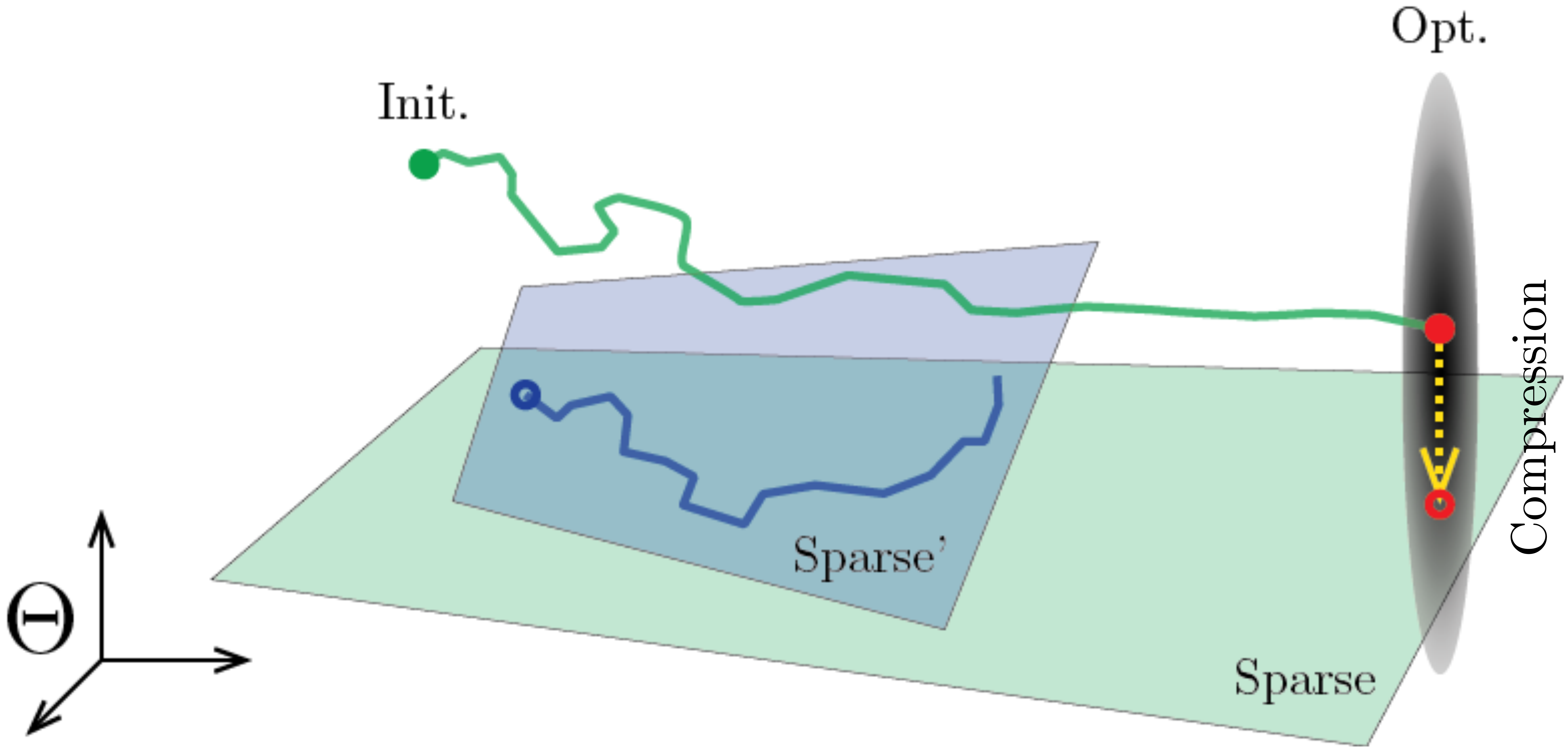
Parameter efficient training of deep convolutional neural networks by dynamic sparse reparameterization

