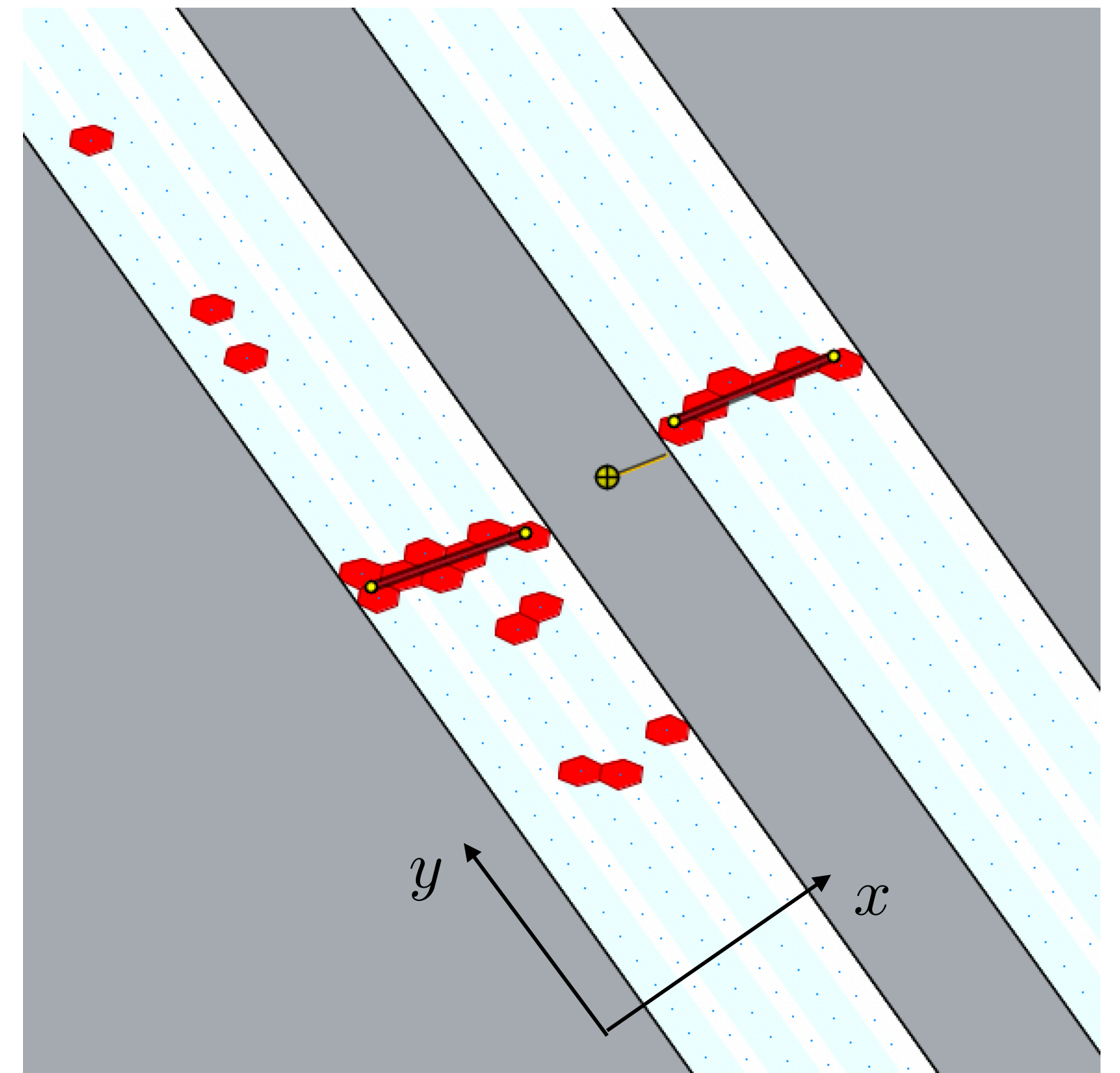# New DC Clustering and Effects on Tracking Efficiency

Tongtong Cao
CLAS12 Software Meeting
Aug 8, 2024

Introduction

- Some issues in DC clustering algorithms cause that some clusters are lost.

- New clustering attempts to retrieve real clusters, while not increasing too much extra noise clusters.

- About old DC clustering:

  - Local coordinates: x axis is for layers, and y axis is for wires

  - Firstly, clumps, comprised by hits in a continuous range of y/wire axis, are constructed

  - Then, a pruner is used to trim hits on clumps

  - Next, if hit distribution in a clump is complicated so that quality of linear fitting is failed, a splitter by algorithm of the Hough line transform is used to find all combos of hits, who are close to a straight line, so that clusters are constructed by the hit combos,.

 - Clusters are required to involve at least 4 layers

Issue 1: A bug in Splitter

- The bug is located at https://github.com/JeffersonLab/coatjava/blob/development/reconstruction/dc/src/main/java/org/jlab/rec/dc/cluster/ClusterCleanerUtilities.java#L174

- The bug causes that plenty of cluster candidates, distinguished by the splitter, are lost.

Issue 2: Too tight for # of layers in clusters >= 4

- Due to existence of dead wires, mis-killing of SNR and AI-denoising, and edge effects, some real hits do not exist, or are mis-killed.

- Such cases of clusters might be lost if clusters are required to involve at least 4 layers

- Solution:

  ‣ Categorize clusters into two types: normal and exceptional

  ‣ Layers in clusters >= 4 for normal clusters, and Layers in clusters >= 3 for exceptional clusters

  ‣ Exceptional clusters could be such cases:

    1. First two (layers 1&2) or last two (layers 5&6) layers are lost

    2. One or more layers are skipped
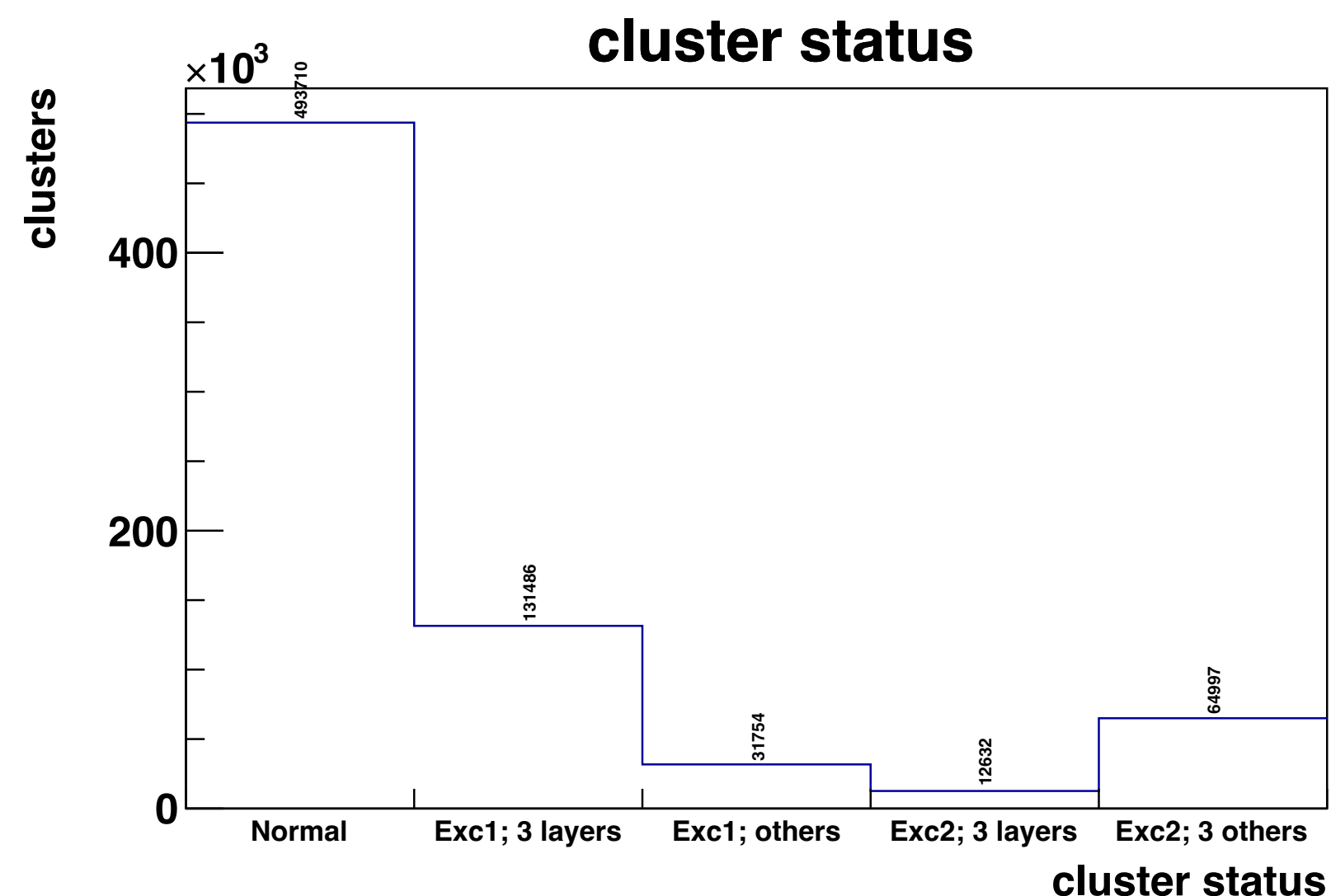
**Examples for exceptional clusters involving only 3 layers**
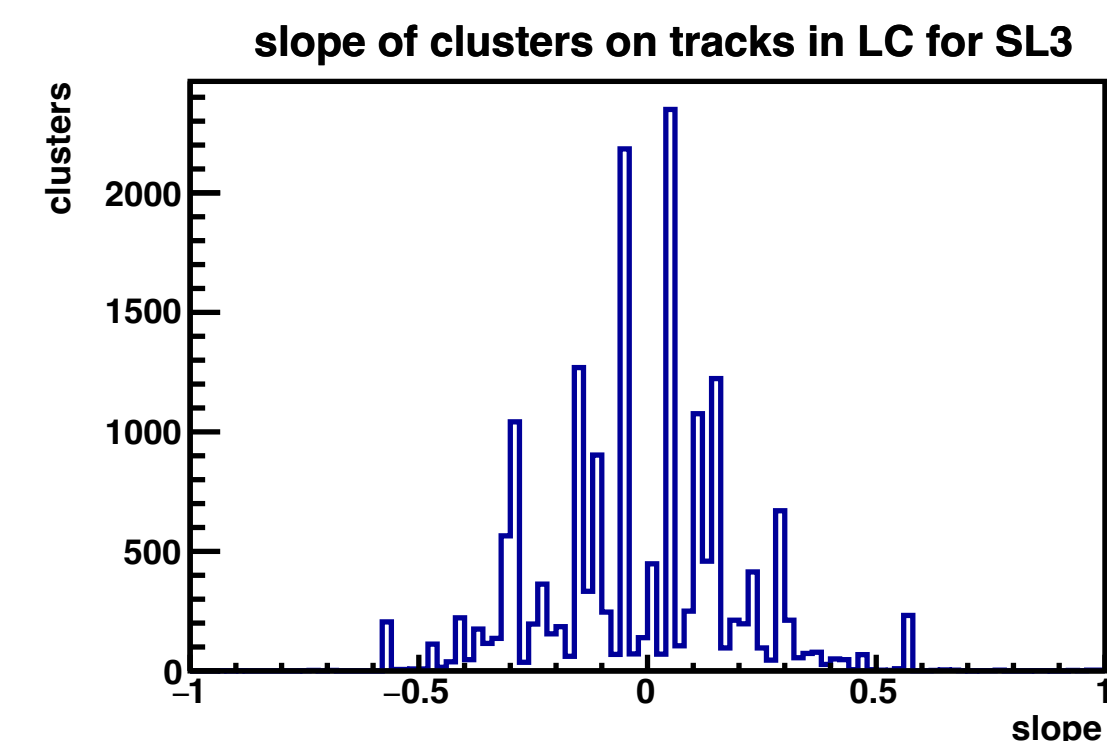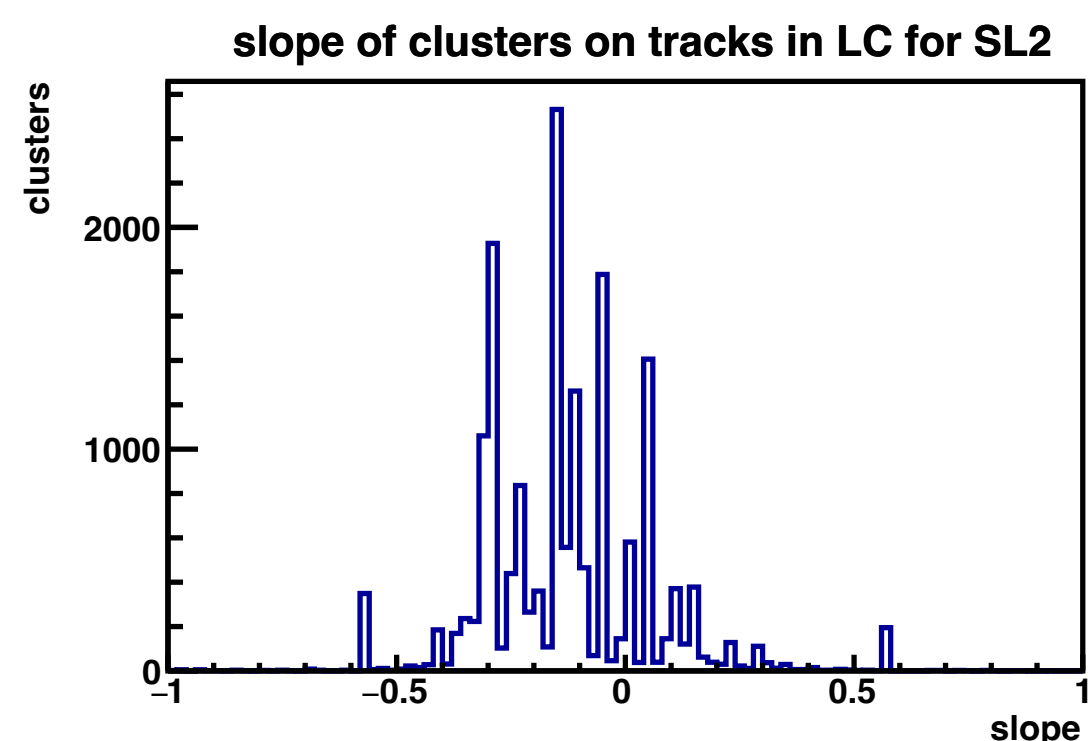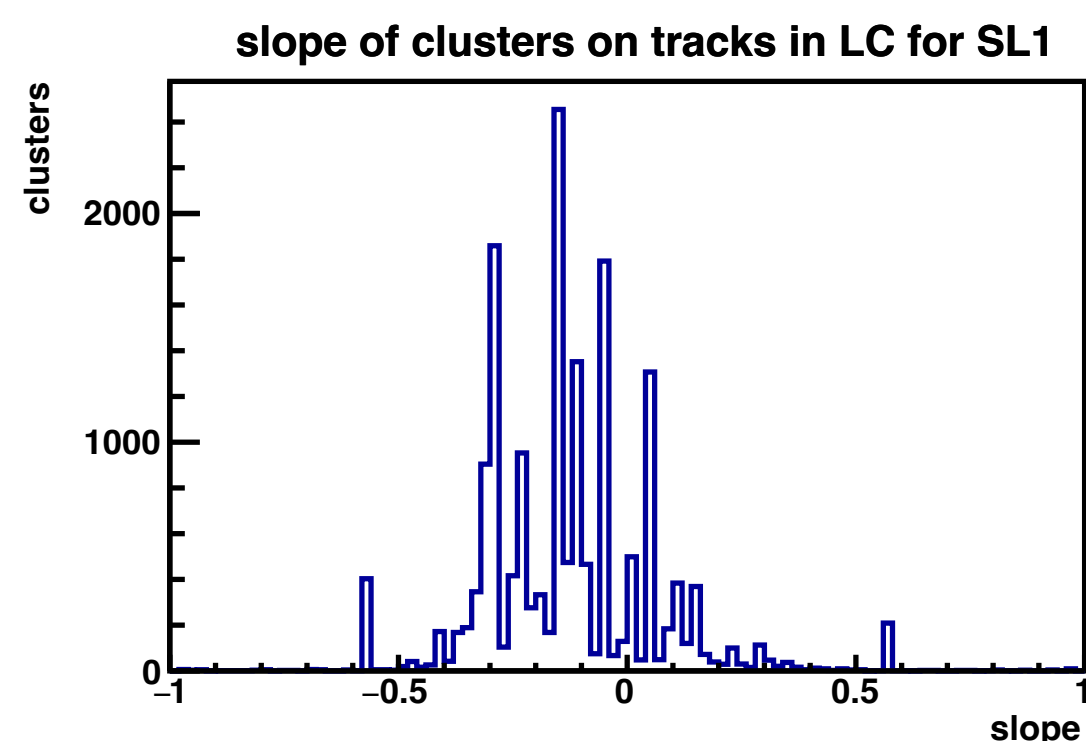

