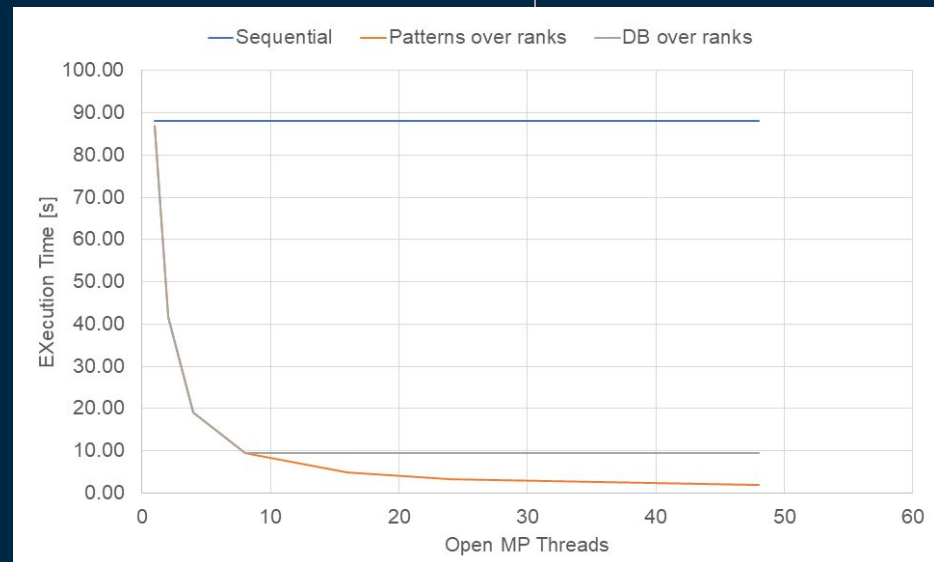


Patterns over ranks:
scales with threads

Strategies scaling: multiple patterns



Database over ranks:
scales with ranks



Patterns over ranks:
scales with threads

Notes

The background features a dark blue field with a pattern of small squares in teal, pink, and orange, and thin white vertical lines of varying heights.

Notes

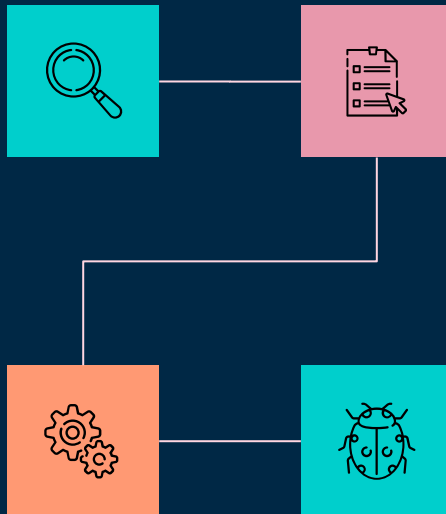
Performance Reading Database

Timer after all the ranks
received patterns and buffer.

Bug OpenMP Threads

We don't see the results expected.
We added a special flag
`TESTPERFORMANCENOLEVENSHTEIN`

Problem with sleep in CUDA



Assumptions

Program executed by 2 Ranks
No Multiple GPUs

Bug Database Over Ranks

Sometimes happens that the approach
`DatabaseOverRanks` see more matches
than the real ones.

The background is a dark blue gradient. It is decorated with various geometric elements: thin white vertical lines of varying lengths, small squares in teal, orange, and pink, and larger squares in teal and orange. These elements are scattered across the slide, creating a modern, minimalist aesthetic.

Thank you
for your attention