FACULTY OF ENGINEERING
AND ARCHITECTURE

FLEXIBLE MATRIX MULTIPLICATION KERNELS ON GPUS

Thomas Faingnaert
1 July 2020

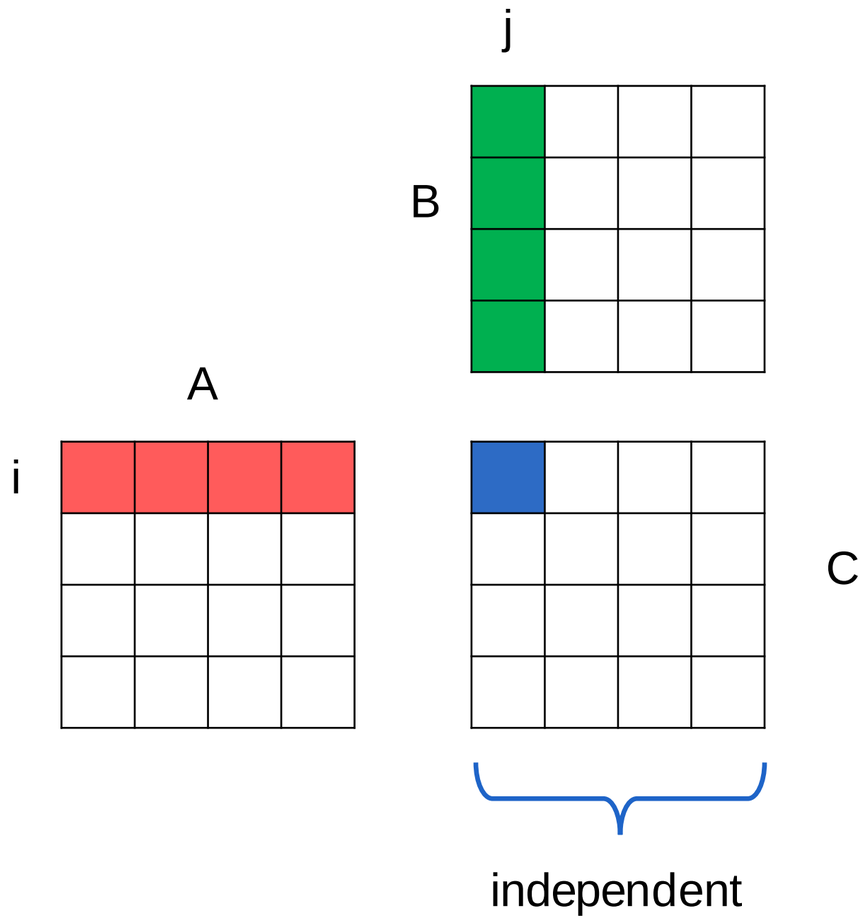
Supervisor: Prof. Dr. Ir. Bjorn De Sutter
Counsellor: Dr. Tim Besard

GHENT
UNIVERSITY

GENERAL MATRIX MULTIPLICATION (GEMM)

C := A * B + C

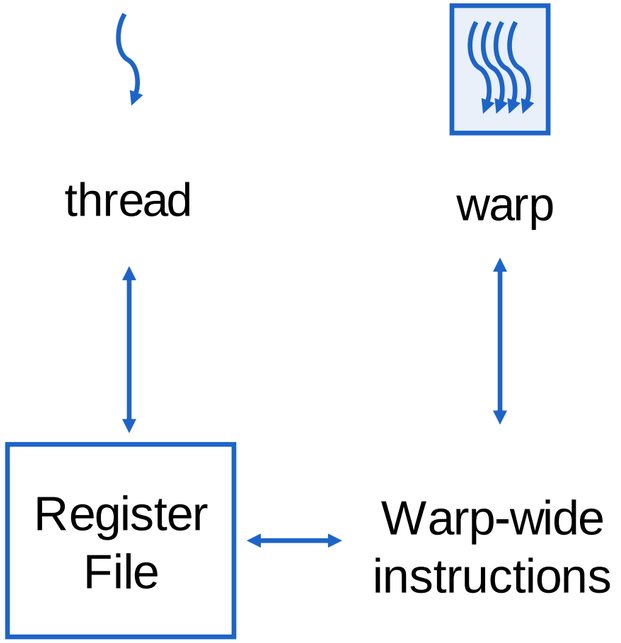
```
for i = 1 : N
  for j = 1 : N
    for k = 1 : N
      C[i, j] += A[i, k] * B[k, j]
    end
  end
end
```

