HLT (and other things) with GPU

N. Katayama

CHEP 2010, Taipei

Oct 21, 2010

Outline

Introduction

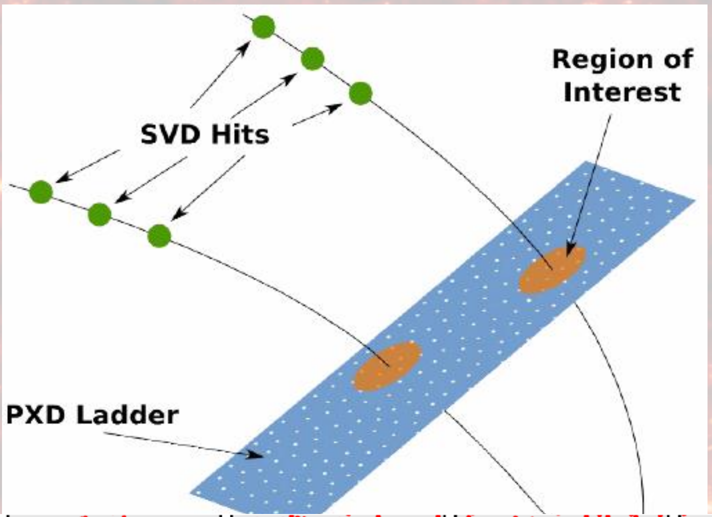
Inflation, CMB and GPU

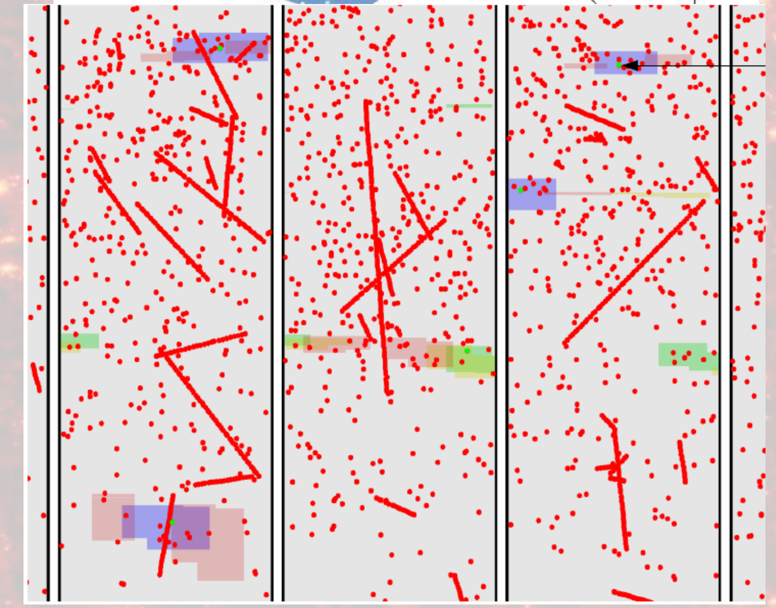
GPU/many core CPU

- We have been using event (trivial) parallelism in our experiments’ data reduction/analysis
  - Each event is independent
  - Each event is not so large
  - We always have many events
- Now I have met HEP applications which could make use of GPU
  - “Event” is large (correlation in a large data sample)
  - A lot of computation for an “event”
  - (Very similar to) image processing

Pixel detector for Belle II

- Belle II will take several tens more data than Belle
  - Lots of backgrounds
    - From strip detector to pixel detector
  - 100 times more data
    - Real time data reduction in software
  - Red dots are hits. We want to keep hits in blue region (ROI)
- We need to do it online





Belle II High Level Trigger

This is option #3.
There is a plan to use
FPGA to reduce PXD
h

PXD

Other detectors

Sorry. We have just started to work on GPU here and I don’t have much to report yet

You’ve heard about Belle HLT in the morning

Inflation, CMB and GPU

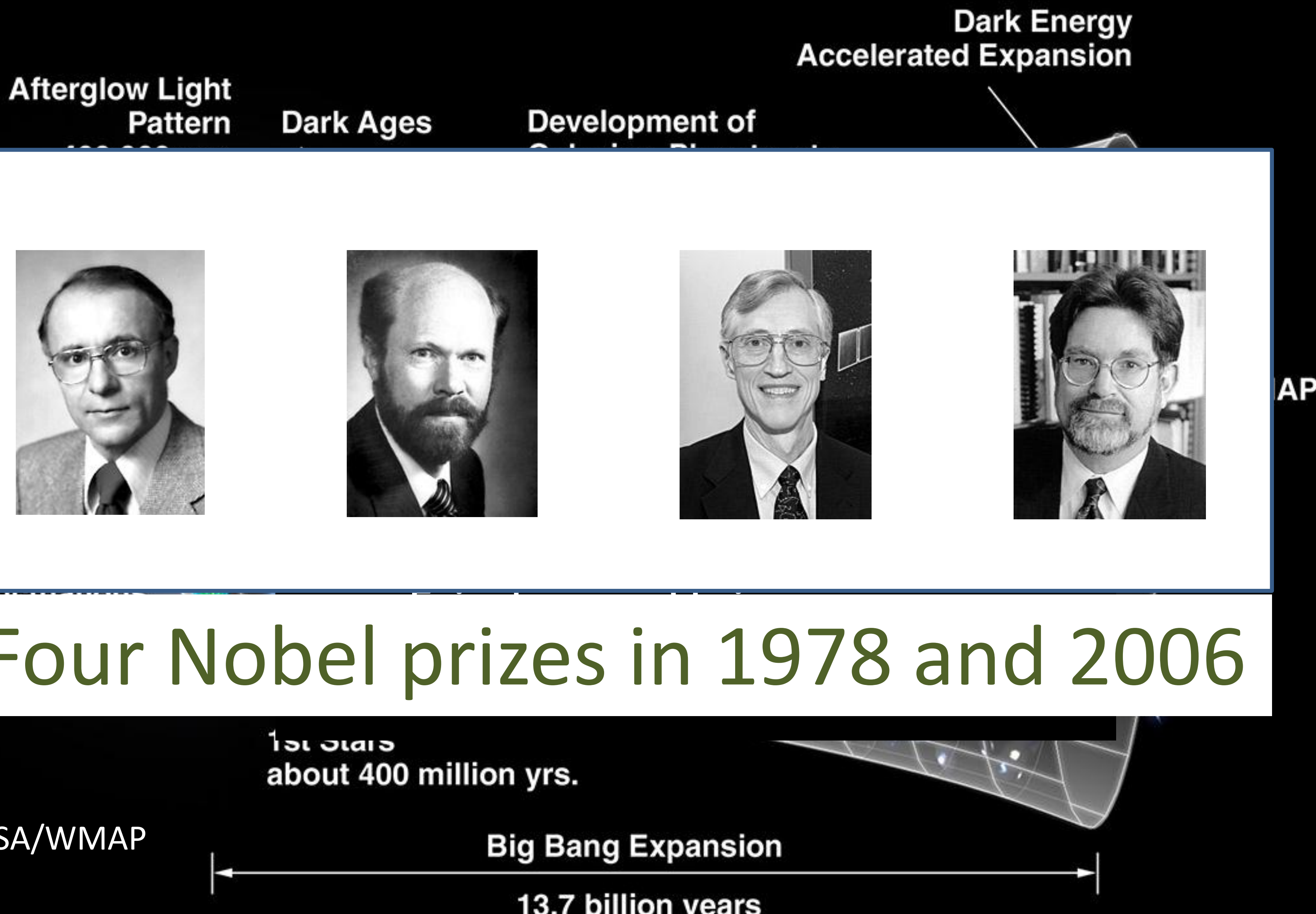
This really is high energy physics

Energy scale of the inflation is $\sim 10^{16}$GeV

Inflation

- Inflation started 10^{-36} sec. after the birth of our universe and lasted for 10^{-34} sec.
- During that period, the universe expanded of the order of e^{60}, from Plank scale to a meter or so (our observable universe)
- Inflation was caused by a particle (field) of energy scale of 10^{16} GeV
- Cosmic Microwave Background Radiation (CMB) is the probe to measure its energy scale
  - Let me use the parameter **"r"** to represent the scale

CMB is the fossil light from the Big Bang



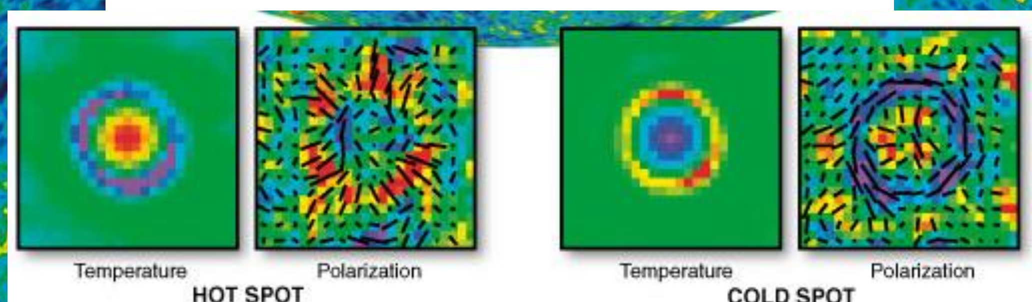
Four Nobel prizes in 1978 and 2006

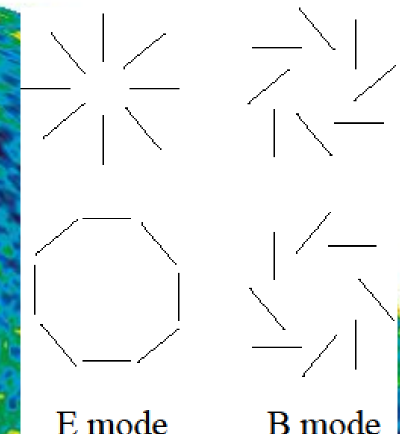
NASA/WMAP

Spatial anisotropy of CMB

CMB can be polarized

WMAP 7yrs result(2010)



