

Preliminary Performance on INTT Hit Carryover

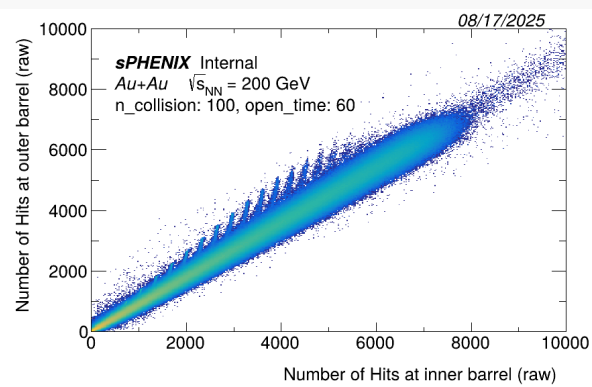
Ryotaro Koike
Kyoto University

Overview

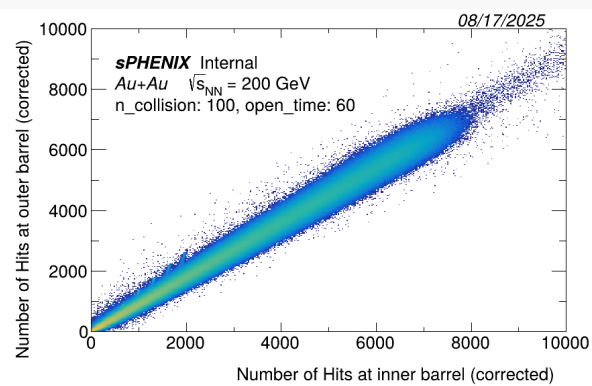
2

- I'd like to request an approval of performance preliminary for these plots which demonstrate the cause of INTT hit carryover and its offline recovery.

① before correction

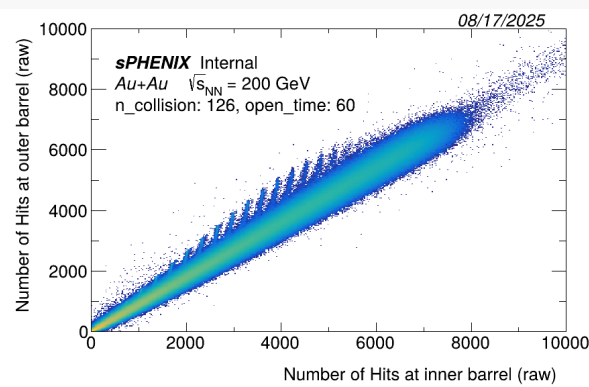


② after correction

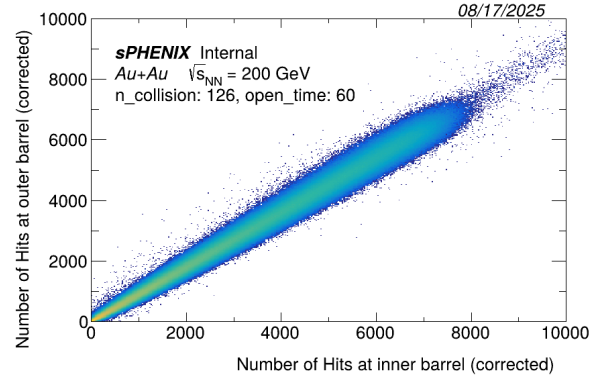


Previous default parameter

③ before

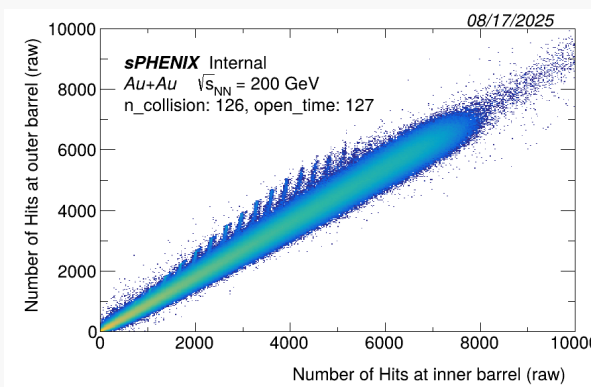


④ after

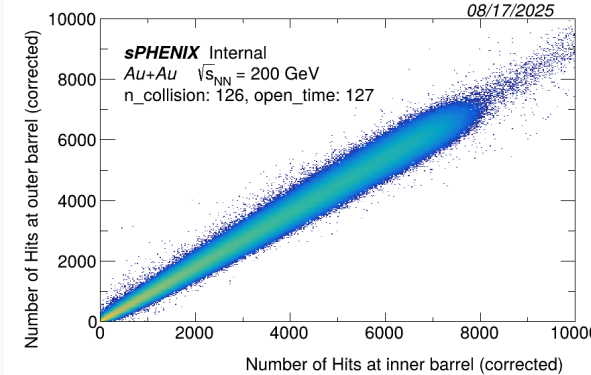


Different parameter

⑤ before

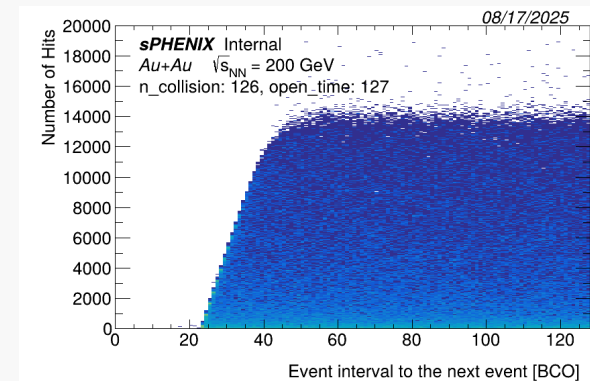


⑥ after



Current default parameter

⑦



- INTT special runs for hit carryover study

Table 1: List of runs used for the analysis.

