Overview

- Existing frameworks (PyTorch, TensorFlow) focus on intra-operator parallelization.
- Only utilizing intra-operator parallelism suffers from device under-utilization problem, especially for small op & power.
- Therefore, we propose IOS — a dynamic programming algorithm scheduling inter-operator parallelization of CNN models.

Explore More Schedules is Important

- **Sequential Schedule**: the default choice for most frameworks, but leads to insufficient utilization as only one operator at a stage.
- **Wavefront Schedule**: a greedy method that execute all available operators stage by stage. It is sub-optimal due to unbalanced schedule.
- **IOS Schedule (ours)**: explores schedule space exhaustively, balances the computation in each stage, and best utilizes the hardware.

Inter-Operator Scheduler

- Latency $S_0 = \min \{ \text{Latency}[S] \}$
- $S$ is the ops to be scheduled
- $S'$ is a candidate for last stage of $S$
- $S'$ can be a last stage of $S$ if $S'$ is the ops to be scheduled
- $S'$ is the last stage candidate
- $S''$: NOT Last stage candidate

The time complexity of the dynamic programming is: $O(n^d)$

- $n$: number of operators
- $d$: maximum number of concurrent operators

Incorporate Schedule Specialization

- **Specialization for Batch Sizes**: consistent improvement for larger batch sizes
- **Specialization for Devices**: orthogonal and can be combined to further boost the performance

More Active Warps

- **IOS Accelerates Inference**
- **Comparison of cuDNN-based Frameworks**
- **Comparison of Different Schedules**

Schedule Specialization

- **TVM-AutoTune** vs. **IOS**: AutoTVM and IOS are orthogonal and can be combined to further boost the performance