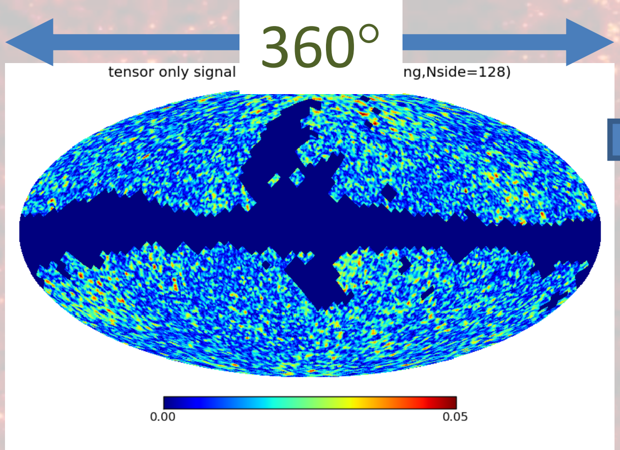


- Overlay many hot and cold spots and see polarization around the spots
- A clear correlation of temperature and E-mode seen

WMAP 5-year

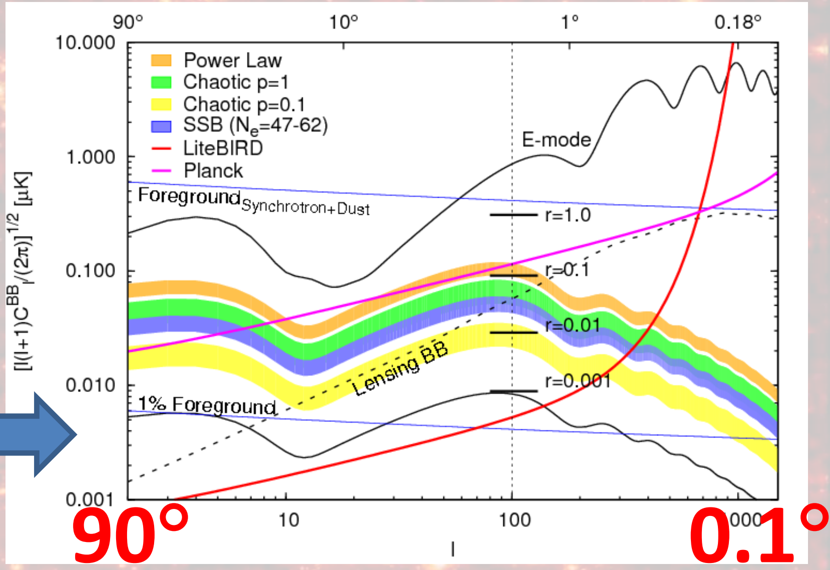
B mode polarization and inflation

- Quantum fluctuation ⇒ Space-time fluctuation ⇒ (thru Inflation) ⇒ Gravitational wave (GW)
- GW h_x (h_+) produces B (E) mode polarization
  - **B mode polarization can only be produced by GRAVITY**
- B mode polarization is 10^{-8} smaller than temperature
- We want to observe B mode polarization of the size 1–45 degree (field of view)



We need to go to <u>space</u>

B mode power spectrum



Likelihood method

- CMB is Gaussian on pixels (=region of Universe)
- Pixel space likelihood ⇒ most sensitive, unbiased but computationally intensive (Large factor $\times n_p^3$)

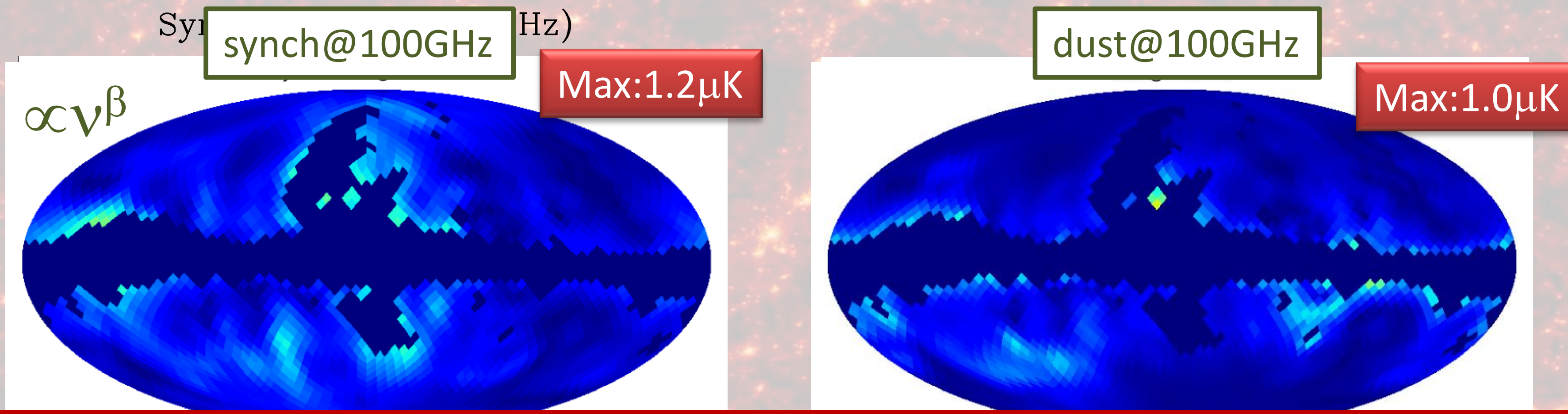
$$L(\boldsymbol{m}|r)d\boldsymbol{m} = \frac{\exp\left[-\frac{1}{2}\boldsymbol{m}^t(S(r)+N)^{-1}\boldsymbol{m}\right]}{|S(r)+N|^{1/2}}\frac{d\boldsymbol{m}}{(2\pi)^{2n_p/2}}$$

where

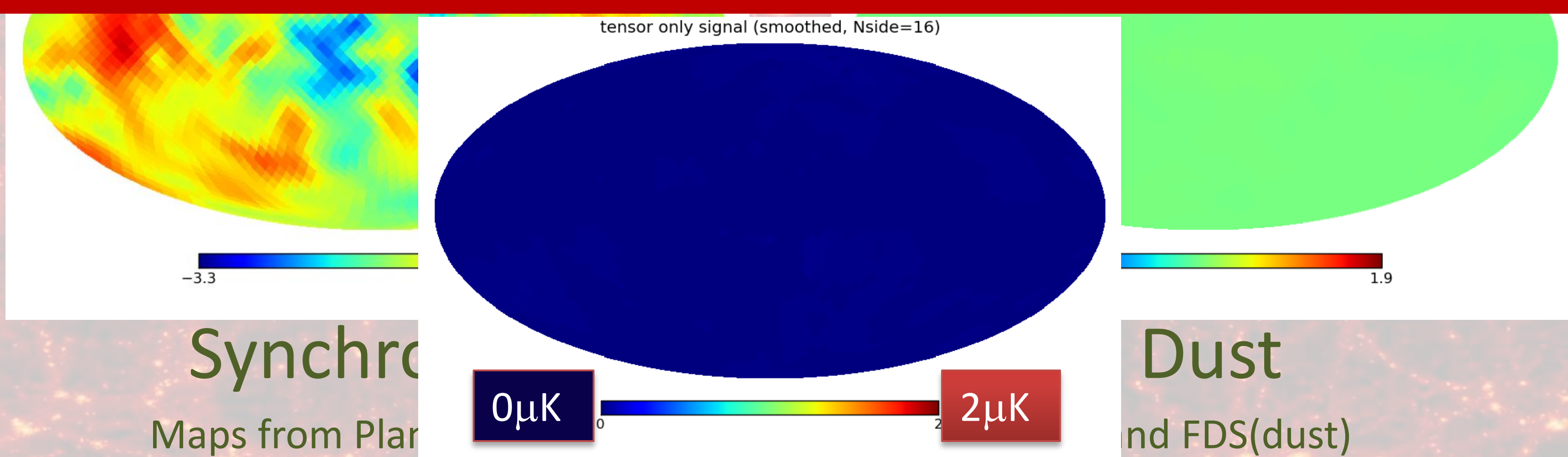
- ***m*** is Q and U map vector of length $2n_p$
- S is signal covariance matrix of ($2n_p \times 2n_p$)
  - S is a function of cosmological parameters
- N is noise covariance matrix
  - N is not diagonal due to correlation

$n_p = 3072$
(~4°)

