Simulation of an Upgraded Tracker Layout

Harry Cheung (Fermilab)

Outline

Status of Upgrade Simulations

Issues and Priorities for Upgraded Tracker Layout

Summary of Mark Pesaresi's trigger doublet talk

Current Scope for Simulations

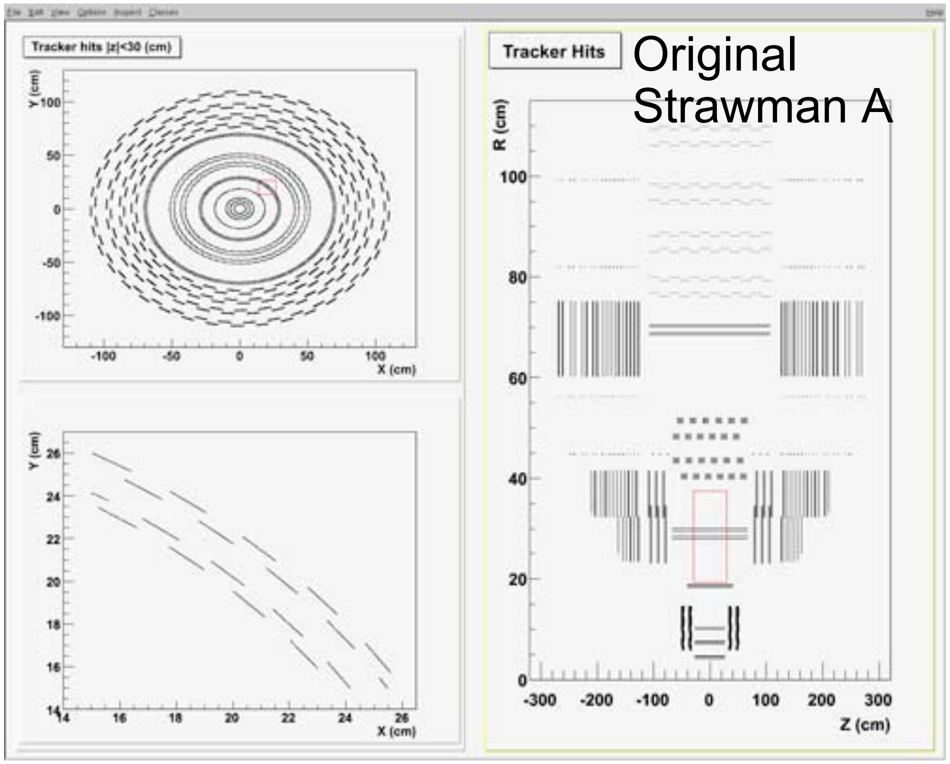
Summary

Status of upgrade simulation

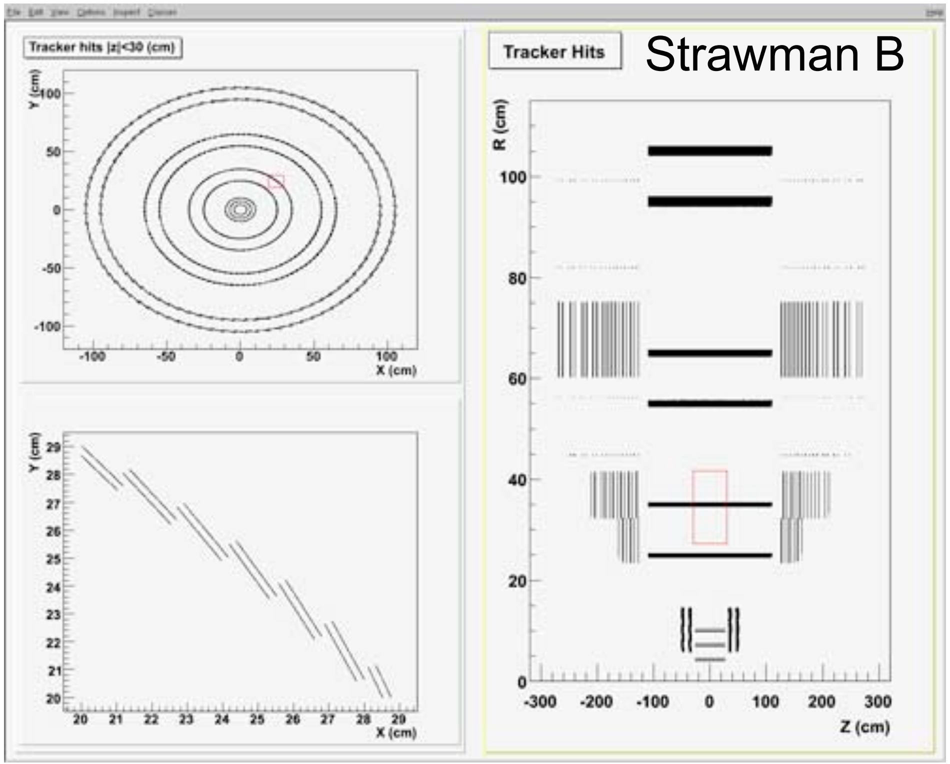
- The Tracker upgrade simulations working group has created upgrade simulation software
  - We have a modified version of the FastSimulation that can properly account for the tracking system granularity (runs faster than the Geant simulation)
  - We have two example strawman geometries set up that can be configured to study various geometry layouts (aimed at Phase 2)
    - Numbers and location in radius of layers
    - Addition of strixels (long pixels), mini-strips, and trigger doublet layers
    - Configurable pixel/strixel granularity in XML files
  - We have a very long barrel detector strawman
    - Not yet in CVS
  - We have a phase 1 geometry with 4 barrel pixel layers
    - A Phase 1 geometry using to Roland's proposal/options 1-5
    - Not yet worked out a forward pixel phase 1 geometry
  - We are using the standard tracking performance validation packages
    - Work to do in simplifying the performance packages for our studies
    - Work to do in enabling fast running at the highest pileup, and more realistic pileup for the FastSimulation

Geometry Layouts

- We have a number of strawman layouts
  - Original and more realistic Strawman A
  - Strawman B with superlayers of doublets
  - Long barrel strawman
  - Phase 1 strawman (pixel detector)