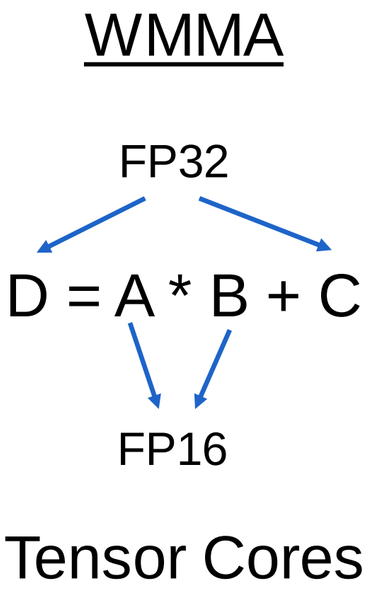


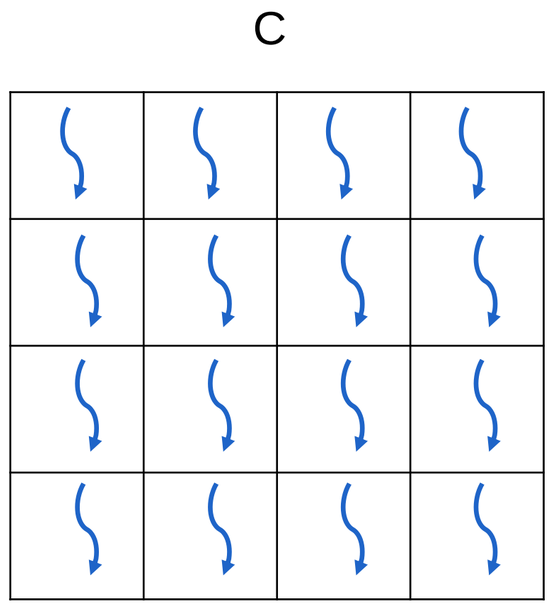
GHENT UNIVERSITY

GENERAL MATRIX MULTIPLICATION ON GPUS

Massively parallel processing



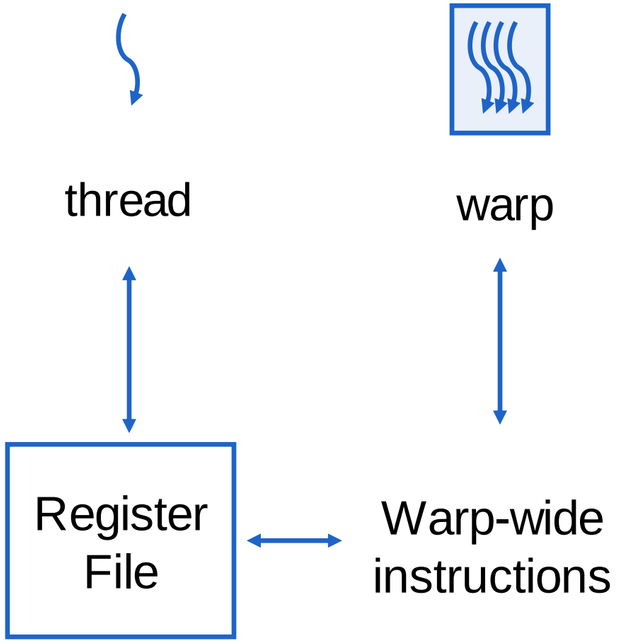




GHENT UNIVERSITY

GENERAL MATRIX MULTIPLICATION ON GPUS

Massively parallel processing



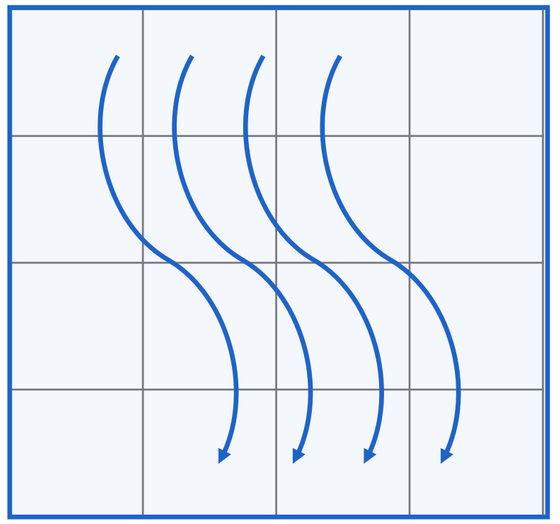
WMMA

FP32

D = A * B + C

FP16

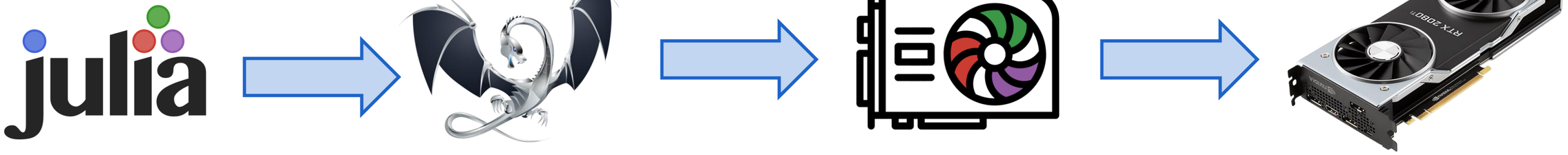
Tensor Cores



Tensor Cores (typically 16 x 16)

