Reconstructing 3D hit information directly from 2D projections

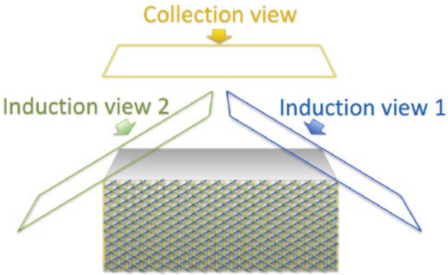
Sep 16, 2019

Chris Backhouse – University College London
for the DUNE collaboration

Introduction

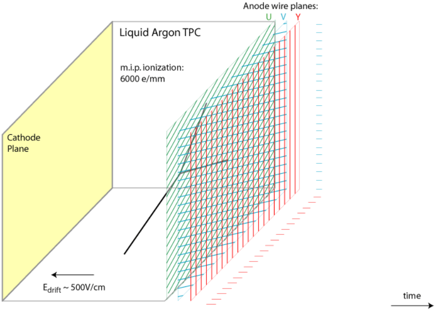
- "3D" neutrino detectors normally only provide 2D projections
- Usually reconstruct 2D objects and combine into 3D
- I am presenting a different approach to go directly to 3D hits

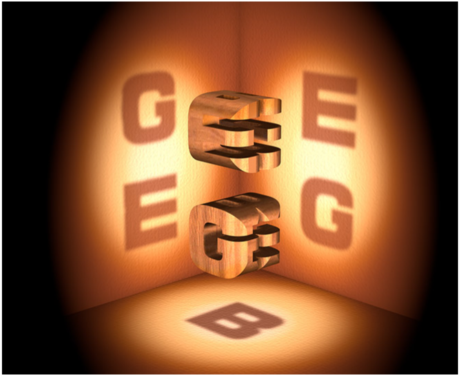
- Problem statement
- Prior art
- Regularization
- SpacePointSolver & WireCell
- Future directions



"Three-dimensional Imaging for Large LArTPCs" arXiv:1803.04650
lar.bnl.gov/wire-cell

Problem statement



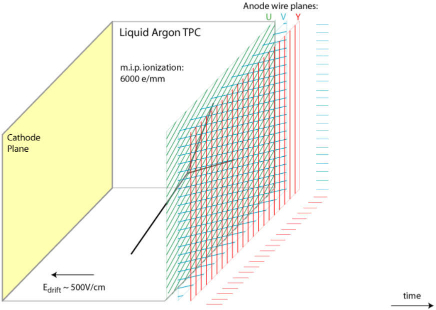


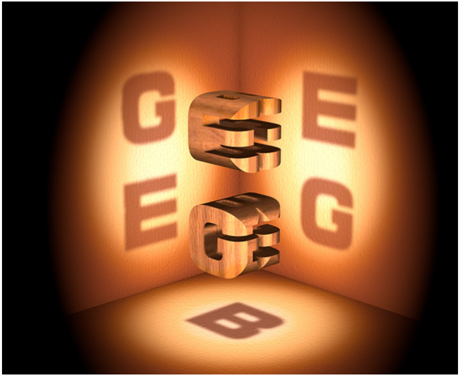
- Given observed charges q_i find deposits in 3D space p_j such that

$$\sum_j^{\text{sites}} T_{ij} p_j = q_i$$

where $T_{ij} \in 0, 1$ encodes which 3D points could cause which hits

Problem statement



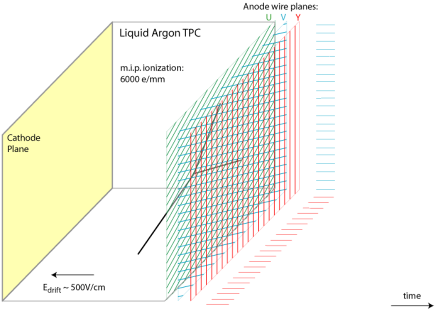


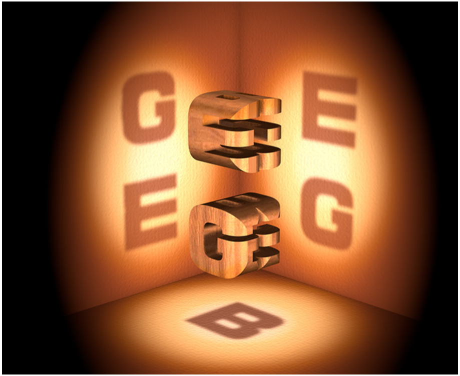
- Given observed charges q_i find deposits in 3D space p_j such that

