CS 744: TENSORFLOW

Shivaram Venkataraman

Fall 2019

ADMINISTRIVIA

- Assignment 1 Grading: In progress
- Assignment 2 due on ~~Thu~~! Friday
- Course Project: Introduction due Oct 17th

Machine Learning

Bismarck

Supervised learning, Unified Interface
Shared memory, Model fits in memory

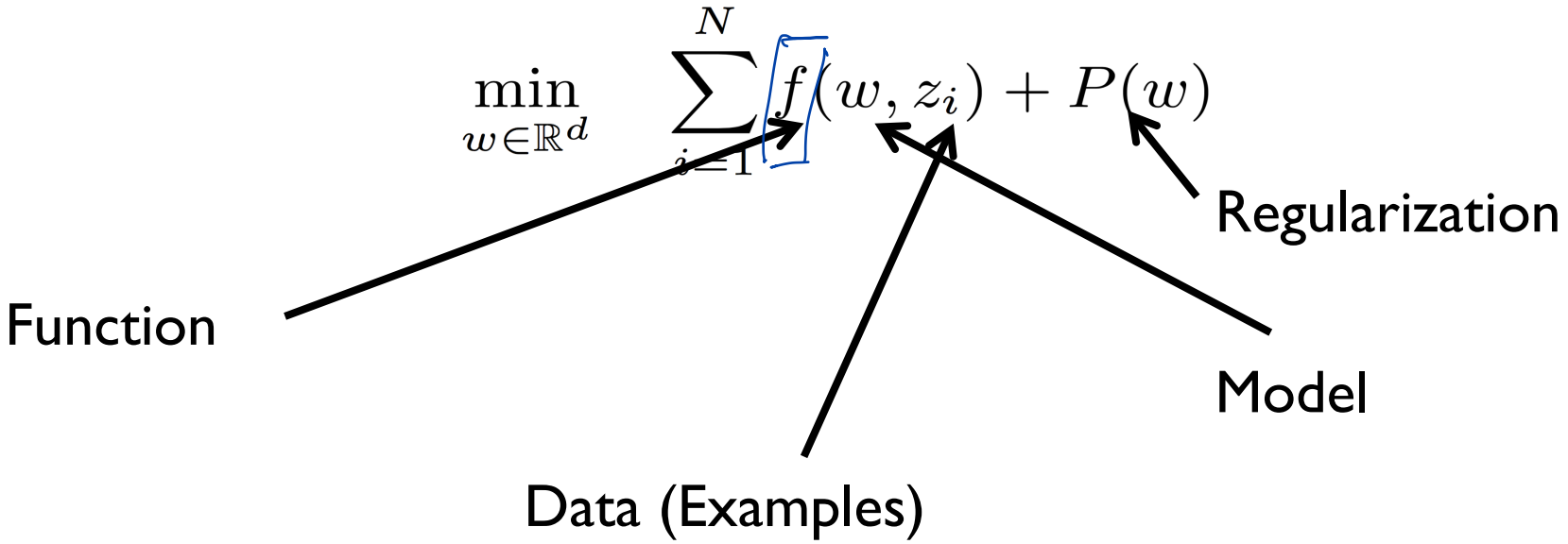
