

Phylanx Frontend and Optimization Opportunities

Theano¹: Overview

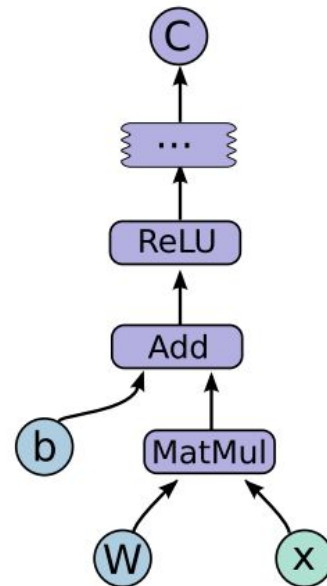
- High-level domain-specific language tailored to numeric computation.
- Symbolic defining of mathematical expressions.
- Python interface.
- Compile symbolic expressions to C for CPU and/or GPU.
- Represent symbolic mathematical expressions as bipartite DAGs
- Graph node types:
 - **Variable:** representing data:
 - TensorType
 - GpuArrayType
 - Sparse
 - **Apply:** representing the application of mathematical operations.

Theano: Overview

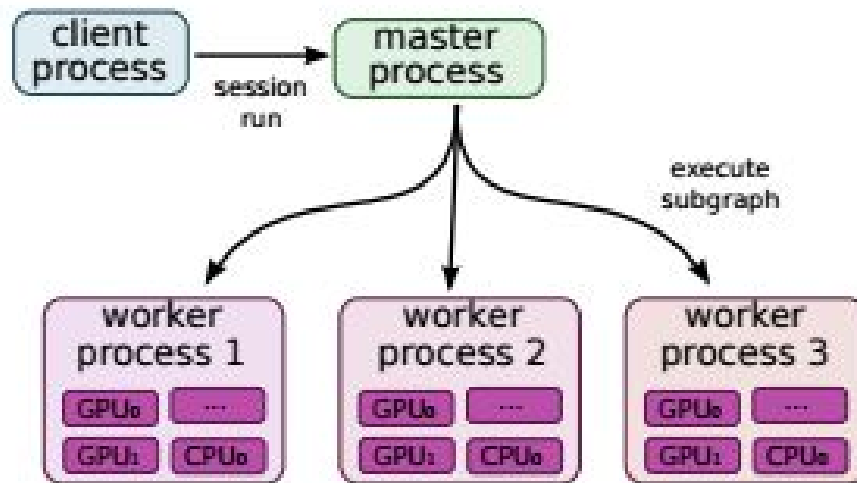
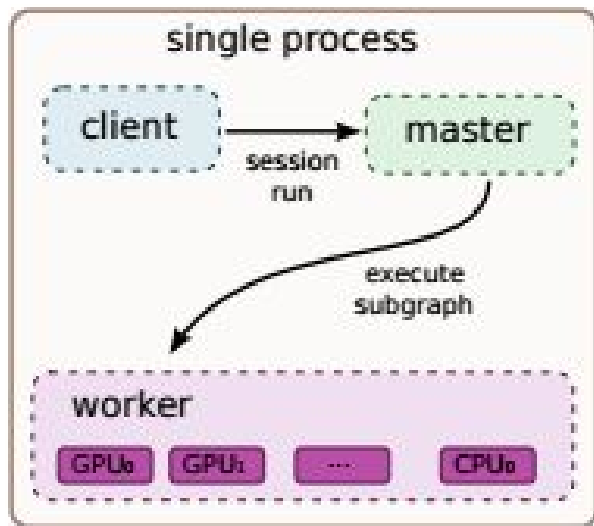
- Support for looping and branching in expressions.
- Automatic speed and stability optimizations.
 - Canonicalize
 - Stabilize
 - Specialize
 - Multiple backends
 - ...

TensorFlow²: Overview

- Symbolic math library for dataflow programming.
 - Mostly used for machine learning applications focusing on deep learning.
- Dataflow graph
 - Nodes: Operations
 - Edges: Tensors (multidimensional arrays.)
- Multi-stage programming
 - Construction phase
 - Execution phase



TensorFlow: Overview



Spartan³: Overview

- Distributed array framework written in Python + Cython.
- Built-ins that directly compute on arrays.
- Lazy-evaluation execution policy.
- Automatic partitioning of n-dimensional arrays:
 - Access pattern of array elements.
 - Access mode of arrays.
 - Arrays' shape and size.
 - Communication cost.

