ATLAS Pileup and Overlay Simulation

Andy Haas (NYU) on behalf of the ATLAS Collaboration

LHC Detector Simulations, CERN
March 18-19, 2014

https://indico.cern.ch/event/279530/
Introduction

- In addition to hard interaction:
  - Pileup from other collisions in current and surrounding bunch crossings
  - Cosmics, Beam-gas, Beam-halo, Cavern background, Detector noise, ...

- Overlay method used by BaBar, D0, ...
- Pileup MC method mostly used so far at ATLAS
  - Overlay being used for some studies, specialized analyses, and Heavy-Ion

Option 1: “Pileup MC” (current default)

Simulate all processes in MC* and mix together in proper ratios with realistic timing

Option 2: “Overlay”

Simulate only hard interaction in MC and overlay a “random” data event to include all backgrounds**

*Cosmics, beam-gas, beam-halo are small → not currently included in pileup MC

**Statistics of rare background events such as beam-halo will be very poor. Must trigger on a signature of the signal MC event to accurately model background rates.
Option 1: Simulating Pileup

(1) Run the event generation for “minbias” for single pp interactions → Pythia8, A2 tune, MSTW2008LO PDFs → Inelastic (non-diffractive) and single/double diffractive

(2) Run GEANT4 on each minbias event to simulate detector energy/time “HITS”

(3) Combine multiple (thousands!) of HITS events during digitization → Use representative #interactions per bunch crossing, shifted in time, to reproduce in-time and out-of-time pileup → Sample bunch spacing/pattern within the sensitive time window of ATLAS detectors [-800,800] ns

(4) Add model of detector noise separately

(5) Add cavern background separately (see talk by Jochen Meyer)
ATLAS Sensitive Time Window

-800 ns to 800 ns

MDT
LAr
CSC
Tile
RPC
TGC
TRT
SCT
Pixels
 BCM

Bunch Crossing -32
Bunch Crossing 0
Bunch Crossing 32

No effect on Trigger BC
Could effect Trigger BC
No effect on Trigger BC

25 ns tick
Simulating Pileup: Bunch Structure

Example of a pile-up model with fixed 50ns spacing between colliding BCIDs:

<table>
<thead>
<tr>
<th>Signal</th>
<th>Minbias</th>
<th>Cavern</th>
</tr>
</thead>
</table>

25ns tick ('bunch')

Filled BCs

- The pile-up/detector response is affected by the position of the triggering BCID in the bunch train (see later).
- Bunch structure modelling is included in the pile-up simulation.
  - Patterns can be up to 3564 elements in length and wrap-around if required.
  - Each triggering BCID is picked from the colliding BCIDs in the pattern, with a probability proportional to the relative luminosities of each bunch crossing.
Simulating Pileup: Variable (Bunch) Luminosity

- Well known that $<\mu>$ varies over time.

  Run 180164 (April 2011):
  peak $<\mu>$=7.6

  Run 190300 (October 2011):
  peak $<\mu>$=13.4

- $\mu$ can also vary greatly from BCID to BCID in data, as the plots above show.
- Both in-time and out-of-time pile-up effects are important.
- **Problem:**
  - Simulating samples at a fixed $<\mu>$ value makes it difficult to re-weight MC to data...
Impacts of Pileup

- Do not expect a significant impact on tracking, nor muons, nor even electrons and photons.
- But sizeable impact on jets (+ETmiss) and \( \tau \).
- LAr drift-time is \( \sim 500 \text{ ns} \) and out-of-time bunches have impact on measurement.
- Bipolar pulse shaping designed so that \( \langle ET \rangle \sim 0 \) for 25 ns bunch-spacing and uniform intensity per BX.
- Optimal performance will require correction per cell type in \( \eta \)-bins and as a function of luminosity to set average measured ET to \( \sim 0 \).
- Correction was applied for all 2012 data and MC...
Impacts of Pileup

- Energy in calorimeter in cone around electrons → higher at start of bunch train, before negative tails from out-of-time pileup contribute
- Modeled well by simulation, but also explicitly corrected for at the cell level during reconstruction for 2012 data and MC
Simulating Pileup: Digitization

- For each MC job, create a cache of minbias events in memory
  - Only read in / cache the parts of each event that are needed
    (e.g. discard HITS in silicon strips outside [-50,50] ns)

- Generating huge samples of minbias background is expensive!
  - 20M minbias events simulated for 2012 (10M “low-pt”, 10M “high-pt”)
    - “low-pt”: no AntiKt6Truth jets with $p_T > 35$ GeV
    - ~7 TB storage at each MC production site

- Reuse simulated minbias events across various MC samples
- Also reuse “low-pt” out-of-time minbias events within a MC sample
Simulating High Luminosity Pileup

• For High luminosities previous pile-up approach has issues...