Exc1

Exc2

4
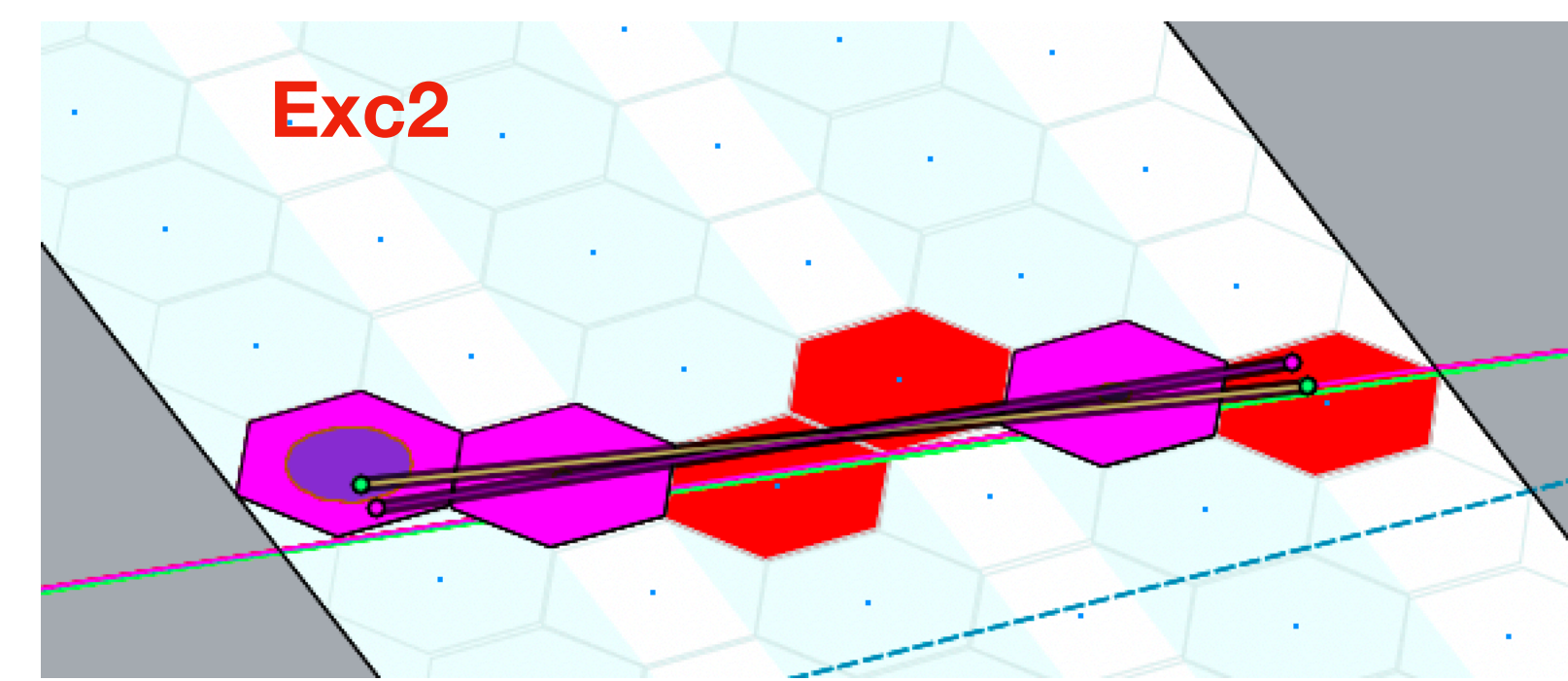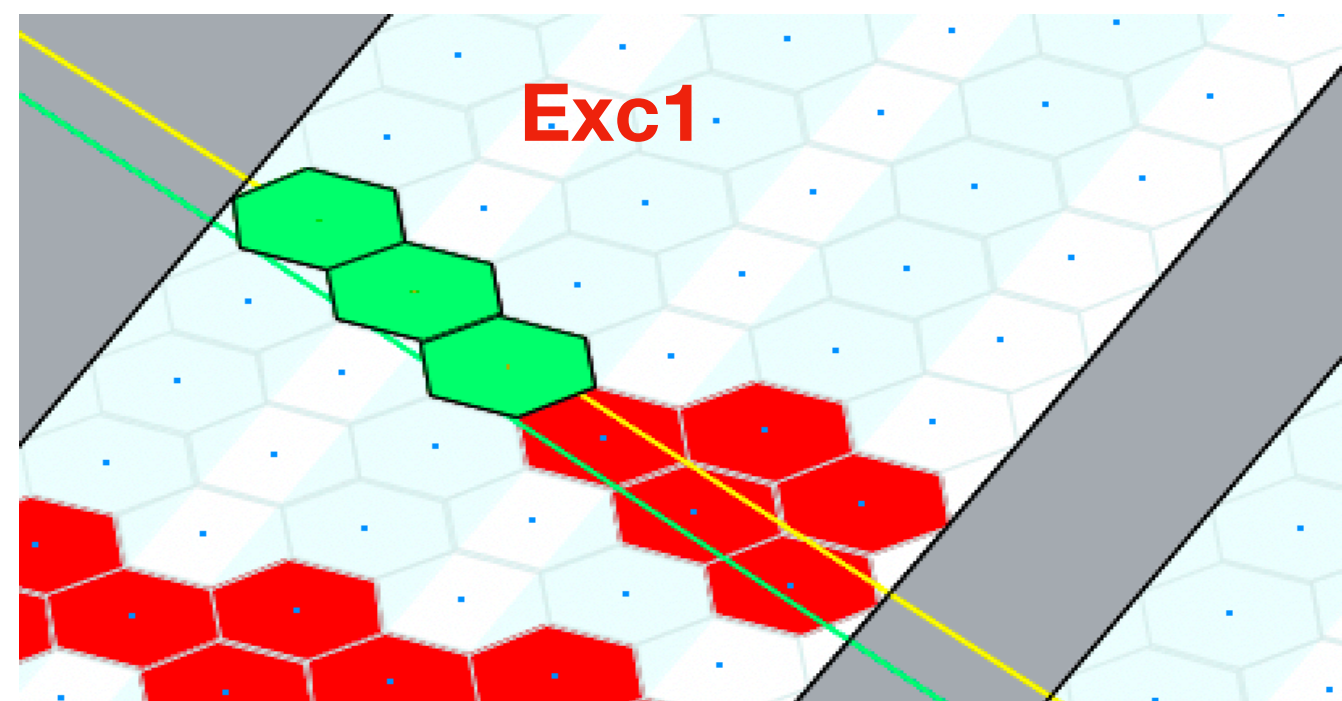
Issue 3: Update OverlappingClusterResolver()
Selection of overlapping clusters from splitter

- Cancel the requirement that slopes of overlapped clusters must be close.

- Instead, from list of overlapped clusters, remove clusters whose absolute value of slope is out of $\tan(30^0)$ for R1&R2.

Slope of clusters on TB tracks in local coordinates
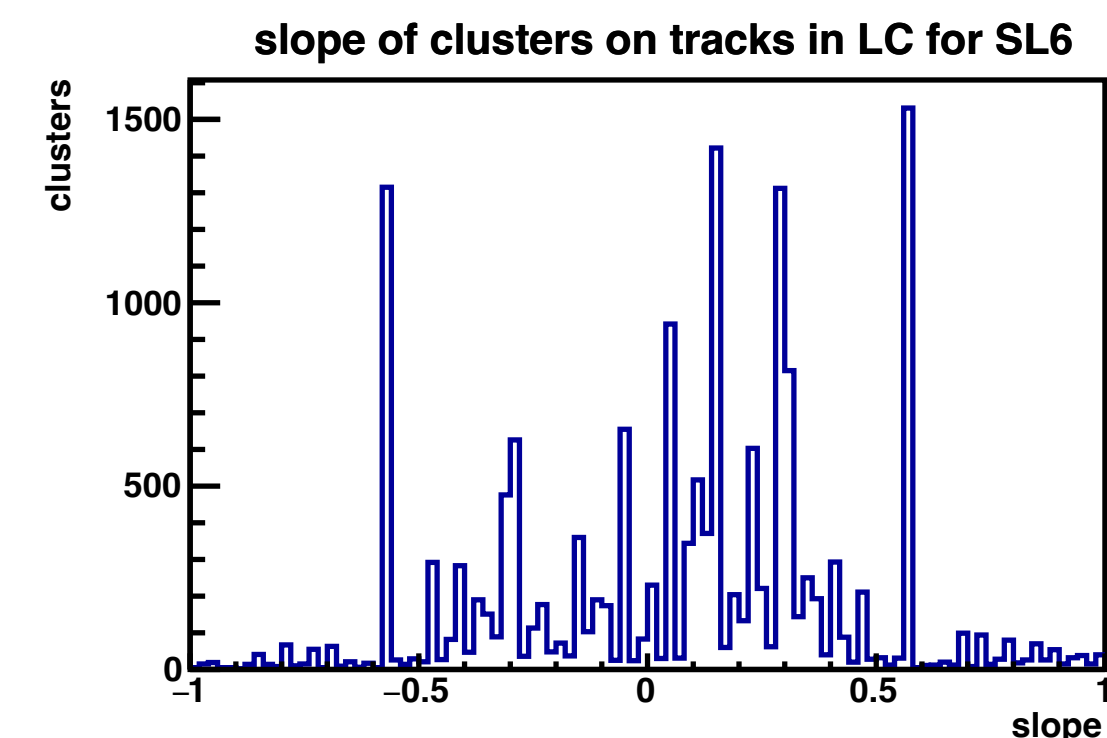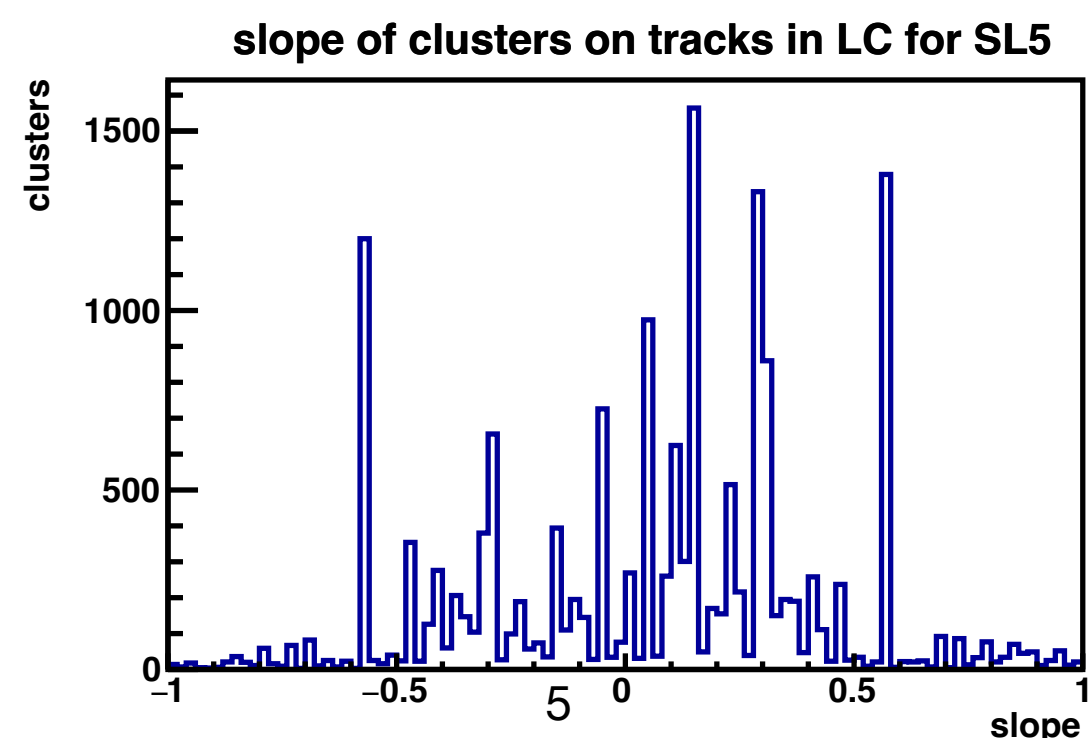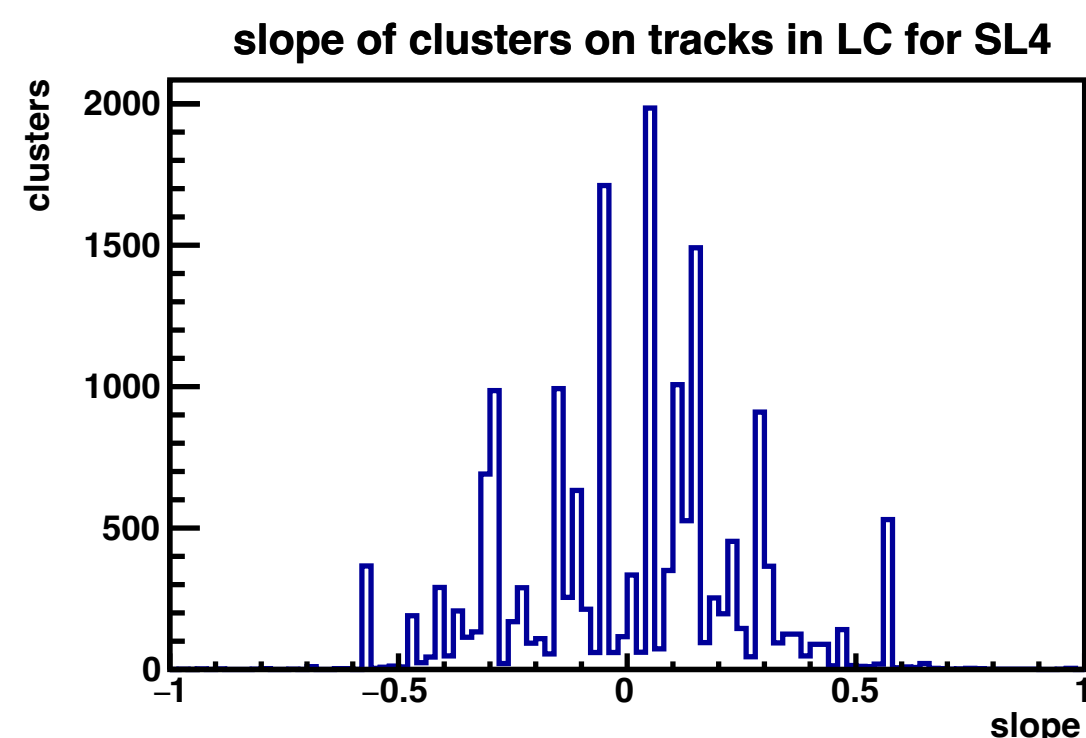
Issue 3: Update OverlappingClusterResolver()
Order of overlapping clusters

- In old clustering, overlapping clusters are ordered base on cluster size, and the first cluster with the maximized size in the order list is chosen.

- However, there is no order for clusters with the same size. It causes that overlapping clusters with the same size could be chosen at different orders of loop for finding of overlapping clusters.

- In new clustering, clusters with the same size are ordered based on linear fitting quality as the secondary priority for order of overlapping clusters.

Two 6-hit clusters with 5 hits overlapped from old clustering.

```
Hit1_ID :      17      76      98     116     172     180     202     210     242     250
       +      260     270     296     302     314     317     317     323     334     340
       +      356     373     377     392     407     436     453     453     474     477
       +      511     550     555     575     593     609     700     714     730     739
       +      735     765     804     812     855     861     882     880     882     930
       +      976     992    1009    1018    1026    1035
Hit2_ID :      11      73      97     115     175     181     205     212     244     251
       +      258     276     295     300     313     314     314     327     330     337
       +      366     371     383     395     409     435     458     458     473     479
       +      518     548     561     572     592     608     704     718     732     740
       +      736     763     807     815     851     865     879     882     879     929
       +      973     994    1008    1019    1028    1033
Hit3_ID :      12      72      99     120     173     182     204     213     245     252
       +      256     273     297     305     315     313     318     326     332     336
       +      360     370     379     393     408     440     452     452     478     480
       +      513     545     564     574     594     606     701     715     731     738
       +      738     761     803     810     852     862     881     879     881     934
       +      977     995    1006    1012    1024    1031
Hit4_ID :      13      68     100     117     176     183     203     211     246      -1
       +       -1     275      -1      -1      -1     315     310     324     333     339
       +       -1     372     381     398     406     439     459     457      -1      -1
       +      516     546      -1     576     591     607     702     720     728      -1
       +       -1      -1     809     811     854     871     873     881     873     935
       +      974     993    1010    1017    1023    1034
Hit5_ID :      16      75     102     119     174     179     206     209      -1      -1
       +       -1     274      -1      -1      -1      -1     311     325     331     338
       +       -1     375     380     397     405     437     451     459      -1      -1
       +      514     547      -1     577      -1     605     705     719      -1      -1
       +       -1      -1     802     813     853     863     886     873     872     937
       +      975     991    1011    1016    1027      -1
Hit6_ID :      -1      71     103     118     177     178     207     214      -1      -1
       +       -1     272      -1      -1      -1      -1     316     322     329     343
       +       -1     374     382     394     410     438     460     451      -1      -1
       +      517     549      -1     578      -1     610     703     717      -1      -1
       +       -1      -1     808     814     856     864      -1     872     886      -1
       +      972     996    1007    1015    1025      -1
```

```
avgWire :   47.8000  54.6250  35.2857  43.5000  28.5000  44.0000  36.6667  47.3333  22.0000  25.0000
       +    28.6667  86.6667  28.6667  32.3333  95.3333  95.0000  94.0000  42.0000  53.0000  62.8333
       +    43.3333  63.5000  76.4286  49.1429  55.5000  27.5000  80.5000  80.4286  29.6667  32.3333
       +    73.4444  76.5000  91.3333  70.4286  53.0000  87.5000  90.7143  88.7143  23.2500  66.6667
       +    65.6667  89.6667  24.5000  30.3333  16.0000  26.5714  32.6000  32.5000  32.5000  67.6000
       +    27.6667  28.7143  23.3333  32.2500  24.0000  45.0000
```

6

Issue 4: Hits are shared by multiple clusters

The two banks are output from DC clustering. A cluster ID is assigned for a hit in HitBasedTrkg::Hits.

**size of cluster vs. clusterInHitBank**
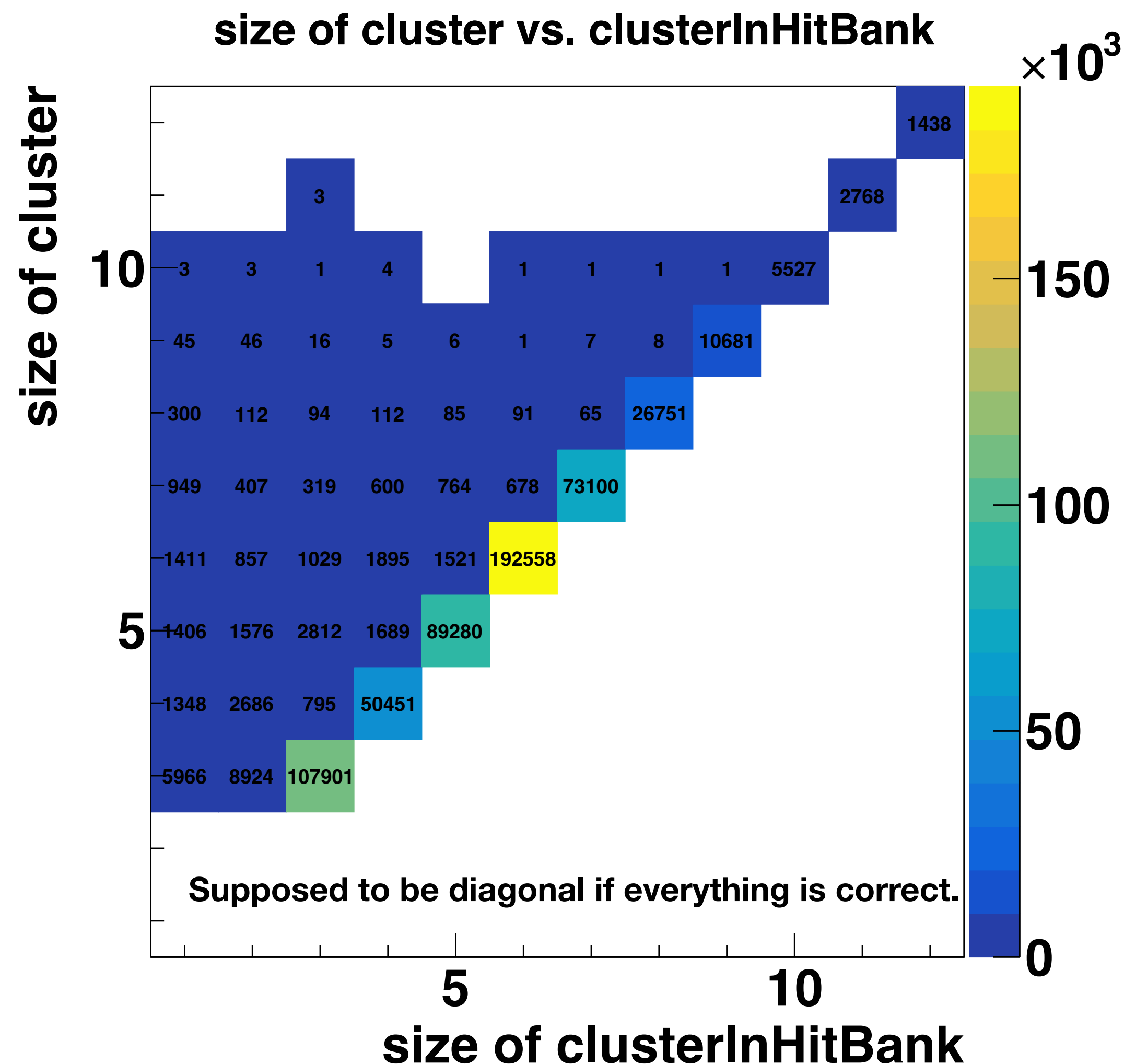
- For part of clusters, size of clusters in the cluster bank is larger than size of clusters which are constructed by hits in the hit bank according to cluster ID of hits.

