Proyecto trasgo Project

timtrack

A new formalism for charge particle track reconstruction with timing detectors

Juan A. Garzón
LabCAF/ Univ. Santiago de Compostela

Outline

- The Trasgo project
- About TimTrack
- TimTrack: matrix formalism
- TimTrack: examples
  - Particle going through a set of strips with time readout
  - HADES-GSI MDCs (Mini Drift Chambers)
- Summary

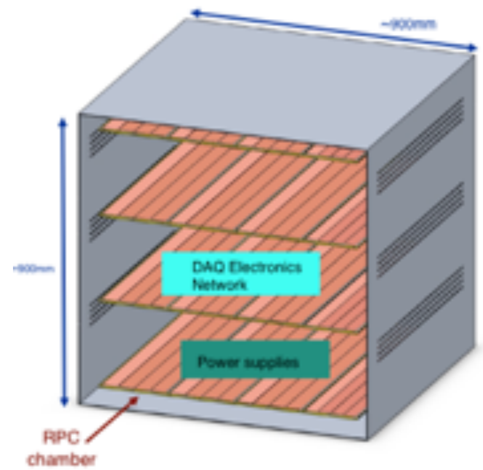
About the name:

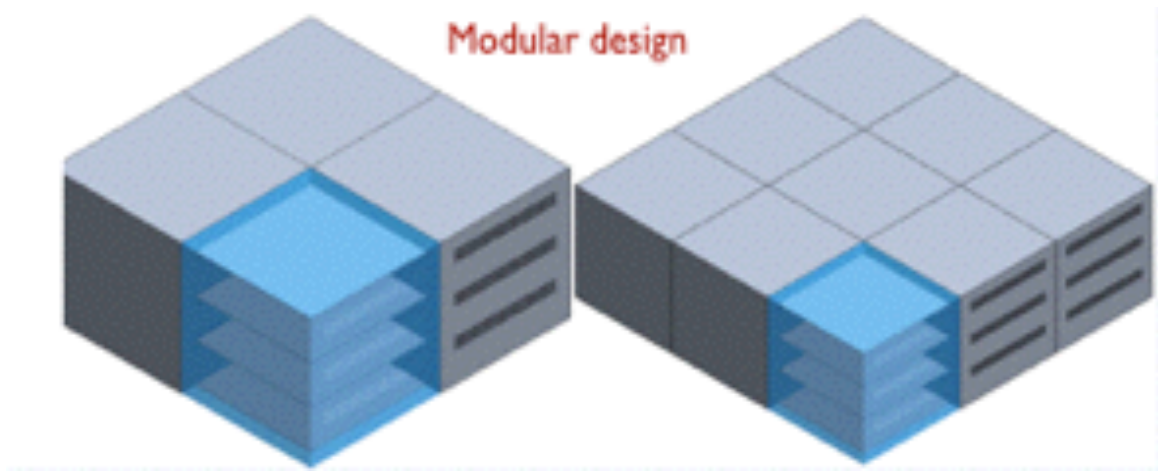
TRASGO (TRAck reconStructinG bOx) = ~Elf = ~Lutin

The TRASGO project

About the concept:

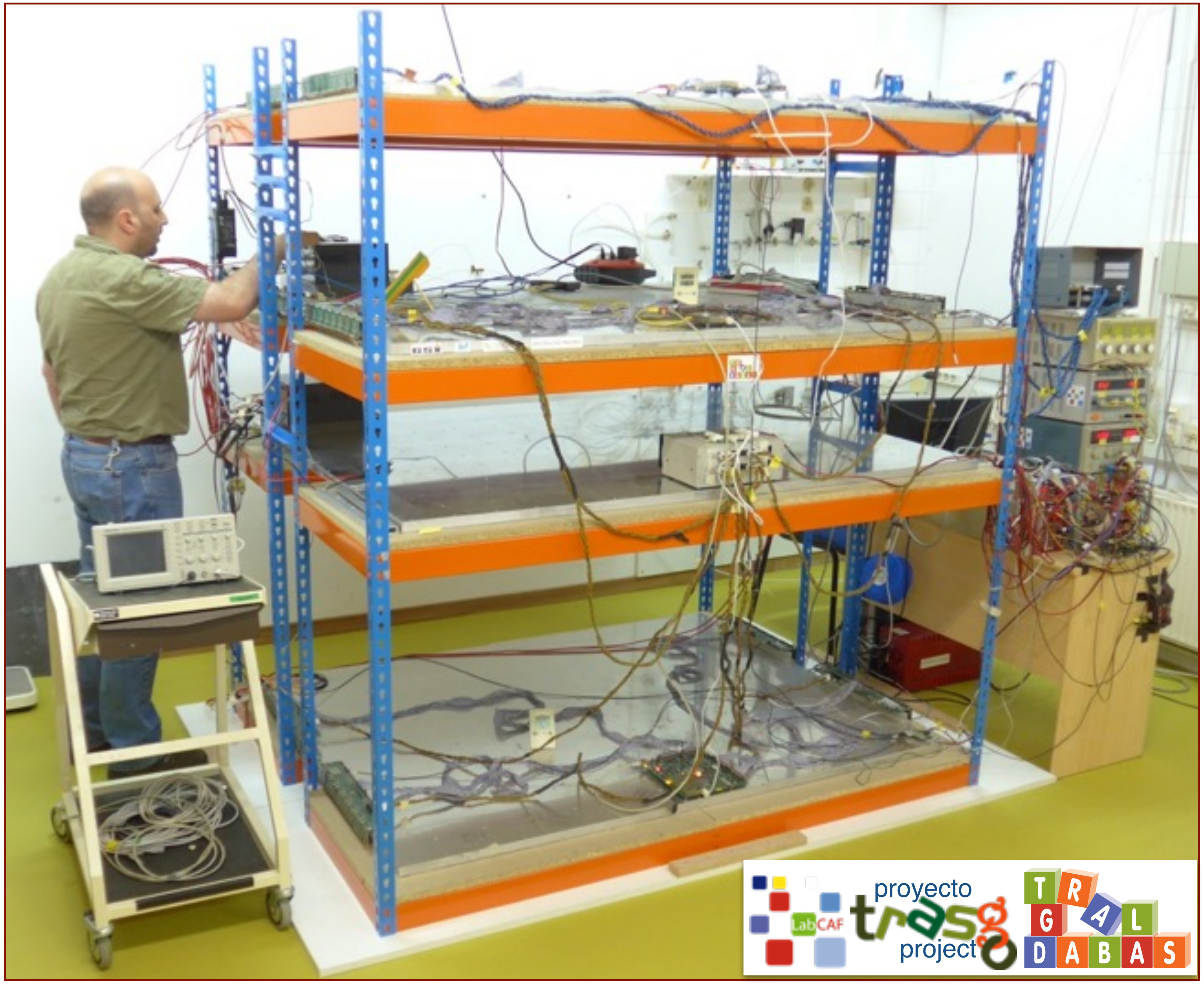
As a complement to the existing cosmic ray detection techniques, the TRASGO program aims the development of cosmic rays detectors fulfilling the following requirements:





- Tracking detector
- Timing, position and angular resolution as good as possible
- Sensitive to both Muon / Electromagnetic showers (software separation)
- Modular concept
- TimTrack reconstruction software

