

TreeSearch Track Reconstruction for GEMs

Ole Hansen

Jefferson Lab

SBS Collaboration Meeting
19 March 2010

Motivation

- Existing tracking simulation very basic
- January 2010 Technical Review:
“We recommend to provide a full simulation of the tracking algorithm”
- ⇒ Realistically simulate occupancy of chambers (esp. front trackers)
- ⇒ Demonstrate efficient track reconstruction for high-occupancy conditions in GEMs

Motivation

- Existing tracking simulation very basic
- January 2010 Technical Review:
“We recommend to provide a full simulation of the tracking algorithm”
- ⇒ Realistically simulate occupancy of chambers (esp. front trackers)
- ⇒ Demonstrate efficient track reconstruction for high-occupancy conditions in GEMs

Motivation

- Existing tracking simulation very basic
- January 2010 Technical Review:
“We recommend to provide a full simulation of the tracking algorithm”
- ⇒ Realistically simulate occupancy of chambers (esp. front trackers)
- ⇒ Demonstrate efficient track reconstruction for high-occupancy conditions in GEMs

Motivation

- Existing tracking simulation very basic
- January 2010 Technical Review:
“We recommend to provide a full simulation of the tracking algorithm”
- ⇒ Realistically simulate occupancy of chambers (esp. front trackers)
- ⇒ Demonstrate efficient track reconstruction for high-occupancy conditions in GEMs

Tree Search Algorithm

- Suggested by Dell'orso *et al.*, NIM A **287**, 436 (1990)
- Recursive template matching
- Fast and efficient (speed and memory)
- Proven at HERMES with various drift chambers
- Used by Qweak (based on HERMES code) for HDCs and VDCs
- Well-debugged code exists for Hall A BigBite MWDCs
- Appears suitable for SBS front tracking system

Tree Search Algorithm

- Suggested by Dell'orso *et al.*, NIM A **287**, 436 (1990)
- **Recursive** template matching
- Fast and efficient (speed and memory)
- Proven at HERMES with various drift chambers
- Used by Qweak (based on HERMES code) for HDCs and VDCs
- Well-debugged code exists for Hall A BigBite MWDCs
- Appears suitable for SBS front tracking system

Tree Search Algorithm

- Suggested by Dell'orso *et al.*, NIM A **287**, 436 (1990)
- **Recursive** template matching
- Fast and efficient (speed and memory)
- Proven at HERMES with various drift chambers
- Used by Qweak (based on HERMES code) for HDCs and VDCs
- Well-debugged code exists for Hall A BigBite MWDCs
- Appears suitable for SBS front tracking system

Tree Search Algorithm

- Suggested by Dell'orso *et al.*, NIM A **287**, 436 (1990)
- **Recursive** template matching
- Fast and efficient (speed and memory)
- Proven at HERMES with various drift chambers
- Used by Qweak (based on HERMES code) for HDCs and VDCs
- Well-debugged code exists for Hall A BigBite MWDCs
- Appears suitable for SBS front tracking system

Tree Search Algorithm

- Suggested by Dell'orso *et al.*, NIM A **287**, 436 (1990)
- **Recursive** template matching
- Fast and efficient (speed and memory)
- Proven at HERMES with various drift chambers
- Used by Qweak (based on HERMES code) for HDCs and VDCs
- Well-debugged code exists for Hall A BigBite MWDCs
- Appears suitable for SBS front tracking system

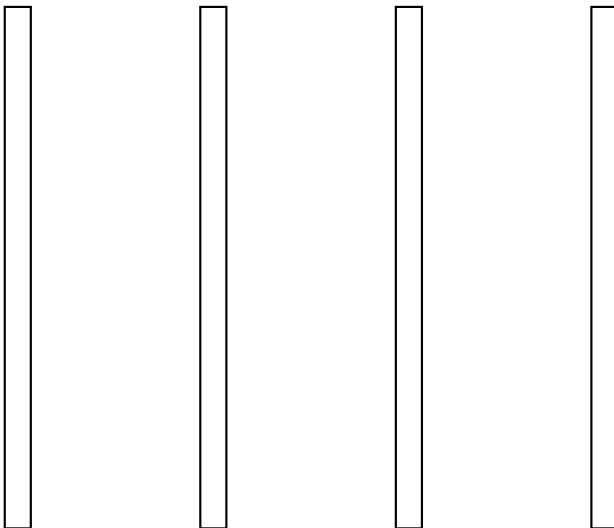
Tree Search Algorithm

- Suggested by Dell'orso *et al.*, NIM A **287**, 436 (1990)
- **Recursive** template matching
- Fast and efficient (speed and memory)
- Proven at HERMES with various drift chambers
- Used by Qweak (based on HERMES code) for HDCs and VDCs
- Well-debugged code exists for Hall A BigBite MWDCs
- Appears suitable for SBS front tracking system

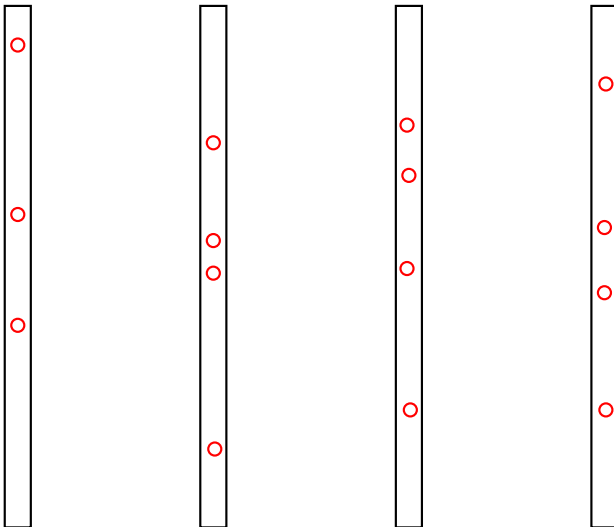
Tree Search Algorithm

- Suggested by Dell'orso *et al.*, NIM A **287**, 436 (1990)
- **Recursive** template matching
- Fast and efficient (speed and memory)
- Proven at HERMES with various drift chambers
- Used by Qweak (based on HERMES code) for HDCs and VDCs
- Well-debugged code exists for Hall A BigBite MWDCs
- Appears suitable for SBS front tracking system