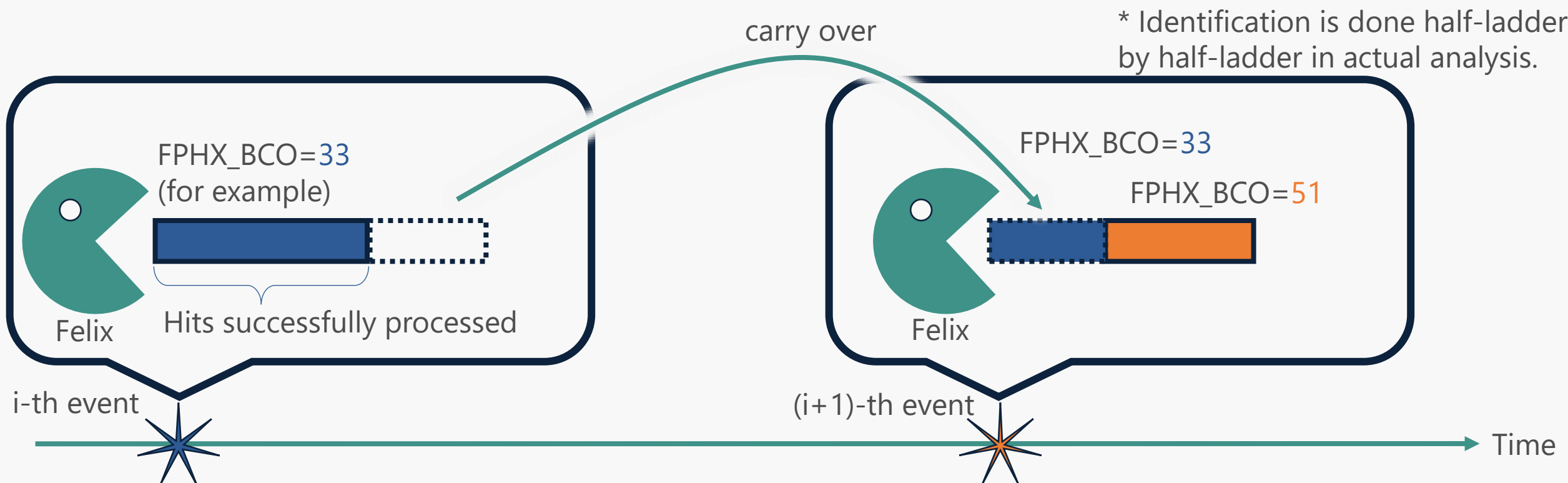
Run	n_collision [BCO]	open_time [BCO]
71345	100	60
71346	126	60
71347	126	127

- Run condition:
 - 111 x 111
 - Au+Au collision
 - INTT in the big partitoin
- Event selection
 - First 10 M events were used.
 - MBD z-vertex < 10 m cut was applied.
- Analysis note: <https://sphenix-invenio.sdcc.bnl.gov/records/933a9-mxv40>

Mechanism of hit carryover

4

- Felix carries over hits in i -th event that could not be processed until next trigger comes to the $(i+1)$ -th event.
- Felix starts the process of $(i+1)$ -th event with those carried-over hits.



The cause of INTT hit carryover

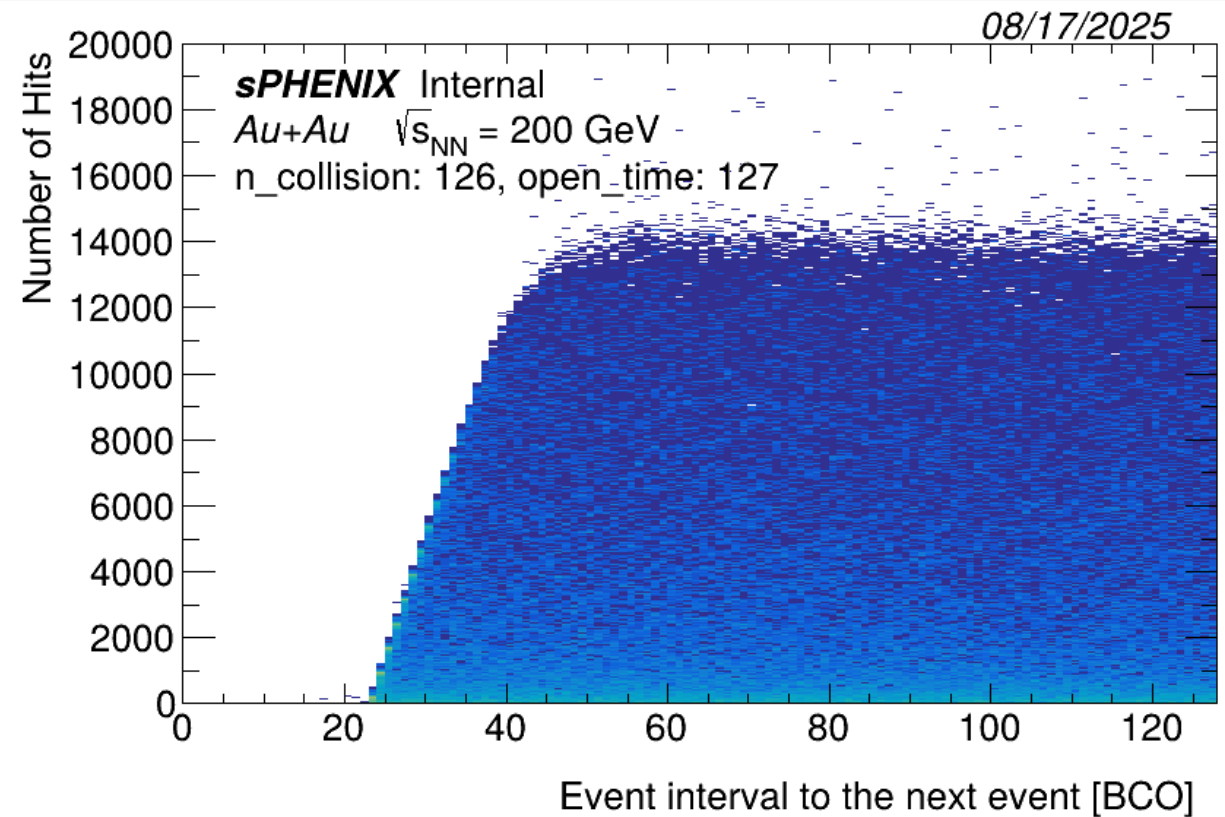
5

What this plot shows: Number of hits (successfully) recorded as a function of interval to the next event.

