

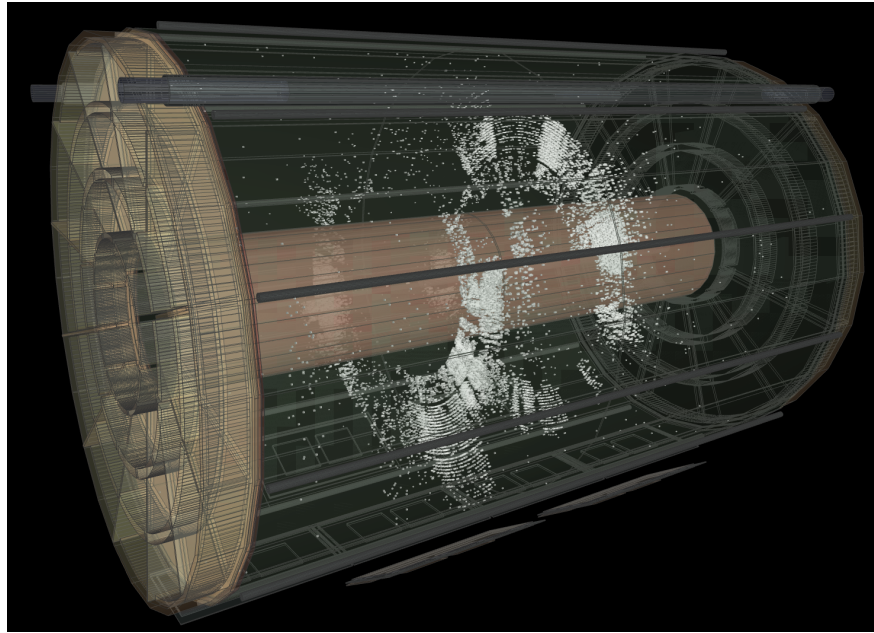
## TPC Distortion Stripe Map, Hit Maps, and Distortion Magnitude Maps

Benjamin Kimelman (Vanderbilt University), Ross Corliss (Stony Brook University)

### General Info

All code for generating these figures and accompanying input files can be found in [https://github.com/bkimelman/QM2023\\_macros](https://github.com/bkimelman/QM2023_macros)

## 7 Run11011 Event Display



**Figure 1:** Event display of Run11011

8 Figure 1 shows an event display of Run11011, which was generated using the sPHENIX  
9 online event display tool (<https://www.sphenix.bnl.gov/edisplay/>). The ".json" file  
10 which went into this event display was created by Thomas Marshall and Aditya  
11 Prasad Dash, and was used to generate an animation of this run which has already  
12 been approved as preliminary (see <https://indico.bnl.gov/event/20190/> for the indico  
13 page and [https://indico.bnl.gov/event/20190/contributions/79225/attachments/48895/  
14 83248/sPHENIX\\_TPC\\_Multiple\\_Collisions\\_Event\\_Displays-3.pdf](https://indico.bnl.gov/event/20190/contributions/79225/attachments/48895/83248/sPHENIX_TPC_Multiple_Collisions_Event_Displays-3.pdf) for the accompanying note).  
15 A still figure is necessary for posters and this helps to demonstrate how the electrons from the  
16 diffuse laser flash travel as a sheet through the TPC.

