Status of the identification procedures for INDRA-Vamos experimentsn e494s and e503

Guilain ADEMARD et Nicolas LE NEINDRE

Tuesday, 16th October 2012

Checking of the status of the ID procedures

- **Grids :** KVIDGChIoSi (CI-SI and CI-CSI), KVTGID (Tassan-Got's fits for SI-CSI), KVIDGCsI (CSI)
- **Identification classes :** KVIDChIoSiCorr (CI-SI), KVIDChIoCorrCsI (CI-CSI), KVIDSiCorrCsI (SI-CSI), KVIDCsI (CSI)
- **ID codes** (see doc of KVINDRACodeMask)
- **quality codes** (see doc of KVIDZAGrid : :GetQualityCode())

CI-SI ID telescope

e494s and e503

Class : KVIDChIoSiCorr

IDMapX : $PG - pedestal_{PG}$ in Si, where PG is calculated from GG if GG<3900

IDMapY : $PG - pedestal_{PG}$ in ChIo, where PG is calculated from GG if GG<3900

CI-SI grids

e494s

File : grilles_ChIo_Si.dat

Grid : KVIDGChIoSi_e494s

- "Seuil_ChIo"

e503

File :

Grid_ChIoSi_C4-9_R322-606.dat

Grid : KVIDGChIoSi

- "Punch_through"
- "Seuil_Si"
- "Bragg_line"
- "Emax_Si"

DONE :

- Simplifying the idtelescope KVIDChIoSiCorr to resolve conflicts due to the overridings of data/function mumbers
- Adding a new idgrid KVIDGChIoSi_e494s in order to have a new cut line and a new appropriated statut code for the identification
- Correcting bad ID-grid class name in file of e494s

TO BE DONE :

- Draw the "punch_through" line in the grids
- Use the same grid class in e494s and e503 (just change the class name in the ascii file)
  $\Rightarrow$ Draw the "Seuil_ChIo" line in the grids of e503

CI-SI grid quality code

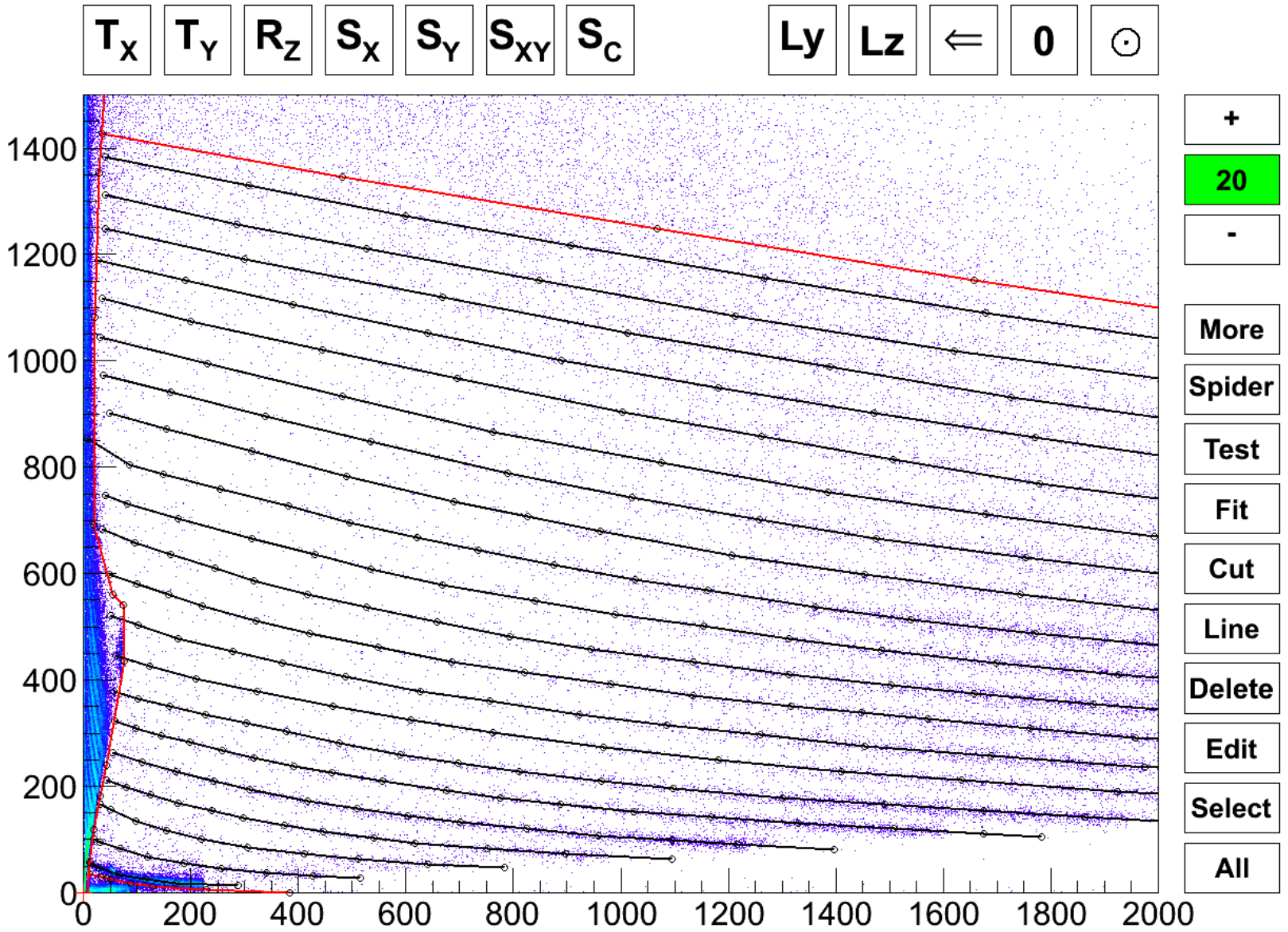
class KVIDGChIoSi_e494s: public KVIDGChIoSi