Multiplicity ($\sim$ the y-axis) is independent from the event interval (the x-axis).

Step-function-like upper limit exist due to a limitation in processing capability per BCO

⑦



- This plot is a proof that our understanding of the mechanism is right.

Correction to hit counts

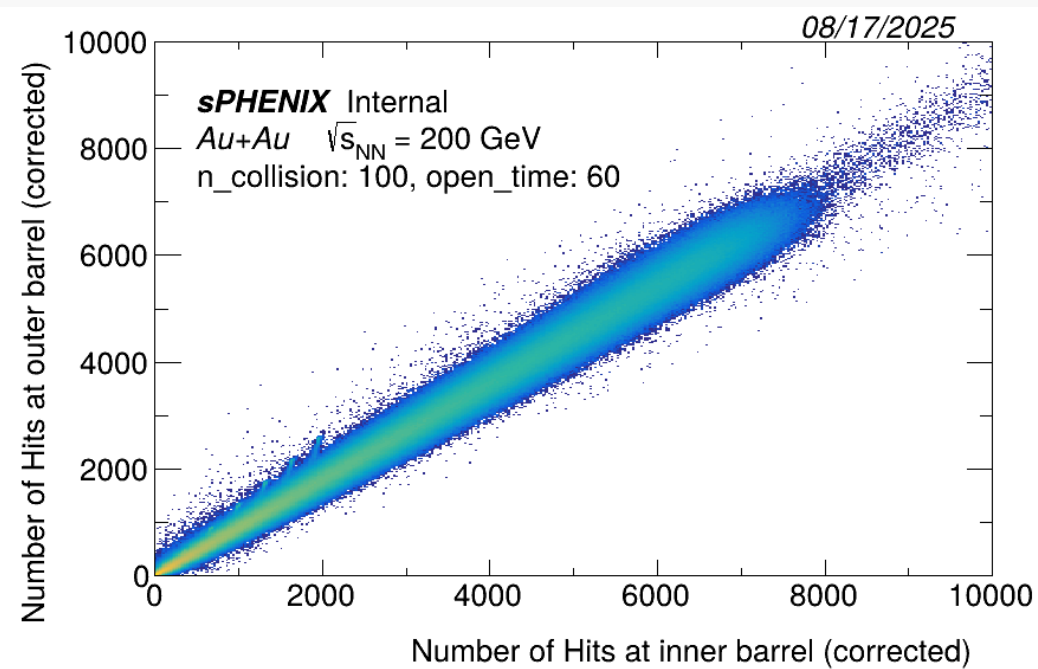
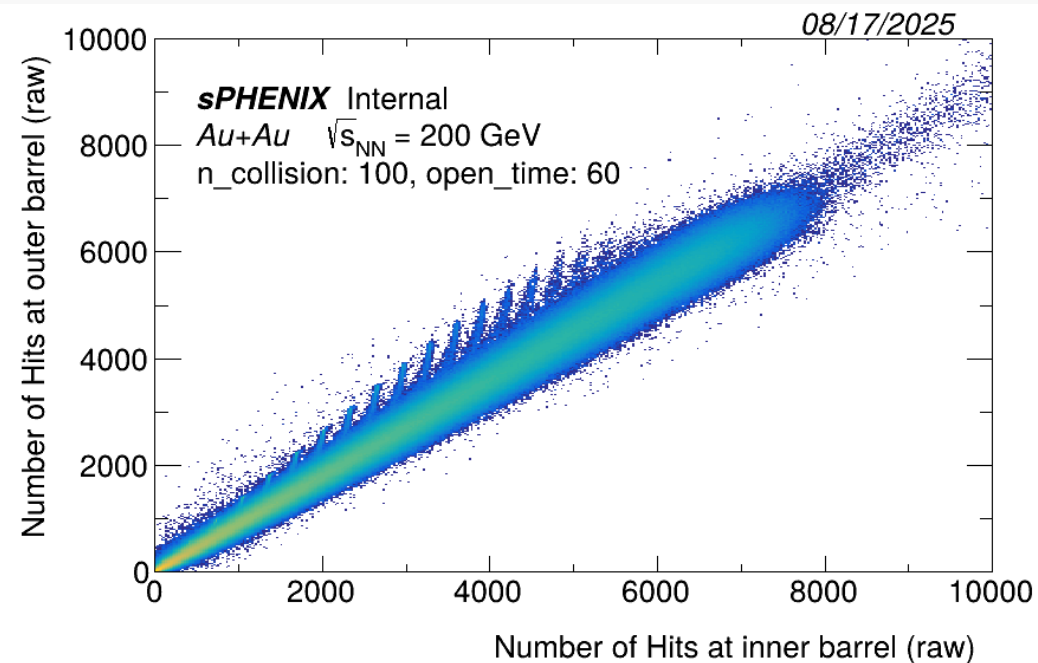
- We corrected the number of hits by the number of carried-over hits.
- Some branches remained in the previous default parameter setting.

① before correction

Table 1: List of runs used for the analysis.