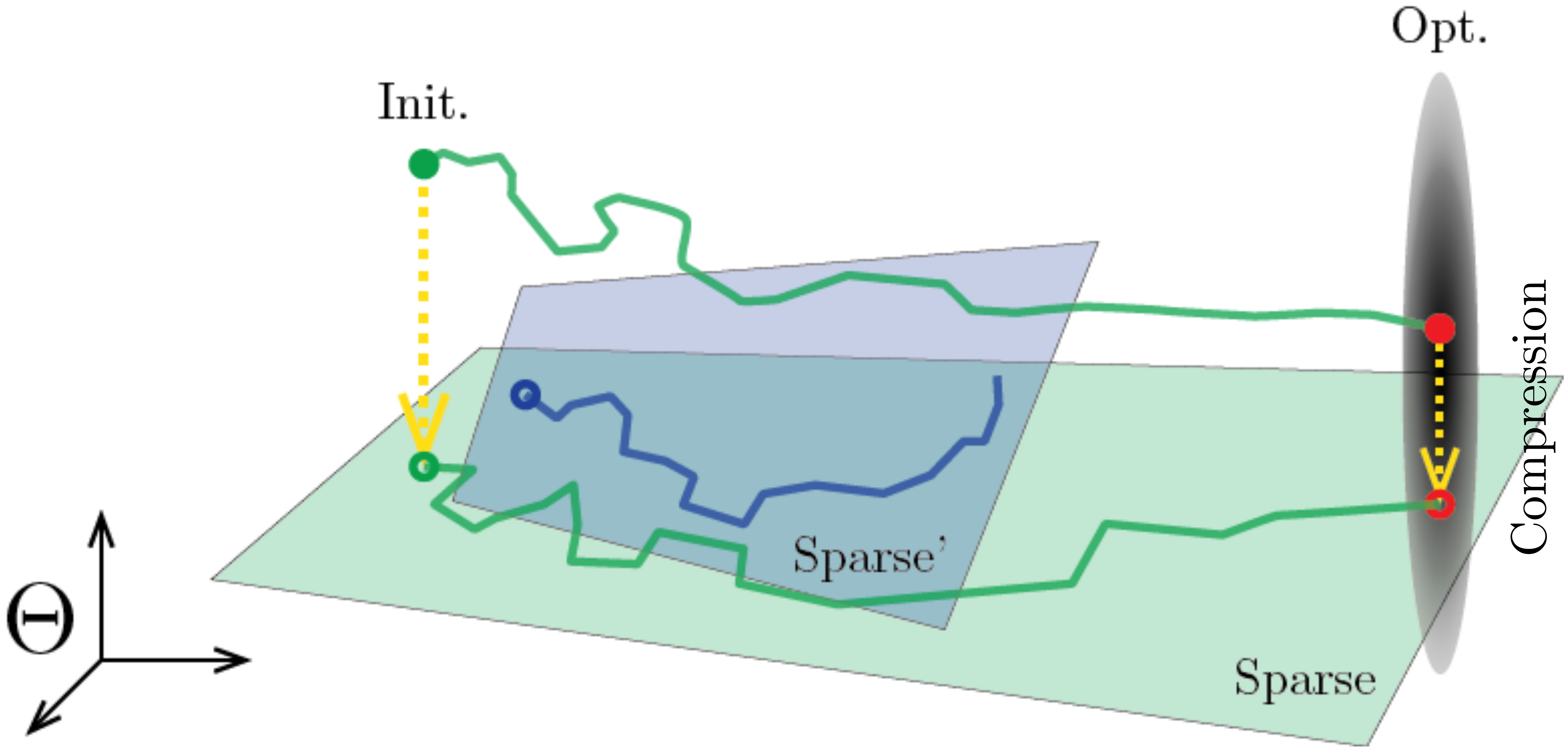
Hesham Mostafa (Intel AI)
Xin Wang (Intel AI, Cerebras Systems)