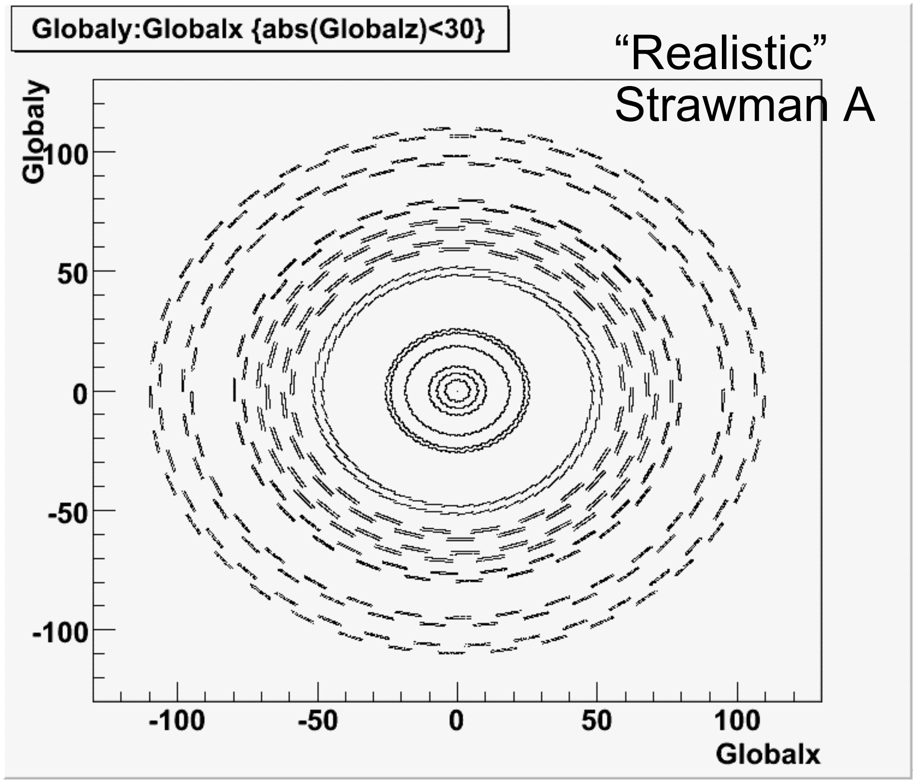
Original Strawman A



Strawman B

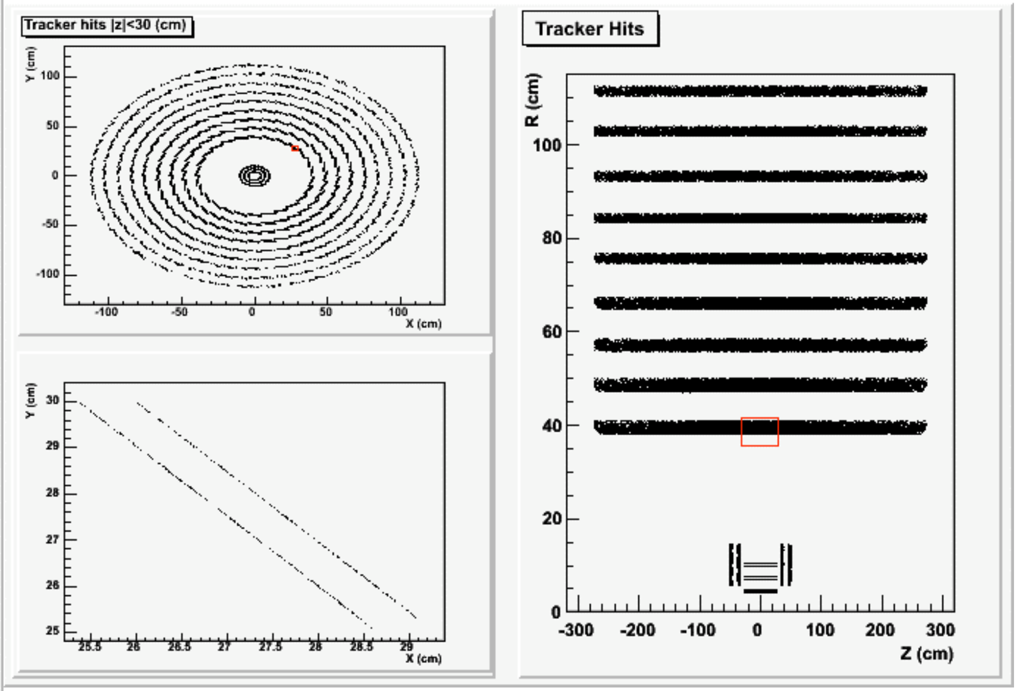
Geometry Layouts

- Strawman Geometry was supposed to limit the phase space
  - Already many geometry layout variations to simulate and study
  - Layout will be much easier once we know what track triggering method we need and what the “trigger layer(s)” look like (Doublet? Cluster shape?)
- Worse for forward region, no track trigger idea yet?



“Realistic” Strawman A

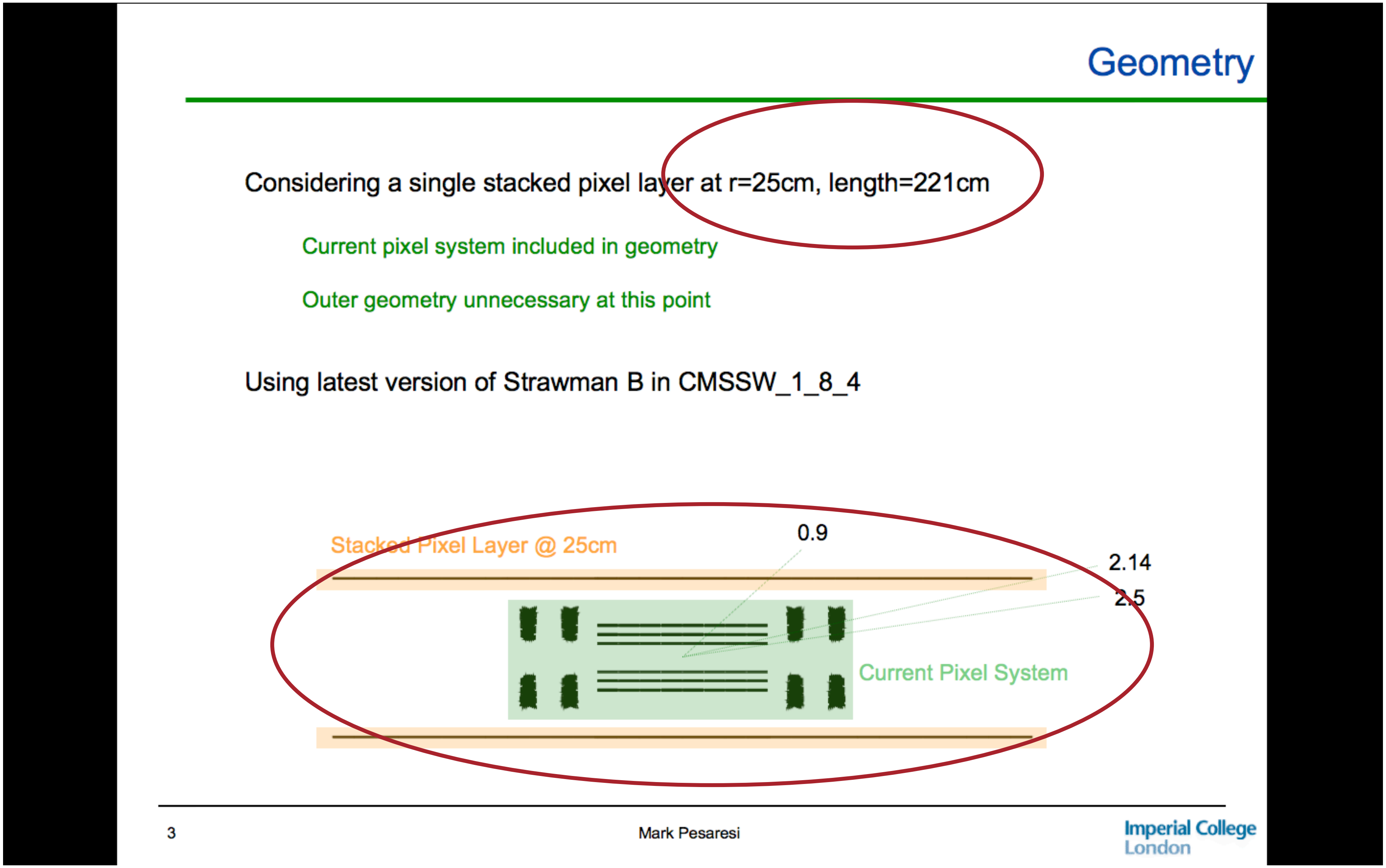
Long Barrel Strawman



Track Trigger Layers

- Top priority to see whether a (buildable) trigger doublet would work, how many are needed and what their parameters should be
  - This can be studied in any of the strawman geometries
    - Want to study both a single doublet and a “stack” of 2 doublets
  - Mark Pesaresi is studying trigger doublet performance in Strawman B
    - Studying p_T thresholds for both a single doublet and pair of doublets
    - See Mark’s talk from yesterday’s Tracker session (layout and simulation)