Run	n_collision [BCO]	open_time [BCO]
71345	100	60
71346	126	60
71347	126	127

② after correction



Correction to hit counts

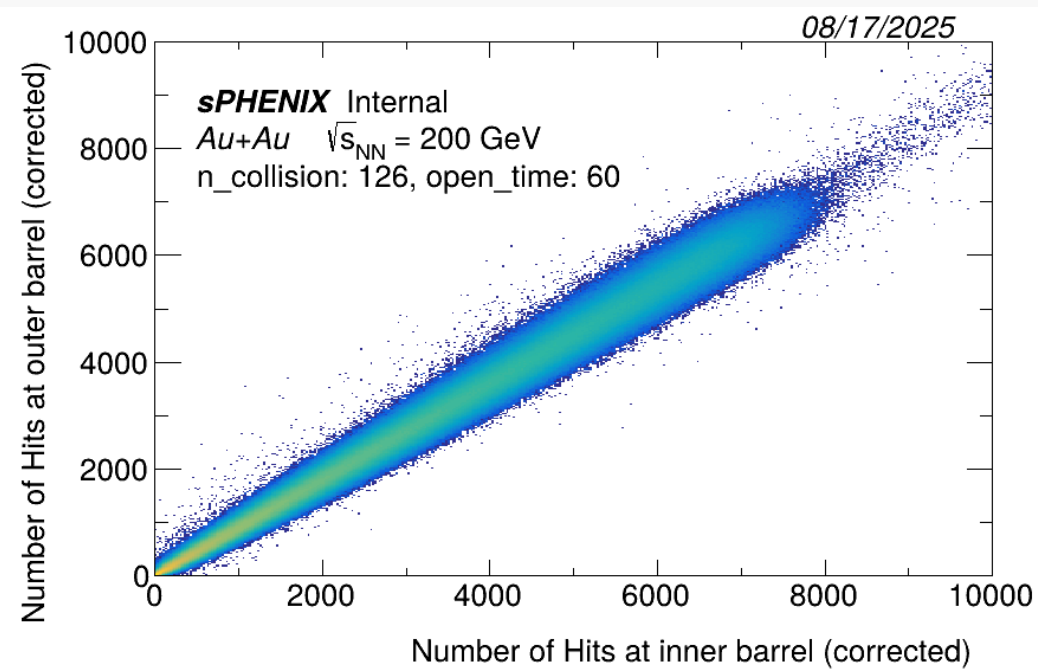
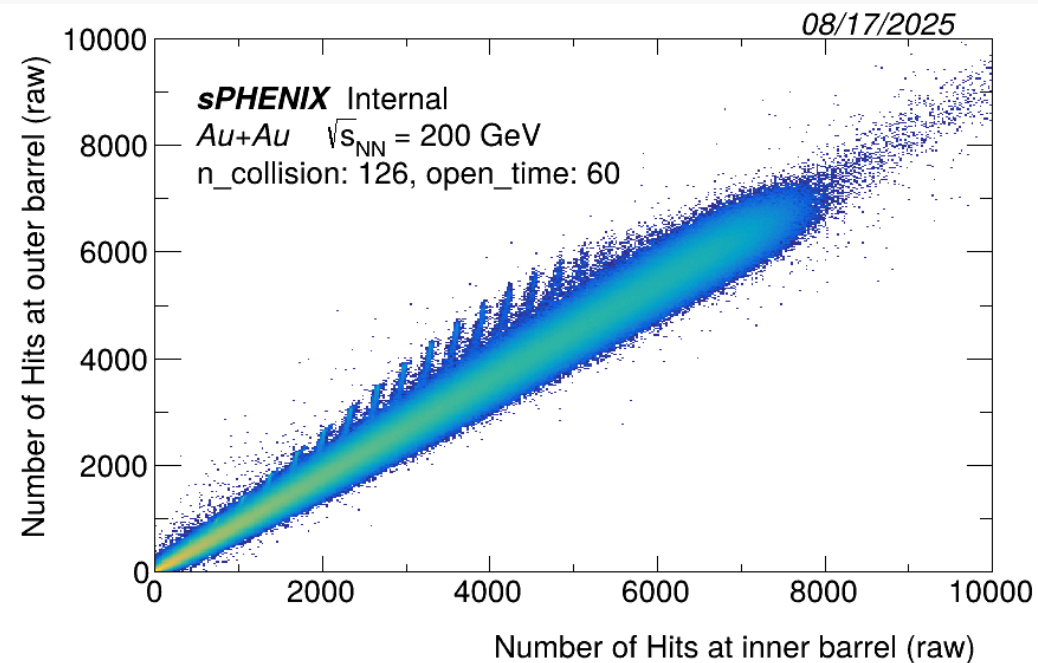
- All branches returned back to the normal position if we extend `n_collision`.

③ before correction

Table 1: List of runs used for the analysis.