Parameter Server

Large datasets, large models (PB scale)
Shard data, parameters

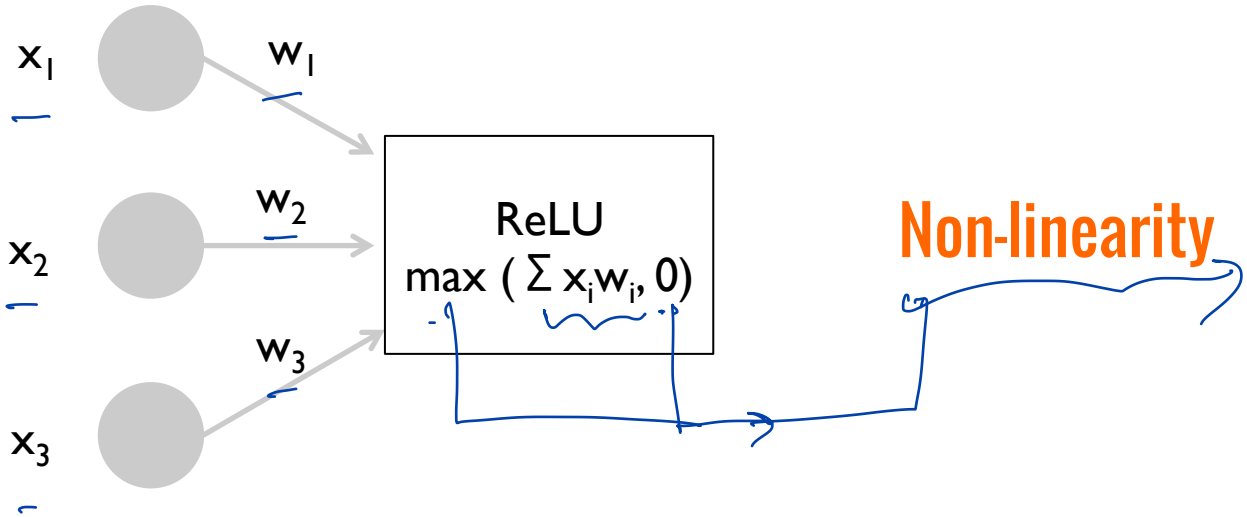
Consistency model
Fault tolerance

WHAT IS DEEP LEARNING?

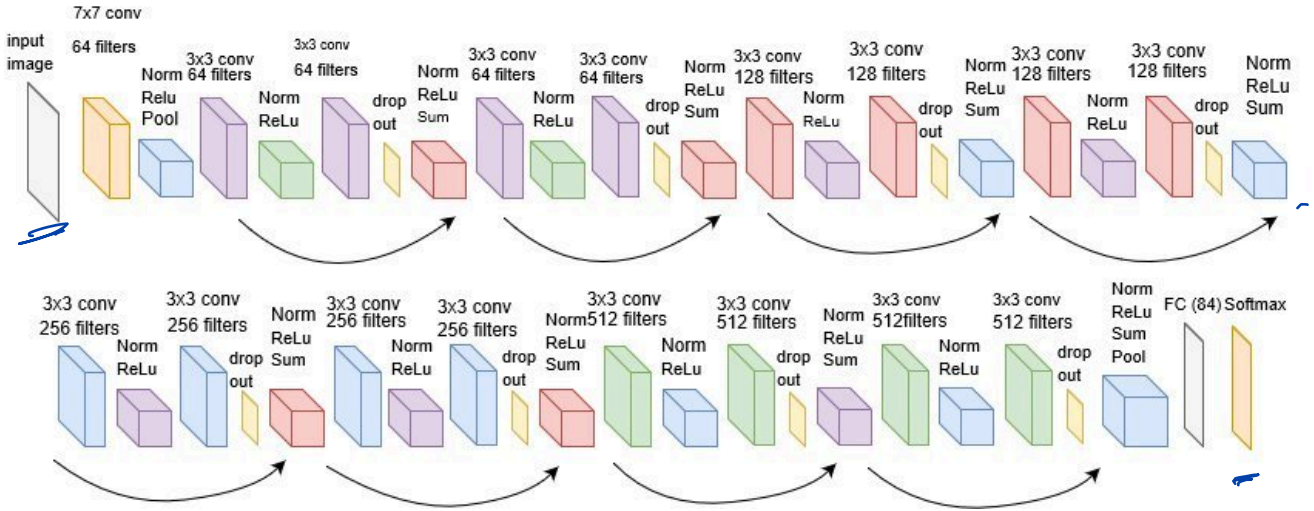
OPTIMIZATION



DEEP LEARNING



DEEP LEARNING



ResNet18

Convolution
ReLU
MaxPool
Fully Connected
SoftMax
...