- It means that some hits of some clusters are lost in clustering.

- The reason is that some hits are shared by two or even more clusters, but ID of cluster associated with a such hit is assigned to be ID of last cluster among hit-overlapping clusters, instead of IDs of all associated clusters.

HitBasedTrkg::Hits   **According to cluster ID of hits, cluster is constructed.**

Effect of Issue4

- After clustering (AI prediction of cluster combos), hits from the bank HitBasedTrkg::Hits are read into a map with key of cluster ID.

- Then, segment is recomposed with limit of cluster size for both conventional and AI-assisted tracking.

- Since information of cluster ID is lost for hit-overlapped clusters except the last cluster in the hit bank, hits shared with clusters are lost in recomposition except last hit-overlapped cluster.

- Further, since limit of cluster size, some clusters are lost in the following process after segment recomposition.

- Some of clusters with larger size are not lost, but some hits belong to them are missed.

- The issue affects both conventional tracking and AI-assisted tracking.

size of cluster vs. clusterInHitBank

8
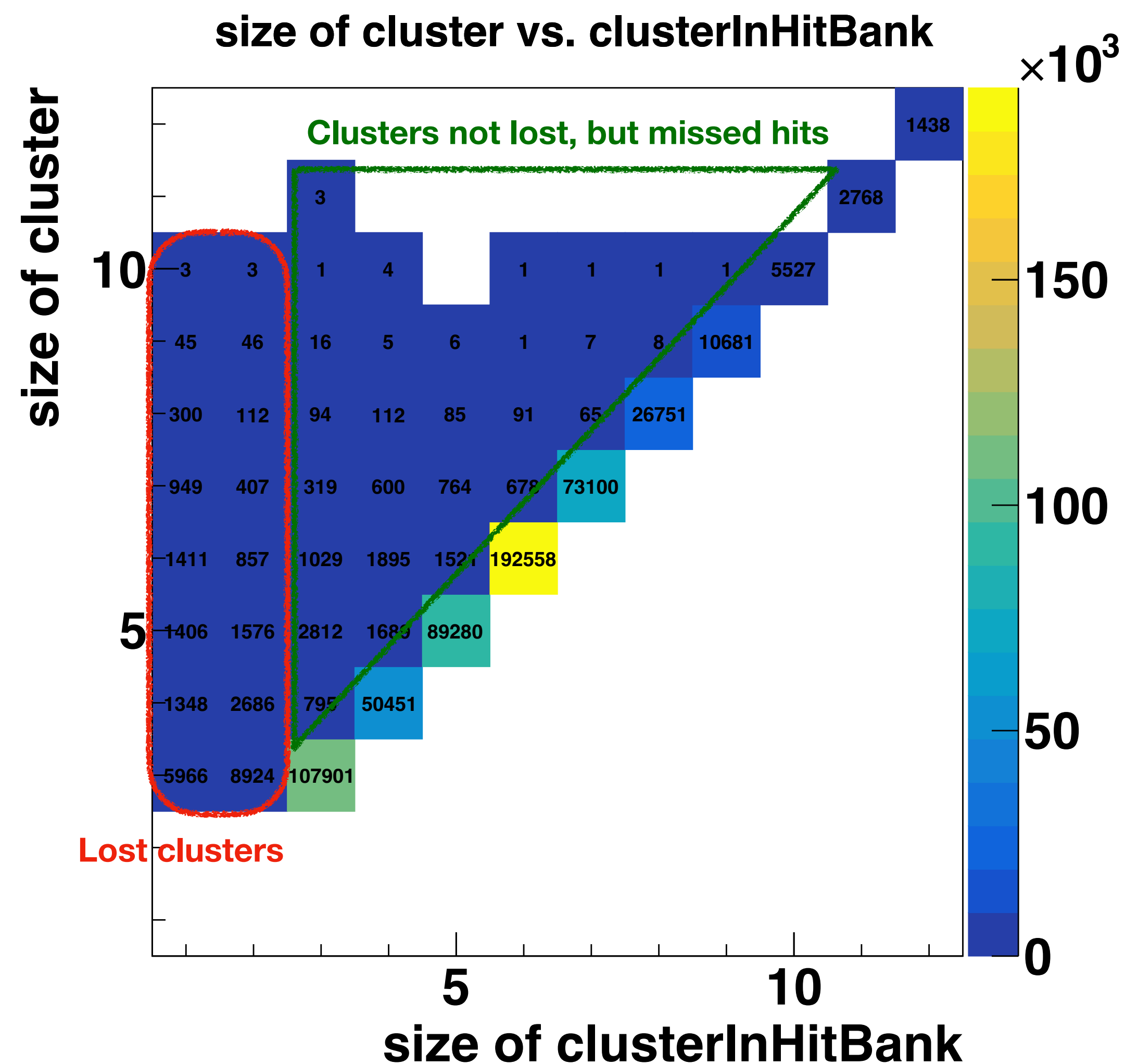
How to Fix the Issue4?

- For hits shared with multiple clusters, duplicate them with different id and associated cluster ID, and add duplicated hits into HitBasedTrkg::Hits.

- Connection between HitBasedTrkg::Hits and DC::TDC is through id in HitBasedTrkg::Hits and index of DC::TDC. With addition of duplicated hits, a new item called as indexTDC is added into HitBasedTrkg::Hits.

- In DC reconstruction and tracking, DC::TDC is input only for DC clustering, and HitBasedTrkg::Hits is output of DC clustering. With the above update, full hits, including original hits and duplicated hits, will enter the following process for reconstruction and tracking.

- Accordingly, ClusterFinder::findClumps() in PatternRec::RecomposeSegments() is updated.

After Issue4 Fixing

size of cluster vs. clusterInHitBank
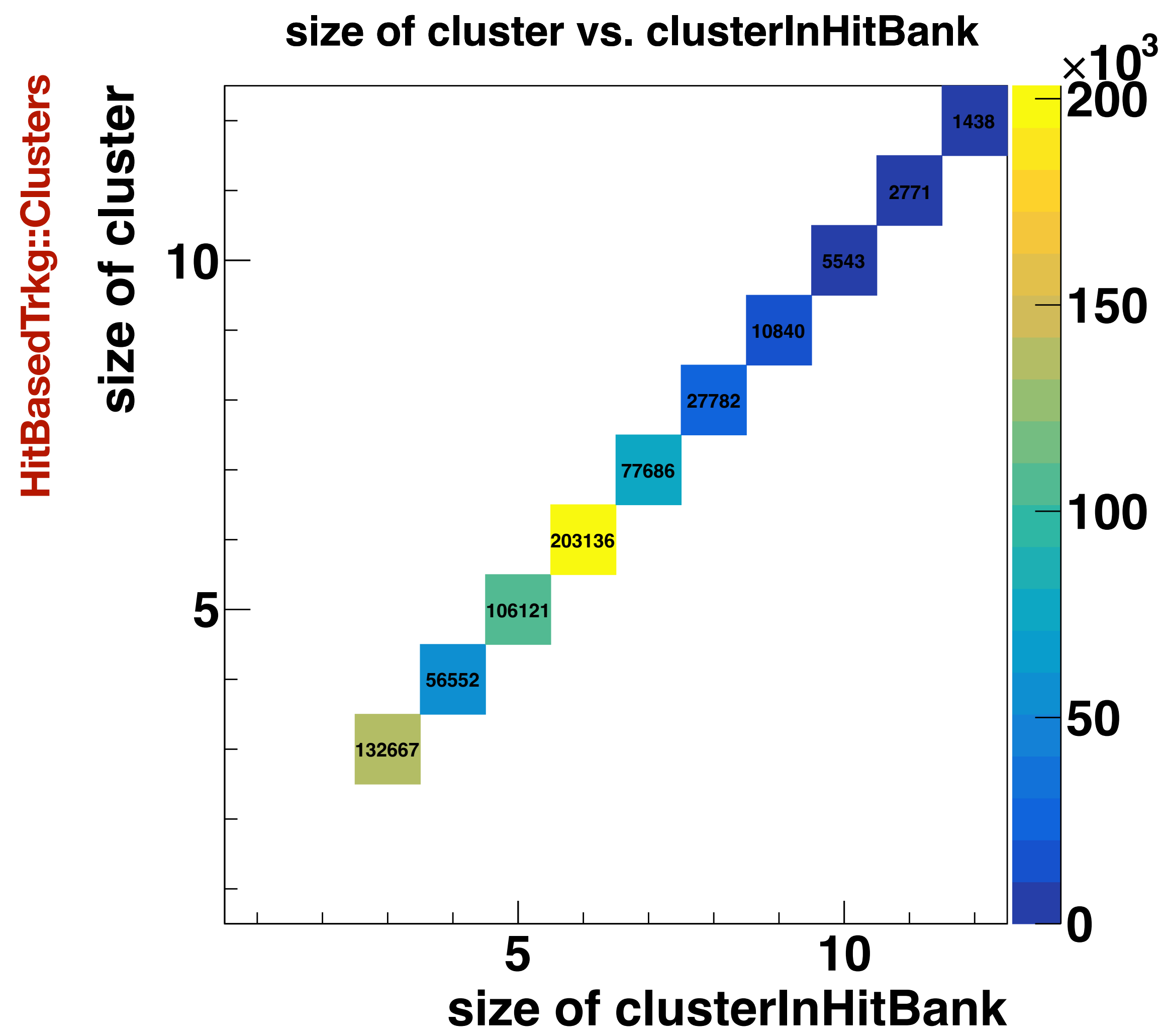
Exactly diagonal as expected

HitBasedTrkg::Clusters

size of cluster

size of clusterInHitBank

HitBasedTrkg::Hits   According to cluster ID of hits, cluster is constructed.

Comparison of Clusters
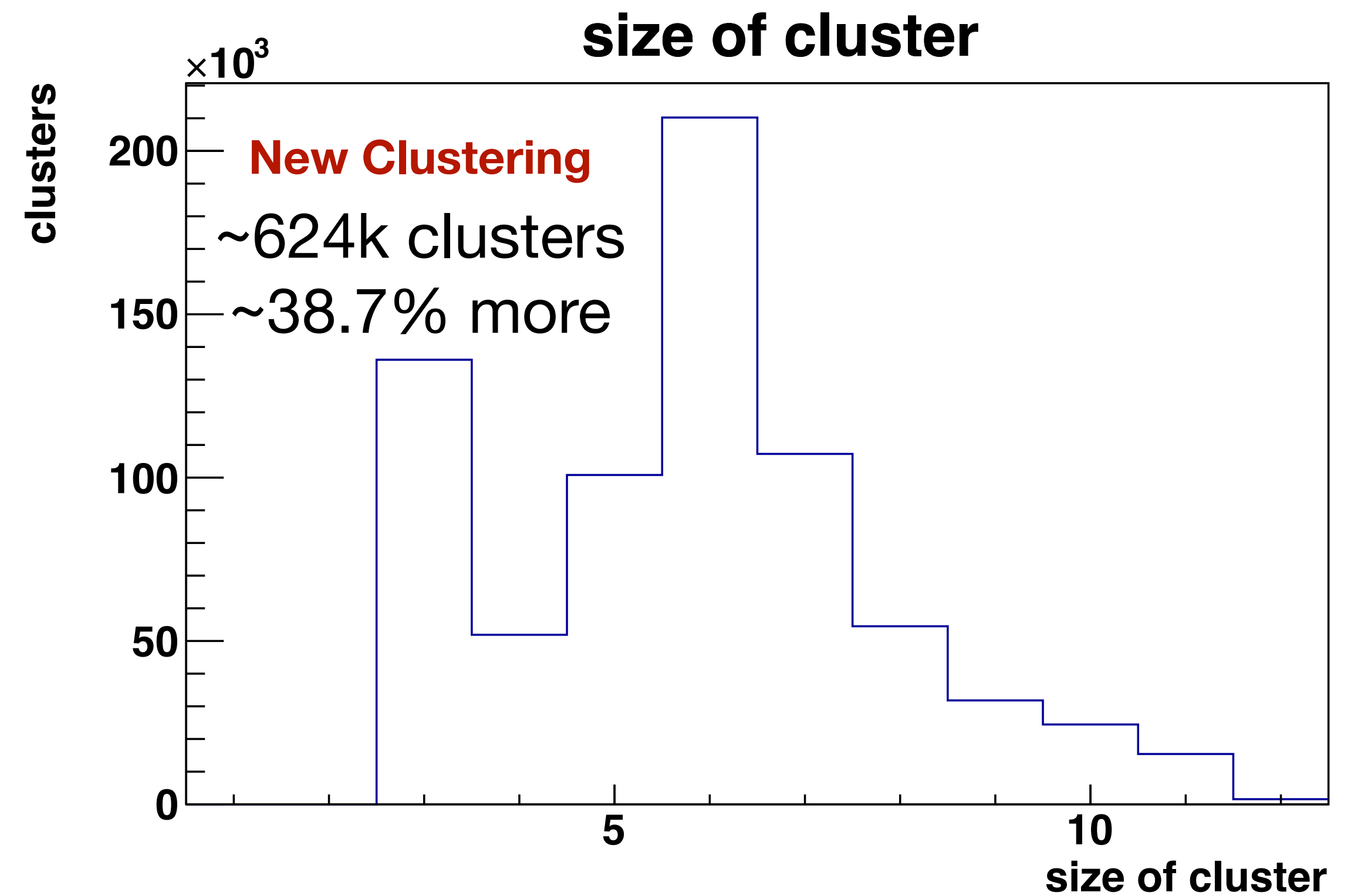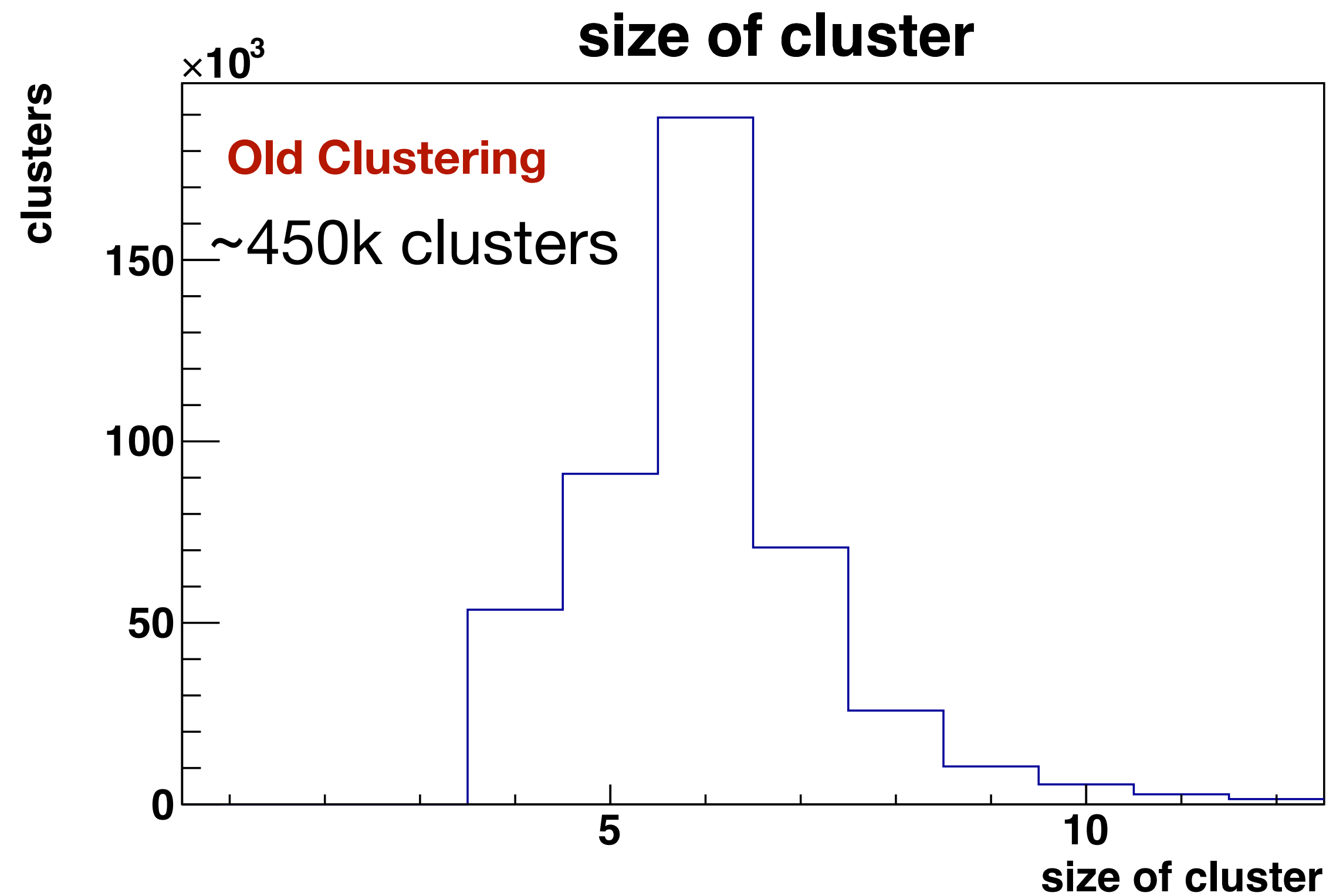
10000 events

Not just 3-hit clusters are involved, but also other-size clusters are increased.

