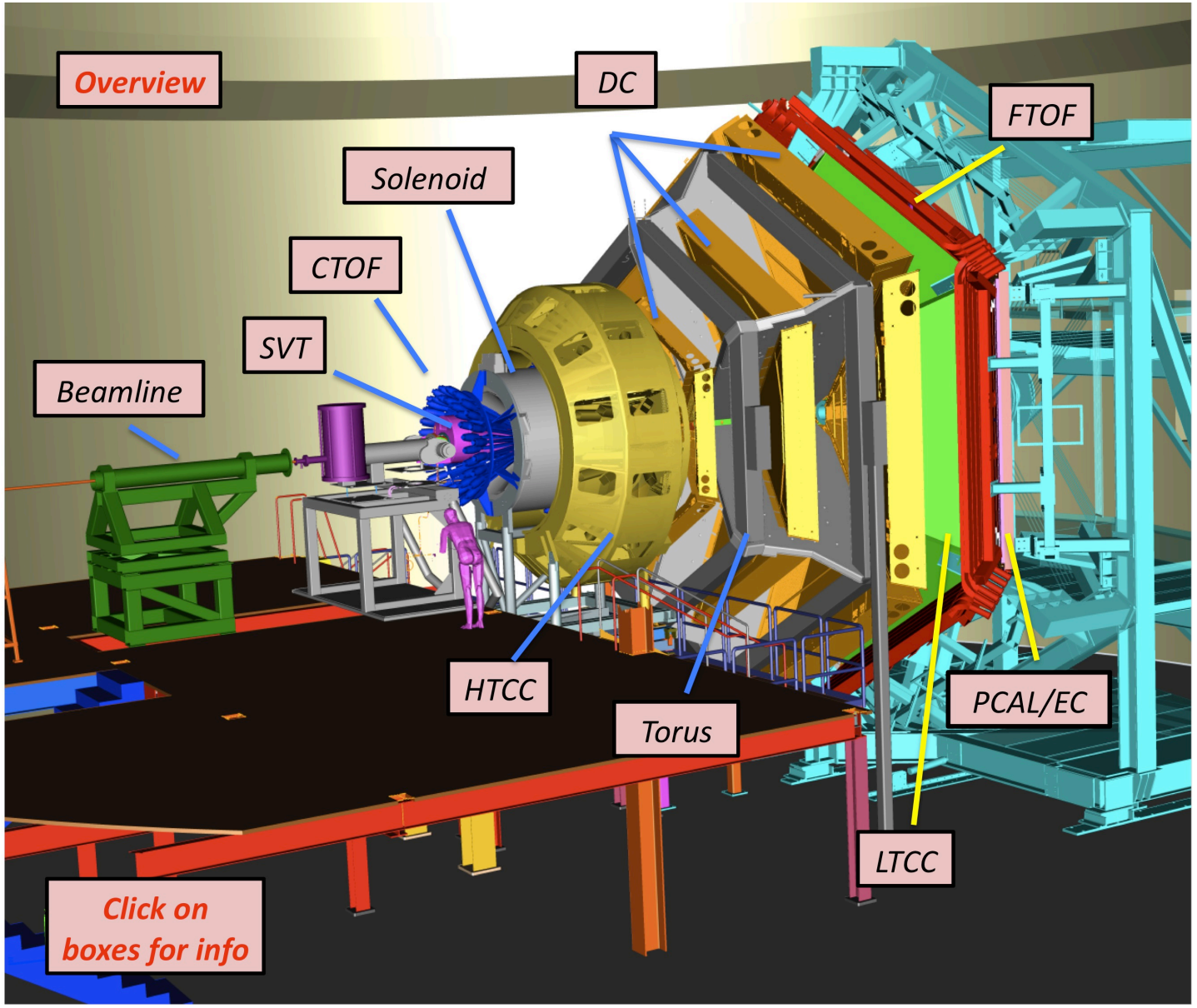
snr: Segmented Noise Removal

Something of an alternative approach

David Heddle, CNU & JLab

The Hall B CLAS Detector

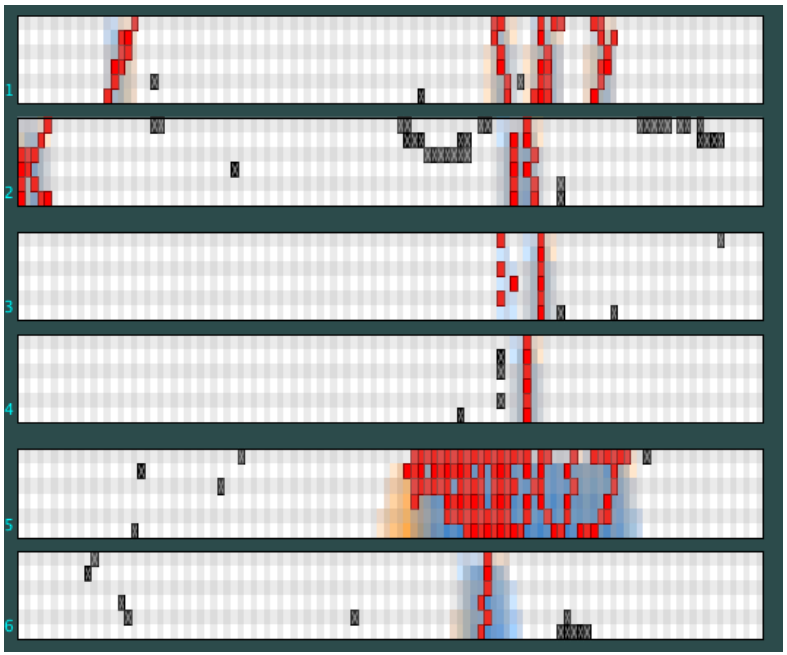


Introduction

Data in the CLAS drift chambers contains segments (parts of tracks), uncorrelated noise (random hits), and correlated noise (out of time tracks and secondary low-energy particles.)

The noise multiplies the combinatoric possibilities when looking for and linking segments into complete tracks.

There are also inefficiencies, when wires that should record a hit fail to do so, which can result in “missing layers”.