Galactic foregrounds



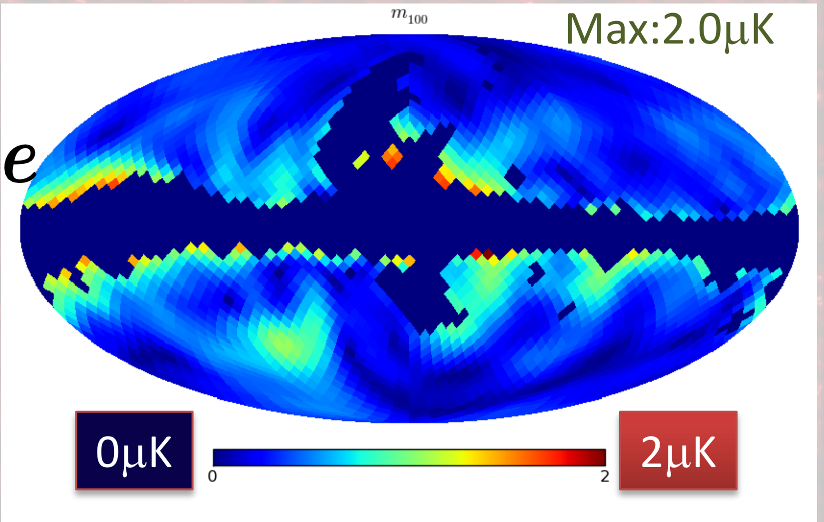
Remove foregrounds with multiband maps assuming morphology



Dust cleaning with 2 bands

- Simple cleaning (E. Komatsu)

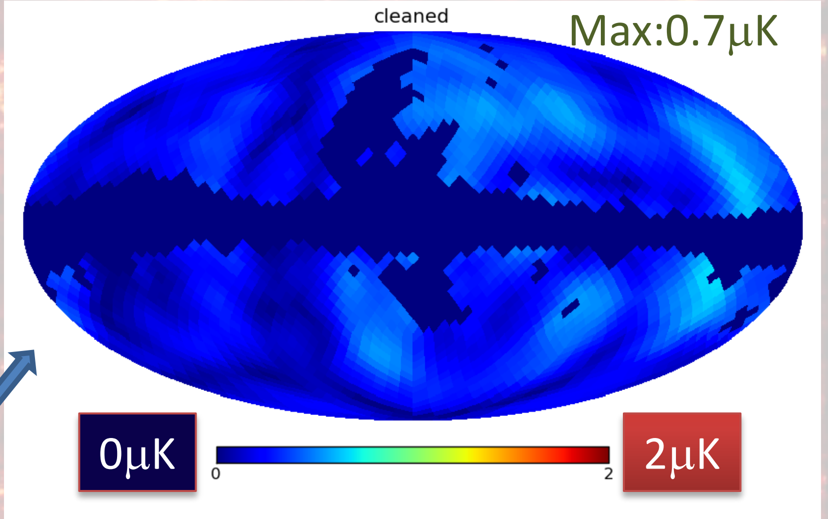
$$\begin{aligned} \boldsymbol{m}_{100} &= \boldsymbol{cmb} + \boldsymbol{dust}_{100} + \boldsymbol{noise} \\ \boldsymbol{m}_{240} &= \boldsymbol{cmb} + \boldsymbol{dust}_{240} \\ \boldsymbol{dust}(\nu) &= \boldsymbol{dust}_0 (\frac{\nu}{\nu_0})^{\beta} \\ \chi^2 &= (\boldsymbol{m}_{100} - \alpha \boldsymbol{m}_{240})^T (S+N)^{-1} (\boldsymbol{m}_{100} - \alpha \boldsymbol{m}_{240}) \end{aligned}$$



- minimize χ^2 with respect to α
- In L

$$\boldsymbol{m} \Rightarrow \frac{\boldsymbol{m}_{100} - \alpha \boldsymbol{m}_{240}}{1-\alpha}$$

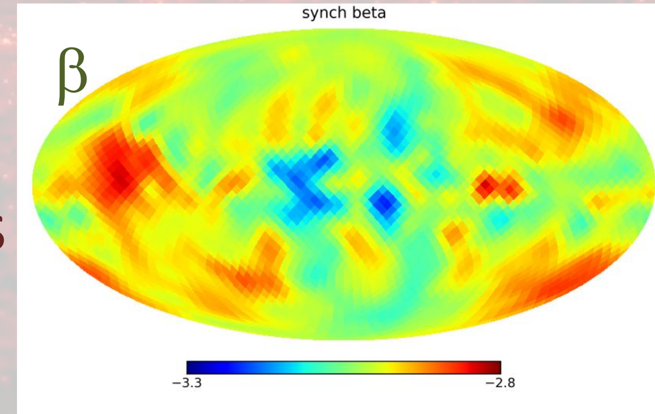
$$N \Rightarrow \frac{N}{(1-\alpha)^2}$$

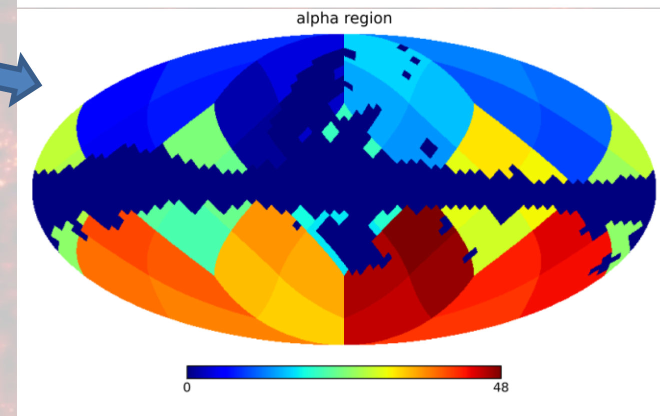


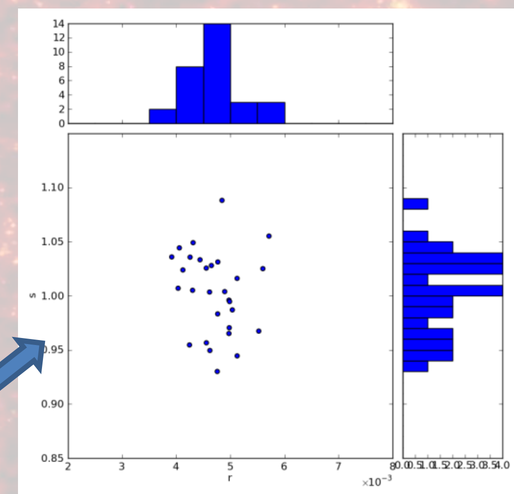
Note: These map include both dust and synch

Dust+Synch cleaning

- Try cleaning dust+synch using 3 bands
- Make $\alpha_{synch}(\theta, \phi)$ dependent
  - 48 alpha regions
- Maximization using 51 parameters
- $(S+N)^{-1}$ depends on all parameters because of $N/(1-\alpha)^2$
- It is computationally prohibitive to maximize L using exact $(S+N)^{-1}$
- Use "nominal alphas" by analytically solving $\dfrac{\partial \chi^2}{\partial \alpha_i} = 0$ and use it in $N/(1-\alpha)^2$
- $(S+N/(1-\alpha)^2)^{-1}$ is a function of r and s

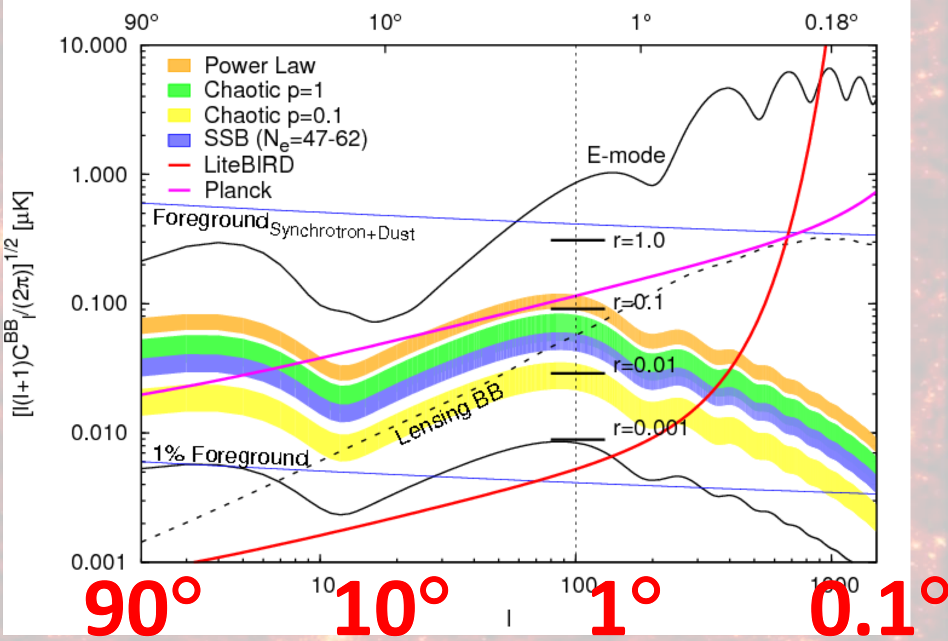






More finer pixels ?

- We've been using the pixel size of 3072 for up to ~4°. We invert matrices of 6144 × 6144 many times while optimizing 51 parameters
- We are not using information from structure smaller then 4°
  - The good part of the spectra extends to ~1°.
  - Matrix size is 4 times larger for ~2° and 16 , ~1°.
  - Computation increases 64 and 4096 times more



One fit will take 1 day for ~2° and 64 days, ~1° on a latest small computer

Use GPUs?