Comparison of AI-predicted DC Cluster Combos

10000 events

# of predicted DC cluster combos

Old Clustering

~20.7k combos

14619

6066

# of predicted DC cluster combos

New Clustering

~24.2k combos

17992

6275

~3.4% more   ~23.1% more
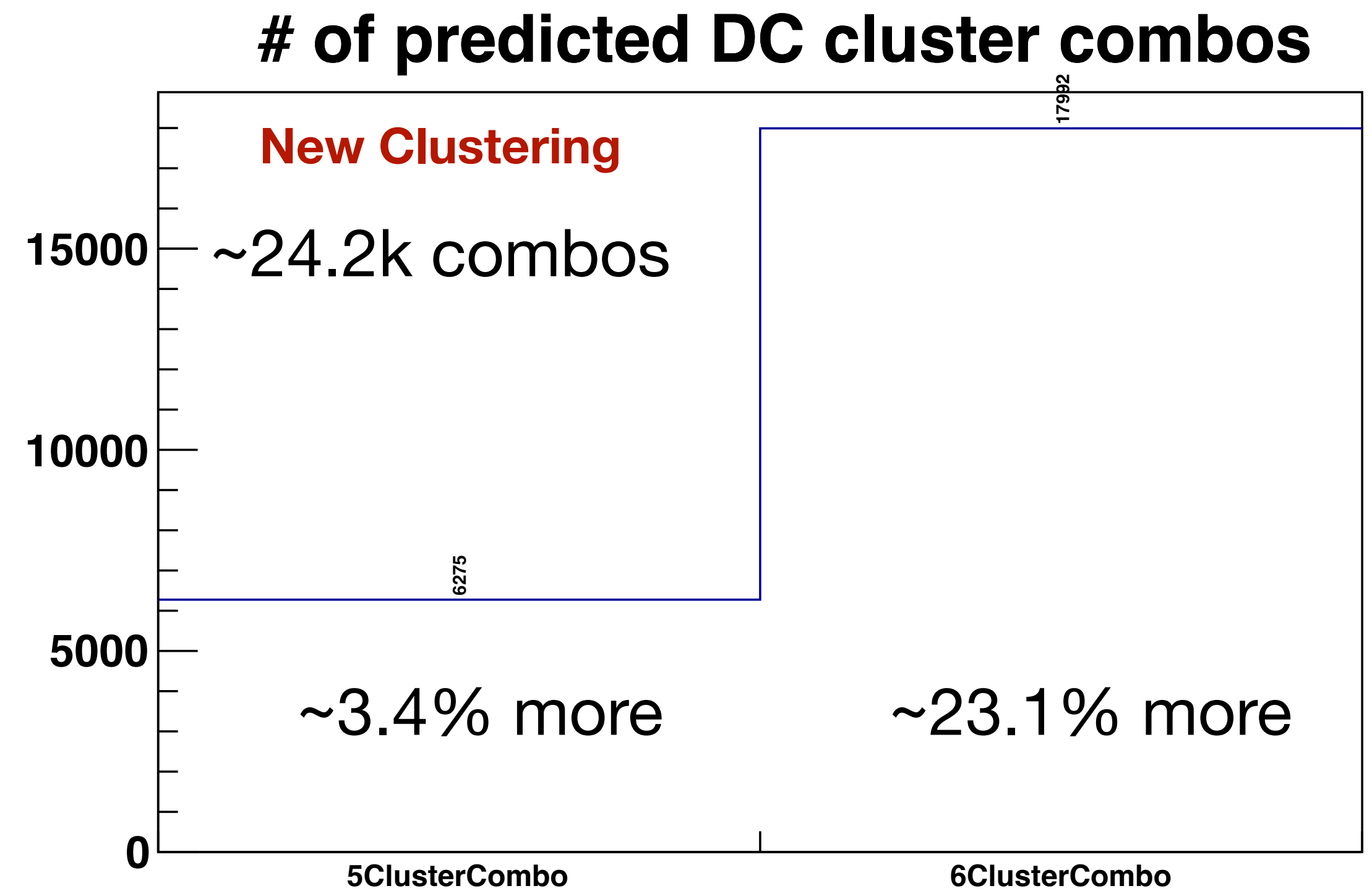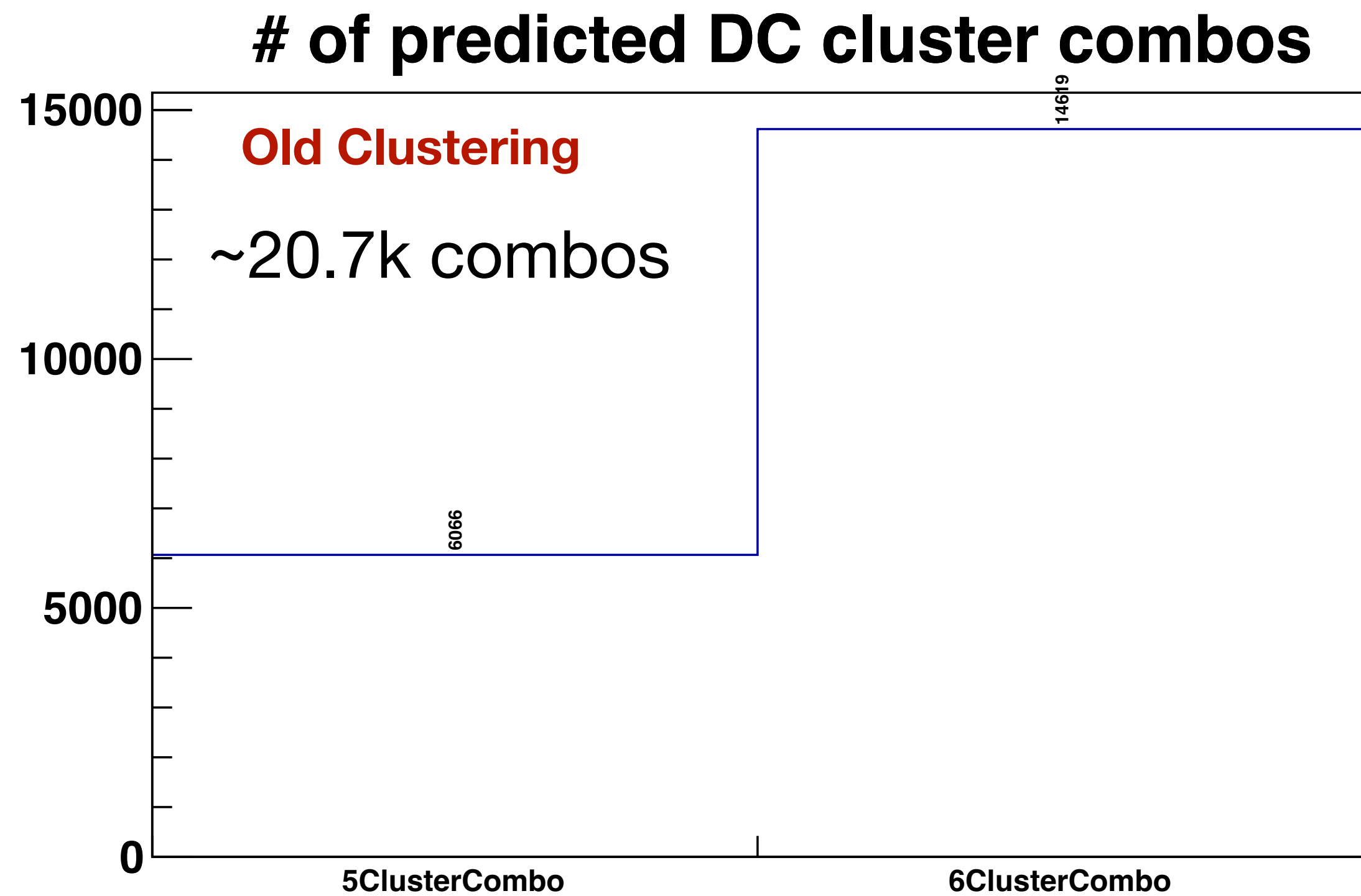
of AI-assisted HB tracks

1000 events	# of AIHB tracks	% of more tracks
Before update	1899	
Fix bug in the splitter	1942	2.3%
Categorize clusters and set different limits for different types of clusters	2014	6.1%
Update OverlappingClusterResolver() for selection overlapped clusters from splitter	2017	6.2%
Fix issue for hits sharing by clusters	2091	10.1%

cumulative

of AI-assisted TB tracks

1000 events	# of AITB tracks	% of more tracks
Before update	1752	
Fix bug in the splitter	1781	1.7%
Categorize clusters and set different limits for different types of clusters	1831	4.5%
Update OverlappingClusterResolver() for selection overlapped clusters from splitter	1832	4.6%
Fix issue for hits sharing by clusters	1884	7.5%

cumulative

Comparison of TB Tracks from AI-assisted Tracking
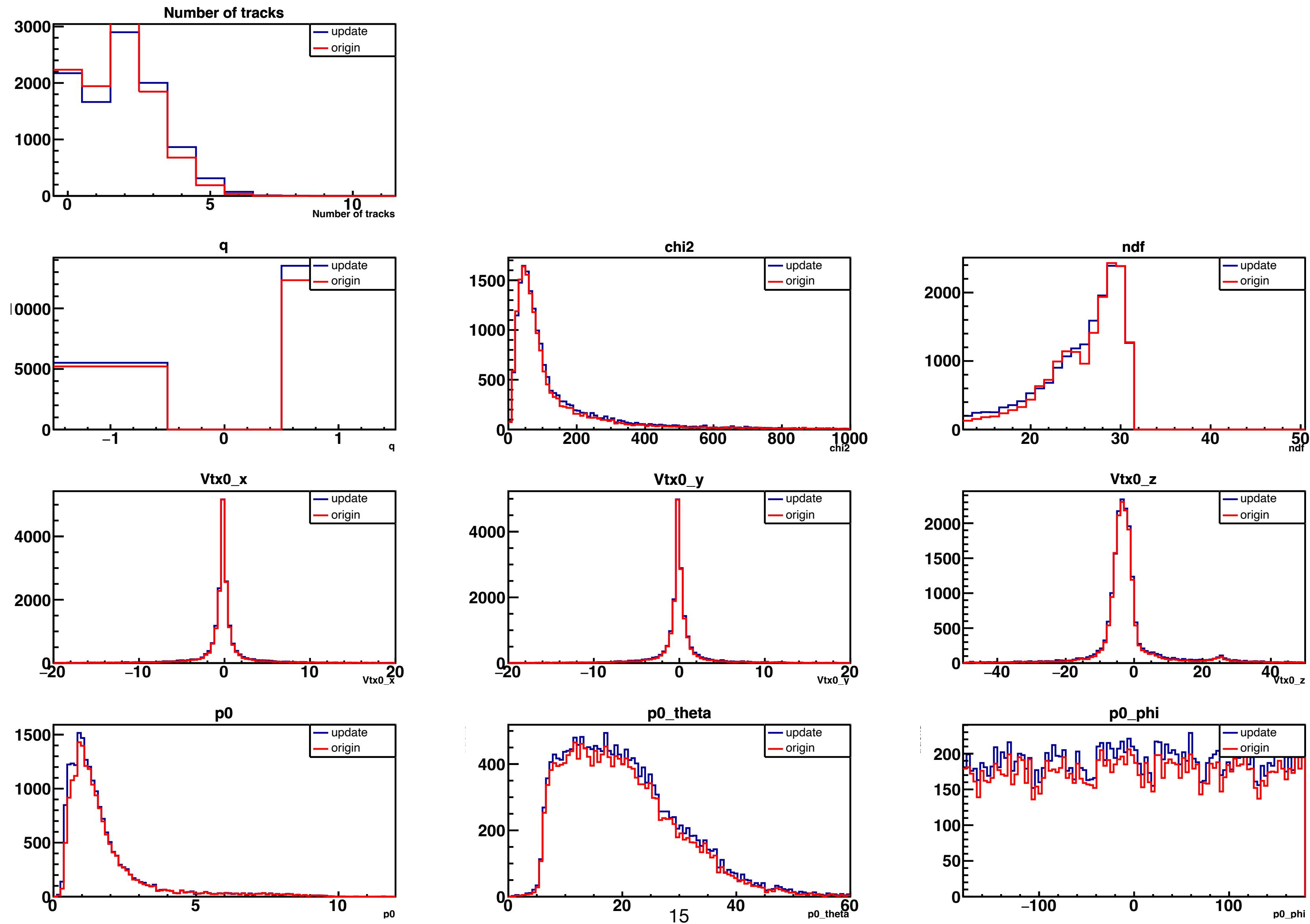
Tracking Efficiency for RGA In-bending