MODEL TRAINING

$$w^{(k+1)} = w^{(k)} - \alpha_k \nabla f(w^{(k)})$$

Initialize w

For many iterations:

Compute Gradient

Update model

End

Stochastic Gradient Descent

Gradient using backprop
Compute Intensive!

DIST BELIEF SHORTCOMINGS

- Written in C++, hard to experiment
- Writing new types of layers
- New optimization methods
- Execution pattern fixed
  - ↳ Push, Pull

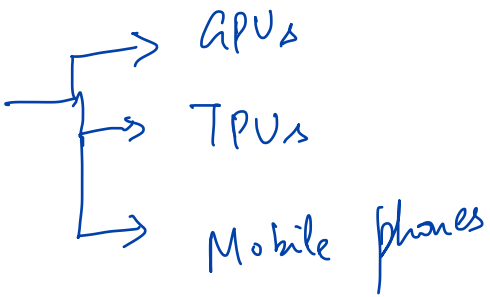
"Flexibility" in how work was partitioned and how comm. happened

TENSORFLOW: DESIGN PRINCIPLES

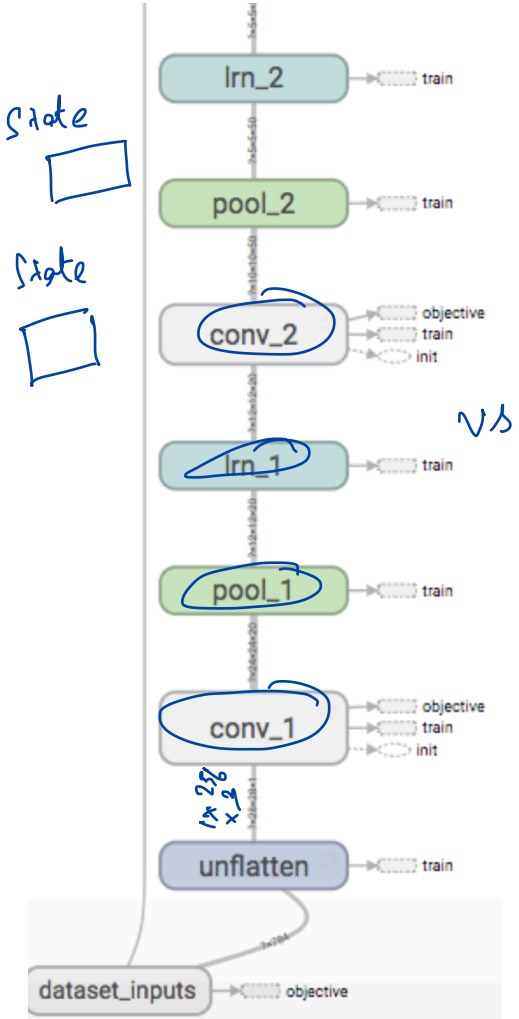
- Dataflow graphs of primitive operators