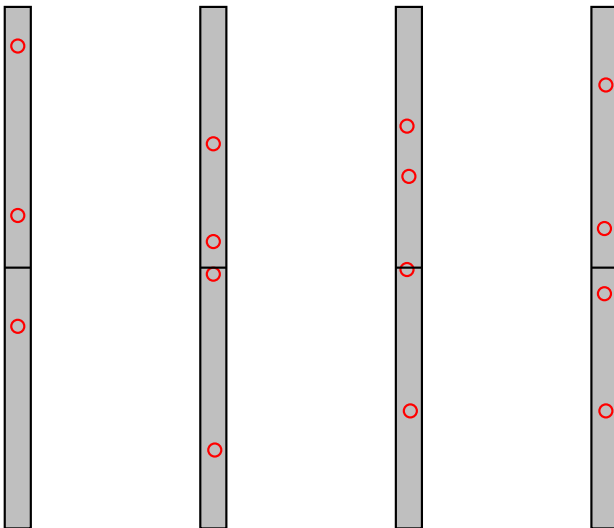
Successive Approximation Method



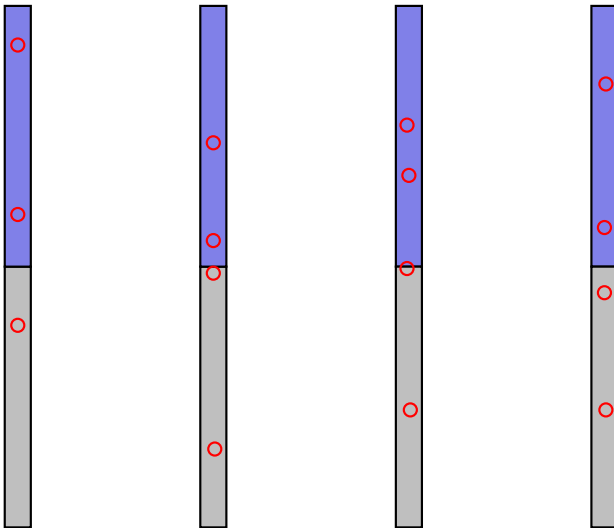
Successive Approximation Method



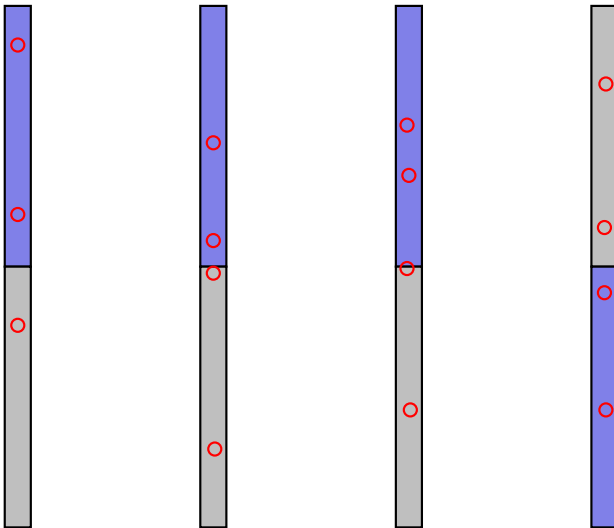
Successive Approximation Method



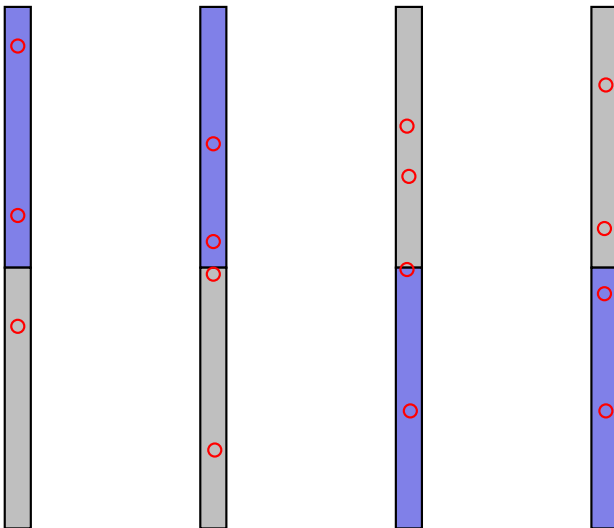
Successive Approximation Method



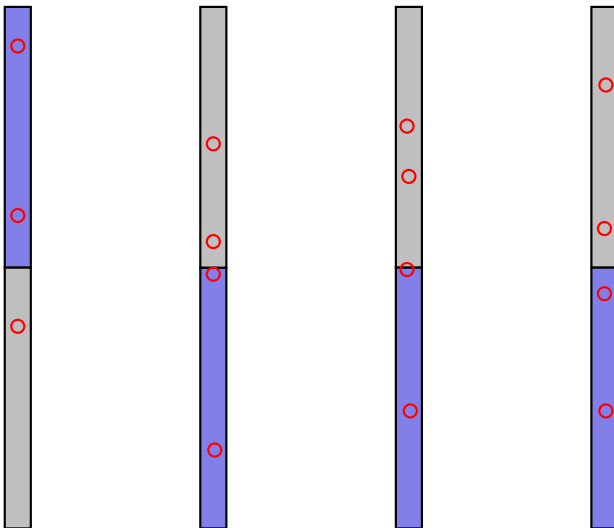
Successive Approximation Method



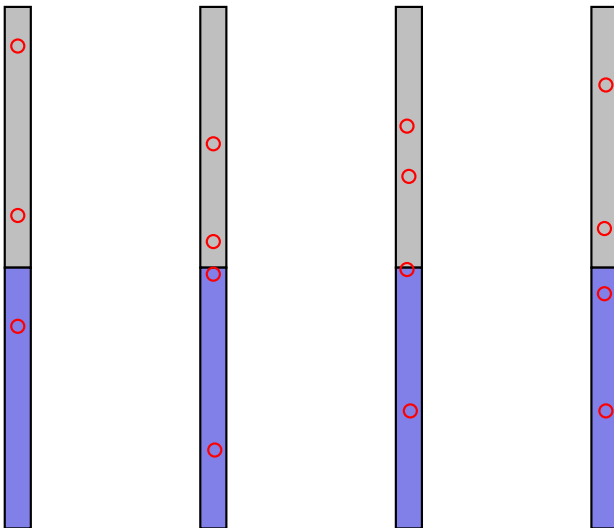
Successive Approximation Method



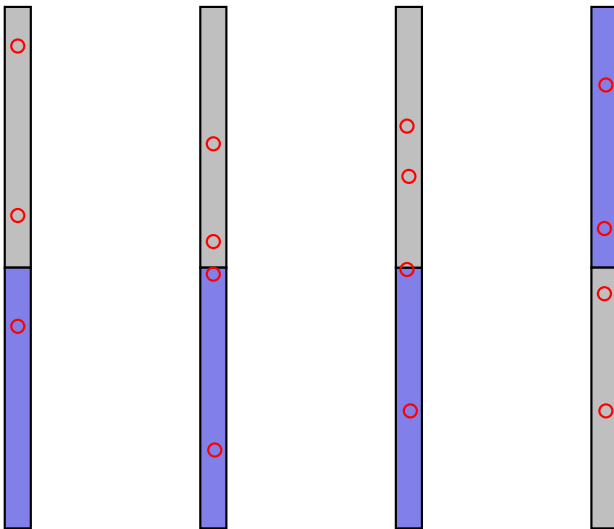