pass2

coatjava v10.1.0
with new tracking

up-to-date coatjava
+ new DC clustering

Tracking Efficiency for RGA Out-bending

AI-denoising

AI-assisted tracking

pass2

coatjava v10.1.0
with new tracking

up-to-date coatjava
+ new DC clustering

17
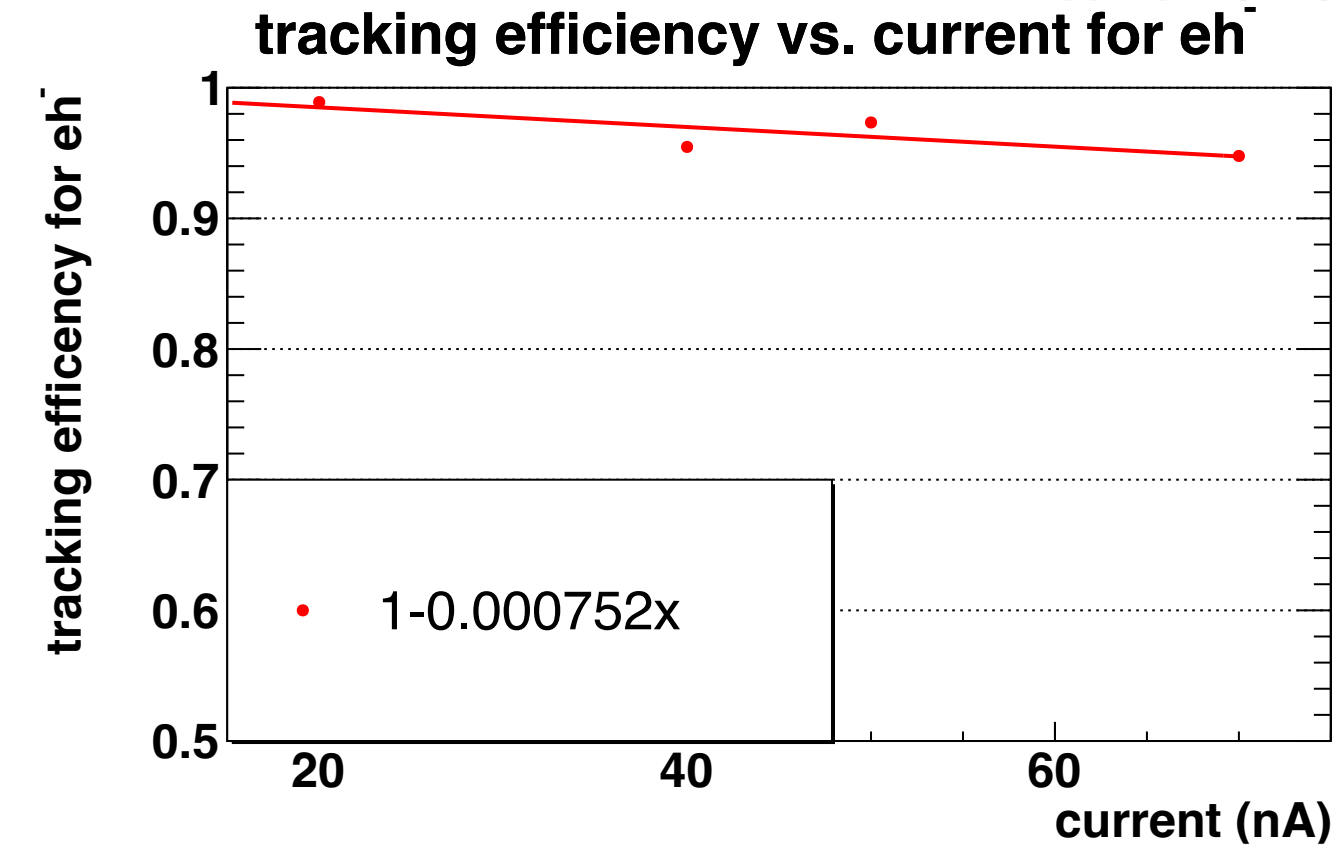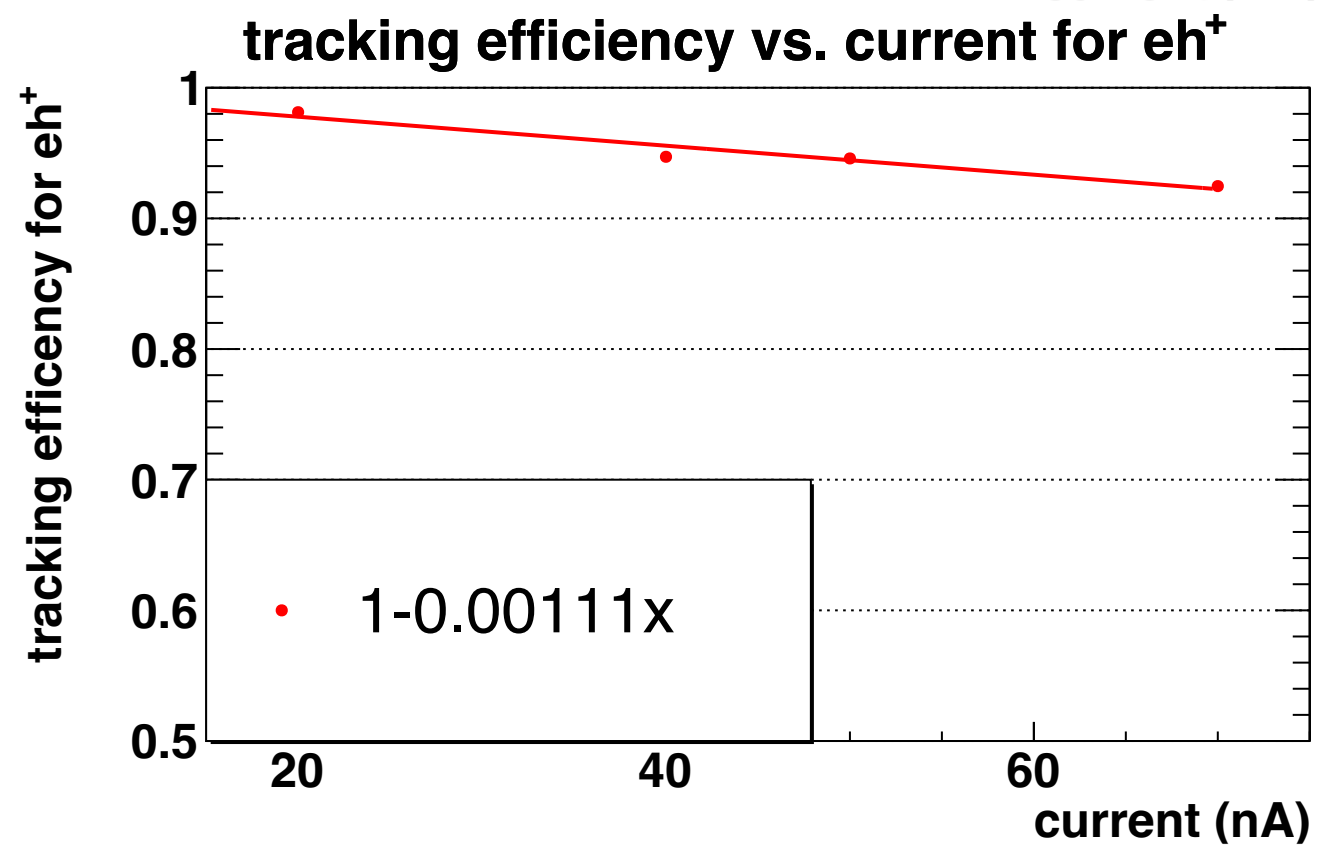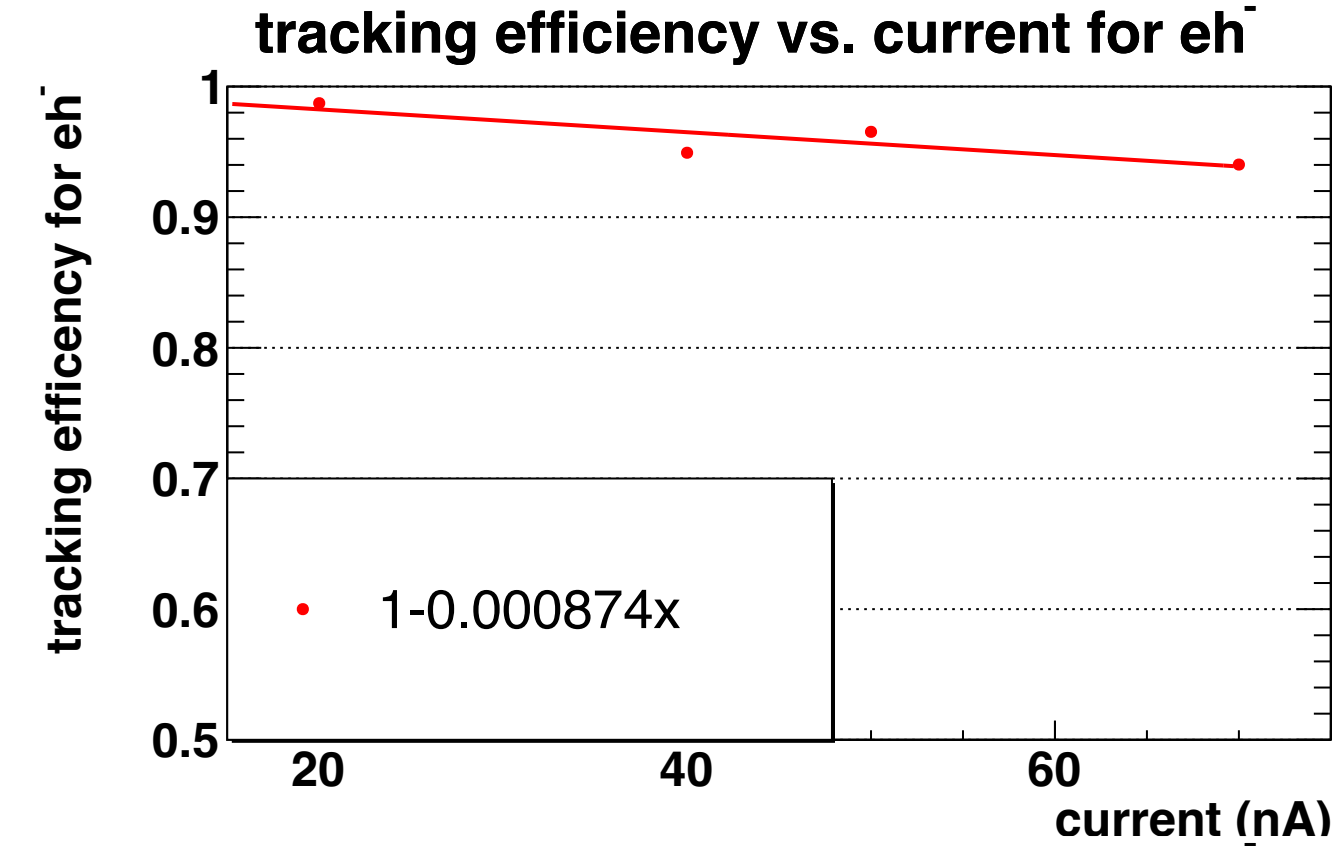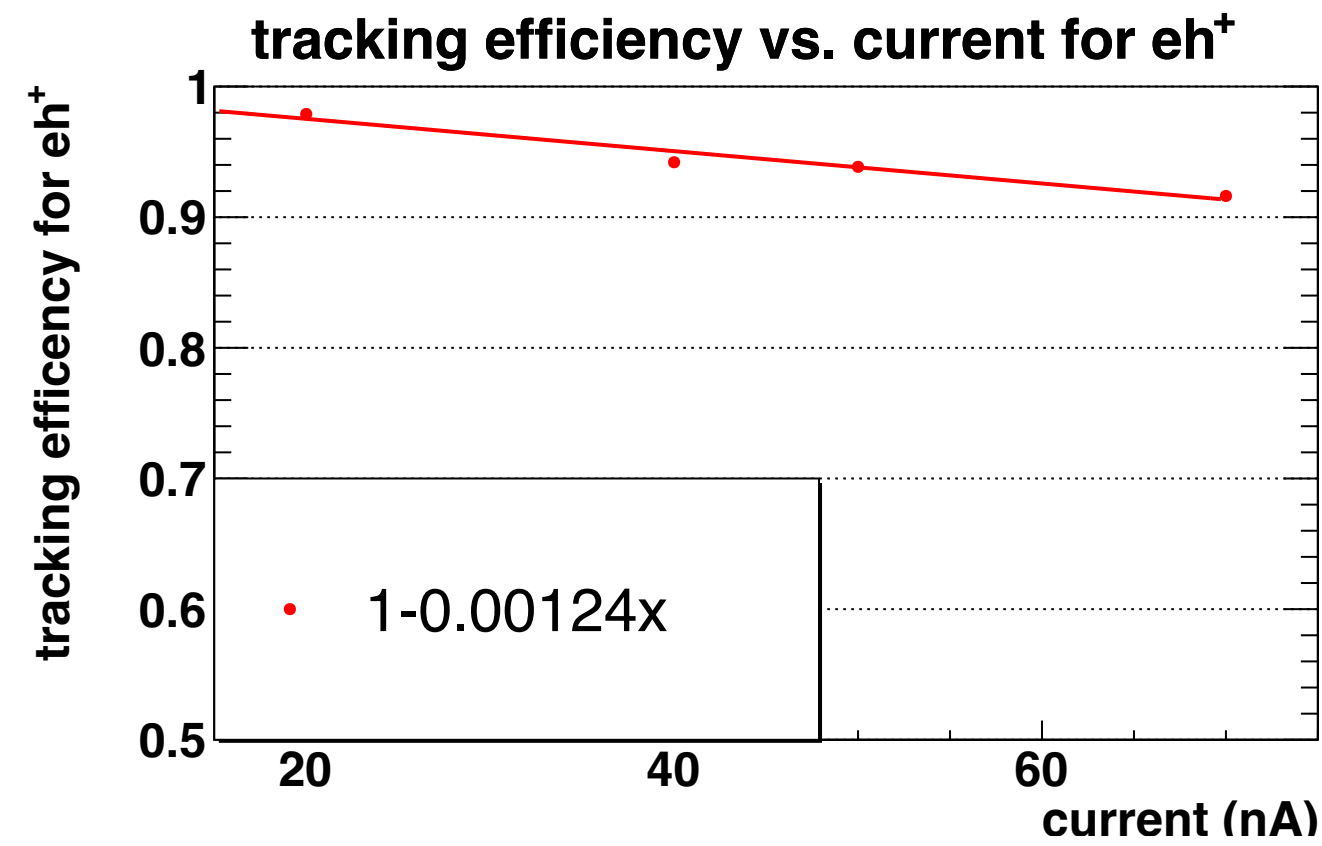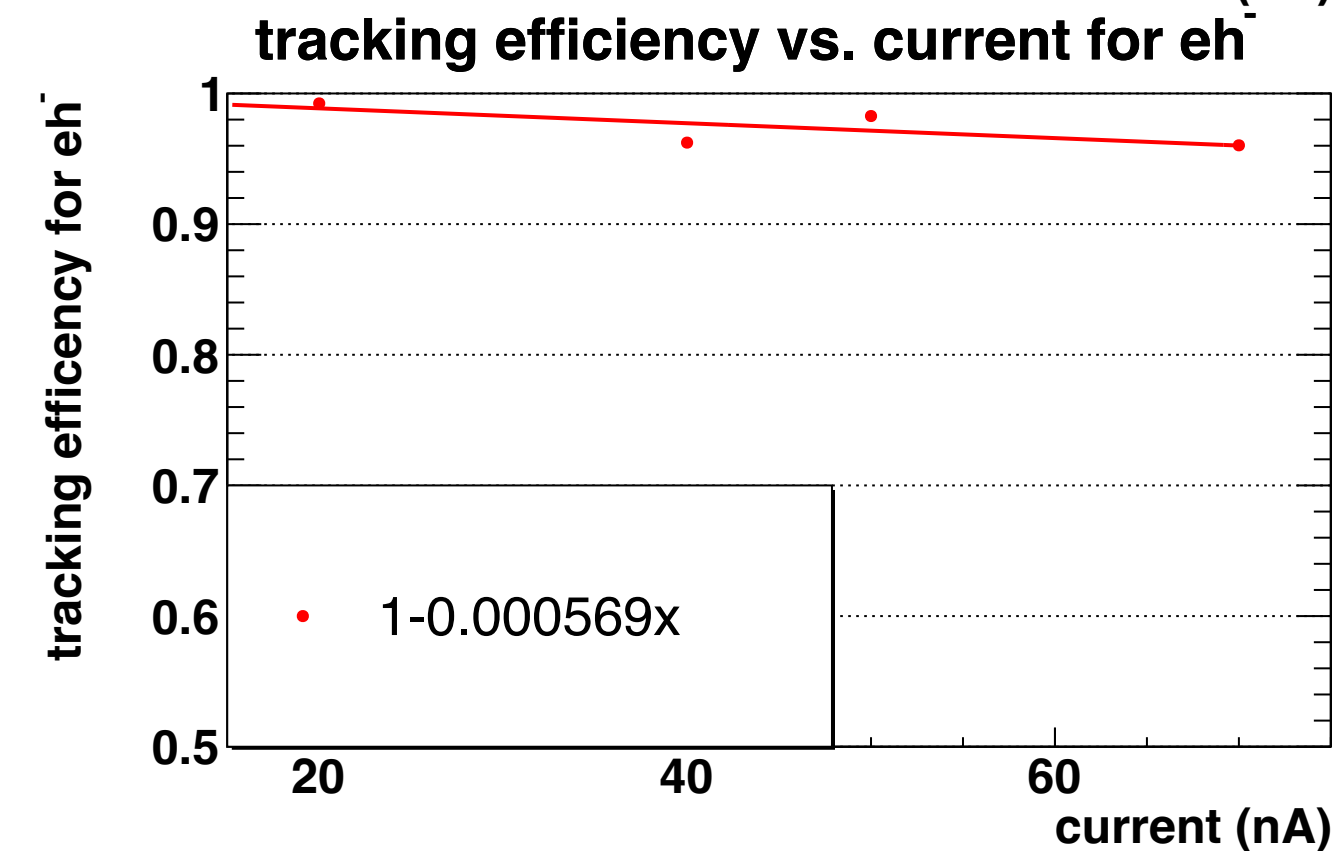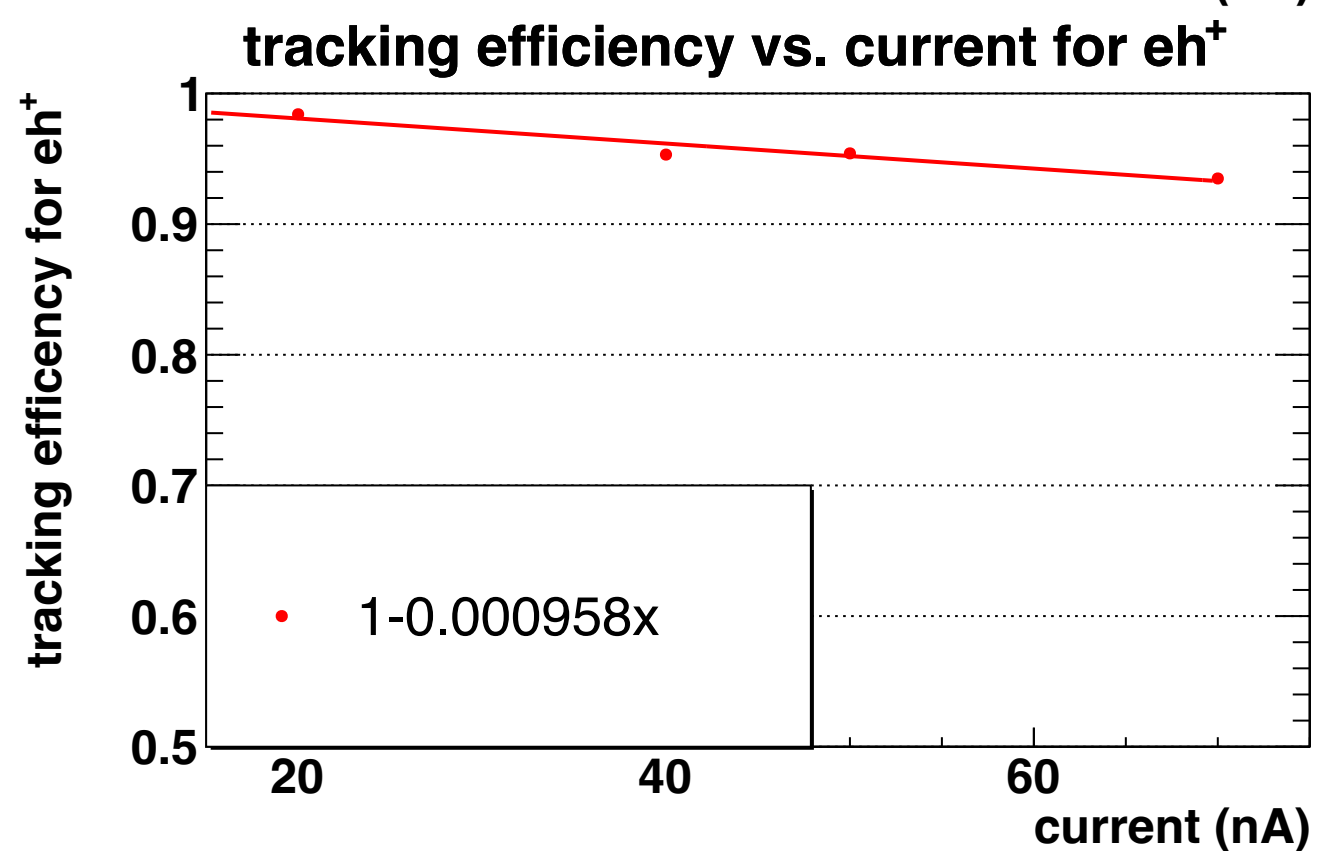
Ratio of Event Reconstruction for RGA

Ratio = new clustering / old clustering

in-bending

out-bending
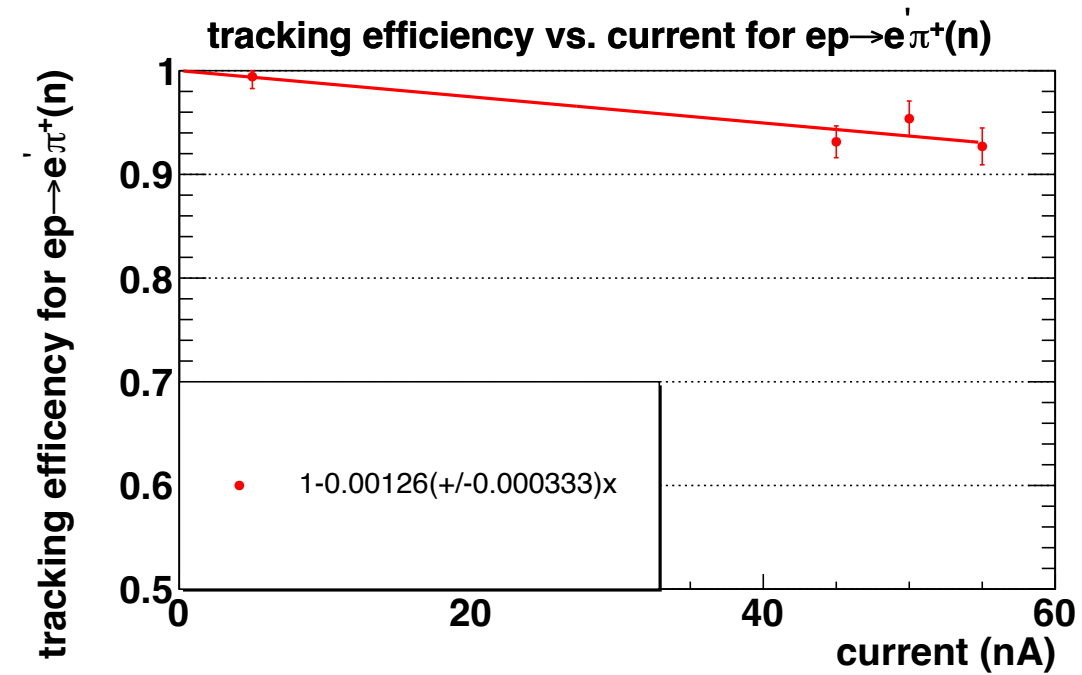

ratio of new/old

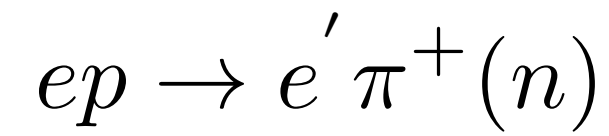ratio of +new clustering/new tracking
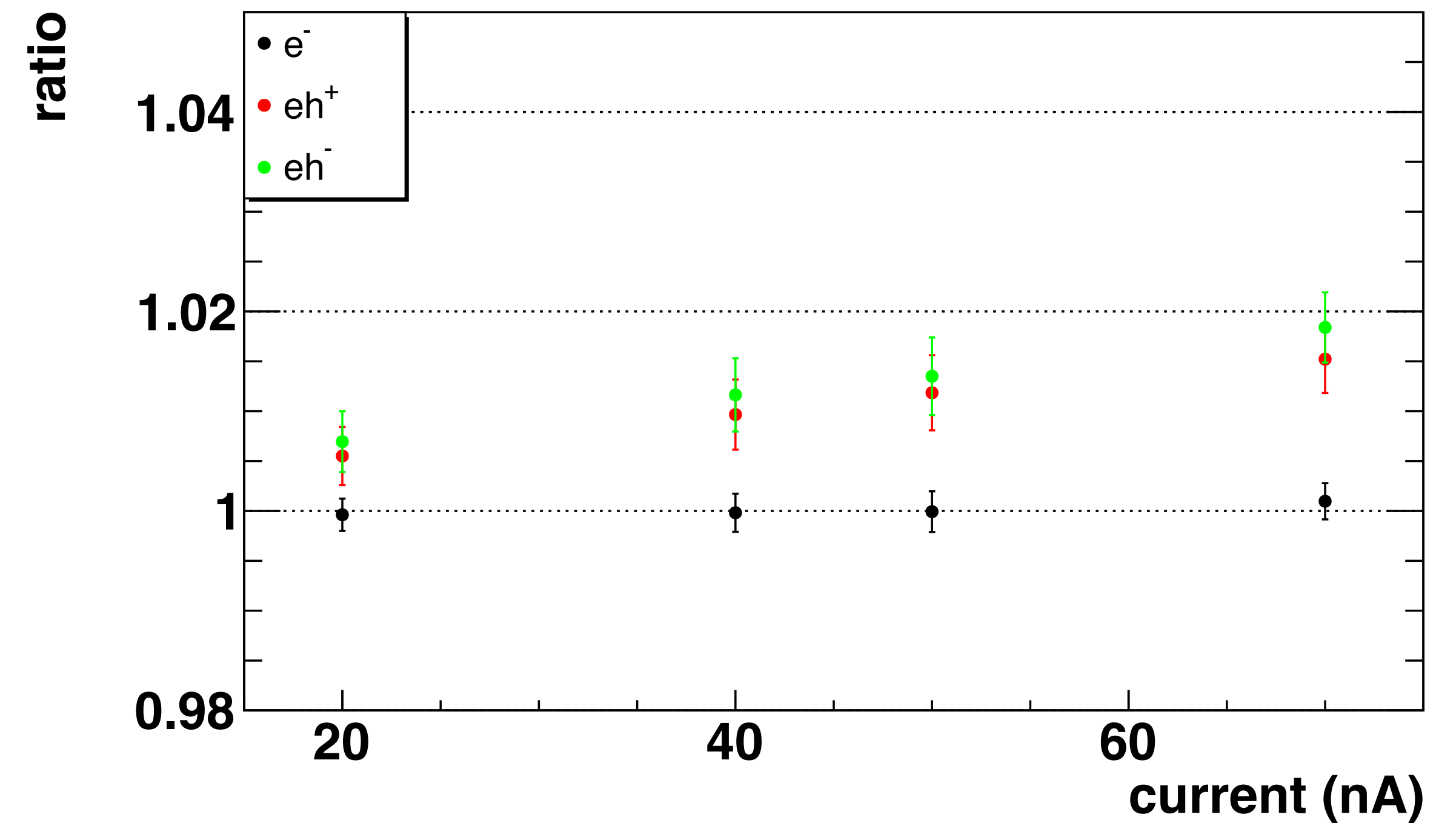
Efficiency = # of events for channel / # of events with electron