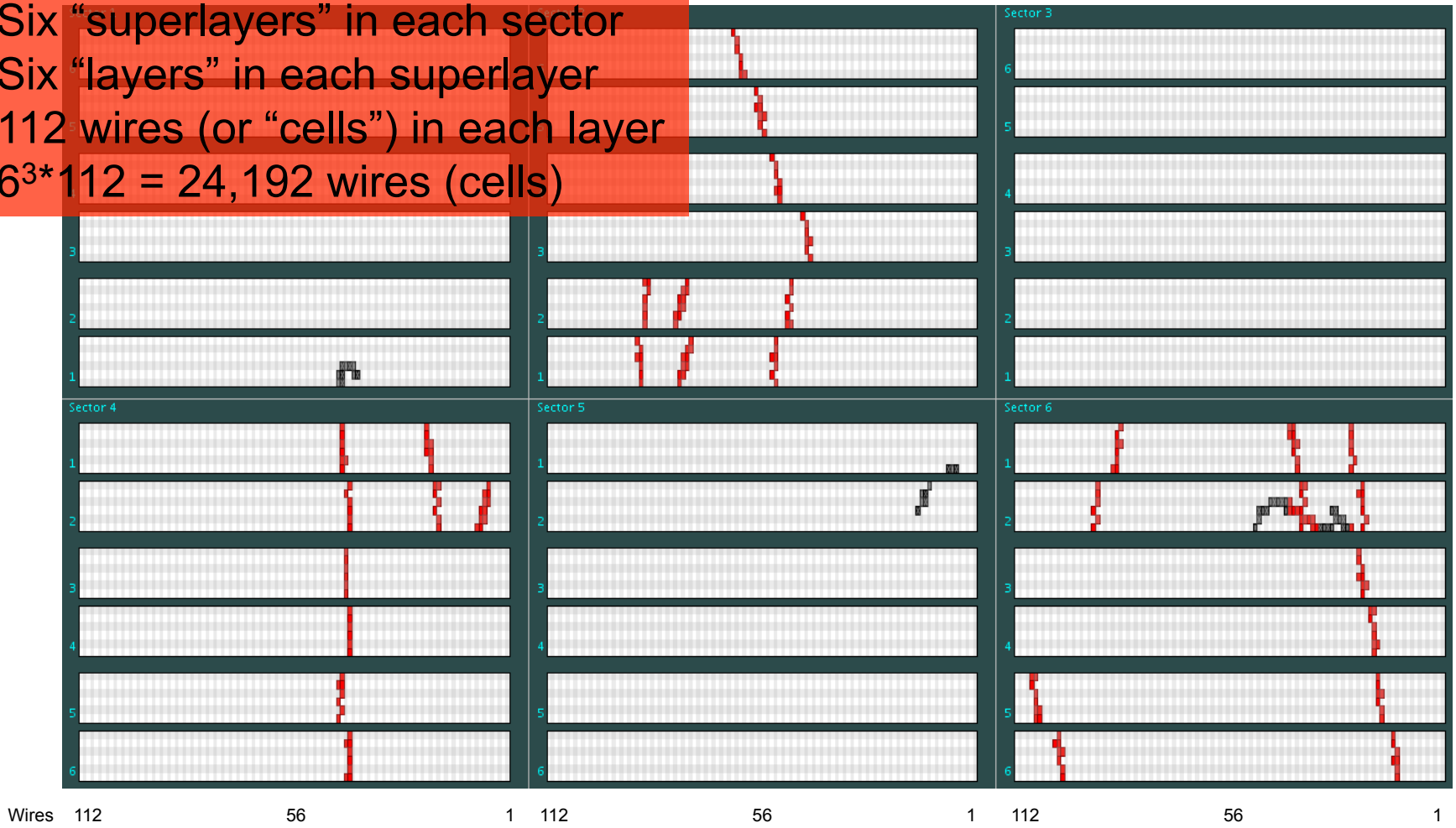


A representation of one sector of the CLAS drift chambers. Tracks originate at the top of the figure and traverse six *superlayers* as numbered. Each superlayer consists of six *layers* stacked vertically. Each layer contains 112 wires numbered, historically, from right to left. The hits are color coded to indicate what data (in red) the *snr* algorithm will preserve and what (in black) it discards as noise. The orangeish and bluish coloring are the *masks*, an artifact of *snr*, as described below.

A segment is a superlayer construct representing a potential part of an entire track—which is typically comprised of six linked segments, one per superlayer.