$$\sum_j^{\text{sites}} T_{ij} p_j = q_i$$

where $T_{ij} \in 0, 1$ encodes which 3D points could cause which hits

Problem statement



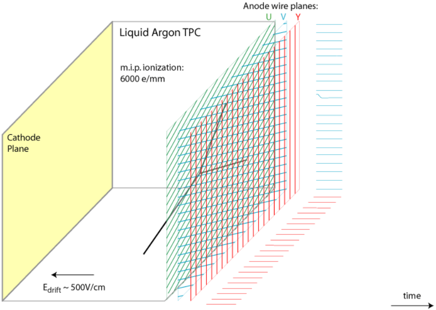


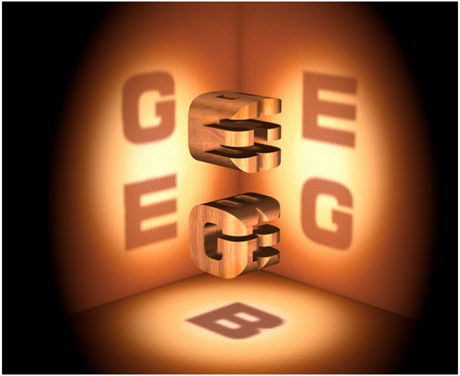
- Given observed charges q_i find deposits in 3D space p_j such that

$$\sum_j^{\text{sites}} T_{ij} p_j = q_i$$

where $T_{ij} \in 0, 1$ encodes which 3D points could cause which hits

Problem statement



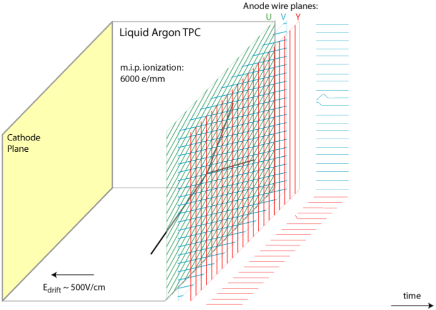


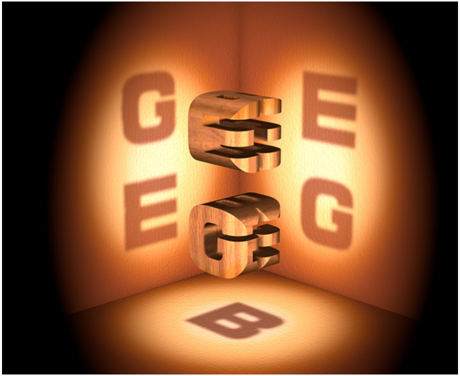
- Given observed charges q_i find deposits in 3D space p_j such that

$$\sum_j^{\text{sites}} T_{ij} p_j = q_i$$

where $T_{ij} \in 0, 1$ encodes which 3D points could cause which hits

Problem statement



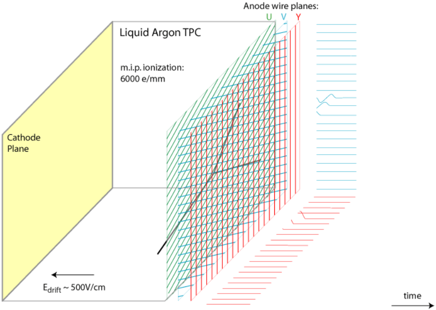


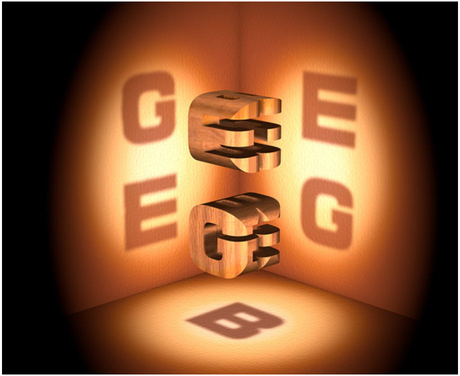
- Given observed charges q_i find deposits in 3D space p_j such that

$$\sum_{j}^{\text{sites}} T_{ij} p_j = q_i$$

where $T_{ij} \in 0, 1$ encodes which 3D points could cause which hits

Problem statement



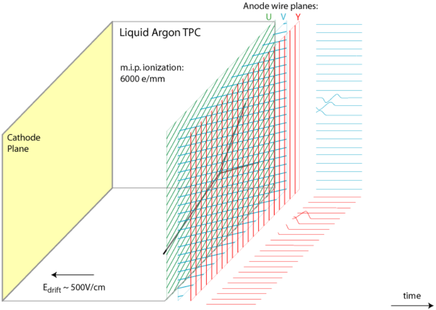


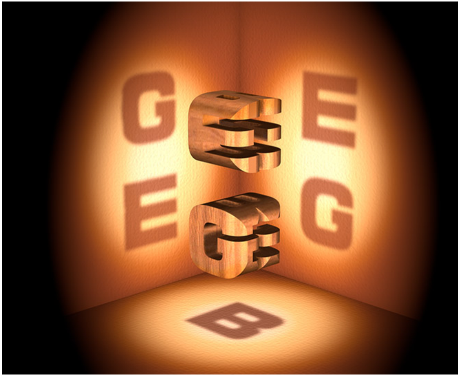
- Given observed charges q_i find deposits in 3D space p_j such that