Cholesky decomposition

- We need
  - Log(det|C|) and C^{-1} of the matrices

    We need double precision

    4° — 6144×6144

    2° — 24576×24576

    1° — 98304×98304

  - We can write $C = LL^T$ using Cholesky decomposition
  - Then we can compute L^{-1} and then compute C^{-1}
  - We have been using BLAS/LAPACK (ATLAS, MKL)
  - We started CUDA coding using code examples on the net (http://www.ast.cam.ac.uk/~stg20/cuda/cholesky/index.html)

CPU/GPU Results

	6144×6144 (GFLOPS)	12288×12288 (GFLOPS)
CPU (i-7 920) (numerical recipes)	0.61	0.62
CPU(i-7 X980@3.33GHz) MKL(6core)	68	72
GTX480 (our CUDA code)	108	115
C2050 (our CUDA code)	87	91
C2050 (CULA)	159	190

>×2 faster

- Peak performance of C2050 is supposed to be >500 GFLOPS
- GTX480 DP speed is suppressed to ¼ of C2050 (but is faster)
- CUDA 3.1 and CULA 2.1 are used
- Floats are faster as expected
- Copy from/to CPU to/from GPU not included (but is significant)
- Cholesky decomposition cannot saturate GPU
- Cannot do 24576×24576(not enough memory on one C2050)

Conclusions and Plan

- Traditional (Intel MKL, ATLAS) code does very well
- Will continue to program for GPU
  - $L \Rightarrow L^{-1}$ (dtrtri in lapack)
  - $C^{-1} = L^{-1} (L^{-1})^T$ (dlauum in lapack)
  - Compute Likelihood $\propto X^T C^{-1} X / \det(\log|C|)$
- We need to work on memory footprint to do the larger matrix
- We must learn to efficiently program GPU for each algorithm to get best performance
  - but is promising as the technique can be used on many core machines

backup slides

Likelihood fit procedure

- Assume r is the only unknown cosmological parameter
  - Pre-compute ${}_{\pm 2}Y_{\ell m}$ (p), $C_\ell^{[Q,U]}$(p,p') for $2 \leq \ell \leq 47$
  - Pre-compute $S_{r=1}^{tensor}$ and S^{scalar}, N
  - Pre-compute $(r \times S_{r=1}^{tensor} + S^{scalar} + N)^{-1}$ for r=0.0001, 0.0002, 0.0003,,, and store them

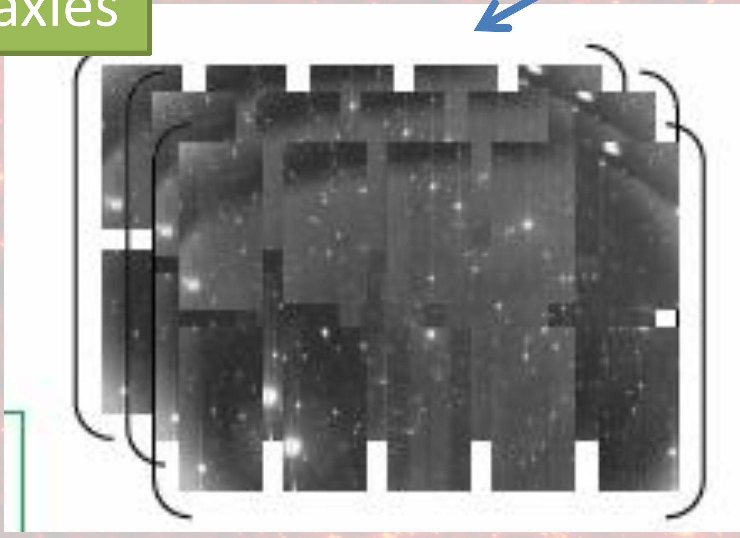
Y_{lm}: Spin weighted Spherical harmonics

- Generate map (N_{side}=128) assuming r=r_{input}, add noise, smooth to N_{side}=16, apply P06 mask
- Compute ln L(**m**|r) for r=0.0001, 0.0002, 0.0003 and find r which gives max. ln L(**m**|r)
  - repeat for many realizations
  - get Δr from 0.5 = Δln L
- Plot resulting r, compute mean of r and Δr, etc.
  - repeat for r_{input} = 0.001, 0.003, 0.01,,,

Search for Dark Energy (HSC)

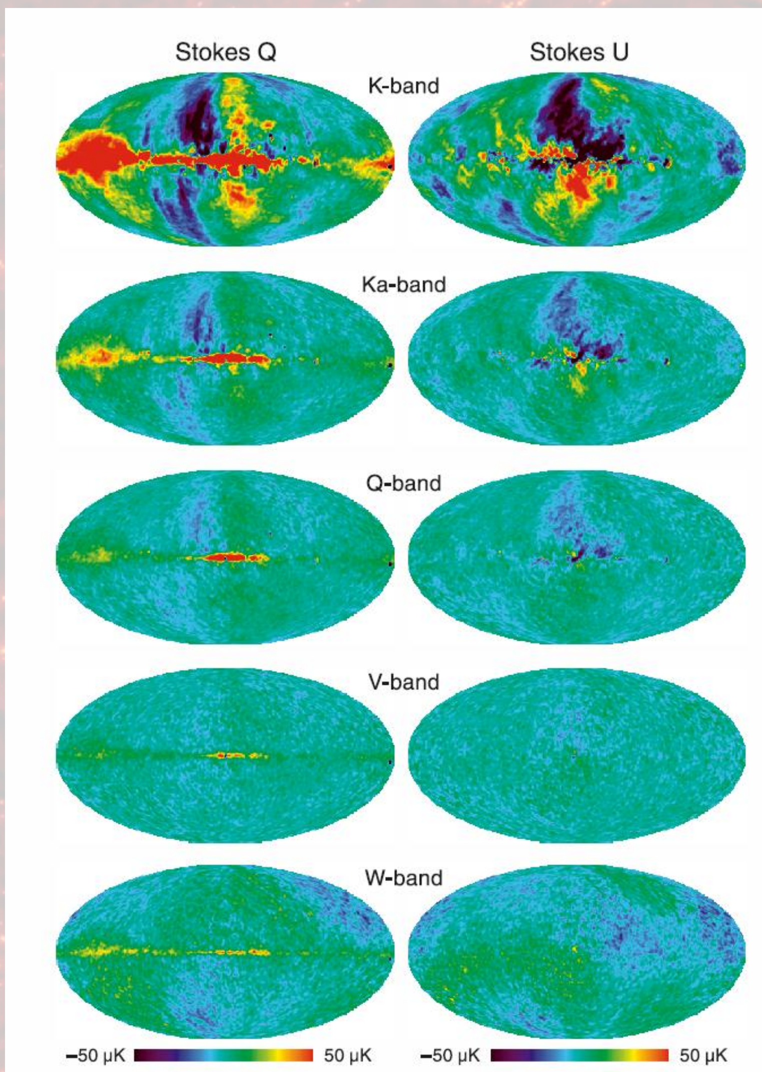
This is HC
10 CCDs

We want to know shapes of many galaxies

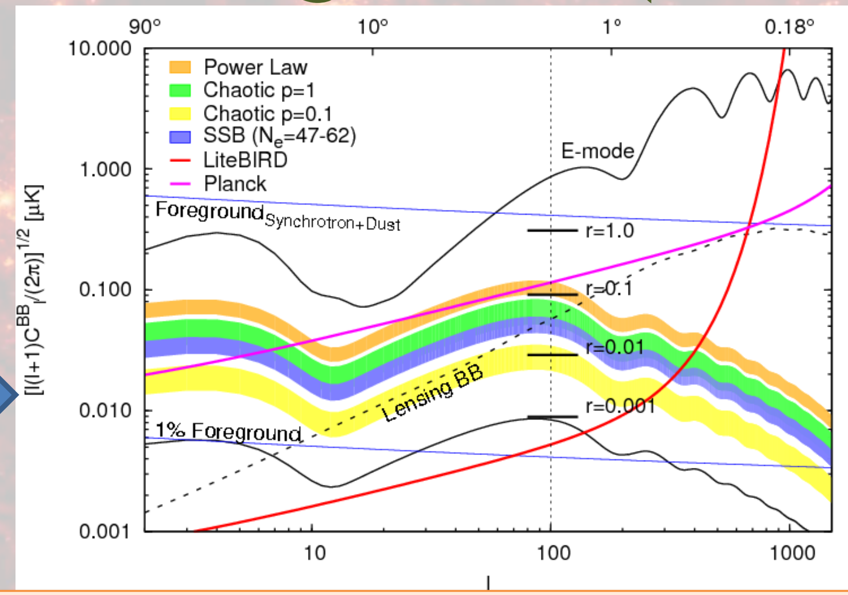


- Hyper Suprime Cam @Subaru telescope will have 104 CCDs (1G pixel per shot)
  - Must stack images
- If possible, we want to do image processing in real time so that we can give feed back to observers
  - Sky changes during observation
  - How many minutes do we need for the next exposure?
- Using known stars, we overlay images

Discover primordial gravitational wave in Cosmic Microwave Background(CMB)



Polarized data from WMAP



In likelihood we need to invert matrix of the size N_p (N_p^3 computations)

LiteBIRD is a small satellite experiment. We need to optimize the design parameter using simulations.