Run	n_collision [BCO]	open_time [BCO]
71345	100	60
71346	126	60
71347	126	127

④ after correction



Correction to hit counts

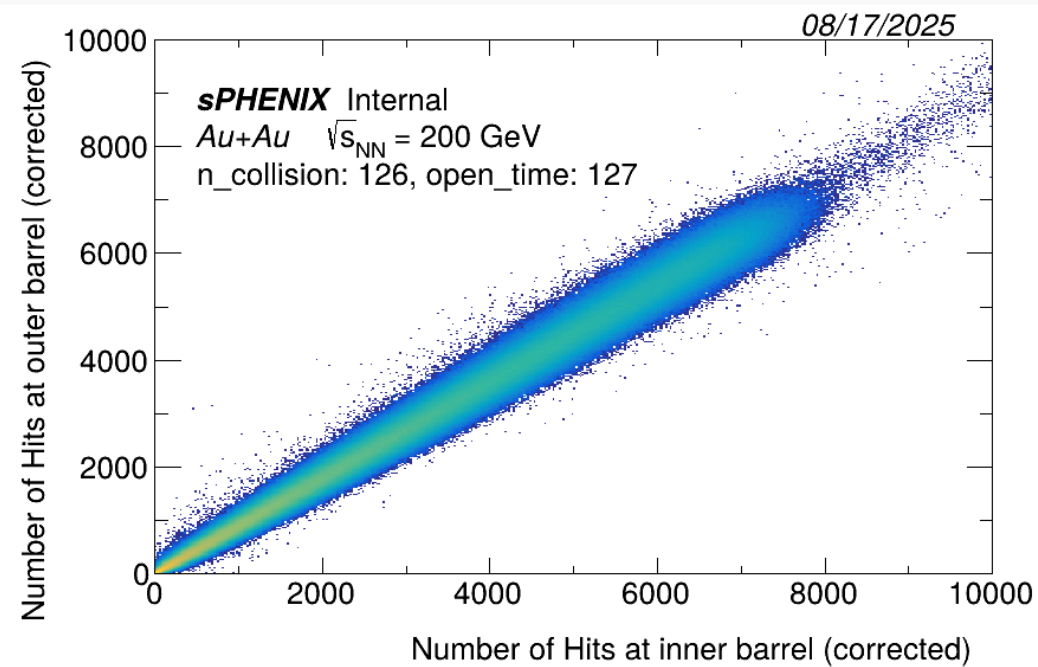
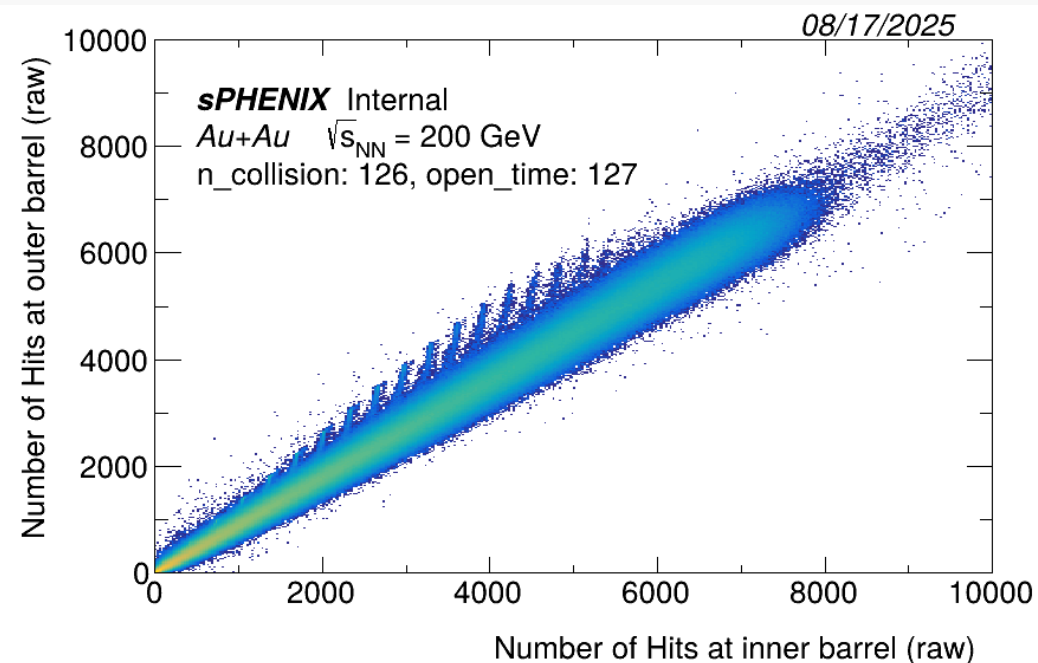
- All branches returned back to the normal position if we extend `n_collision`.

⑤ before correction

Table 1: List of runs used for the analysis.

Run	n_collision [BCO]	open_time [BCO]
71345	100	60
71346	126	60
71347	126	127

⑥ after correction

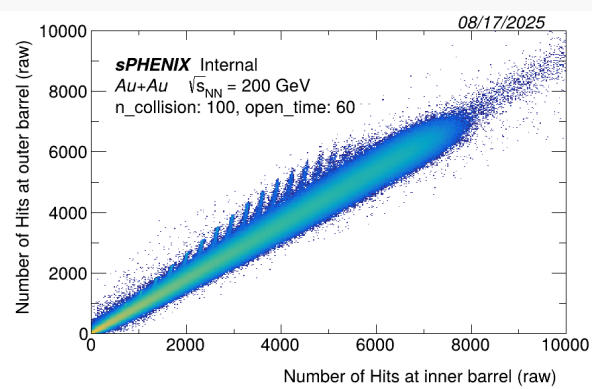


Overview

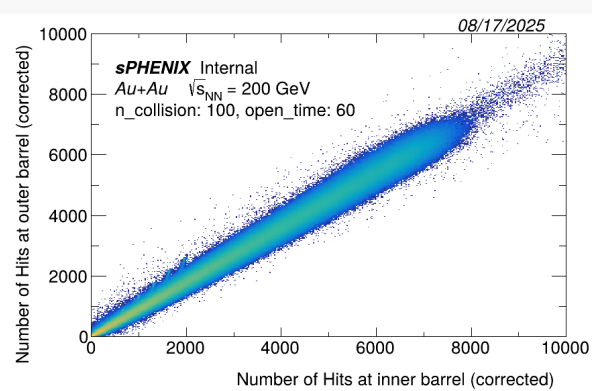
9

- I'd like to request an approval of performance preliminary for these plots which demonstrate the cause of INTT hit carryover and its offline recovery.