$$\sum_j^{\text{sites}} T_{ij} p_j = q_i$$

where $T_{ij} \in 0, 1$ encodes which 3D points could cause which hits

Problem statement



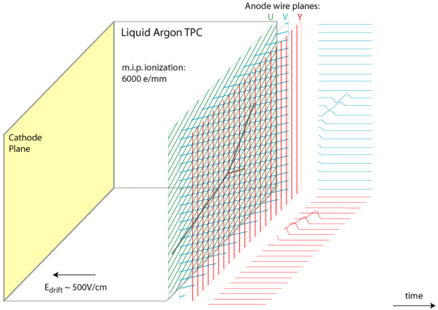


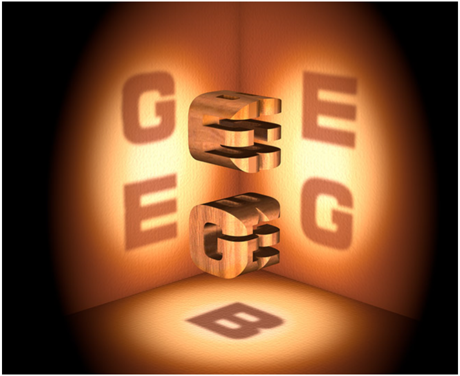
- Given observed charges q_i find deposits in 3D space p_j such that

$$\sum_j^{\text{sites}} T_{ij} p_j = q_i$$

where $T_{ij} \in 0, 1$ encodes which 3D points could cause which hits

Problem statement



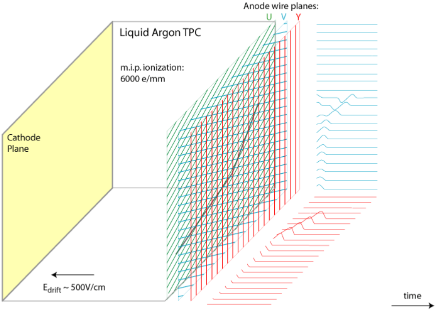


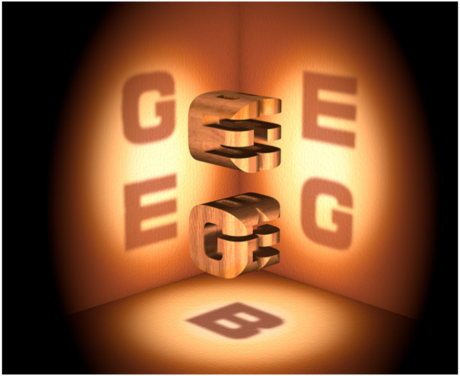
- Given observed charges q_i find deposits in 3D space p_j such that

$$\sum_j^{\text{sites}} T_{ij} p_j = q_i$$

where $T_{ij} \in 0, 1$ encodes which 3D points could cause which hits

Problem statement



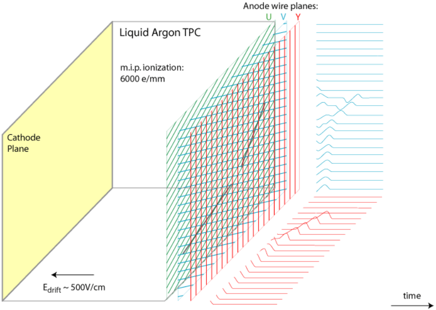


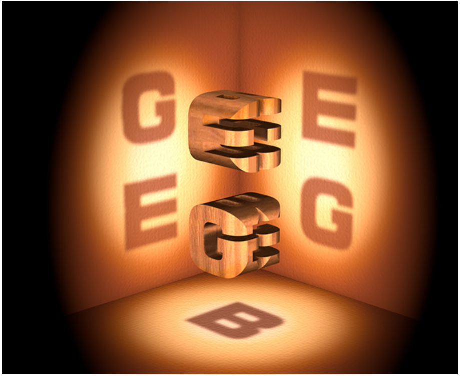
- Given observed charges q_i find deposits in 3D space p_j such that

$$\sum_j^{\text{sites}} T_{ij} p_j = q_i$$

where $T_{ij} \in 0, 1$ encodes which 3D points could cause which hits

Problem statement



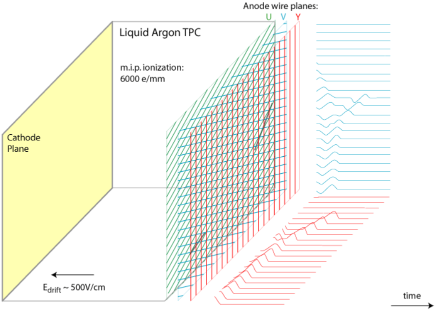


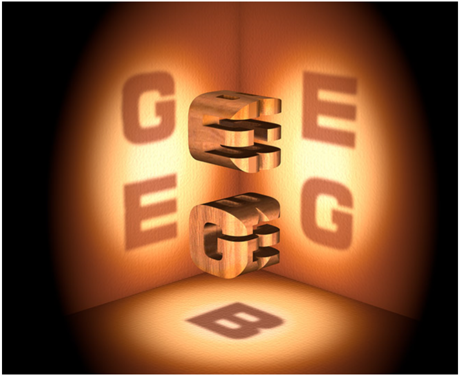
- Given observed charges q_i find deposits in 3D space p_j such that