• Consider a typical upgrade scenario:
  • 200 pp-collisions per colliding BCID
  • fixed 50ns spacing between colliding BCIDs
    ➔ ATLAS would be sensitive to 33 colliding BCIDs
      ➔ $33 \times \sim 200 \times 2 = O(13200)$ background events (minimum bias+cavern) required per single signal event!

• Having this many simulated events in memory at once is not feasible, so an alternative must be found...
Simulating High Luminosity Pileup

- The previous pile-up approach (AKA the “Algorithm” approach):
  - digitizes the information from all required bunch crossings for a given sub-detector before moving on to the next sub-detector.
  - Background event info cached to allow re-use.
- The “PileUpTools” approach:
  - provides one filled bunch crossing at a time to all sensitive sub-detectors.
  - Background events are read as required and discarded from memory after each filled bunch crossing is processed.
    - Sacrifice caching of background to save memory.
    - Resulting increase in I/O Time means an increased wall-clock time.
  - A single pile-up Athena Algorithm calls an Athena AlgTool for each sub-detector. The AlgTools know the time window for which they are sensitive to bunch crossings.
  - Digits/RDOs are produced from intermediate information cached locally by the sub-detector tools, after all filled bunch-crossings have been processed.
PileUpTools Memory Savings (32-bit)

i686-slc5-gcc43-opt

32 bit Algorithms (MC11 Code using MC10 inputs)
\( \frac{d<vmem>}{d<nMinbias>} = 26.4 \text{ MB/bkg event} \)

32 bit PileUpTools (MC11 Code using MC10 inputs)
\( \frac{d<vmem>}{d<nMinbias>} = 8.4 \text{ MB/bkg event} \)

Algorithm Approach exceeds the 32-bit addressable memory limit here.
Option 1.5: Simulation Using Embedding

- For specialized studies, such as $Z \rightarrow \tau \tau$ simulation, also perform *embedding* at reconstruction (not digitization) level

- remove muon tracks and simulated calorimeter energy from data
- replace by full-sim $Z \rightarrow \tau \tau$ decays, generated with Tauola
- re-run full event reconstruction: pile-up, jets and $E_{T}^{miss}$ from data
- normalization from MC prediction (trigger effect not simulated)
- validation with $\mu \rightarrow \mu$ embedding (data to data) and Alpgen MC

Constrain systematics on tau-ID and $Z \rightarrow \tau \tau$ kinematics / mass ...

Underlying event and $Z$ $p_T$ distribution from data

---

**Embedded $Z \rightarrow \tau \tau$ MC in data event**
Option 2: Overlay pileup (and noise) from data

1. Define data period to simulate and select “random” (RAW) zerobias data events, proportional to luminosity (details later...)
2. Simulate hard-scatter events (GEANT4) with conditions matching each selected data event (beamspot, alignments, dead modules, etc.)
   → Note: running GEANT4 on geometry with data alignments
3. Overlay each zerobias data event with matching GEANT4 event at the detector channel level, then digitize combined signals
4. Reconstruct the combined event as data
Zerobias datasets

- Trigger L1_ZB fires 1 turn after EM14 → proportional to bunch lumi
  - Prescaled to keep ~1 Hz in 2011, ~10 Hz in 2012 → ~65 M events
  - No zero-suppression: about 3MB/event

- Sample zerobias events from lumiblocks in the desired time-period to reproduce the instantaneous luminosity profile of L1_EM30 (account for changing prescales, etc.)

```
Zerobias RAW filelist

<table>
<thead>
<tr>
<th>Zerobias RAW filelist</th>
<th>“map” file</th>
<th>Output file (100 events)</th>
</tr>
</thead>
<tbody>
<tr>
<td>data12 8TeV...lbn00345.0001...RAW</td>
<td>6 wanted</td>
<td>Zerobias_001.RAW</td>
</tr>
<tr>
<td>data12 8TeV...lbn00345.0002...RAW</td>
<td>4 wanted</td>
<td></td>
</tr>
<tr>
<td>data12 8TeV...lbn00346.0001...RAW</td>
<td>1 wanted</td>
<td></td>
</tr>
<tr>
<td>data12 8TeV...lbn00347.0001...RAW</td>
<td>3 wanted</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>About ~60 files</td>
<td></td>
</tr>
</tbody>
</table>
```

Now works in MC production system!