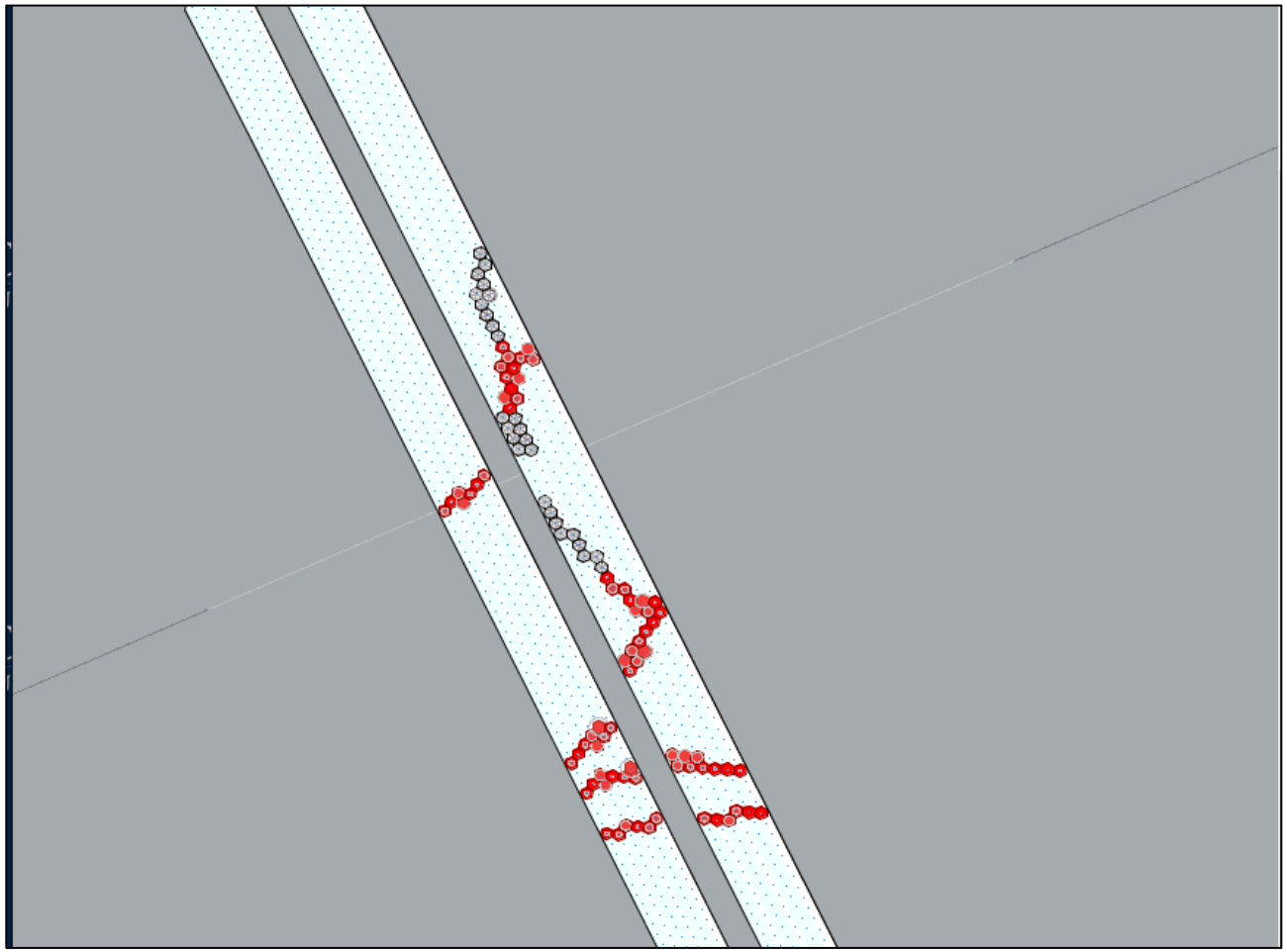
CLAS Drift Chambers

Six Sectors
Six “superlayers” in each sector
Six “layers” in each superlayer
112 wires (or “cells”) in each layer
6^3*112 = 24,192 wires (cells)



This is an unrealistically clean simulation. The “real” hits are shown in red, the uncorrelated noise hits are in black. A real event will have approximately 10 times more hits.

The Problem: Finding segments in the presence of noise and missing layers



Note that noise is not the only problem. There are also inefficiencies in the form of wires that were hit but did not fire, resulting in segments with “missing layers”.

(Partial) Solution

- Identify the noise with a very fast algorithm
- Remove the noise, which will make the full track-finding faster and data files smaller
- The algorithm can allow
  - A small rate of “leakers” (noise that sneaks through, generally in the vicinity of a real track)
  - Absolutely **zero** rate of “false alarms” which is when real data are misidentified as noise.