GPU is hot

- GPGPU (general purpose graphics processing unit) is becoming popular
  - In 2008, I was looking at Cell CPUs (used in play station 3) and
  - (Q2/3, 2010)
    - 520-630 double precision Gflops/GPU(peak) as opposed to 78 Gflops for the current generation
    - These numbers do not include transfer time between memories of CPU and GPU

It promises 1/10 of cost and 1/20 of power consumption

C1060

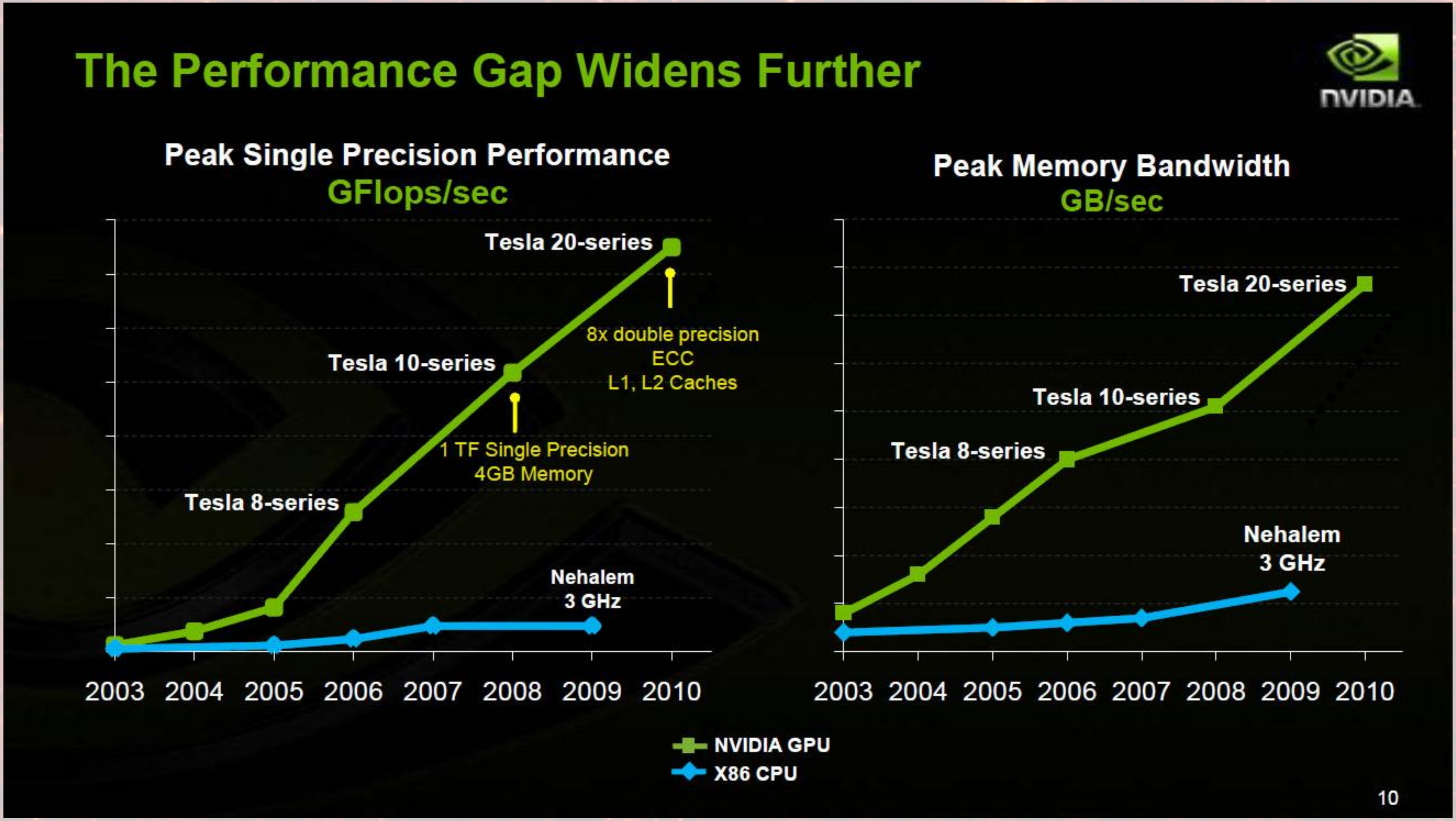
C2070

CPU and GPU

Processor	Intel Core 2 Extreme QX9650	NVIDIA TESLA C2070
Transistors	820 million	1.4 billion
Processor clock	3 GHz	1.3 GHz?
Cores	4	512
Cache / Shared Memory	6 MB x 2	16-48 KB/768KB(L2)
Threads executed per clock	4	512
Hardware threads in flight	4	24576
Memory controllers	Off-die	384bit
Memory Bandwidth	12.8 GBps	64bytes/clock?

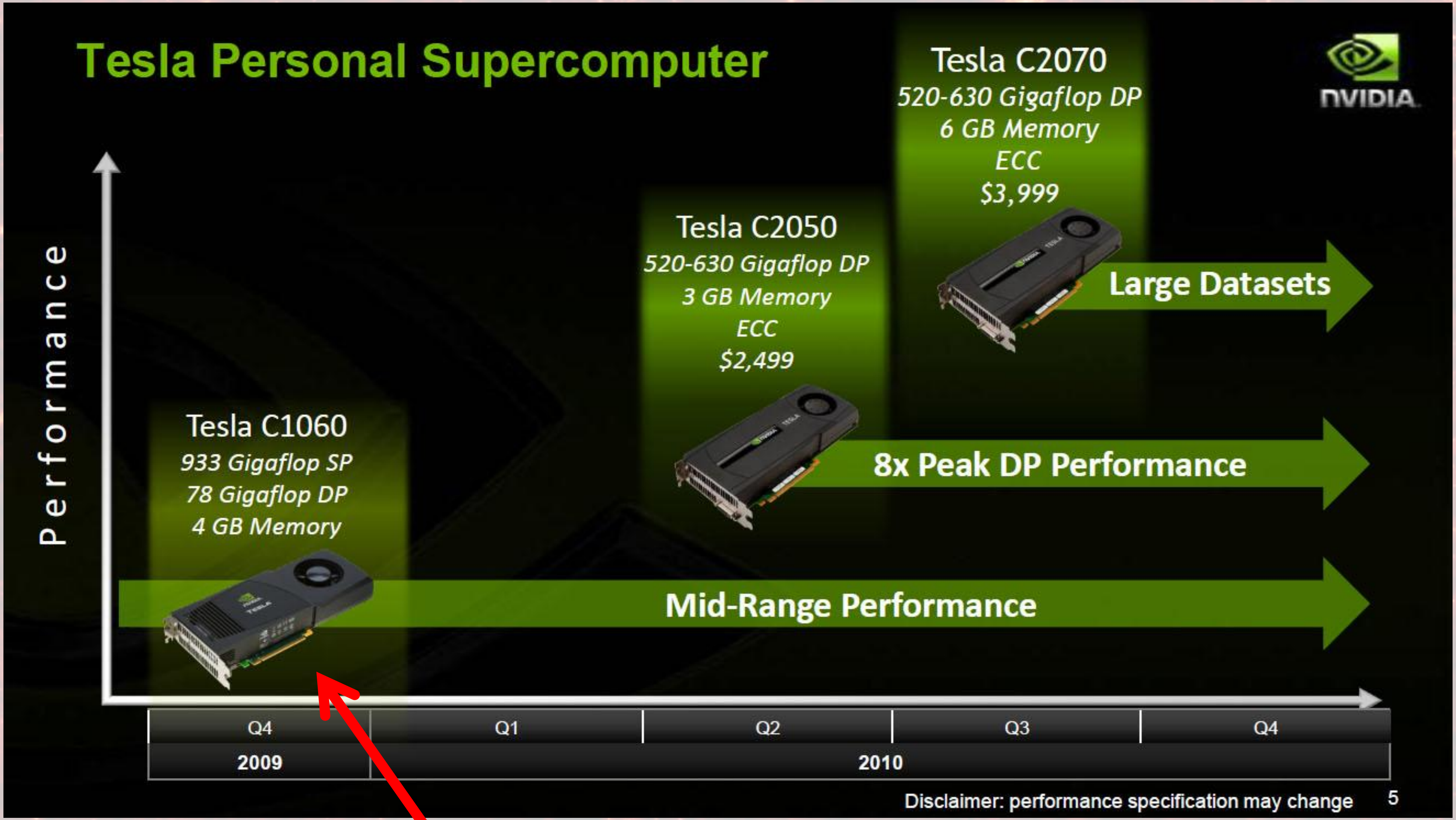
PCIe x16 < 8GB/s aggregate

Fermi GPU



>600 GFLOPS/sec (Double Precision)/chip

It's real



Only <$1,000 now

Belle II

Super KEKB collider

Contribution ID: 561
The SuperKEKB accelerator status

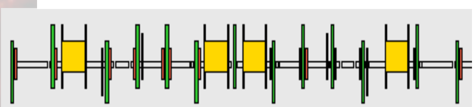
Funded!