Easy: post-training (sparse) compression

Hard: direct training of sparse networks

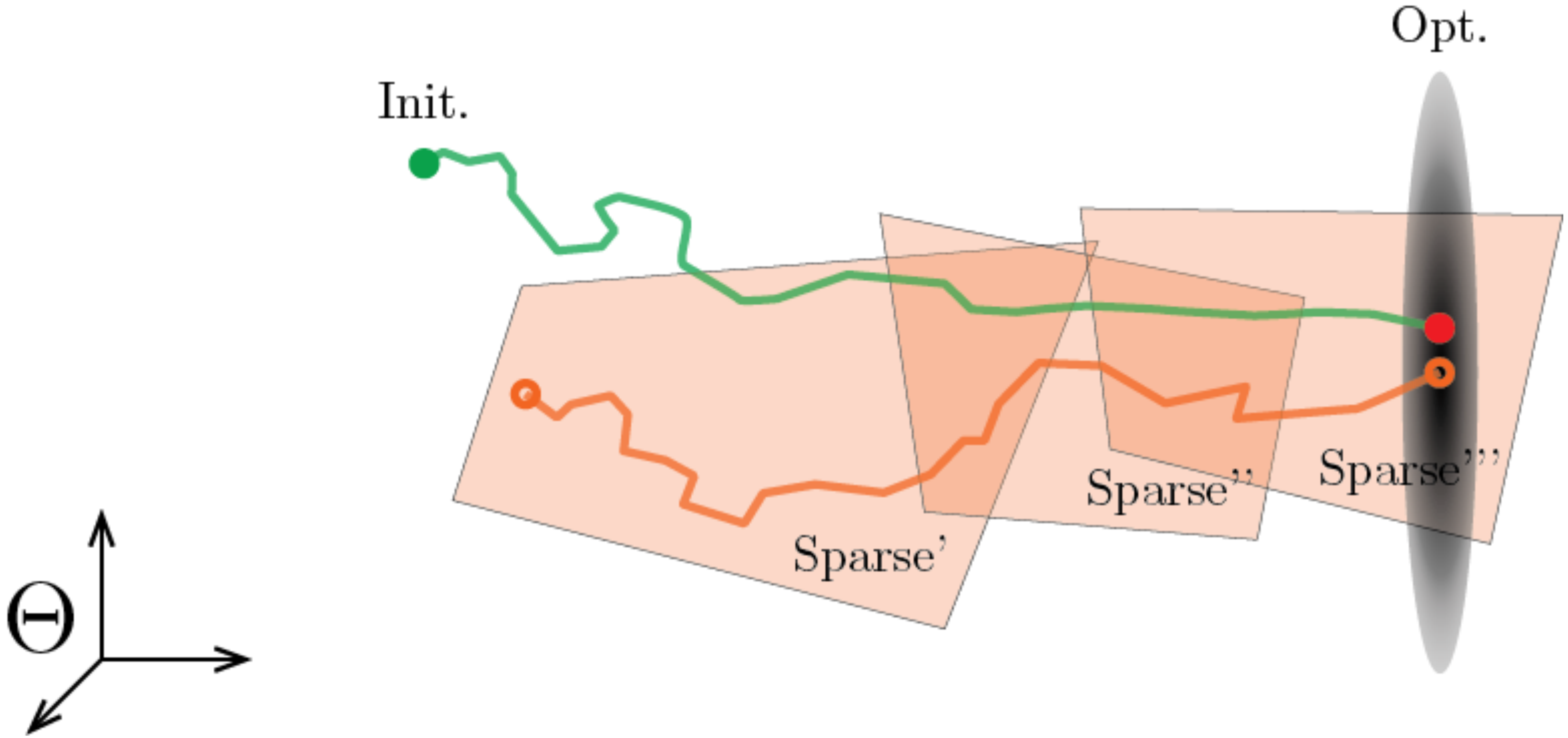


“Winning lottery tickets” (Frankle & Carbin 2018):
post hoc identification of trainable sparse nets



Dynamic sparse reparameterization (ours):