Mark Pesaresi's Doublet Study



Geometry

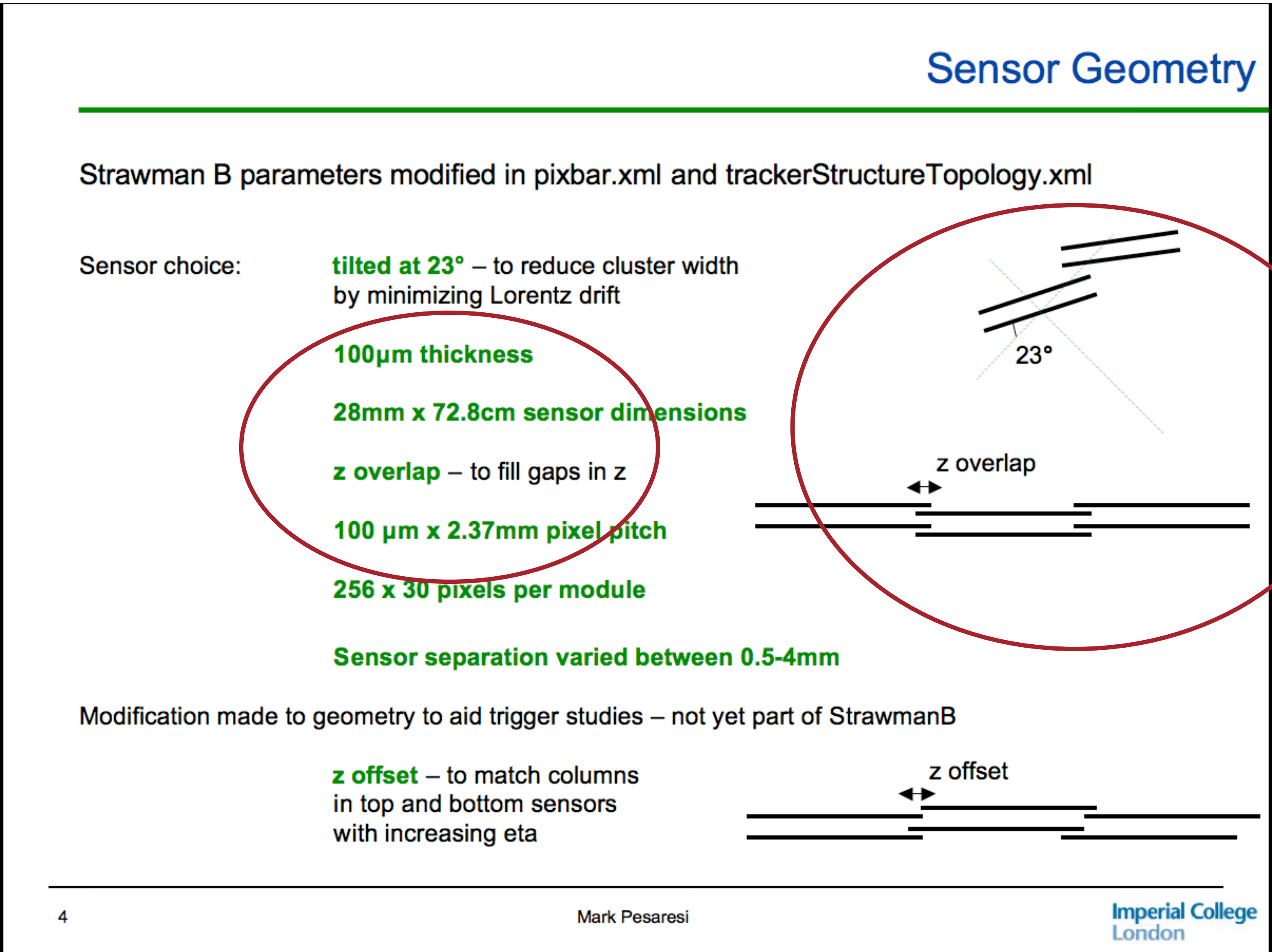
Considering a single stacked pixel layer at r=25cm, length=221cm

- Current pixel system included in geometry
- Outer geometry unnecessary at this point

Using latest version of Strawman B in CMSSW_1_8_4

3 Mark Pesaresi Imperial College London

Mark Pesaresi's Doublet Study



Sensor Geometry

Strawman B parameters modified in pixbar.xml and trackerStructureTopology.xml

Sensor choice: **tilted at 23°** – to reduce cluster width by minimizing Lorentz drift

100μm thickness

28mm x 72.8cm sensor dimensions

z overlap – to fill gaps in z

100 μm x 2.37mm pixel pitch

256 x 30 pixels per module

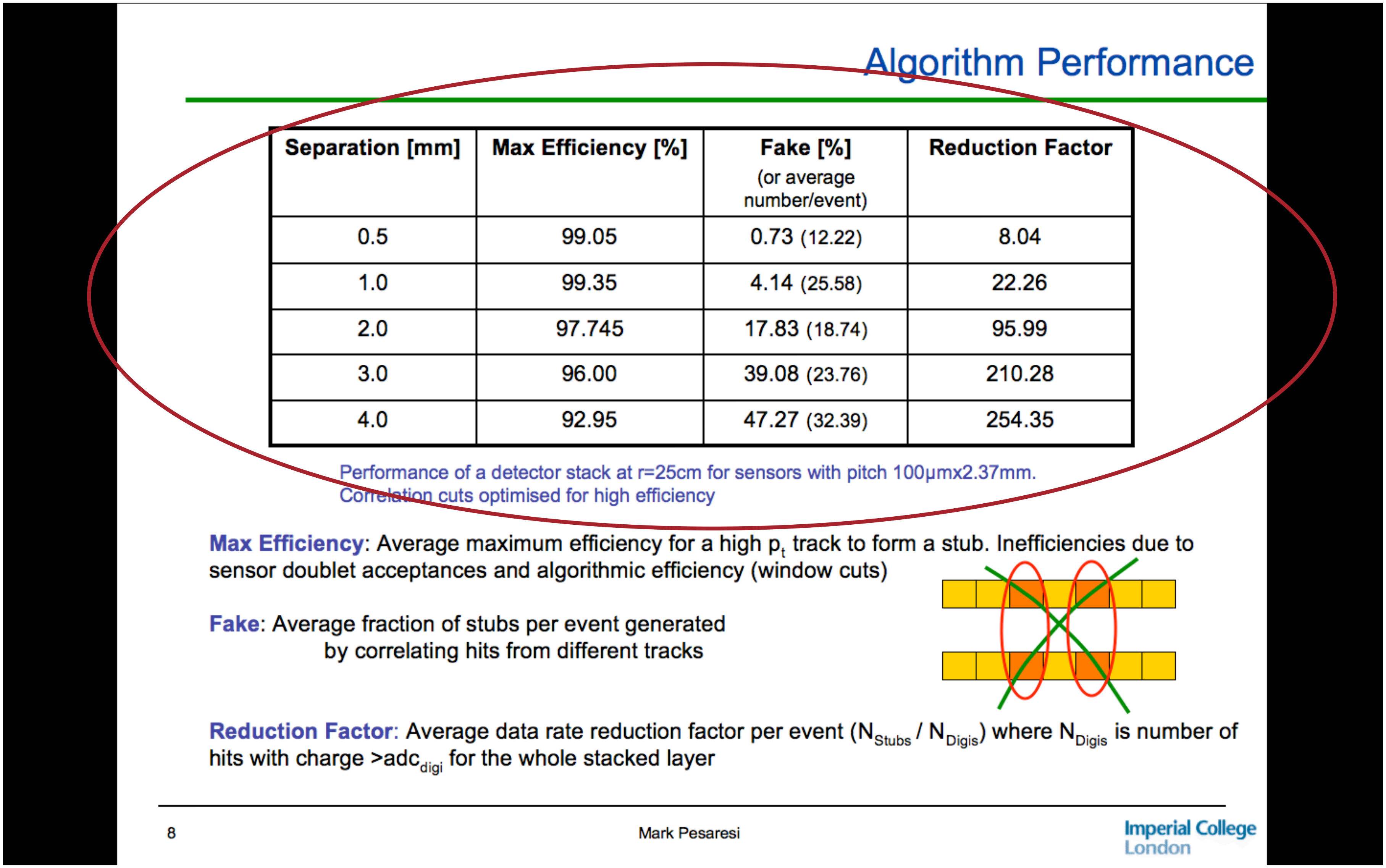
Sensor separation varied between 0.5-4mm