Successive Approximation Method



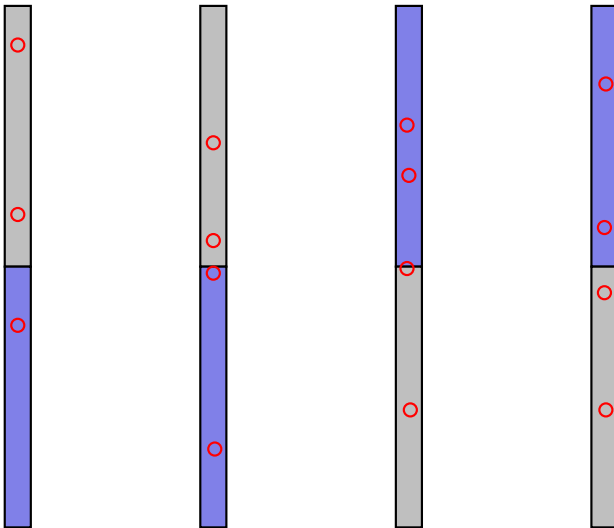
Successive Approximation Method



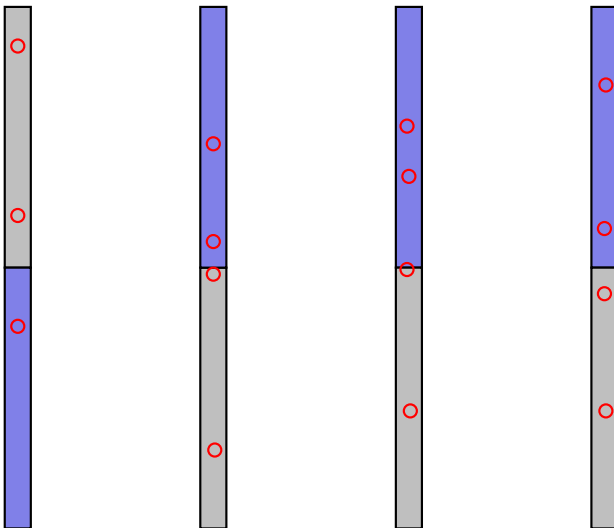
Successive Approximation Method



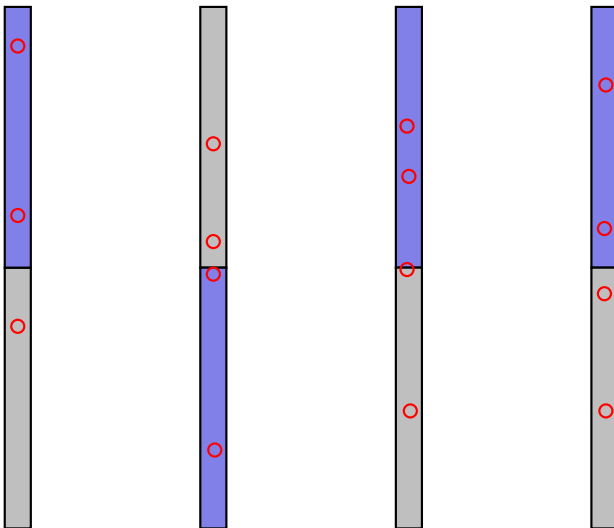
Successive Approximation Method



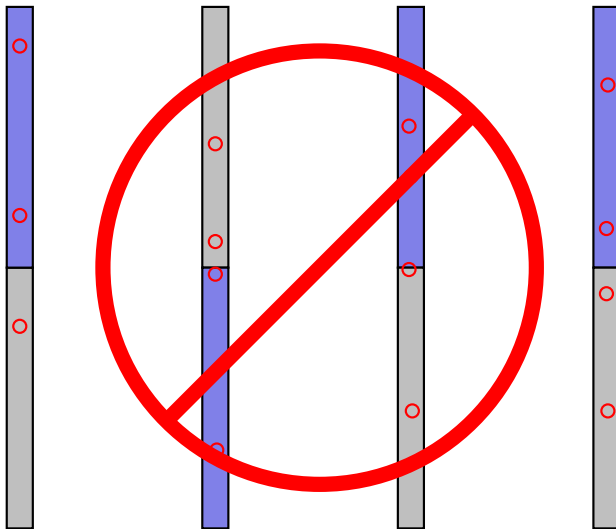
Successive Approximation Method



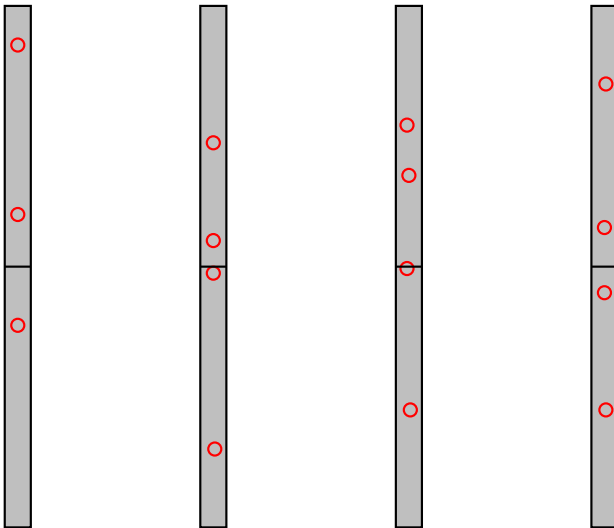
Successive Approximation Method



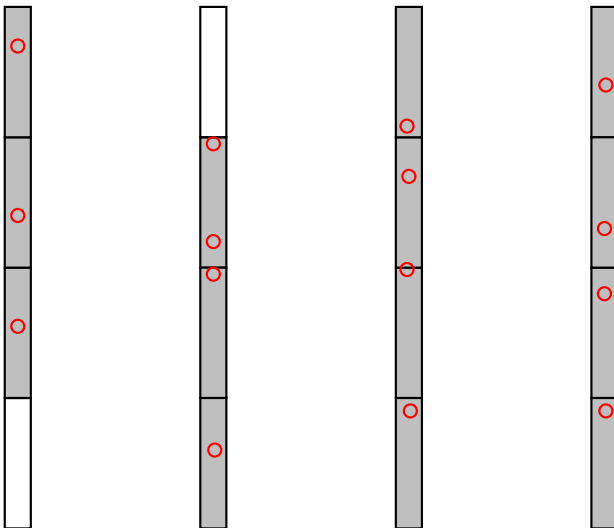
Successive Approximation Method



