

Fast Vertexing update: new class and minbias results

update Dec.2025

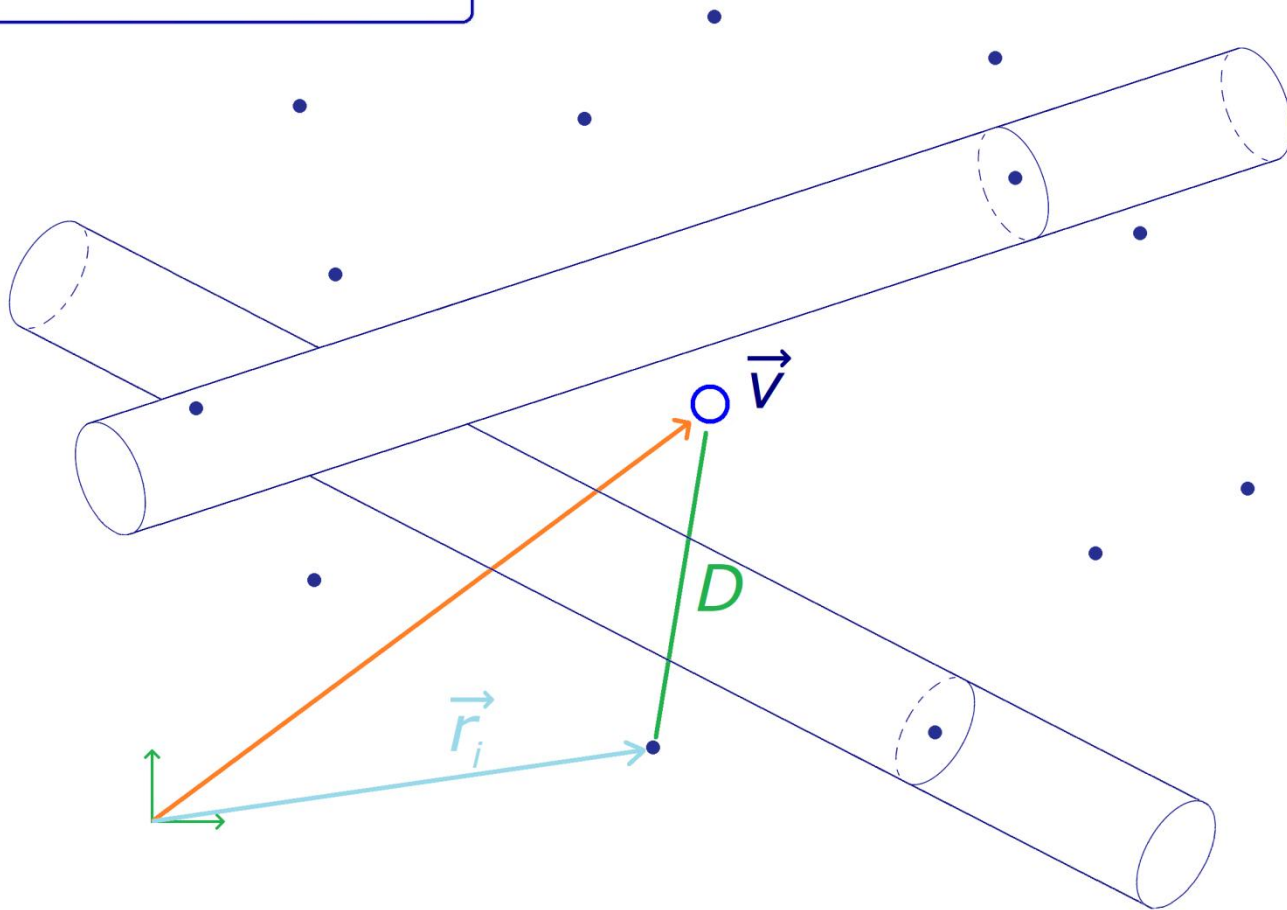
C++ library NXV4

- 1 *Vertexing - short reminder*
- 2 *What is new: code **overhaul***
- 3 ***MinBias** - results*
- 4 *Conclusion*