GHENT UNIVERSITY

TOOL FLOW



Codegen changes

WMMA API

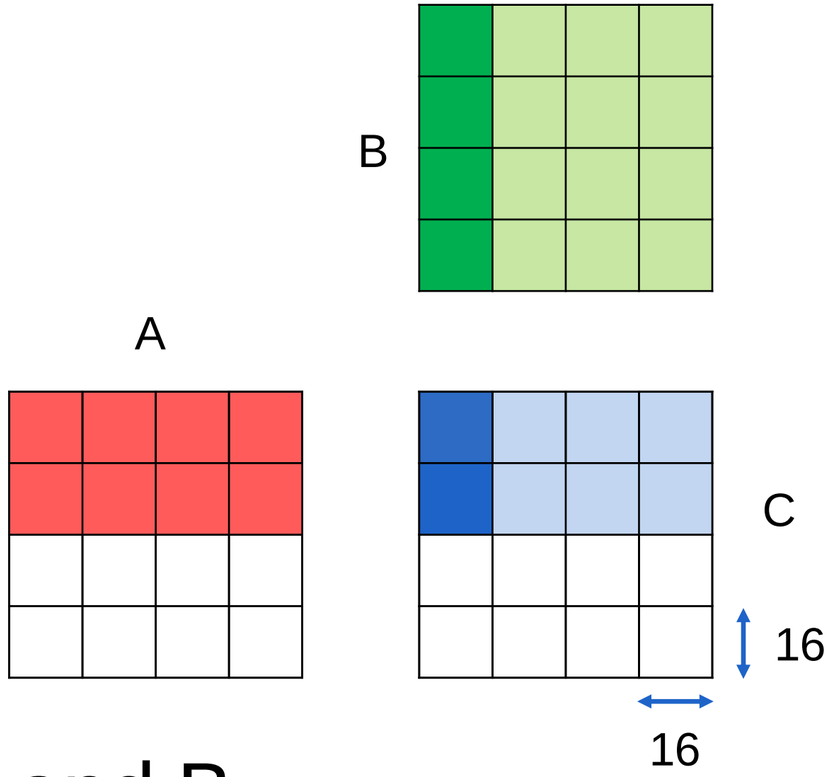
Tensor Cores

GHENT UNIVERSITY

GENERAL MATRIX MULTIPLICATION ON GPUS

C := A * B + C

```
for i = 1 : 16 : N
  for j = 1 : 16 : N
    for k = 1 : 16 : N
      # calculate a 16 x 16 matrix mult.
    end
  end
end
```

