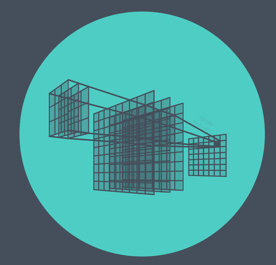
www.lift-project.org

Towards Mapping Lift To Deep Neural Networks

Naums Mogers
Aaron Smith, Dimitrios Vytiniotis
Michel Steuwer, Christophe Dubach
Ryota Tomioka

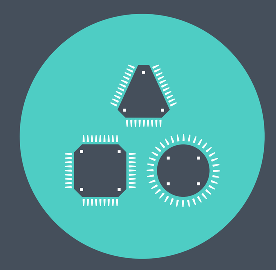
1 Context

- GEMM is ubiquitous in Deep Neural Networks (DNNs)
- It is the basis of both stencil and im2col convolution methods



- Hardware accelerators use N-dimensional computational units
- These units are exposed in ISAs via coarse-grained operators:

```
VVAdd32, VVAdd64, MVAdd64, MVAdd129
VVMul64, VVMul128, MVMul64, MVMul128
```



2 The problem

- How can we combine device-specific operators optimally?
- How can we make the optimisations performance portable?
- How can we automate and abstract the process from the user?

3 The Lift approach

3.1 Concept

1. Separate algorithm (WHAT) from implementation (HOW)
2. Detect and rewrite patterns

GEMM^imperative

```
for (int i = 0; i < M; i++) {
  for (int j = 0; j < N; j++) {
    for (int k = 0; k < K; k++) {
      temp[k + K*N*i + K*j] =
        A[k + K*i] * B[k + K*j];
    }
    for (int k = 0; k < K; k++) {
      C[j + N*i] +=
        temp[k + K*N*i + K*j];
    }}}
```

→ GEMM^functional

```
A >> Map(Arow =>
  B >> Map(Bcol =>
    Zip(Arow, Brow) >>
    Map(ScalarMul) >>
    Reduce(0, Add)
  )
)
```

→ GEMM^optimised

```
MMul(A, B)
```

3.2 Functional data-parallel IR Language

- Data types

```
Int, Arrays
Float8 / Float16 / Float32
```

- Algorithmic patterns

```
Map, Slide, Reduce, Zip
Join, Split
```

- Address space operators

```
toChip, toDram, toOutput
```

- Arithmetic operators

```
ScalarAdd, VVAdd, MVAdd, MMAdd
ScalarMul, VVMul, MVMul, MMMul
VVRelu, VVTanh
```

