ACCEPTANCE IN FORWARD DETECTORS

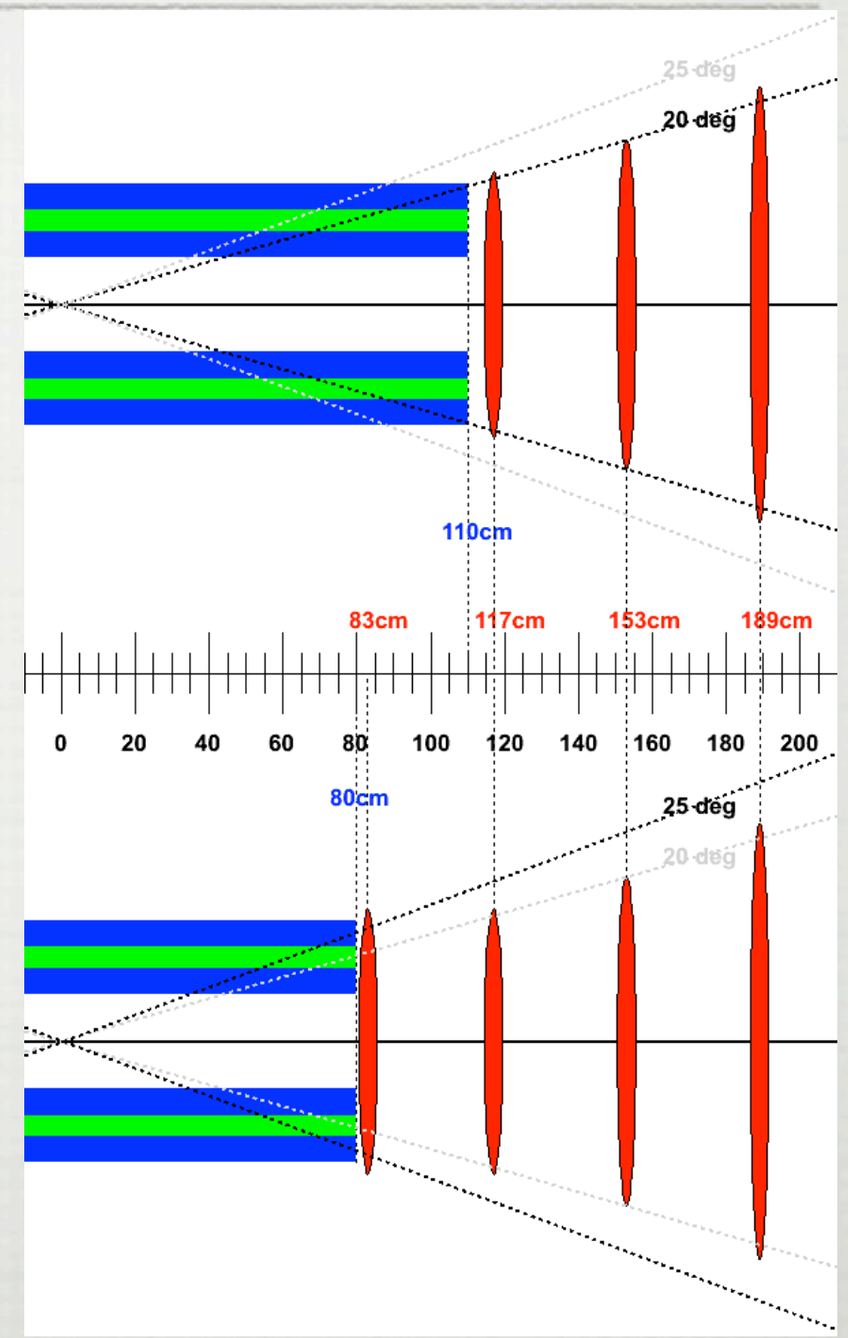
RADOSLAW KARABOWICZ

GSI, DARMSTADT

XXXII PANDA CM, MARCH 2010

PANDA GEOMETRIES

- LONG CENTRAL TRACKER:
  - STT coverage down to ~20deg
  - GEM coverage from ~20Deg down
- SHORT CENTRAL TRACKER:
  - STT coverage down to ~25deg
  - GEM coverage from ~25Deg down

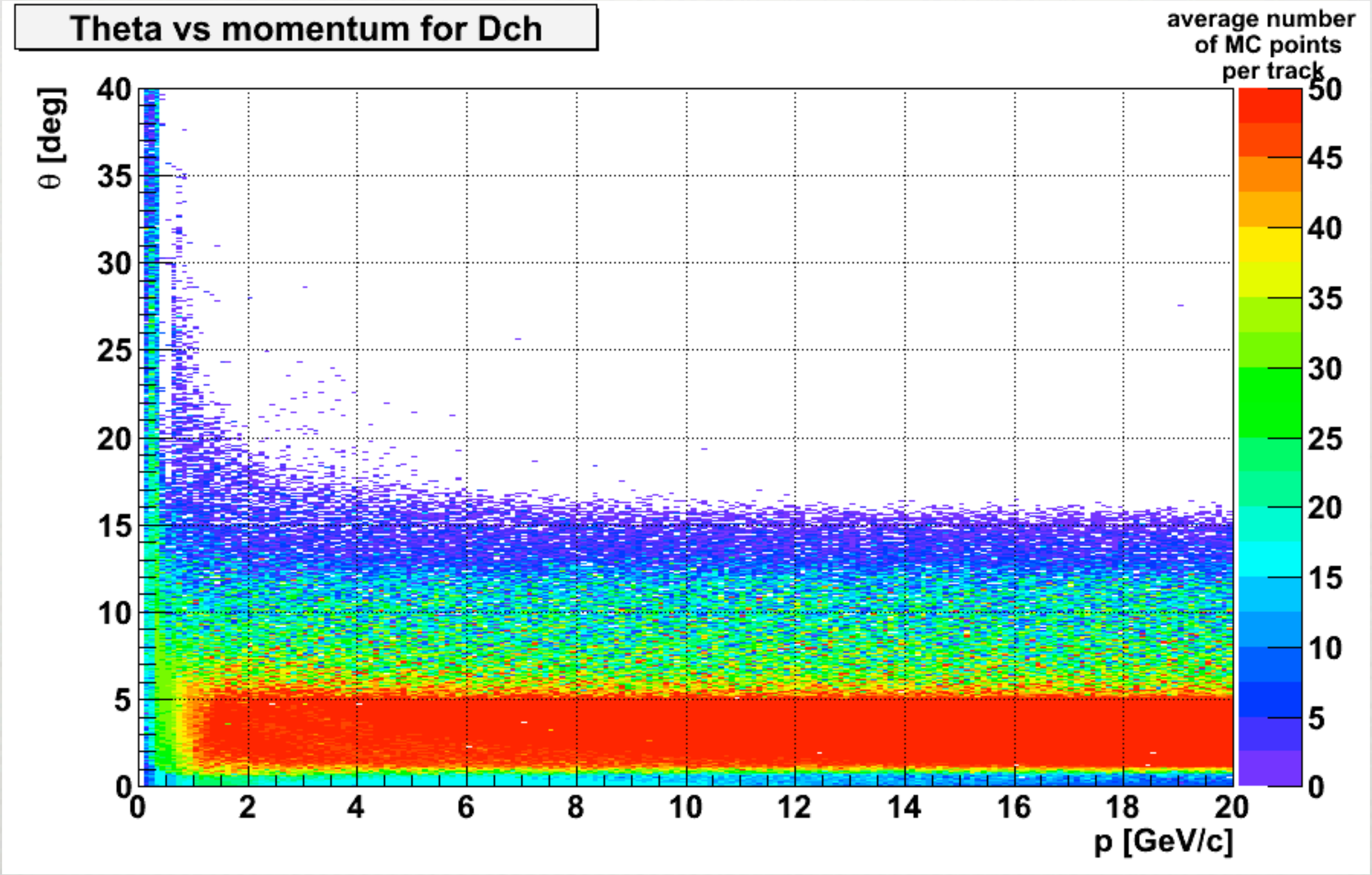


SIMULATIONS

- ☐ 1 MILLION MUONS SHOT IN TWO COMPARED GEOMETRIES
- ☐ FROM VERTEX (0cm,0cm,0cm)
- ☐ WITH EVENLY DISTRIBUTED MOMENTA:
  - $|p| \in (0.1\text{GeV/c}, 20\text{GeV/c})$
  - $\vartheta \in (0°, 40°)$
  - $\varphi \in (0°, 360°)$