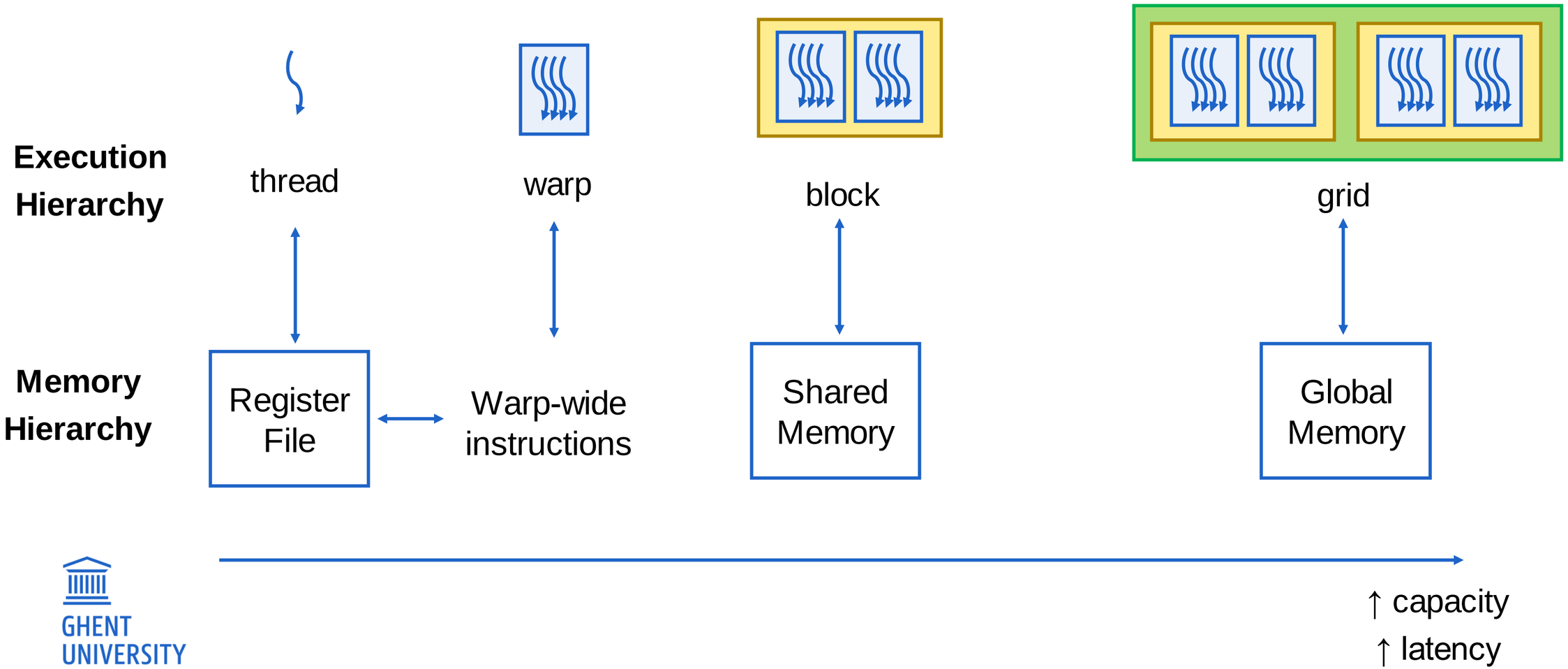


Inefficient: large working set to store A and B

→ Exploit data reuse: temporal locality

GENERAL MATRIX MULTIPLICATION ON GPUS

Massively parallel processing

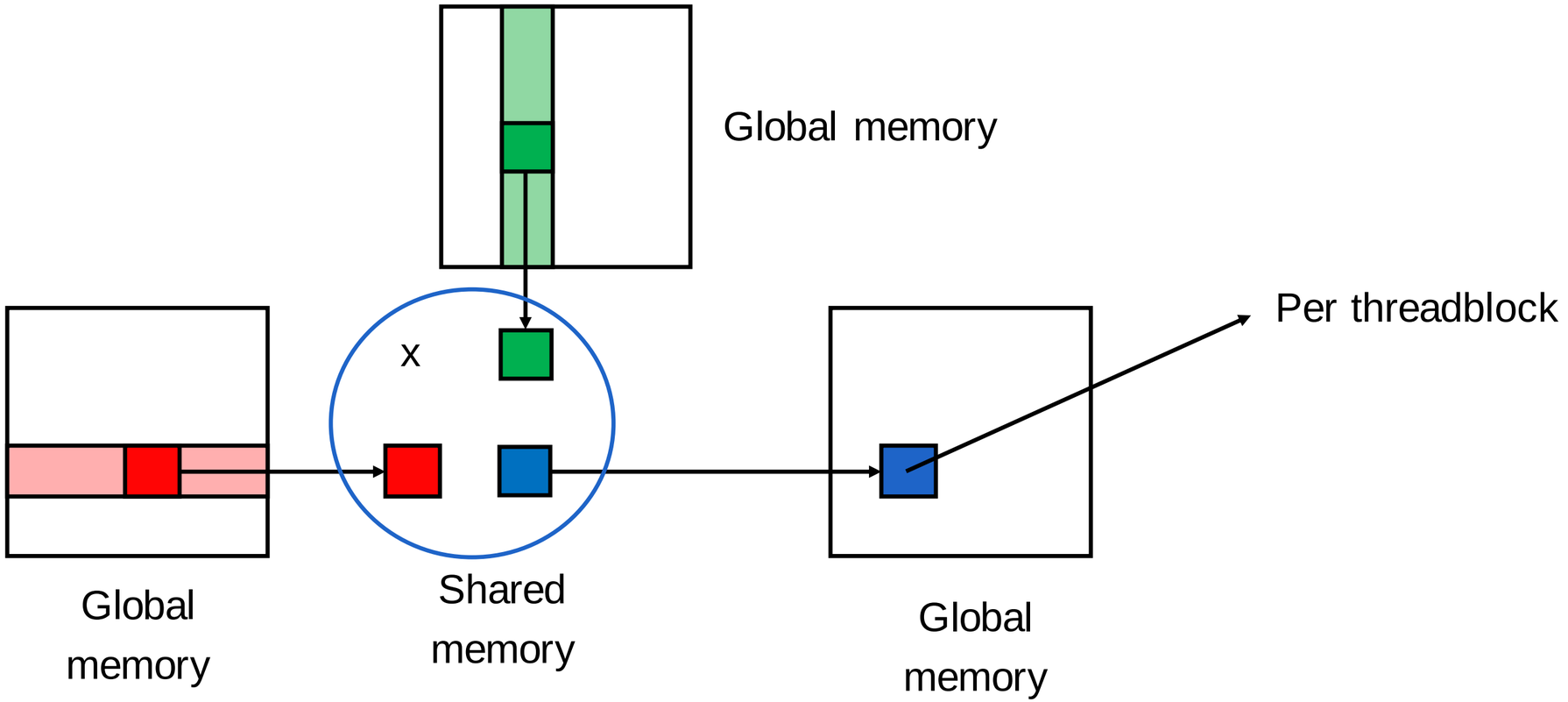


GHENT UNIVERSITY

DESIGN OF PERFORMANT GEMM ON GPUS

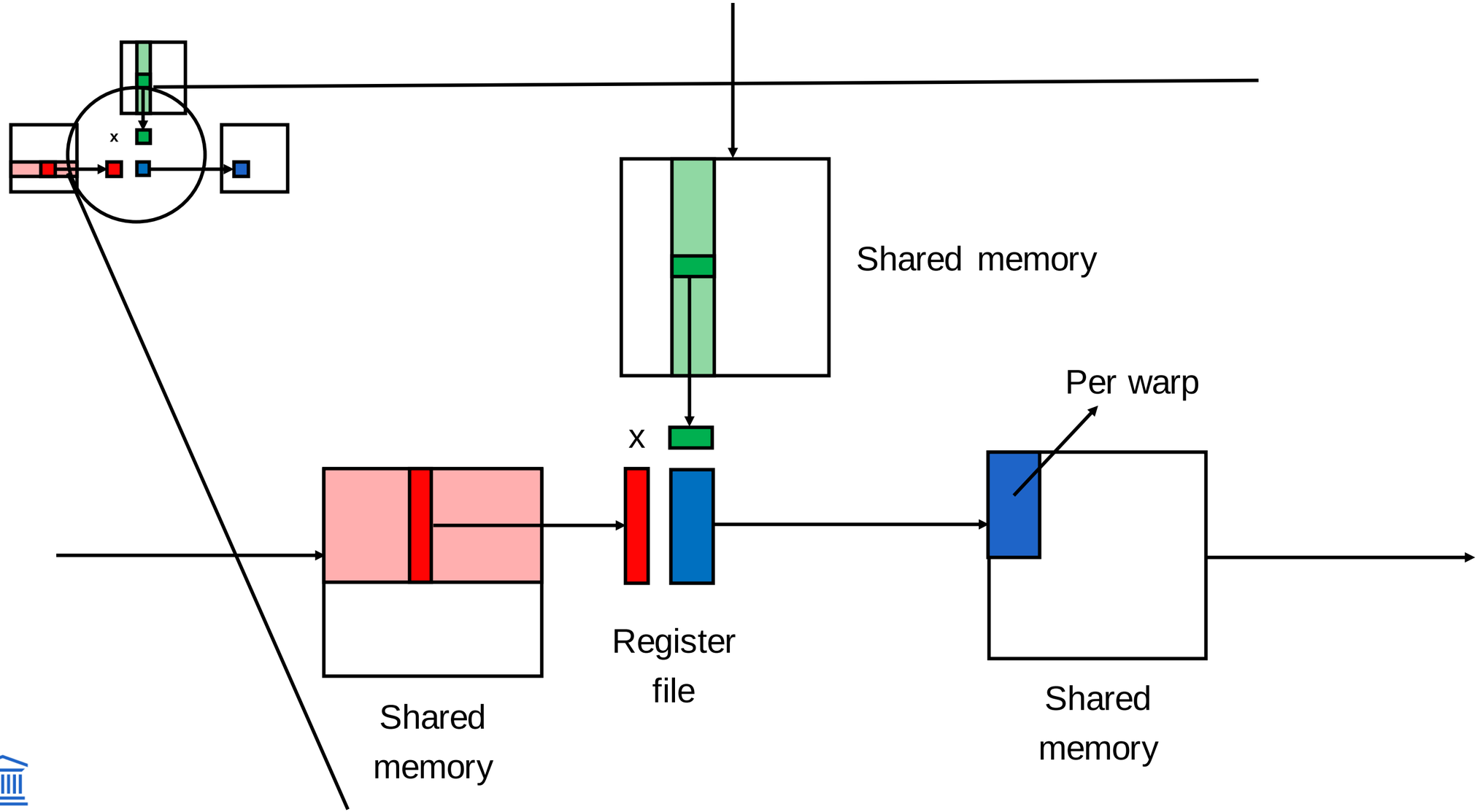
Small N: perfect temporal locality

Large N: explicit blocking techniques



GHENT UNIVERSITY

DESIGN OF PERFORMANT GEMM ON GPUS



GHENT UNIVERSITY

GENERAL MATRIX MULTIPLICATION

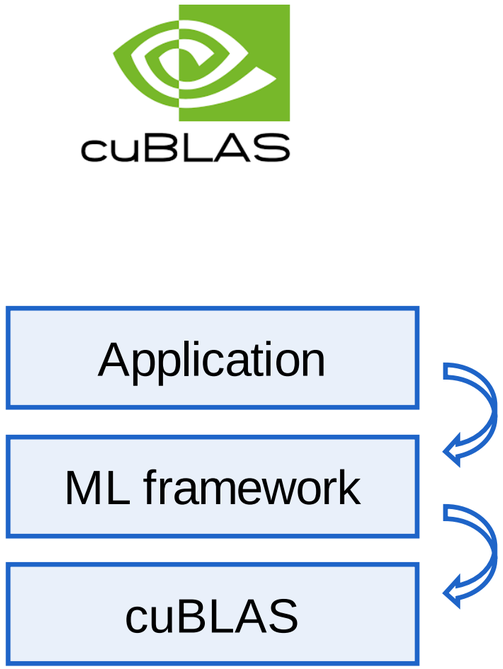
C := A * B + C

```
for block_tile_i = ...
   ⋱
     for k = ...
         C[i, j] += A[i, k] * B[k, j]
     end
   ⋰
end
```

cuBLAS
cuDNN

GHENT UNIVERSITY

THE NEED FOR FLEXIBILITY



Problem:

- Only limited set of kernels

GHENT UNIVERSITY

THE NEED FOR FLEXIBILITY