3.3 Rewrite rules

- Generic and customisable
- 3 levels: DSL, algorithmic, hardware
- Extensible

Split-join rule

```
Map(f)
  ↧   ↧   ↧
Split(n) >>
Map(Map(f)) >>
Join
```

Map fusion rule

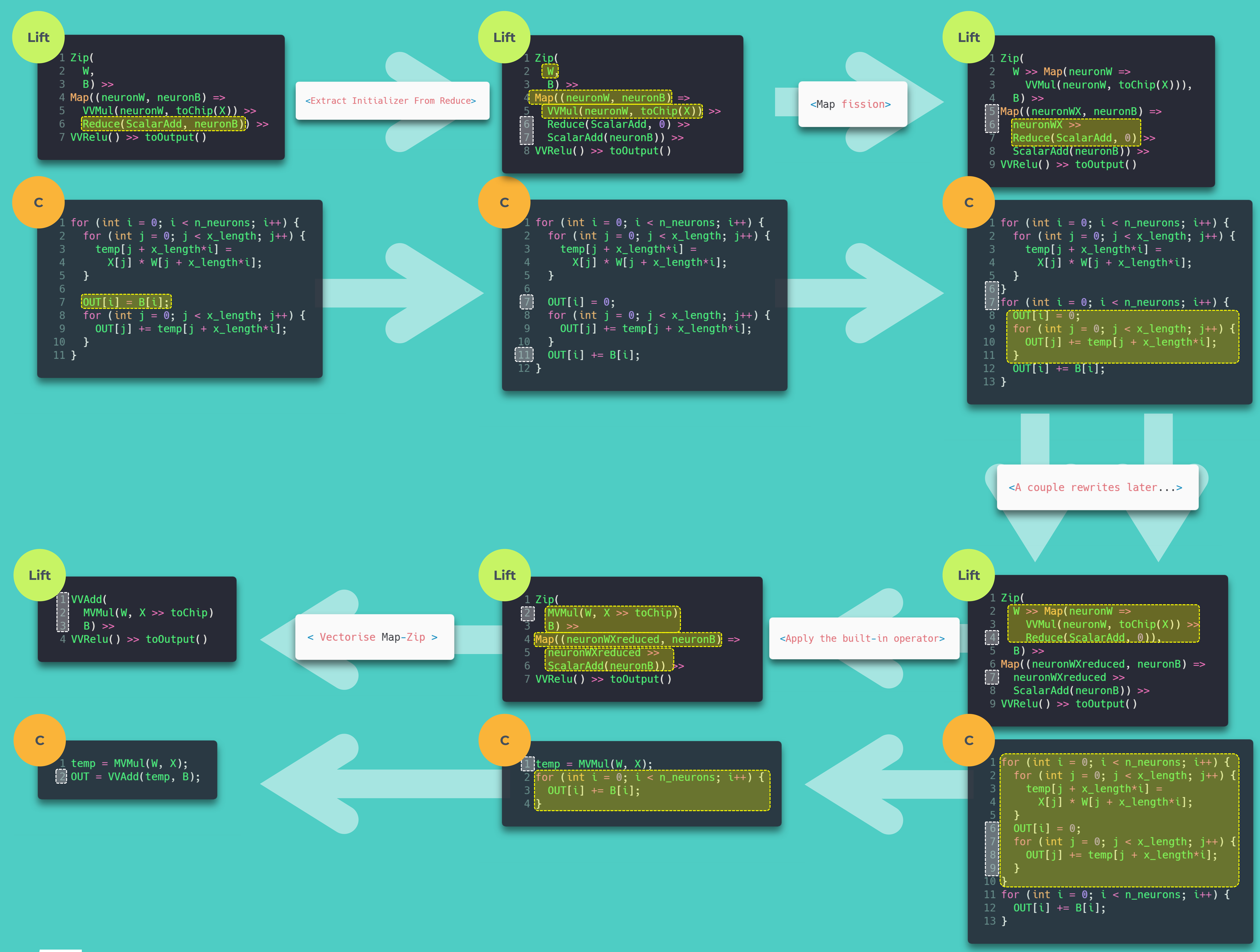
```
Map(f) >> Map(g)
  ↧   ↧   ↧
Map(f >> g)
```

GEMV rule

```
matrix >> Map(row =>
  VVMul(row, vector) >>
  Reduce(ScalarAdd, 0))
      ↧   ↧   ↧
MVMul(matrix, vector)
```

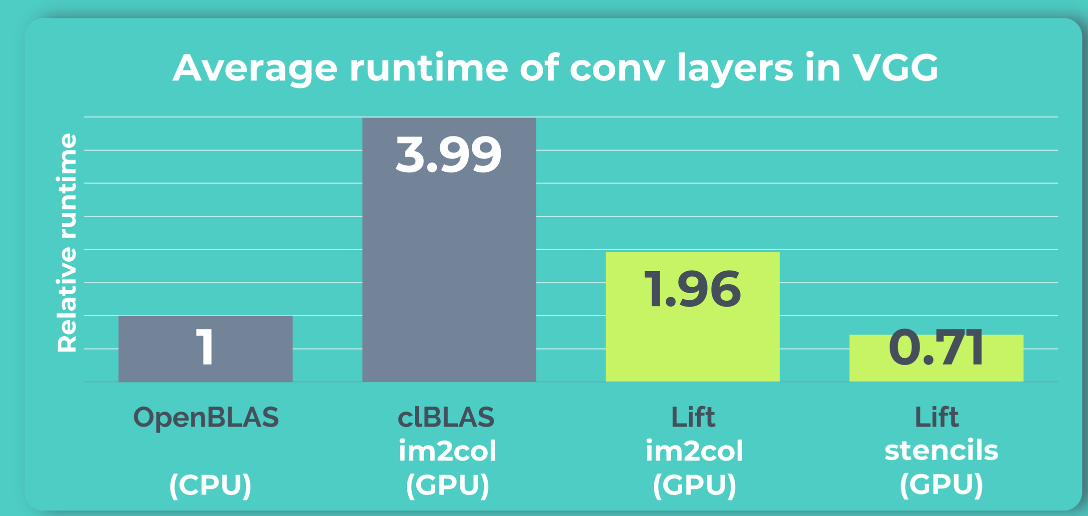
4 Example rewriting

A fully connected layer



5 Preliminary results

- Functional correctness on the BrainWave accelerator
- Performance measurements on Mali GPU



Microsoft Research
EPSRC Centre for Doctoral Training in Pervasive Parallelism
THE UNIVERSITY of EDINBURGH informatics