El detector TRAGALDABAS de la USC



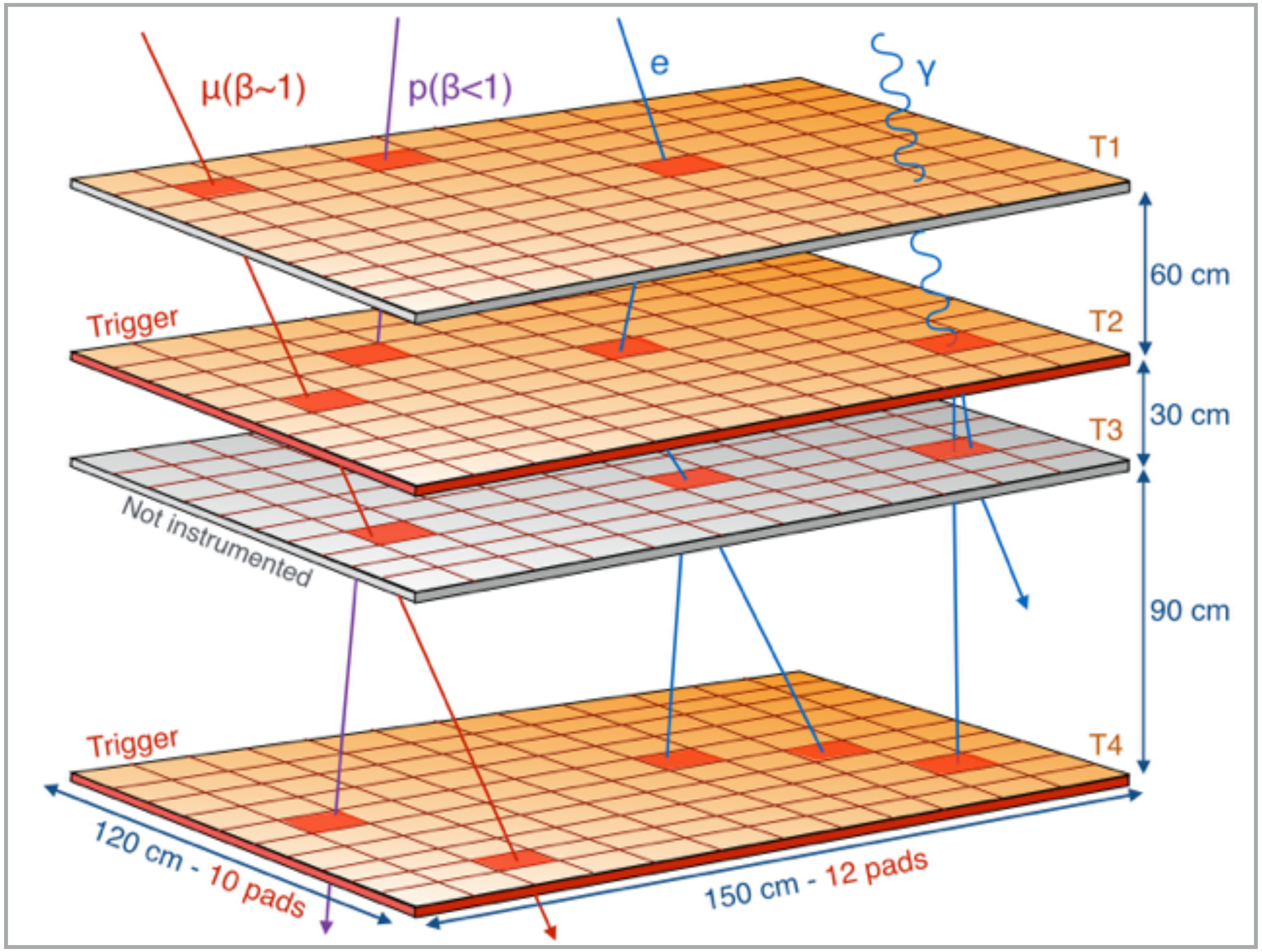
Tragaldabas:

TRAsGo para el AnaLisis De la Alta y Baja atmósfera y de la Actividad Solar

TRAsGo for the AnaLysis of the nuclear matter Decay, the Atmosphere, the earth B-field And the Solar activity

The TRAGALDABAS detector at the USC

Particle Identification



TimTrack software allows to reconstruct tracks without any external trigger time stamp and the velocity of the particle as free parameters

Main features:

- TimTrack is a Least Squares fitting Method using a general Matrix Formalism

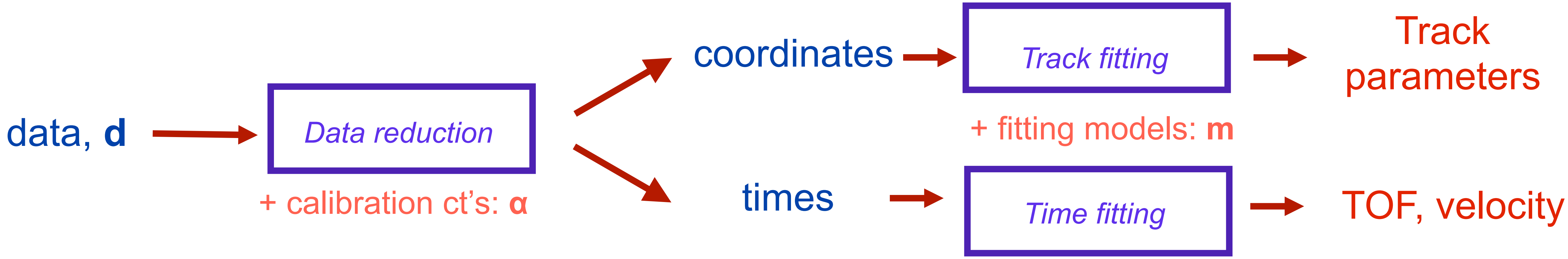
- TimTrack puts all the significative parameters in the fitting model.

- TimTracks works with direct measurements, without any data reduction.

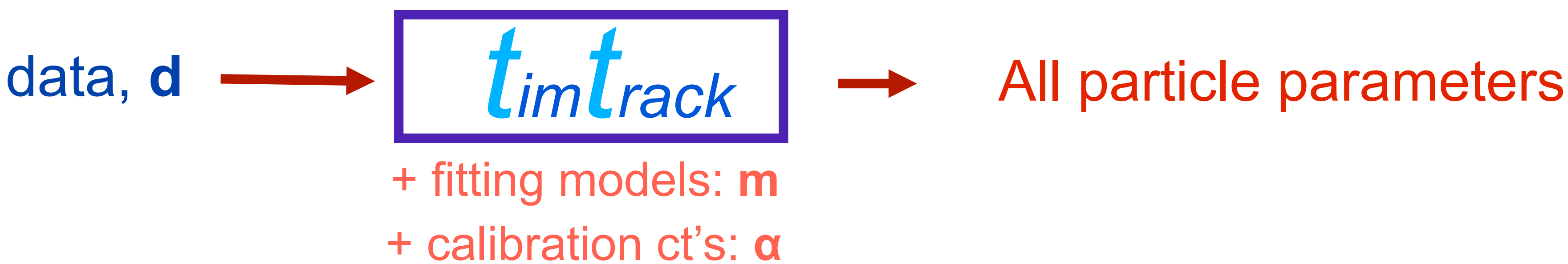
About TimTrack

TimTrack vs. traditional fitting methods:

Traditional track fitting methods:

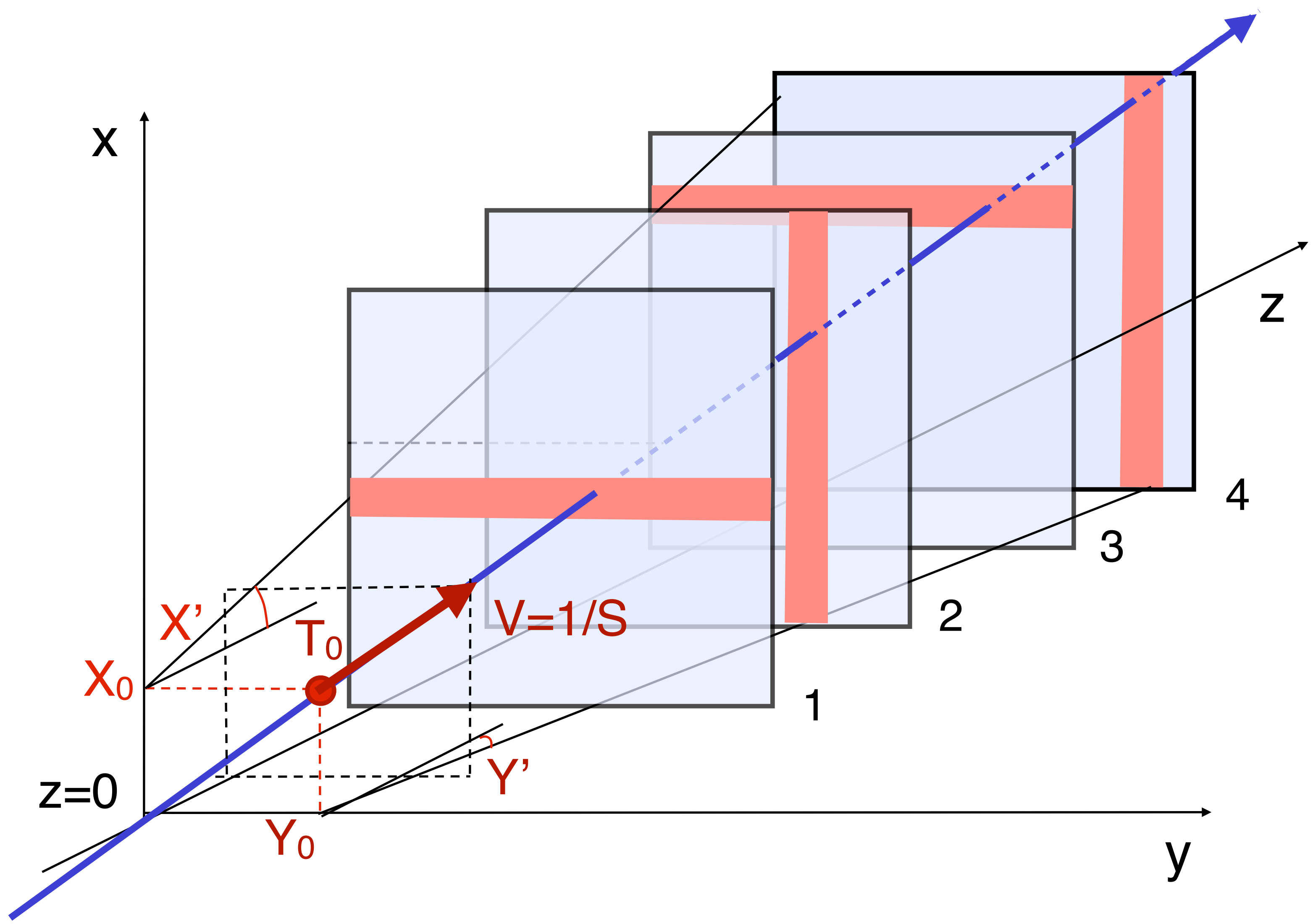


TimTrack:



About TimTrack

Example: a particle trough a set of position and time sensitive detectors



The basic, or minimal, set of fitted parameters is:

s = {X0, X', Y0, Y', T0, S=1/V}

We call it SAETA (SmAllest sET of pArameters) = arrow

TimTrack: matrix formalism

TimTrack is based on the Least Squares Method (LSM)

Least Squares Method (LSM):

Let suppose that we have:

- a set of n_m data: $\mathbf{d}$
- a set of n_s parameters: $\mathbf{s}$
- a model relating both sets: $\mathbf{d}=\mathbf{m}(\mathbf{s})$

The Least Squares Method (LSM) consists in finding the set of parameters $\mathbf{s}$ minimizing the function S:

$$S = \sum_i \left(\frac{d_i - m_i(\mathbf{s})}{\sigma_i} \right)^2$$

TimTrack: matrix formalism

The Least Squares Method (LSM): linear model

1. $$S = [\mathbf{d} - \mathbf{m}(\mathbf{s})]' \cdot W \cdot [\mathbf{d} - \mathbf{m}(\mathbf{s})]$$ (*)

2. $$\mathbf{m}(\mathbf{s}) = G \cdot \mathbf{s} + \mathbf{g}(\mathbf{s}) \qquad \mathbf{g}(\mathbf{s}) = \mathbf{m}(\mathbf{s}) - G \cdot \mathbf{s} = \mathbf{g}_0$$

3. $$K = G' \cdot W \cdot G$$
$$\mathbf{a} = G' \cdot W \cdot (\mathbf{d} - \mathbf{g}_0)$$
$$s_0 = (\mathbf{d} - \mathbf{g}_0)' \cdot W \cdot (\mathbf{d} - \mathbf{g}_0)$$
$$\rightarrow \quad S = \mathbf{s}' \cdot K \cdot \mathbf{s} - 2\mathbf{s}' \cdot \mathbf{a} + s_0$$ (*)

K: configuration matrix
a: reduced vector of data

4. $$\frac{\partial S}{\partial \mathbf{s}} = K \cdot \mathbf{s}_m - \mathbf{a} = \mathbf{0} \quad \rightarrow \quad K \cdot \mathbf{s}_m = \mathbf{a}$$
$$\mathbf{s}_m = K^{-1} \cdot \mathbf{a}$$

5. $$\mathcal{E} = \left(\frac{1}{2} \frac{\partial^2 S}{\partial \mathbf{s}^2} \right)^{-1} = K^{-1}$$

$$\mathbf{s} = \mathcal{E} \cdot \mathbf{a}$$

that we call the “sea” equation

The Least Squares Method (LSM): non linear model

- We look for an initial value of the set of parameters $\mathbf{s}_0$, that is near to the minimum of S.

Step 2.

We calculate $G(\mathbf{s})$ and then, we build: $\mathbf{g}(\mathbf{s}) = \mathbf{m}(\mathbf{s}) - G(\mathbf{s}) \cdot \mathbf{s}$

Step 3.

We calculate the matrix K and the vector $\mathbf{a}$ at $\mathbf{s} = \mathbf{s}_0$:

$$K_0 = G'(\mathbf{s}_0) \cdot W \cdot G(\mathbf{s}_0)$$
$$\mathbf{a}_0 = G'(\mathbf{s}_0) \cdot W \cdot (\mathbf{d} - \mathbf{g}(\mathbf{s}_0))$$

Step 4.

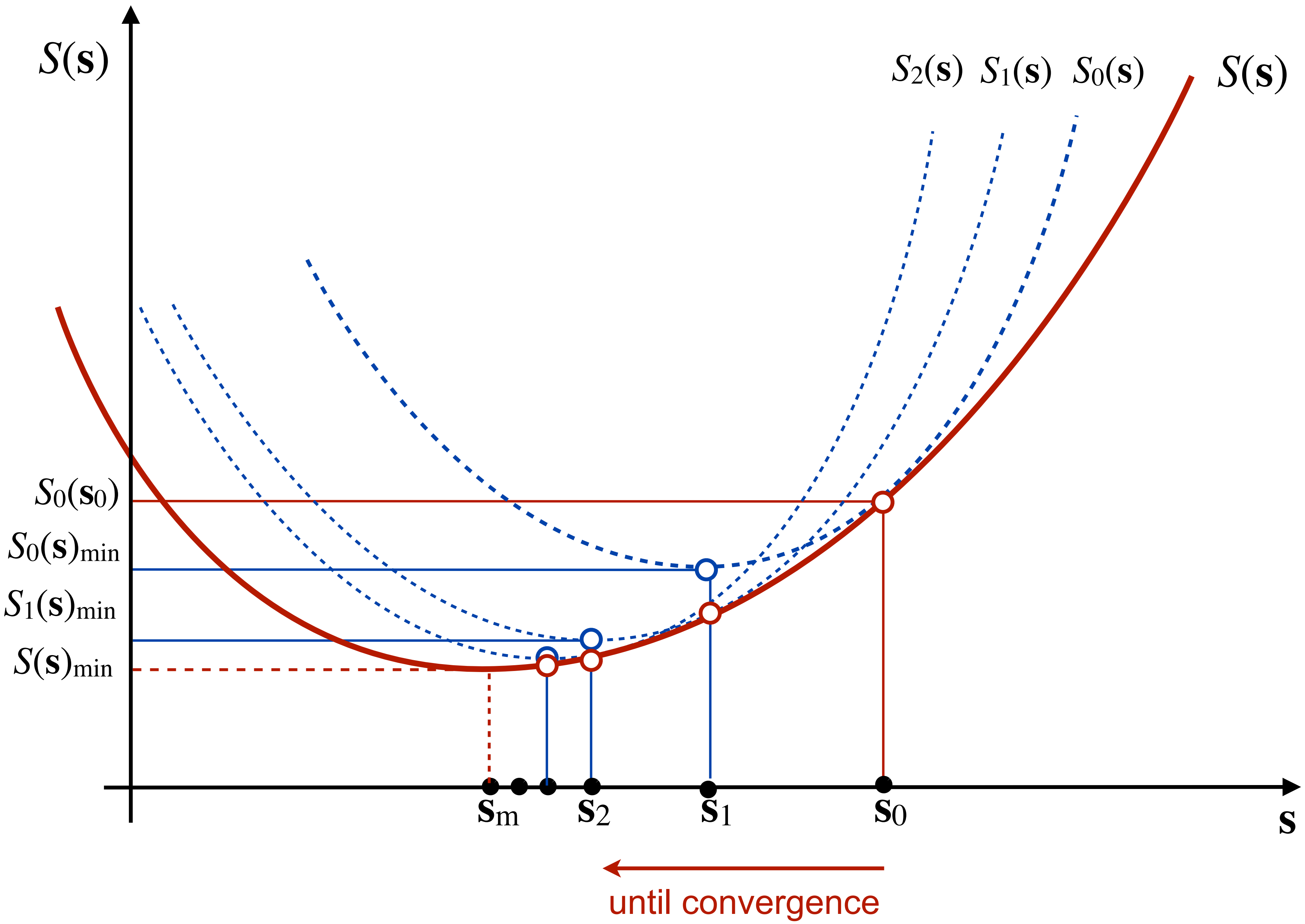
We estimate a new set of parameters, $\mathbf{s}_1$, using the equation:

$$\mathbf{s}_1 = K_0^{-1} \cdot \mathbf{a}_0$$

We repeat this step iteratively until the convergence is reached:

$$\mathbf{s}_i = K_{i-1}^{-1} \cdot \mathbf{a}_{i-1}$$

The Least Squares Method (LSM): non linear model



The Least Squares Method (LSM): non linear model, with constraints

- Let consider the set of constraint equations:

$$\mathbf{f}(\mathbf{s}) = 0 \qquad (n_c \text{ equations})$$

- We minimize iteratively the Lagrange function:

$$L(\mathbf{s}) = (\mathbf{d} - \mathbf{m}(\mathbf{s}))' \cdot W \cdot (\mathbf{d} - \mathbf{m}(\mathbf{s})) + 2 \cdot \lambda' \cdot \mathbf{f}(\mathbf{s})$$