Modification made to geometry to aid trigger studies – not yet part of StrawmanB

z offset – to match columns in top and bottom sensors with increasing eta

4 Mark Pesaresi Imperial College London

Mark Pesaresi's Doublet Study



Algorithm Performance

Separation [mm]	Max Efficiency [%]	Fake [%] (or average number/event)	Reduction Factor
0.5	99.05	0.73 (12.22)	8.04
1.0	99.35	4.14 (25.58)	22.26
2.0	97.745	17.83 (18.74)	95.99
3.0	96.00	39.08 (23.76)	210.28
4.0	92.95	47.27 (32.39)	254.35

Performance of a detector stack at r=25cm for sensors with pitch 100μmx2.37mm. Correlation cuts optimised for high efficiency

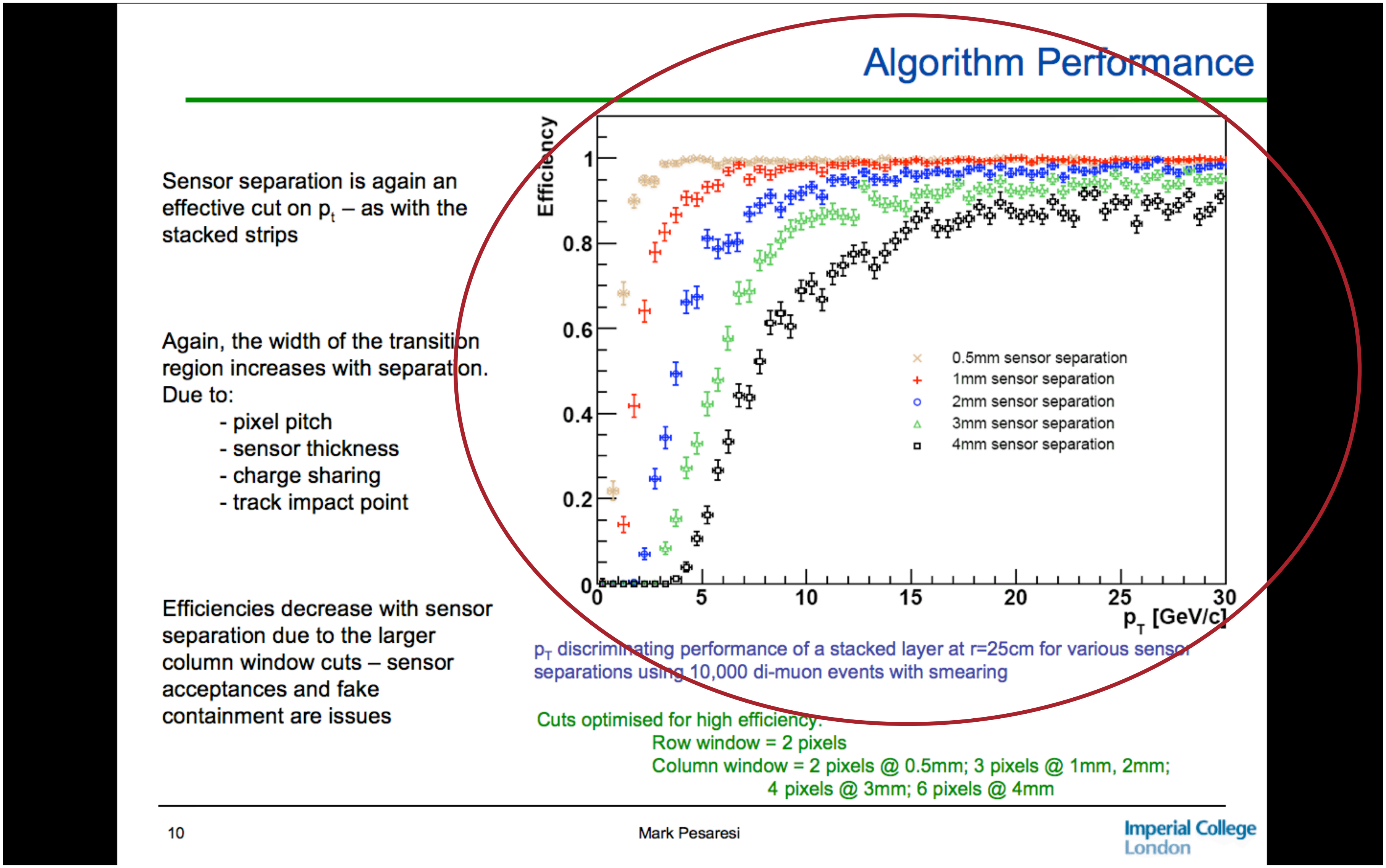
Max Efficiency: Average maximum efficiency for a high p_t track to form a stub. Inefficiencies due to sensor doublet acceptances and algorithmic efficiency (window cuts)

Fake: Average fraction of stubs per event generated by correlating hits from different tracks

Reduction Factor: Average data rate reduction factor per event (N_{Stubs} / N_{Digis}) where N_{Digis} is number of hits with charge $>adc_{digi}$ for the whole stacked layer

8 Mark Pesaresi Imperial College London

Mark Pesaresi’s Doublet Study



Algorithm Performance

Sensor separation is again an effective cut on p_t – as with the stacked strips

Again, the width of the transition region increases with separation. Due to:

- pixel pitch
- sensor thickness
- charge sharing
- track impact point

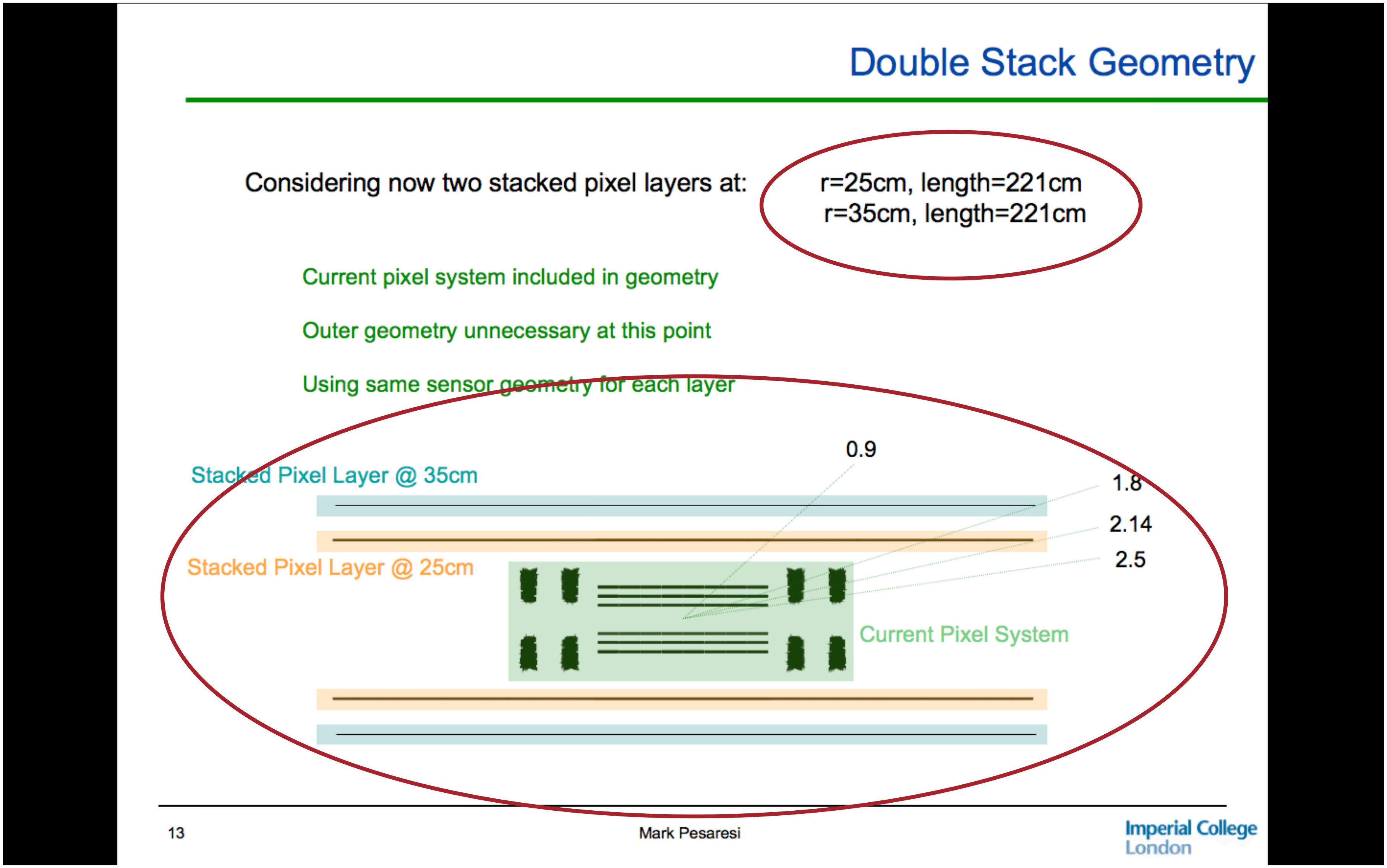
Efficiencies decrease with sensor separation due to the larger column window cuts – sensor acceptances and fake containment are issues