- Deferred execution: Symbolic dataflow graph

- Heterogeneous accelerators



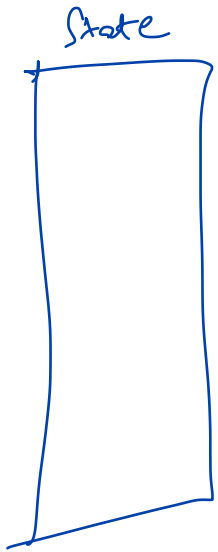
DATAFLOW GRAPH



State

State

VS



Vertices
- Operations ≃ Computation
- State associated ≡ Mutable

Edges
- Tensors

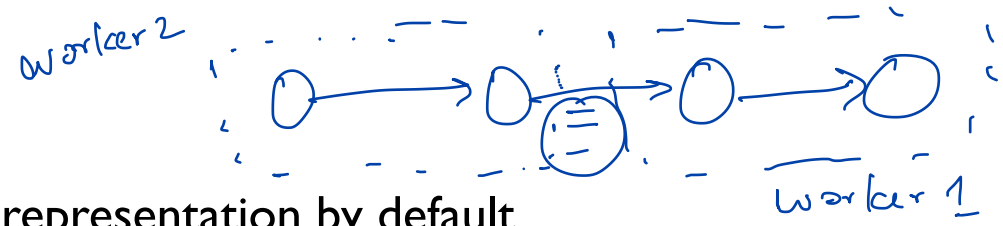
Unify computation and state management

EXECUTION MODEL

Multiple concurrent executions
of overlapping sub-graphs

Vertices have mutable state;
shared between executions

DATAFLOW GRAPH ELEMENTS



Edges: Tensors

- N-dimensional arrays, *dense* representation by default
- Operations take in tensors and return tensors

