Status report on

DNNROI sigproc

Hokyeong Nam
Chung-Ang University

Time offsets from DepoFluxSplat

- (Previous) DepoTransform → (Current) DepoFluxSplat
- DepoTransform: grid start point shift was the solution
- DepoFluxSplat: rasterize the depo to a grid, but still correcting by grid start point shift, which gives double correction

```
292             // Depo is at response plane.  Find its time at the collection
293             // plane assuming it were to continue along a uniform field.
294             // After this, all times are nominal up until we add arbitrary
295             // time offsets at output.
296             const double nominal_depo_time = depo->time() + m_origin / m_speed;
```

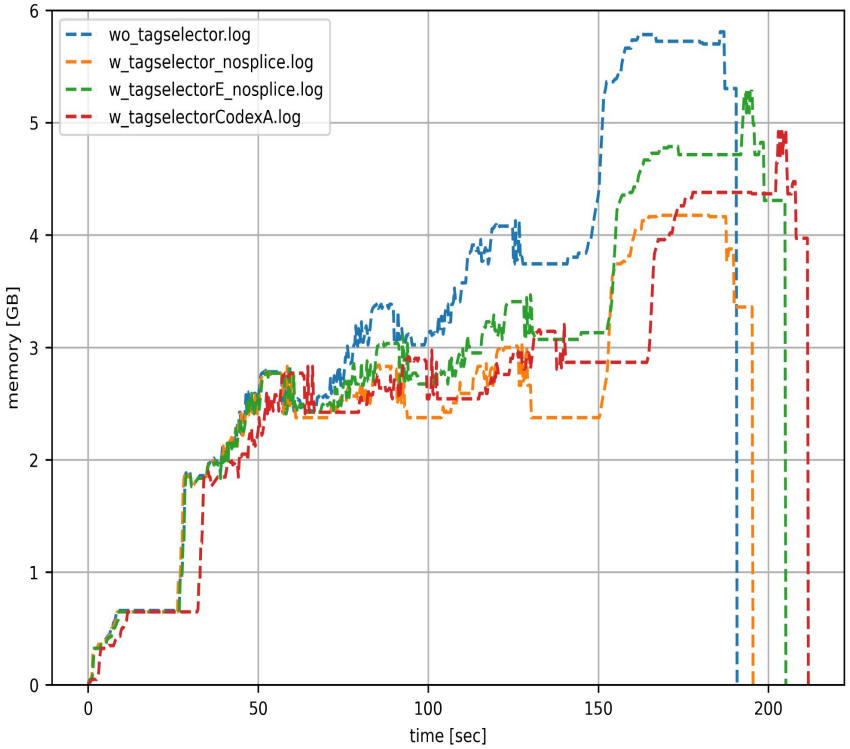
```
========== Peak Time Comparison (single wire) ==========
wire = 9390
DNN SP     peak time mu = 3781.39
Deposplat  peak time mu = 3792.82
delta_t (DNN - Deposplat) = -11.43 ticks
=======================================================
```

- Remaining time offset of ~ 11 ticks in PDVD

```
========== Peak Time Comparison (single wire) ==========
wire = 3850
DNN SP     peak time mu = 4418.11
Deposplat  peak time mu = 4413.96
delta_t (DNN - Deposplat) = 4.15 ticks
=======================================================
```