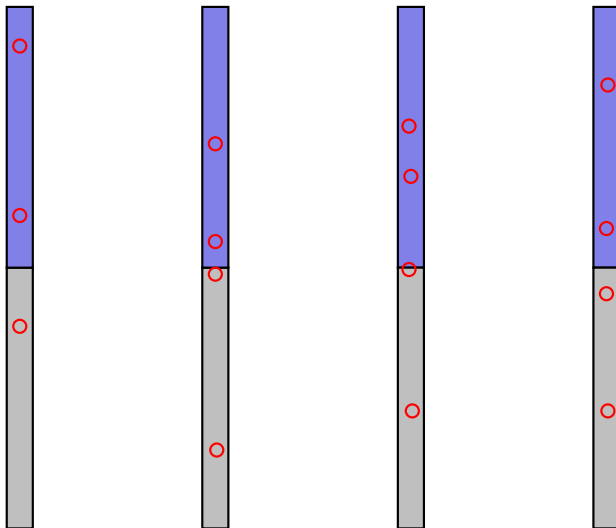
Successive Approximation Method



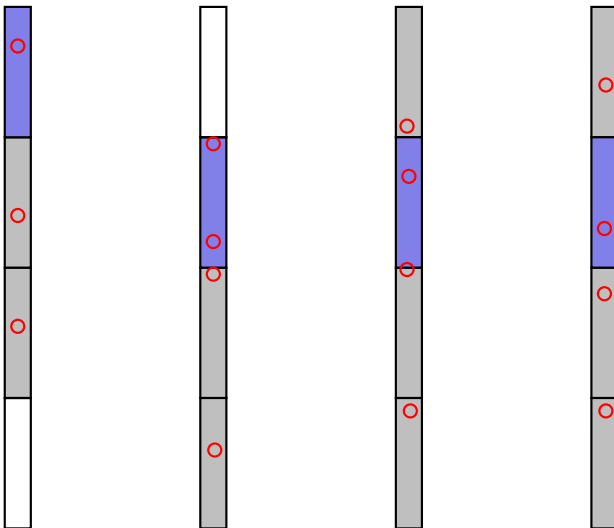
Successive Approximation Method



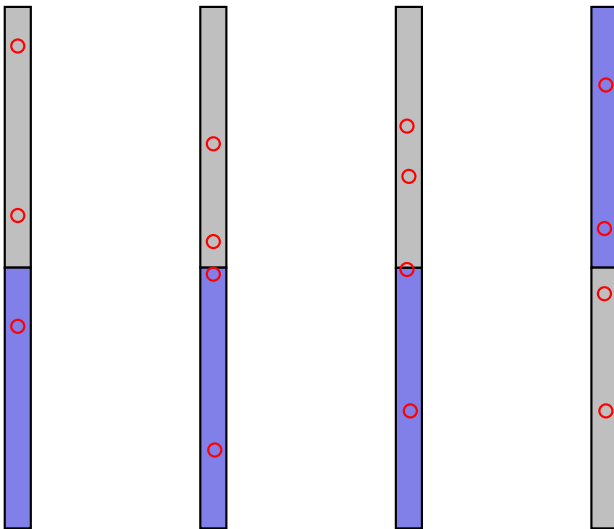
Successive Approximation Method



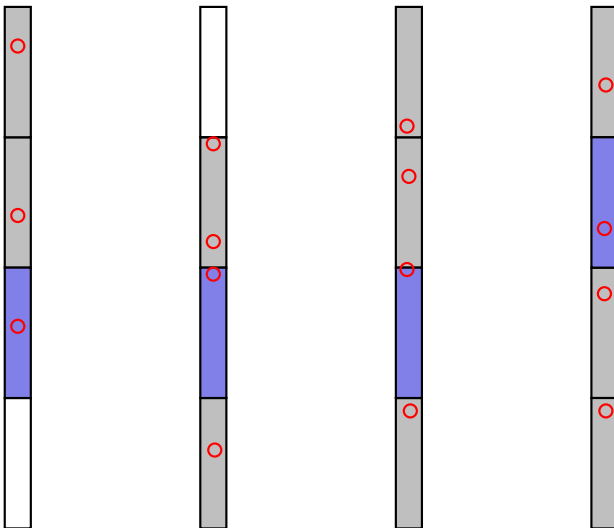
Successive Approximation Method



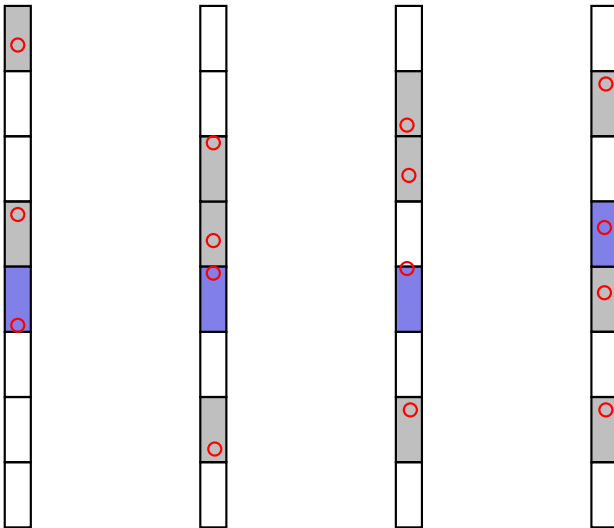
Successive Approximation Method



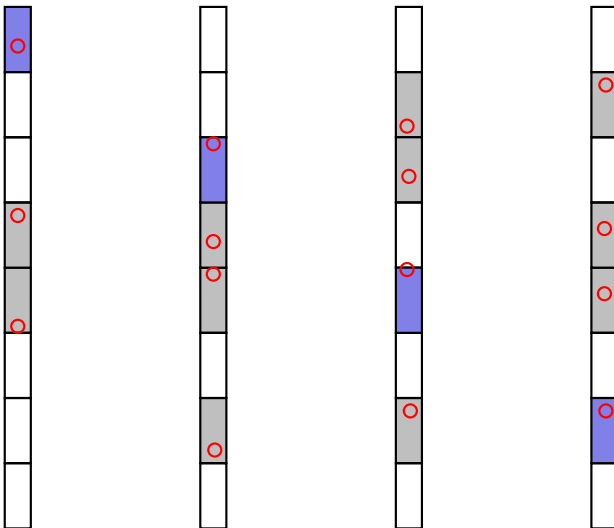
Successive Approximation Method



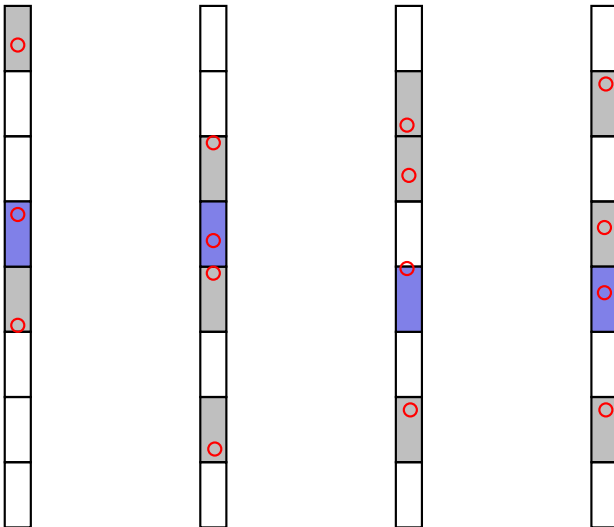
