**Generative Surrogates for Fast Simulation: TPC Case**

**MPD TPC tracker**
- Main tracker in the central barrel
- PID provided through $dE/dx$
- Computation bottleneck in the electron drift simulation \(\Rightarrow\) **Fast simulation approach required**

**Fast simulation with a neural network**
- TPC readout: 95,232 sensitive pads x 310 time buckets per bunch crossing
  - \(\approx 30M\) numbers to generate
- **Factorize the problem:**
  - Split all tracks into small segments contributing to individual pad rows
  - Contributions localized in space and time
    - 8 pads in a pad row
    - 16 time buckets
  - **Assume additive contributions from the segments**
  - **Responses conditioned on segment parameters**
    - Location & direction
    - Plan to incorporate other characteristics (particle type, momentum, etc.)

**Validation of the model**
- **At raw response level:**
  - Describe responses with 6 parameters:
    - Pad/time barycenters (2), widths (2) and covariance (1), integrated amplitude (1)
  - Compare distributions for each of these parameters between GAN and detailed simulation
  - Comparison as a function of track segment characteristics
- **At reconstruction level:**
  - Coordinate resolution, reconstruction efficiencies, and other...
  - **Very good agreement**
  - Few inconsistencies well understood and will be improved in the next iteration of model development

**Production pipeline**
- **Goal:** make an automated solution
- **Various simulation configurations may arise**
- **Requirements**
  - Automated training
    - Training data generation
    - Model training, evaluation and selection
  - **Model library**
    - Database for storage and prompt model retrieval
  - **Design choices (preliminary)**
    - Store models in ONNX format, ONNXruntime for inference
  - **Airflow** to manage the workflow
    - Pipelines described as Directed Acyclic Graphs (DAGs)
    - Tasks and dependencies defined in Python
    - Then scheduled and executed by Airflow
  - **MLflow** for model library
    - Use the “Model Registry” component
      - Model versioning and tagging
      - Web interface
      - PEP8 API to download a model for inference
    - Customizable model description in the web interface
      - Can be modified through API (e.g., automatically attach validation results to the model description)

**Alternative: generating the 6 parameters directly**
- **The 6 low-level parameters contain most of the information important for reconstruction:**
  - Barycenters \(\leftrightarrow\) cluster coordinates
  - Widths and covariance \(\leftrightarrow\) two-track resolution
  - Amplitude \(\leftrightarrow\) $dE/dx$
- **Idea:** generate just the 6 parameters, then build a 8x16 response matrix deterministically (discretized gaussian)
  - Gaussian discretization introduces bias
    - Can correct for it automatically by plugging biased generator responses to the discriminator
- **Motivation:** generating in lower-dimensional space may be simpler (quicker and/or lead to better quality)
- **However, experiments show that it requires a lot of hyper-parameter tuning to get to a reasonable quality**
  - Do GANs “prefer” higher-dimensional representations?..