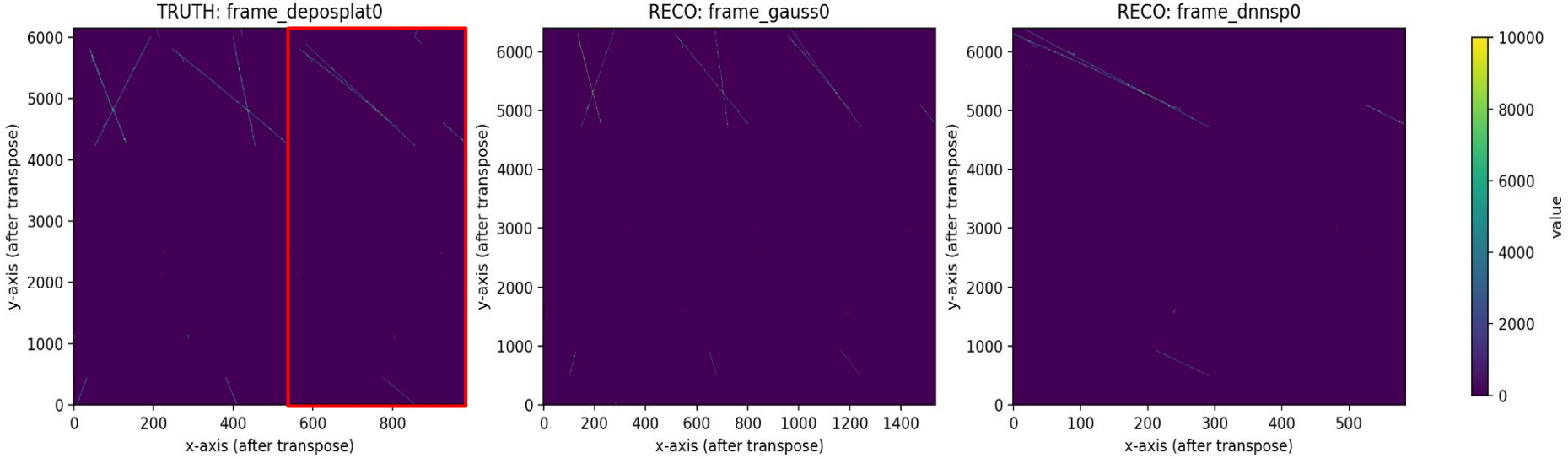
- Remaining time offset of ~ 4 ticks in PDHD

TagSelector for spliced pipeline



- Blue shows the baseline
- Orange shows the result with TagSelector, But:
  - dnnsp results exist in magnify root file
  - no dnnsp results in artroot file
- By re-arranging the data processing pipeline, I got Green and Red that dnnsp results contained in the final

PDVD sample production




```
frame_dnnsp0              Dataset {584, 6400}
frame_gauss0              Dataset {1536, 6400}
frame_loose_lf0           Dataset {1536, 6400}
frame_mp2_roi0            Dataset {1536, 6400}
frame_mp3_roi0            Dataset {1536, 6400}
```

```
channels_deposplat0       Dataset {557}
frame_deposplat0          Dataset {557, 3474}
tickinfo_deposplat0       Dataset {3}
```

- CRU, 1536 channels = 476 + 476 + 584
- Are we going to train based on CRU?

- Different {channels, ticks} compared to reco
- Previously resolved in PDHD sample production

Back Up

TagSelector for spliced pipeline

sigproc/inc/WireCellSigProc/TagSelector.h

```
+ /**
+    Make a new output frame with a set of traces selected from the
+    input based on trace tags.
+  */
+
+ #ifndef WIRECELLSIGPROC_TAGSELECTOR
+ #define WIRECELLSIGPROC_TAGSELECTOR
+
+ #include "WireCellIface/IFrameFilter.h"
+ #include "WireCellIface/IConfigurable.h"
+ #include "WireCellAux/Logger.h"
+
+ #include <string>
+ #include <vector>
+
+ namespace WireCell {
+     namespace SigProc {
+
+         class TagSelector : public Aux::Logger,
+                             public IFrameFilter,
+                             public IConfigurable
+         {
+           public:
+             TagSelector();
+             virtual ~TagSelector();
+
+             /// IFrameFilter interface.
+             virtual bool operator()(const input_pointer& in, output_pointer& out);
+
+             /// IConfigurable interface.
+             virtual void configure(const WireCell::Configuration& config);
+             virtual WireCell::Configuration default_configuration() const;
+
+           private:
+             std::vector<std::string> m_tags;
+             int m_count{0};
+         };
+     }  // namespace SigProc
+ }  // namespace WireCell
+
+ #endif
```

