Spira: Exploiting Voxel Data Structural Properties for Efficient Sparse Convolution in Point Cloud Networks

Dionysios Adamopoulos [1,2] Anastasia Poulopoulou [2] Georgios Goumas [2] Christina Giannoula [1]

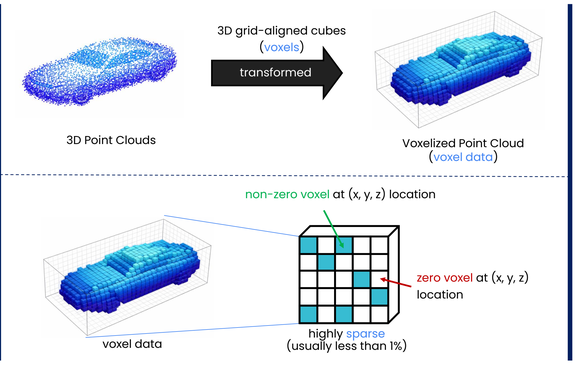
[1] Max Planck Institute for Software Systems [2] National Technical University of Athens

GitHub Paper

1. Summary

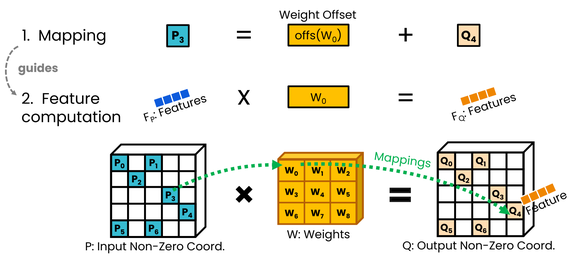
- Sparse Convolution (SpC) is a key operator in point cloud networks used in autonomous driving, robotics, and AR/VR.
- SpC operates on voxel data that have inherent structural properties; however, existing works do not exploit them.
- **Spira** is the first SpC engine that intelligently leverages structural properties of voxel data, unlocking significant performance gains.
- Spira improves inference performance by 1.68x on average across six GPU architectures.

2. Background on Voxel Data

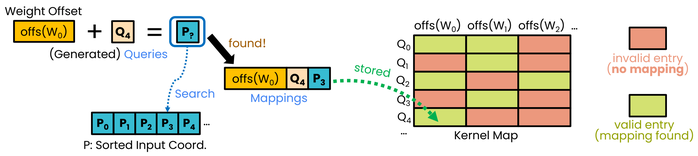


3. Background on Sparse Convolution

- Sparse Convolution (SpC) preserves the sparsity pattern of the input in the output.
- SpC computes the output features in 2 steps:



1. Mapping Step: Creates the mappings between the input, weight offsets and output non-zero voxels and stores them in a Kernel Map



2. Feature Computation Step: There are two different execution dataflows to process the kernel map and produce the output features:

(A) Output-Stationary: performs better when the kernel map has large #valid entries

(B) Weight-Stationary: performs better when the kernel map has small #valid entries

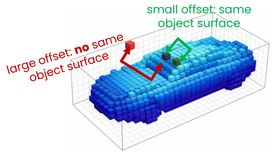
4. Motivation: Voxel Data Properties

We identify three important properties on real-world voxel data:

Integer Property Voxel coordinates are *integer-valued.*

Bounded Property Voxel coordinates are *spatially constrained* due to technological sensor limits.

Neighboring Property Neighboring voxel coordinates of the same object surface *are likely* to exist in *small* offset displacements.

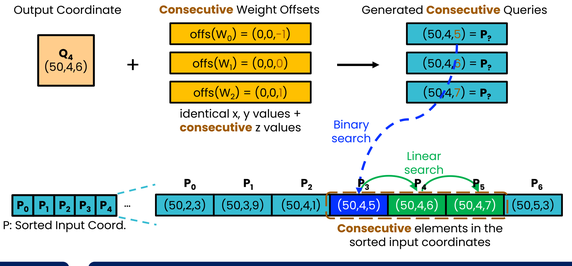


Existing SpC engines [e.g., TorchSparse++'23, Minuet'24] do not leverage voxel data properties

5. Spira Design

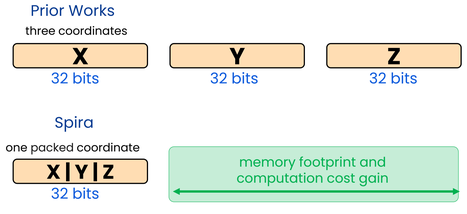
One-Shot Z-Delta Search in Mapping

- We leverage the **Integer Property**
- Key Observation I: Consecutive weight offsets in the z-axis produce consecutive queries.
- Key Observation II: Consecutive queries can only be found in consecutive elements in the sorted input coordinate array.
- Spira executes the first query via a binary search and resolves the remaining consecutive queries via a localized linear search.



