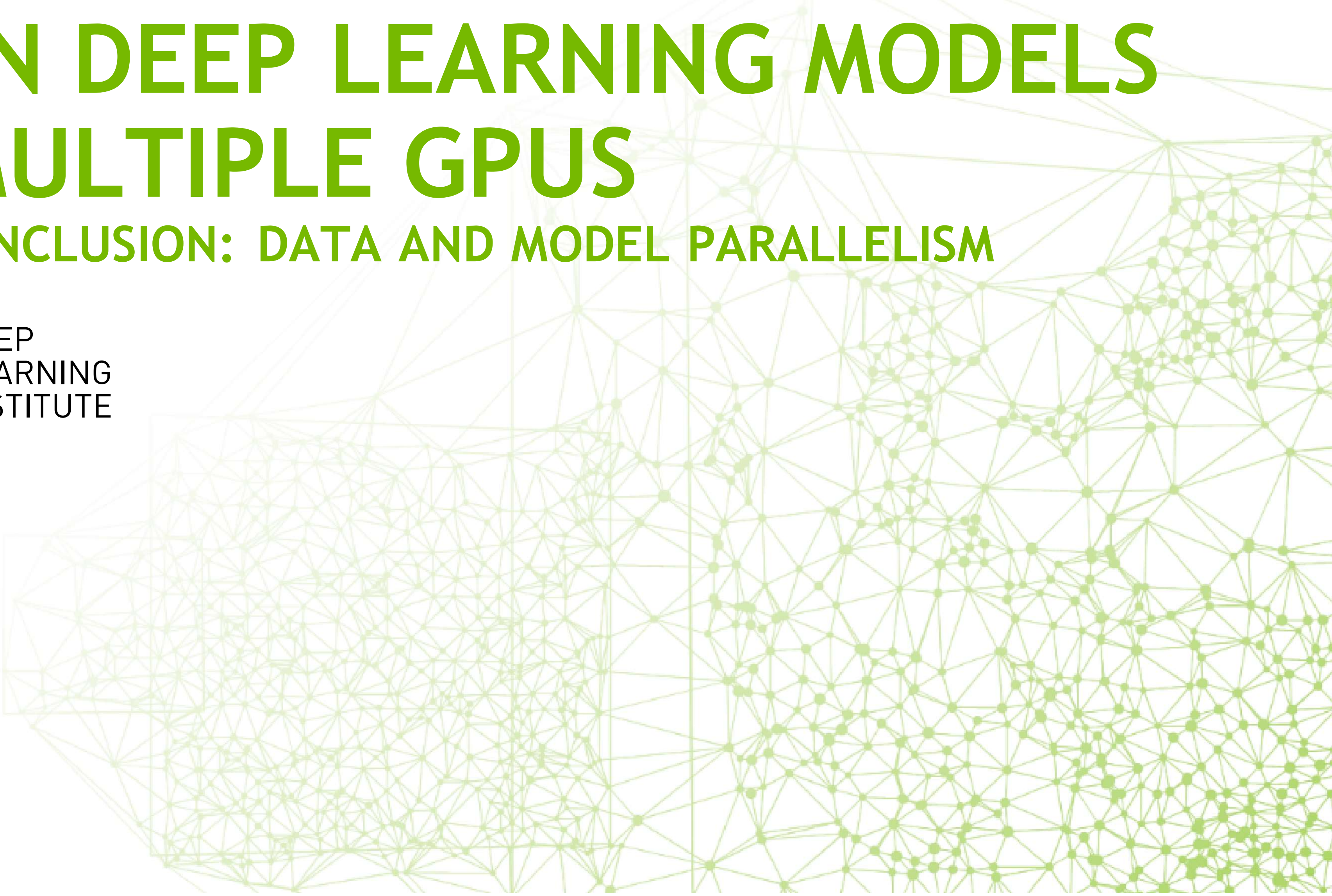
DATA PARALLELISM: HOW TO TRAIN DEEP LEARNING MODELS ON MULTIPLE GPUS

LAB 1 CONCLUSION: DATA AND MODEL PARALLELISM

NVIDIA

DEEP
LEARNING
INSTITUTE



DATA PARALLELISM

Focus of this course

How can we take advantage of multiple GPUs to reduce the training time?