This is just 1 job for 100 events. 500 jobs needed for a 50k dataset. 1000 50k datasets for 50M events.
Pileup Simulation vs. Data Overlay

- **Drawbacks compared to pileup simulation:**
  - Less accurate when combining overlapping background and signal on the same channel for some subdetectors (e.g. silicon)
    - Zerobias data is readout zero-suppressed for some detectors (e.g. pixels)
  - Background reconstructed with MC R-T relations (TRT, MDT), so slight resolution degradation for background tracks and muons
  - Potential GEANT4 geometry overlaps when using data alignments
  - Limited (but large) statistics of background data
  - Can't simulate future detector geometries (for some upgrade studies)
  - Don't have the background truth information – it's data!

- **Overlay advantages:**
  - Real pileup data events – no generator tuning
  - #vertices and inst. luminosity match data – no event weighting
  - Real mix of BCID variation, in-time/out-of-time pileup, satellite bunches
  - Real detector noise, occupancy – including cavern background
  - Real detector conditions (beamspot, dead channels, etc.)
  - Less CPU and memory need at high luminosity...
Pileup Simulation vs. Data Overlay

- Digitization of \( \approx \) thousands of added simulated HITS events grows \( \approx \) linearly with the number of pileup interactions
- Overlay only has to add \textit{one} data event, independent of luminosity!

- Digitization time now \( \approx \) small compared to GEANT4 time
- More critical later, especially for fast simulation methods
- Other methods (fast digi) also being explored (see talk by Robert Harrington)

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{chart.png}
\caption{CPU per event [ms] vs. \( <\mu> \) per filled bunch crossing}
\end{figure}
Overlay MC for pp Physics

- One example: “Search for Higgs decay to long-lived particles”
  - Reconstruct displaced vertex in muon system
  - Background from punch-through jets and cavern background hits
  - Must model cavern background accurately!
ATLAS Upgrade Studies Using Overlay

- Cavern background is nicely modeled in overlay
- Overlay *multiple* zerobias data events to simulate higher luminosity!
  - Noise is double-counted, but negligible in the muon system
  - Validated using 3 low-luminosity events compared to 3x luminosity data
- Saturation of MDT End-cap Inner tubes confirmed
  - Helped to motivate New Small Wheel Upgrade
Heavy Ion Simulation Using Overlay

- Almost no pileup in heavy-ion collisions, but..
  - Underlying event is huge, difficult and costly to simulate!
- Use (minbias) HI data, overlay hard parton interaction simulated at the same event vertex position
  - ~20M HI overlay MC events produced
  - Used for many HI results successfully ...

Measurement of the correlation of jets with high pT isolated prompt photons in lead-lead collisions at \( \sqrt{s} = 2.76 \) TeV
Conclusions

- ATLAS uses several methods for modeling pileup and other detector backgrounds in our simulation
  - Simulated Pythia8 minbias is currently used for most simulations
  - Embedding is used for specialized studies, e.g. for $Z \rightarrow \tau \tau$
  - Overlay is an alternate method, currently used for some performance studies, pp and HI physics analyses, and detector upgrade studies

- Working to improve accuracy and speed of all these methods
- Pileup will become increasingly important with larger inst. luminosity
- We must model pileup efficiently to simulate larger datasets

$H \rightarrow ZZ \rightarrow 4\mu @ 2.6e33$

$H \rightarrow ZZ \rightarrow 4\mu @ 2.6e34$
Pile-up Digitization: vmem breakdown

x86_64-slc5-gcc43-opt
$\langle \mu \rangle = 8.0$, Fixed 25ns bunch-spacing.

This contribution increases fastest with luminosity.

One approach to save memory under validation is to filter truth info in the background HITS files.
Impacts of Pileup

- Jet offsets from pile-up are modelled to <50%
- Remaining differences from BCID-to-BCID beam current variation were not modelled in 2011 MC

ATLAS Preliminary

Data

Monte Carlo

Jet Offset [GeV]

Distance from last empty bunch [ns]