p_T discriminating performance of a stacked layer at r=25cm for various sensor separations using 10,000 di-muon events with smearing

Cuts optimised for high efficiency:
Row window = 2 pixels
Column window = 2 pixels @ 0.5mm; 3 pixels @ 1mm, 2mm;
4 pixels @ 3mm; 6 pixels @ 4mm

10 Mark Pesaresi Imperial College London

Mark Pesaresi’s Doublet Study



Double Stack Geometry

Considering now two stacked pixel layers at:

r=25cm, length=221cm
r=35cm, length=221cm

Current pixel system included in geometry

Outer geometry unnecessary at this point

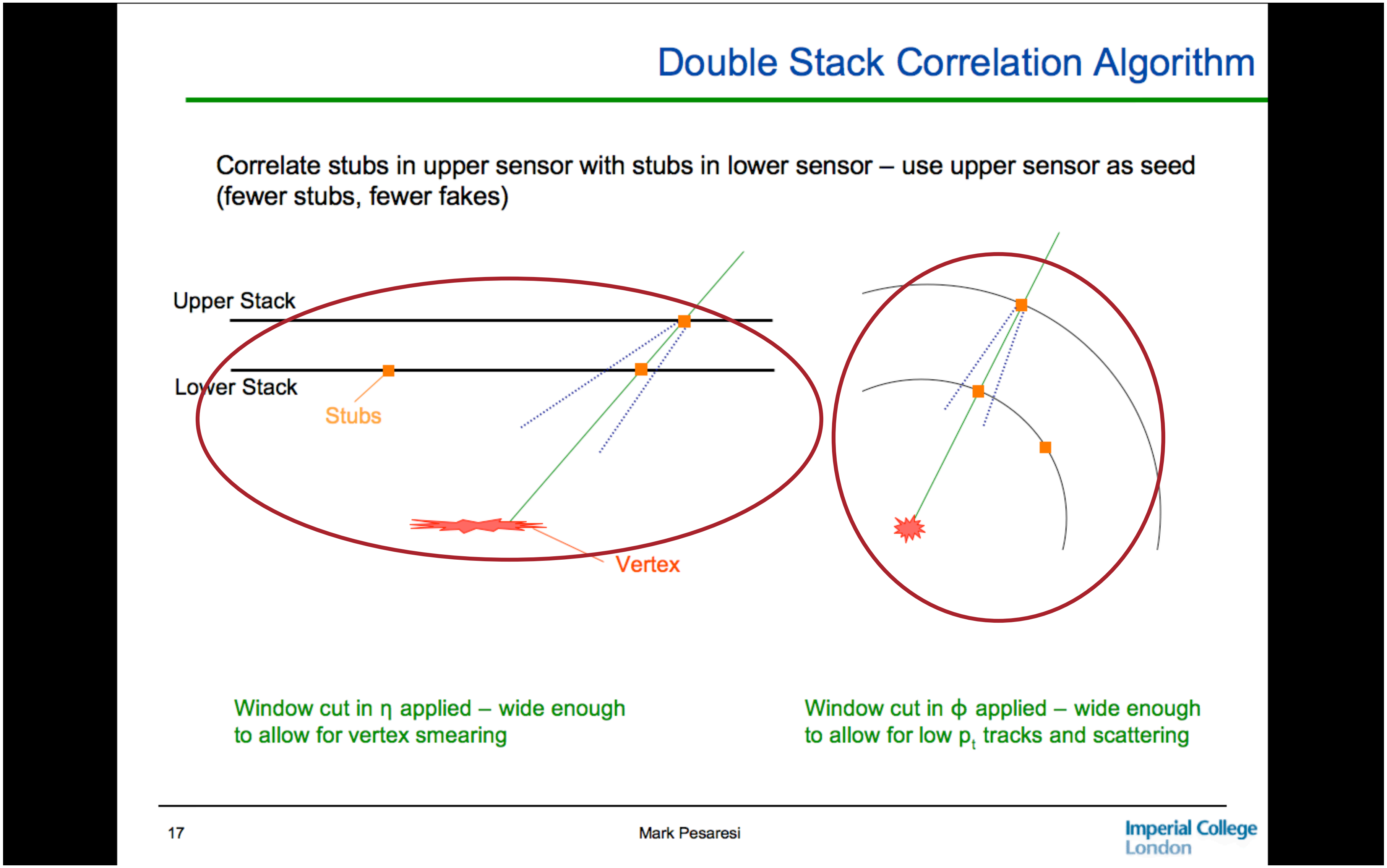
Using same sensor geometry for each layer