Goal: framework for flexible GEMM

Which flexibility is needed?

- Literature study
- CUTLASS

GHENT UNIVERSITY

THE NEED FOR FLEXIBILITY

C := A * B + C

```
for block_tile_i = ...
   ⋱
     for k = ...
          C[i, j] += A[i, k] * B[k, j]
     end
   ⋰
end
```

Memory layout?

- N vs. T
- NCHW vs. NHWC
- Tensors

GHENT UNIVERSITY

THE NEED FOR FLEXIBILITY

C := A * B + C

```
for block_tile_i = ...
   ⋱
     for k = ...
         C[i, j] += A[i, k] * B[k, j]
     end
   ⋰
end
```

Data types?

- FP16, FP32, FP64
- Mixed precision
- Complex numbers
- User-defined type

GHENT UNIVERSITY

THE NEED FOR FLEXIBILITY

C := f(A * B + C)

```
for block_tile_i = ...
    ⋱
      for k = ...
          C[i, j] += f(A[i, k] * B[k, j])
      end
    ⋰
end
```

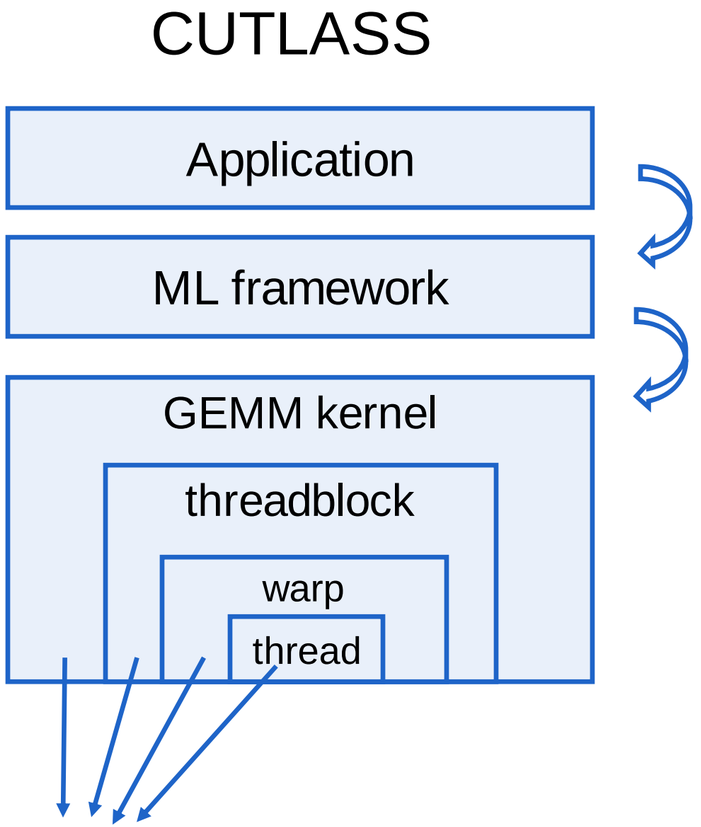
Elementwise operations?