- 1 *Vertexing - short reminder*
- 2 *What is new: code **overhaul***
- 3 ***MinBias** - results*
- 4 *Conclusion*

Short Reminder

$$|v\rangle = \left(\overline{\sigma_D^{-2}} \right)^{-1} \overline{\sigma_D^{-2} |r_i\rangle}$$



- 1 *Vertexing - short reminder*
- 2 *What is new: code **overhaul***
- 3 ***MinBias** - results*
- 4 *Conclusion*

Steps - reminder

- *data* - $DOCA_{xyz}$ and p_{xyz}
- *initial selection* - calculates the first value of the vertex
- *vertex loop* - calculates the vertex again by adding tighter conditions
- *loop stop condition* = less than 3 tracks or 10 iterations

Last talk - conditions

Initial selection

- $DOCA_{xy} < 3mm$
- $DOCA_z < 10mm$
- $pT > 0.5 \text{ GeV}/c$

Iterative selection

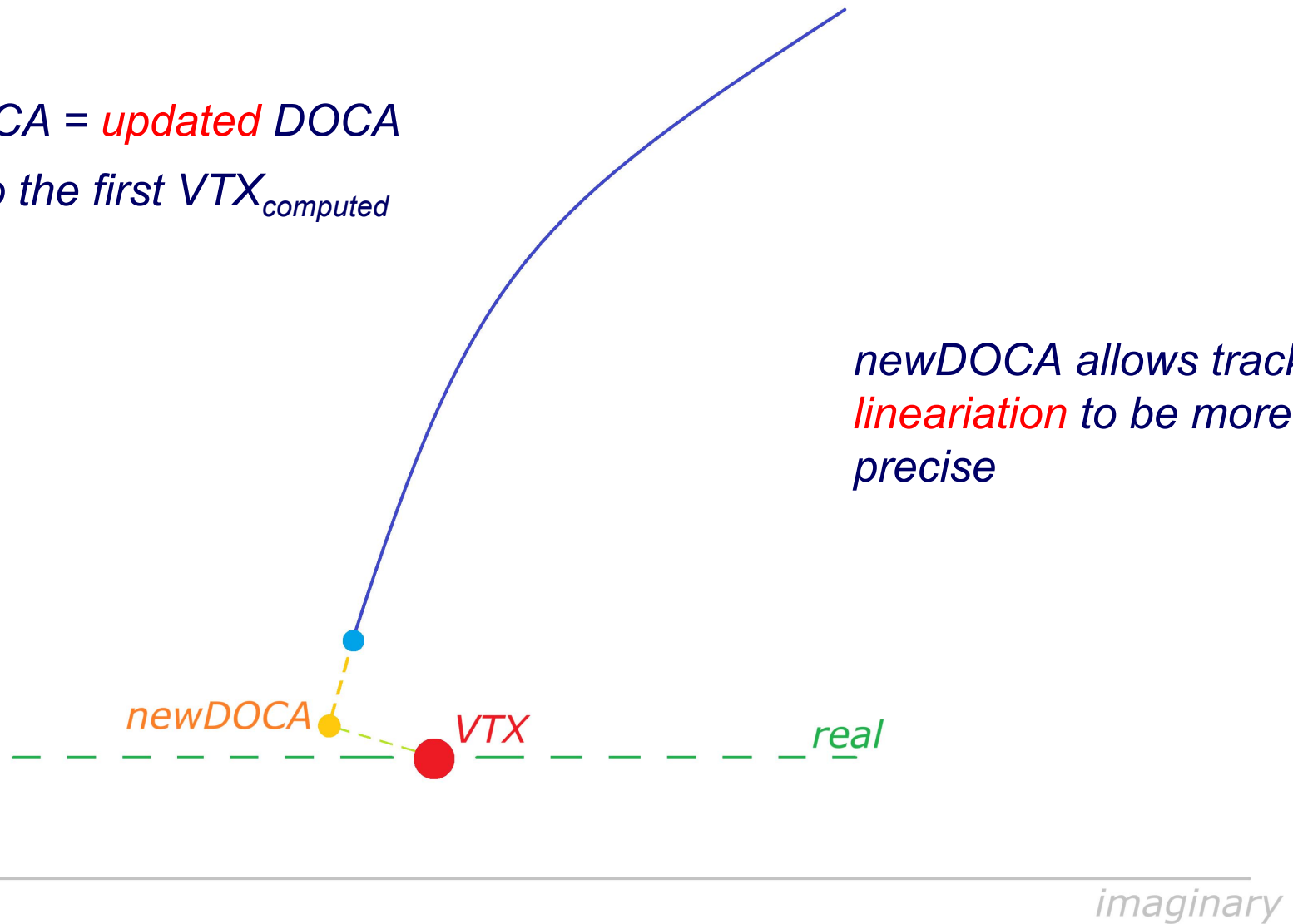
- $DOCA_{xy} \neq 0.3mm$
- $DOCA_z \neq 1.0mm$
- $pT \neq 0.05 \text{ GeV}/c$