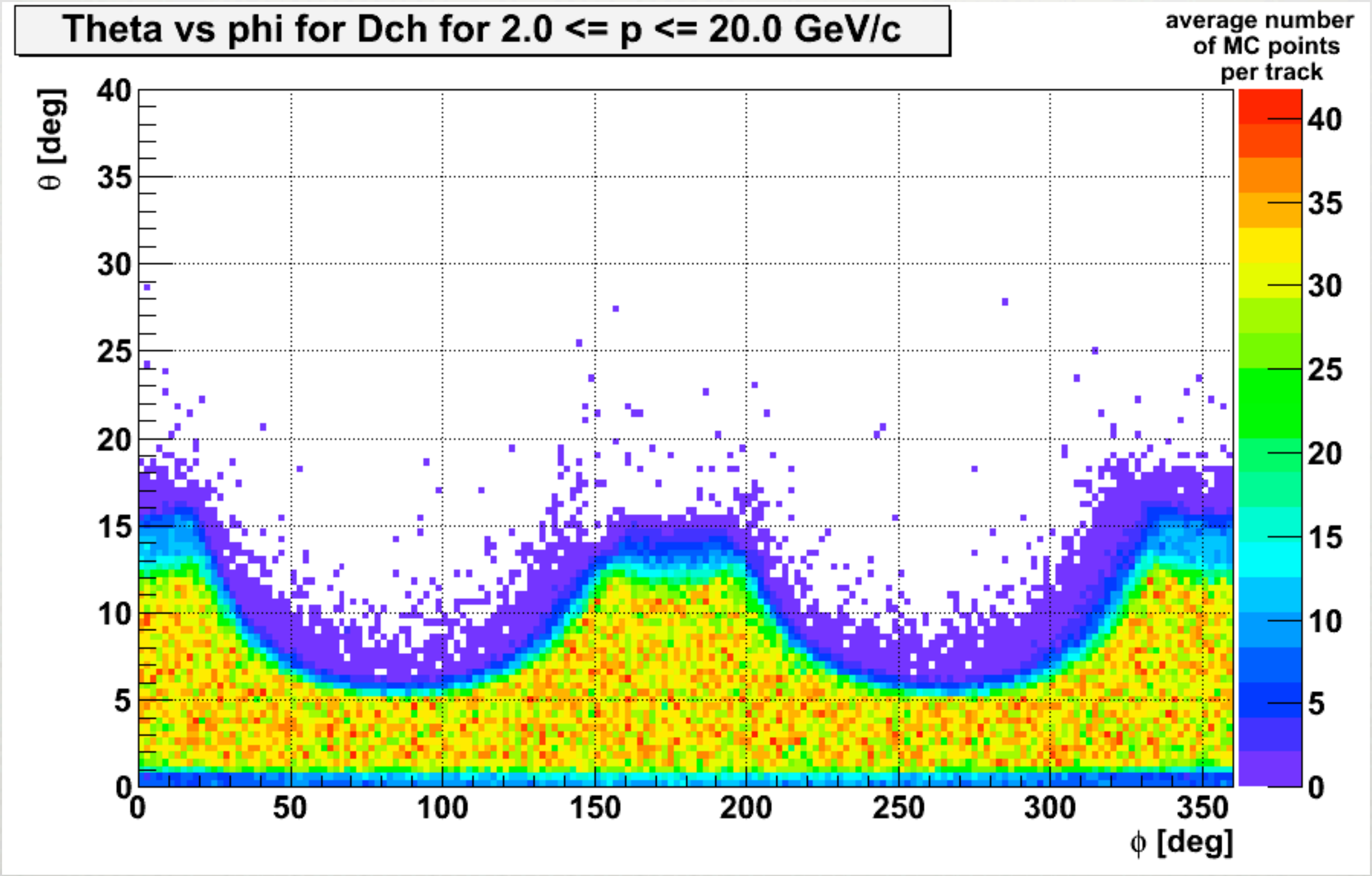
ACCEPTANCE MAP, DCH,

Θ VS MOMENTUM



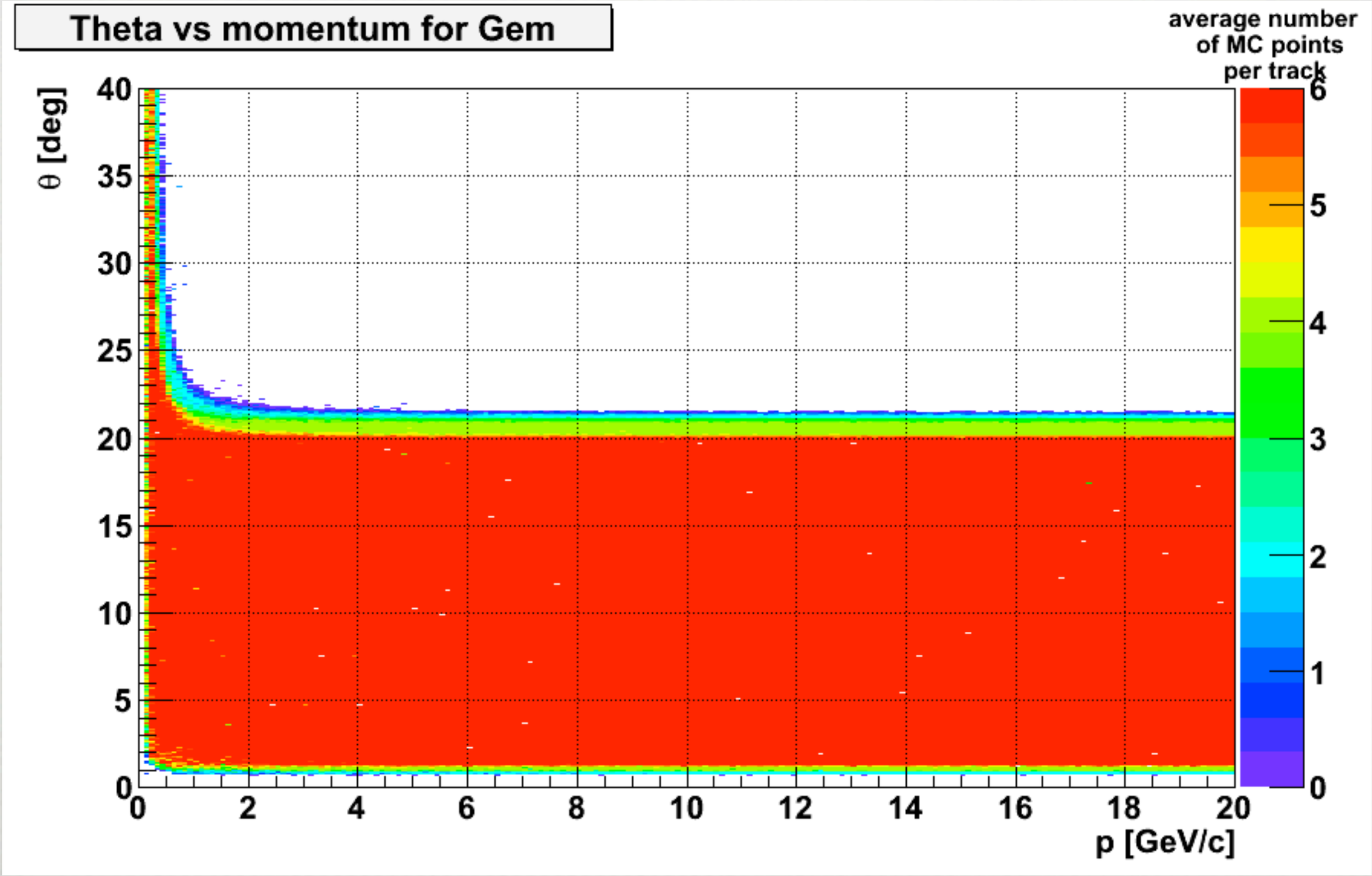
ACCEPTANCE MAP, DCH,

Θ VS Φ



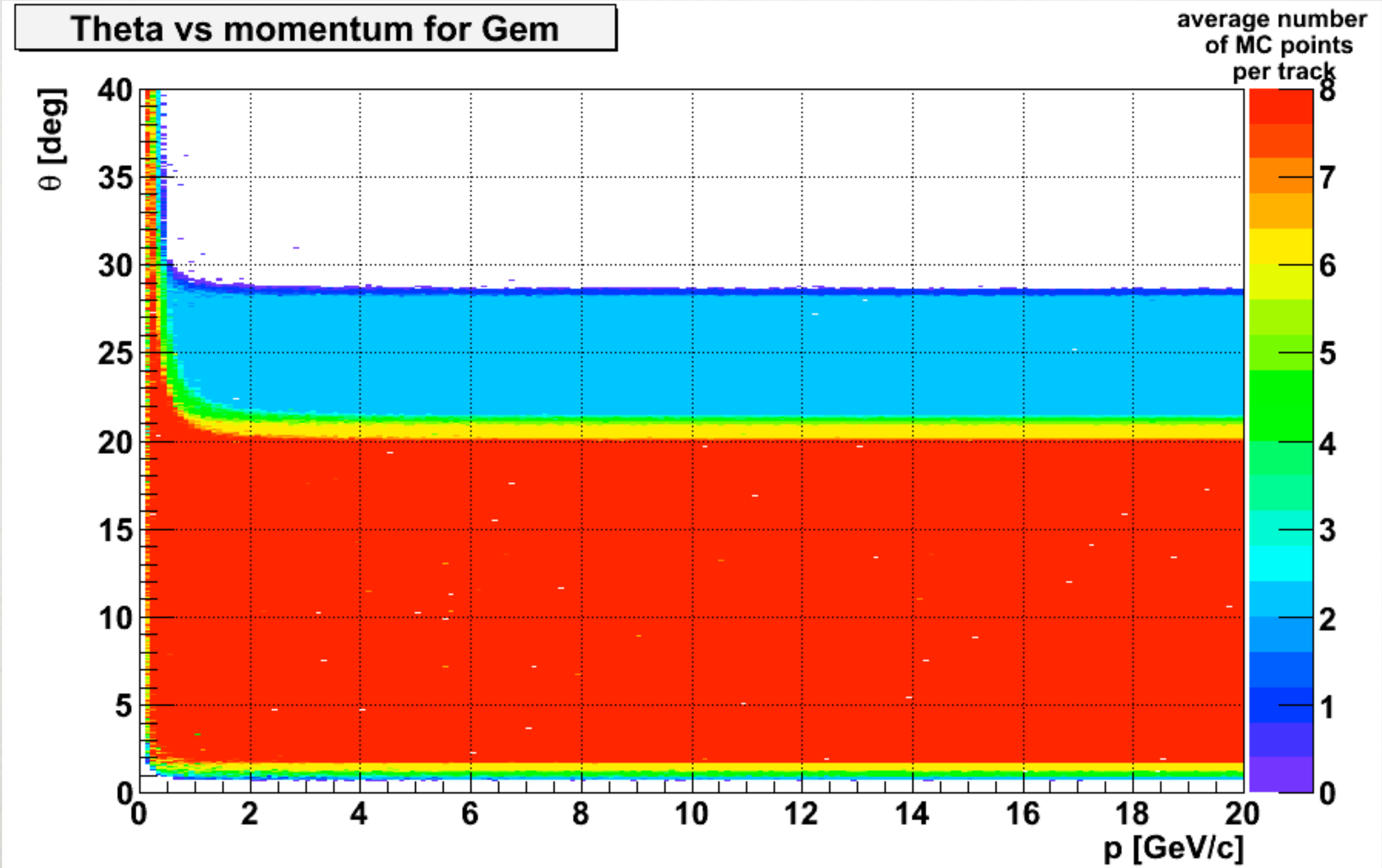
ACCEPTANCE MAP, LONG, GEM,

Θ VS MOMENTUM



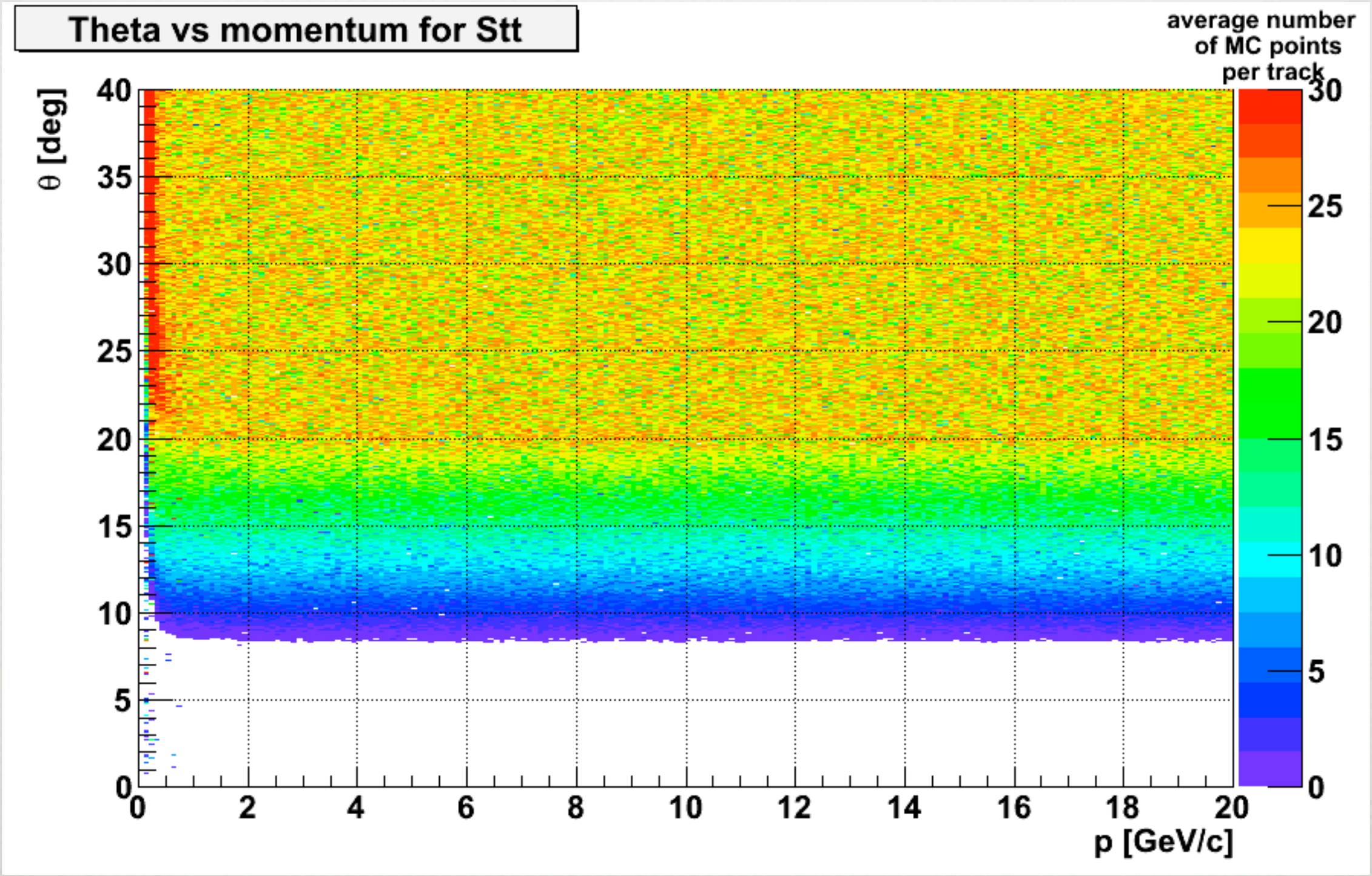
ACCEPTANCE MAP, SHORT, GEM,

Θ VS MOMENTUM



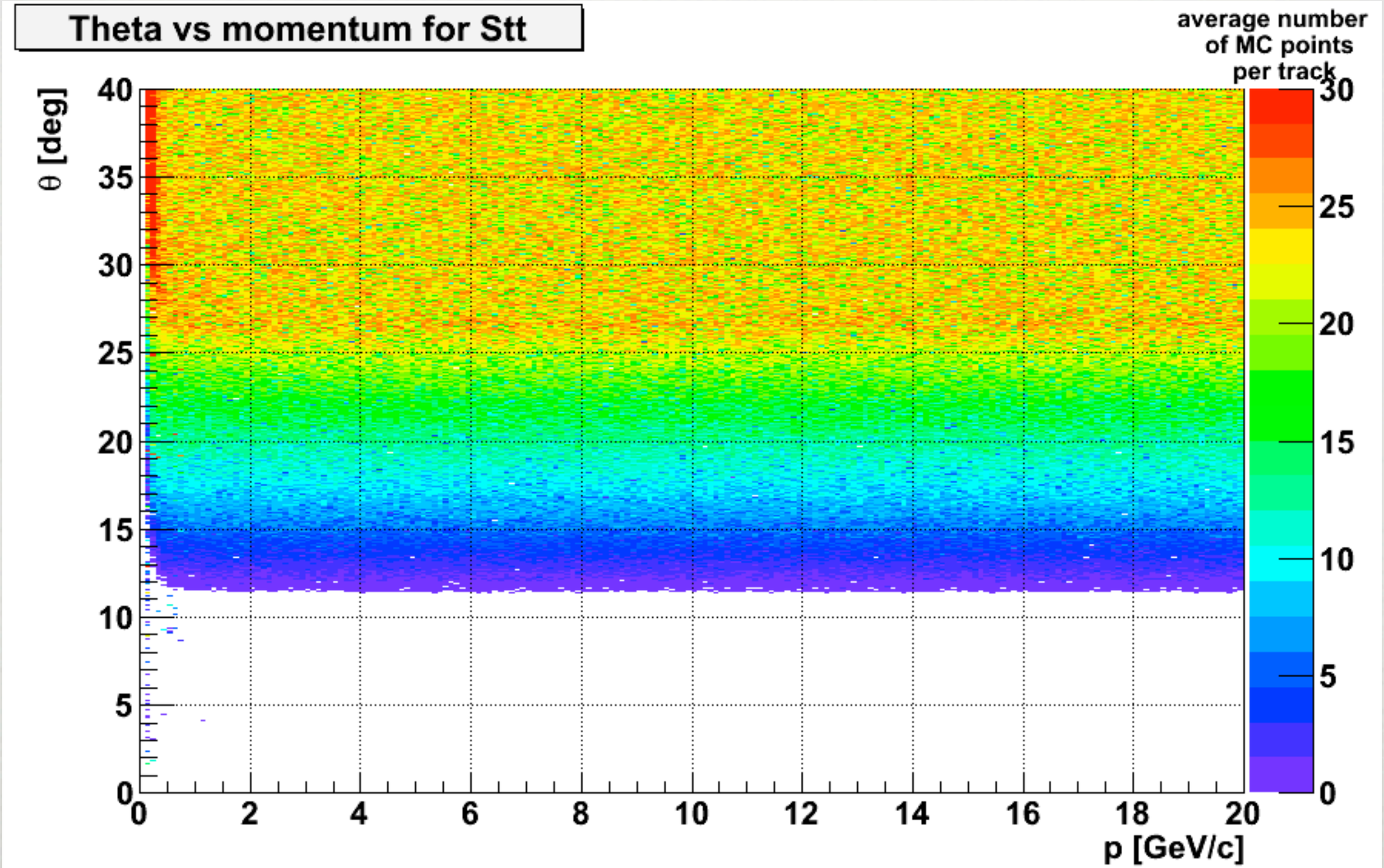
ACCEPTANCE MAP, LONG, STT,

Θ VS MOMENTUM



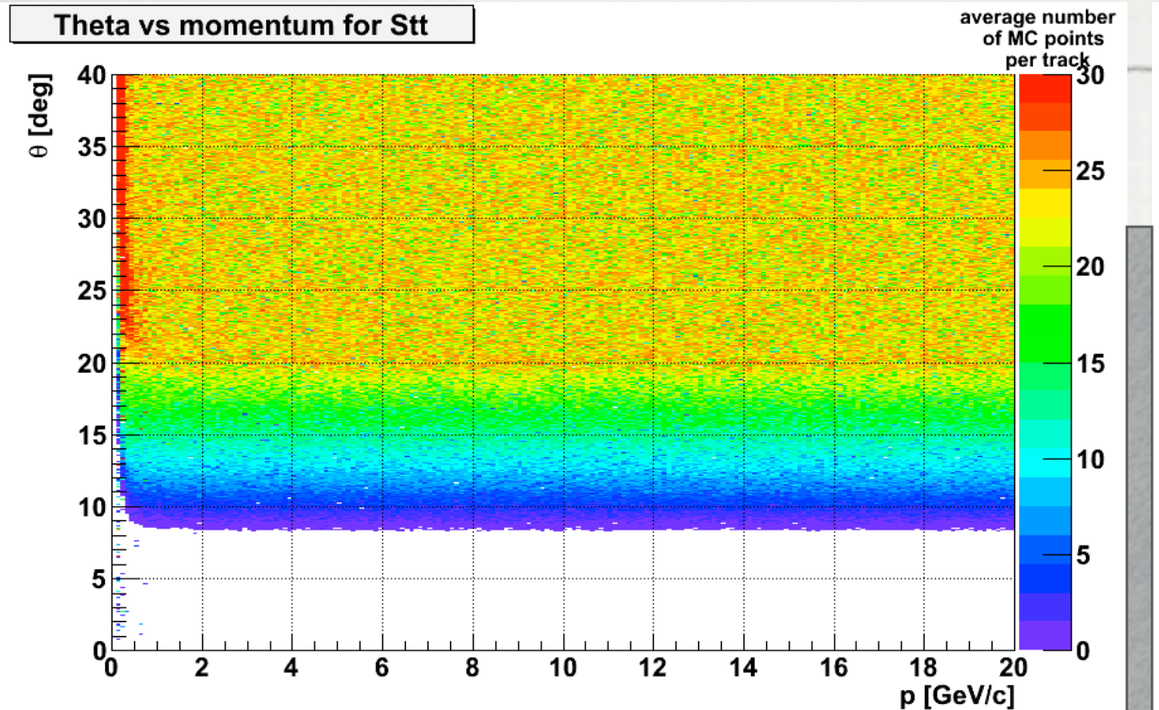
ACCEPTANCE MAP, SHORT, STT,

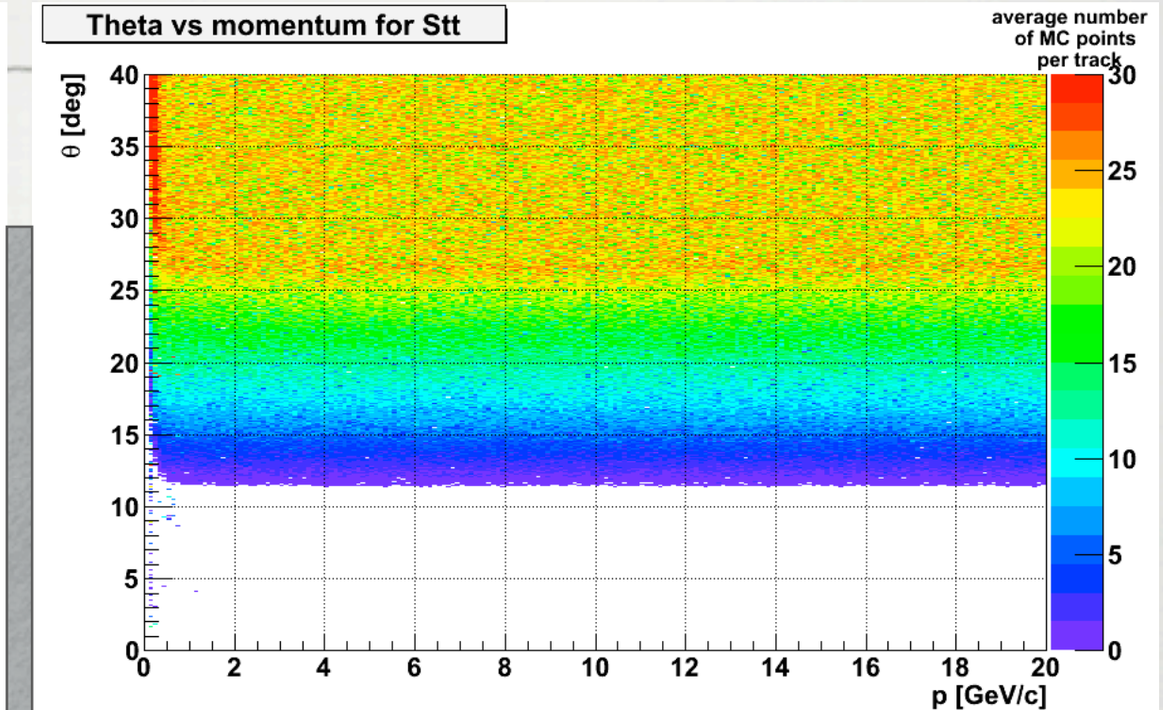
Θ VS MOMENTUM

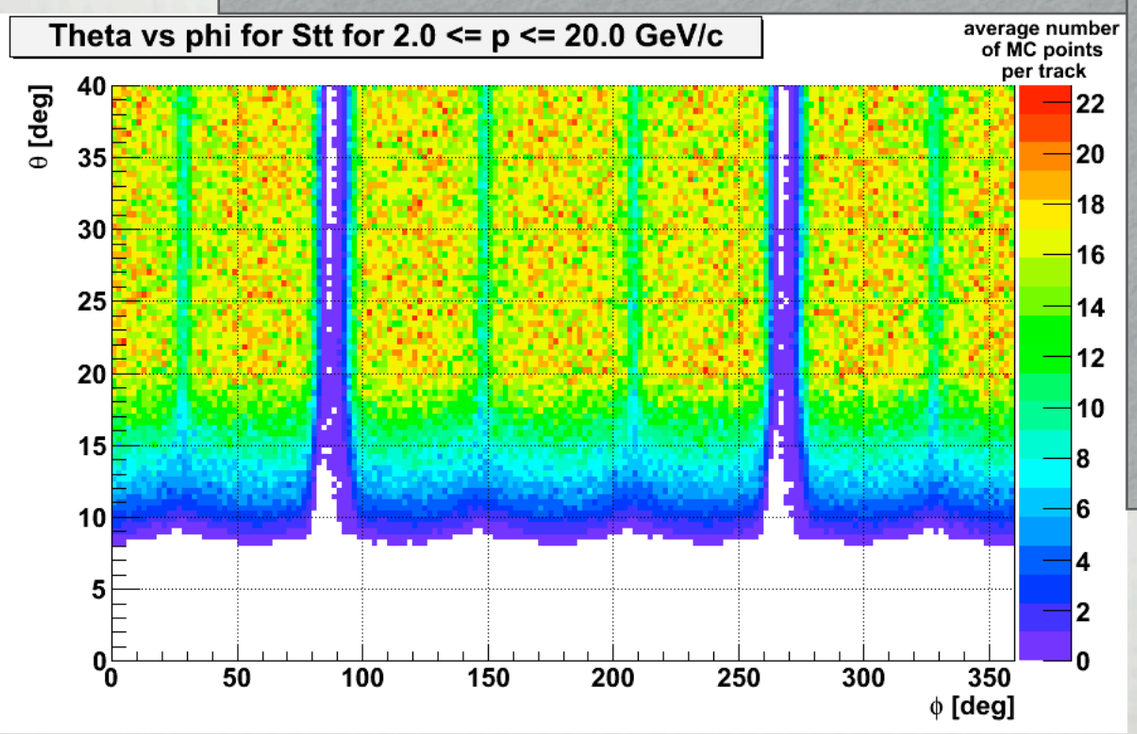


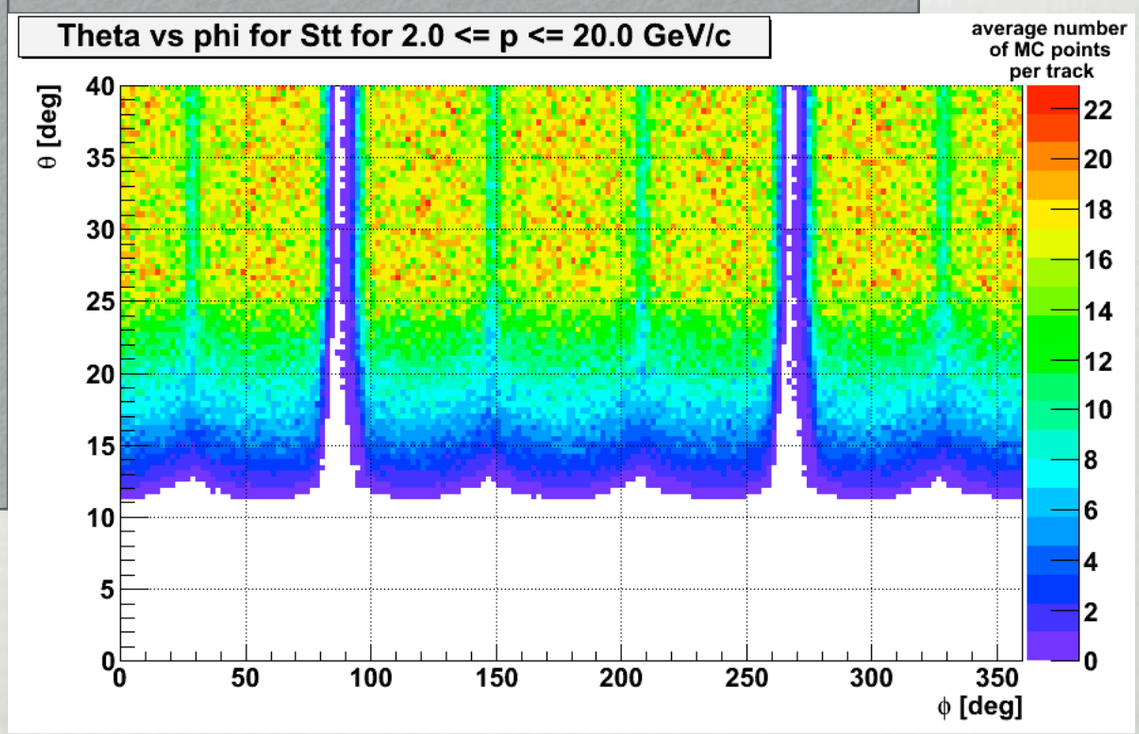
ACCEPTANCE MAP, STT,

LONG ALL SHORT



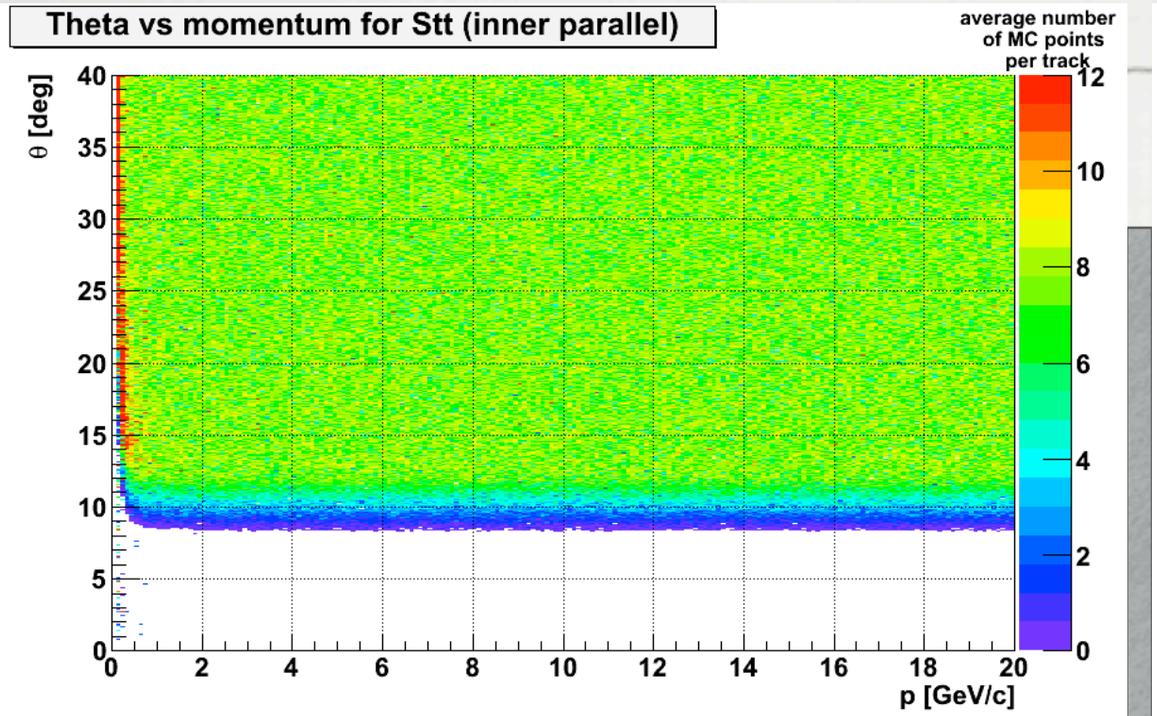


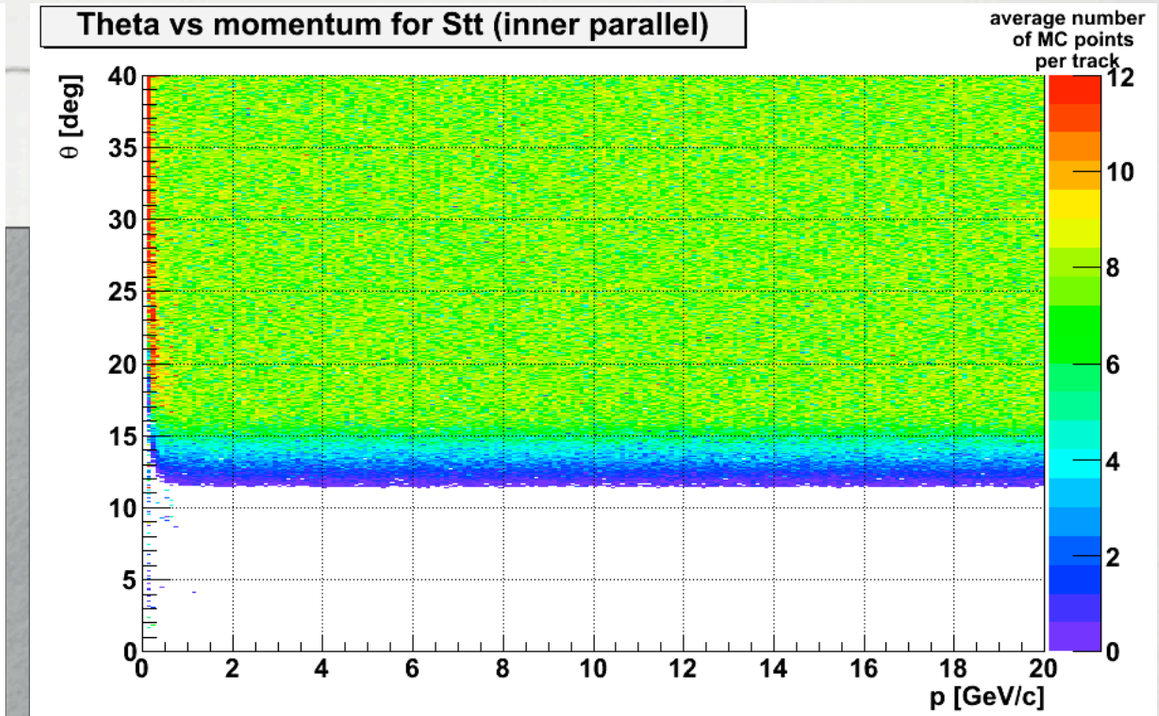


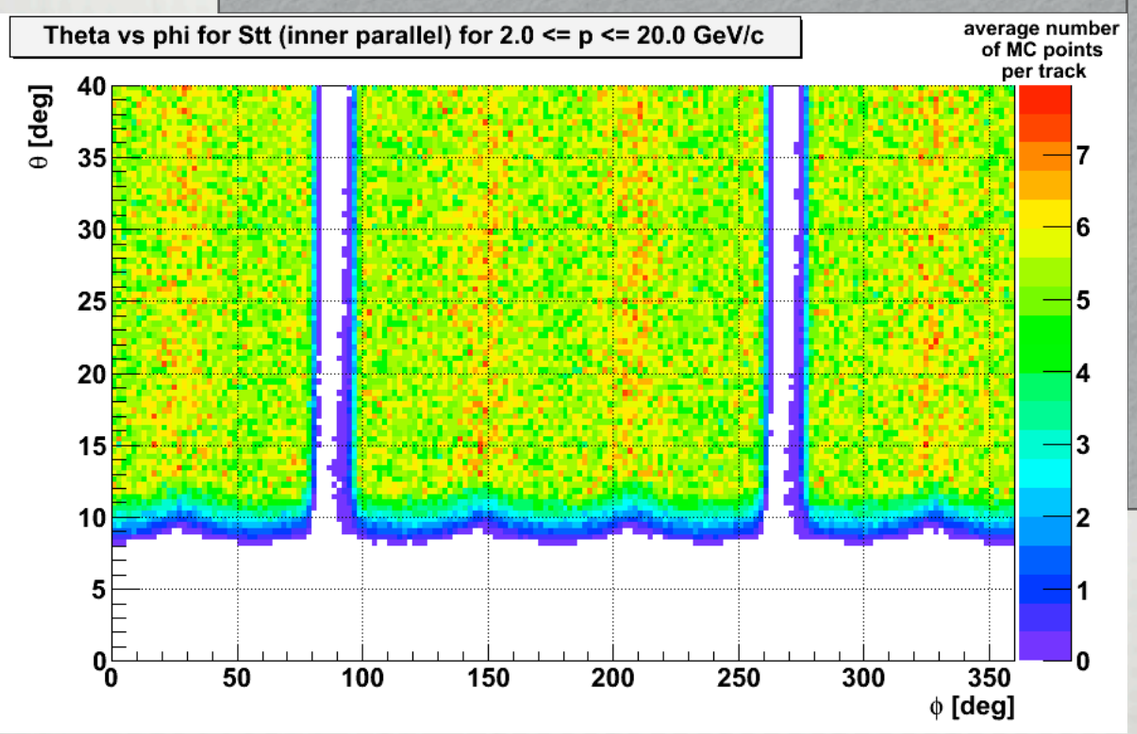


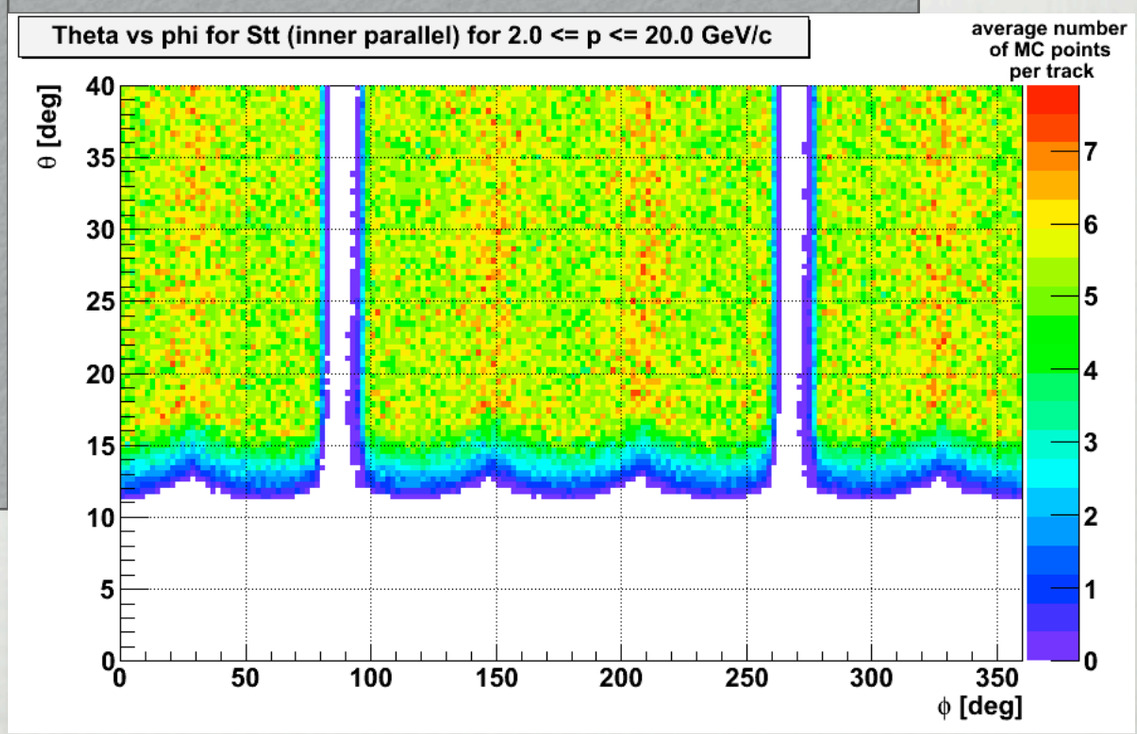
ACCEPTANCE MAP, STT,

LONG **INNER PARALLEL** SHORT



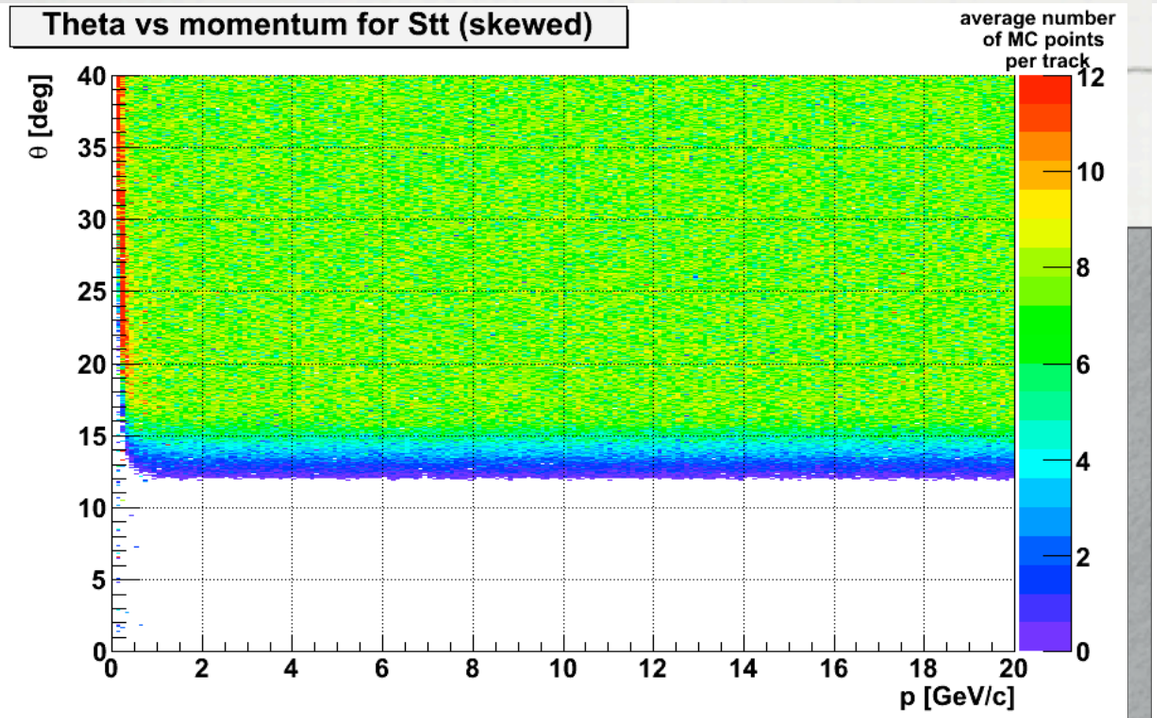


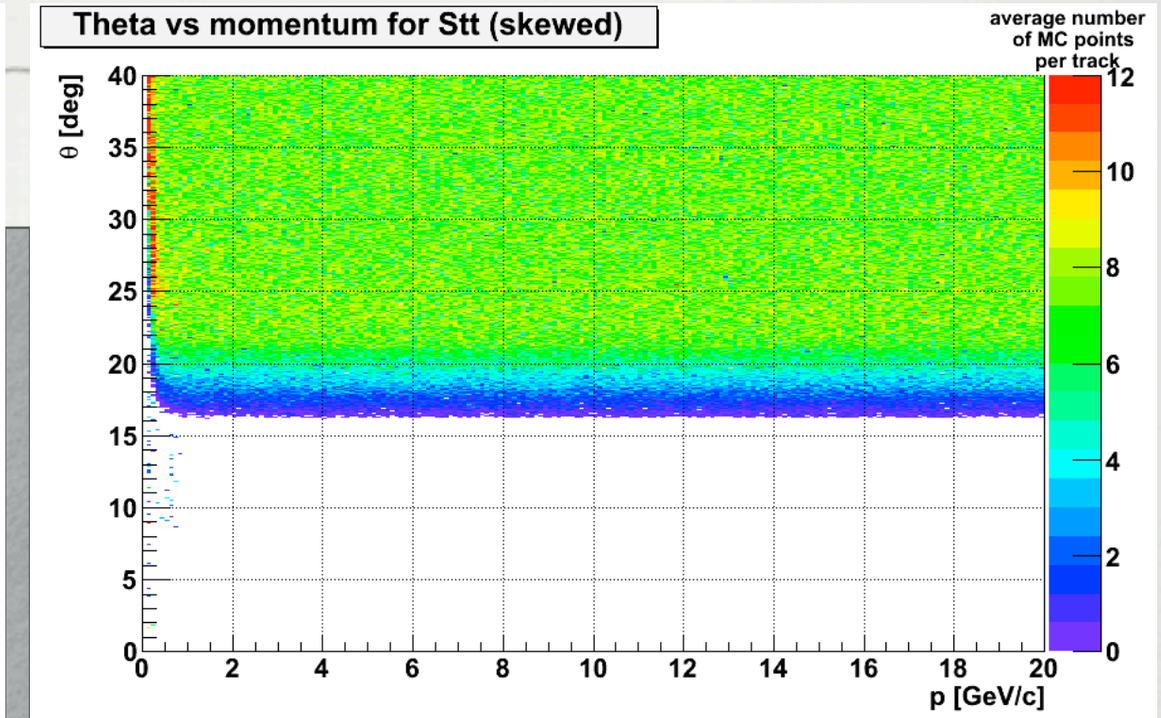


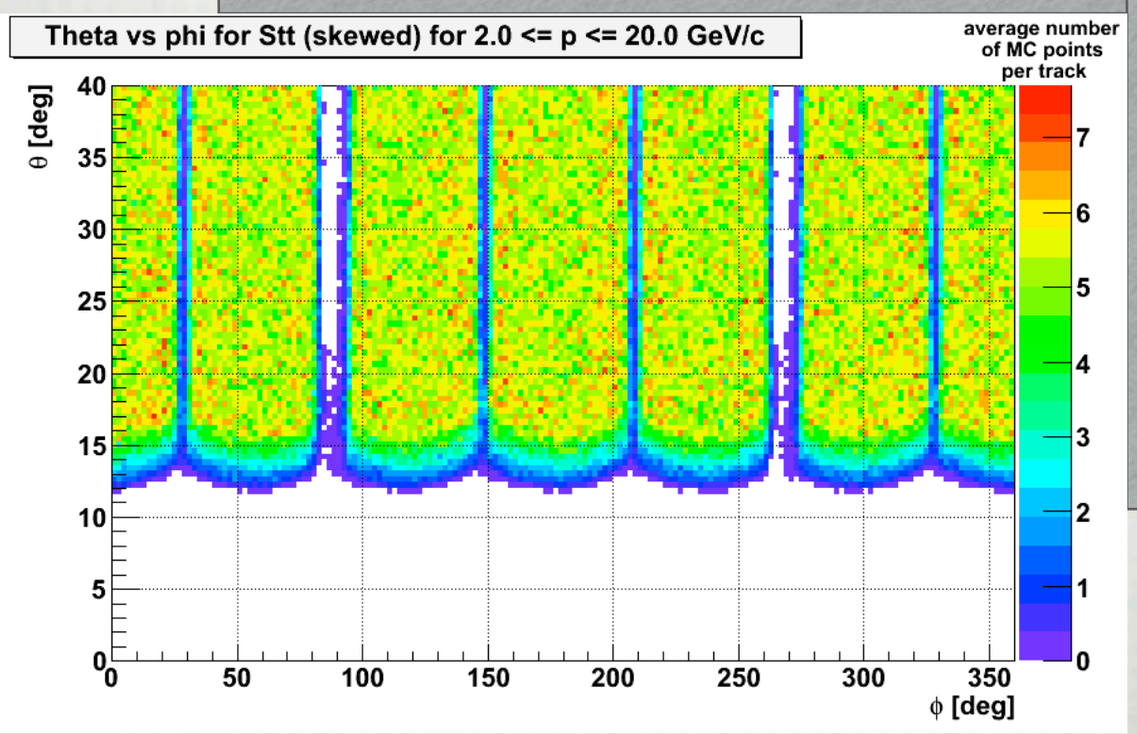