## 17 Truth Map Creation Summary

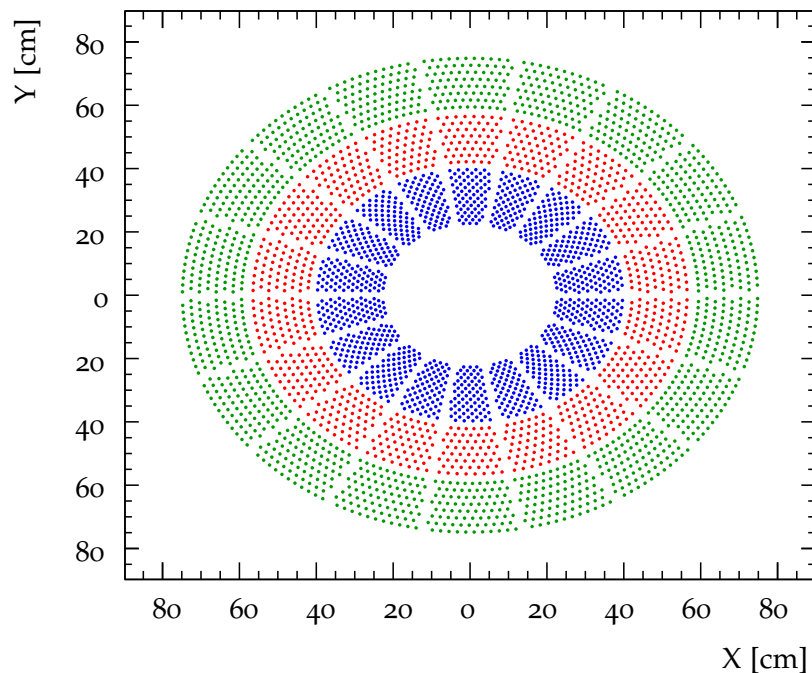
18 First, the stripe pattern must be made and put into a TTree, which uses the nominal centers of the  
19 Aluminum stripes from simulation. This can be done using the *make\_hits.C* macro, which will  
20 generate the root file *inputFiles/truthSectors.root*. This macro follows the exact same procedure  
21 for determining the stripe positions as the official sPHENIX code, which currently does this same  
22 procedure each time the code is run. This output file contains a TTree with the stripID, 3-vector  
23 position, and sector number. Next, *drawTruthPatternXY.C* is run, which selects either the North  
24 or South side (depending on the argument passed, South was used for 2) and draws the center  
25 positions of the stripe pattern.

26 The full series of commands to generate 2 starting in the base directory of the repository is:

```

root -l
gROOT-> LoadMacro("sPhenixStyle.C")
SetsPhenixStyle()
.L make_hits.C
getAllHits()
.x drawTruthPatternXY.C(false)

```



**Figure 2:** Map of simulated positions of Aluminum stripes on the South side of the Central Membrane.

## 27 Hit Displays

28 Figures 3 show the pad row vs phi with the color axis as the ADC for two runs with the diffuse  
 29 laser: 11011 (top) and 11028 (bottom). Overlaid on both in open circles are the Al stripe positions  
 30 from the Central Membrane, which show a reasonable match to the hits.

31 In order to  
 32 obtain these, Fun4All must be run on the PRDFs using two modules: *TpcRawDataDecoder*  
 33 and *TpcClusterizer*. The latter has minor additions to store the hit and cluster data in TTrees,  
 34 which get written to a file for ease of access later (modified code can be found in [https://github.com/bkimelman/coresoftware/tree/clusterizer\\_debugMode](https://github.com/bkimelman/coresoftware/tree/clusterizer_debugMode)).  
 35 One file is used for each run:

36 /sphenix/lustre01/sphnxpro/commissioning/tpc/beam/TPC\_ebdc07\_beam-00011011-0000.prd  
37 and  
38 /sphenix/lustre01/sphnxpro/commissioning/tpc/beam/TPC\_ebdc07\_beam-00011028-0000.prd.  
39 The macro /sphenix/u/bkimelman/macros/detectors/sPHENIX/Fun4All\_clusterizer.C was used  
40 to run over these files.

41 The output files from the Fun4All macro are used as inputs to *drawPad33.C* to get the hit  
42 positions along with *inputFiles/truthSectors.root* to get the stripe truth positions. These files  
43 have been copied to the repository and are *inputFiles/Run11011\_ebdc07\_clusterizer.root* and  
44 *inputFiles/Run11028\_ebdc07\_clusterizer.root*.