Vertices: Operations

- Tensor → Tensor functions

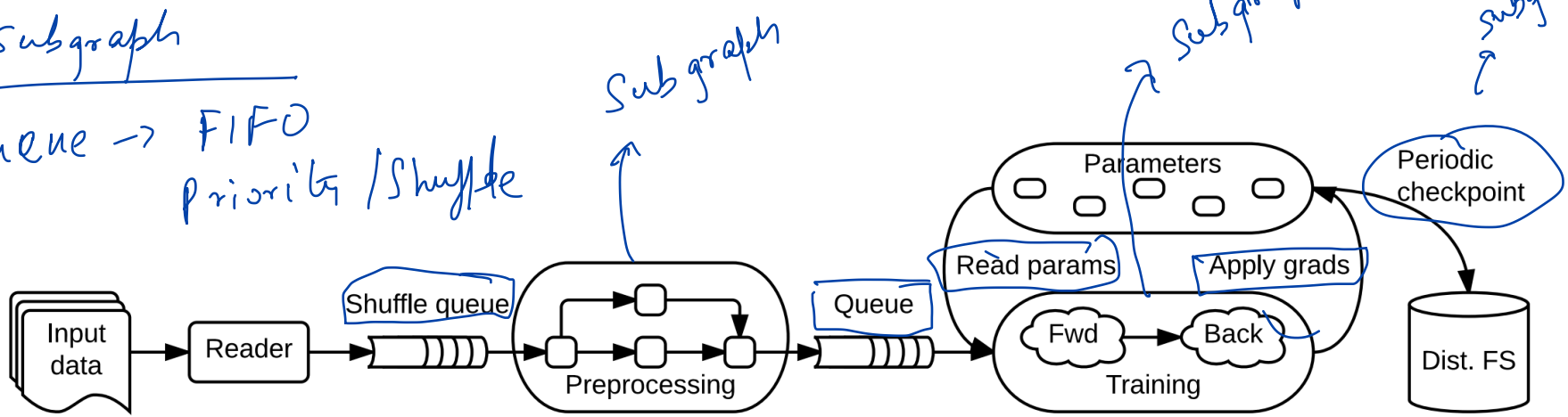
Stateful Operations

- Variables
- Queues

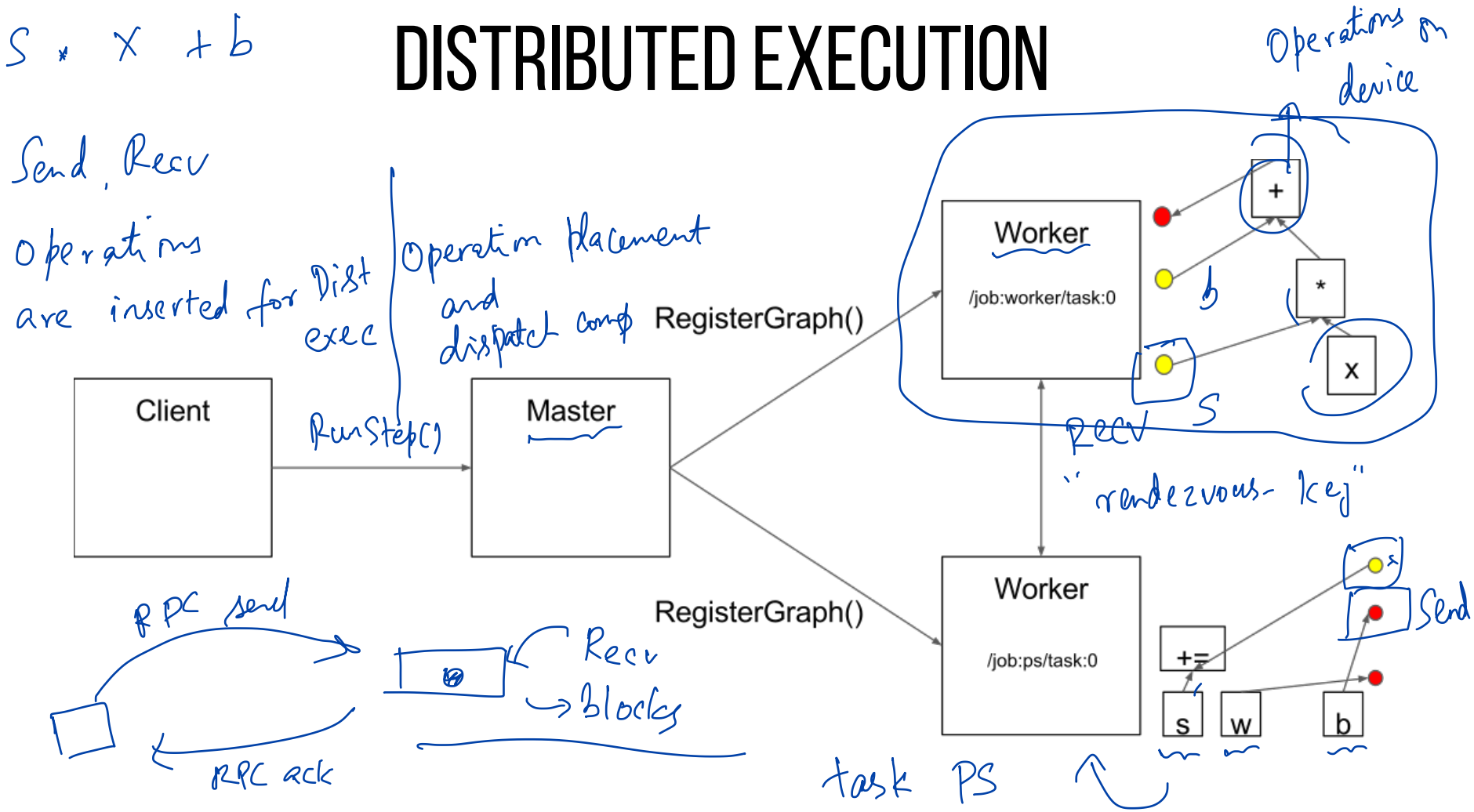
Variables → Mutable State
- Read
- Assign
- Add

Queues → Concurrency
- Bounded size
- Back pressure

PARTIAL EXECUTION



DISTRIBUTED EXECUTION



EXECUTION MODES

Partial Execution

- Input batches from queue
- Concurrent training steps
- Shared model
- “Horizontal” parallelism?

