ITTF Vertex Finder Performance

Jonathan Gans

1) Standard v. ITTF Vertex Finder

2) Au+Au DATAVertex Finding Comparison

   ITTF Tracking and Vertex Finding v. Standard Tracking and Vertex Finding

3) pp Embedding Vertex Finding Comparision

   ITTF pp Vertex Finding v. Standard Tracking and Vertex Finding

4) The future

Standard algorithm For Gold

Iterative procedure:

- Find z seed using global tracks
- First pass with loose dca cut in xy plane (1cm) and 3d dca cut (10 cm).
- Second pass using tighter 3D dca cut (2 cm)
- Final pass using tighter 3D dca cut (0.5 cm)

Resolution:

$\sigma(x) = \sigma(y) = 0.60\ \text{cm}/\sqrt{N}$

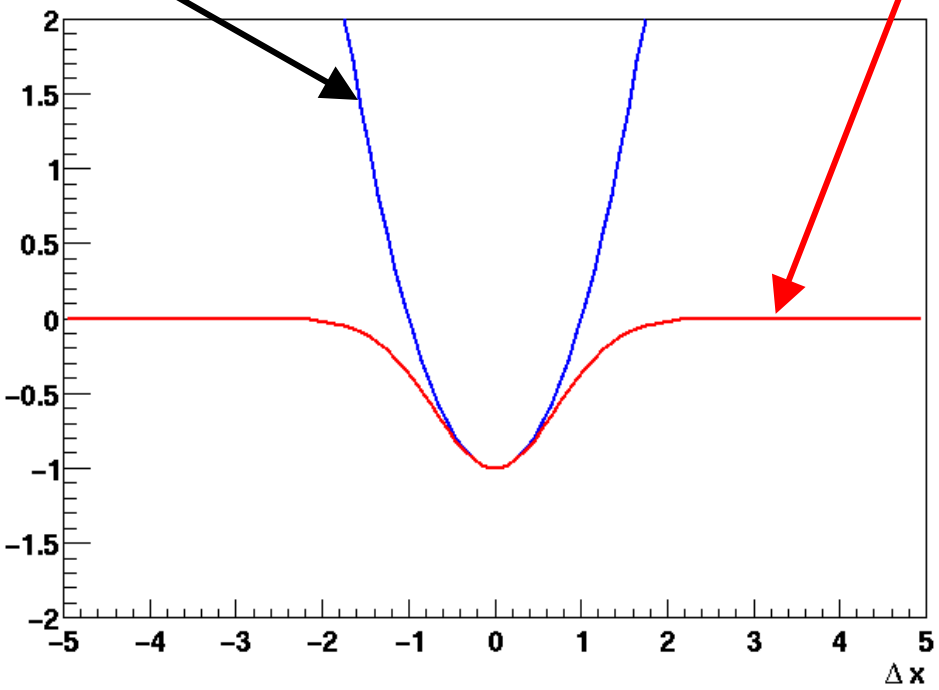
$\sigma(z) = 0.85\ \text{cm}/\sqrt{N}$

ITTF Vertex Finder

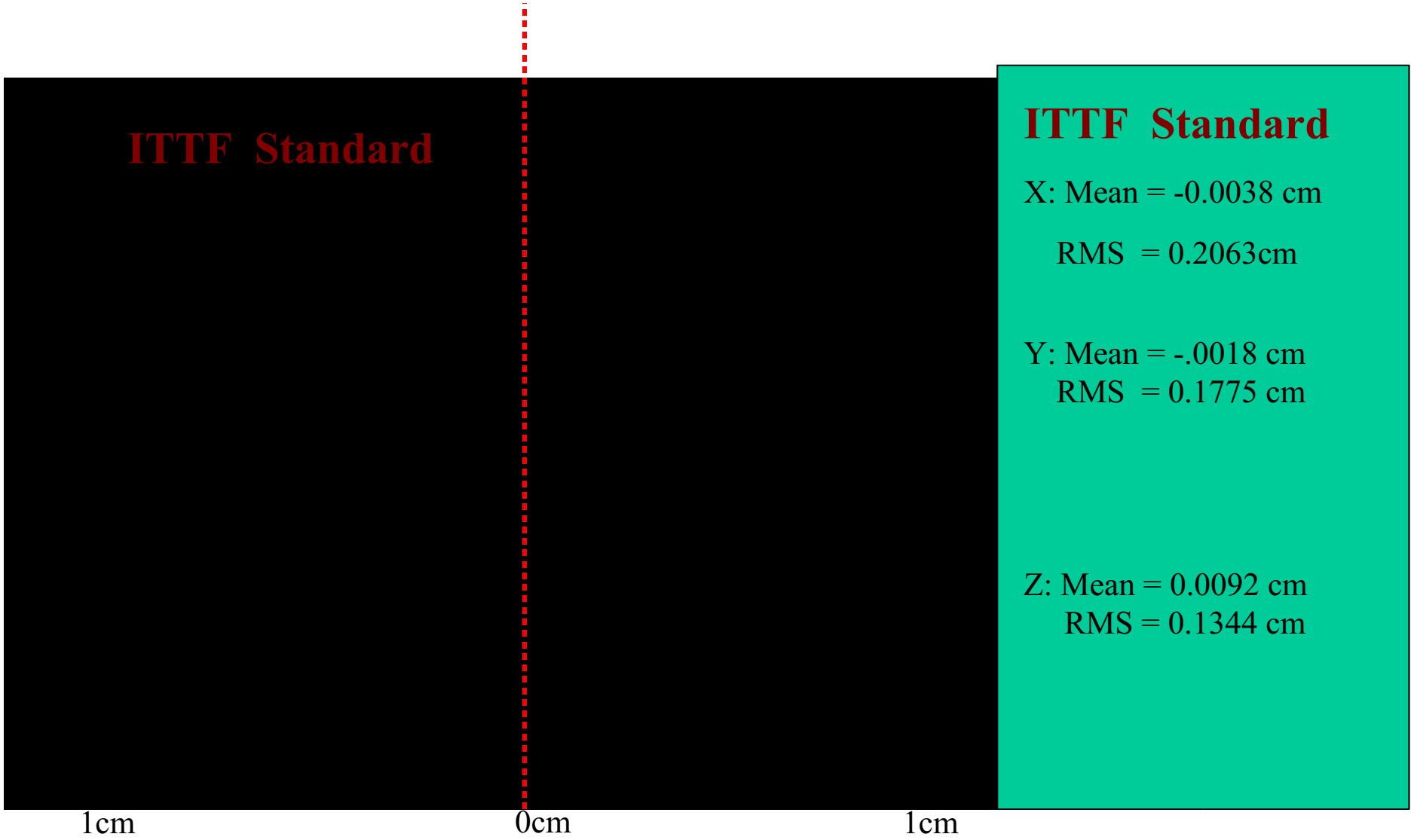
No outlier removal!

Replace $\chi 2$ function with robust fitting potential:

$$\chi^2 : L \propto \frac{1}{N}\sum_{i=1}^{N}\frac{(\Delta x_i)^2}{\sigma_i^2} \quad \Rightarrow \quad \text{Robust} : L \propto \prod_{i=1}^{N}\exp\left(\frac{-(\Delta x_i)^2}{2\sigma_i^2}\right)$$