$$\sum_j^{\text{sites}} T_{ij} p_j = q_i$$

where $T_{ij} \in 0, 1$ encodes which 3D points could cause which hits

Problem statement



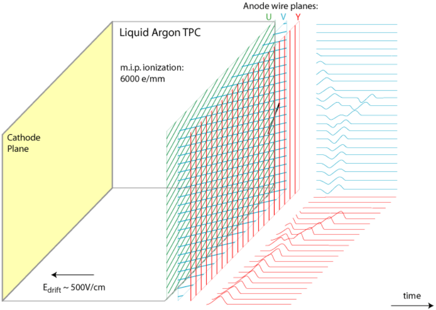


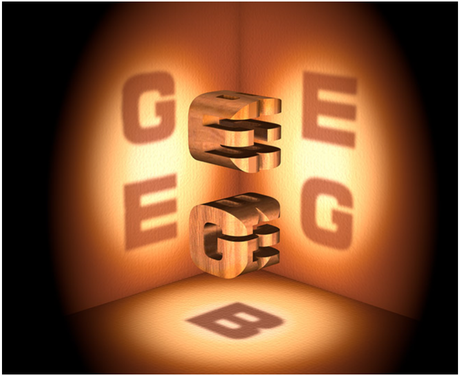
- Given observed charges q_i find deposits in 3D space p_j such that

$$\sum_{j}^{\text{sites}} T_{ij} p_j = q_i$$

where $T_{ij} \in 0, 1$ encodes which 3D points could cause which hits

Problem statement



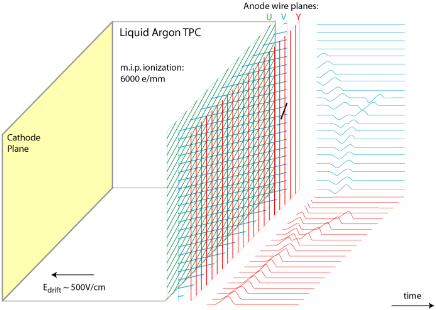


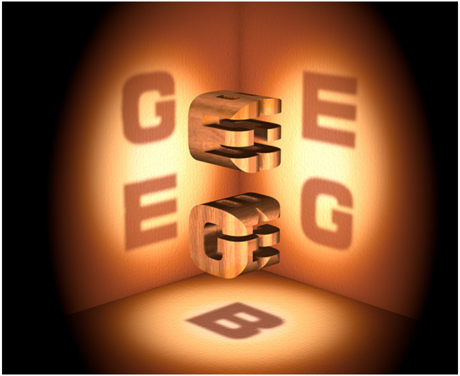
- Given observed charges q_i find deposits in 3D space p_j such that

$$\sum_{j}^{\text{sites}} T_{ij} p_j = q_i$$

where $T_{ij} \in 0, 1$ encodes which 3D points could cause which hits

Problem statement



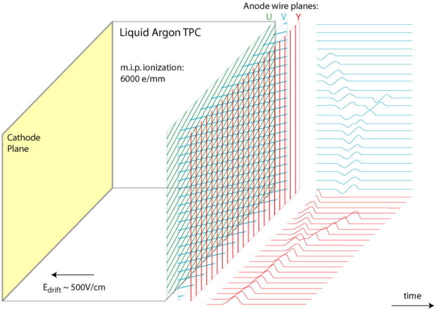


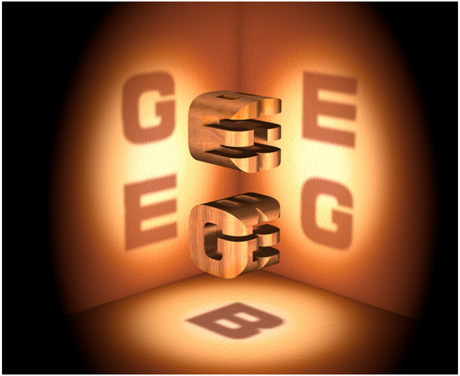
- Given observed charges q_i find deposits in 3D space p_j such that

$$\sum_{j}^{\text{sites}} T_{ij} p_j = q_i$$

where $T_{ij} \in 0, 1$ encodes which 3D points could cause which hits

Problem statement



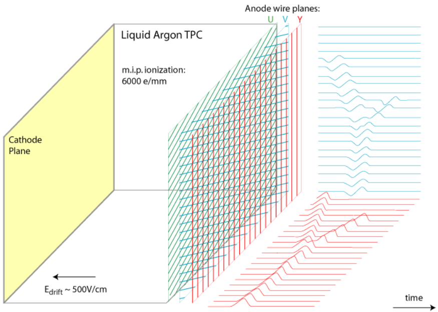


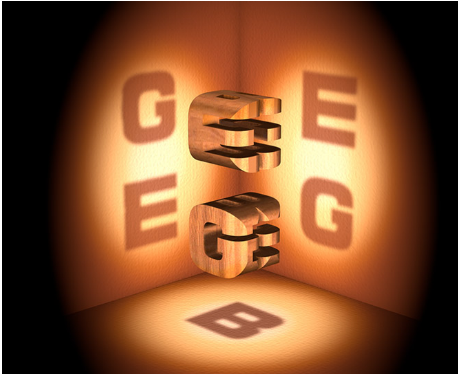
- Given observed charges q_i find deposits in 3D space p_j such that