Distributed Execution

- Operations placed on devices
- Account for colocation
- Manual placement decisions?
- Send-Recv to stitch subgraphs

CONTROL FLOW

- Support for RNNs, LSTMs
- Switch and Merge operators to support conditionals
- Enter, Exit, NextIteration to support while loops

```
input = ...   # A sequence of tensors
state = 0     # Initial state
w = ...       # Trainable weights

for i in range(len(input)):
  state, out[i] = f(state, w, input[i])
```

EXTENSIONS

f : logistic regression

∇f : ∇ logistic regression

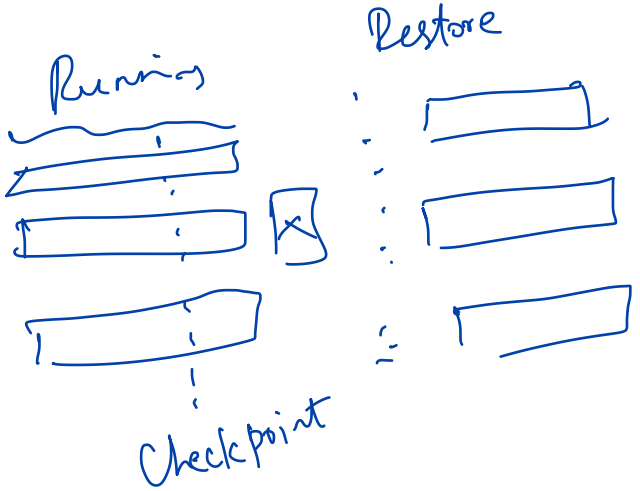
Automatic Differentiation → Powerful

- Given a symbolic expression, generate its gradient
- Also extend to control flow operations

Fault Tolerance

- User-level checkpointing → 1GB
- Save and Restore operations in graph
- Not necessarily consistent ?

Model Parameters

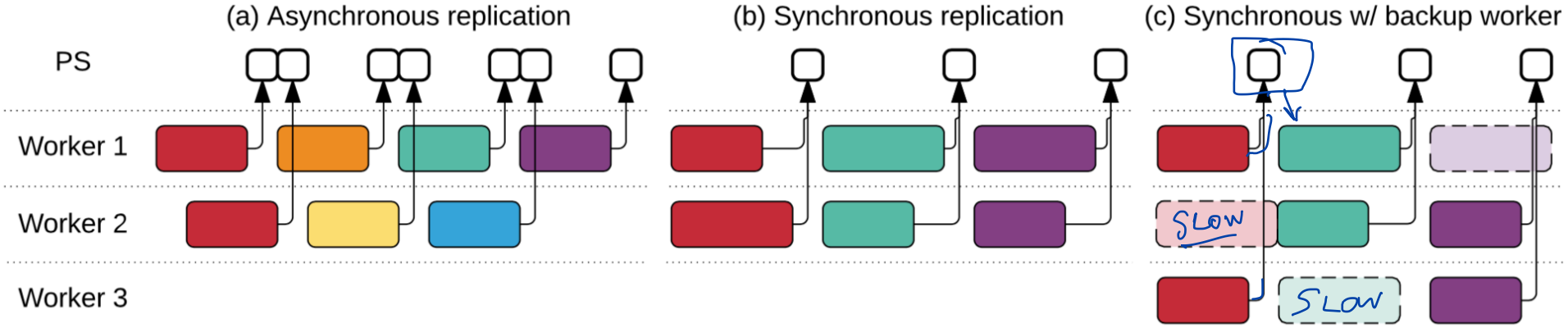


SYNC VS ASYNC

Guarantees?

No such guarantees?