Successive Approximation Method



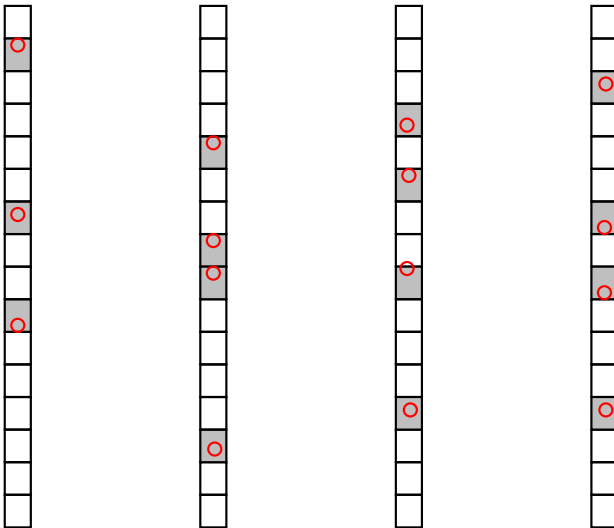
Successive Approximation Method



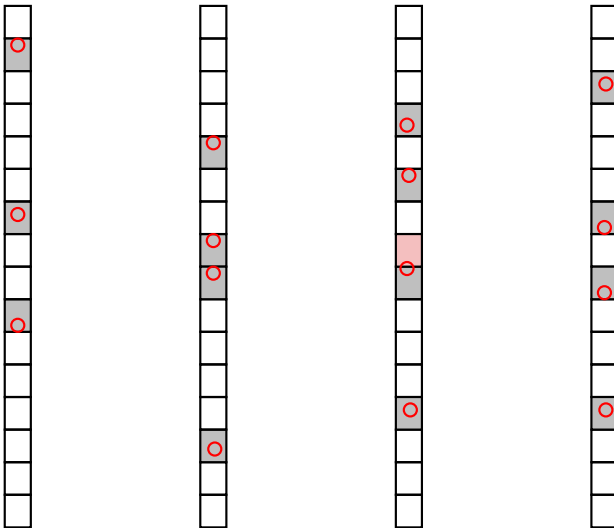
Successive Approximation Method



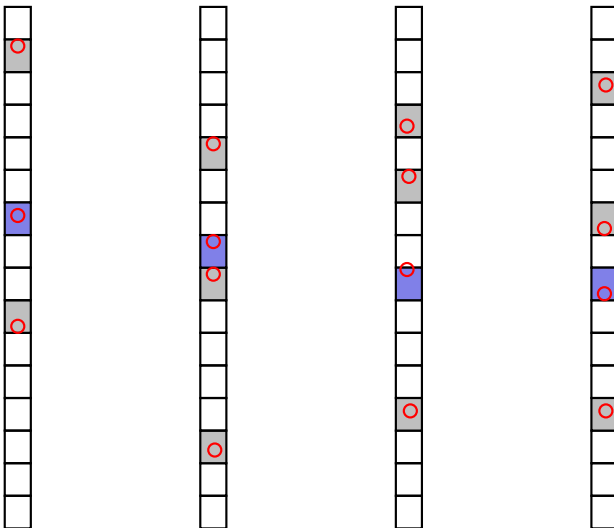
Successive Approximation Method



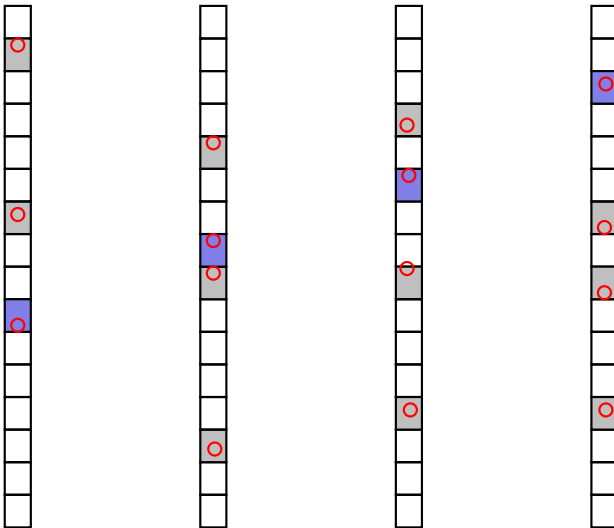
Successive Approximation Method



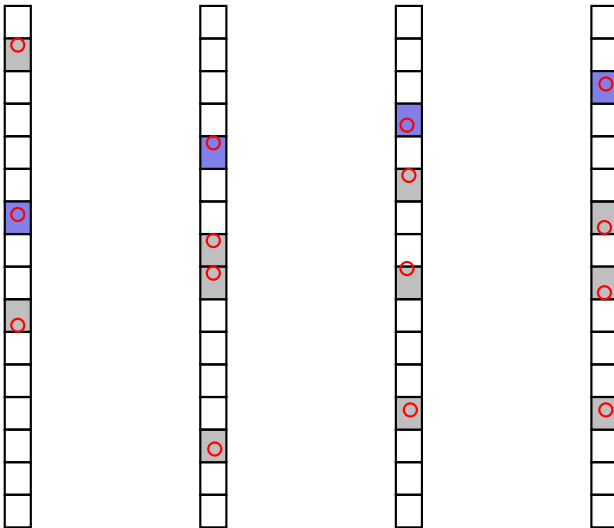
Successive Approximation Method



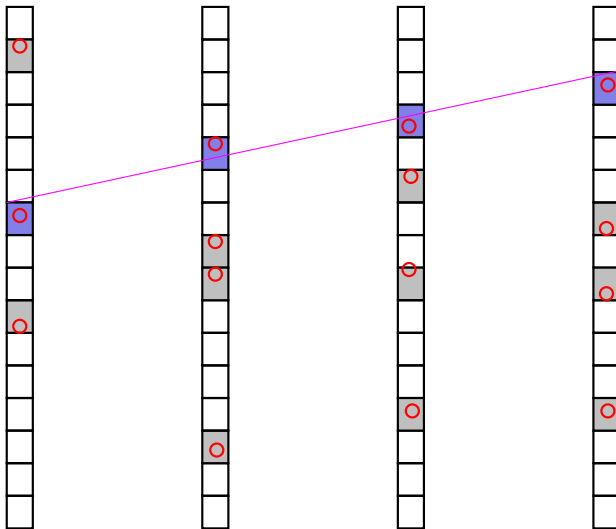
Successive Approximation Method



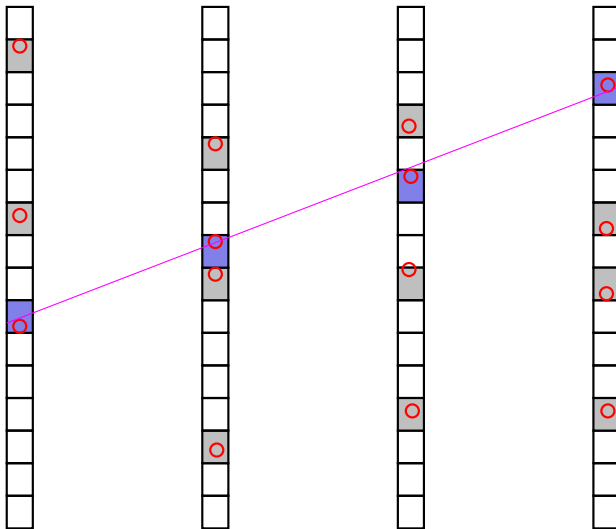