DATA VS MODEL PARALLELISM

Comparison

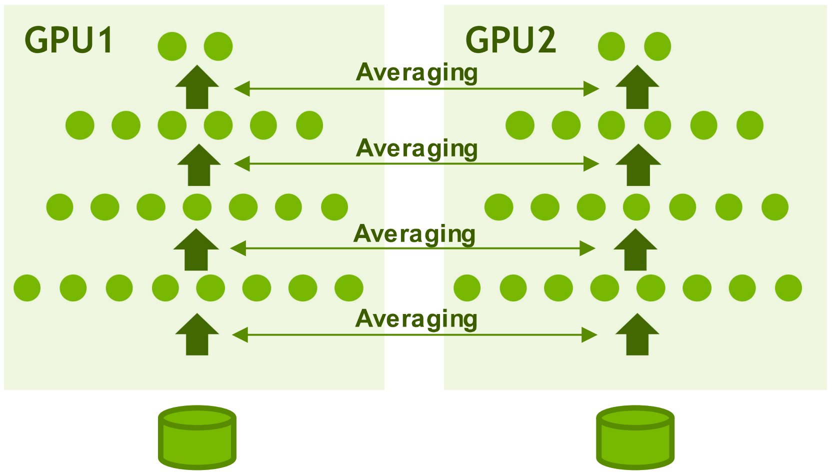
- Data Parallelism
  - Allows you to speed up training
  - All workers train on different data
  - All workers have the same copy of the model
  - Neural network gradients (weight changes) are exchanged

- Model Parallelism
  - Allows you to use a bigger mode
  - All workers train on the same da
  - Parts of the model are distribut across GPUs
  - Neural network activations are exchanged