$$\sum_{j}^{\text{sites}} T_{ij} p_j = q_i$$

where $T_{ij} \in 0, 1$ encodes which 3D points could cause which hits

Problem statement



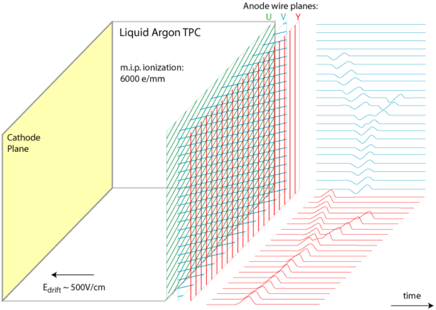


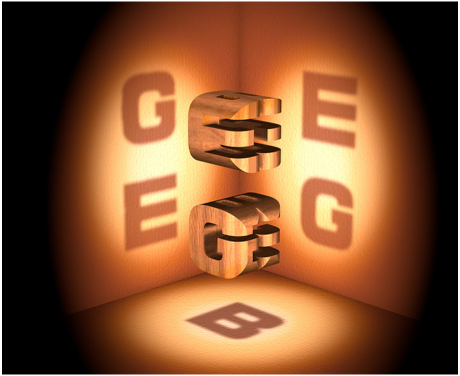
- Given observed charges q_i find deposits in 3D space p_j such that

$$\sum_{j}^{\text{sites}} T_{ij} p_j = q_i$$

where $T_{ij} \in 0, 1$ encodes which 3D points could cause which hits

Problem statement



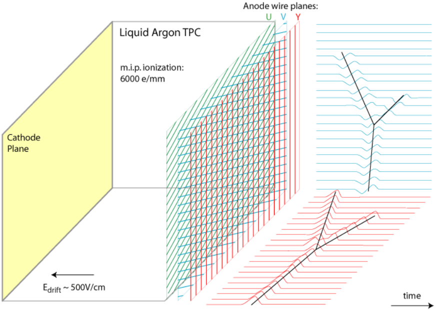


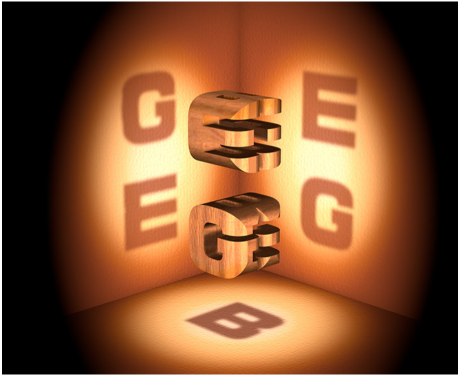
- Given observed charges q_i find deposits in 3D space p_j such that

$$\sum_{j}^{\text{sites}} T_{ij} p_j = q_i$$

where $T_{ij} \in 0, 1$ encodes which 3D points could cause which hits

Problem statement



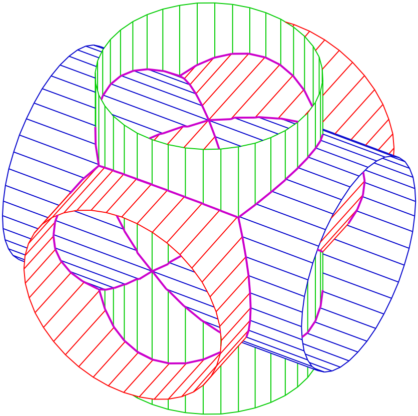


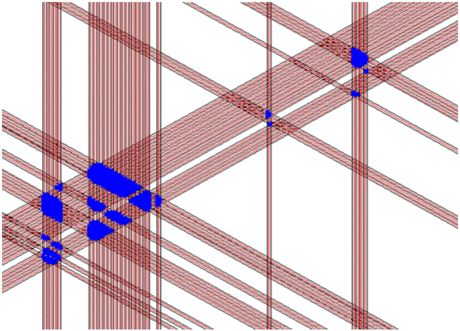
- Given observed charges q_i find deposits in 3D space p_j such that

$$\sum_j^{\text{sites}} T_{ij} p_j = q_i$$

where $T_{ij} \in 0, 1$ encodes which 3D points could cause which hits

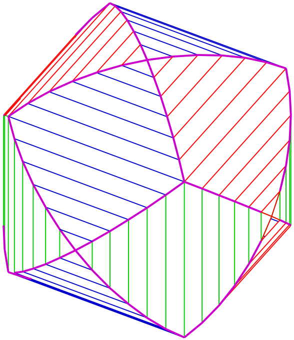
Problem statement

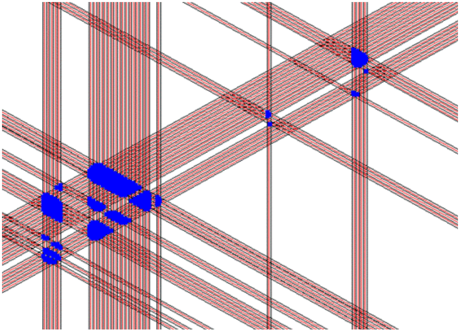




- Underconstrained problem – $3N$ measurements for N^3 unknowns
- A form of unfolding problem – need some kind of regularization