13

Mark Pesaresi

Imperial College
London

Mark Pesaresi's Doublet Study



Double Stack Correlation Algorithm

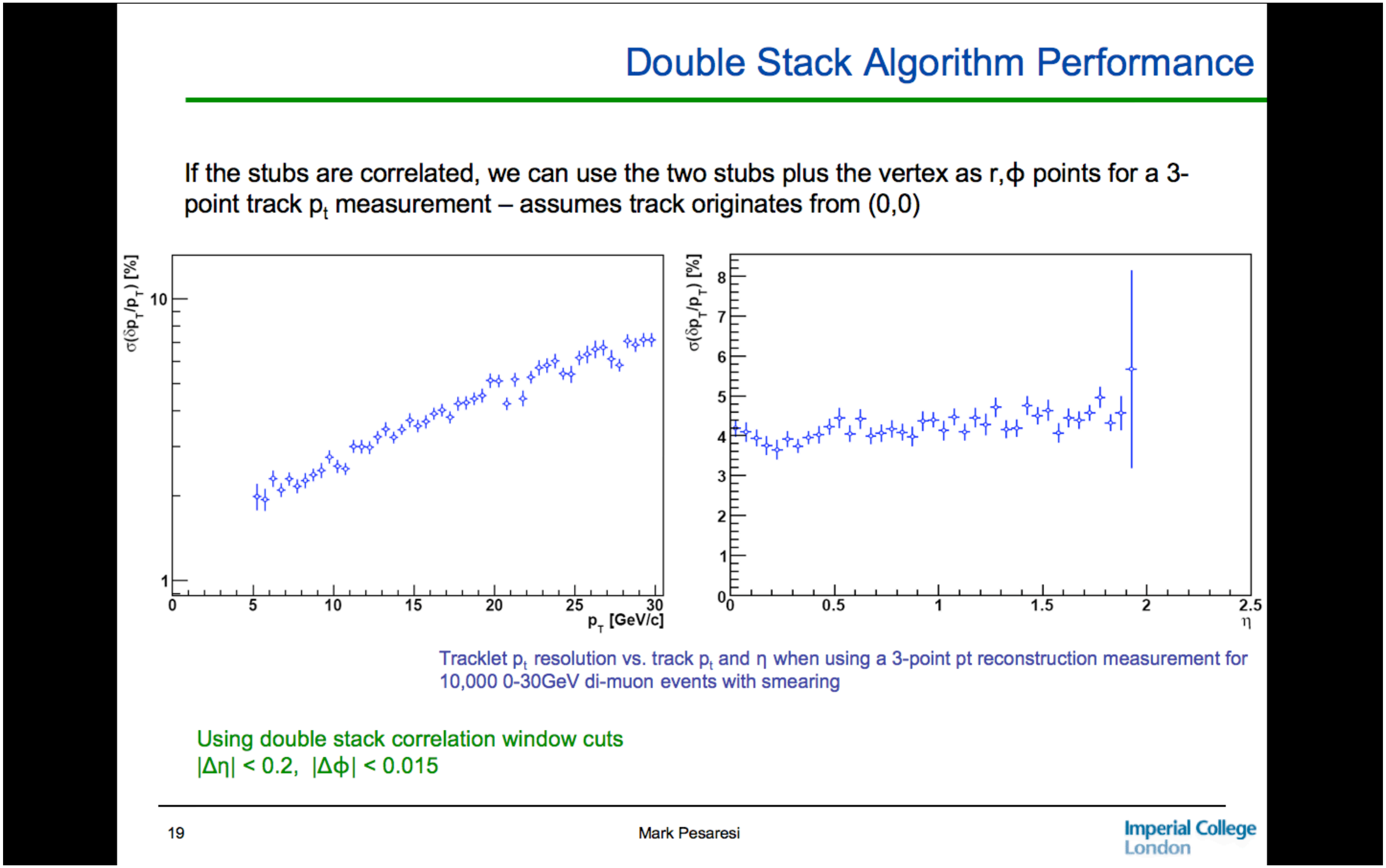
Correlate stubs in upper sensor with stubs in lower sensor – use upper sensor as seed (fewer stubs, fewer fakes)

Window cut in η applied – wide enough to allow for vertex smearing

Window cut in ϕ applied – wide enough to allow for low p_t tracks and scattering

17 Mark Pesaresi Imperial College London

Mark Pesaresi’s Doublet Study



Double Stack Algorithm Performance

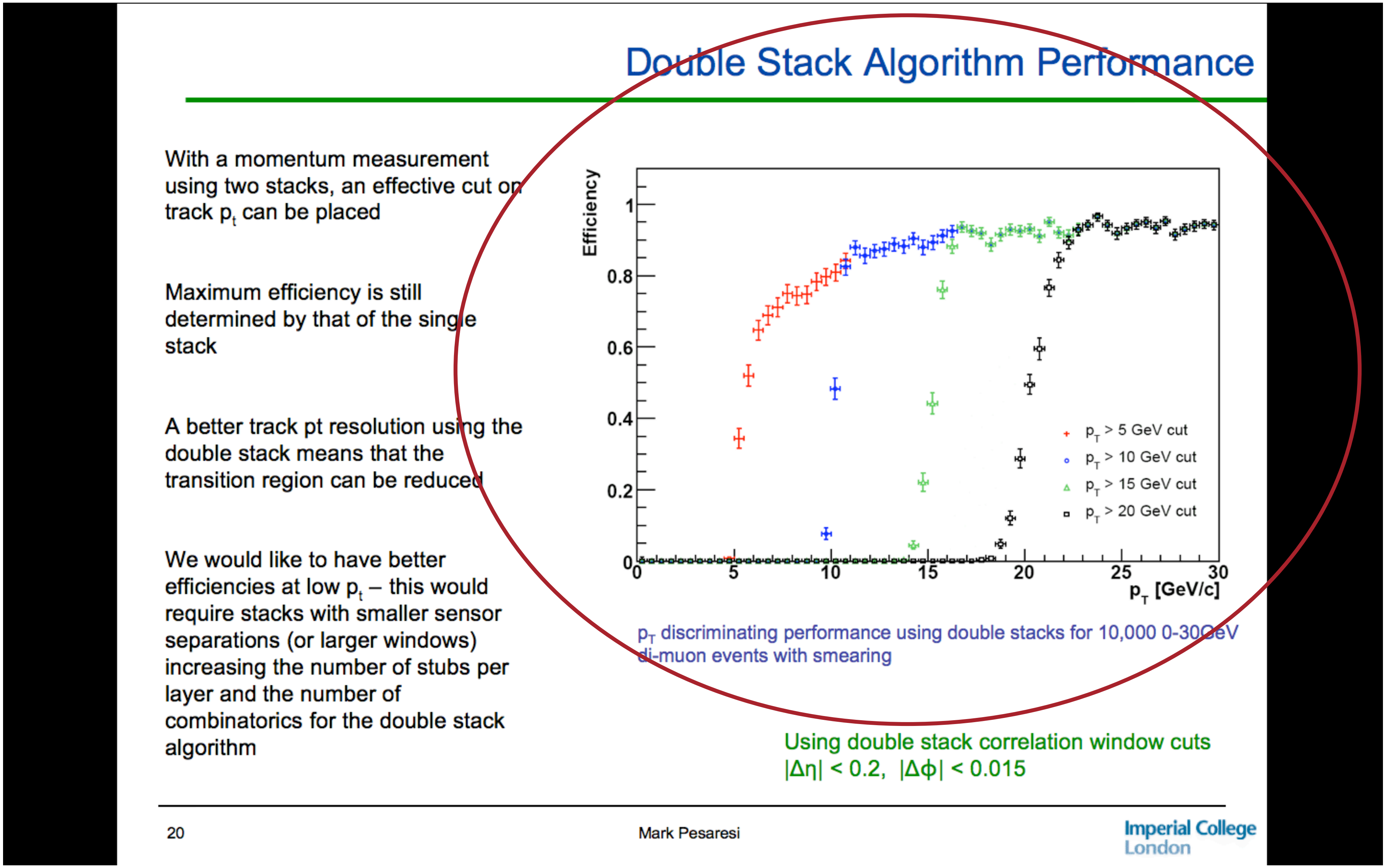
If the stubs are correlated, we can use the two stubs plus the vertex as r,ϕ points for a 3-point track p_t measurement – assumes track originates from (0,0)

Tracklet p_t resolution vs. track p_t and η when using a 3-point pt reconstruction measurement for 10,000 0-30GeV di-muon events with smearing

Using double stack correlation window cuts
$|\Delta\eta| < 0.2$, $|\Delta\phi| < 0.015$

19 Mark Pesaresi Imperial College London

Mark Pesaresi’s Doublet Study



Double Stack Algorithm Performance

With a momentum measurement using two stacks, an effective cut on track p_t can be placed

Maximum efficiency is still determined by that of the single stack

A better track pt resolution using the double stack means that the transition region can be reduced

We would like to have better efficiencies at low p_t – this would require stacks with smaller sensor separations (or larger windows) increasing the number of stubs per layer and the number of combinatorics for the double stack algorithm

p_T discriminating performance using double stacks for 10,000 0-30GeV di-muon events with smearing

Using double stack correlation window cuts
$|\Delta\eta| < 0.2$, $|\Delta\phi| < 0.015$