training-time structural exploration



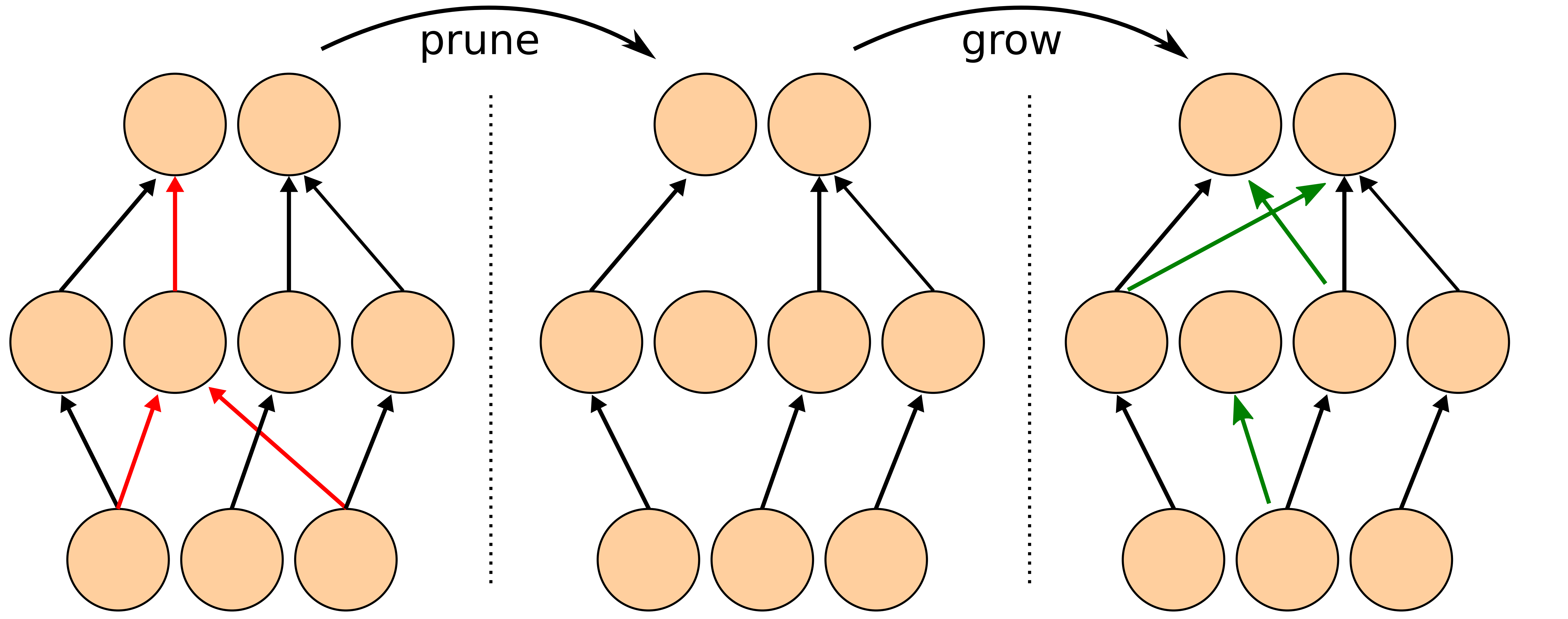
Direct training sparse nets to generalize as well as post-training compression:

is this possible? -***YES***

Directly trained sparse nets:

are they "winning lottery tickets"? -***NO***

Dynamic sparse reparameterization



for each sparse parameter tensor $\mathbf{W}_i$ **do**

  $(\mathbf{W}_i, k_i) \leftarrow \texttt{prune\_by\_threshold}(\mathbf{W}_i, H)$ ▷ k_i is the number of pruned weights

  $l_i \leftarrow \texttt{number\_of\_nonzero\_entries}(\mathbf{W}_i)$ ▷ Number of surviving weights after pruning

end for

$(K, L) \leftarrow (\sum_i k_i, \sum_i l_i)$ ▷ Total number of pruned and surviving weights

$H \leftarrow \texttt{adjust\_pruning\_threshold}(H, K, \delta)$ ▷ Adjust pruning threshold