- We introduce the jacobian matrix of the constraint functions:

$$R(\mathbf{s}) = \partial_{\mathbf{s}} \mathbf{f}(\mathbf{s})$$

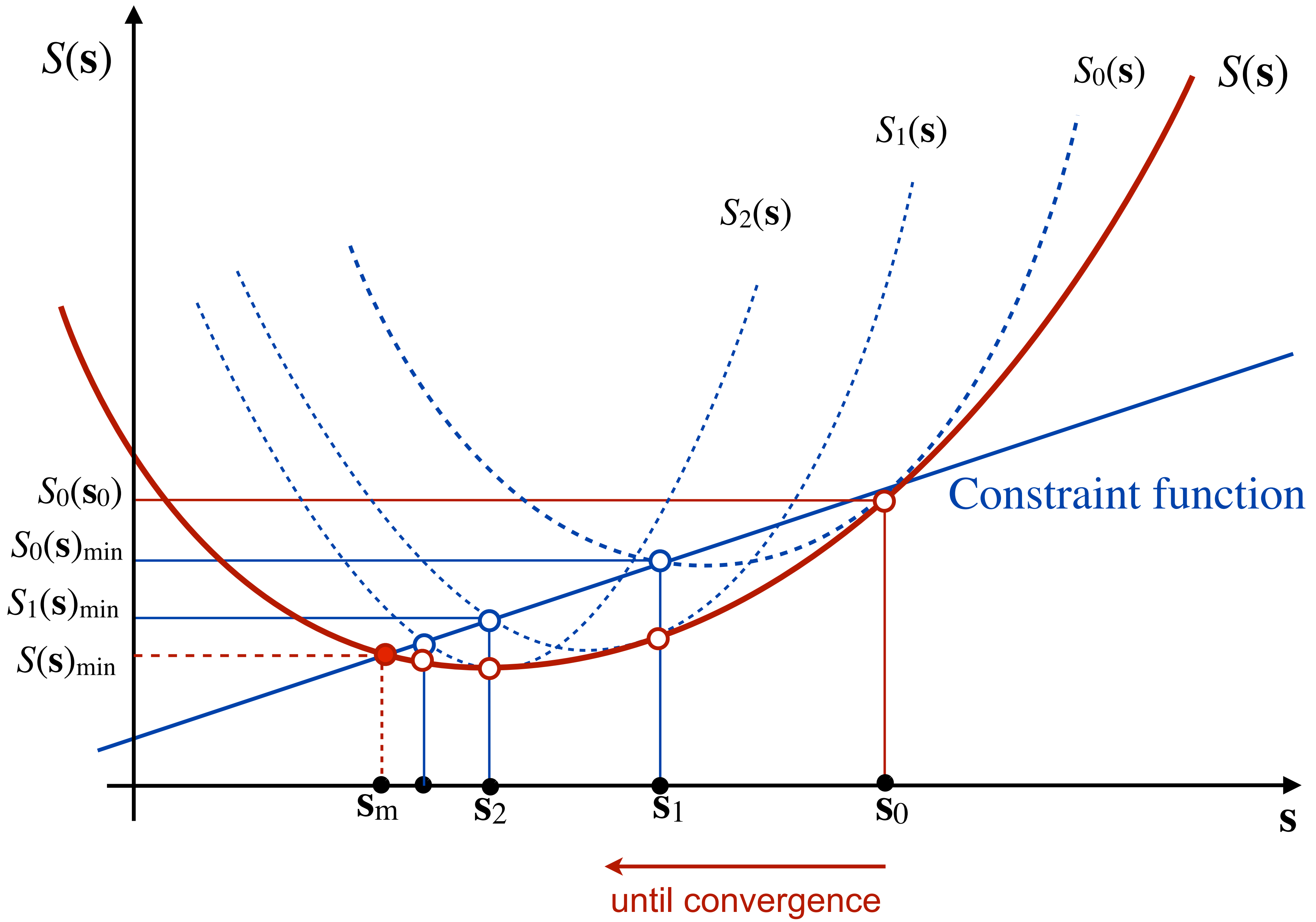
- Starting from a point $\mathbf{s} = \mathbf{s}_0$, the iterative process becomes :

$$\begin{pmatrix} \delta \mathbf{s}_1 \\ \lambda_1 \end{pmatrix} = \begin{pmatrix} K_0 & R_0' \\ R_0 & 0 \end{pmatrix}^{-1} \cdot \begin{pmatrix} \mathbf{a}_0(\mathbf{s}_0) - K_0 \cdot \mathbf{s}_0 \\ -\mathbf{f}_0 \end{pmatrix}$$

and the next-step solution is: $\mathbf{s}_1 = \mathbf{s}_0 + \delta \mathbf{s}_1$

- The iteration is repeated until convergence.

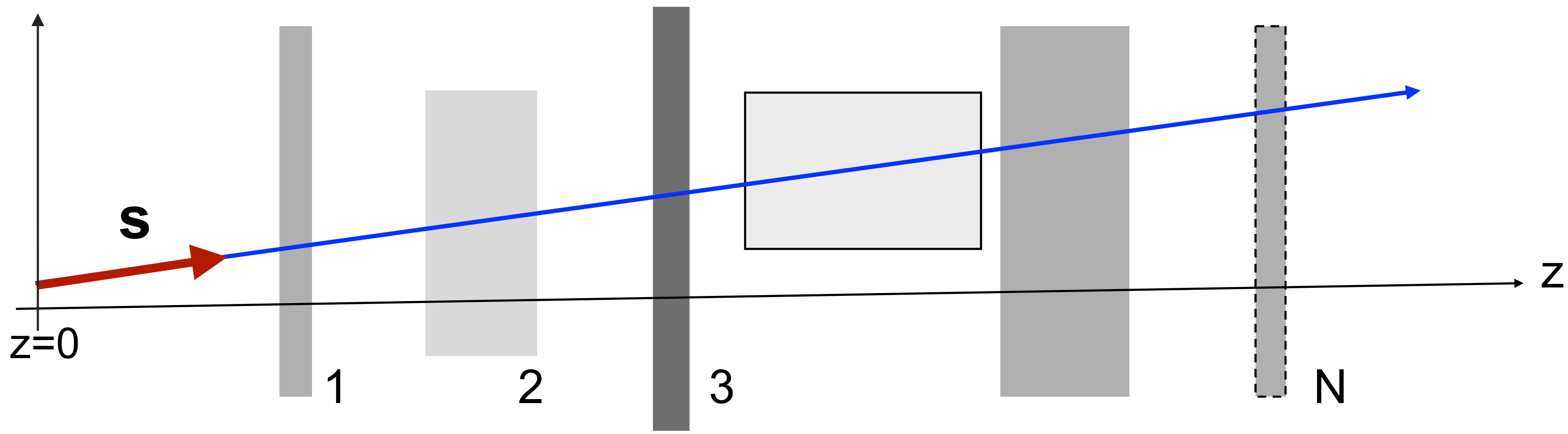
The Least Squares Method (LSM): non linear model, with constraints



TimTrack: matrix formalism

The Least Squares Method (LSM): extension to a many detectors system

If we have a set of different detectors (different models):



models:	$\mathbf{m}_1$	$\mathbf{m}_2$	$\mathbf{m}_3$	$\mathbf{m}_4$	$\mathbf{m}_5$	....
data (f.e.):	(x,y)	(x,t)	(x,y,t)	(β)	(γ)	....
K matrices:	K_1	K_2	K_3	K_4	K_5	....
a vectors:	$\mathbf{a}_1$	$\mathbf{a}_2$	$\mathbf{a}_3$	$\mathbf{a}_4$	$\mathbf{a}_5$	

and, finally:

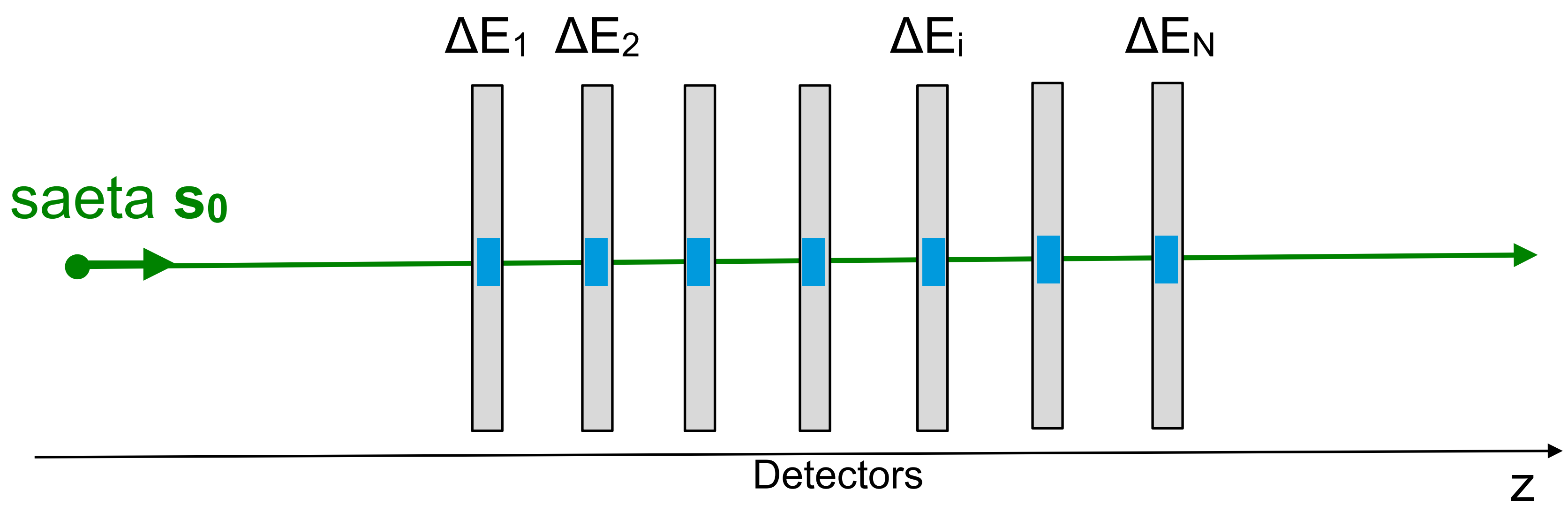
$$K = \sum K_i$$

$$\mathbf{a} = \sum \mathbf{a}_i \quad \longrightarrow \quad \boxed{\mathbf{s} = \mathcal{E} \cdot \mathbf{a}}$$

$$\mathcal{E} = K^{-1}$$

TimTrack: energy loss extension

The measured energy loss, Q= ΔE/Δx, can be related to the slowness:



The Bethe- Bloch formula can be written as funcion of the slowness S (=1/β) as:

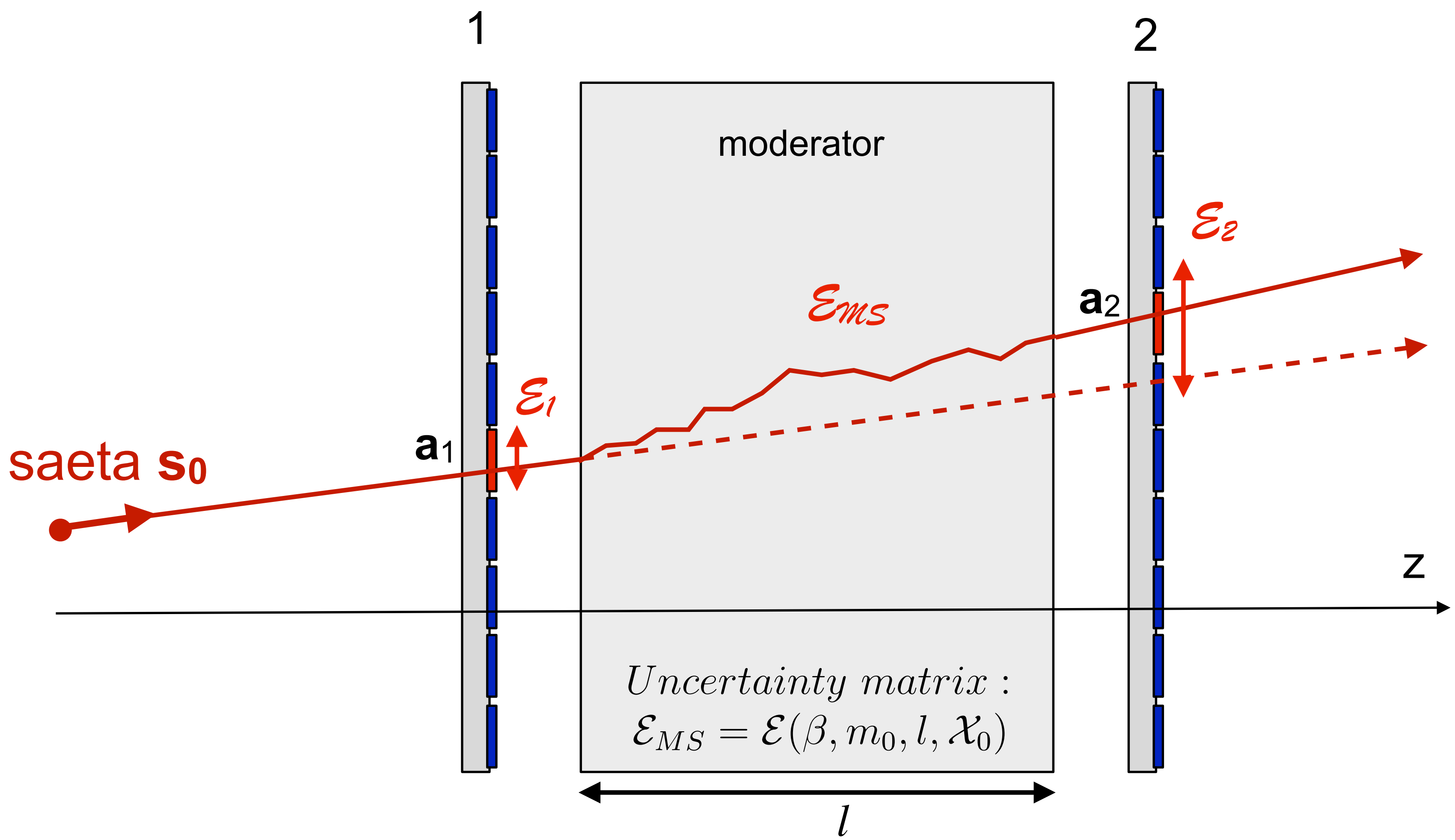
$$-\frac{dE}{dx} \simeq k \cdot S^2 \left(\frac{1}{2} \ln \frac{2m_e c^2 T_{max}}{(S^2-1)^2 I^2} - \frac{1}{S^2} - \frac{\delta(S)}{2} \right)$$

that can be approximated and simplified as:

$$-\frac{dE}{dx} \simeq k \left(S^2 \ln \frac{1}{(S^2-1) I_c} - 1 \right)$$

TimTrack: multiple scattering

Coulomb's multiple scattering



K matrices: K_1 $K_{MS} = \mathcal{E}_{MS}^{-1}$ $K_2 = \mathcal{E}_2^{-1}$

Following the TT formalism, we expect:

- $\mathbf{a}_1 = K_1\, \mathbf{s}_0$
- $\mathbf{a}_2 = K_{MS,2}\, \mathbf{s}_0$ with $K_{MS,2} = (\mathcal{E}_{MS}^{-1} + \mathcal{E}_2^{-1})^{-1}$

$\Rightarrow$ $\mathbf{s} = K_1^{-1}\, \mathbf{a}_1 + K_{MS,2}^{-1}\, \mathbf{a}_2$

Calibration parameters estimation:

The model of the data, $\mathbf{m}$, can be expanded as a function of both, the set of parameters **s** and the set of some alignment and calibration constants **α**:

$$\mathbf{m} = \mathbf{m}(\mathbf{s};\boldsymbol{\alpha})$$

and

$$\mathbf{m}(\mathbf{s};\boldsymbol{\alpha}) = \left(\frac{\partial \mathbf{m}}{\partial \mathbf{s}}\right)\mathbf{s} + \left(\frac{\partial \mathbf{m}}{\partial \boldsymbol{\alpha}}\right)\boldsymbol{\alpha} + \mathbf{g}(\mathbf{s};\boldsymbol{\alpha})$$

$$\mathbf{m}(\mathbf{s},\boldsymbol{\alpha}) = \left(\frac{\partial \mathbf{m}}{\partial \mathbf{s}} \;\; \frac{\partial \mathbf{m}}{\partial \boldsymbol{\alpha}}\right)\begin{pmatrix} \mathbf{s} \\ \boldsymbol{\alpha} \end{pmatrix} + \mathbf{g}(\mathbf{s};\boldsymbol{\alpha})$$

$$\mathbf{m}(\mathbf{s};\boldsymbol{\alpha}) = G_A \cdot \mathbf{s}_\alpha + \mathbf{g}(\mathbf{s};\boldsymbol{\alpha})$$

where: $G_A = \left(\frac{\partial \mathbf{m}}{\partial \mathbf{s}} \;\; \frac{\partial \mathbf{m}}{\partial \boldsymbol{\alpha}}\right)$ and $\mathbf{s}_\alpha = \begin{pmatrix} \mathbf{s} \\ \boldsymbol{\alpha} \end{pmatrix}$

Calibration parameters estimation:

The matrix G_A takes the form:

$$G_A = \begin{pmatrix} \frac{\partial m_1}{\partial s_1} & \dots & \frac{\partial m_1}{\partial s_i} & \dots & \frac{\partial m_1}{\partial s_n} & \frac{\partial m_1}{\partial \alpha_1} & \dots & \frac{\partial m_1}{\partial \alpha_j} & \dots & \frac{\partial m_1}{\partial \alpha_m} \\ \vdots & & \vdots & & \vdots & \vdots & & \vdots & & \vdots \\ \vdots & & \vdots & & \vdots & \vdots & & \vdots & & \vdots \\ \frac{\partial m_d}{\partial s_1} & \dots & \frac{\partial m_d}{\partial s_i} & \dots & \frac{\partial m_d}{\partial s_n} & \frac{\partial m_d}{\partial \alpha_1} & \dots & \frac{\partial m_d}{\partial \alpha_j} & \dots & \frac{\partial m_d}{\partial \alpha_m} \end{pmatrix}$$

Calibration parameters estimation:

Particles with well known parameters may be used to determine the unknown calibration constants.

$$G_A = \begin{pmatrix} \frac{\partial m_1}{\partial s_1} & \cdots & \frac{\partial m_1}{\partial s_i} & \cdots & \frac{\partial m_1}{\partial s_n} & \frac{\partial m_1}{\partial \alpha_1} & \cdots & \frac{\partial m_1}{\partial \alpha_j} & \cdots & \frac{\partial m_1}{\partial \alpha_m} \\ \vdots & & \vdots & & \vdots & \vdots & & \vdots & & \vdots \\ \vdots & & \vdots & & \vdots & \vdots & & \vdots & & \vdots \\ \frac{\partial m_d}{\partial s_1} & \cdots & \frac{\partial m_d}{\partial s_i} & \cdots & \frac{\partial m_d}{\partial s_n} & \frac{\partial m_d}{\partial \alpha_1} & \cdots & \frac{\partial m_d}{\partial \alpha_j} & \cdots & \frac{\partial m_d}{\partial \alpha_m} \end{pmatrix}$$

The columns corresponding both to the known parameters and to the known calibration constants can be eliminated from G_A before calculating the K matrix and finding the solution:

$$\mathbf{s}_\alpha = \begin{pmatrix} \mathbf{s} \\ \boldsymbol{\alpha} \end{pmatrix}$$

Calibration parameters estimation:

We develope the formalism as always:

$$K_A = G'_A \cdot W \cdot G_A$$

$$\mathbf{a}_{\boldsymbol{\alpha}} = G'_A \cdot W \cdot (\mathbf{d} - \mathbf{g}(\mathbf{s};\boldsymbol{\alpha}))$$

and

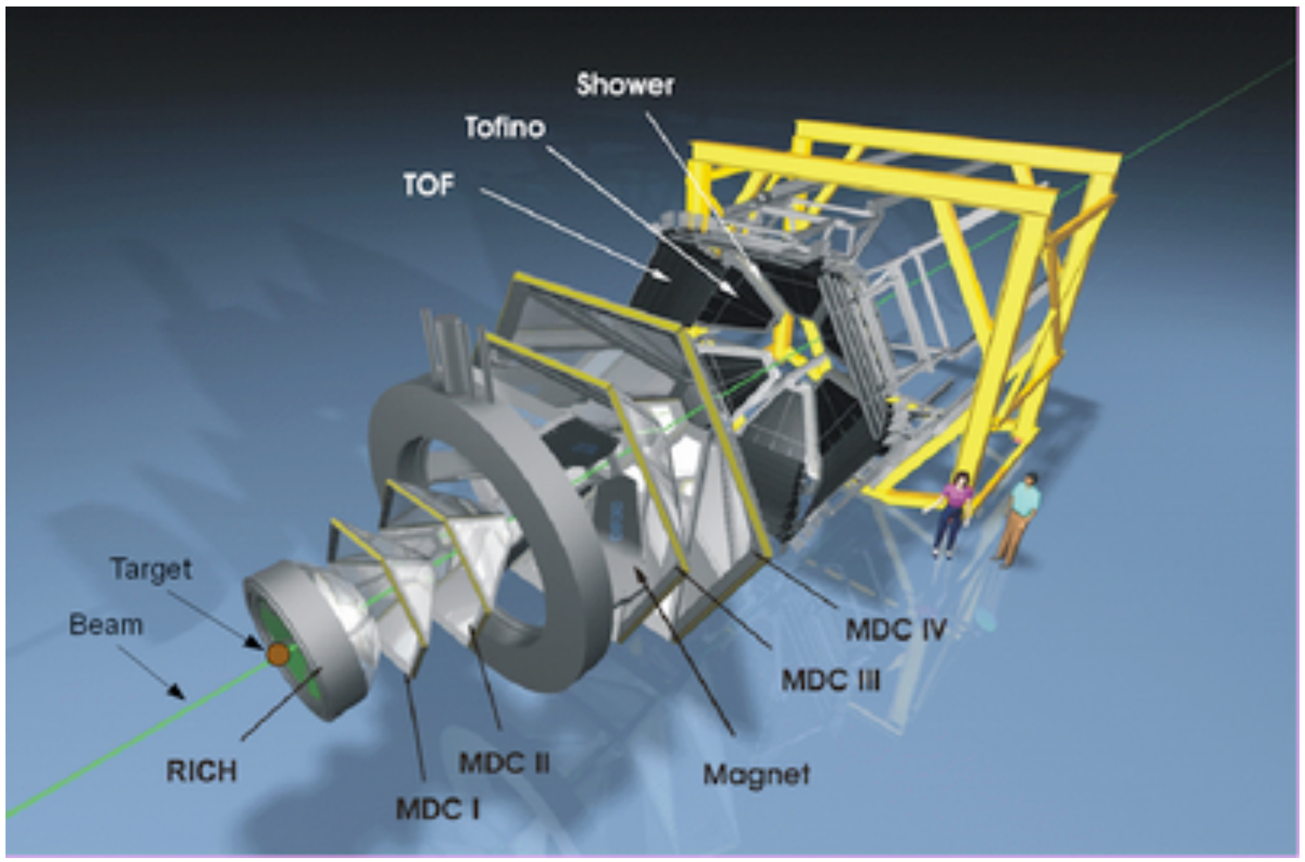
$$\boxed{\mathbf{s}_{\boldsymbol{\alpha}} = K_A^{-1} \cdot \mathbf{a}_{\boldsymbol{\alpha}}}$$

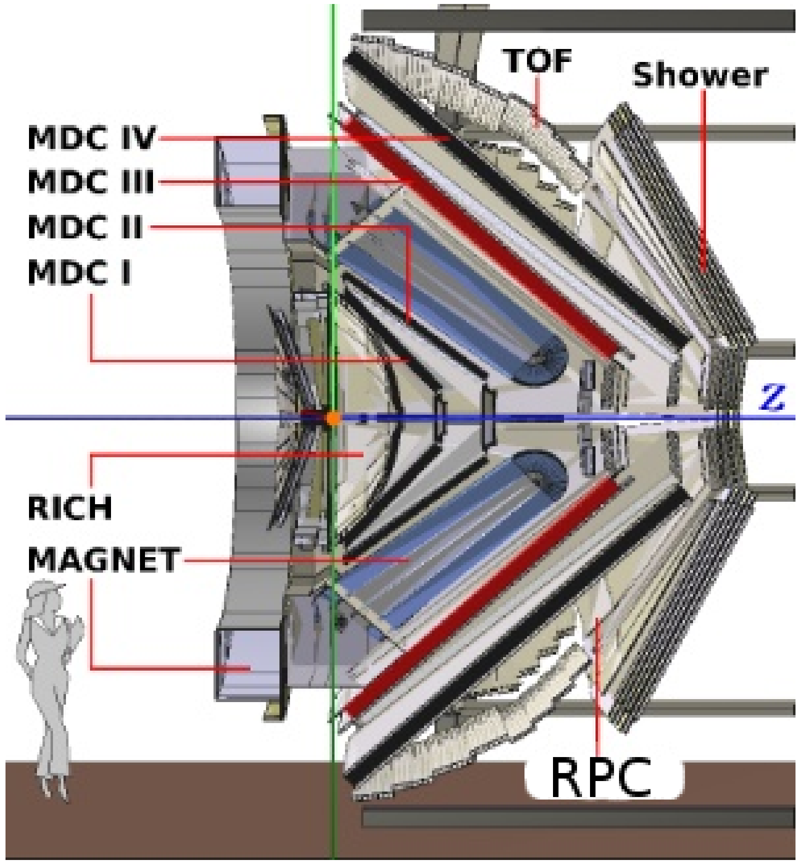
where:

$$\mathbf{s}_{\alpha} = \begin{pmatrix} \mathbf{s} \\ \boldsymbol{\alpha} \end{pmatrix}$$

TimTrack: examples

HADES MDCs (Mini Drift Chambers)