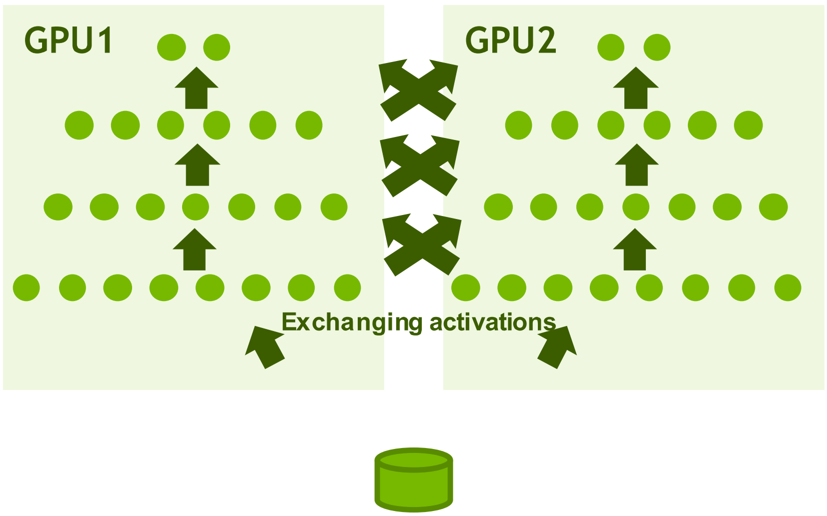
DATA VS MODEL PARALLELISM

Comparison

- Data Parallelism

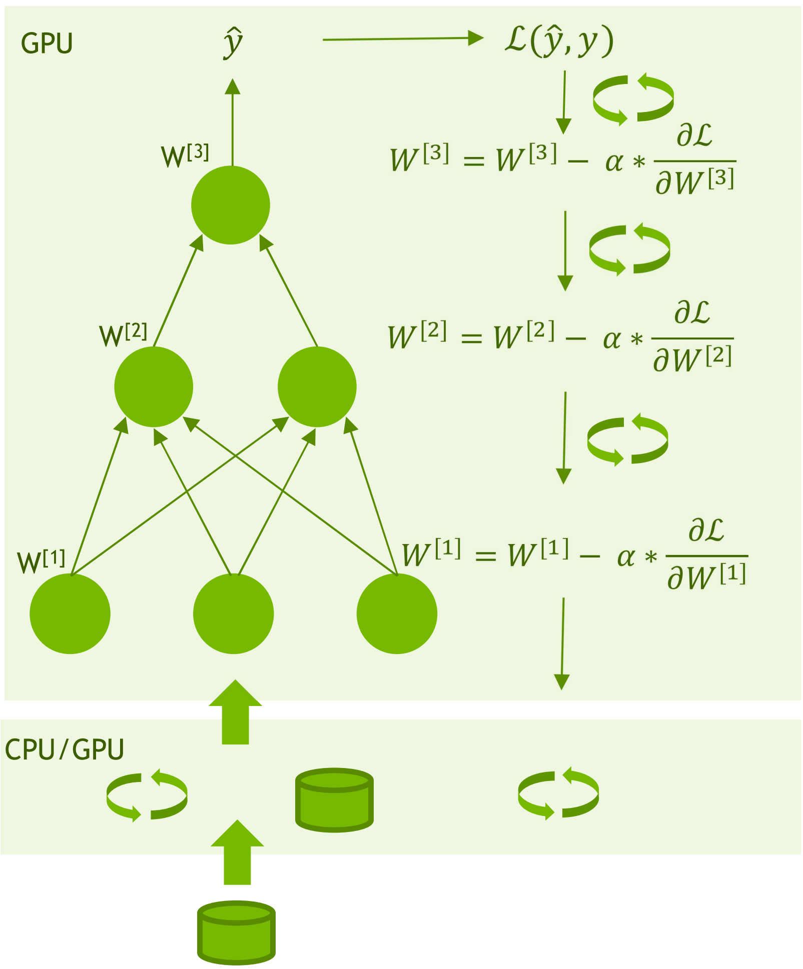


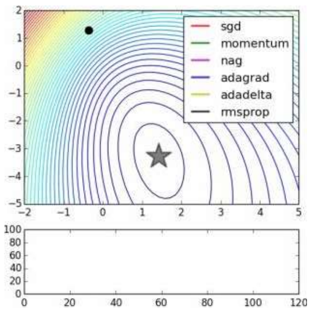
- Model Parallelism



TRAINING A NEURAL NETWORK

Single GPU

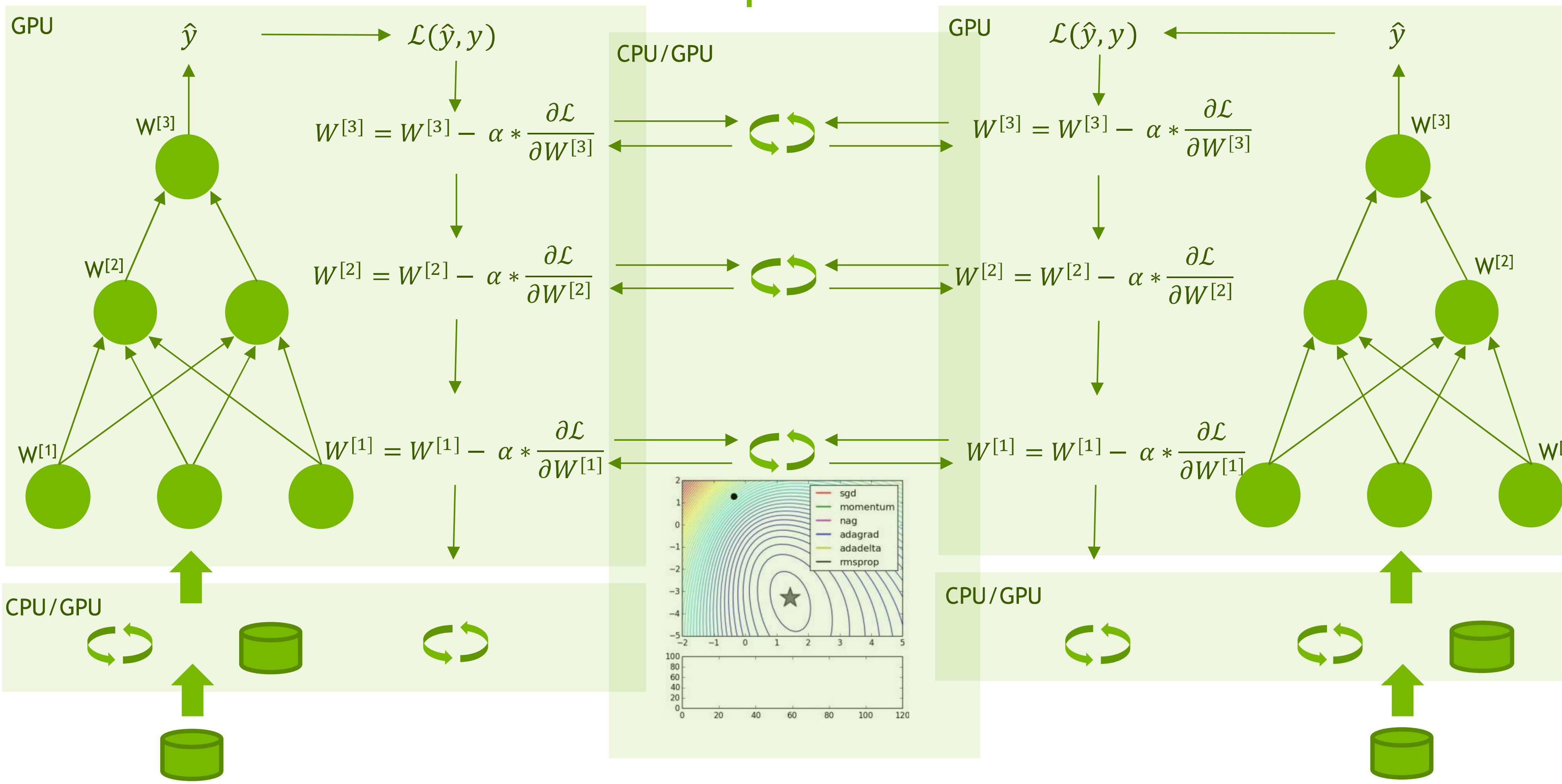

GPU
$\hat{y}$
$\mathcal{L}(\hat{y}, y)$
W[3]
W[2]
W[1]
$W^{[3]} = W^{[3]} - \alpha * \frac{\partial \mathcal{L}}{\partial W^{[3]}}$
$W^{[2]} = W^{[2]} - \alpha * \frac{\partial \mathcal{L}}{\partial W^{[2]}}$
$W^{[1]} = W^{[1]} - \alpha * \frac{\partial \mathcal{L}}{\partial W^{[1]}}$
CPU/GPU



1. Read the data
2. Transport the data
3. Pre-process the data
4. Queue the data
5. Transport the data
6. Calculate activations for layer one
7. Calculate activations for layer two
8. Calculate the output
9. Calculate the loss
10. Backpropagate through layer three