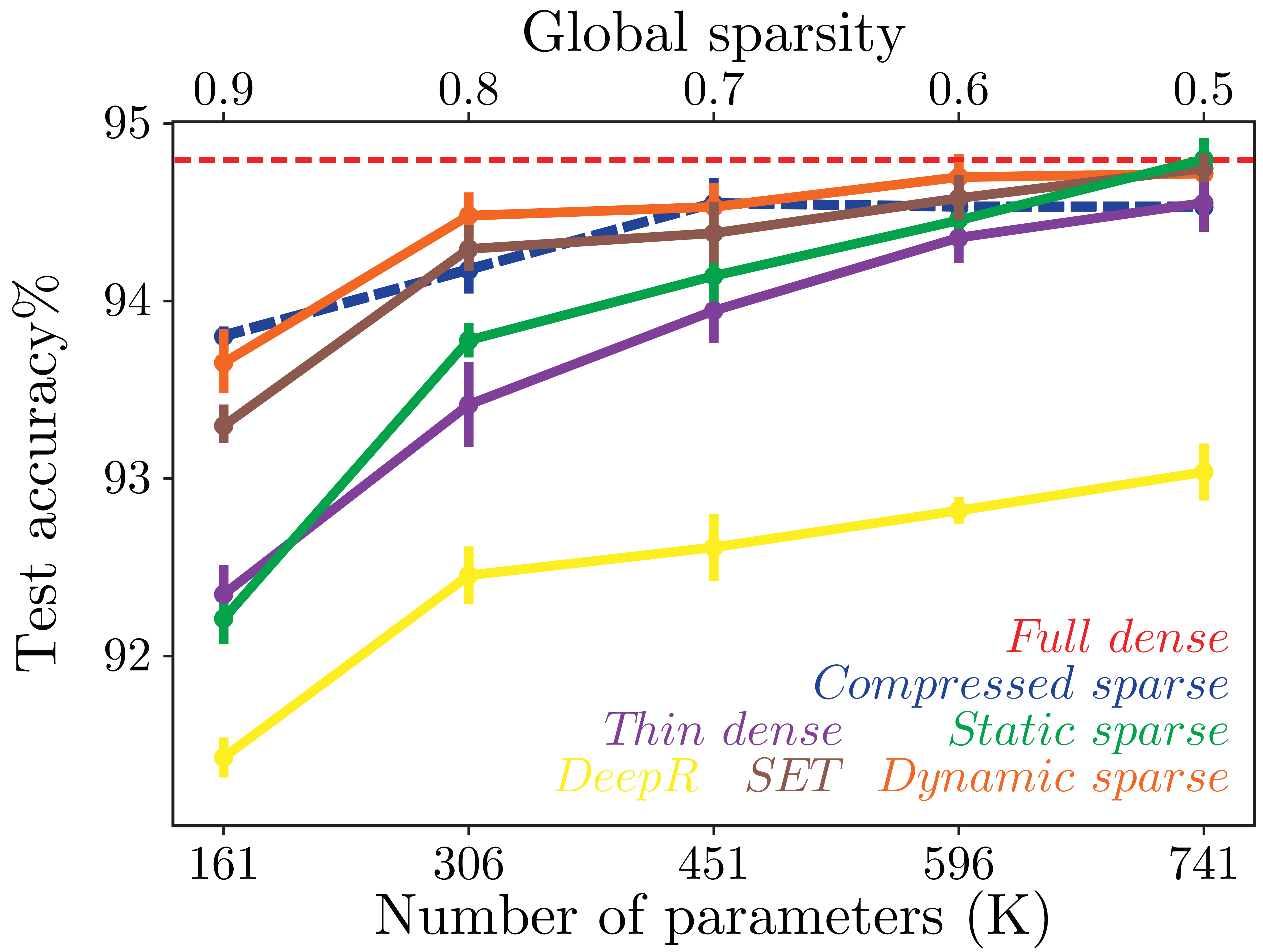
for each sparse parameter tensor $\mathbf{W}_i$ **do**

  $\mathbf{W}_i \leftarrow \texttt{grow\_back}(\mathbf{W}_i, \frac{l_i}{L} K)$ ▷ Grow $\frac{l_i}{L} K$ zero-initialized weights at random in $\mathbf{W}_i$