Successive Approximation Method



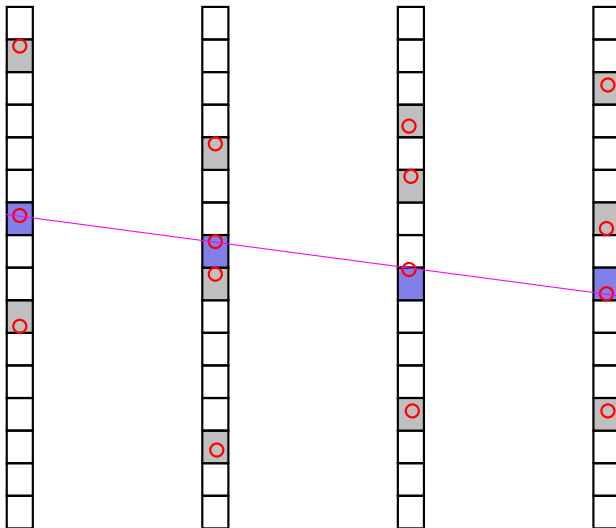
Successive Approximation Method



Successive Approximation Method



Successive Approximation Method



Key Advantages

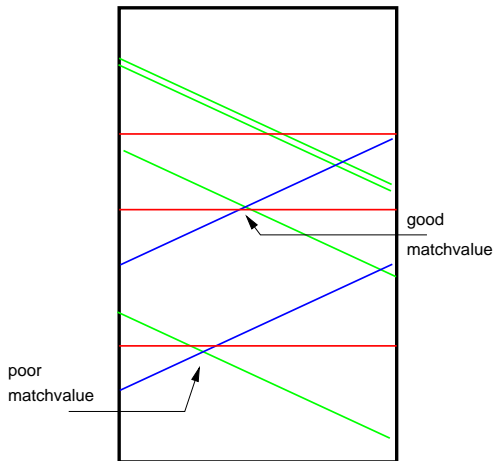
- Tree-like search process allows fast template **lookup**: $\mathcal{O}(\log N_{\text{bins}})$
- Symmetry and self-referential properties of the templates allow efficient storage. Only “base patterns” need to be stored: $\mathcal{O}(1\text{MB})$.