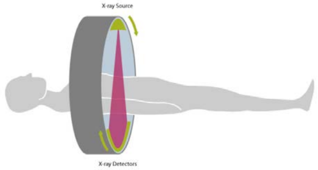
Problem statement

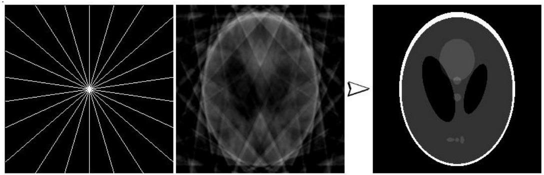




- Underconstrained problem – $3N$ measurements for N^3 unknowns
- A form of unfolding problem – need some kind of regularization

Prior art





available portion of the spectrum (11 radial lines)

Back-projection estimate

Estimate after convergence (exact reconstruction)

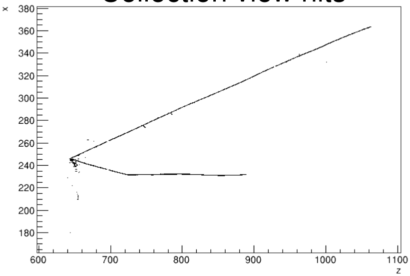
- "Compressed sensing" – recover original image from surprisingly little information if you have a model, *e.g.* the image is sparse

- Mathematical proofs mostly use a random transfer matrix
- We have a lot of structure/correlations (*e.g.* isochronous tracks)

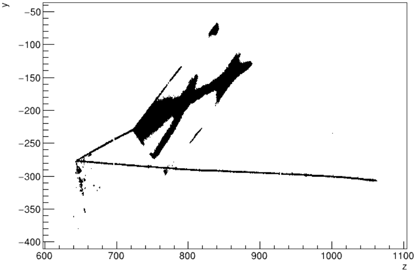
Candes, Romberg, and Tao, *"Stable Signal Recovery from Incomplete and Inaccurate Measurements"* arXiv:math/0503066

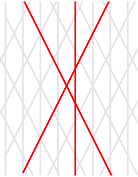
Example – DUNE FD ν_μ sim

Collection view hits



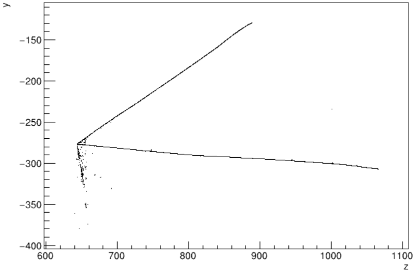
All coincidences – orthogonal view



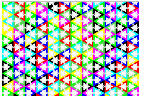


SpacePointSolver distributes collection wire charge over 10μs/5mm “triplets”

Truth – orthogonal view

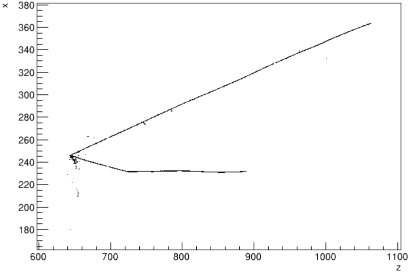


WireCell distributes charge among 3D “cells”

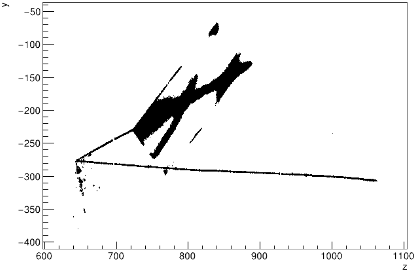


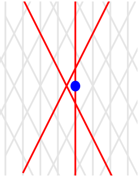
Example – DUNE FD ν_μ sim

Collection view hits



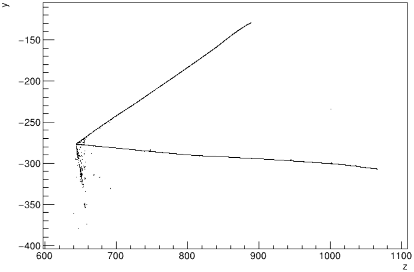
All coincidences – orthogonal view



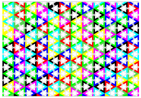


SpacePointSolver distributes collection wire charge over 10μs/5mm "triplets"