Efficiency for Physics Channels: In-bending
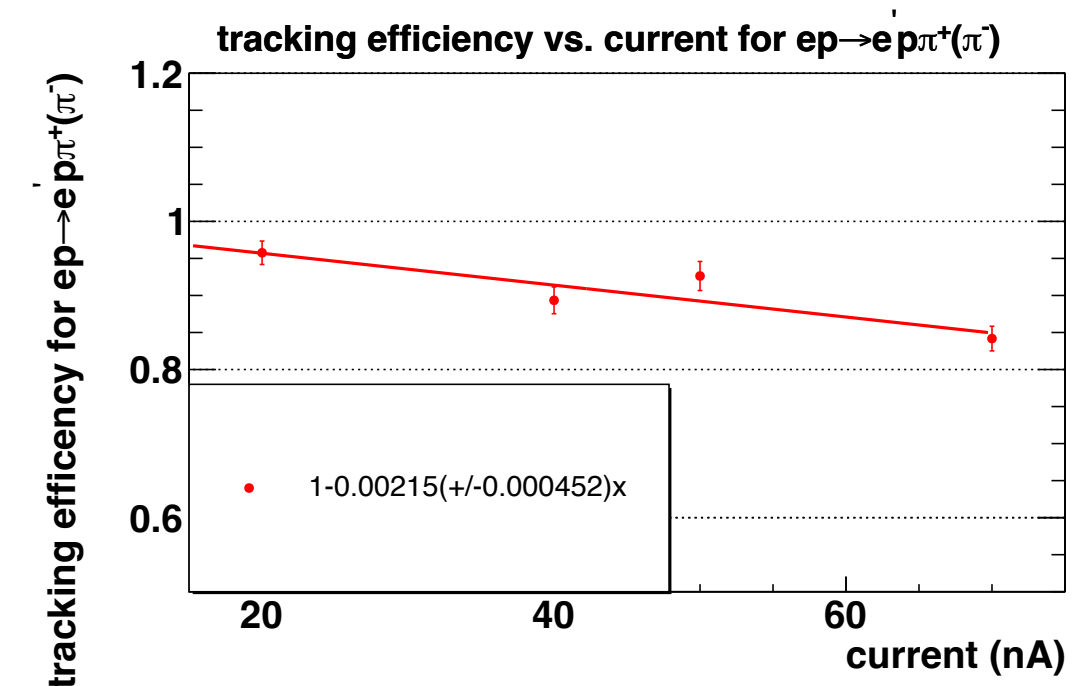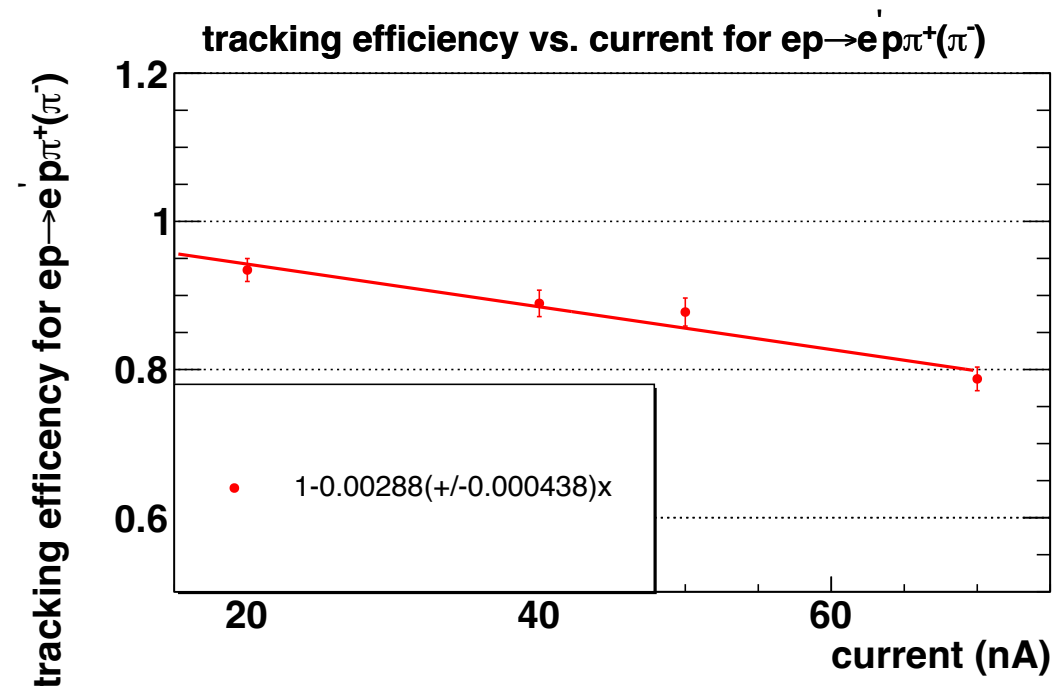
Old clustering   **New clustering**
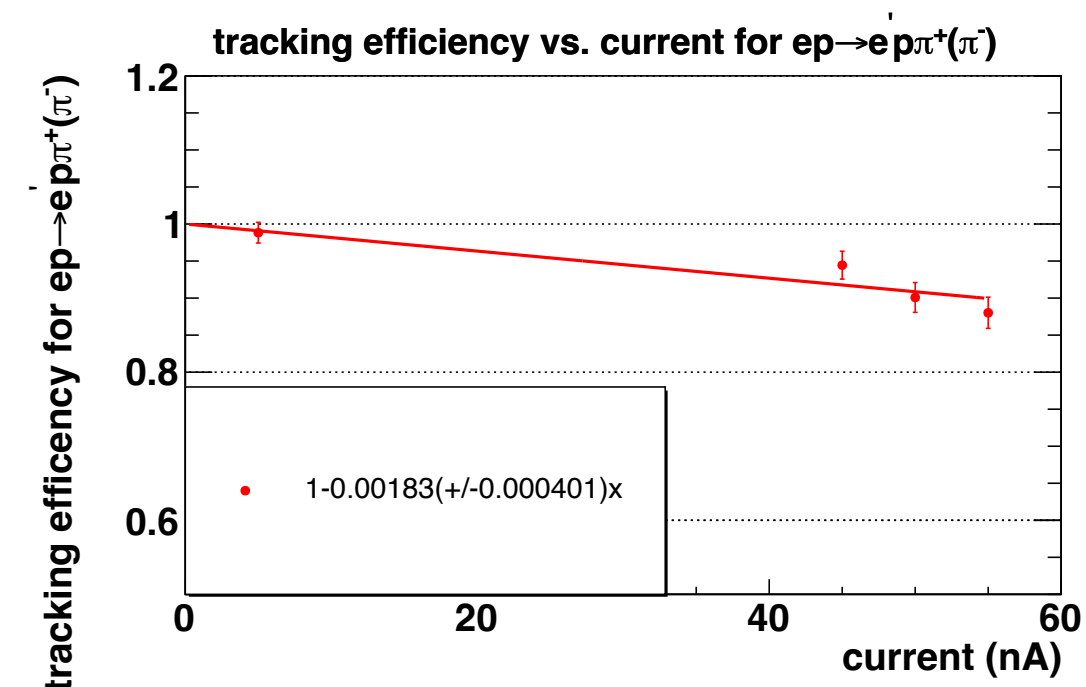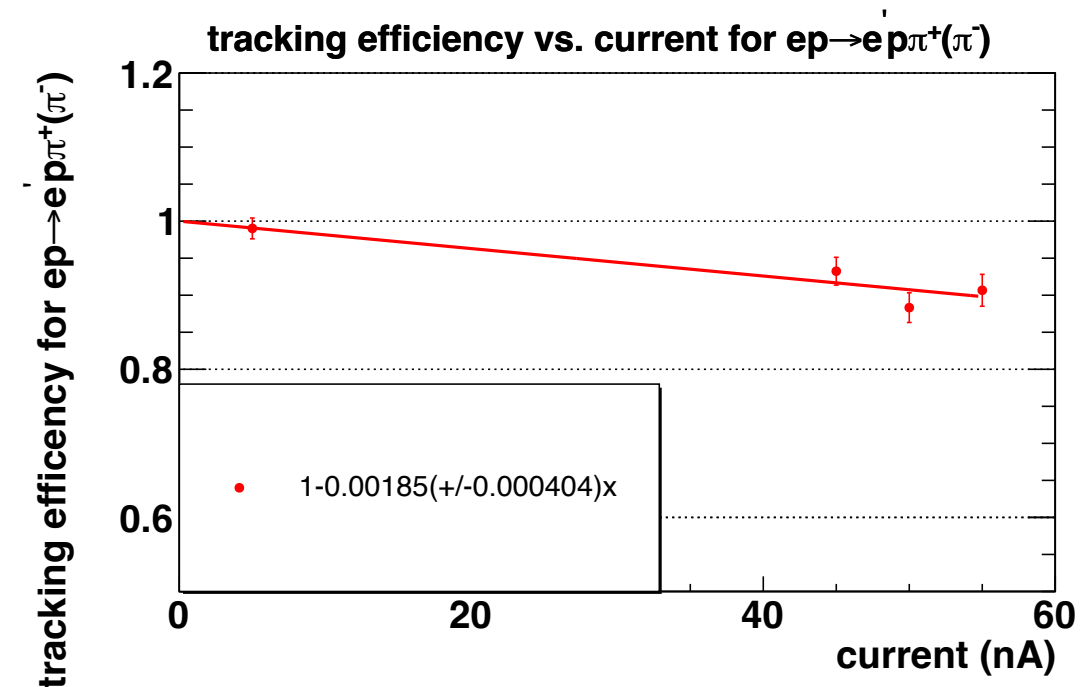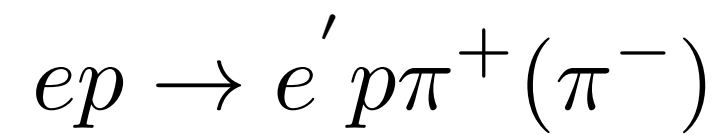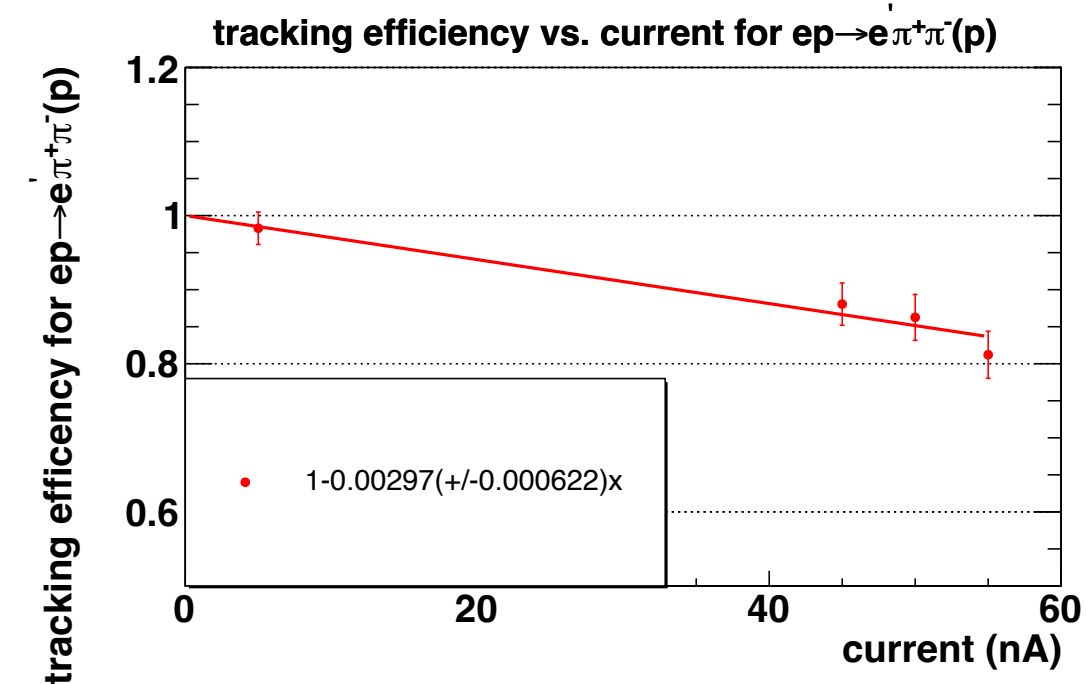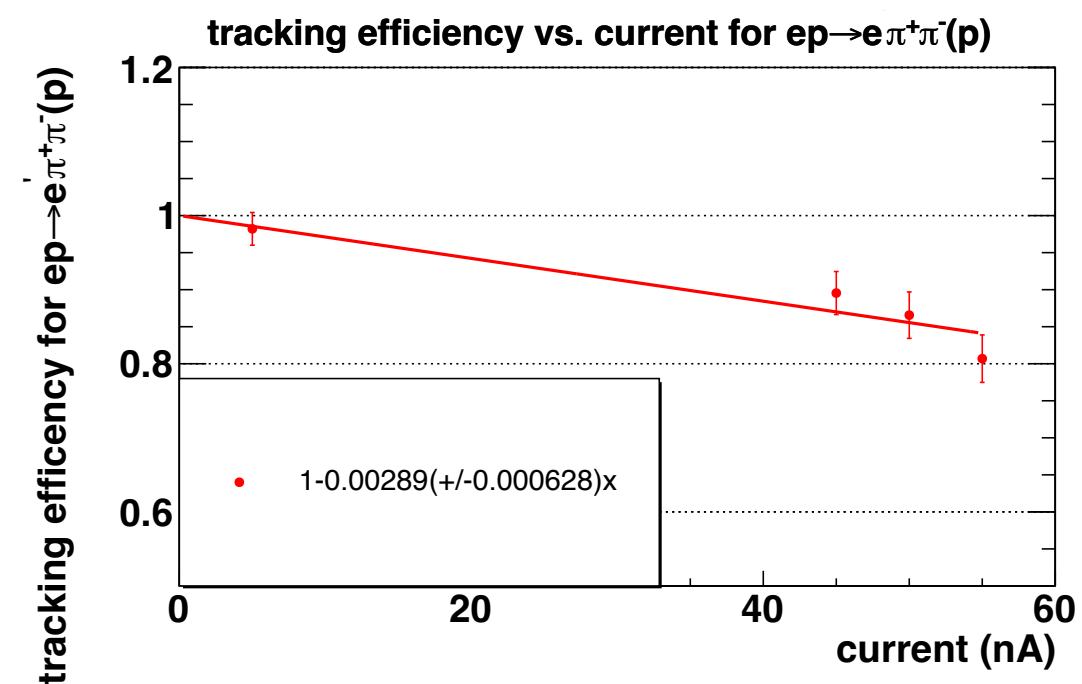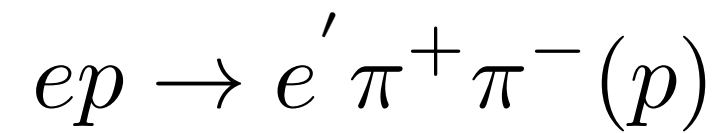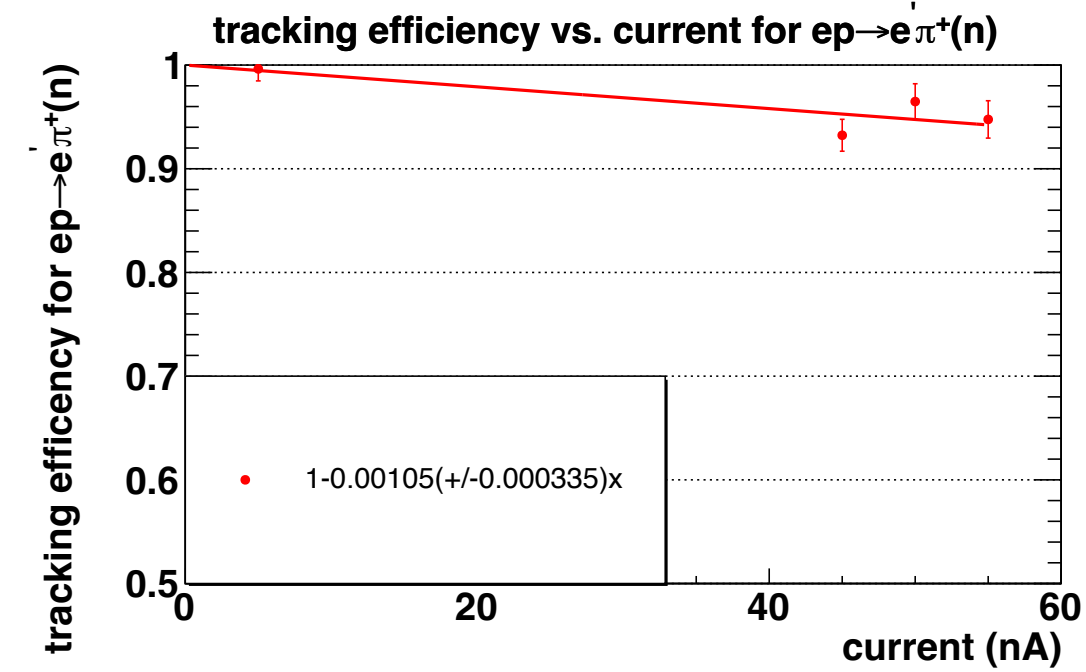
$$ep \to e' \pi^+ (n)$$