end for

Closed gap between post-training compression and direct training of sparse nets

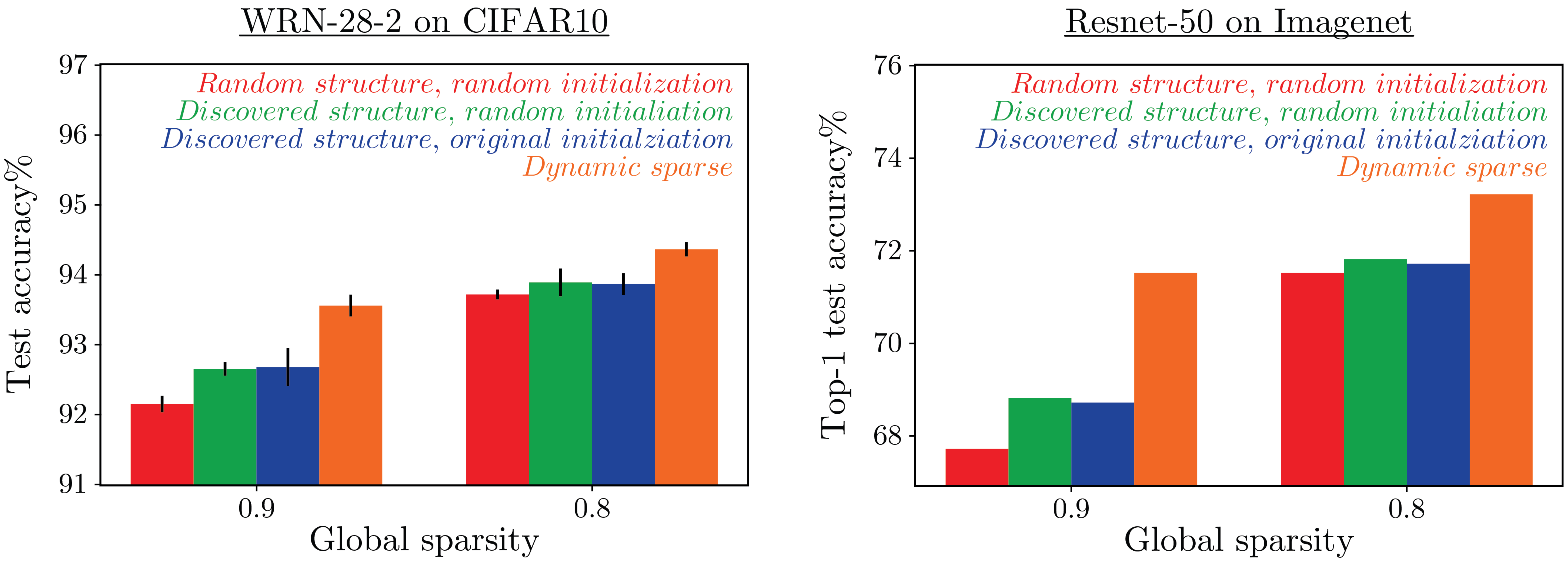
WRN-28-2 on CIFAR10



Resnet-50 on Imagenet

Sparsity (# Param)	0.8 (7.3M)		0.9 (5.1M)		0.0 (25.6M)	
Thin dense	72.4 [-2.5]	90.9 [-1.5]	70.7 [-4.2]	89.9 [-2.5]		
Static sparse	71.6 [-3.3]	90.4 [-2.0]	67.8 [-7.1]	88.4 [-4.0]		
DeepR (Bellec et al., 2017)	71.7 [-3.2]	90.6 [-1.8]	70.2 [-4.7]	90.0 [-2.4]	74.9 [0.0]	92.4 [0.0]
SET (Mocanu et al., 2018)	72.6 [-2.3]	91.2 [-1.2]	70.4 [-4.5]	90.1 [-2.3]		
Dynamic sparse (Ours)	**73.3** **[-1.6]**	**92.4** **[0.0]**	**71.6** **[-3.3]**	**90.5** **[-1.9]**		
Compressed sparse (Zhu & Gupta, 2017)	73.2 [-1.7]	91.5 [-0.9]	70.3 [-4.6]	90.0 [-2.4]		

Directly trained sparse nets are not “winning tickets”: **exploration of structural degrees of freedom is crucial**



Visit our poster:

Wednesday, Pacific Ballroom #248

PARAMETER EFFICIENT TRAINING OF DEEP CONVOLUTIONAL NEURAL NETWORKS BY DYNAMIC SPARSE REPARAMETERIZATION

Hesham Mostafa[1], **Xin Wang**[1,2].

1. Artificial Intelligence Products Group, Intel Corporation; 2. Cerebras Systems.

Overview

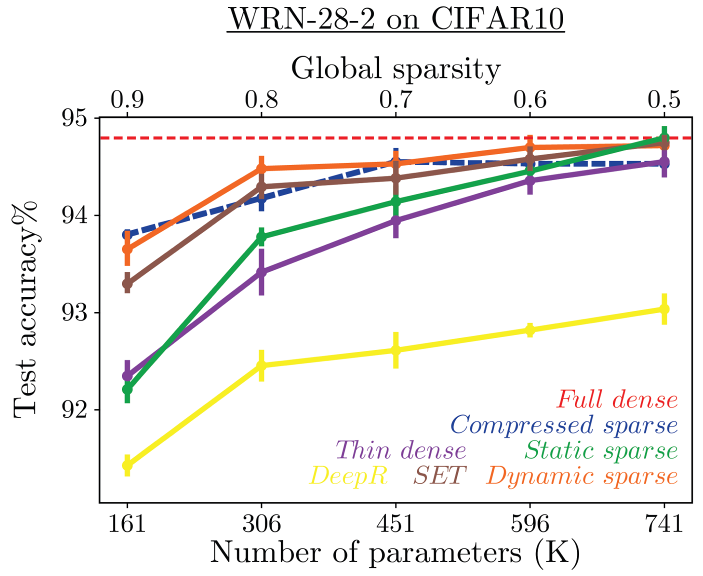
- It has long been thought that **direct training of a small, sparse deep convolution netwrork *de novo*** is much more difficult than post-training compression of a large, dense model.
- Here we challenged this belief by presenting a **dynamic sparse reparameterization** technique that closed the performance gap between iterative pruning of a dense model and direct training of a sparse one.
- We further showed that **"lottery tickets"** (Frankle & Carbin, 2018) **do not always exist**, and **training-time structural exploration is crucial** to learning by sparse networks, so much so that **adding structural degrees of freedom is often more effective than adding extra free parameters**.