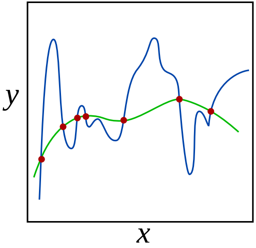
Truth – orthogonal view

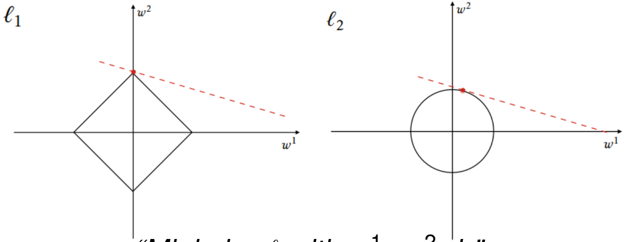


WireCell distributes charge among 3D "cells"



Regularization



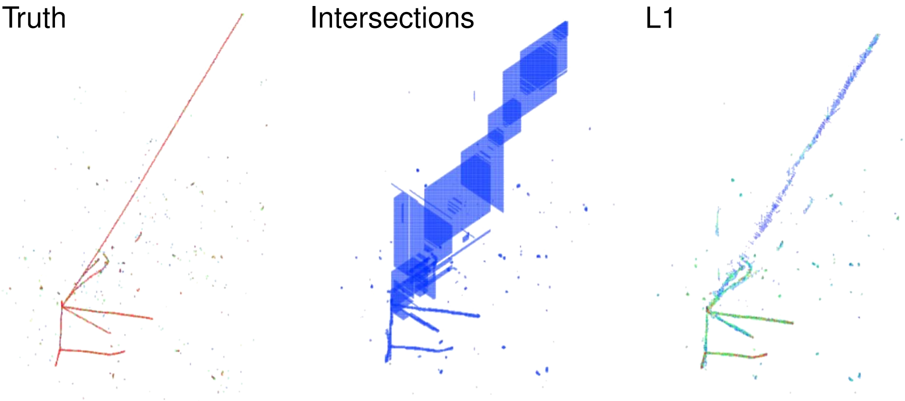


"Minimize ℓ with $w^1+w^2=k$"

$$L^0 = \sum_i \begin{cases} 0 & p_i = 0 \\ 1 & p_i > 0 \end{cases} \qquad L^1 = \sum_i |p_i| \qquad L^2 = \sum_i p_i^2$$

- Want simplest charge distribution that explains the observations
- Regularization concept familiar from unfolding problems
- Here "simplest" = "sparsest" *i.e.* minimize L^0 – NP-hard problem
- But solution minimizing L^1 norm will also be sparse in general
- Space not fully differentiable, but it is single-minimum'd

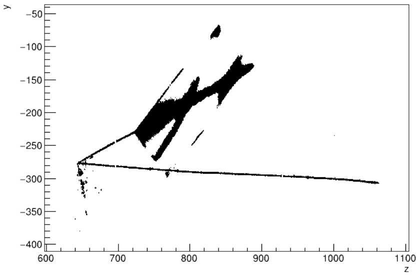
Regularization in WireCell



- Addition of L^1 regularization greatly improves reconstruction
- Previous approach was a brute-force search to minimize L^0

SpacePointSolver built-in L^1

All coincidences



$$\text{minimize } \chi^2 = \sum_{i}^{\text{iwires}} \left(q_i - \sum_{j}^{\text{sites}} T_{ij} p_j \right)^2$$

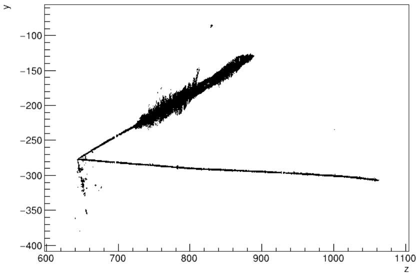
$$\text{subject to } p_j \geq 0 \text{ for all } j$$

$$\text{and } \sum_{j}^{\text{sites}} U_{jk} p_j = Q_k \text{ for all } k$$

- SpacePointSolver distributes collection wire charge over relevant triplets
- Total charge $\sum p_i \equiv \sum Q_k$ constant by construction
- So a form of L^1 regularization is built into the foundations

SpacePointSolver built-in L^1

Minimize χ^2



$$\text{minimize } \chi^2 = \sum_i^{\text{iwires}} \left(q_i - \sum_j^{\text{sites}} T_{ij} p_j \right)^2$$

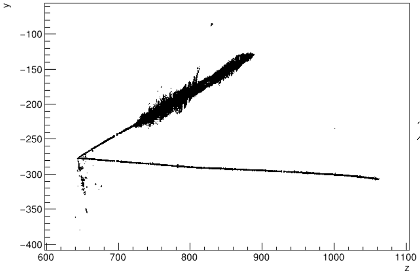
$$\text{subject to } p_j \geq 0 \text{ for all } j$$

$$\text{and } \sum_j^{\text{sites}} U_{jk} p_j = Q_k \text{ for all } k$$

- SpacePointSolver distributes collection wire charge over relevant triplets
- Total charge $\sum p_i \equiv \sum Q_k$ constant by construction
- So a form of L^1 regularization is built into the foundations