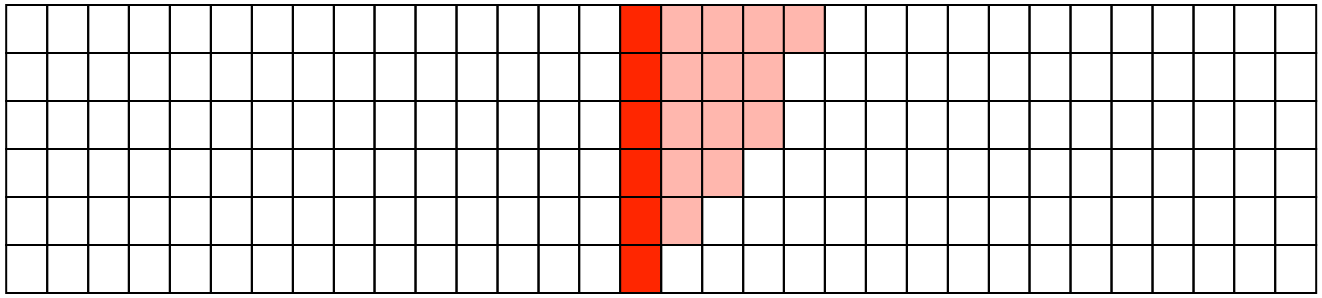
Algorithm

- Uses bitwise algorithm on extended words. (Each Drift Chamber layer is represented by composite 64 bit integers)
- Effectively turns a single CPU into 64 parallel processors—*this algorithm is very fast.*
- Programmable
  - Accommodates any number of missing layers
  - No special layer
  - Set for different curvatures (momenta)
  - Within parameter coverage
    - Some false negatives (noise that was missed and left behind)
    - **No** false positives (good data identified as noise and mistakenly removed)

Method

- Created arbitrary size words (e.g., 112 bits)
- Use one bit per wire
- Developed common (and one uncommon) bitwise operators on these extended words

Programming: Layer Shift Parameters



Layer	*Shift*
6	4
5	3
4	3
3	2
2	1
1	0

What makes a segment candidate?

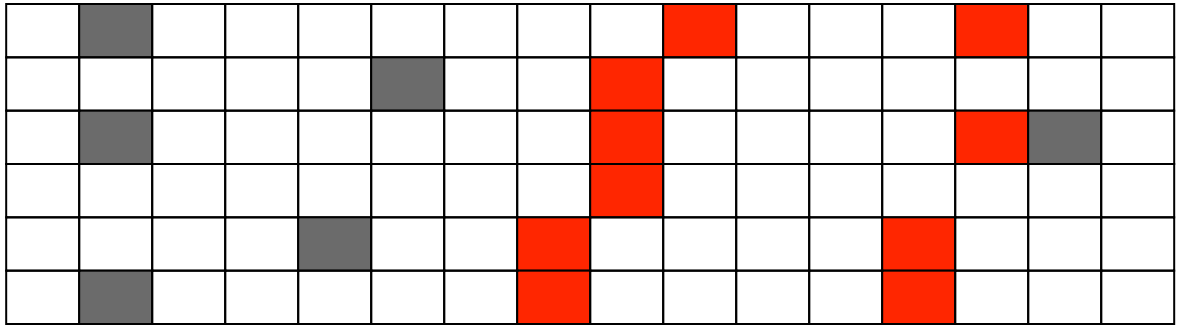
• Ignoring (for now) missing layers, a right bending segment candidate has a hit at wire *n* and at least one hit in a red cell in all other layers.

• There are left shifts as well, and they need not be symmetric.

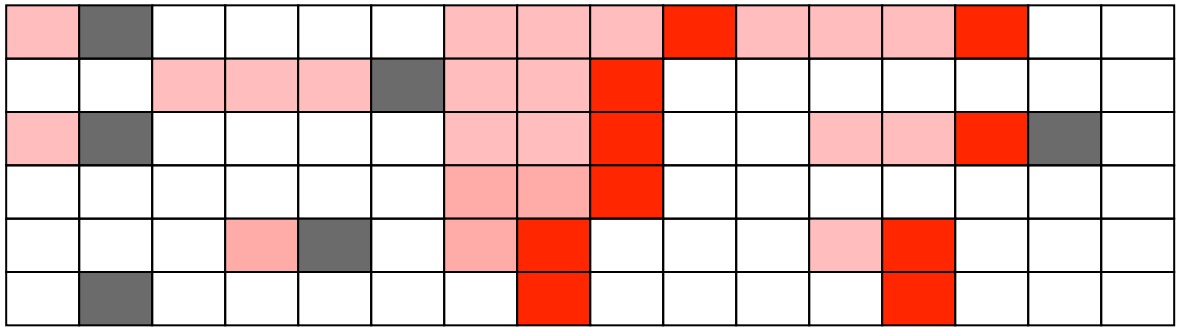
Example: Right Benders

- **Allow two missing layers**
- **Layer Shifts: 0,1,2,2,3,3**

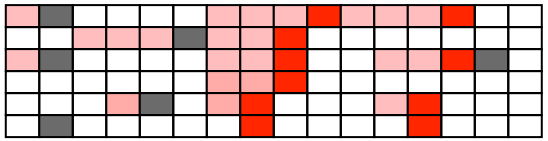
Data



Step 0: “Bleed” the data *left* based on the layer shifts



Data (The coloring is for visualization only)