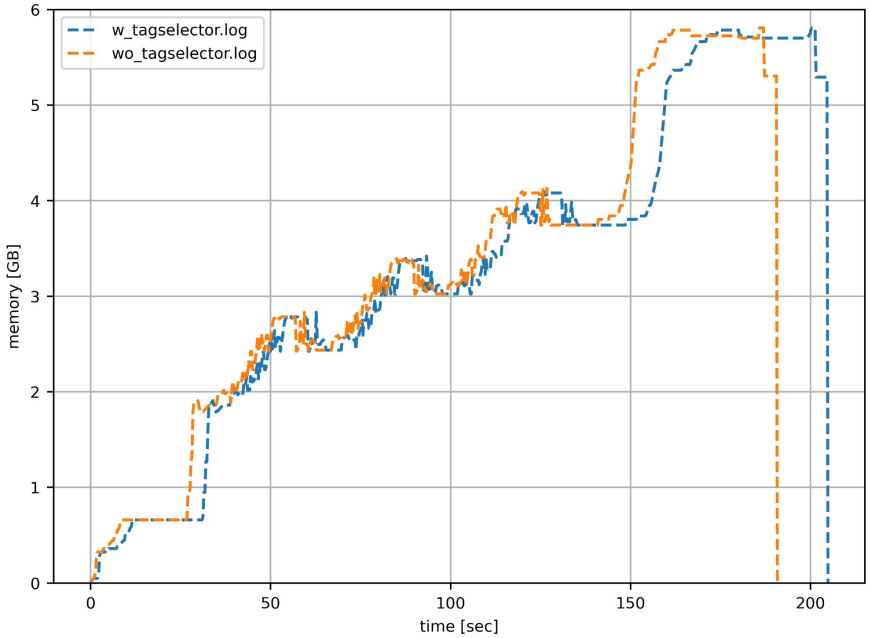
sigproc/src/TagSelector.cxx

```
+ #include "WireCellSigProc/TagSelector.h"
+ #include "WireCellAux/SimpleFrame.h"
+ #include "WireCellAux/FrameTools.h"
+
+ #include "WireCellUtil/NamedFactory.h"
+
+ #include <sstream>
+ #include <unordered_map>
+
+ WIRECELL_FACTORY(TagSelector, WireCell::SigProc::TagSelector, WireCell::IFrameFilter, WireCell::IConfigurable)
+
+ using namespace WireCell;
+ using namespace WireCell::SigProc;
+
+ TagSelector::TagSelector()
+     : Aux::Logger("TagSelector", "glue")
+ {
+ }
+
+ TagSelector::~TagSelector() {}
+
+ WireCell::Configuration TagSelector::default_configuration() const
+ {
+     Configuration cfg;
+
+     /// Only traces with these tags will be in the output.
+     cfg["tags"] = Json::arrayValue;
+
+     return cfg;
+ }
+
+ void TagSelector::configure(const WireCell::Configuration& cfg)
+ {
+     auto jtags = cfg["tags"];
+     int ntags = jtags.size();
+     m_tags.clear();
+     m_tags.resize(ntags);
+     for (int ind = 0; ind < ntags; ++ind) {
+         m_tags[ind] = jtags[ind].asString();
+     }
+ }
+
+ bool TagSelector::operator()(const input_pointer& in, output_pointer& out)
+ {
+     out = nullptr;
+     if (!in) {
+         log->debug("see EOS at call={}", m_count);
+         ++m_count;
+         return true;  // eos
+     }
+
+     if (m_tags.empty()) {
+         log->warn("TagSelector configured with no tags, passing through input frame");
+         out = in;
+         return true;
+     }
```

```
404 local sp_tag_selector = g.pnode({
405   type: 'TagSelector',
406   name: 'sp_tag_selector',
407   data: {
408     tags: [
409       tag
410       for n in std.range(0, std.length(tools.anodes) - 1)
411       for tag in ['gauss%d' % n, 'wiener%d' % n]
412     ],
413   },
414 }, nin=1, nout=1);
415 local outgr = g.pipeline([ofanin, sp_tag_selector, outretagger, wcls_output.sp_signals, osink]);
416 // local outgr = g.pipeline([ofanin, outretagger, osink]);
417
418 local edge_selector(e) = std.startsWith(e.tail.node, "OmnibusSigProc:");
419 local fanout_factory(n,e) = { type:'FrameFanout', name:"splice%d"%n, data:{multiplicity: 2} }; // "2-wire" splice
420
421 local spliced_graph =
422   if save_tradsp then
423     g.splice(graph, outgr, edge_selector, fanout_factory)
424   else
425     graph;
```

- To prevent overmemory usage by spliced data pipeline, we need a filter so that only gauss/weiner results will be saved
- A new function named TagSelector is developed and tested

TagSelector for spliced pipeline



- The memory consumption measured using activity logger based on top command
- The TagSelector didn’t work as it is designed for

Plans

- Debugging TagSelector and make it work as expected

- Make pull request
  - Wire-Cell Toolkit: TagSelector
  - dunereco: updated jsonnet configruation files for PDHD and PDVD

- PDVD tasks
  - Working with Haiwang to solve the time offset issue at its source
  - Running a small PDVD sample production for the DNN training