Spartan: Theory

- Access patterns of all operations are categorized by 5 high-level operators:
 - **Map:** $A_{Out} = \text{map}(f_{map}, A_1, A_2, \dots)$
 - Cost: size of all input arrays whose tiling differ from A_1 .
 - **Filter:** $A_{Out} = \text{filter}(f_{predicate}, A_{In})$
 - Cost: zero
 - **Fold:** $A_{Out} = \text{fold}(f_{accumulate}, A_{In}, axis)$
 - Cost: zero along *axis*, otherwise, size of A_{In} .
 - **Scan:** $A_{Out} = \text{scan}(f_{accumulate}, A_{In}, axis)$
 - Cost: zero along *axis*, otherwise, size of A_{In} .
 - **Join and Update:** $A_{Out} = \text{join_update}(f_{join}, f_{accumulate}, A_1, A_2, \dots, axis_1, axis_2, \dots, output_shape)$
 - Cost: zero for each array A_i zero along *axis*_{*i*}, otherwise, size of A_i for $f_{accumulate}$
+ size of A_{out} for f_{join}

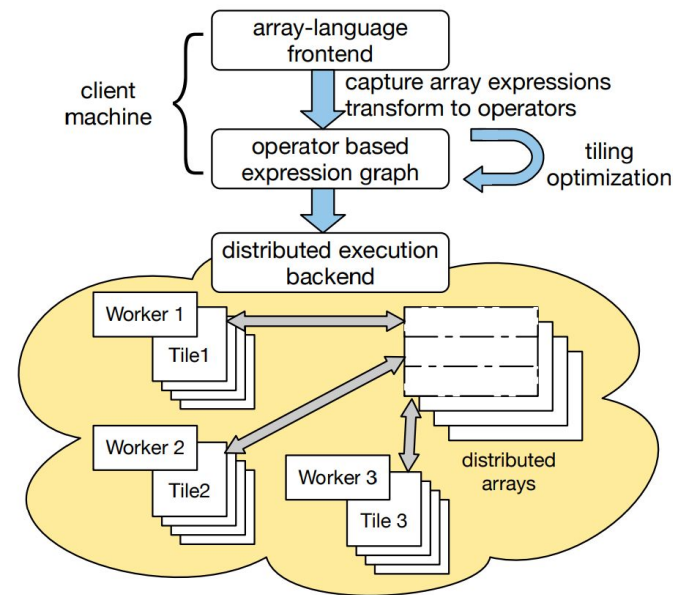
Spartan: Design

Frontend

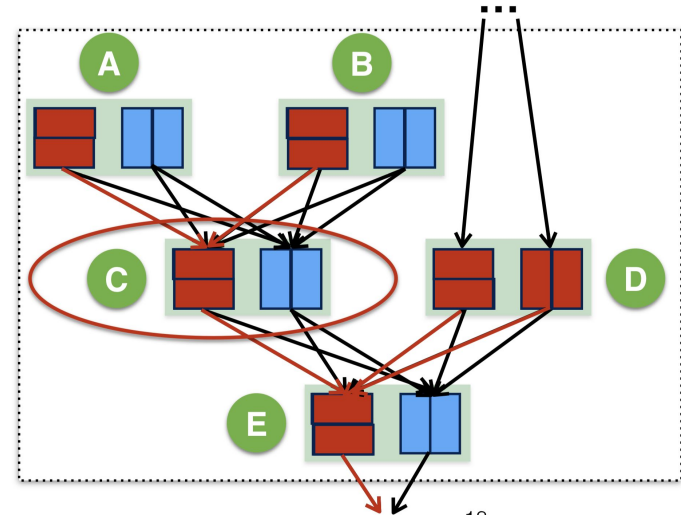
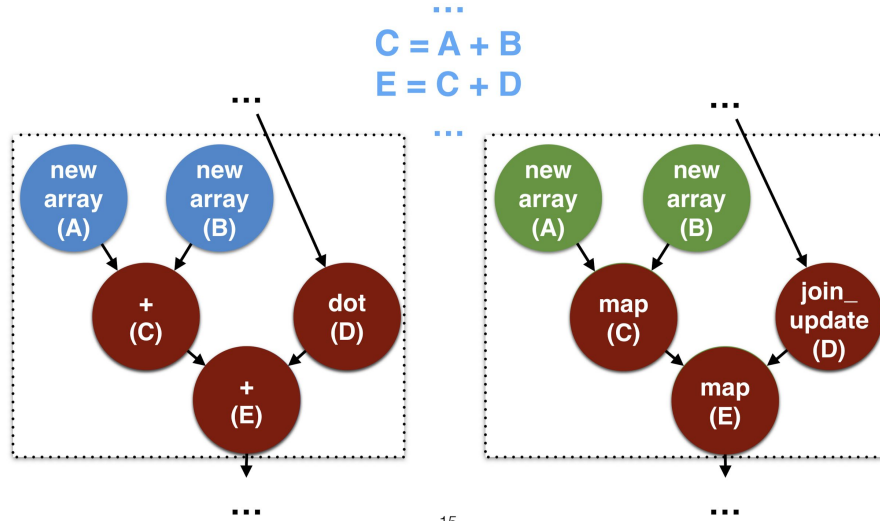
- Turn the user program into an expression graph of high-level operators.
- Run a greedy search algorithm to find a good tiling
- Pass the tiled expression graph to the backend for execution.