Key Advantages

- Tree-like search process allows fast template **lookup**: $\mathcal{O}(\log N_{\text{bins}})$
- Symmetry and self-referential properties of the templates allow efficient **storage**. Only “base patterns” need to be stored: $\mathcal{O}(1\text{MB})$.

3D Matching

Chamber front view



Repeat for each chamber group along z

3D Track Fitting

Fit the linear equations

$$\mathbf{A}\hat{\beta} = \mathbf{y}$$

using the coordinates y_i of the best 2D fits in all planes i ,

$$y_i = \begin{pmatrix} x + m_x z_i \\ y + m_y z_i \end{pmatrix} \cdot \begin{pmatrix} \cos \alpha_i \\ \sin \alpha_i \end{pmatrix},$$

where x , m_x , y , m_y are the parameters to be fitted (β_k).

The fit is done by Cholesky decomposition of the normal equation

$$(\mathbf{A}^T \mathbf{W} \mathbf{A}) \hat{\beta} = (\mathbf{A}^T \mathbf{W}) \mathbf{y},$$

where \mathbf{W} is the weight matrix (cf. ROOT's `TLinearFitter`).

NB: Each plane can be at arbitrary z

Requirements

- **Straight tracks**
 - 3+ planes per projection (coordinate)
 - Normally: 3+ projections (but see discussion later)

Requirements

- Straight tracks
- 3+ planes per projection (coordinate)
- Normally: 3+ projections (but see discussion later)

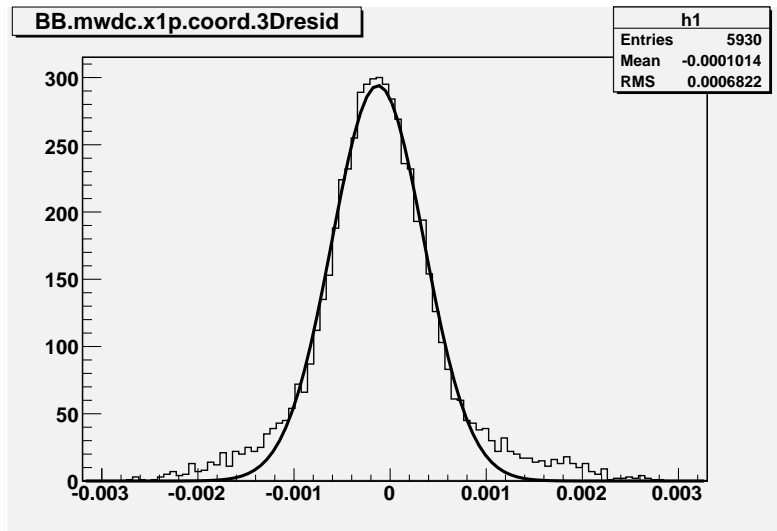
Requirements

- Straight tracks
- 3+ planes per projection (coordinate)
- Normally: 3+ projections (but see discussion later)

BigBite MWDC Code

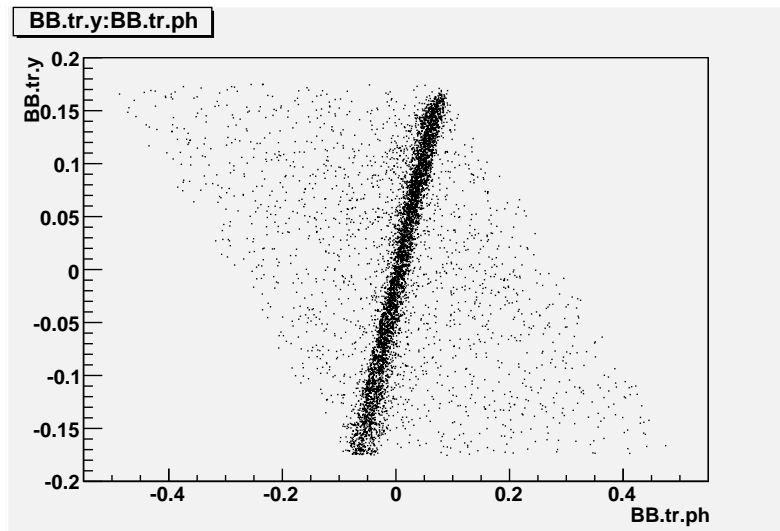
- Written in 2008 for E04-007 & Transversity
- Used in production replay
- Well tested and debugged by now
- Parallelized
- Event display available
- Very good analysis speed

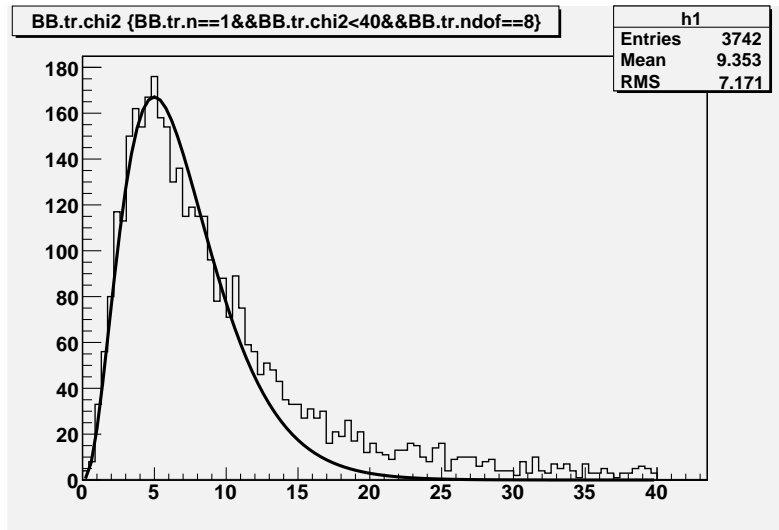
Residuals (E04-007 online results!)