– Scaling

– Activation function

GHENT UNIVERSITY

FLEXIBLE GEMM APIS

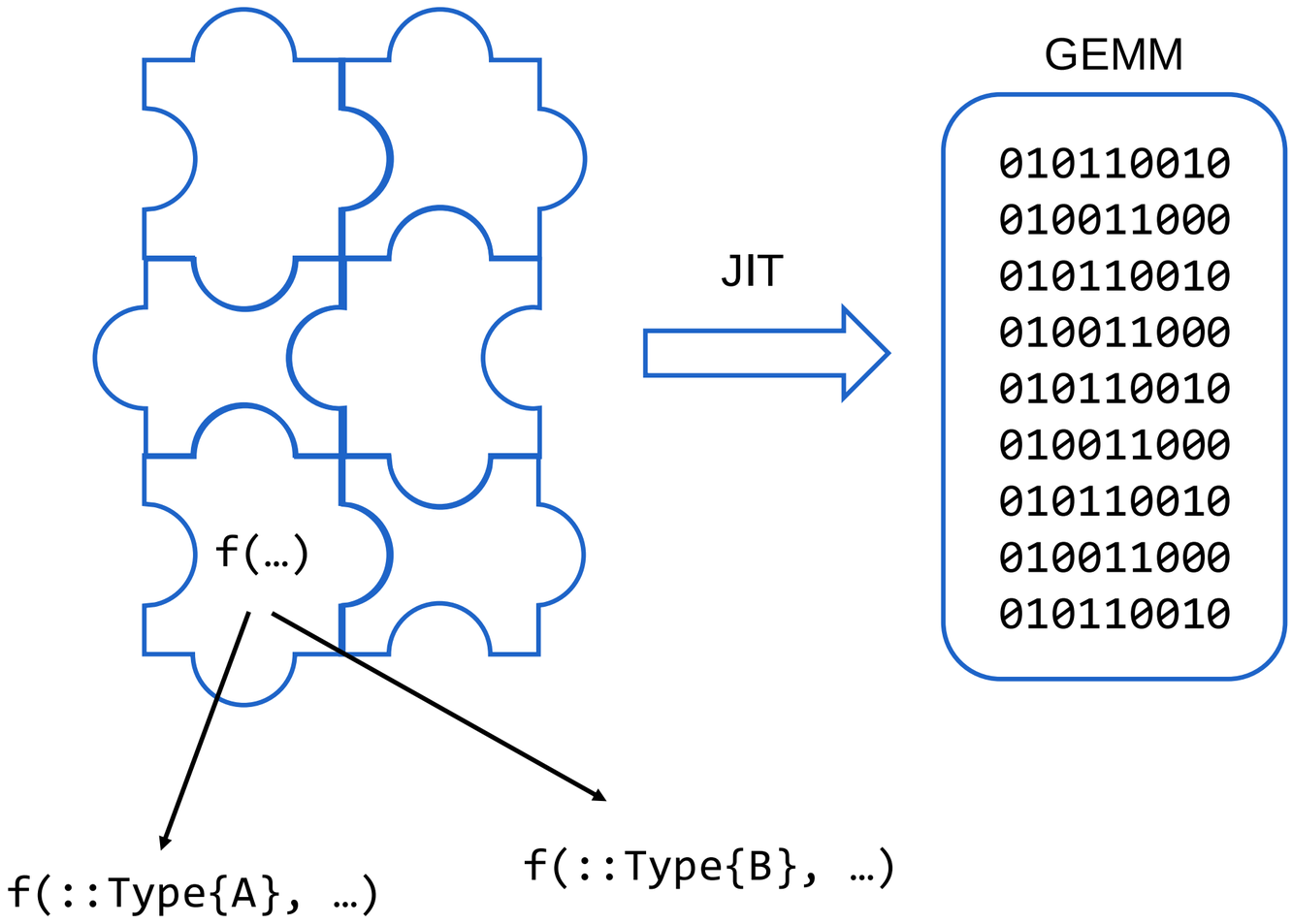


Templates specialised on element datatype, memory layout, …

- Large number of components
- Low-level C++

GHENT UNIVERSITY