Backend

- Create distributed arrays according to the assigned tilings.
- Evaluate each operator by scheduling parallel tasks among a collection of workers.



Spartan: Example



Spartan: Limitations

- Only aims at minimizing network communication.
- The cost profile of *join_update* is not always known:
 - Assumes the upper bound of cost for the join function, i.e., size of the output array.
 - Requires hints from user.
- Tiling algorithm is not refined.
- Estimate the size of the sparse matrices.
- Does not support looping and branching in expressions.

Phylanx: Objectives

- Distributed.
- Symbolic math interface.
- Decoupled optimization and execution engines.
- Multifold optimizations
 - Data decomposition.
 - Graph optimization.
 - Architectural.

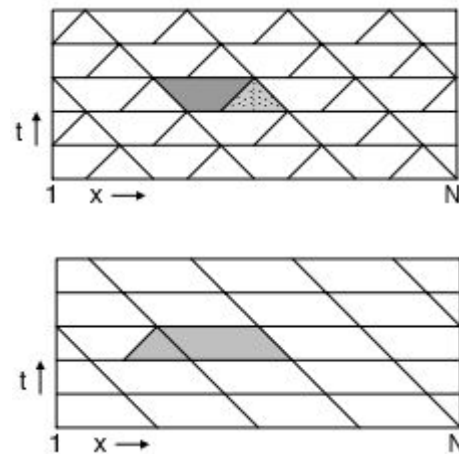
Phylanx: Frontend

- Sympy⁴
 - A Computer Algebra System (CAS)
 - Written in Python + Cython.
 - Provides symbolic arithmetic to many branches of mathematics.
 - Automatic evaluation to the canonical form.

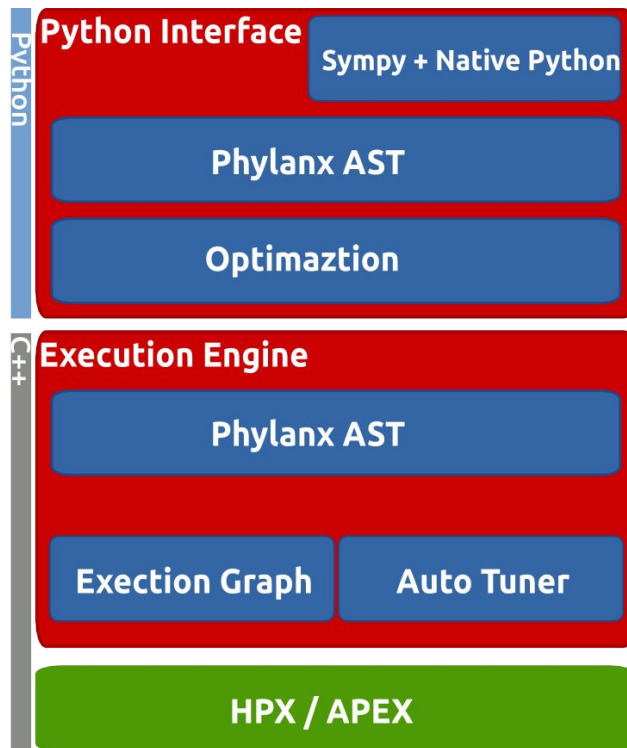
Quick Demo

Phylanx: Optimizations

- Domain decomposition, e.g.,
 - Tile shapes: rectangles, trapezoids, diamonds, ...
 - Split tiling.
 - Overlapped tiling
- Graph optimizations, e.g.,
 - Fuse operations on same memory locations.
 - Common subexpression elimination.
 - Hoisting loop invariants.
- Architectural (Selection and tuning), e.g.,
 - Heuristics
 - Program characteristics
- Misc.
 - Caching
 - Partial execution



Phylanx: Architecture



Thank you!

References

1. <https://github.com/Theano/Theano>
2. <https://github.com/tensorflow/tensorflow>
3. <https://github.com/spartan-array/spartan>