45 The full series of commands to generate 3 starting in the base directory of the repository is:

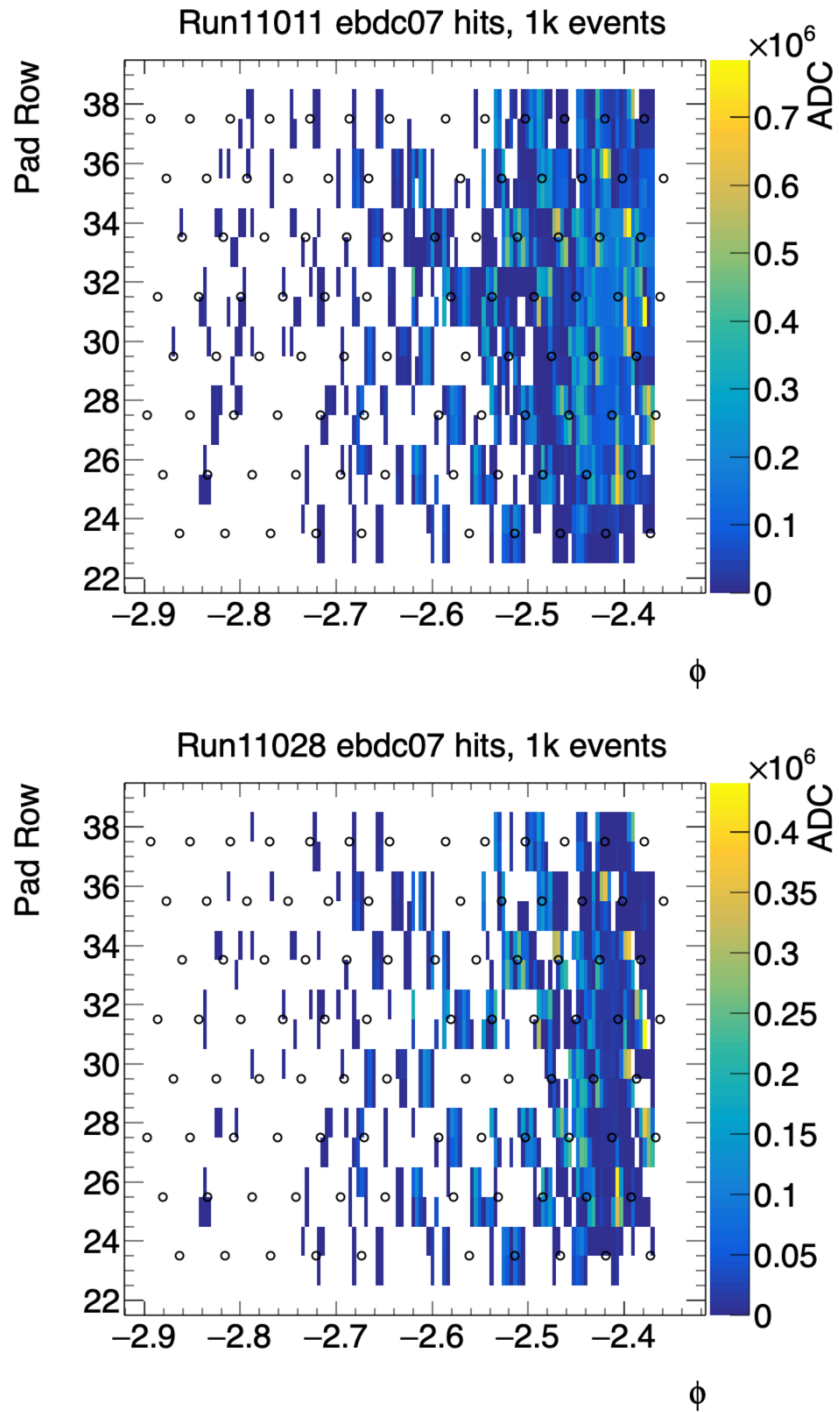
```
root -l  
gROOT-> LoadMacro("sPhenixStyle.C")  
SetsPhenixStyle()  
.L make_hits.C  
getAllHits()  
.x drawPad33.C
```