Updates:

- now a *class* - more *modular*
 - more *functional*
- *newDOCA* method
- eliminate the *worst track*
 - still working on conditions
- selection cuts: in sigmas, not absolute val.; χ^2/dof ; no. hits

- newDOCA = *updated* DOCA
closer to the first $VTX_{computed}$

newDOCA allows track
linearisation to be more
precise



Vertexing Class

Fast vertexing code $\Rightarrow$ is now a *class* of its own

- selection of tracks

```
void vxi::sel(double lxy, double lzz, double lpt)
```

- worst track

```
void vxi::wst()
```

- vertexing

```
void vxi::vxx(const std::string FX)
```

- newDOCA

```
if (FX.find("newDOCA") != std::string::npos) newDOCA = true;
```

- track summaries are a class

```
class trk
```

```
trk (int wPDG, int wNpt, double wCX,  
     int wQQ, double wRR,  
     double wxr, double wyr, double wzr,  
     double wpx, double wpy, double wpz,  
     double wtx, double wty, double wtz ) :
```

Selection of tracks

```
void vxi::sel(double lxy, double lzz, double lpt)
```

1	<u>Nt_{new}</u>
2	
7	
4	
6	
5	
3	

- next iterations - till Nt_{new}

- *faster !*

- 1 *Vertexing - short reminder*
- 2 *What is new: code **overhaul***
- 3 ***MinBias - results***
- 4 *Conclusion*

```

// instantiate
//
//      tracks      Nlim       $\chi^2/\text{dof}$  lim       $\sigma_{xy}$  lim       $\sigma_z$  lim      pt min
vxi VXi(NTT, TRK, eno, ecx, exy, ezz, ept) ;

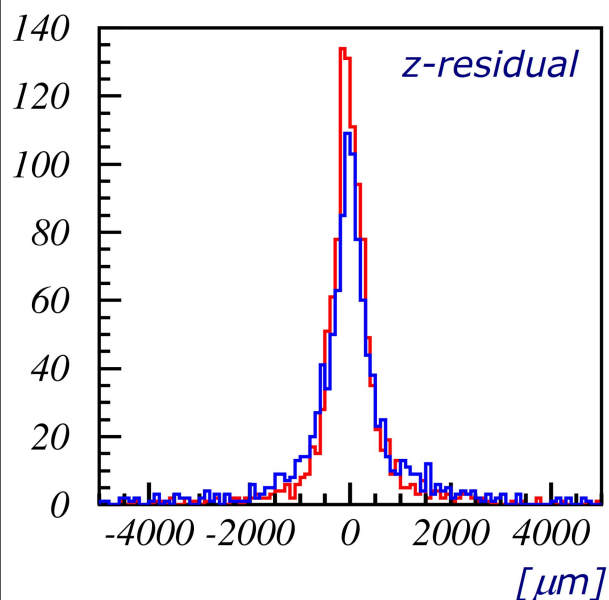
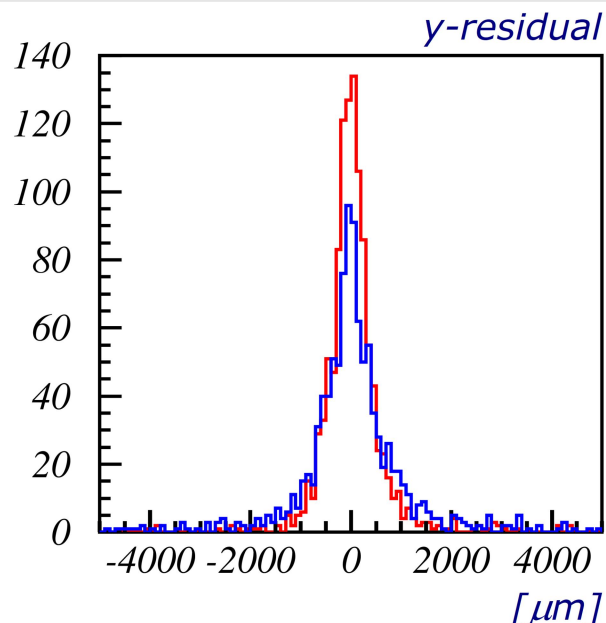
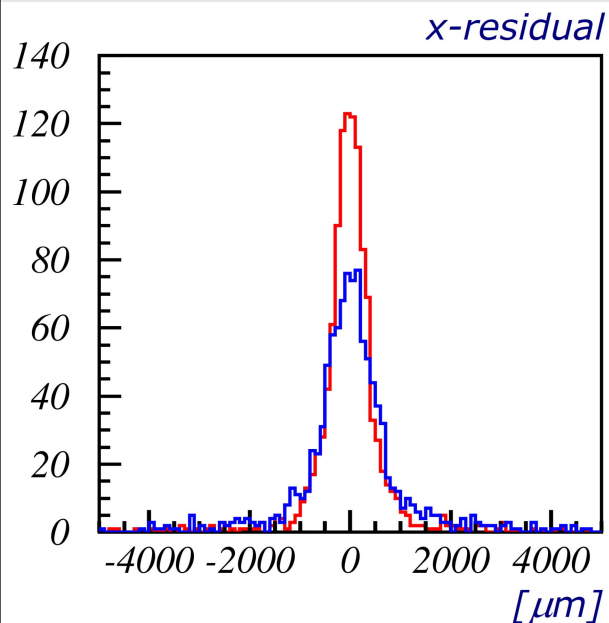
// first estimate