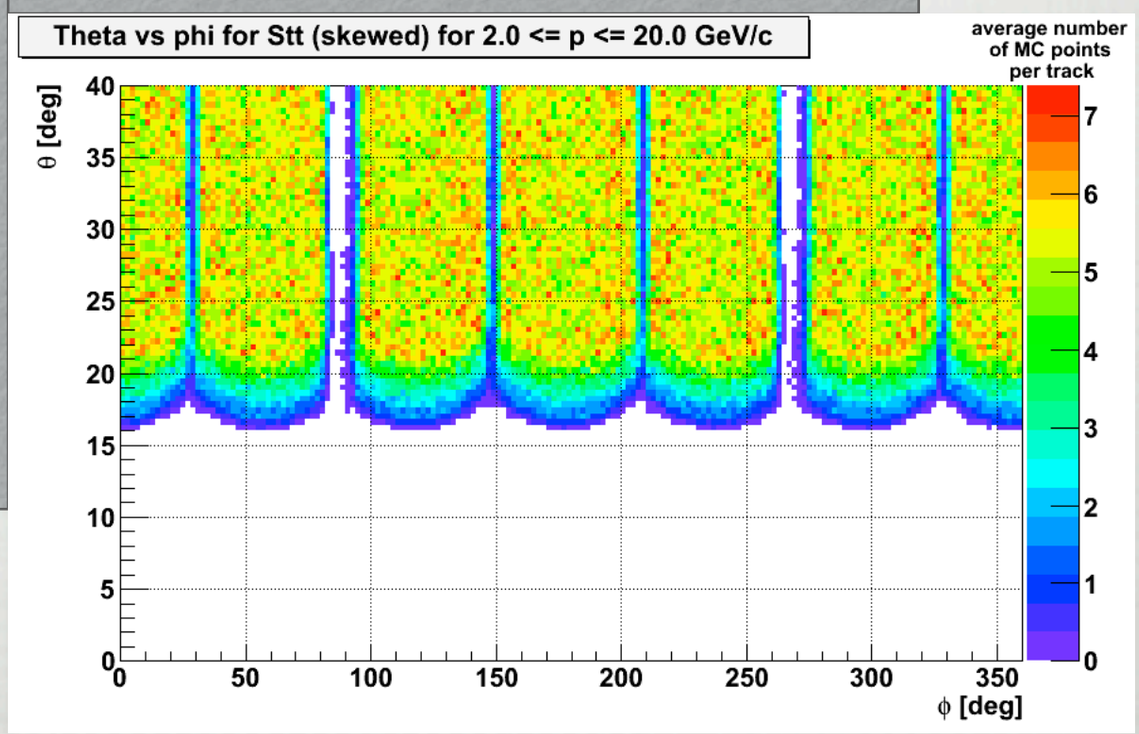
ACCEPTANCE MAP, STT,

LONG **SKEWED** SHORT



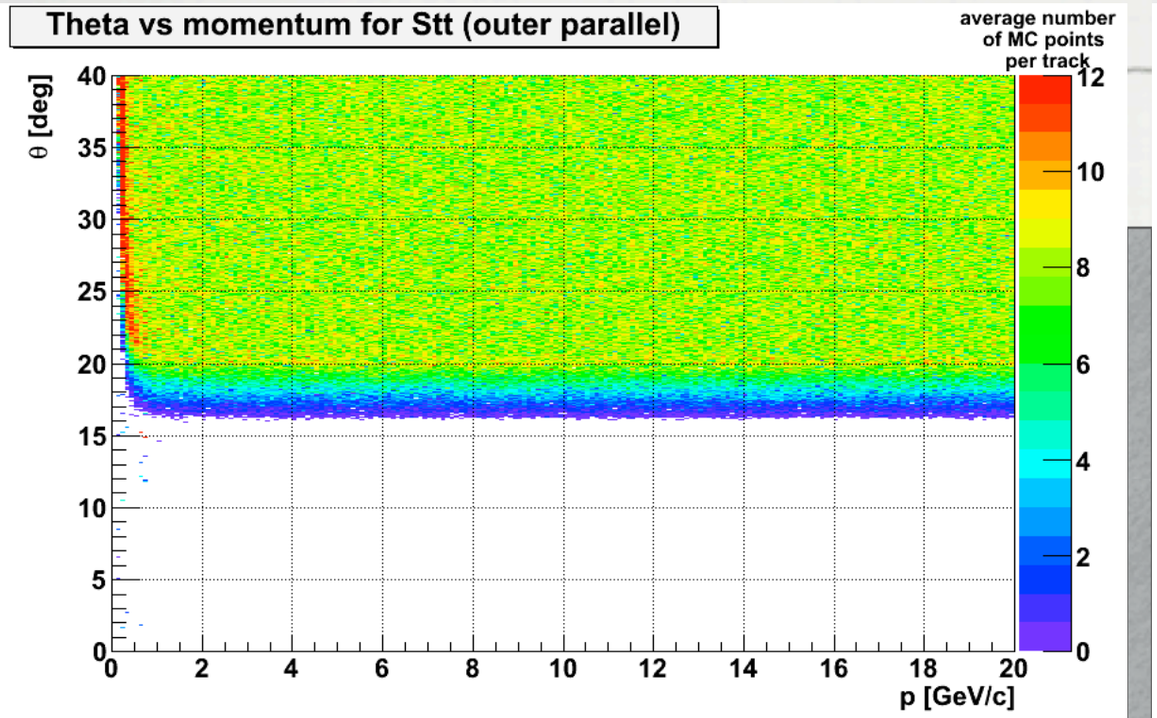


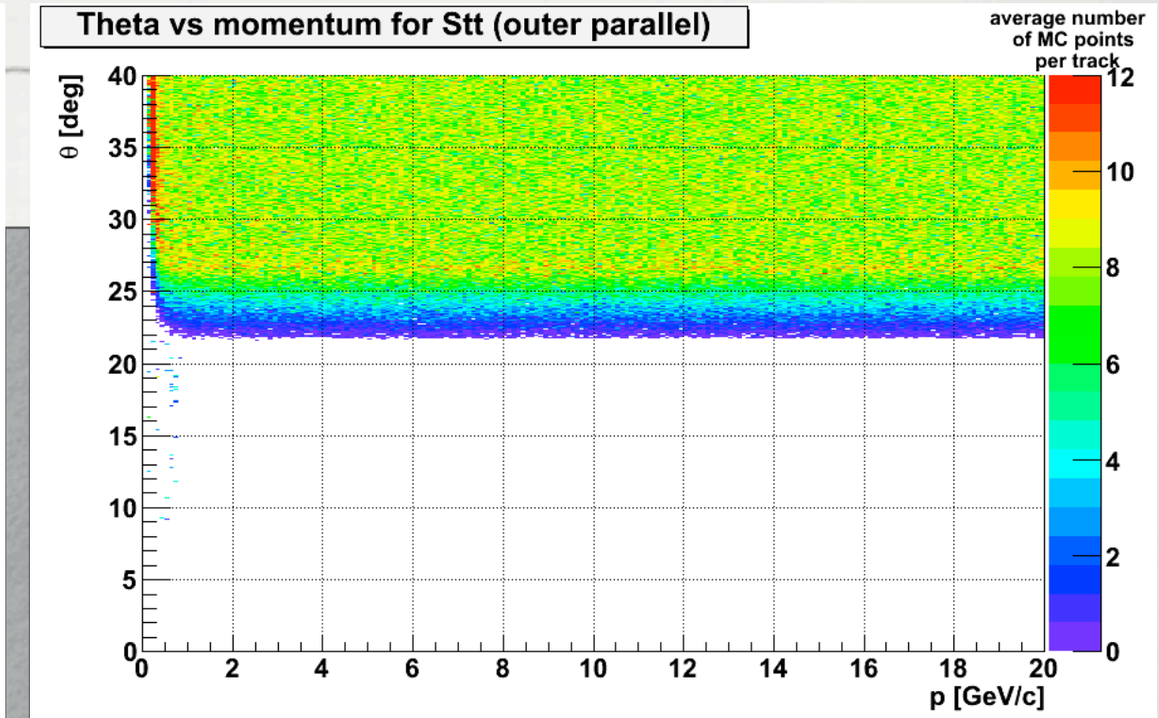


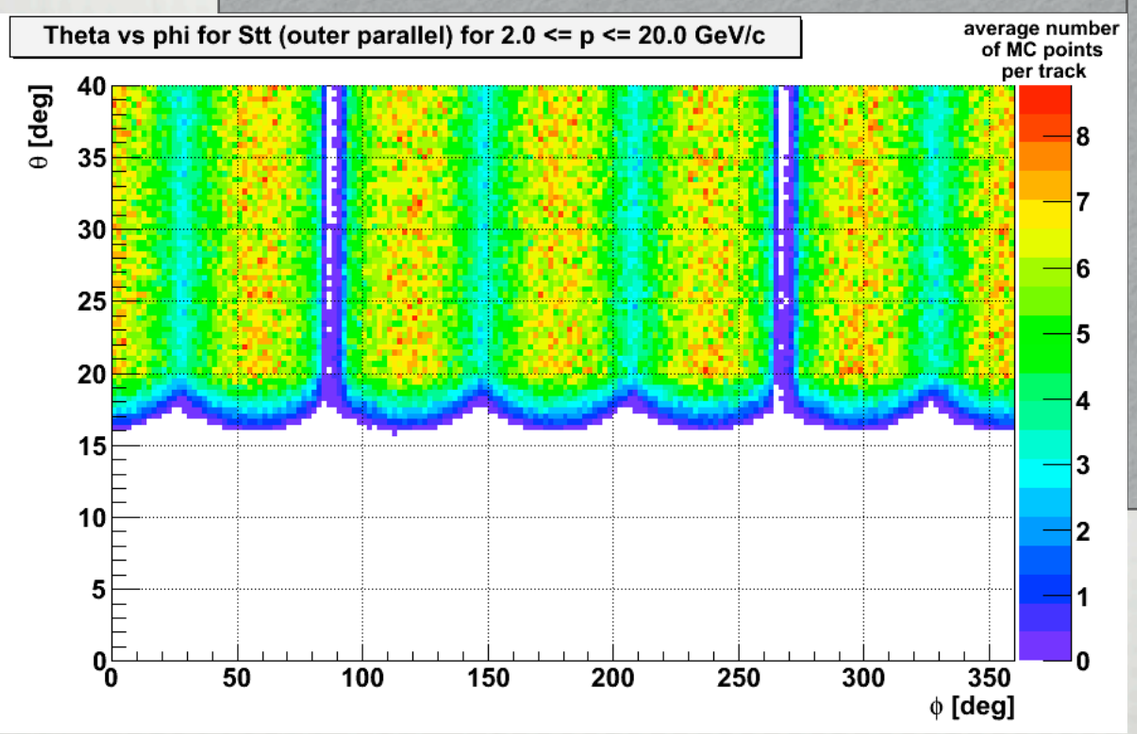


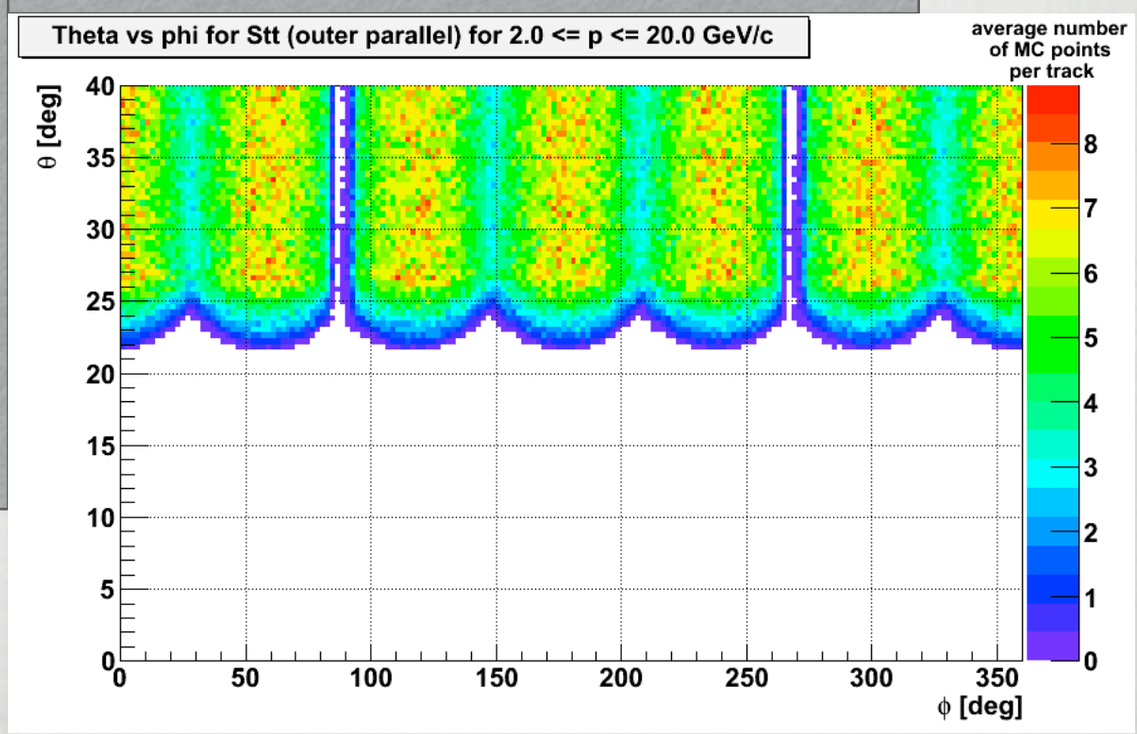
ACCEPTANCE MAP, STT,

LONG **OUTER PARALLEL** SHORT

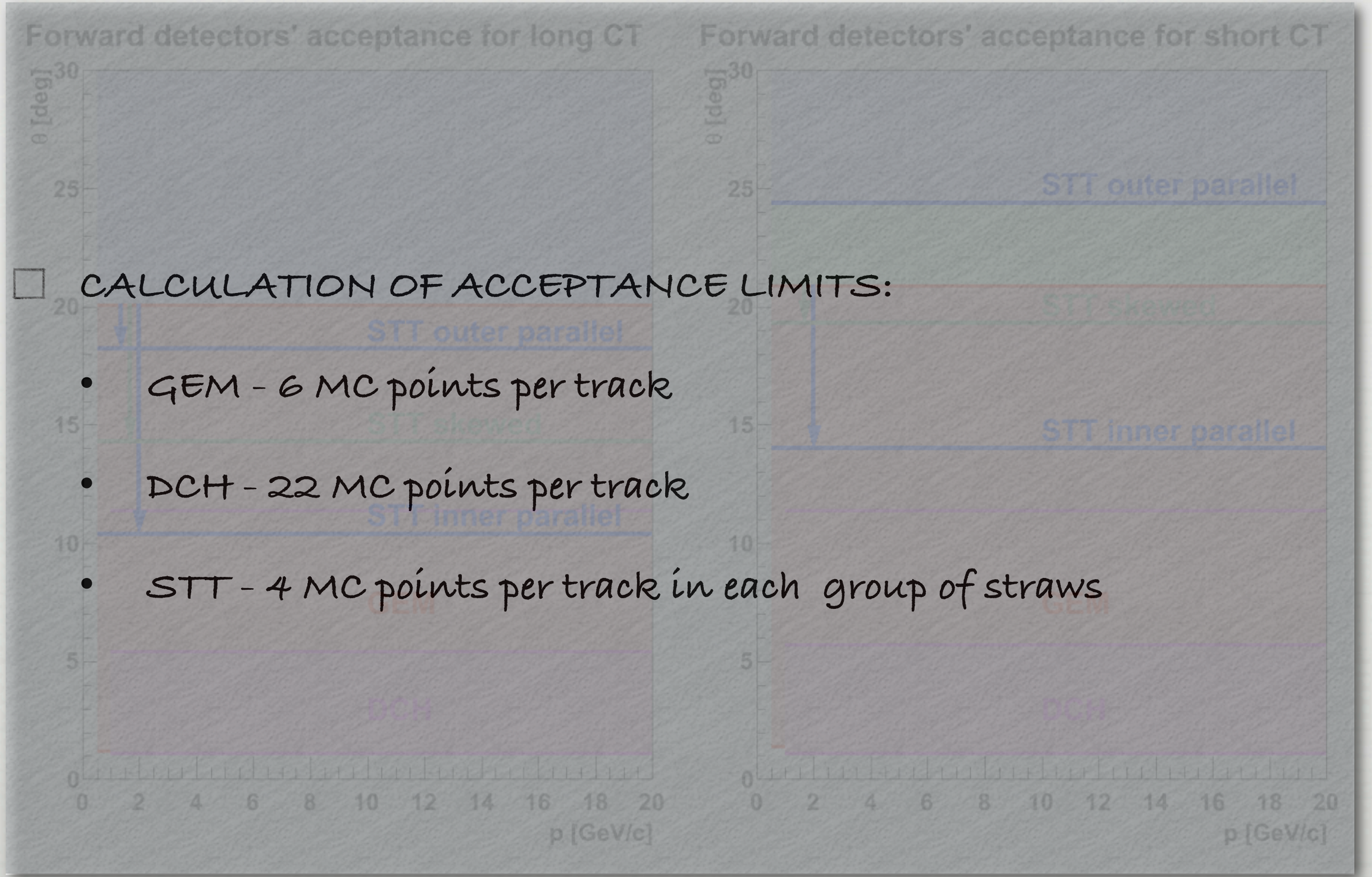






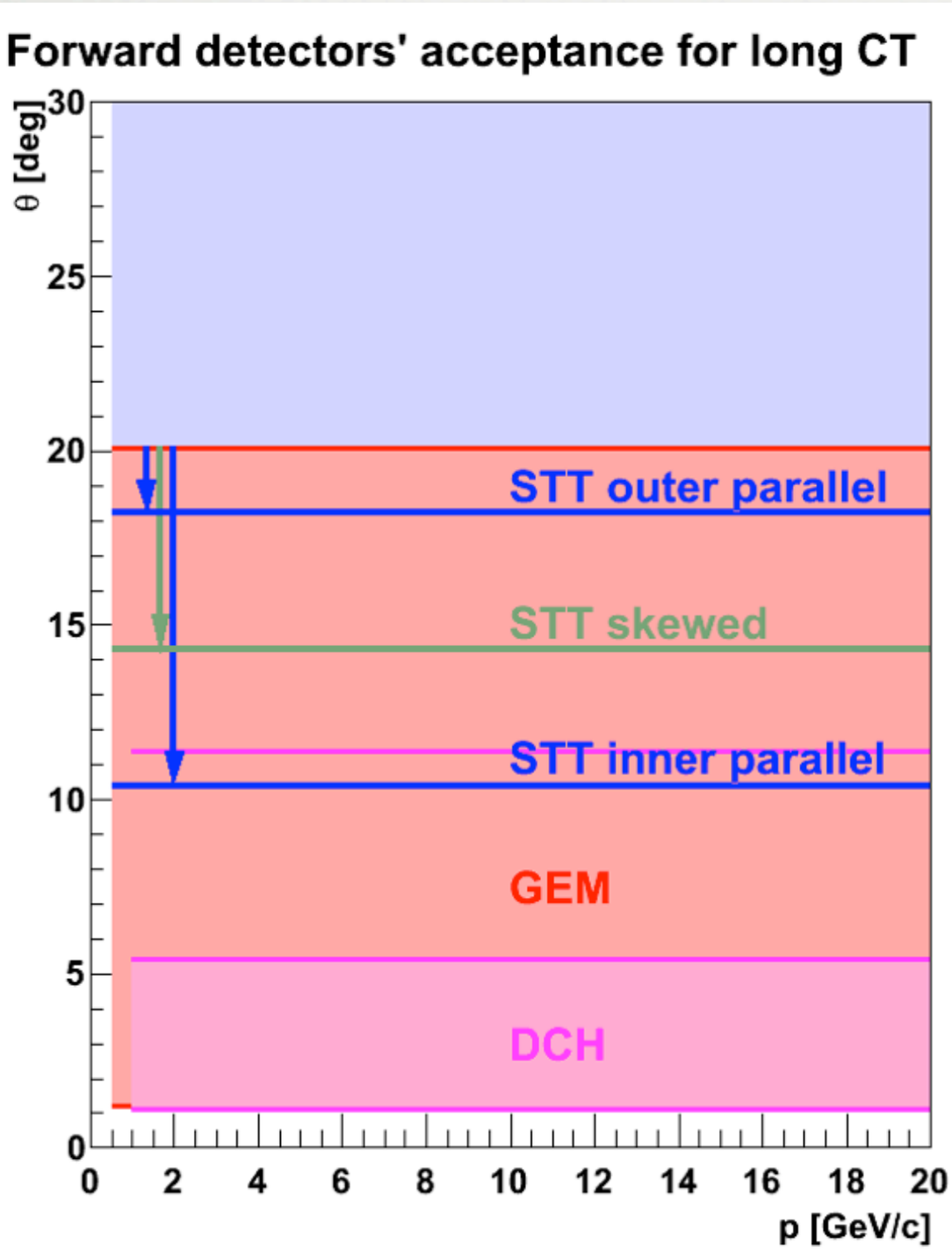


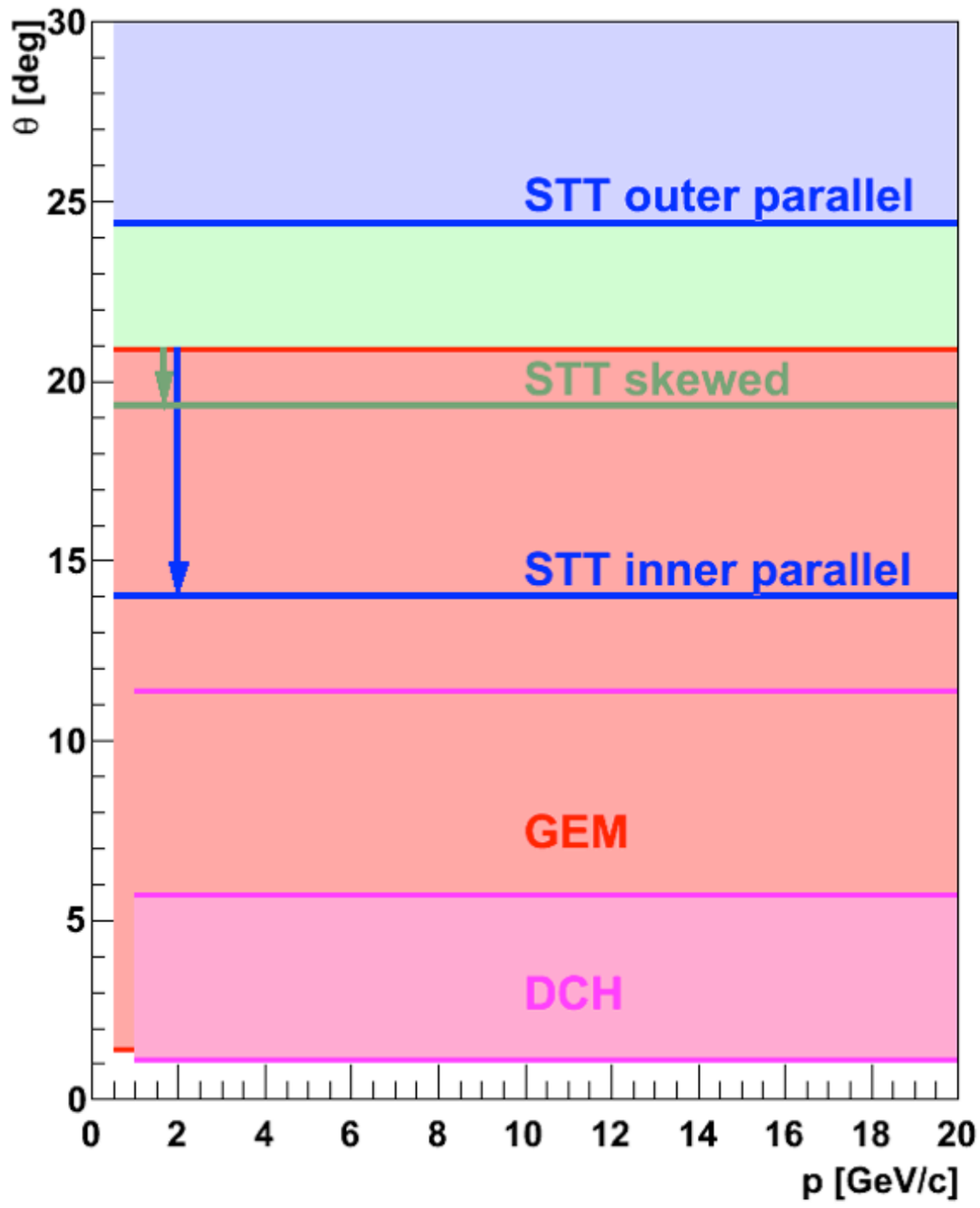
SCHEMATIC ACCEPTANCE MAPS



- CALCULATION OF ACCEPTANCE LIMITS:
  - GEM - 6 MC points per track
  - DCH - 22 MC points per track
  - STT - 4 MC points per track in each group of straws

SCHEMATIC ACCEPTANCE MAPS





SUMMARY

- SHORTENING STT BY ~30CM REDUCES STT Θ ACCEPTANCE BY ~5 DEGREES

- ADDING A GEM STATION AT ~83CM INCREASES GEM Θ ACCEPTANCE BY ~1DEGREE AND PROVIDES TWO ADDITIONAL HIGH-RESOLUTION MEASUREMENT POINTS FOR ANOTHER ~7 DEGREES IN Θ

- THE REGION OF OVERLAPPING STT & GEM ACCEPTANCE REDUCES FROM ~6 DEGREES TO ~2 DEGREES