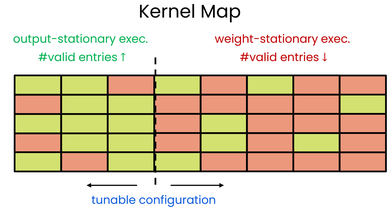
Packed-Native Mapping Step

- We leverage the **Integer Property** + **Bounded Property**
- Key Observation III: Voxel coordinates require less bits in the x-, y-, z- axis.
- Spira packs all coordinates into a single integer number.
- We pack only once at the network start; perform mapping steps as packed-native.



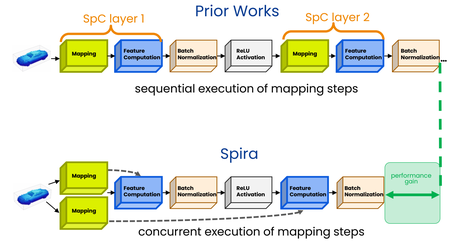
Adaptive Hybrid Dataflow Execution

- We leverage the **Neighboring Property**
- Key Observation IV: Kernel map columns of weight offsets that are associated with smaller offset displacements have larger #valid entries.
- Different weight offsets in kernel map can be processed with either output- or weight-stationary dataflow execution.
- The configuration is tunable, allowing a best-fit execution.



Network-Wide Mapping

- Key Observation V: Mapping *steps have* ***no*** *true dependencies* neither with feature computation steps nor between them.
- Spira executes all mapping steps concurrently at network start and significantly improves GPU SM utilization.

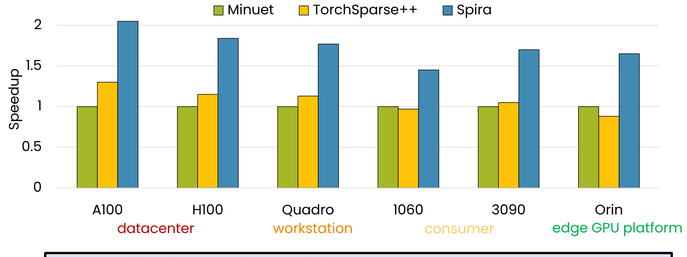


6. Methodology

- 3x Point Cloud Networks: ResNet, Large ResNet, UNet
- 3x Point Cloud Datasets: Waymo, SemanticKITTI, ScanNet
- 6x GPUs:
  - A100, H100: *datacenter*
  - Quadro RTX 5000: *workstation*
  - GTX 1060, RTX 3090: *consumer*
  - Jetson Orin AGX: *edge GPU platform*
- Comparison Points
  - **TorchSparse++** [Tang et al., MICRO'23]
  - **Minuet** [Yang et al., EuroSys'24]

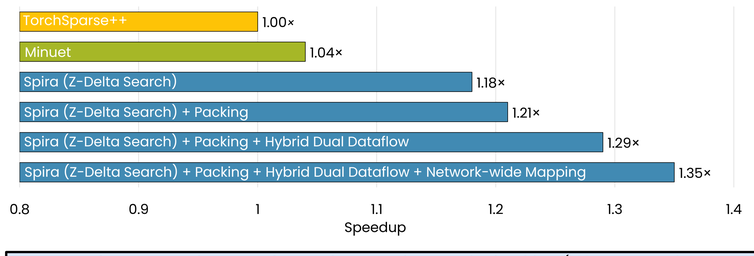
7. Evaluation

Averaged inference performance across 3 networks and 3 datasets



Spira significantly outperforms existing works by 1.68× averaged across six GPUs

Performance benefits of Spira's key ideas in UNet



Largest performance gain comes from Spira's Z-Delta search algorithm (1.18× over TorchSparse++)

Officially artifact evaluated as available, functional and reproducible.