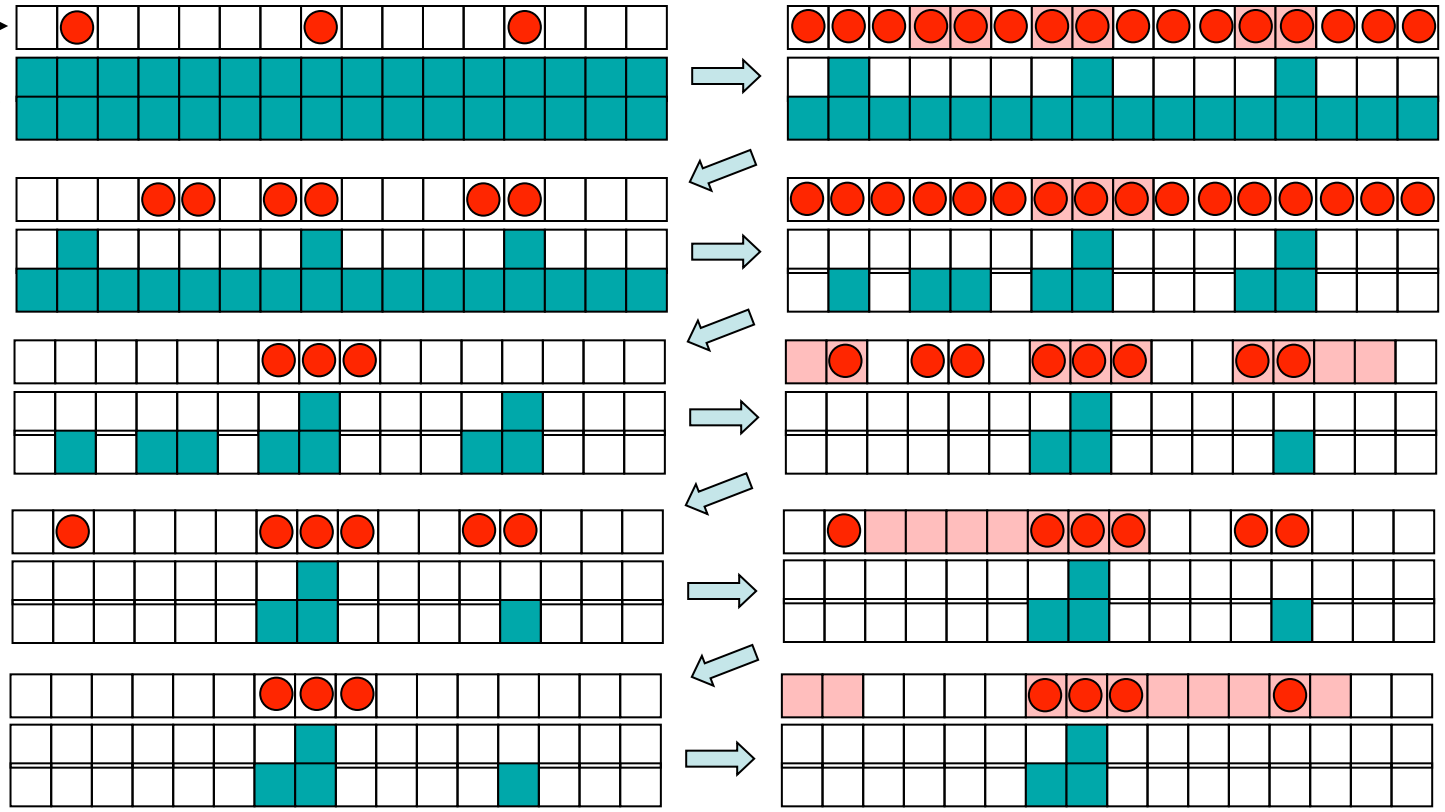
Step 1: segments = segments .AND. Data from current layer (initialize as layer 1)