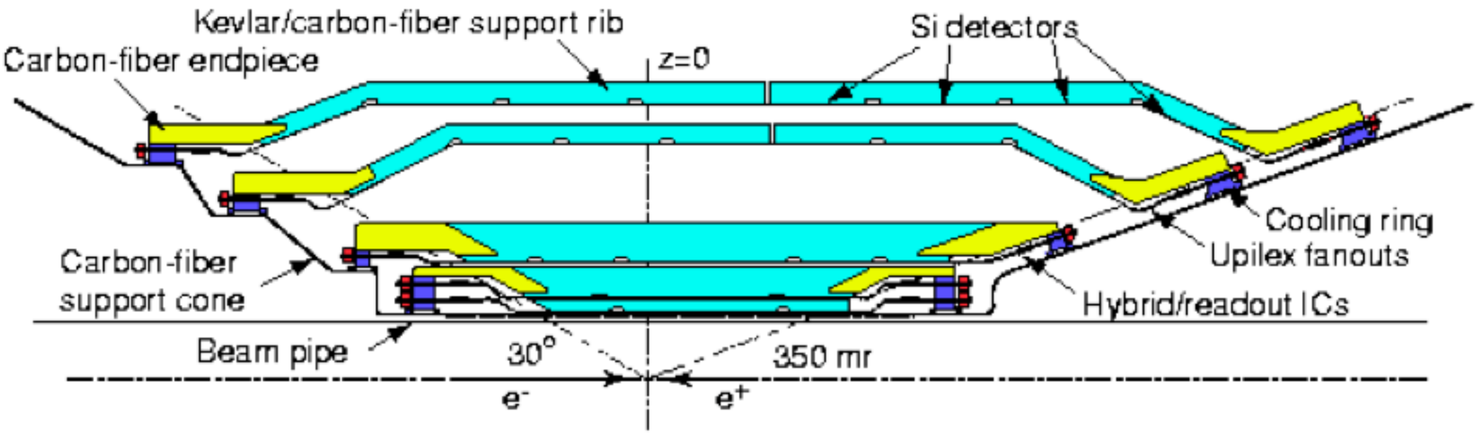
Track Trigger Layers

- Top priority to see whether a (buildable) trigger doublet would work, how many are needed and what their parameters should be
  - This can be studied in any of the strawman geometries
    - Want to study both a single doublet and a “stack” of 2 doublets
  - Mark Pesaresi is studying trigger doublet performance in Strawman B
    - Studying p_T thresholds for both a single doublet and pair of doublets
  - Eric Brownson and Matthew Jones looking at the L1 single muon trigger rate with Fastsim,
    - Will study effectiveness of Mark’s trigger doublet points and vectors
- How much does the performance of trigger doublets depend on
  - Exact Structure of the doublets?
  - Material of doublets and whole construction?
  - Need a robust trigger…
- What are workable alternatives?
  - Fabrizio Palla is studying track triggering using cluster shapes
- Must tackle the forward region for track triggering!

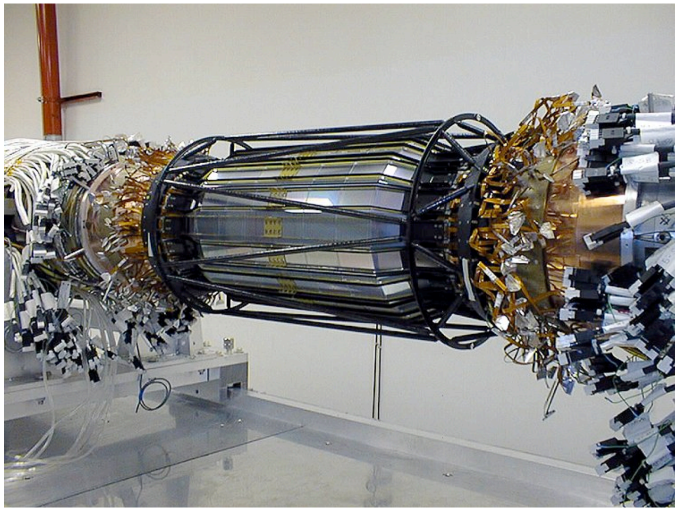
Tracking System Layout

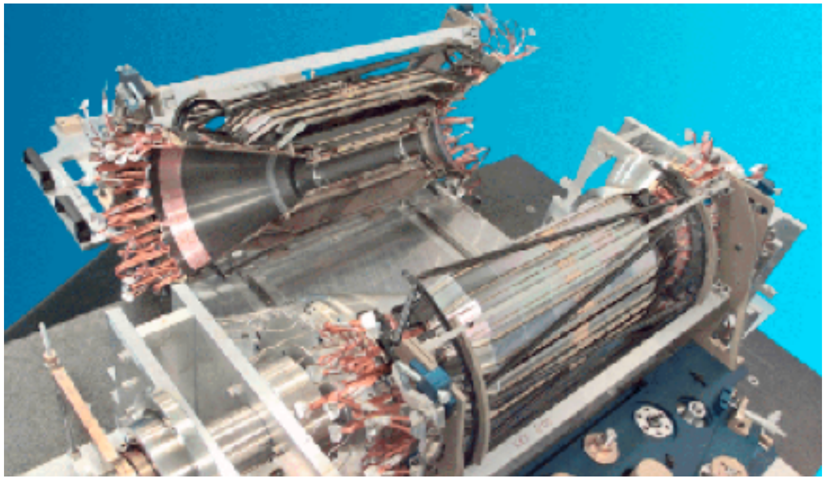
- Once we know what the triggering layers should look like we can start narrowing the tracking system layout variations to decide on a baseline layout
  - Geometry layout tool will be very useful to quickly compare layouts: can compare many statistics (including surface, channels, occupancy, power, cost, bandwidth)
  - Tracker Layout Task Force will have an important role to help us converge to a viable baseline layout geometry (e.g. define realistic ladder and module structures; realistic material budgets and cooling layout; possible channel counts; overall detector construction, etc.)
  - What do we do about track triggering in forward region?
    - Can we give guidelines regarding what is feasible? E.g.
      - Is there more possibility to take data off-detector than in the barrel?
      - Use same technology for correlating forward disks as stacks of doublets?
      - Can we consider a cone/"elliptical" forward detector?

Aside: BaBar SVT



- **5 layers** of double-sided, AC coupled silicon wafers





Tracking System Layout

- Once we know what the triggering layers should look like we can start narrowing the tracking system layout variations to decide on a baseline layout
  - Geometry layout tool will be very useful to quickly compare layouts: can compare many statistics (including surface, channels, occupancy, power, cost, bandwidth)
  - Tracker Layout Task Force will have an important role to help us converge to a viable baseline layout geometry (e.g. define realistic ladder and module structures; realistic material budgets and cooling layout; possible channel counts; overall detector construction, etc.)
  - What do we do about track triggering in forward region?
    - Can we give guidelines regarding what is feasible? E.g.
      - Is there more possibility to take data off-detector than in the barrel?
      - Use same technology for correlating forward disks as stacks of doublets?
      - Can we consider a cone/"elliptical" forward detector?
- Setting up a new geometry layout in the simulation
  - Once we have an idea of what the baseline layout looks like we can build the new layout relatively easily
    - How much configurability?