SpacePointSolver cross-term

Minimize χ^2

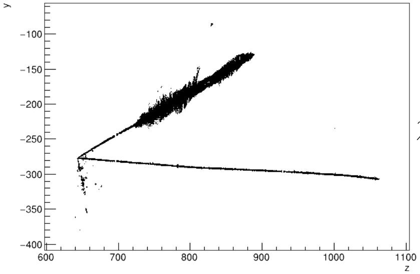


$$\chi^2 = \sum_i^{\text{iwires}} \left(q_i - \sum_j^{\text{sites}} T_{ij} p_j \right)^2$$

- Knows the solution should be *sparse* but it should also be *compact*
- Room for one more term in the χ^2 while preserving minimizability

SpacePointSolver cross-term

Minimize χ^2

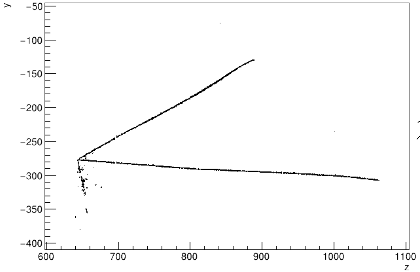


$$\chi^2 = \sum_i^{\text{iwires}} \left(q_i - \sum_j^{\text{sites}} T_{ij} p_j \right)^2 - \sum_{ij}^{\text{sites}} V_{ij} p_i p_j$$

- Knows the solution should be *sparse* but it should also be *compact*
- Room for one more term in the χ^2 while preserving minimizability
- Lower χ^2 for a solution that places the p's closer together

SpacePointSolver cross-term

Minimize χ^2 inc. cross-term

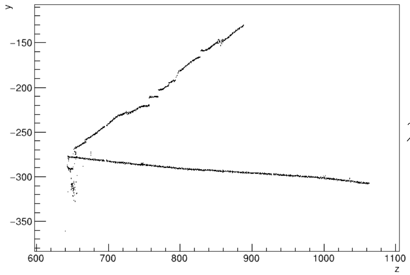


$$\chi^2 = \sum_i^{\text{iwires}} \left(q_i - \sum_j^{\text{sites}} T_{ij} p_j \right)^2 - \sum_{ij}^{\text{sites}} V_{ij} p_i p_j$$

- Knows the solution should be *sparse* but it should also be *compact*
- Room for one more term in the χ^2 while preserving minimizability
- Lower χ^2 for a solution that places the *p*'s closer together
- Form of *V* is ad-hoc. Used $V_{ij} = \lambda \exp\left(-\frac{|\vec{r}_1 - \vec{r}_2|}{2\text{cm}}\right)$
- λ controls regularization strength. Too strong and solution degrades again

SpacePointSolver cross-term

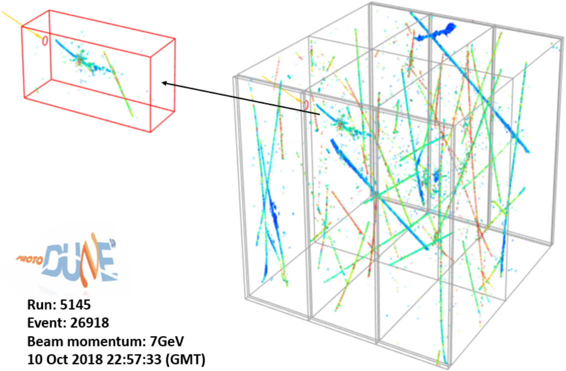
Over-regularized



$$\chi^2 = \sum_i^{\text{iwires}} \left(q_i - \sum_j^{\text{sites}} T_{ij} p_j \right)^2 - \sum_{ij}^{\text{sites}} V_{ij} p_i p_j$$

- Knows the solution should be *sparse* but it should also be *compact*
- Room for one more term in the χ^2 while preserving minimizability
- Lower χ^2 for a solution that places the *p*'s closer together
- Form of *V* is ad-hoc. Used $V_{ij} = \lambda \exp\left(-\frac{|\vec{r}_1 - \vec{r}_2|}{2\text{cm}}\right)$
- λ controls regularization strength. Too strong and solution degrades again

Application



- SpacePointSolver and WireCell are available within larsoft
- Compatible with geometries other than DUNE (including two-view geometries such as DUNE-DP, Lariat, Argoneut)
- Intended as the first step of a natively-3D reconstruction chain
- SpacePointSolver already finding use for ProtoDUNE wire-wrapping disambiguation (>99% correct disambiguation)

Future directions

- Using wire hits can be inconvenient *e.g.* with steep tracks
- Look into unfolding 2D waveforms directly to a 3D charge cloud?

- Definition of the interaction term is ad-hoc and could be tuned
- Can an optimum function somehow be defined from the data? Covariance of truth hit distributions??

- Compressed Sensing ideas are powerful
- Where else can we apply them?

Thanks!