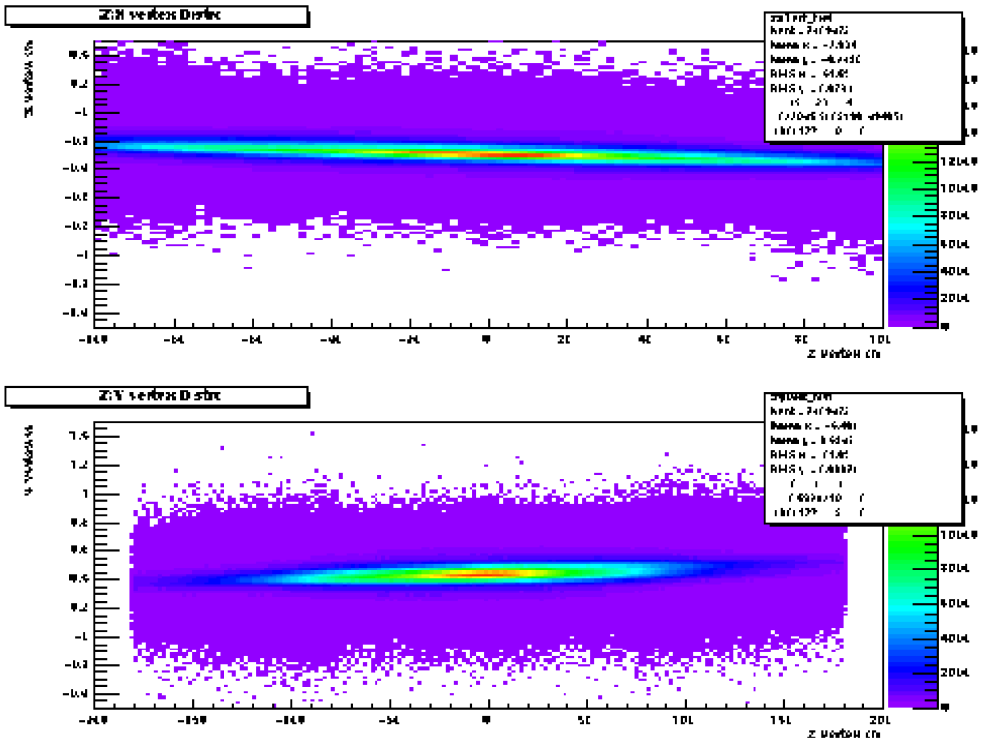
Au+Au DATAVertex Finding Comparison
ITTF Tracking/VF v. Standard Tracks/VF



Vertex Finding in Proton + Proton

For ITTF Finder, algorithm is different

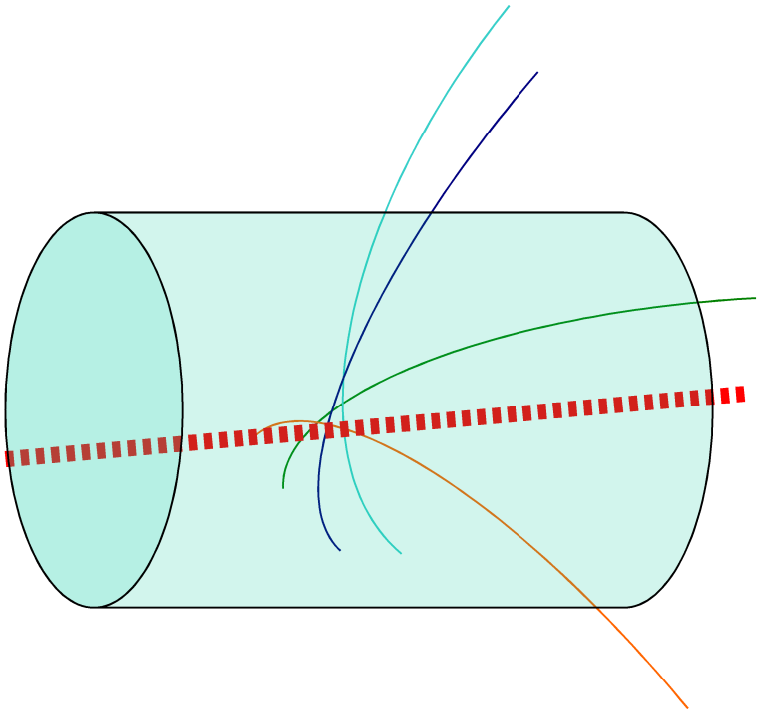
Only Fit in One Dimension (z)