① before correction

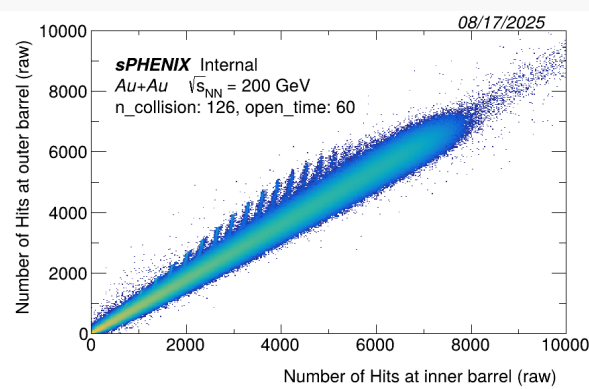


② after correction

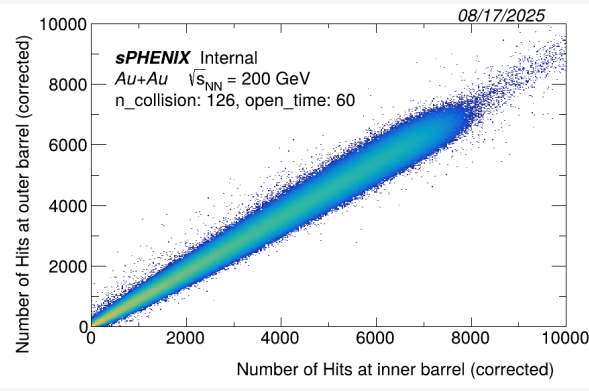


Previous default parameter

③ before

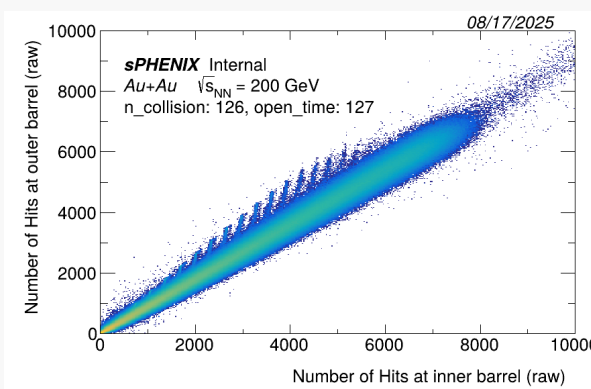


④ after

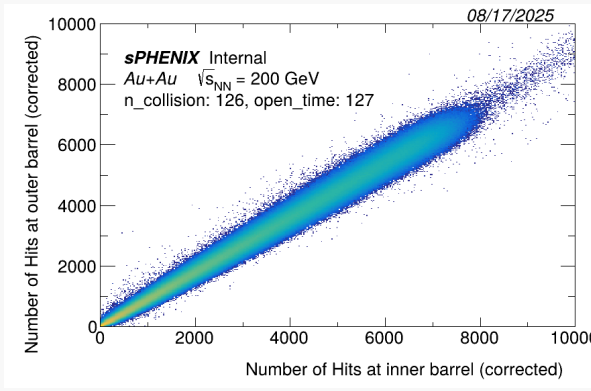


Different parameter

⑤ before

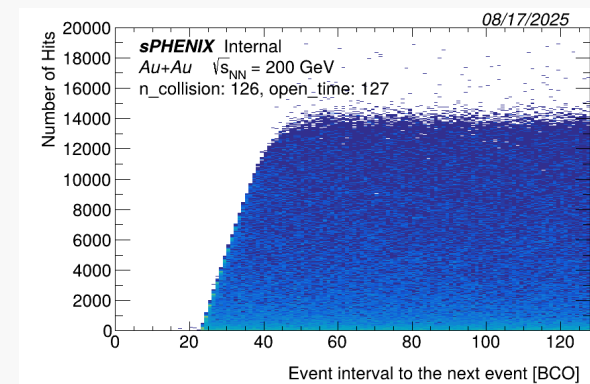


⑥ after



Current default parameter

⑦

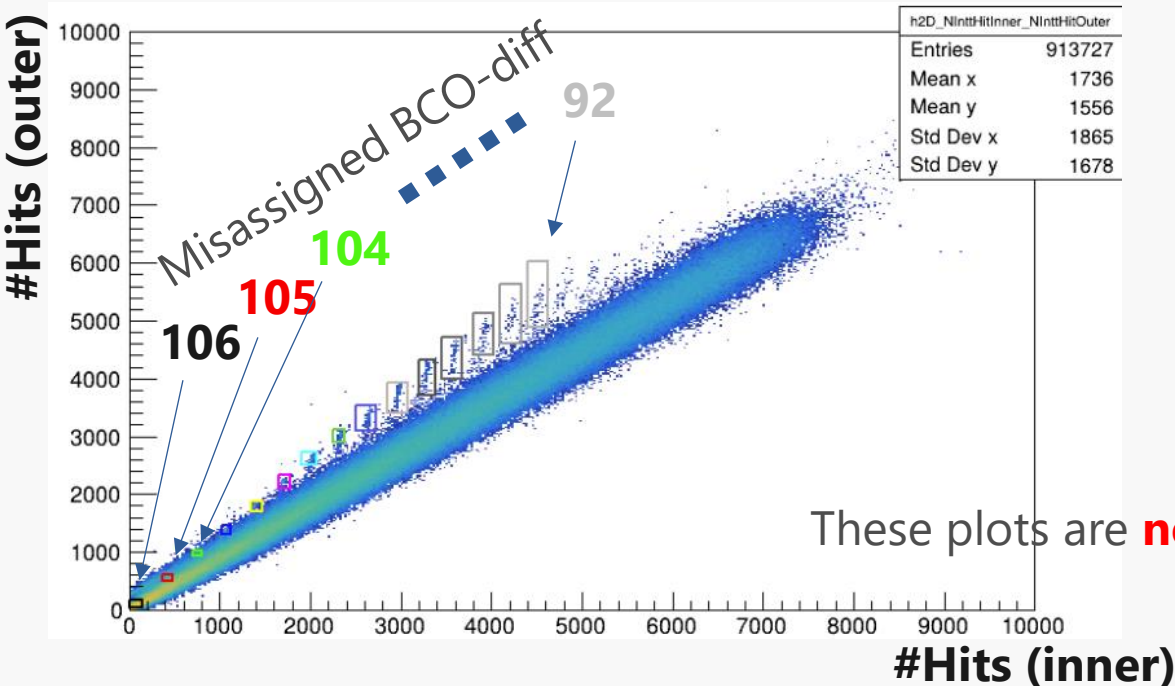


Backup

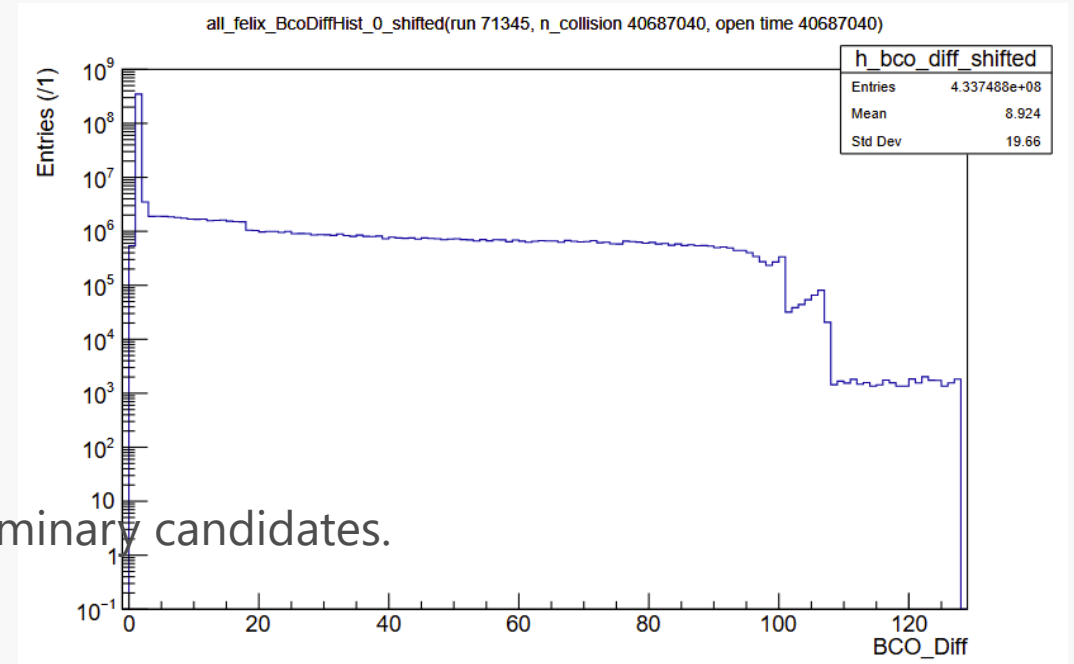