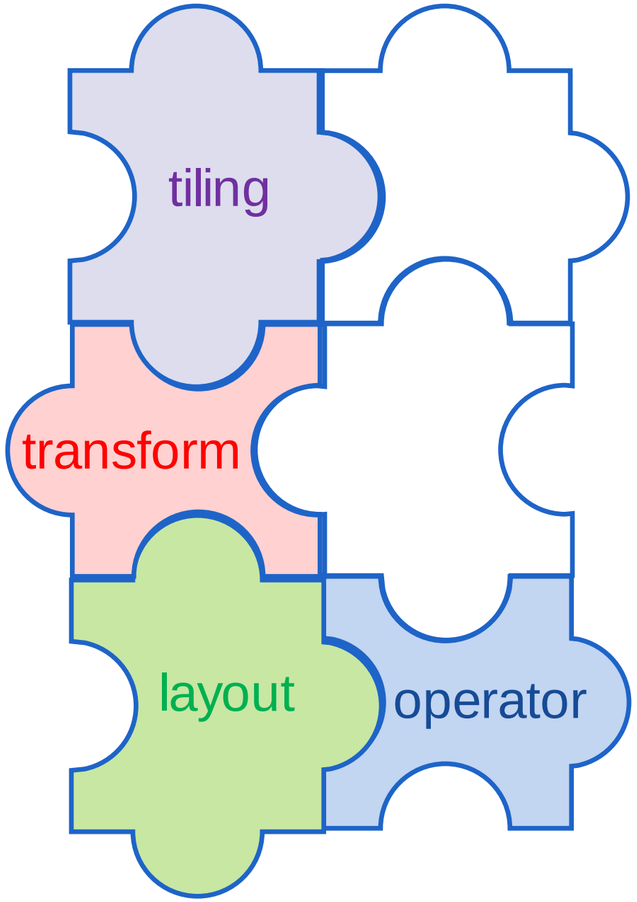
FLEXIBLE GEMM APIS IN JULIA



GHENT UNIVERSITY

FLEXIBLE GEMM APIS IN JULIA

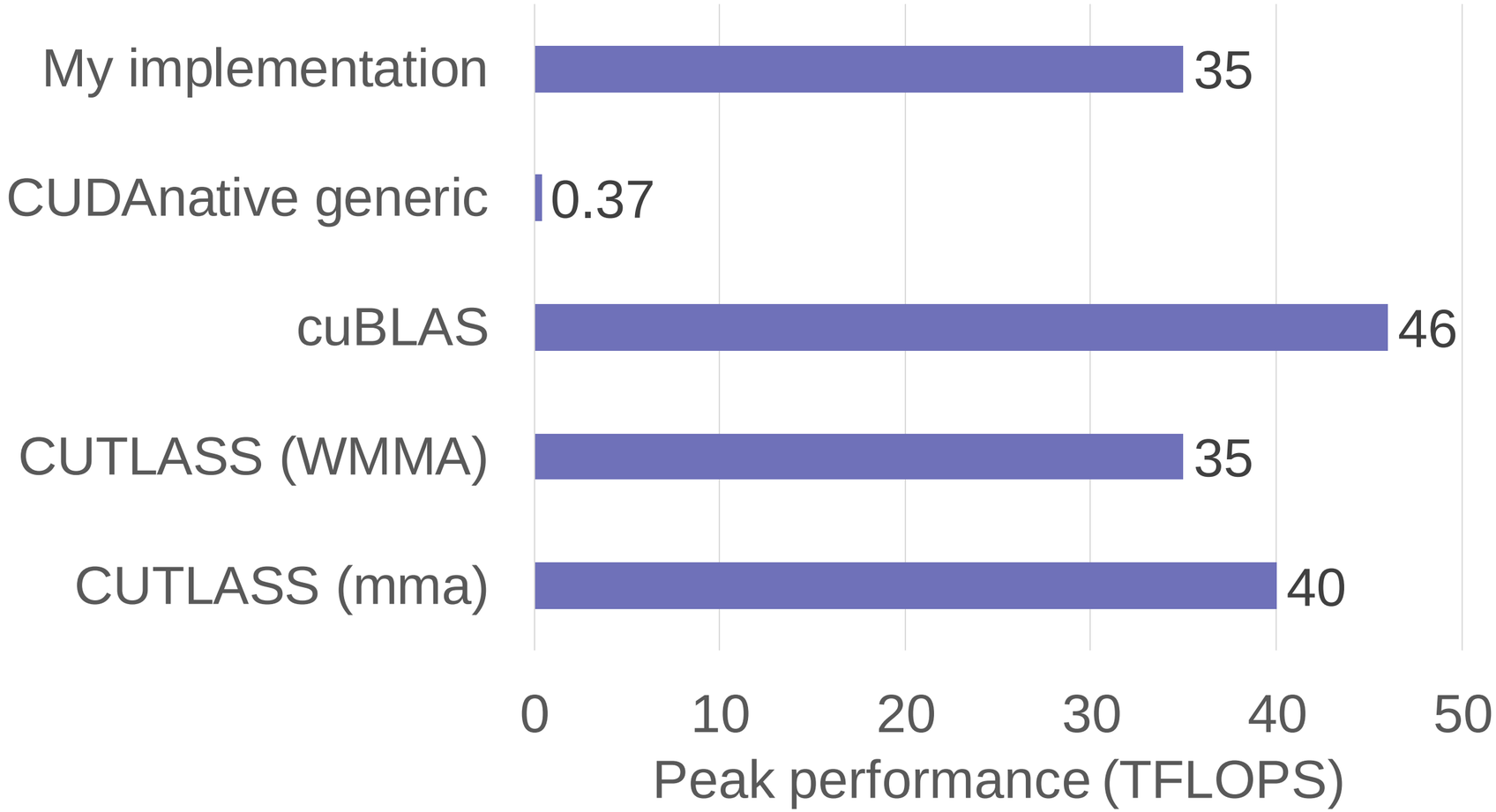
```
for block_tile_i = ...
    ⋱
      for k = ...
          C[i, j] += f(A[i, k] * B[k, j])
      end
    ⋰
end
```