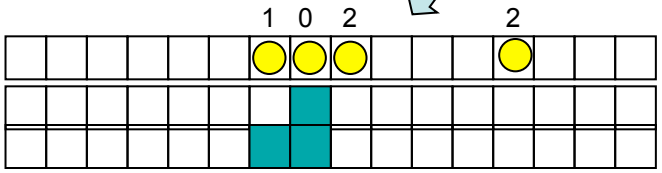
Step 2: fill in missing segments using "missing" pool—until depleted.

Segment candidates

"missing layer" pool

shaded boxes from layer 2 data



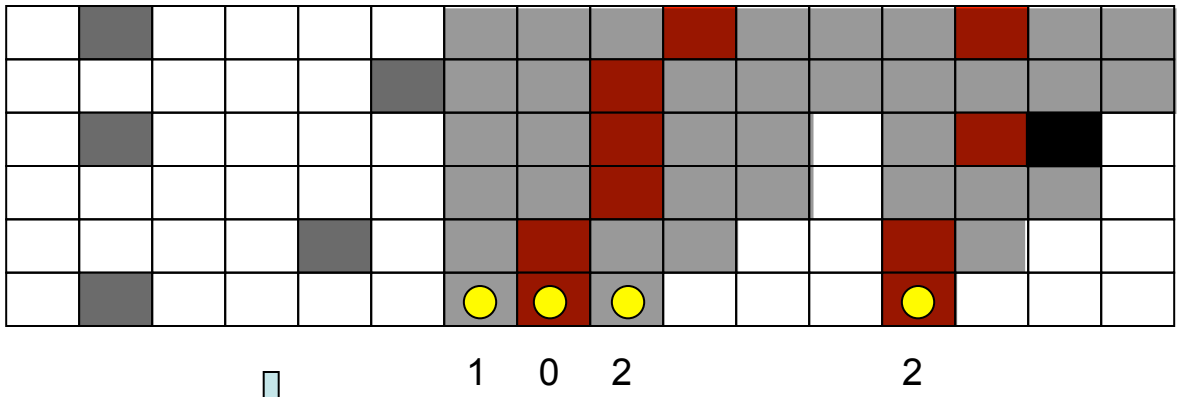


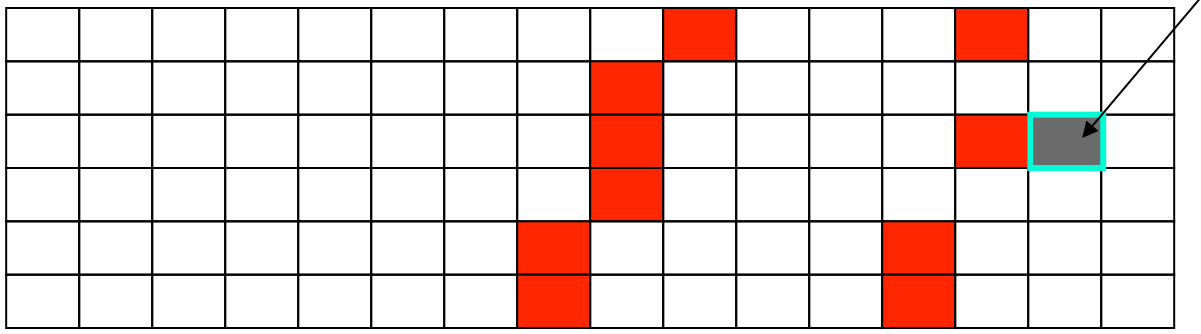
Four candidates. (Two with two missing layers, one with one, one with none)

Repeat Steps 1 and 2 six times

Final Step: Noise Removal

1. Create mask based on segments found and level shifts.
2. .AND. mask with original data





Noise in vicinity of segments is not removed