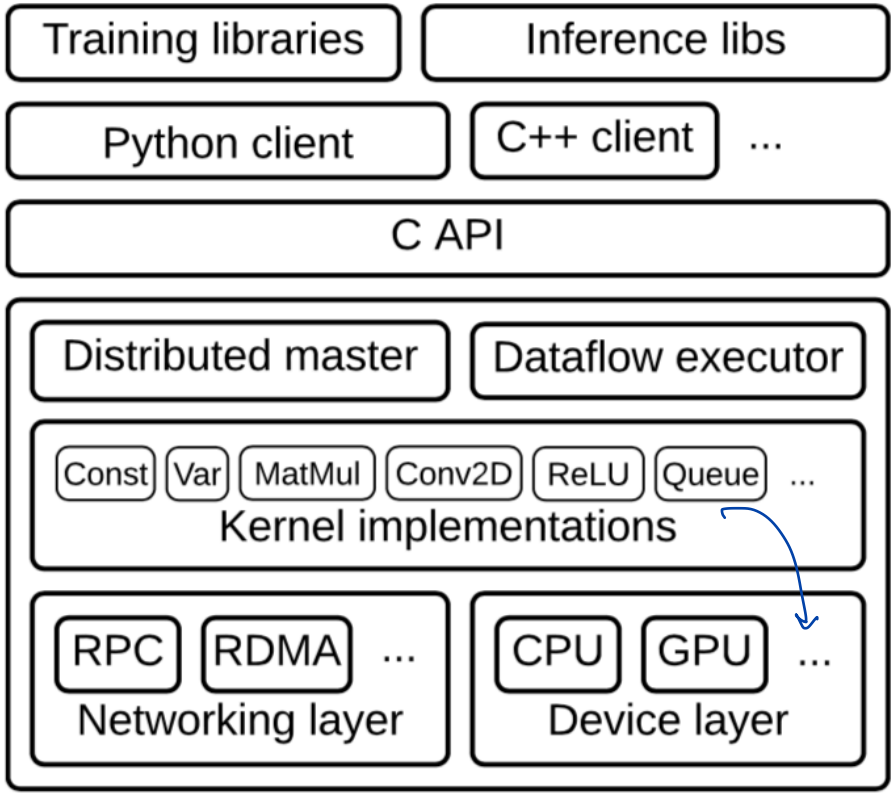
Probability



Stragglers ≡ Fastest 2 out of 3 workers

≈ 48 out of 50

IMPLEMENTATION



Heterogeneous devices

DISCUSSION

https://forms.gle/L9oA69DQe2a7yg3CA

How is the dataflow graph used in Tensorflow similar / different from Apache Spark? What are the implications of that?