CONCLUSIONS

- ACCEPTANCE MAPS FOR STT, GEM AND DCH HAS BEEN PRODUCED FOR TWO DIFFERENT DESIGNS OF CENTRAL TRACKER AND GEM-TRACKER
- IN CASE OF LONG CT AND 3 GEM STATIONS IT LOOKS THAT STANDALONE TRACK FINDERS IN STT AND GEM WILL BE SUFFICIENT TO COVER ACCEPTANCE WITHOUT GAPS IN Θ
- IN CASE OF SHORT CT AND 4 GEM STATIONS A TRACK FINDER USING HITS FROM DIFFERENT DETECTORS HAS TO BE USED
- SIMPLEST CHOICE IS THE USAGE OF THE LHE TRACK FINDER AND COMPARING TRACK EFFICIENCIES AND MOMENTUM RESOLUTION IN THE REGION OF $\Theta \in (20°,25°)$

GLOBAL IDEAL TRACK MERGER

- A GLOBAL IDEAL TRACK MERGER HAS BEEN DEVELOPED RECENTLY
- IT REQUIRED SEVERAL CHANGES TO THE EXISTING TRACK FINDERS
- ALL STANDALONE TRACK FINDERS SHOULD USE AND RETURN ARRAYS OF **PndTrack**s, USING **PndTrackCand** AND **PndTrackCandHit**
- GLOBAL PANDA ENUMERATION VARIABLES SHOULD BE USED CONSEQUENTLY AND CONSISTENTLY ALL THROUGHOUT THE **pandaroot** CODE

PndDetectorList.h