11. Backpropagate through layer two
12. Backpropagate through layer one
13. Execute optimization step
14. Update the weights
15. Return control

TRAINING A NEURAL NETWORK

Multiple GPUs



GPU

$\hat{y}$

$\mathcal{L}(\hat{y}, y)$

$W^{[3]} = W^{[3]} - \alpha * \frac{\partial \mathcal{L}}{\partial W^{[3]}}$

$W^{[2]} = W^{[2]} - \alpha * \frac{\partial \mathcal{L}}{\partial W^{[2]}}$

$W^{[1]} = W^{[1]} - \alpha * \frac{\partial \mathcal{L}}{\partial W^{[1]}}$

CPU/GPU

CPU/GPU

GPU

$\mathcal{L}(\hat{y}, y)$

$\hat{y}$

$W^{[3]} = W^{[3]} - \alpha * \frac{\partial \mathcal{L}}{\partial W^{[3]}}$

$W^{[2]} = W^{[2]} - \alpha * \frac{\partial \mathcal{L}}{\partial W^{[2]}}$

$W^{[1]} = W^{[1]} - \alpha * \frac{\partial \mathcal{L}}{\partial W^{[1]}}$

CPU/GPU

NVIDIA

DEEP
LEARNING
INSTITUTE

www.nvidia.com/dli