//
VXi.vxx("nothing") ;

// update DOCA's
//
VXi.vxx("newDOCA") ;

// select
//
VXi.sel(0.5*exy, 0.3*ezz, 1.2*ept) ;

// straight VTX
//

```



Initial selection

$ecx = 5.00; eno = 3$

$exy = 9.00; // \text{sigma}$

$ezz = 9.00; // \text{sigma}$

$ept = 0.05$

Next selection - 2 times:

- *newDOCA*
- $0.5*exy, 0.3*ezz, 1.2*ept$
- *straight VTX*

MinBias - SPDroot data

- Vertexing*
 - *Updates*
 - now a *class*
 - more *modular*
 - more *functional*
 - *newDOCA* method
 - eliminate the *worst track* - working on cond.
 - selection cuts: in sigmas, not absolute val.
- *Results*
 - very good resolution
 - CPU likely faster than pervious (110 ns/trk)
- In preparation - Ideas*
 - vertex with *curved* tracks
 - allows use of *lower pt* tracks also

MinBias - SPDroot data

International Journal of Modern Physics E
Vol. 33, No. 11 (2024) 2441013 (8 pages)

DOI: [10.1142/S0218301324410131](https://doi.org/10.1142/S0218301324410131)

Track vertexing for trigger applications

Maria Dima ^{*,‡,§}, Mihai-Tiberiu Dima ^{*,†} and Madalina Mihailescu[‡]

**Joint Institute for Nuclear Research,
Dzhelepov Laboratory of Nuclear Problems (DLNP),
141980, Dubna-Russia*

Good afternoon!

Your paper " *Fast Vertexing Code* " has passed the peer review stage, and the conclusion has been reached that it is recommended for publication in the journal "Physics of Elementary Particles and Atomic Nuclei" (PEPAN).

Once all articles have been collected and submitted to the journal's editorial board, we will send you a notification.

Спасибо !