## 46 Predicted Time-Ordered Distortions

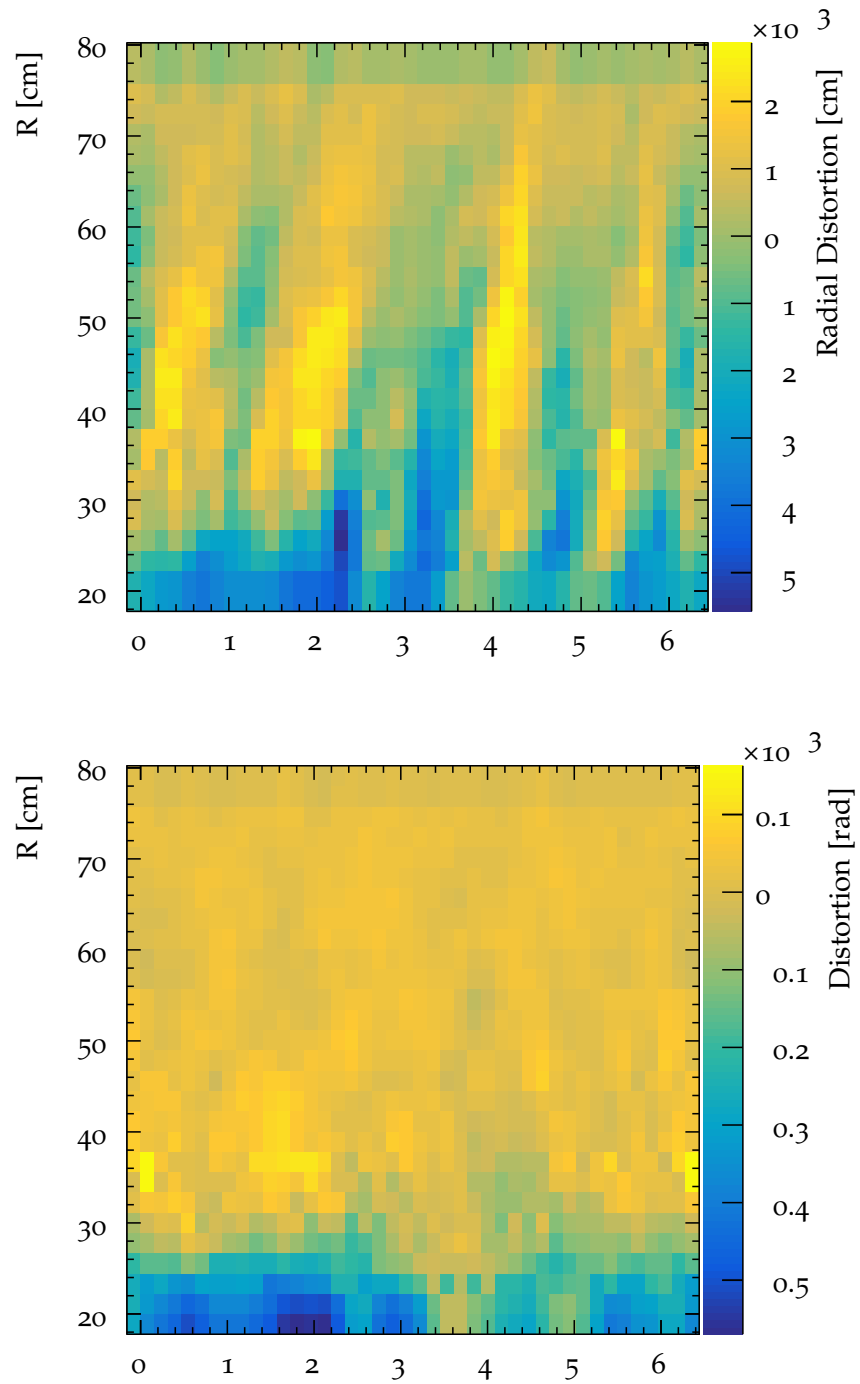
47 Ross Corliss generated predicted 3D static and fluctuation distortions, though the fluctuation  
48 ones are static+fluctuation. In order to get the predicted distortion at the Central Membrane,  
49 the distortion at the position of the Central Membrane for static and static+fluctuation must be  
50 obtained, and the fluctuation component must be isolated using a subtraction of bin contents.  
51 These files can be found on SDCC at /sphenix/user/rcorliss/distortion\_maps/2023.02/  
52 *Summary\_hist\_mdc2\_UseFieldMaps\_AA.event\_0\_bX99528306\_5.distortion\_map.hist.root* (static dis-  
53 tortion) and /sphenix/user/rcorliss/distortion\_maps/2023.02/*TimeOrderedDistortions.root* (fluc-  
54 tuation distortion), with copies of these located in the repository at *inputFiles/staticInput.root*  
55 and *inputFiles/fluctInput.root* respectively.

56 The full series of commands to generate 4 starting in the base directory of the repository is:

```
root -l  
gROOT-> LoadMacro("sPhenixStyle.C")  
SetsPhenixStyle()  
.x drawInputDist.C
```



**Figure 3:** (Top) All hits above threshold for Run11011 ebdc07, first 1000 events. (Bottom) All hits above threshold for Run11028 ebdc07, first 1000 events.



**Figure 4:** (Top) Predicted radial distortion. (Bottom) Predicted phi distortion.