Reason we can push carried-over hits back now 11

- Extending n_collision to 126 from 100 was effective.
- Q: Why?
- A: Carried-over hits were rejected in the next event according to n_collision, since a wrong BCO_diff value was attached to those hits.

(misassigned BCO diff) = (fphx bco)_{carried-over hits from i th event} - (BCO full) _{$i+1$ th event}



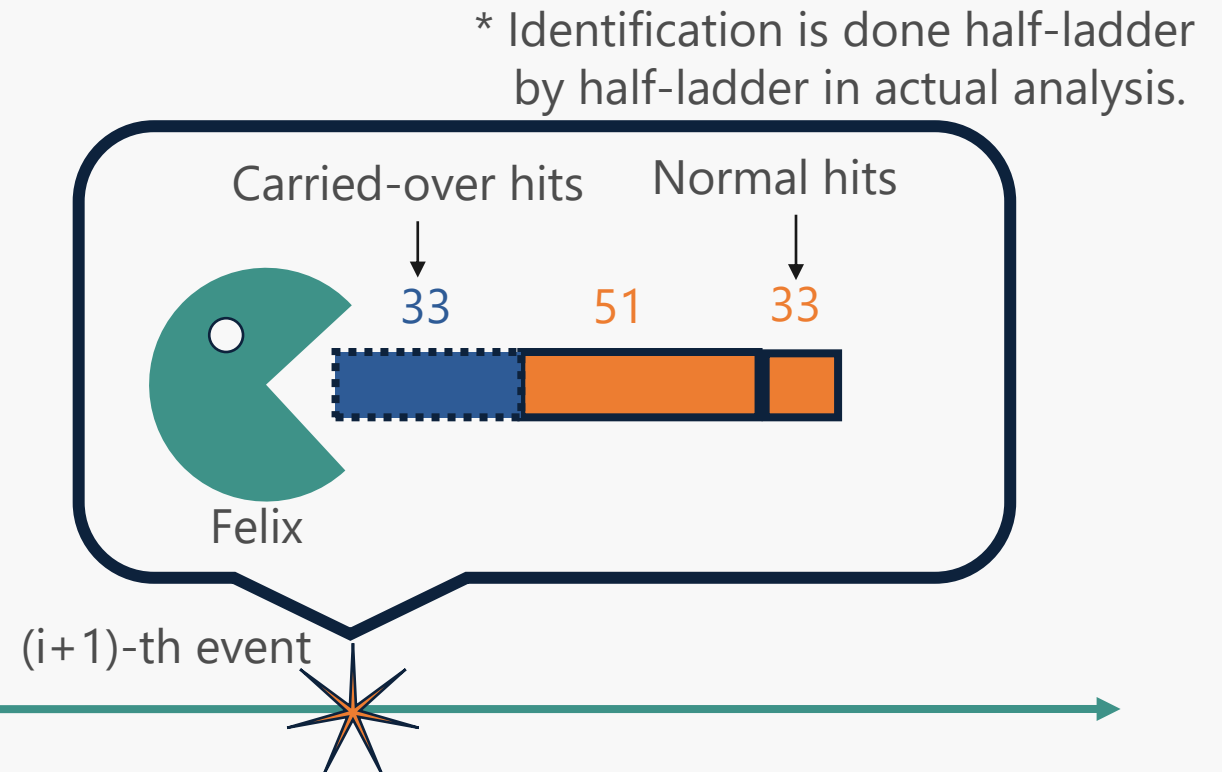
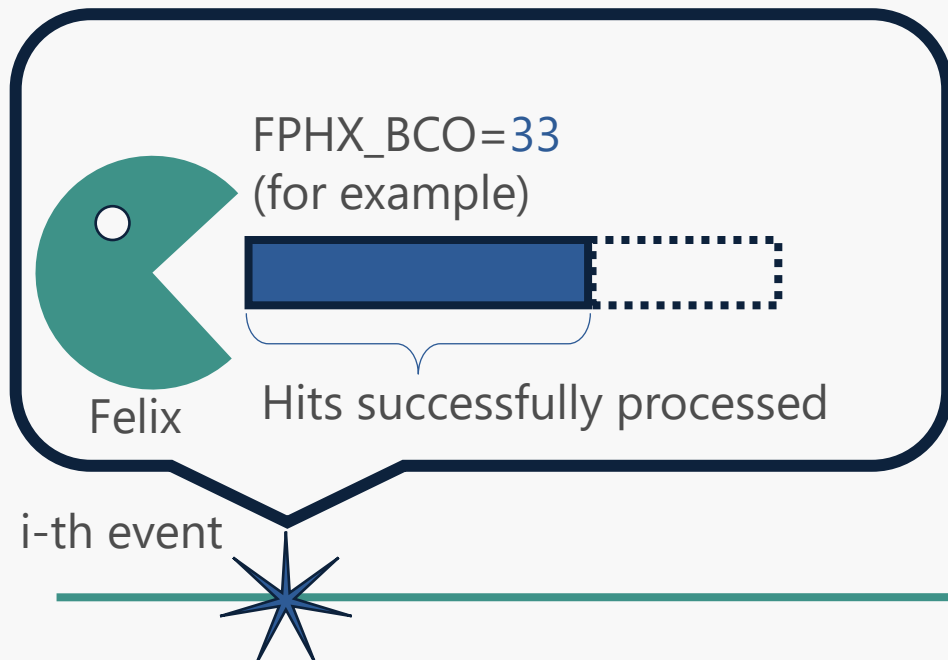
These plots are **not** preliminary candidates.