$$ep \to e' \pi^+ \pi^- (p)$$

$$ep \to e' p \pi^+ (\pi^-)$$

Efficiency = # of events for channel / # of events with electron

Efficiency for Physics Channels: Out-bending

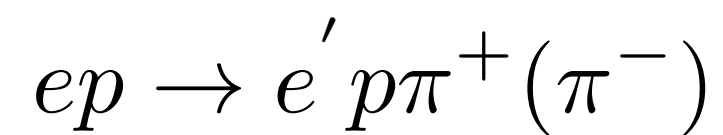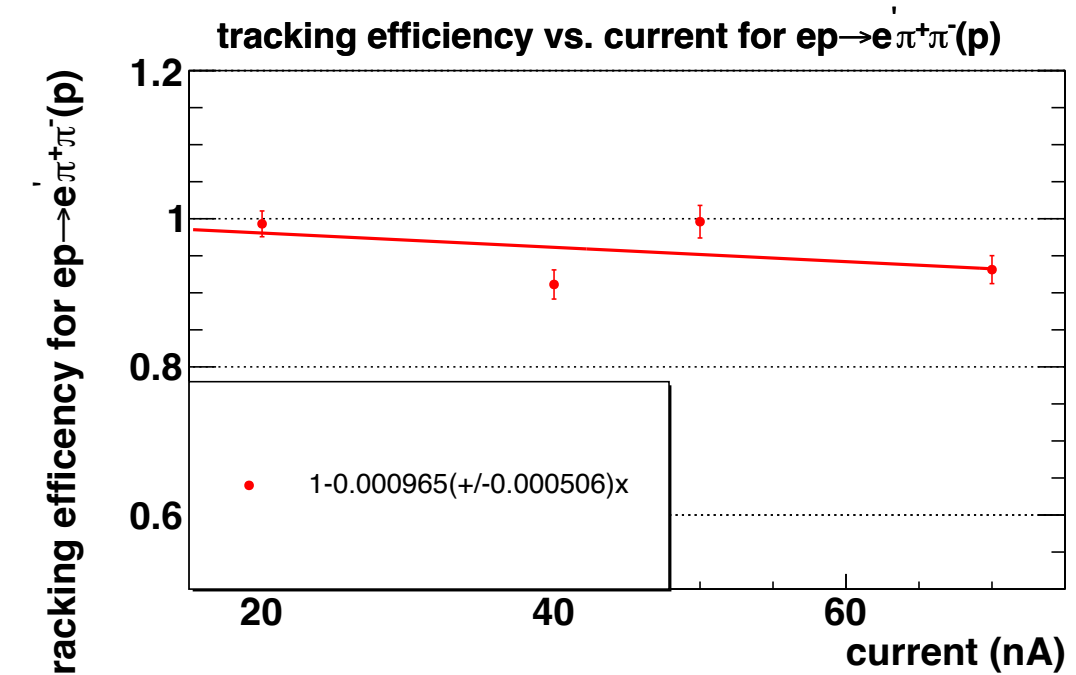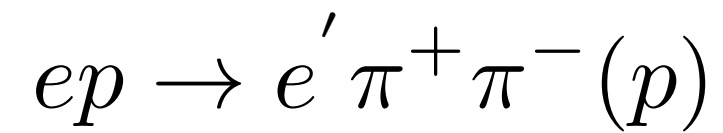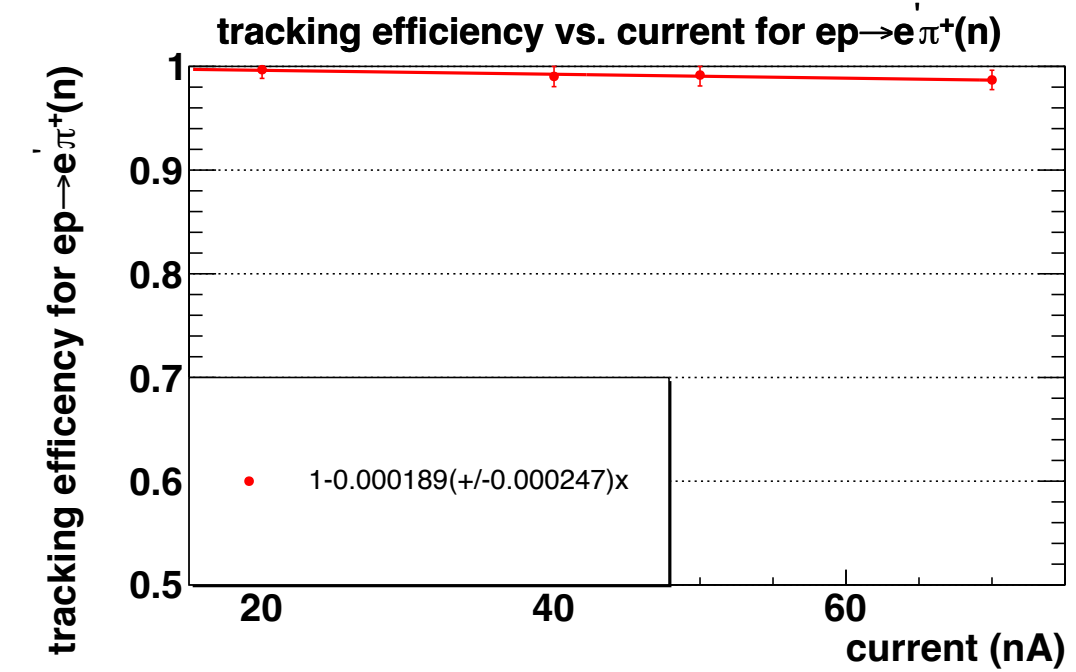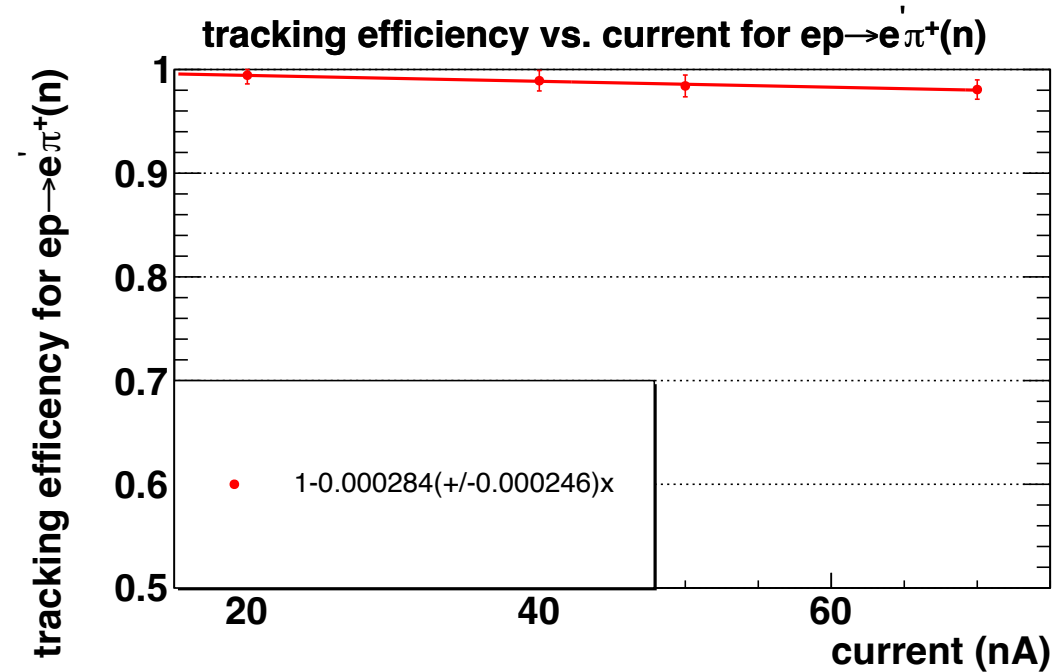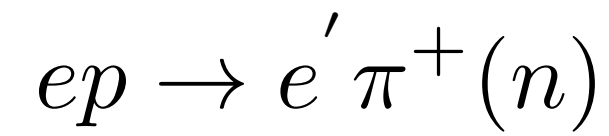**Old clustering**   **New clustering**

$ep \rightarrow e^{'}\pi^{+}(n)$

$ep \rightarrow e^{'}\pi^{+}\pi^{-}(p)$

$ep \rightarrow e^{'}p\pi^{+}(\pi^{-})$
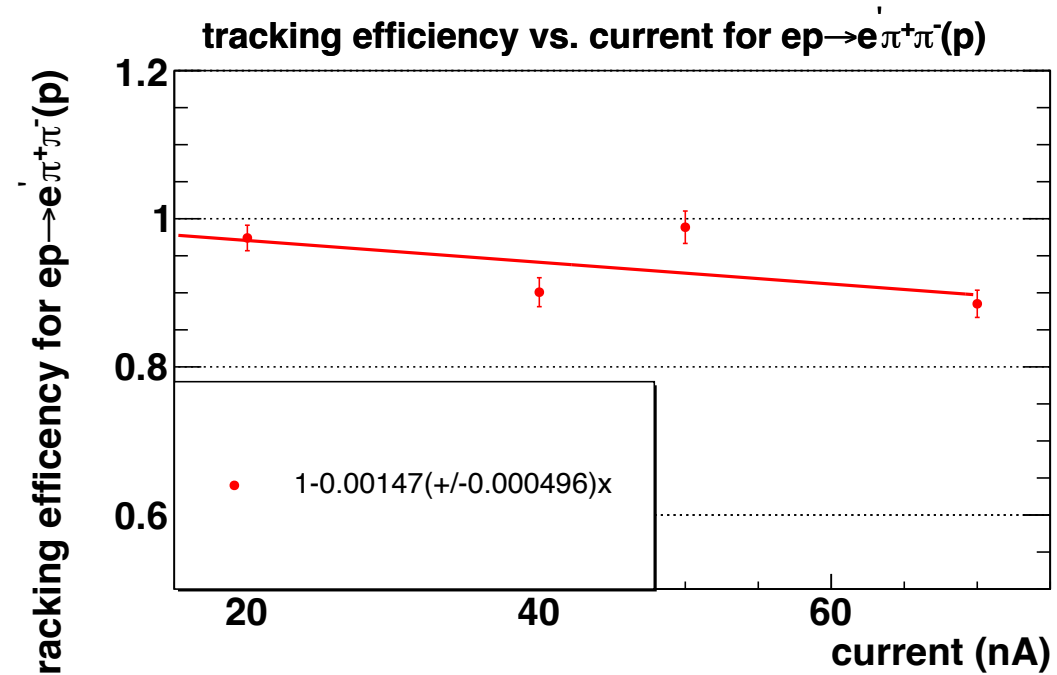
Discussion for Pruner

- Codes: https://github.com/JeffersonLab/coatjava/blob/development/reconstruction/dc/src/main/java/org/jlab/rec/dc/cluster/ClusterCleanerUtilities.java#L750-L800

- The pruner is used to trim hit clump, but it is kind of rough. It could causes that real hits are trimmed, and further clusters are lost.

- In principle, the pruner could be cancelled since the splitter is powerful to handle complicated hit clumps.

- However, if the pruner is cancelled, too many clusters are constructed for some rare cases of complicated hit clumps. It brings trouble to AI model for cluster combo prediction.

- Gagik is working on development of a new AI model with new features. With the new AI model, the pruner will be further studied to determine if it should be canceled.


With pruner

Cancel pruner
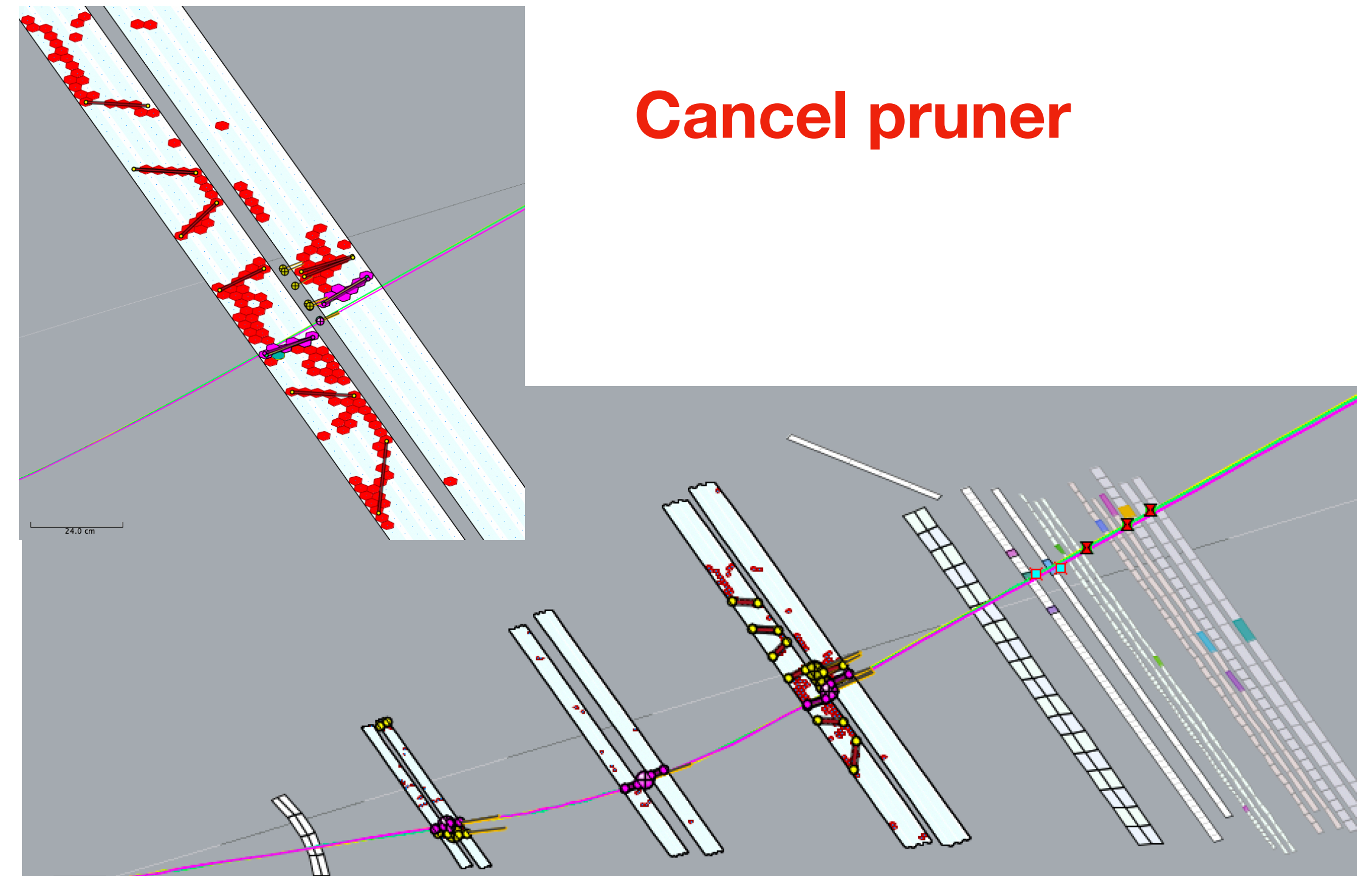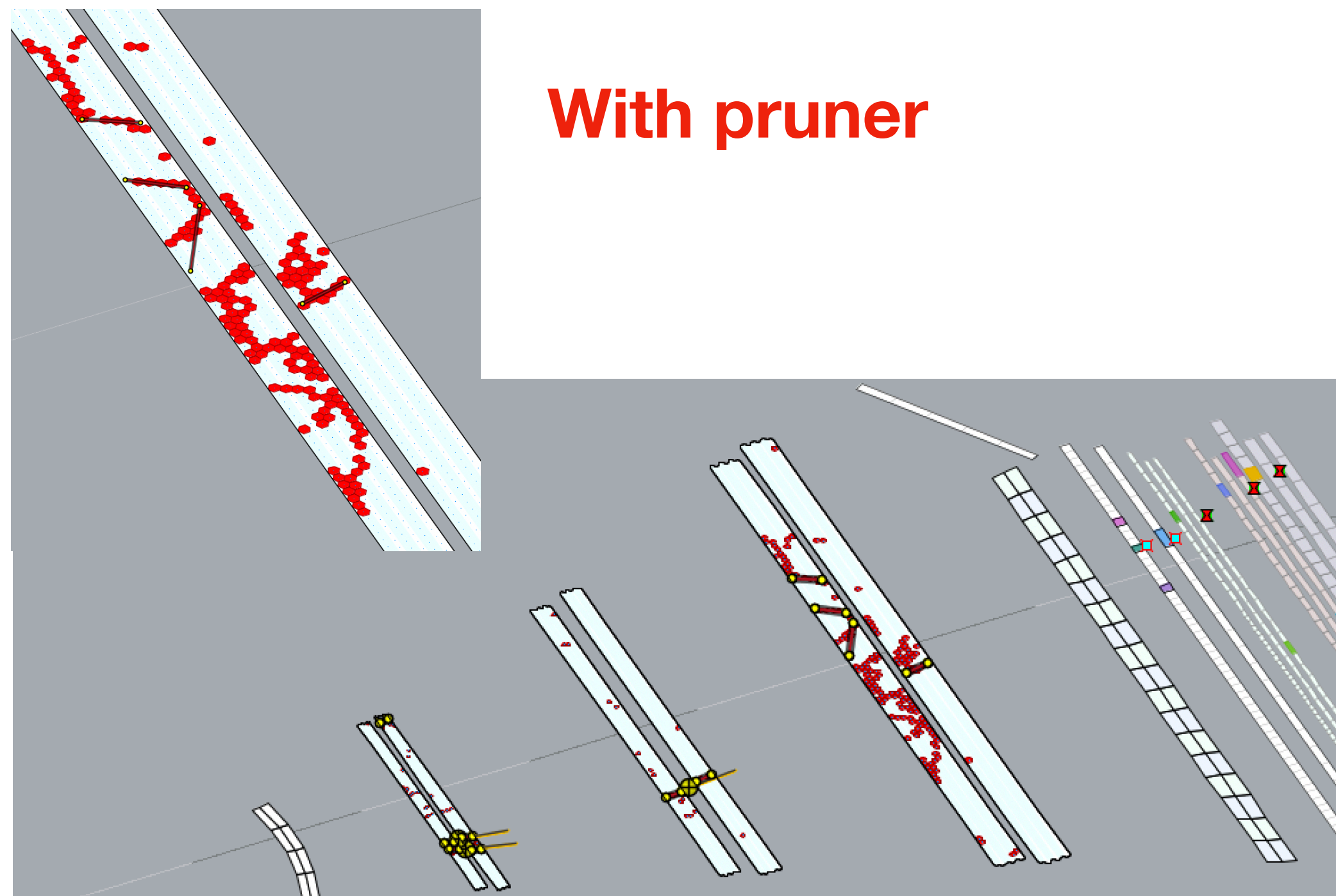
Summary

- With bug fixing and algorithm updating, new clustering further improves tracking efficiency.

- Overall, effects for out-bending are much more significant than in-bending. Slope of tracking efficiency is better than -0.1%/ns for RGA out-bending, while reaches about -0.12%/ns for RGA in-bending.

- Recently, Gagik has been working on a new AI model for cluster combo prediction with new features. It is supposed to recover some missing tracks, and further improve tracking efficiency.

- The pruner applied in clustering will be further investigated with the new AI model.