Track y vs phi (E04-007 online results!)

NB: approx. point target



Track χ^2 (E04-007 online results!)

Required Modifications to BigBite Code for GEMs

Analyzing GEM trackers is easier:

- No L-R ambiguity
- Additional correlations

Required software changes:

- Add weighted averaging of strip signals
- Disable/remove handling of L-R ambiguities
- Exploit additional information from GEMs for 3D matching:
 - ▶ Amplitude correlation
 - ▶ Time correlation

First two items already finished for PREX GEM analysis

Required Modifications to BigBite Code for GEMs

Analyzing GEM trackers is easier:

- No L-R ambiguity
- Additional correlations

Required software changes:

- Add weighted averaging of strip signals
- Disable/remove handling of L-R ambiguities
- Exploit additional information from GEMs for 3D matching:
 - ▶ Amplitude correlation
 - ▶ Time correlation

First two items already finished for PREX GEM analysis

Required Modifications to BigBite Code for GEMs

Analyzing GEM trackers is easier:

- No L-R ambiguity
- Additional correlations

Required software changes:

- Add weighted averaging of strip signals
- Disable/remove handling of L-R ambiguities
- Exploit additional information from GEMs for 3D matching:
 - ▶ Amplitude correlation
 - ▶ Time correlation

First two items already finished for PREX GEM analysis

Required Modifications to BigBite Code for GEMs

Analyzing GEM trackers is easier:

- No L-R ambiguity
- Additional correlations

Required software changes:

- Add weighted averaging of strip signals
- Disable/remove handling of L-R ambiguities
- Exploit additional information from GEMs for 3D matching:
 - ▶ Amplitude correlation
 - ▶ Time correlation

First two items already finished for PREX GEM analysis

Required Modifications to BigBite Code for GEMs

Analyzing GEM trackers is easier:

- No L-R ambiguity
- Additional correlations

Required software changes:

- Add weighted averaging of strip signals
- Disable/remove handling of L-R ambiguities
- Exploit additional information from GEMs for 3D matching:
 - ▶ Amplitude correlation
 - ▶ Time correlation

First two items already finished for PREX GEM analysis

Required Modifications to BigBite Code for GEMs

Analyzing GEM trackers is easier:

- No L-R ambiguity
- Additional correlations

Required software changes:

- Add weighted averaging of strip signals
- Disable/remove handling of L-R ambiguities
- Exploit additional information from GEMs for 3D matching:
 - ▶ Amplitude correlation
 - ▶ Time correlation

First two items already finished for PREX GEM analysis

Required Modifications to BigBite Code for GEMs

Analyzing GEM trackers is easier:

- No L-R ambiguity
- Additional correlations

Required software changes:

- Add weighted averaging of strip signals
- Disable/remove handling of L-R ambiguities
- Exploit additional information from GEMs for 3D matching:
 - ▶ Amplitude correlation
 - ▶ Time correlation

First two items already finished for PREX GEM analysis

Do we need U-V Chambers?

- Most X-Y **correlations** expected to be **random** \Rightarrow U-V information may not sufficiently disambiguate because it is similarly random
- **More like planes** may be better than more coordinates (since TreeSearch is more effective with more planes)
- Unlike drift chambers, **amplitude and timing correlations** available

\Rightarrow Need conclusive answers from Monte Carlo

Do we need U-V Chambers?

- Most X-Y **correlations** expected to be **random** \Rightarrow U-V information may not sufficiently disambiguate because it is similarly random
- **More like planes** may be better than more coordinates (since TreeSearch is more effective with more planes)
- Unlike drift chambers, **amplitude and timing correlations** available

\Rightarrow Need conclusive answers from Monte Carlo

Do we need U-V Chambers?

- Most X-Y **correlations** expected to be **random** \Rightarrow U-V information may not sufficiently disambiguate because it is similarly random
- **More like planes** may be better than more coordinates (since TreeSearch is more effective with more planes)
- Unlike drift chambers, **amplitude and timing correlations** available

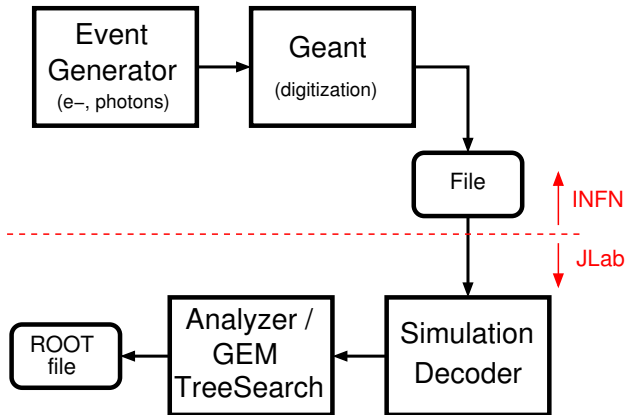
\Rightarrow Need conclusive answers from Monte Carlo

Do we need U-V Chambers?

- Most X-Y **correlations** expected to be **random** \Rightarrow U-V information may not sufficiently disambiguate because it is similarly random
- **More like planes** may be better than more coordinates (since TreeSearch is more effective with more planes)
- Unlike drift chambers, **amplitude and timing correlations** available

\Rightarrow Need conclusive answers from Monte Carlo

Tracking Monte Carlo



Summary

- TreeSearch algorithm appears well suited for SBS Front Tracker reconstruction
- Successfully used by numerous other experiments
- To be tested with GEM data in PREX
- Preparations for SBS Tracking Monte Carlo in progress

Summary

- TreeSearch algorithm appears well suited for SBS Front Tracker reconstruction
- Successfully used by numerous other experiments
- To be tested with GEM data in PREX
- Preparations for SBS Tracking Monte Carlo in progress

Summary

- TreeSearch algorithm appears well suited for SBS Front Tracker reconstruction
- Successfully used by numerous other experiments
- To be tested with GEM data in PREX
- Preparations for SBS Tracking Monte Carlo in progress

Summary

- TreeSearch algorithm appears well suited for SBS Front Tracker reconstruction
- Successfully used by numerous other experiments
- To be tested with GEM data in PREX
- Preparations for SBS Tracking Monte Carlo in progress