Colliding

cting

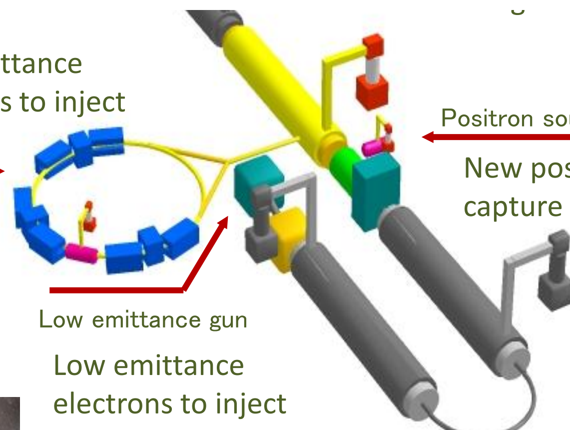
ear the

Repl
with

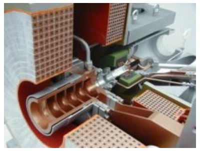


Low emittance positrons to inject

Redesign the lattices of HER & LER to squeeze the emittance

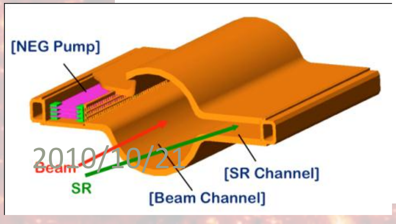


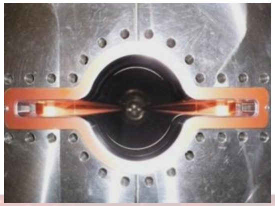
New positron target / capture section



Low emittance electrons to inject

TiN-coated beam pipe with antechambers





$$L = \frac{\gamma_\pm}{2er_e}\left(1+\frac{\sigma_y^*}{\sigma_x^*}\right)\left(\frac{I_\pm \xi_{\pm y}}{\beta_y^*}\right)\left(\frac{R_L}{R_y}\right)$$

Target: L = $8\times10^{35}/cm^2/s$

Belle II HLT

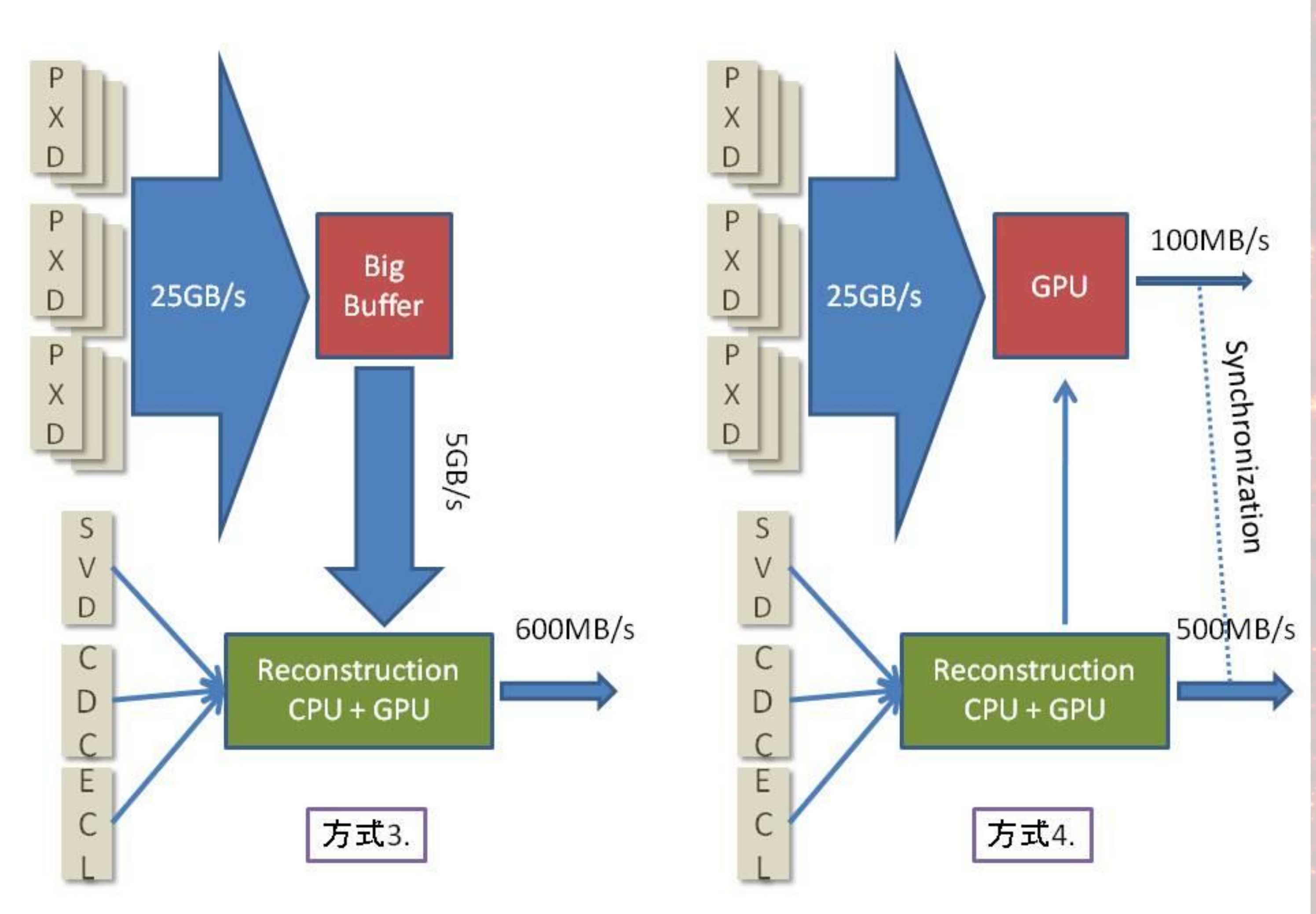
The biggest computing challenge in Belle II

Belle II HLT

- Belle HLT working well (200 CPU core or so)
  - It takes 0.4 second to analyze one Hadronic event
  - L1 trigger rate is ~500Hz, HLT output rate is ~200Hz
  - Real Hadronic events: 100Hz
- Belle II HLT: much harder problem
  - L1 trigger rate 30KHz
  - HLT will reduce to 5KHz
    - Real Hadronic events:5KHz
  - Pixel detector produces 25GB/s data
    - We need to reduce down to 100MB/s or so

10,000 cores needed

Two ways



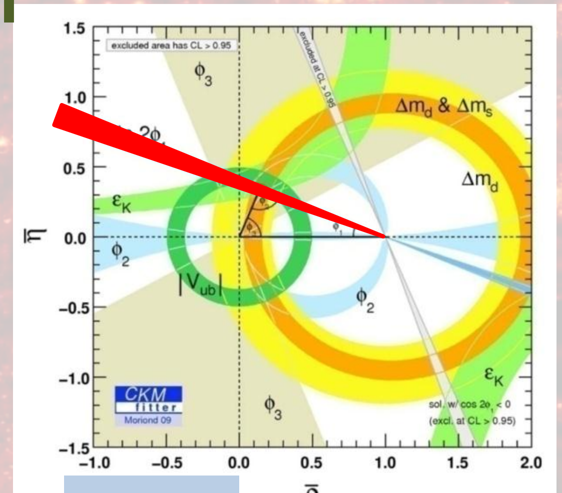
P X D
P X D
P X D
25GB/s
Big Buffer
5GB/s
S V D
C D C
E C L
Reconstruction CPU + GPU
600MB/s
方式3.
P X D
P X D
P X D
25GB/s
GPU
100MB/s
Synchronization
S V D
C D C
E C L
Reconstruction CPU + GPU
500MB/s
方式4.

B Physics with 50ab^{-1}

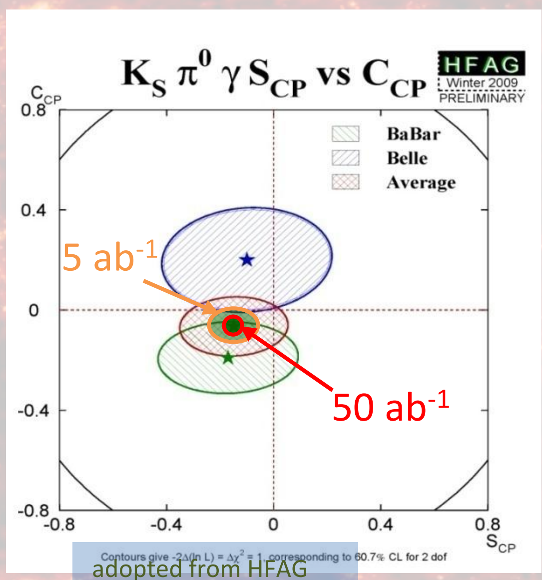
"bread & butter":

- *B* factory:
  $B \to J/\psi K_S$, $\mathcal{B} \sim 4.5 \cdot 10^{-4}$
  $\delta S \sim 0.03$

- Super *B* factory:
  $B \to K^*\gamma$, $\mathcal{B} \sim 4.0 \cdot 10^{-5}$
  $\delta S \sim 0.2 \to \sim$a few %



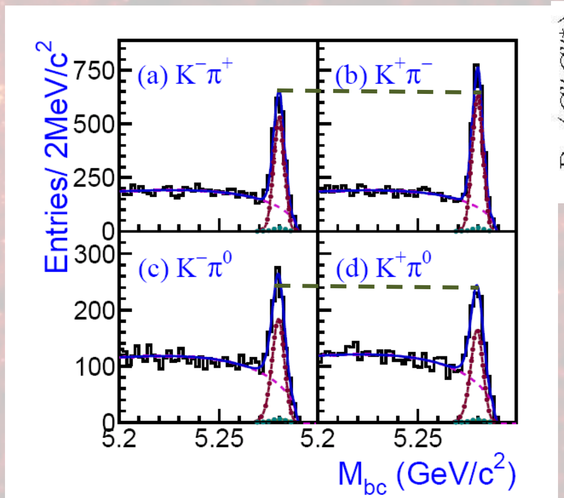
CKM Fitter



adopted from HFAG

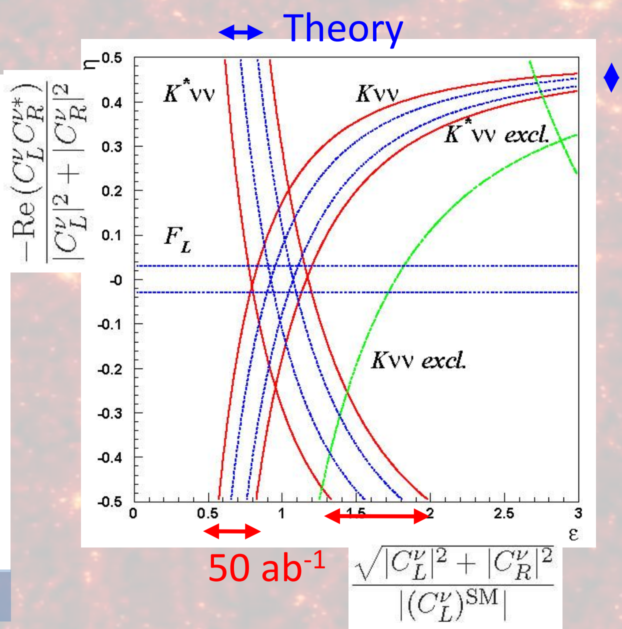
Interesting modes for NP:

- *B* factory:
  $B \to K\pi$, $\mathcal{B} \sim 1.9 \cdot 10^{-5}$



Belle, Nature 452, 332 (2008), 480 fb^{-1}

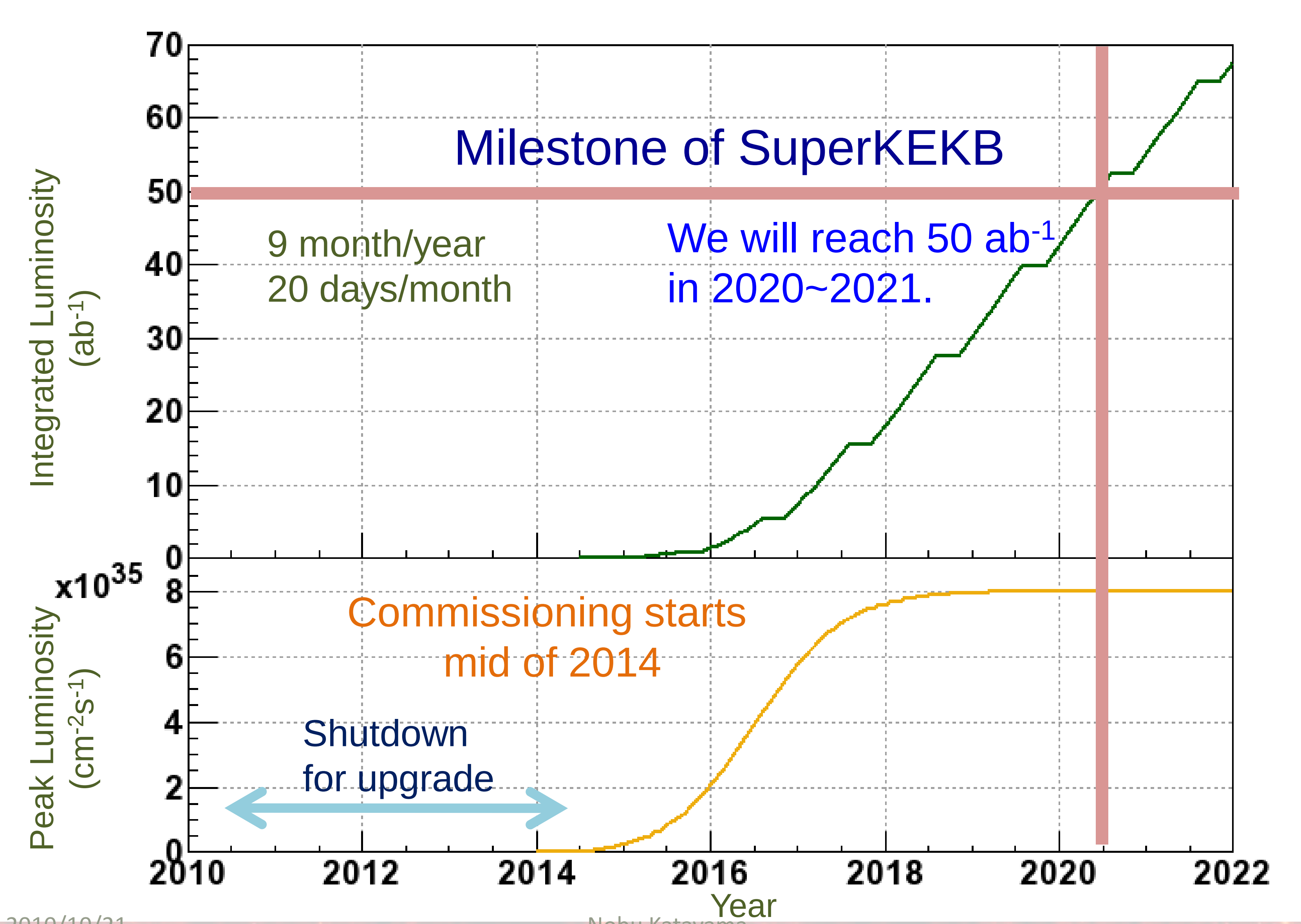
- Super *B* factory:
  $B \to K\nu\nu$, $\mathcal{B} \sim 4 \cdot 10^{-6}$



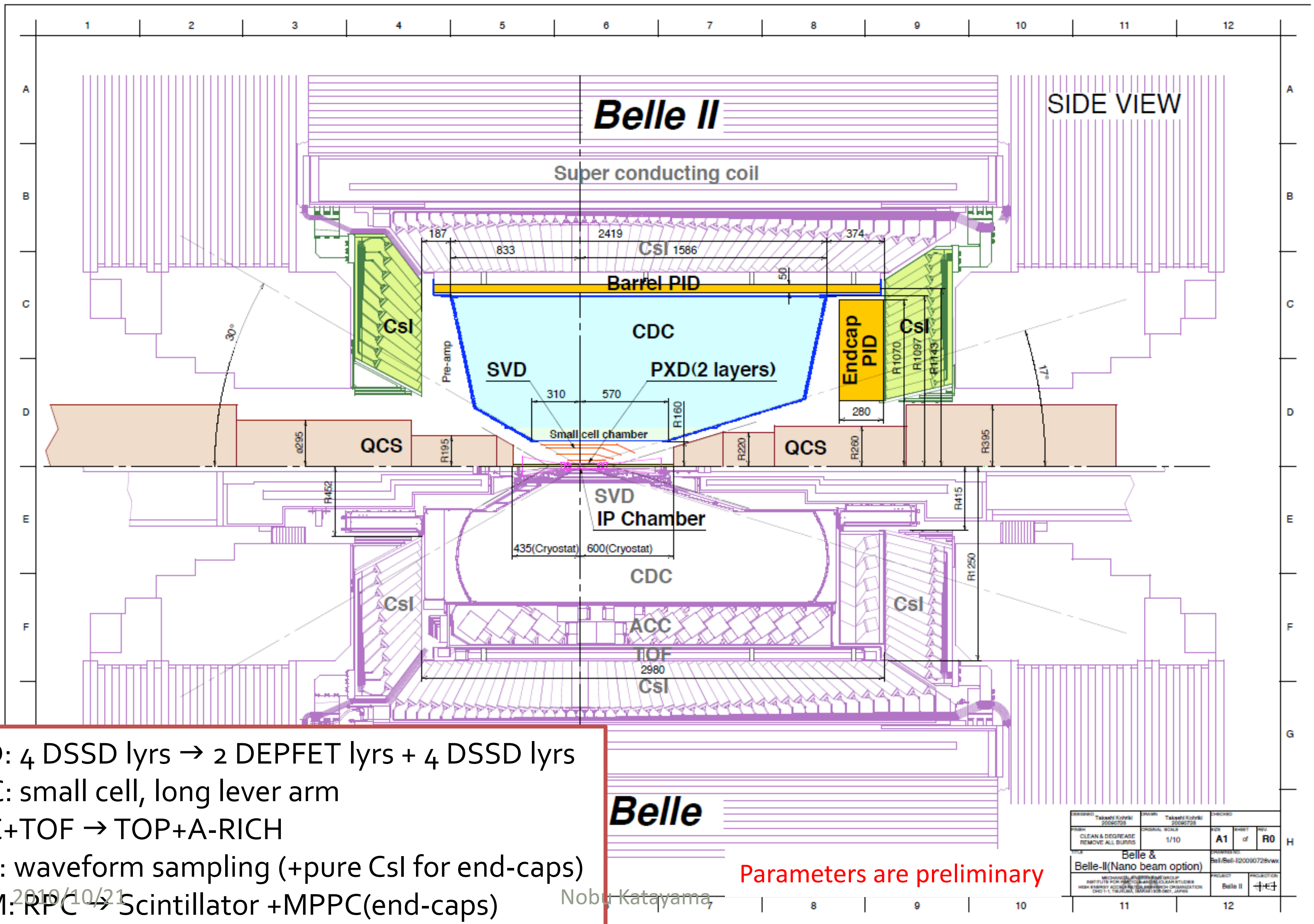
adopted from W. Altmannshofer et al., JHEP 0904, 022 (2009)