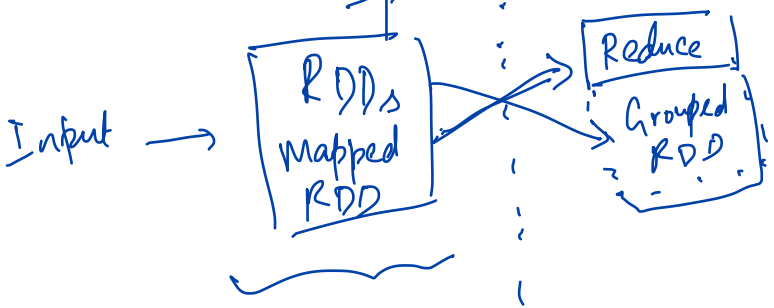
TF

- Vertices: Operations, State
  edges: Tensors

- Multiple subgraphs
  ≈ parallelism within a step

- Vertex resides in 1 node

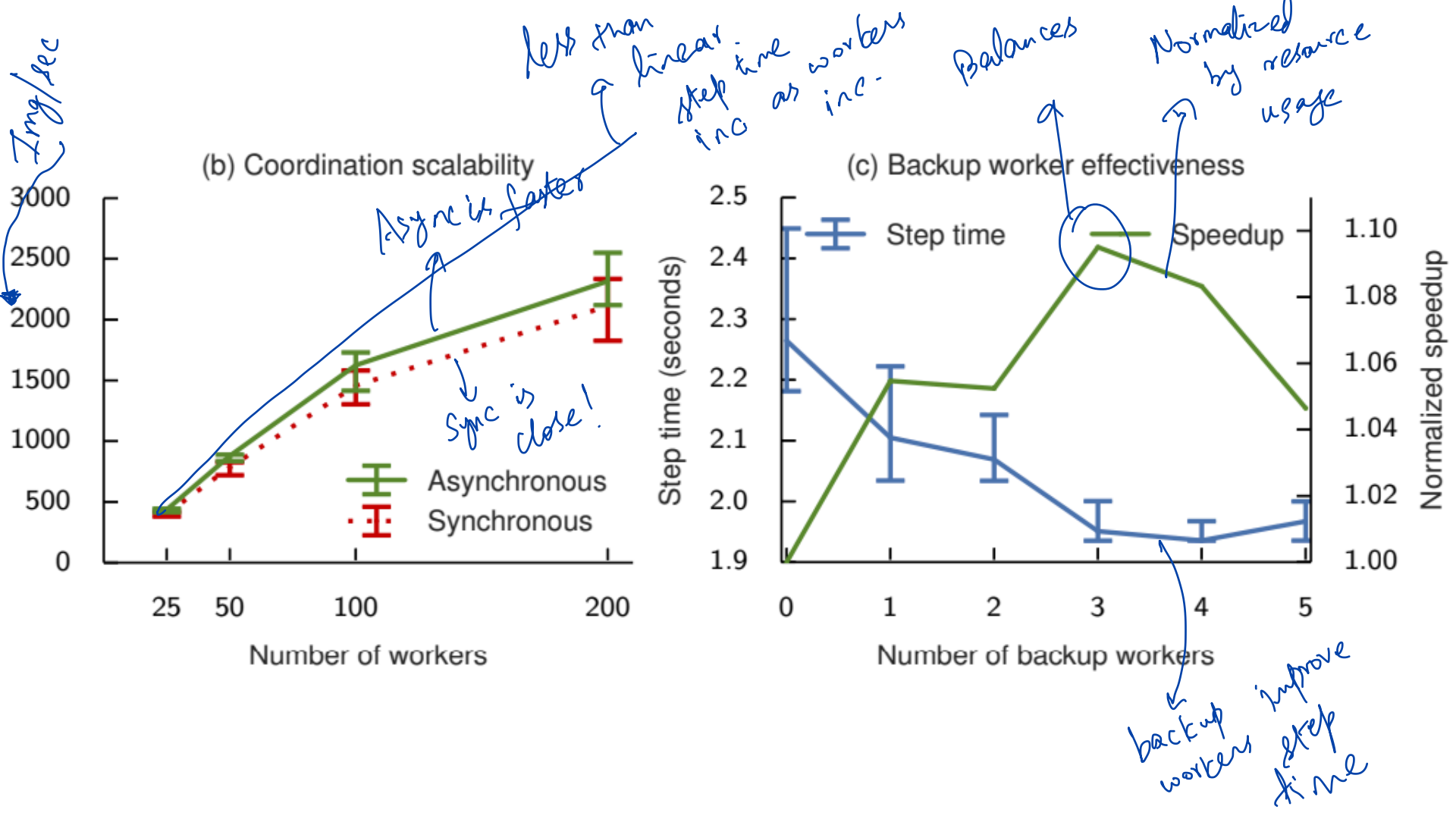
Spark



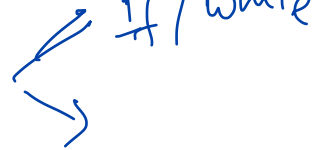
entire cluster

checkpointed

RDDs are partitioned, can do partial recovery


Img/sec
less than linear - step time inc as workers inc.
Balances
Normalized by resource usage
(b) Coordination scalability
Async is faster
Sync is close!
3000
2500
2000
1500
1000
500
0
Asynchronous
Synchronous
25
50
100
200
Number of workers
(c) Backup worker effectiveness
Step time
Speedup
Step time (seconds)
2.5
2.4
2.3
2.2
2.1
2.0
1.9
1.10
1.08
1.06
1.04
1.02
1.00
Normalized speedup
0
1
2
3
4
5
Number of backup workers
backup workers improve step time

What are some shortcomings of the programming model used in Tensorflow? What could be some ways to improve it?

- TF graph is static 

- Placement is low-level not automatic

NEXT STEPS

Next class: Ray

Assignment 2 due this week!

Course project