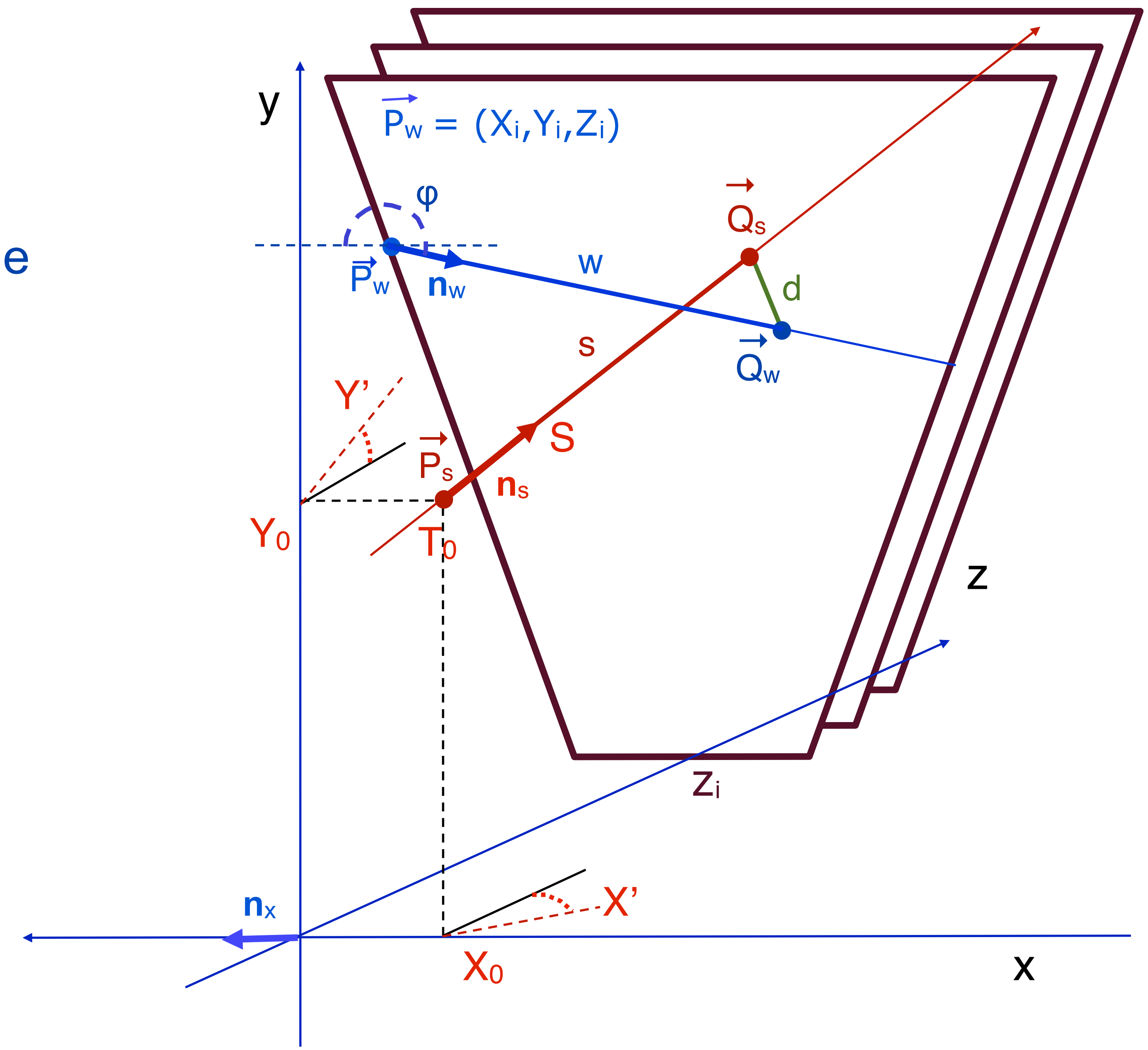


TimTrack: examples

HADES MDCs (Mini Drift Chambers)

In a drift chamber, each wire gives:

1. The wire coordinate
2. A time = TOF + drif time + pulse time
3. dE/dx



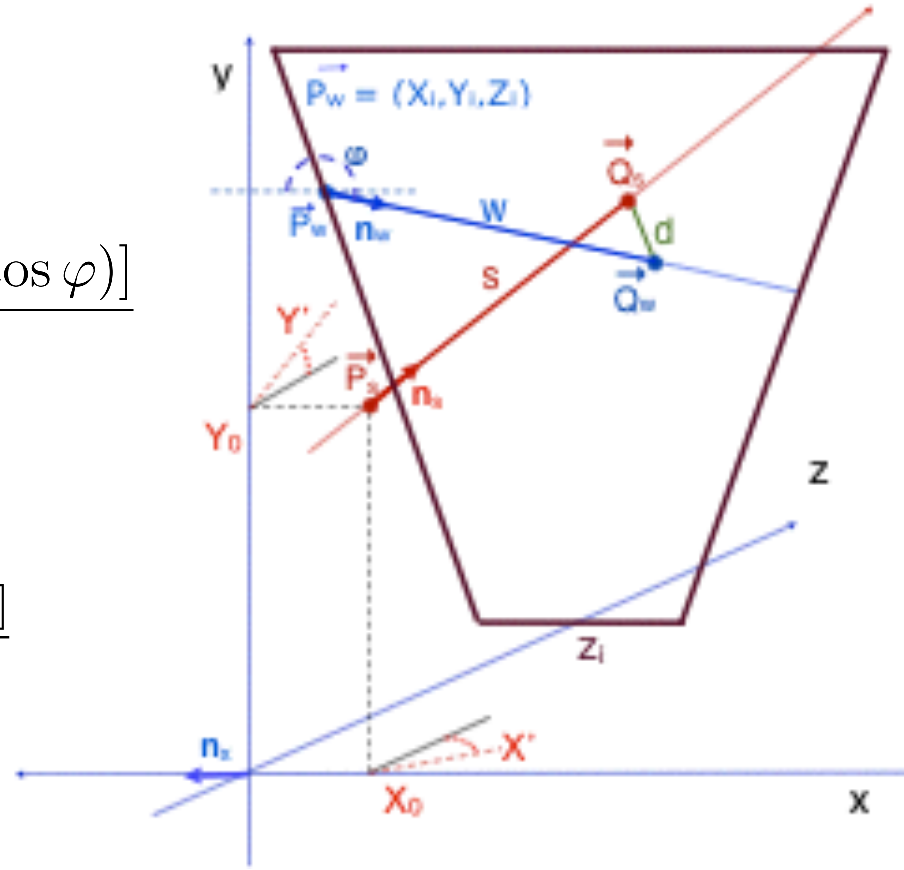
HADES MDCs (Mini Drift Chambers)

Measured time model:

$$s = \frac{z_i \cdot \sqrt{X'^2 + Y'^2 + 1} \cdot [1 - (-X' \sin\varphi + Y' \cos\varphi)(-X'_i \sin\varphi + Y'_i \cos\varphi)]}{1 + (-X' \sin\varphi + Y' \cos\varphi)^2}$$

$$d = \frac{z_i \cdot [-(X' + X'_i) \sin\varphi + (Y' + Y'_i) \cos\varphi]}{\sqrt{1 + (-X' \sin\varphi + Y' \cos\varphi)^2}}$$

$$w = \frac{z_i \cdot [-(X' + X'_i) \cos\varphi + (Y' + Y'_i) \sin\varphi - (-X' \sin\varphi + Y' \cos\varphi)(X' Y'_i - X'_i Y')]}{1 + (-X' \sin\varphi + Y' \cos\varphi)^2}$$



where:

$$X'_i = \frac{X_0 - X_i}{z_i} \qquad Y'_i = \frac{Y_0 - Y_i}{z_i}$$

Finally, the time t measured in a wire is:

$$t = T_0 + \frac{s}{V} + \frac{d}{v_d} + \frac{w}{v_w}$$

$$t = T_0 + s \cdot S + d \cdot s_d + w \cdot s_w$$

TimTrack: examples

HADES MDCs (Mini Drift Chambers)