Identification of carried-over hits.

13

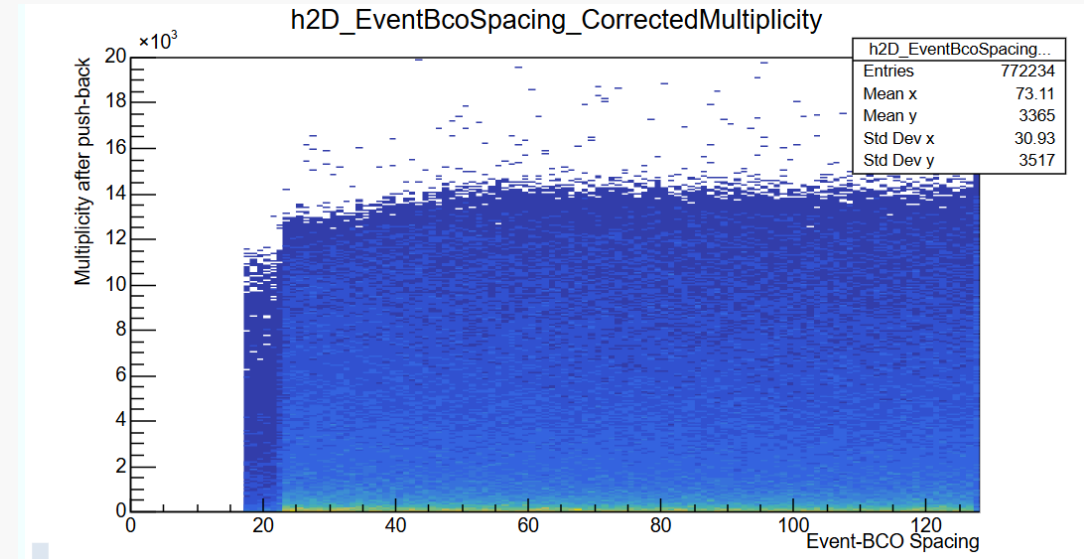
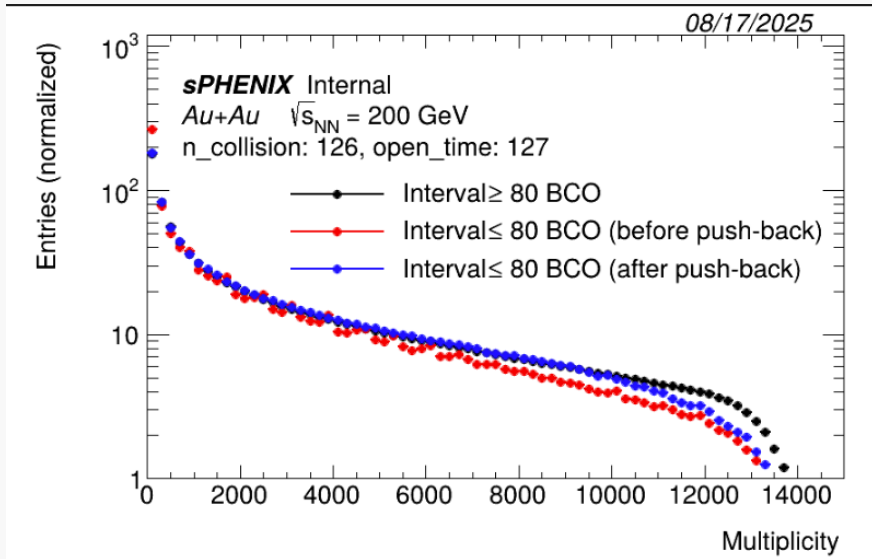
- A hit in $(i+1)$ -th event is considered as carried-over hit
 - if its fphx_bco is the same as the major hits of i -th event,
 - but only if no hits with different fphx_bco were recorded before that hit.



Difference between identification methods

14

First N hits



All hits that have
the same bco