$S(K_S\pi^0\gamma)$, $K\nu\nu$, $\tau\nu$, … are not possible @ LHCb

Luminosity upgrade projection



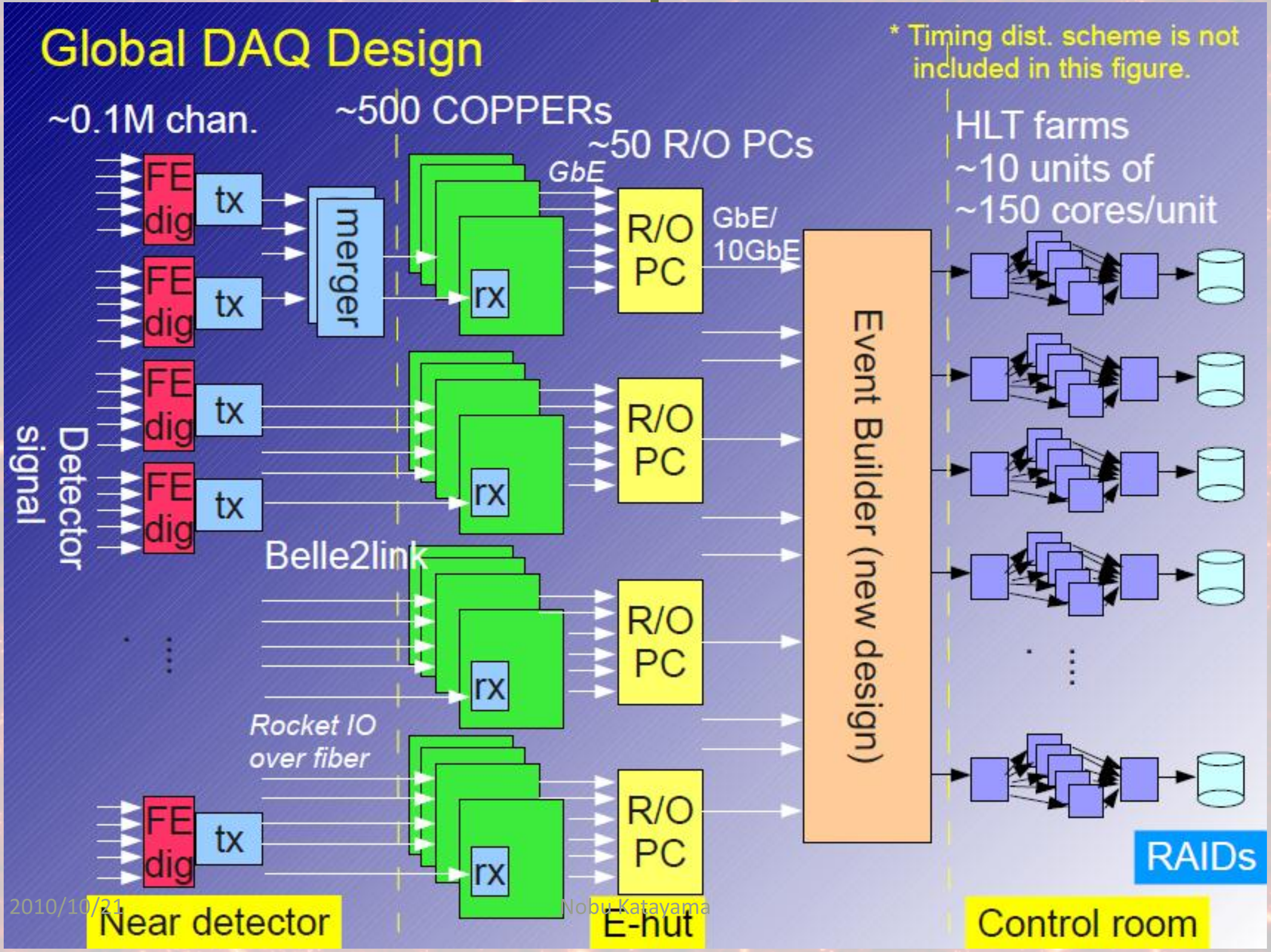
Y. Ohnishi



Parameters are preliminary

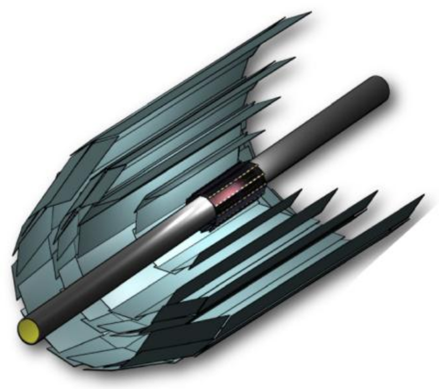
SVD: 4 DSSD lyrs → 2 DEPFET lyrs + 4 DSSD lyrs
CDC: small cell, long lever arm
ACC+TOF → TOP+A-RICH
ECL: waveform sampling (+pure CsI for end-caps)
KLM: RPC → Scintillator +MPPC(end-caps)

Data Acquisition

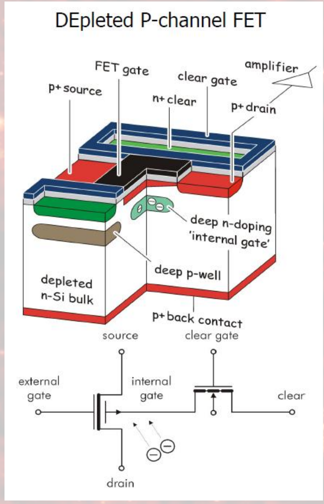


Vertex Detector

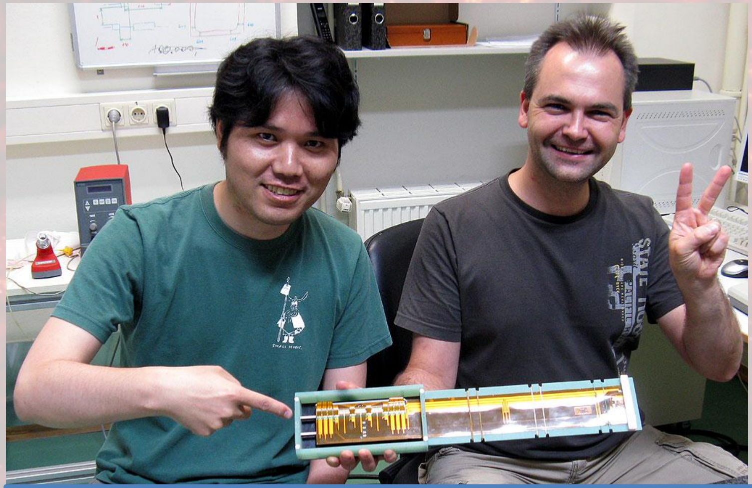
DEPFET:
http://aldebaran.hll.mpg.de/twiki/bin/view/DEPFET/WebHome



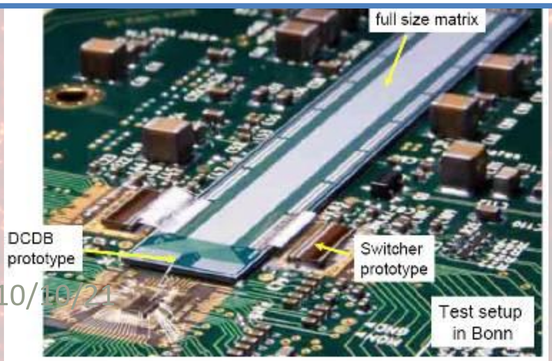
Beam Pipe		**r = 10mm**
DEPFET		
	Layer 1	**r = 14mm**
	Layer 2	**r = 22mm**
DSSD		
	Layer 3	**r = 38mm**
	Layer 4	**r = 80mm**
	Layer 5	**r = 115mm**
	Layer 6	**r = 140mm**



Mechanical mockup of pixel detector



Prototype DEPFET pixel sensor and readout



A prototype ladder using the first 6 inch DSSD from Hamamatsu has been assembled and tested.

Exact covariance matrix

- Elements of S can be written in terms of spin-weighted $Y_{\ell m}$ and C_ℓ
  - S can be written as $r \times S^{tensor} + S^{scalar}$
  - but need to compute $(S+N)^{-1}$ each time

$$
\begin{aligned}
C_{QQ}(x,y) &= \sum_l C_l^{EE} \sum_m W_{lm}(x) W^*_{lm}(y) + C_l^{BB} \sum_m X_{lm}(x) X^*_{lm}(y) \\
C_{QU}(x,y) &= \sum_l C_l^{EE} \sum_m (-W_{lm}(x) X^*_{lm}(y)) + C_l^{BB} \sum_m X_{lm}(x) W^*_{lm}(y) \\
C_{UQ}(x,y) &= \sum_l C_l^{EE} \sum_m (-X_{lm}(x) W^*_{lm}(y)) + C_l^{BB} \sum_m W_{lm}(x) X^*_{lm}(y) \\
C_{UU}(x,y) &= \sum_l C_l^{EE} \sum_m X_{lm}(x) X^*_{lm}(y) + C_l^{BB} \sum_m W_{lm}(x) W^*_{lm}(y)
\end{aligned}
$$

$$
\begin{aligned}
W_{lm}(x) &= (-1)({}_2Y_{lm}(x) + {}_{-2}Y_{lm}(x))/2 \\
X_{lm}(x) &= (-i)({}_2Y_{lm}(x) - {}_{-2}Y_{lm}(x))/2
\end{aligned}
$$

Results and plan

- Full sky
- Masked
- Dust
- Dust + Synchrotron
  - has an offset
  - r ~> 0.01 OK with this simple cleaning method
- Improve foreground removal
  - Though we think it is not a time to build complicated analysis techniques
- Add experimental details