Backup Slides for Cancelling Pruner

Comparison of Clusters
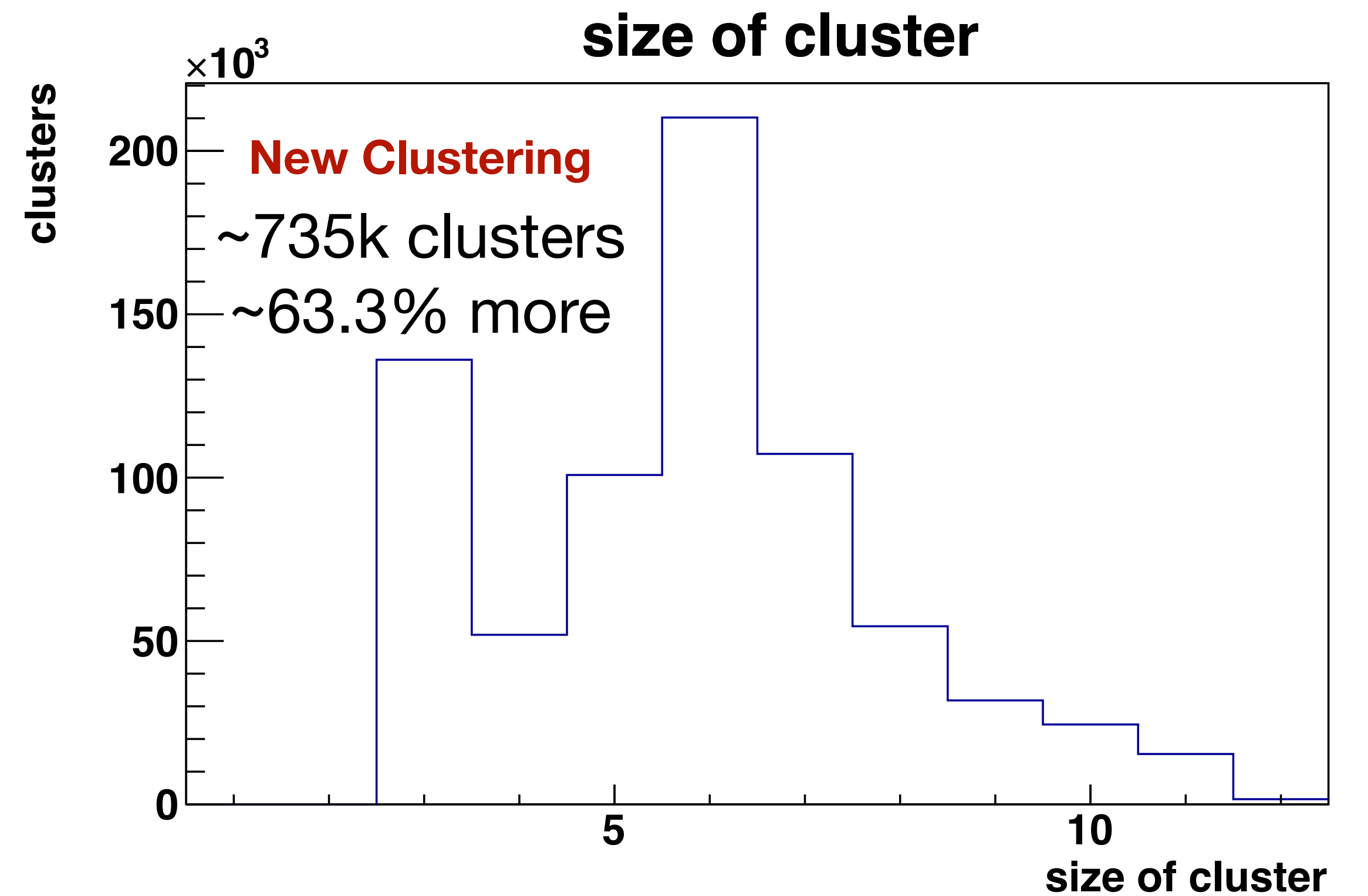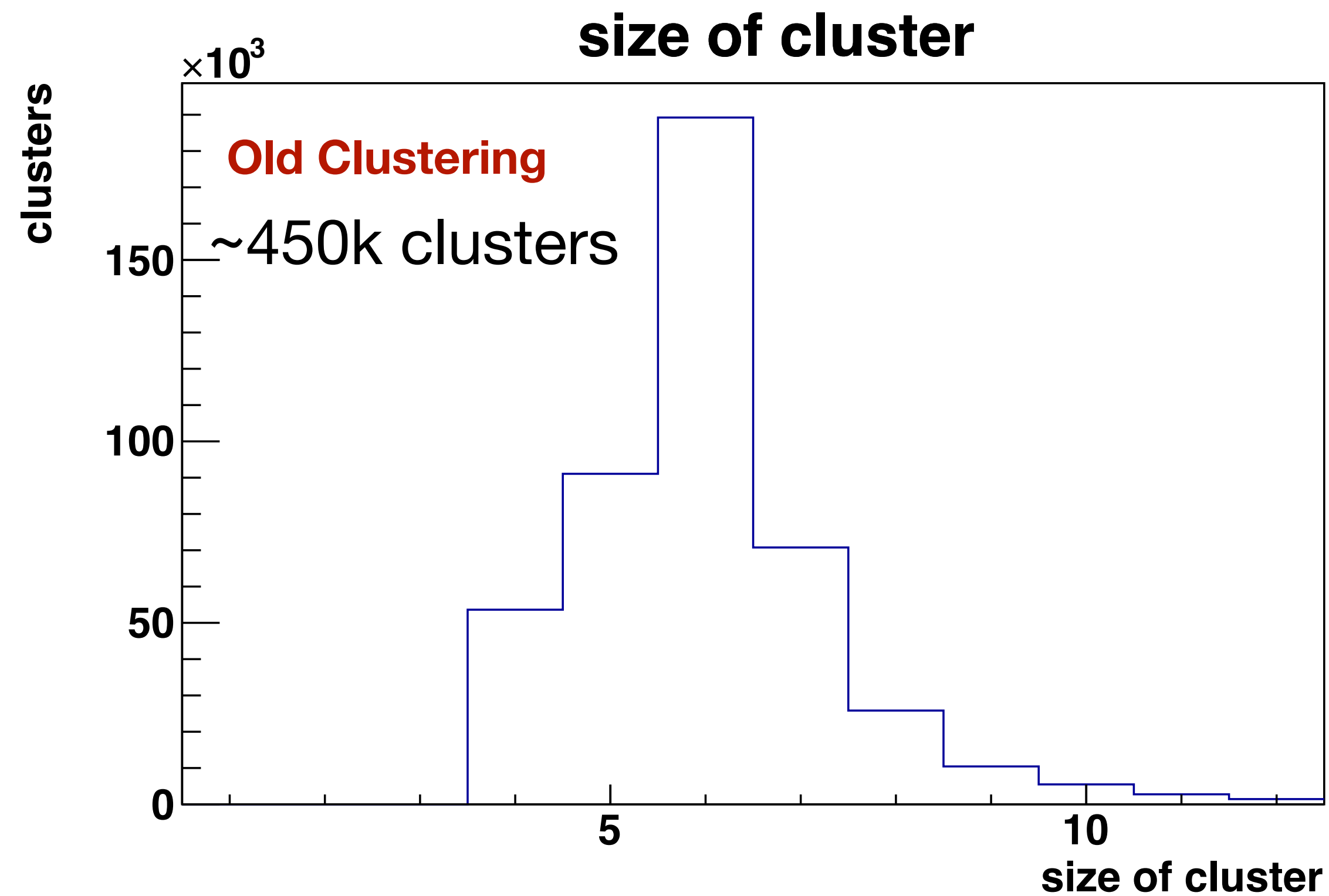
10000 events

Not just 3-hit clusters are involved, but also other-size clusters are increased.

Comparison of AI-predicted DC Cluster Combos

10000 events

# of predicted DC cluster combos

# of predicted DC cluster combos

Old Clustering

~20.7k combos

New Clustering

~25.0k combos

~2.5% more   ~28.6% more
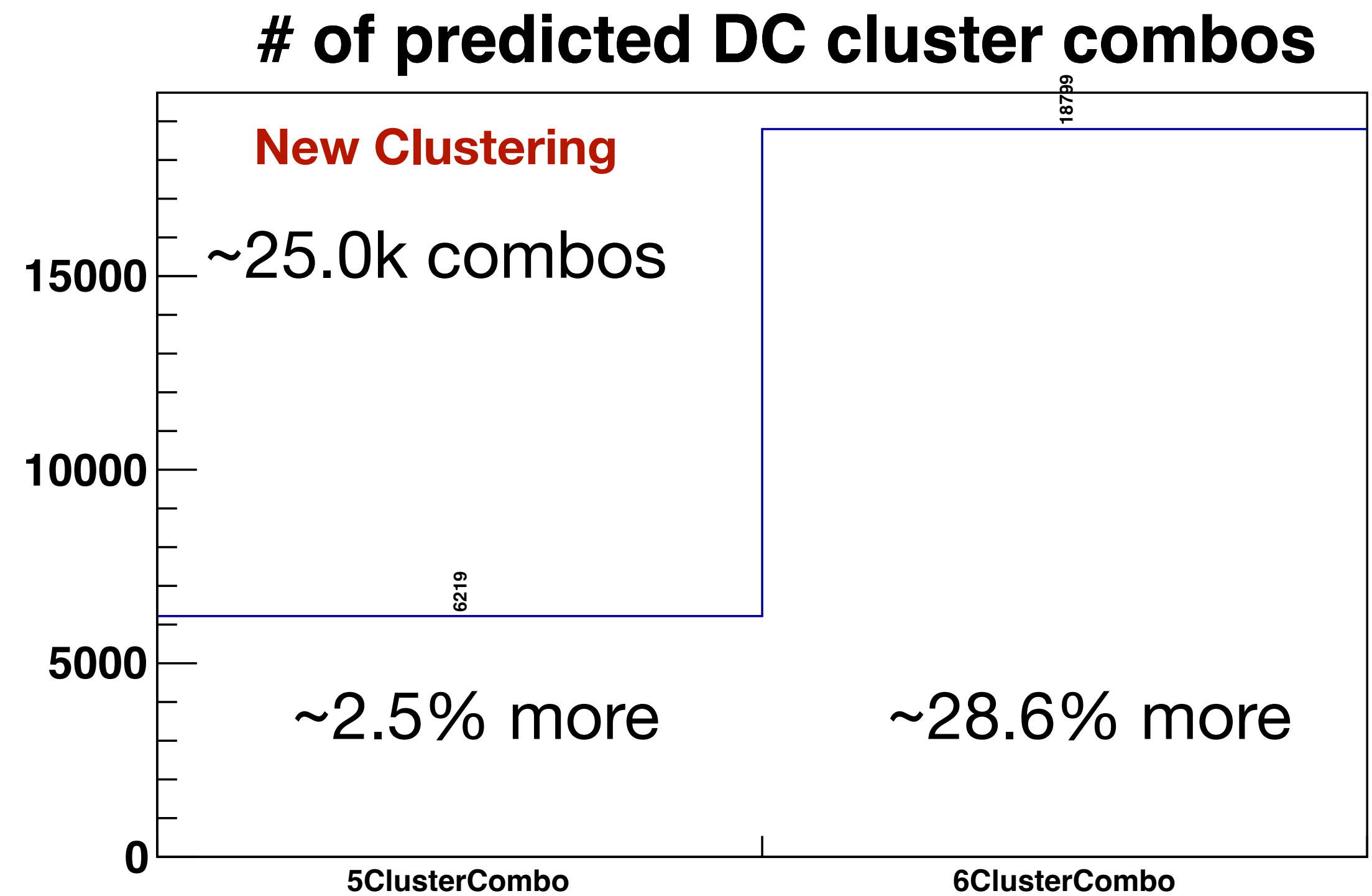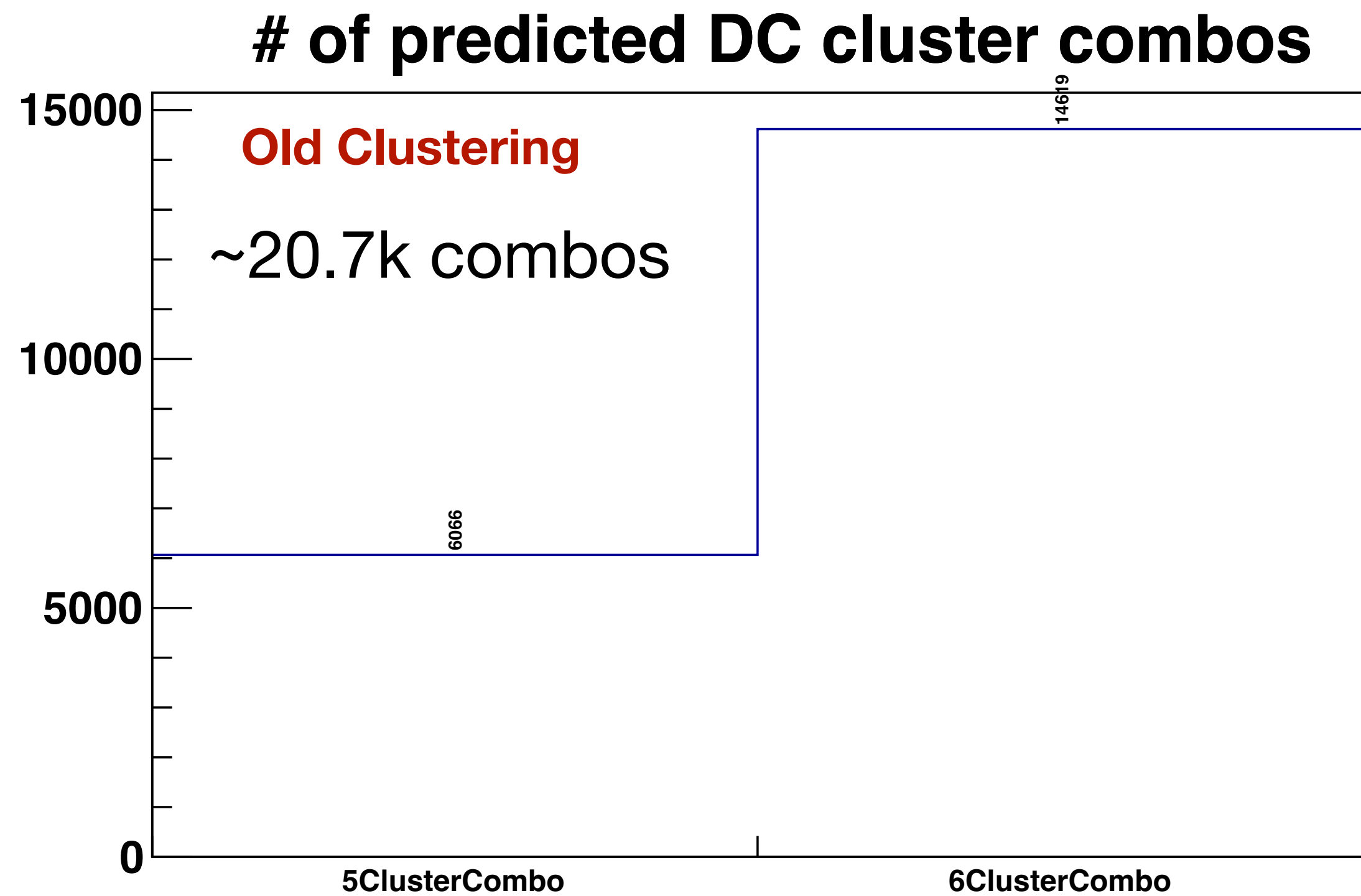
Strongly suggest to re-train AI model with new clustering.

Comparison of TB Tracks from Conventional Tracking
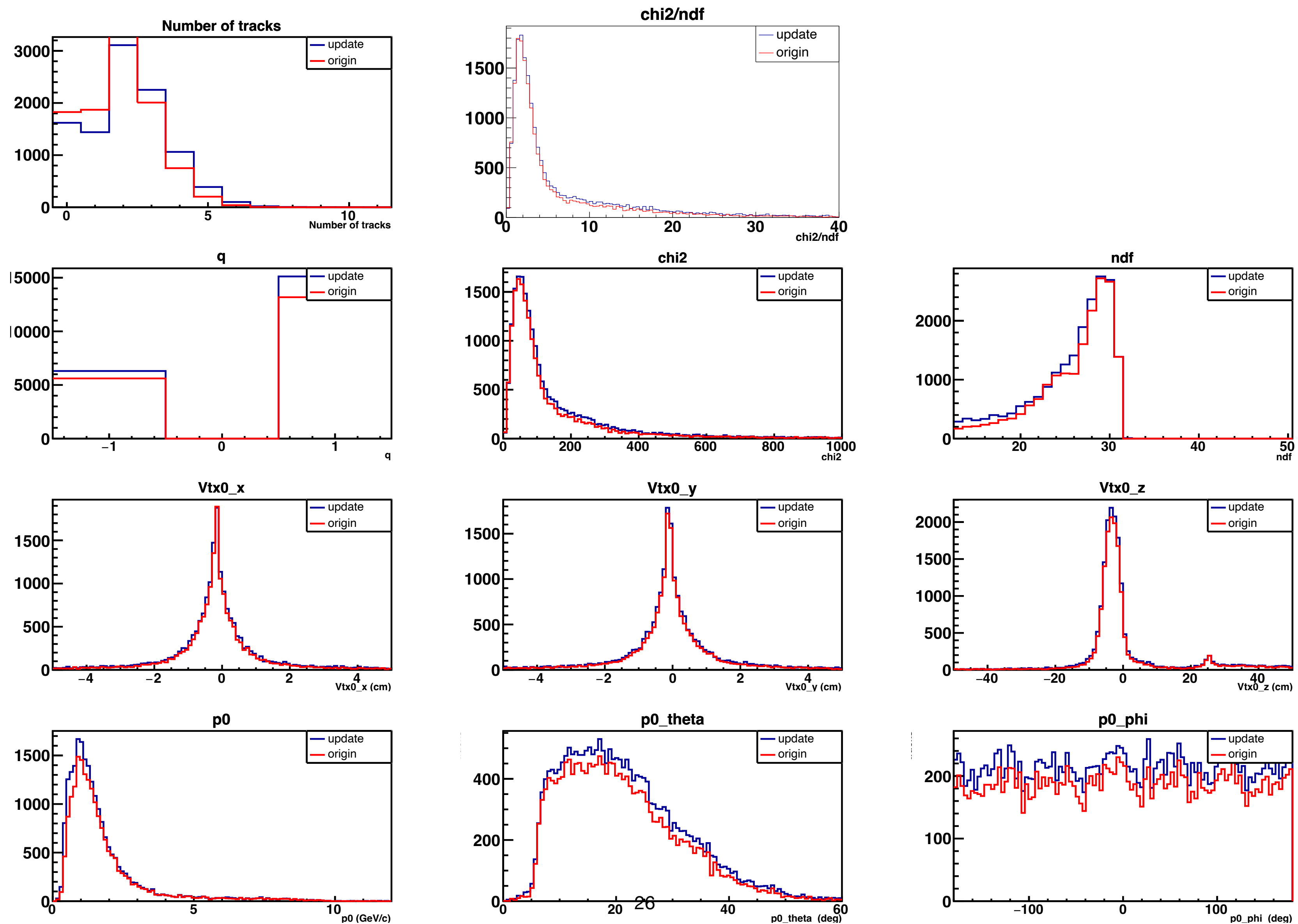
of HB tracks

1000 events	# of HB tracks	% of more tracks	# of AIHB tracks	% of more tracks
Before update	2072		1899	
Fix bug in the splitter	2142	3.4%	1942	2.3%
Categorize clusters and set different limits for different types of clusters	2346	13.2%	2014	6.1%
Update OverlappingClusterResolver() for selection overlapped clusters from splitter	2349	13.3%	2017	6.2%
Fix issue for hits sharing by clusters	2363	14.0%	2091	10.1%
Cancel pruner	2415	16.6%	2127	12.0%

cumulative

of TB tracks

1000 events	# of TB tracks	% of more tracks	# of AITB tracks	% of more tracks
Before update	1870		1752	
Fix bug in the splitter	1934	3.4%	1781	1.7%
Categorize clusters and set different limits for different types of clusters	2090	11.8%	1831	4.5%
Update OverlappingClusterResolver() for selection overlapped clusters from splitter	2090	11.8%	1832	4.6%
Fix issue for hits sharing by clusters	2098	12.2%	1884	7.5%
Cancel pruner	2139	14.4%	1926	9.9%

cumulative