GHENT UNIVERSITY

EVALUATION

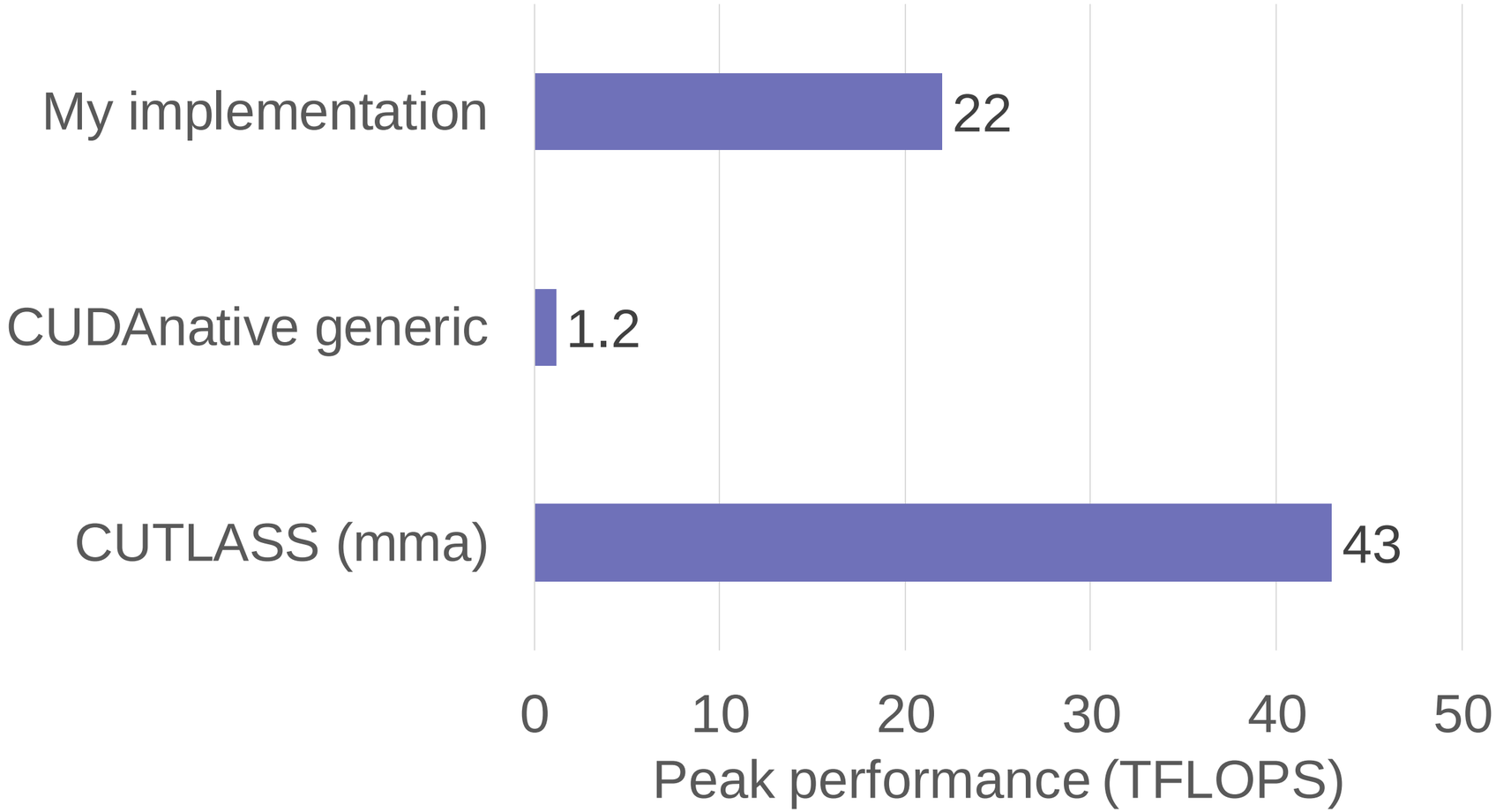
1. Mixed-precision GEMM (WMMA)



GHENT UNIVERSITY

EVALUATION

2. Complex mixed-precision GEMM (WMMA)



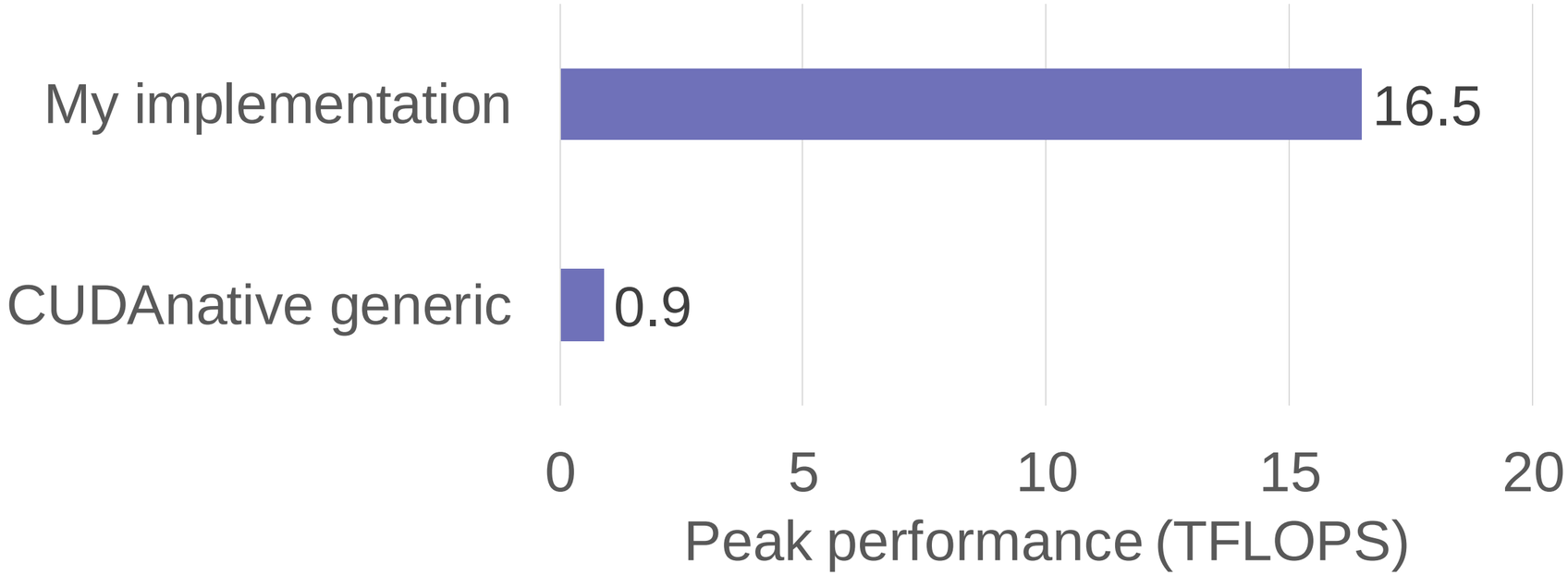
GHENT UNIVERSITY

EVALUATION

3. Dual mixed-precision GEMM (WMMA)

   $(a + b\varepsilon) * (c + d\varepsilon) = ac + (ad + bc)\varepsilon$

   Application: automatic differentiation



GHENT UNIVERSITY

QUESTIONS?

GHENT
UNIVERSITY