Training-time structural exploration by dynamic sparse reparameterization

- Our method is based on a simple **dynamic parameter reallocation** procedure, performed once every hundreds of batch iterations during training, yielding best accuracies at a given sparsity.

for each sparse parameter tensor $\mathbf{W}_i$ **do**
  $(\mathbf{W}_i, k_i) \leftarrow \texttt{prune\_by\_threshold}(\mathbf{W}_i, H)$ ▷ k_i is the number of pruned weights
  $l_i \leftarrow \texttt{number\_of\_nonzero\_entries}(\mathbf{W}_i)$ ▷ Number of surviving weights after pruning
end for
$(K, L) \leftarrow (\sum_i k_i, \sum_i l_i)$ ▷ Total number of pruned and surviving weights
$H \leftarrow \texttt{adjust\_pruning\_threshold}(H, K, \delta)$ ▷ Adjust pruning threshold
for each sparse parameter tensor $\mathbf{W}_i$ **do**
  $\mathbf{W}_i \leftarrow \texttt{grow\_back}(\mathbf{W}_i, \frac{l_i}{L}K)$ ▷ Grow $\frac{l_i}{L}K$ zero-initialized weights at random in $\mathbf{W}_i$
end for

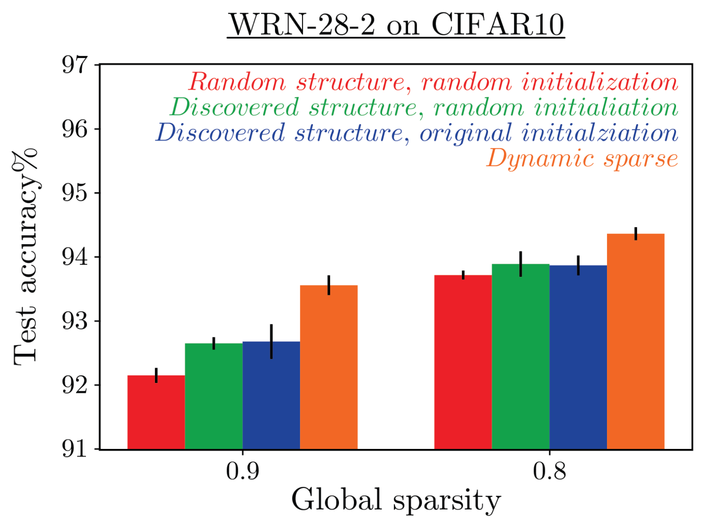
- We benchmarked our *dynamic sparse* training against *full dense* (original overparameterized model), *compressed sparse* (post-training iterative pruning), *thin dense* (small dense model with matching parameter count), *static sparse* (sparse model with fixedstructure), *DeepR* and *SET* (previous dynamic sparse methods).