Runge Kutta method

TimTrack method

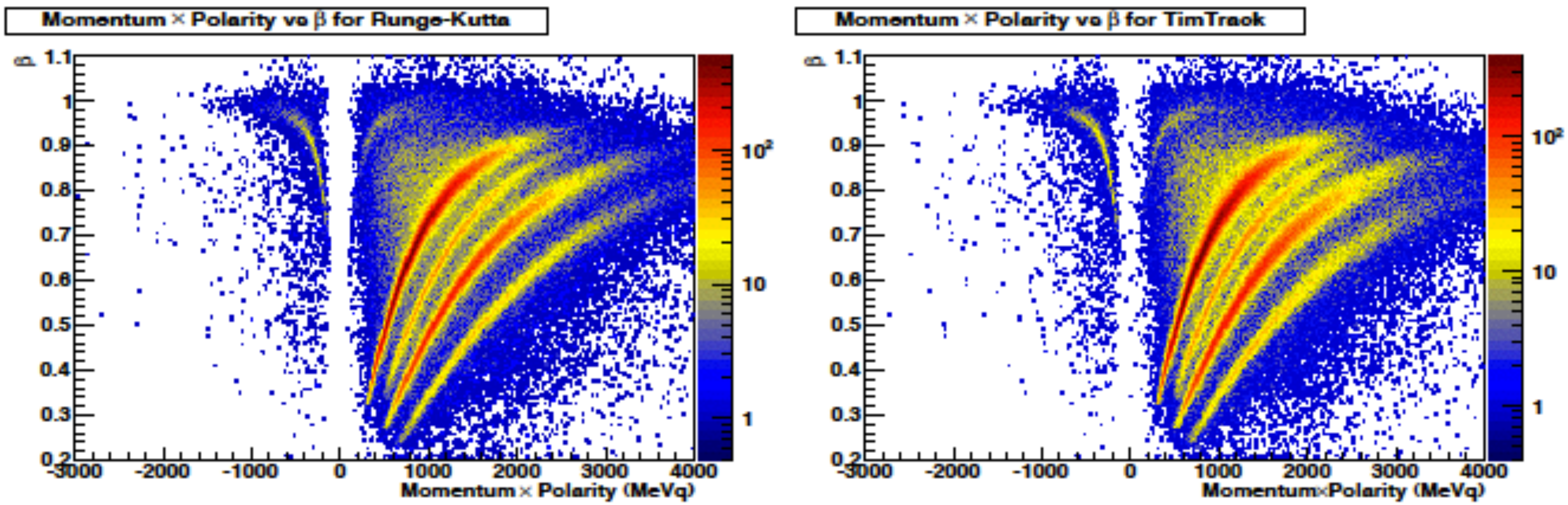
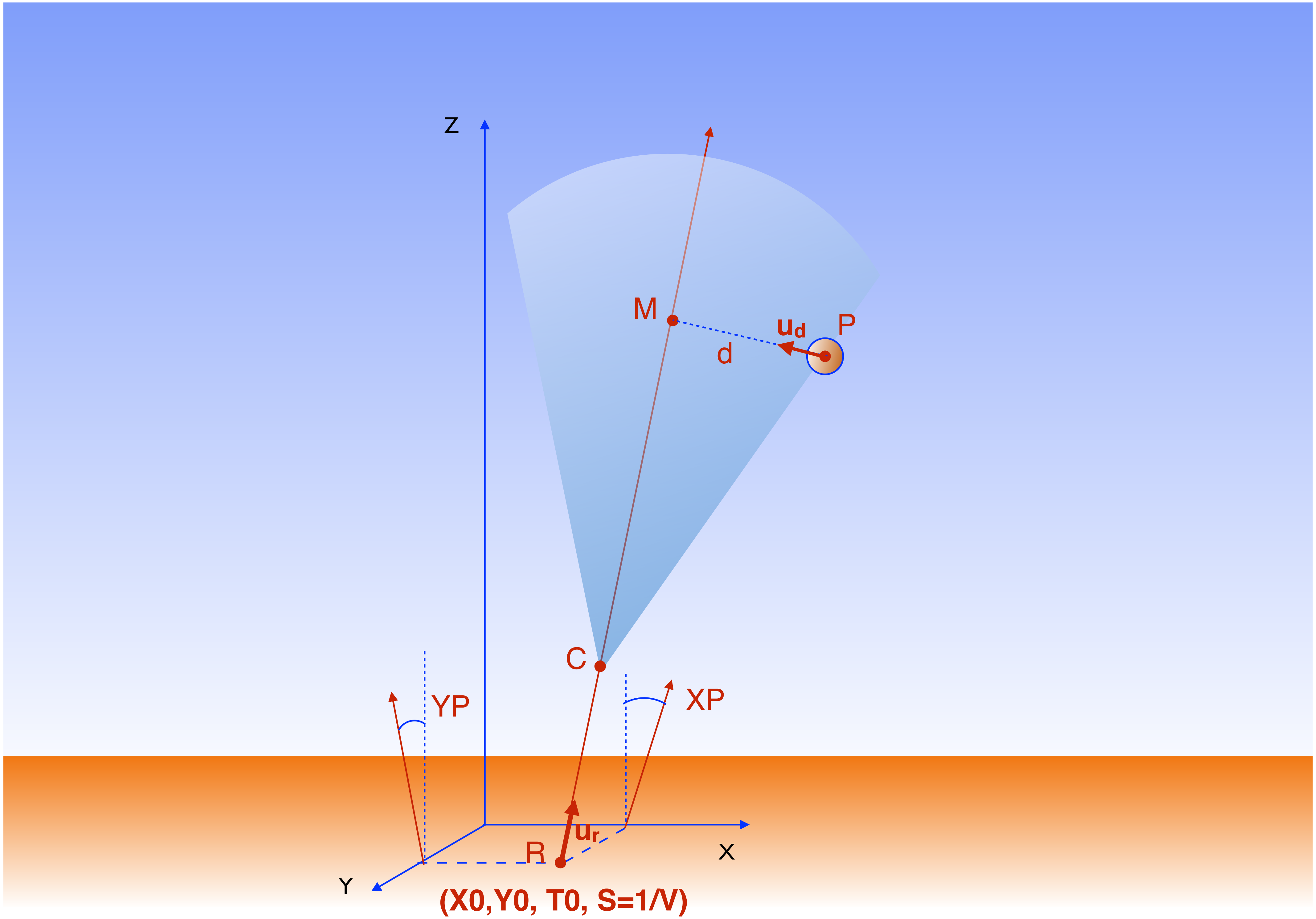


Figure 5.18: Comparative between distributions of β vs momentum for TimTrack and Runge-Kutta.

[Ph.D. thesis of Georgy Kornakov. U.S. Compostela 2012]

TimTrack: examples

ANTARES Observatory



TimTrack vs Kalman Filter

TimTrack		
Parameter space		Measurement space
$\mathbf{s}_p, \mathcal{E}_p$		
	F: Transport	
$\mathbf{s} = F \cdot \mathbf{s}_p$ $\mathcal{E}_s = F \cdot \mathcal{E}_p \cdot F'$		$\mathbf{d}$, $W_d = V_d^{-1}$
	G: Measurement $\mathbf{m(s)} = G \cdot \mathbf{s} + \mathbf{g(s)}$	$V_s = G \cdot \mathcal{E}_s \cdot G'$
		$\mathbf{d}_c = \mathbf{d} - \mathbf{g(s)}$ $V_c = V_d + V_s$
$\mathbf{s}_d = C \cdot \mathbf{d}_c$ $\delta\mathcal{E}_s = \mathcal{E}_s \cdot W_s \cdot \mathcal{E}_s$	$C = \mathcal{E}_s \cdot G' \cdot V_c^{-1}$ $W_s = G' \cdot V_c^{-1} \cdot G$	
$\mathbf{s}_{p+1} = (I - \mathcal{E}_s \cdot W_s) \cdot (\mathbf{s} + \mathbf{s}_d)$ $\mathcal{E}_{p+1} = \mathcal{E}_s - \delta\mathcal{E}_s$		

Kalman Filter		
Parameter space		Measurement space
$\mathbf{r}_p, \mathcal{E}_p$		
	F: Transport	
$\mathbf{r} = F \cdot \mathbf{r}_p$ $\mathcal{E}_r = F \cdot \mathcal{E}_p \cdot F'$		$\mathbf{d}$, V_d
	H: Measurement $\mathbf{m(r)} = H \cdot \mathbf{r} + \boldsymbol{\eta}$	$\mathbf{d}_r = H \cdot \mathbf{r}$ $V_r = H \cdot \mathcal{E}_r \cdot H'$
		$\delta\mathbf{d} = \mathbf{d} - \mathbf{d}_r$ $V_c = V_r + V_d$
$\delta\mathbf{r} = K \cdot \delta\mathbf{d}$ $\delta\mathcal{E}_r = \mathcal{E}_r \cdot W_r \cdot \mathcal{E}_r$	$K = \mathcal{E}_r \cdot H' \cdot V_c^{-1}$ $W_r = H' \cdot V_c^{-1} \cdot H$	
$\mathbf{r}_{p+1} = \mathbf{r} + \delta\mathbf{r}$ $\mathcal{E}_{p+1} = \mathcal{E}_r - \delta\mathcal{E}_r$		

TimTrack: summary

- The TimTrack algorithm offers a suggestive framework for the tracking of charged particles with timing detectors and in complex systems

- TT provides, together with coordinates and slopes, the velocity of the particle and the arrival time at a reference plane.

- TT works directly with the primary data and has a simple matrix form making it easy and fast to implement (sometimes, some extra numerical work is needed: non-conditionated matrices!!!)

- TT deals with linear and non-linear models, including constraints among the parameters

- TT may be used for the simultaneus fit of several particles with common constraints

References (NIM):

J.A.Garzón et al: TimTrack: A new concept for the tracking of charged particles with timing detectors

J.A.Garzón et al TimTrack: A matrix formalism for a fast time and track reconstruction with timing detectors

Two new articles are on the way: “TimTrack with constraints” and “TimTrack filtering”

The end