Non-standard FastSimulation

- Timings for non-standard fastsim with digis and (std.) pattern recognition
  - Timings per event for H → ZZ → 4(μ or e) (based on 1_8_4)
  - On cmslpc (2GHz Intel Xeon)
- No CPU/memory performance issues for generating fastsim samples
  - Need to know importance of out-of-time pileup

Av. Pileup per crossing	Std Fastsim with tracking (sec/event)	Fastsim with Digis (sec/event)		Fullsim (Geant) (sec/event)	
		Digis only	With full track reco	Digis only	With full track reco
0	0.51	0.91	2.38	99.0	101.9
5	0.78	1.27	3.75	119.7	131.3
20	1.84	2.57	11.63	147.1	341.2
40	3.40	4.19	28.48	185.3	1527.3
100	7.35	9.10	162.8	302.6	
200	14.00	17.20	755.3 (seg fault)	539.0 (mem prob)	
400	28.51				

Fastsim Complications: pileup

- We need to simulate at high luminosity
  - Full (Geant4) simulation uses the Mixing Module for pileup
    - Uses min-bias data for pileup from -5 to +3 buckets, merge in simhits
    - Takes lots of memory (improvements in later versions)
  - Fast simulation has only in-time (same bucket) pileup (ok for trigger?)
    - Using min-bias data in same bucket, merge in particles (to generated)
    - Plan to use Mixing Module in a later release
      - Standard uses for simhits ⇒ would need separate minbias files for each new geometry
  - Fastsim tracking detector occupancy differs from Full simulation
    - No out-of-time pileup
    - Fast sim places cuts on minimum track p_t and loopers by default
    - Fast sim does not simulate delta rays
    - Occupancy [%] for pixel layers in MinBias events at pileup ~20
      (modified to lower p_t cut and turn on loopers)

	FullSim	FastSim	Ratio	FullSim (in-time)	FastSim (modified)	Ratio
PXB Layer 1	0.01731	0.007713	2.2	0.01627	0.01252	1.2
PXB Layer 2	0.01253	0.00495	2.5	0.01138	0.00853	1.3
PXB Layer 3	0.01024	0.00363	2.8	0.00938	0.00697	1.3

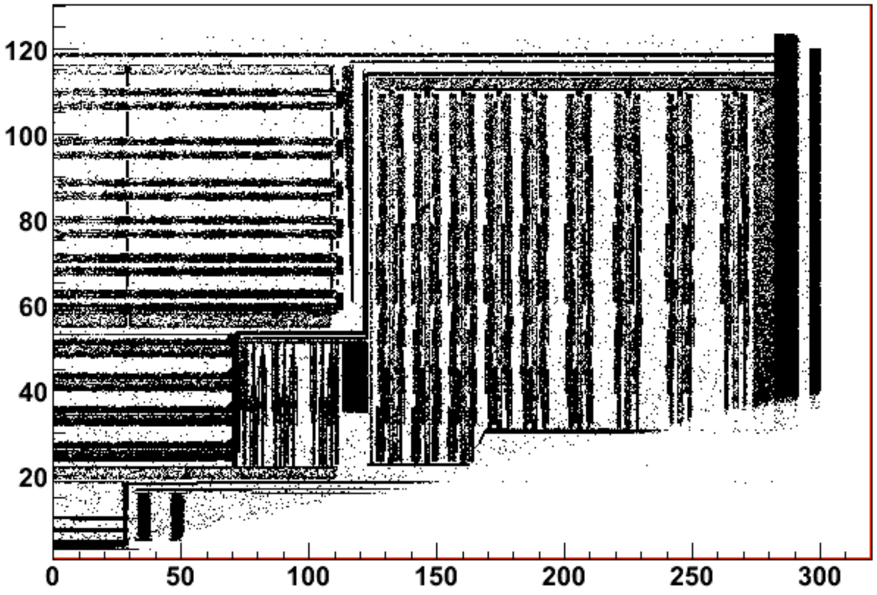
- Need realistic occupancies ⇒ Important to learn when we get real data!

Geometry: Materials

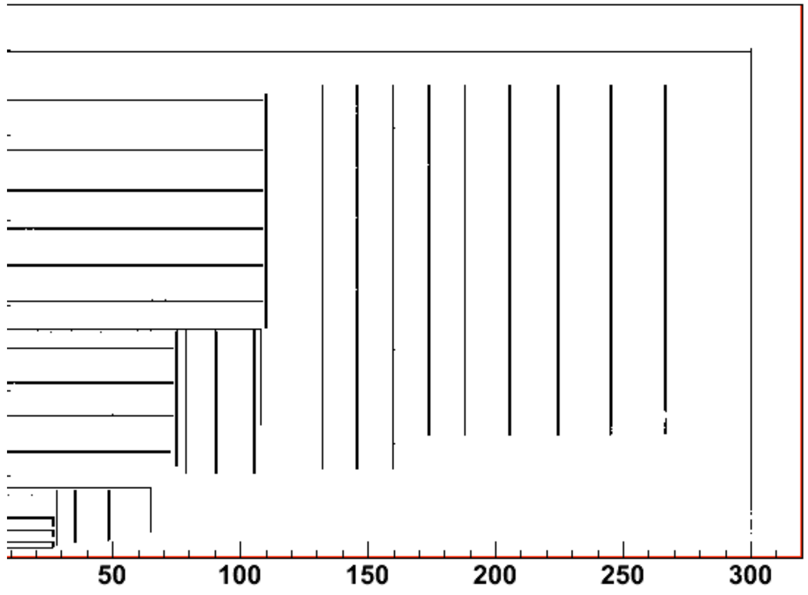
- The FastSimulation geometry uses two geometries
  - Standard Reconstruction (Reco) geometry for location of simhits
  - Separate “interaction geometry” used to trace particles/interactions, consists of nested thin cylinders, “sensor layer” + material layers **tuned and hard coded** to approximate as best as possible the full geometry radiation map
  - <u>We need to get the correct material for each geometry we make</u>

Full Tracker radiography

Geant4 Simulation geometry



FastSimulation Interaction geometry



Current scope for simulation

- Current simulation studies with limited manpower: simulation studies we expect to make progress in the next few months

1. Studies to see whether a (buildable) trigger doublet would work, how many are needed and what their parameters should be
   - Can be studied in any strawman geometry - Mark is using strawman B
   - Will look track doublet info for the L1 single muon trigger rate
2. Studies of a Phase 1 strawman (Roland's options for pixel replacement/upgrade)
   - Including a study of a 4th barrel pixel layer
   - We need to define the Phase 1 Forward Pixel detector
3. Studies of a very long barrel detector of (mini-)strips
   - Study Phase 2 forward region options and doublet at large radius?

Summary

- A substantial amount of work has been done by the upgrade simulation working group to create software to run upgrade simulations (I would like to thank all the people who contributed over the time on this!)
  - We can generate simulations for a number of tracking strawman geometries
  - We have started some simulation studies but have limited manpower
  - Still some issues to deal with for SLHC simulations
    - FastSimulation: out-of-time pileup, occupancy, port geometry to 2_1_X
    - FullSimulation: CPU and memory performance, port geometry to 2_1_X
- Have a focused program to look at tracker doublet performance
  - Mark Pesaresi's doublet study is very encouraging!
    - Still work to be done, e.g. study efficiency in pileup conditions
    - Need to work out realistic/buildable doublet structures
  - Performance for L1 single muon rate will be really interesting!
- Many choices for Phase 2 tracking system layouts
  - Work with Layout task Force to define a baseline strawman in 6-7 months
- We will learn a lot from real data! Flexible enough to use what we learn?

Backup Slides

Upgrade simulation WG

- Tracker upgrade simulation working group
  - https://twiki.cern.ch/twiki/bin/view/CMS/SLHCTrackerSimuSoftTools
    HyperNews: hn-cms-slhc-trackersim@cern.ch

Tracker Upgrade Simulations Working Group

Alice Bean (KU), Avdhesh Chandra (Riverside),

Harry Cheung (FNAL, co-coordinator), Carlo Civinini (Firenze),

John Ellison (Riverside), Kevin Givens (Colorado),

Erik Gottschalk (FNAL), Xingtao Huang (U. of Puerto Rico),

Teruki Kamon (Texas A&M), Matthew Jones (Purdue),

Hector Mendez *(U. of Puerto Rico), Mark Pesaresi (Imperial College),*

Roberto Rossin (UCSB), Jennifer Sibille (KU), Scott Swain (UCSB),

Alessia Tricomi (Catania, co-coordinator), Michael Weinberger (Texas A&M)