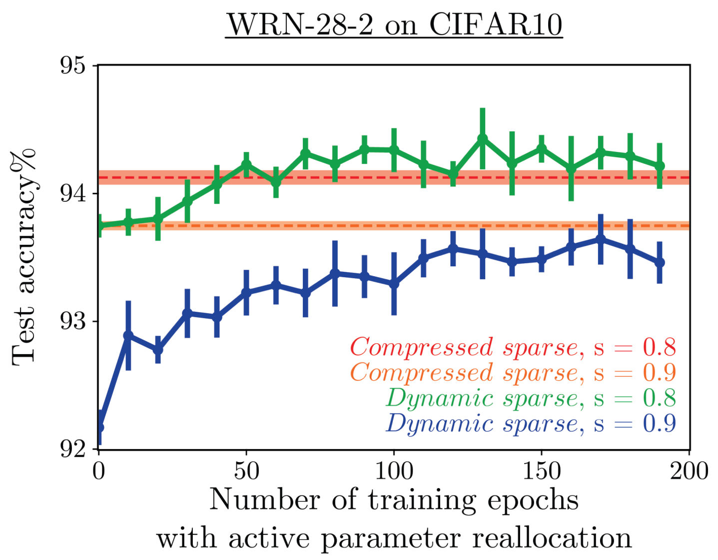


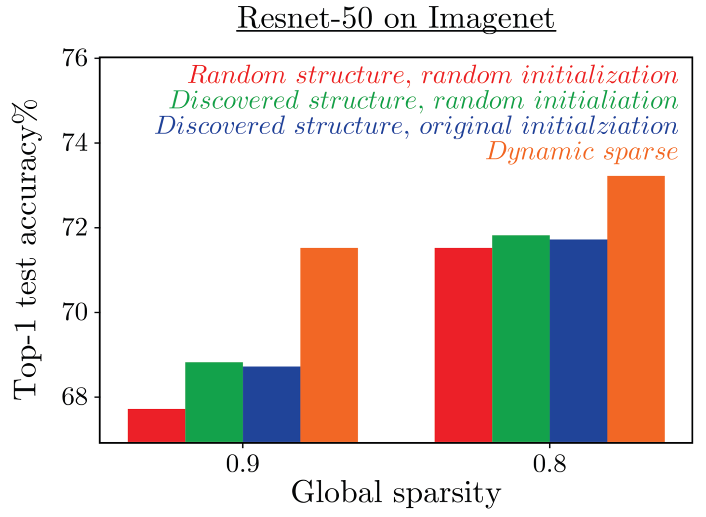
Resnet-50 on Imagenet

Sparsity (# Param)	0.8 (7.3M)		0.9 (5.1M)		0.0 (25.6M)	
Thin dense	72.4 [-2.5]	90.9 [-1.5]	70.7 [-4.2]	89.9 [-2.5]		
Static sparse	71.6 [-3.3]	90.4 [-2.0]	67.8 [-7.1]	88.4 [-4.0]		
DeepR (Bellec et al., 2017)	71.7 [-3.2]	90.6 [-1.8]	70.2 [-4.7]	90.0 [-2.4]	74.9 [0.0]	92.4 [0.0]
SET (Mocanu et al., 2018)	72.6 [-2.3]	91.2 [-1.2]	70.4 [-4.5]	90.1 [-2.3]		
Dynamic sparse (Ours)	**73.3** **[-1.6]**	**92.4** **[0.0]**	**71.6** **[-3.3]**	**90.5** **[-1.9]**		
Compressed sparse (Zhu & Gupta, 2017)	73.2 [-1.7]	91.5 [-0.9]	70.3 [-4.6]	90.0 [-2.4]		

Importance of training-time structural exploration and non-existence of "lottery tickets"







- Furthermore, we investigated whether the sparse network structures our method discovered were **"winning lottery tickets"** (Frankle & Carbin, 2018).
- We found that **neither the connectivity nor the weight initialization** could explain the superior generalization.
- Instead, **simultaneous structural exploration and parameter optimization** is indispensable for reaching the best generalization performance.
- Finally, we found that **network structure converges faster than the network parameters**, suggesting that parameter reallocation need not be active during the entire course of training.

Implications

- We showed that **compact, sparse deep convolutional networks can be effectively trained directly** under a strict, low memory footprint.
- We showed that **sparse networks generalize better**, i.e. in order to achieve **the best accuracy under a strict memory budget**, it is **necessary to use part of the budget to describe connectivity**, rather than spending it all on dense weights.

Frankle and Carbin. **The lottery ticket hypothesis: finding sparse, trainable neural networks.** arXiv:1803.03635 (2018)
Bellec, Kappel, Maass and Legenstein. **Deep rewiring: Training very sparse deep networks.** arXiv:1711.05136 (2017)
Mocanu, Mocanu, Stone, Nguyen, Gibescu and Liotta. **Scalable training of artificial neural networks with adaptive sparse connectivity inspired by network science.** Nature communications (2018) 9:2383.
Zhu and Gupta. **To prune, or not to prune: exploring the efficacy of pruning for model compression.** arXiv:1710.01878 (2017)

arxiv.org