- ON FEBRUARY THE 2ND I HAVE STARTED A DISCUSSION ABOUT THE **fDetectorId** ENUM IN **PndDetectorList.h**. THE DECISION WAS TAKEN TO RENAME IT TO: **fDataId**, BUT I SEE NOW THE CHANGE WAS NOT ENOUGH, CAUSE IT IS MAINLY USED AS:

```
Int_t fDetectorId (class member) = fDataId;
```

- ON FEBRUARY THE 23RD CHRISTIAN REQUESTED HAVING **planeId** INFORMATION AVAILABLE FOR GENFIT. HE PROPOSED PUTTING IT IN **PndTrackCand** BUT I THINK HE REALLY MEANT THE **PndTrackCandHit**. ANYWAYS, NATURAL PLACE TO PUT IT IN IS THE **Int_t fDetectorId**

PndDetectorList.h

```
// -------------------------------------------------------------------------
// -----                  PndDetectorList.header file                  -----
// -----                Created 11/02/09  by M. Al-Turany              -----
// -------------------------------------------------------------------------


/** Unique identifier for all Panda detector systems **/

#ifndef PNDDETECTORLIST_H
#define PNDDETECTORLIST_H 1

enum DetectorId {
    kDCH,kDRC,kDSK,kEMC,kGEM,kLUMI,kMDT,kMVD,kRPC,kSTT,kTPC,kTOF,kHYPG,kHYP};

/** Unique identifier for all Panda Point and Hit types **/

enum fDetectorType {
   kUnknown, kMCTrack,
   kTpcPoint, kTpcCluster,
   kMVDPoint, kMVDDigiStrip, kMVDDigiPixel, kMVDClusterPixel, kMVDClusterStrip, kMVDHitsStrip, kMVDHitsPixel,
   kEmcCluster, kEmcBump,
   kSttPoint, kSttHit, kSttHelixHit,
   kGemPoint, kGemDigi, kGemHit,
   kDchPoint, kDchDigi, kDchHit,
   kTrackCand, kTrack};


enum SensorSide { kTOP, kBOTTOM };

#endif
```

DetectorId
up to 5 bits

fDetectorType (up to 5 bits)
- partially copies info from DetectorId
- wrong name (will be changed to fDataType)

fDetectorId class member

- Is a member of almost every data storage class:
  - fDetectorID, f.e. in FairMCPoint, FairHit
  - fDetectorId, f.e. PndGemDigi
  - fDetID, f.e. in PndMvdDigi
  - fDetId, f.e. in PndTrackCandHit
- What is the idea behind this variable? Should it ultimately become identifying part of any data: digi, hit or track?