$x(z) = -0.304233\ \text{cm} + -0.00065732 * z$

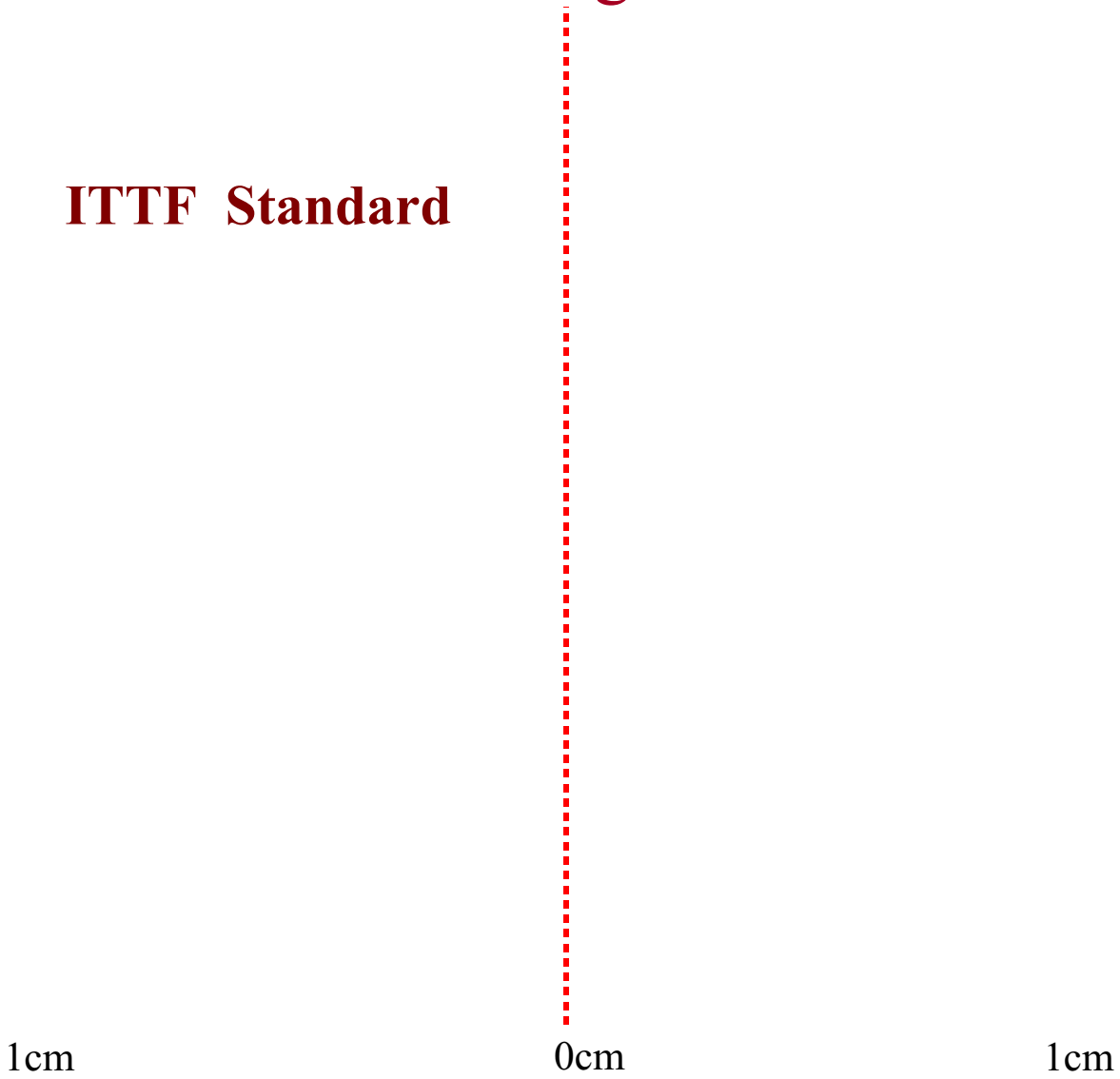
$y(z) = 0.407226\ \text{cm} + 0.00012396 * z$

For Seed Require CTB Match



PP DATAVertex Finding Comparison
ITTF Tracking/VF v. Standard Tracks/VF

ITTF Standard



ITTF Standard

X: Mean = ??? cm

RMS = ?? cm

Y: Mean = ?? cm

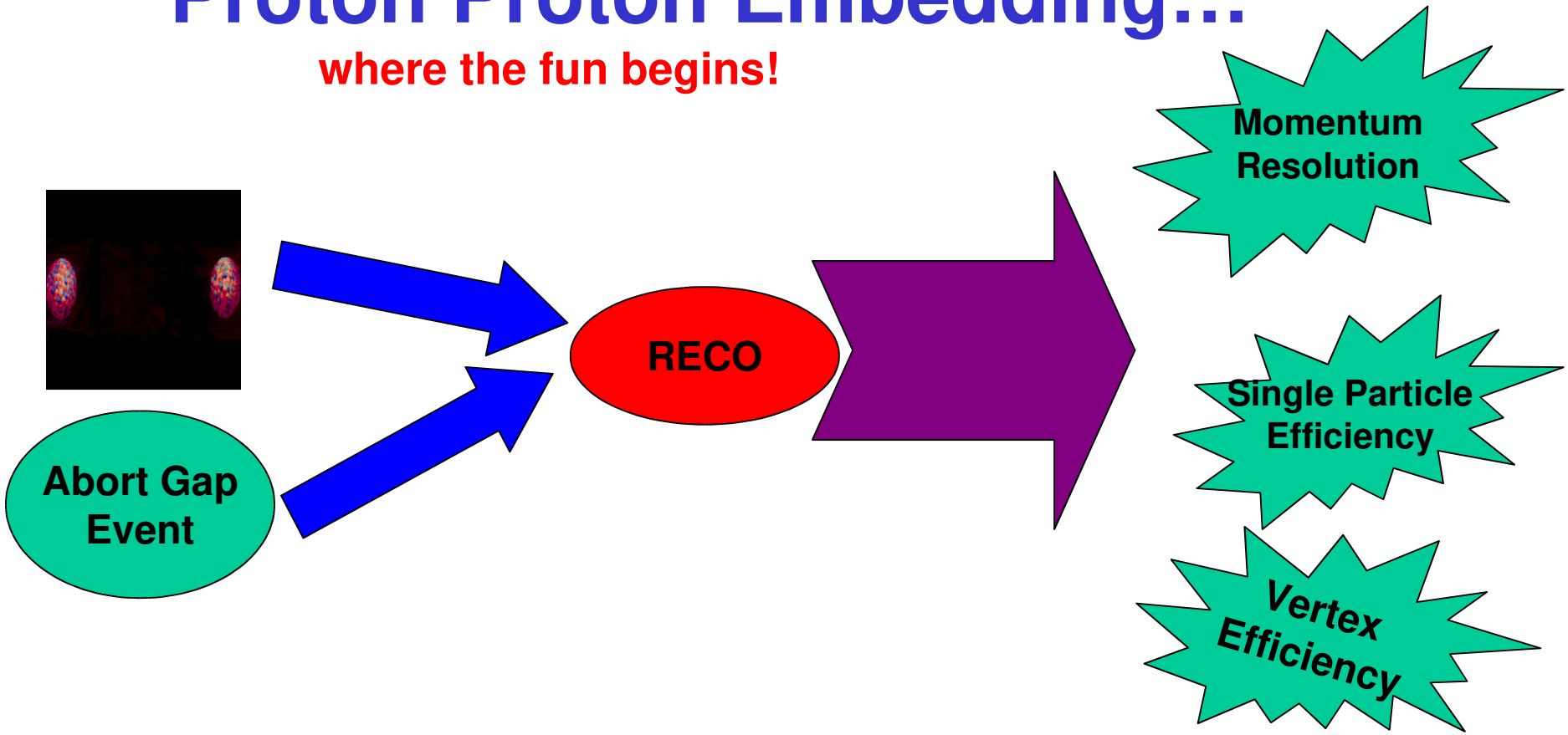
RMS = ?? cm

Z: Mean = ?? cm

RMS = ?? cm

Proton Proton Embedding…

where the fun begins!



A lot of hard work!!

Special Thanks to:

Eric Hjort, Herb Ward, Pablo Yepes, Jan Balewski, Bernd Surrow

Embedded Standard Tracks: ITTF VF v. Standard VF

Resolution and Efficiency

1985 Embedded Events

ppLMV

ppLMV = 74% Efficient

ITTF

ITTF = 73% Efficient

Efficiency Proportional to Multiplicty

Embedded Standard Tracks: ITTF VF v. Standard VF

Resolution and Efficiency

Future and Conclusion

Gold: Need to test efficiency with embedding

Proton: Performs on par with ppLMV
Maybe can be tweaked to outperform ppLMV

Multiple Vertices due to pile-up?