- Can we use the whole 32bits of the value to store valuable information like detector, plane, sensor or even strip number?

Usage of fDetectorId

- GenFit RecoHitFactories. Each data structure should have different number attached to easily recognize, which implementation of RecoHitFactory to use.
  - "So to summarize, the detector ID in the trackCand would consist of a bit mask. Some part of this bit mask would be the kGemHit, kSttHit, kMvdPixelHit, ... which is used to produce hits in the RecoHitFactory. Could we do it a way that this kGemHit, kSttHit, kMvdPixelHit, ... info is in the least significant bits? That way it will be easiest for me. Regards, Christian"
- FairMultiLinkedData. Each data structure should have different number attached to easily distribute MC information over the digis/hits/tracks and so on; and to be able to automatically obtain information about the MC origin of any data structure...
  - "In my point of view all branches stored in a root file should have a unique ID to retrieve the data in an automatic way. This is mandatory to use the MC information propagation as I tried to explain in my presentation during the last EVO meeting. Cheers, Tobias"

fDetectorId class member

- My initial proposition:

4 BITS	5 BITS	5 BITS	18 BITS LEFT
like:	like:	like:	other information like:
MCTrack	MVD	stationID for GEM	fired digi number for MVD
MCPoint	STT	stationID for DCH	fired digi number for GEM
digi	TPC	layerID for STT	fired straw number for DCH
cluster	GEM		sextant number & straw number for STT
hit	DCH		
local track	EMC		
global track	DRC		

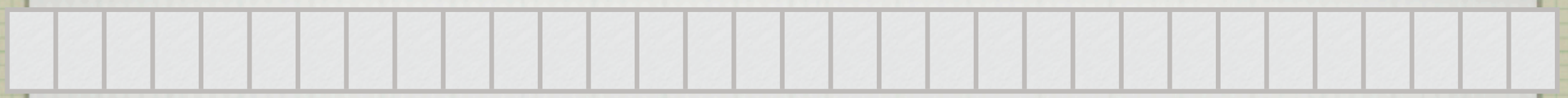
- Problems:
  - not all detectors fit in the scheme: "5 bit LayerID, and rest"
  - possible crazy combinations (dataType = bump && detectorID = MVD)
  - tracks do not have layer ID, global tracks do not have detectorID

final PndDetectorList.h
final fDetectorId class member

- Final meaning that we all agree on it

PndDetectorList.h

Int_t fDetectorId



- Thank you!

Backup slides

Backup slides

STT