KVIDGChIoSi_e494s

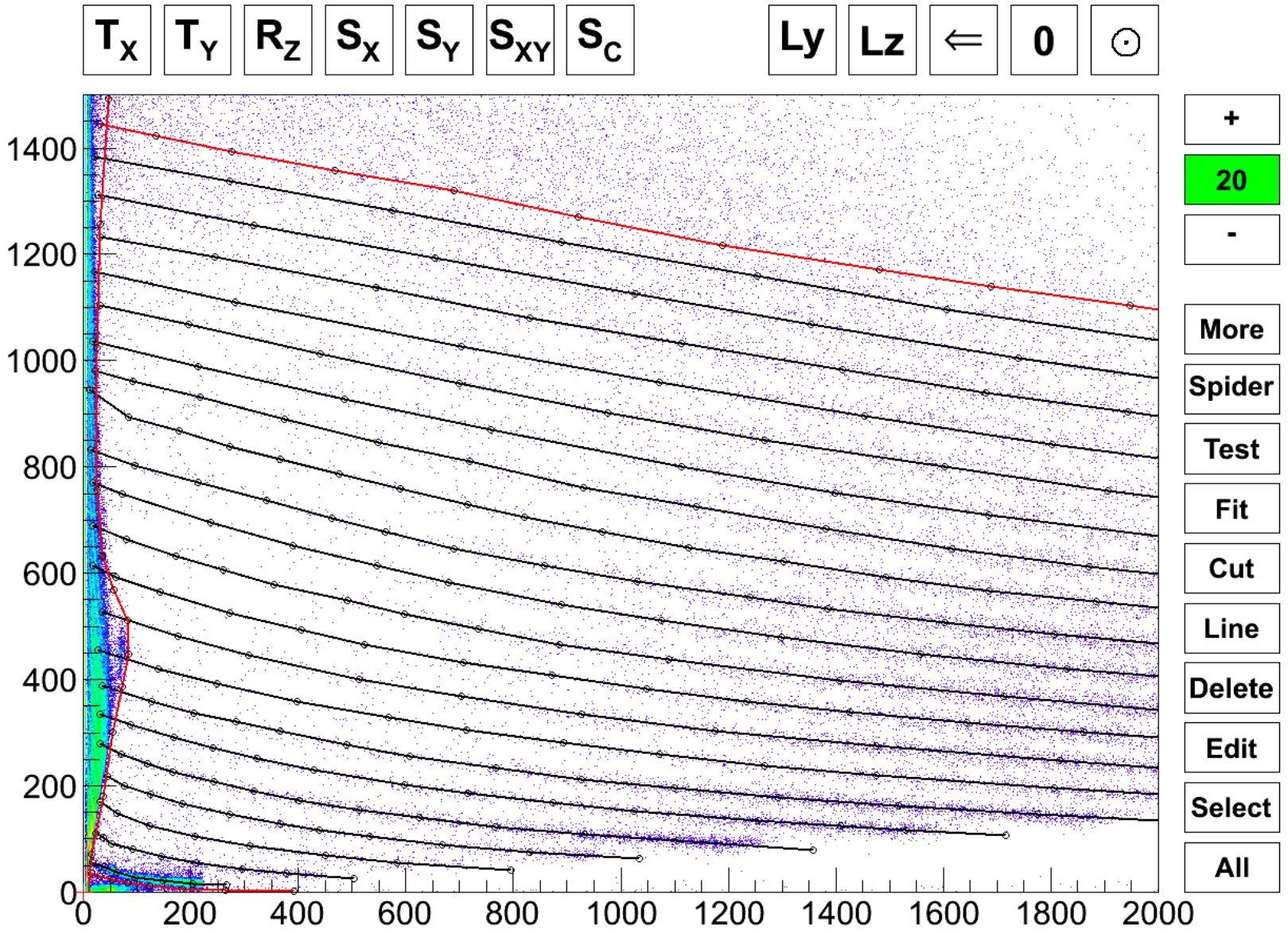
Specific identification grid for e494s

```
Z-only identification grids for INDRA Ionisation chamber-Silicon telescopes.
To identify a particle with this grid, do:
    if( grid->IsIdentifiable(x,y) ){
          grid->Identify(x,y,nuc);
    }

After attempting identification with method Identify, GetQualityCode() method returns
one of the following status codes:

KVIDZAGrid::kICODE0,                  OK
KVIDZAGrid::kICODE1,                  slight ambiguity of Z, which could be larger
KVIDZAGrid::kICODE2,                  slight ambiguity of Z, which could be smaller
KVIDZAGrid::kICODE3,                  slight ambiguity of Z, which could be larger or smaller
KVIDZAGrid::kICODE4,                  point is in between two lines of different Z, too far from either to be considered well-identified
KVIDZAGrid::kICODE5,                  point is in between two lines of different Z, too far from either to be considered well-identified
KVIDZAGrid::kICODE6,                  (x,y) is below first line in grid
KVIDZAGrid::kICODE7,                  (x,y) is above last line in grid
KVIDZAGrid::kICODE8,                  no identification: (x,y) out of range covered by grid
KVIDGChIoSi::k_BelowPunchThrough   "warning: point below punch-through line",
KVIDGChIoSi::k_BelowSeuilSi      "point to identify left of Si threshold line (bruit/arret ChIo?)",
KVIDGChIoSi::k_LeftOfBragg       "point to identify below Bragg curve. Z given is a Zmin",
KVIDGChIoSi::k_RightOfEmaxSi        "point to identify has E_Si > Emax_Si i.e. codeur is saturated. Unidentifiable",
KVIDGChIoSi_e494s::k_BelowSeuilChIo         "point to identify was below of "ChIo threshold" line,

Correctly-identified particles with no ambiguity have quality codes < KVIDZAGrid::kICODE4.
Particles with quality code KVIDGChIoSi::k_BelowPunchThrough are also in principle correctly identified,
but we give a warning that the point is below the line 'Punch_through' delimiting the region in which the
identification lines can be mixed with particles punching through the silicon if no additional condition
is placed on e.g. the CsI detector behind the silicon not firing.
Particles with quality code KVIDGChIoSi::k_LeftOfBragg are below the Bragg curve and the Z attributed
is a minimum value.
Particles with code KVIDZAGrid::kICODE7 are (far) above the last line of the grid, their Z is also a minimum.
```

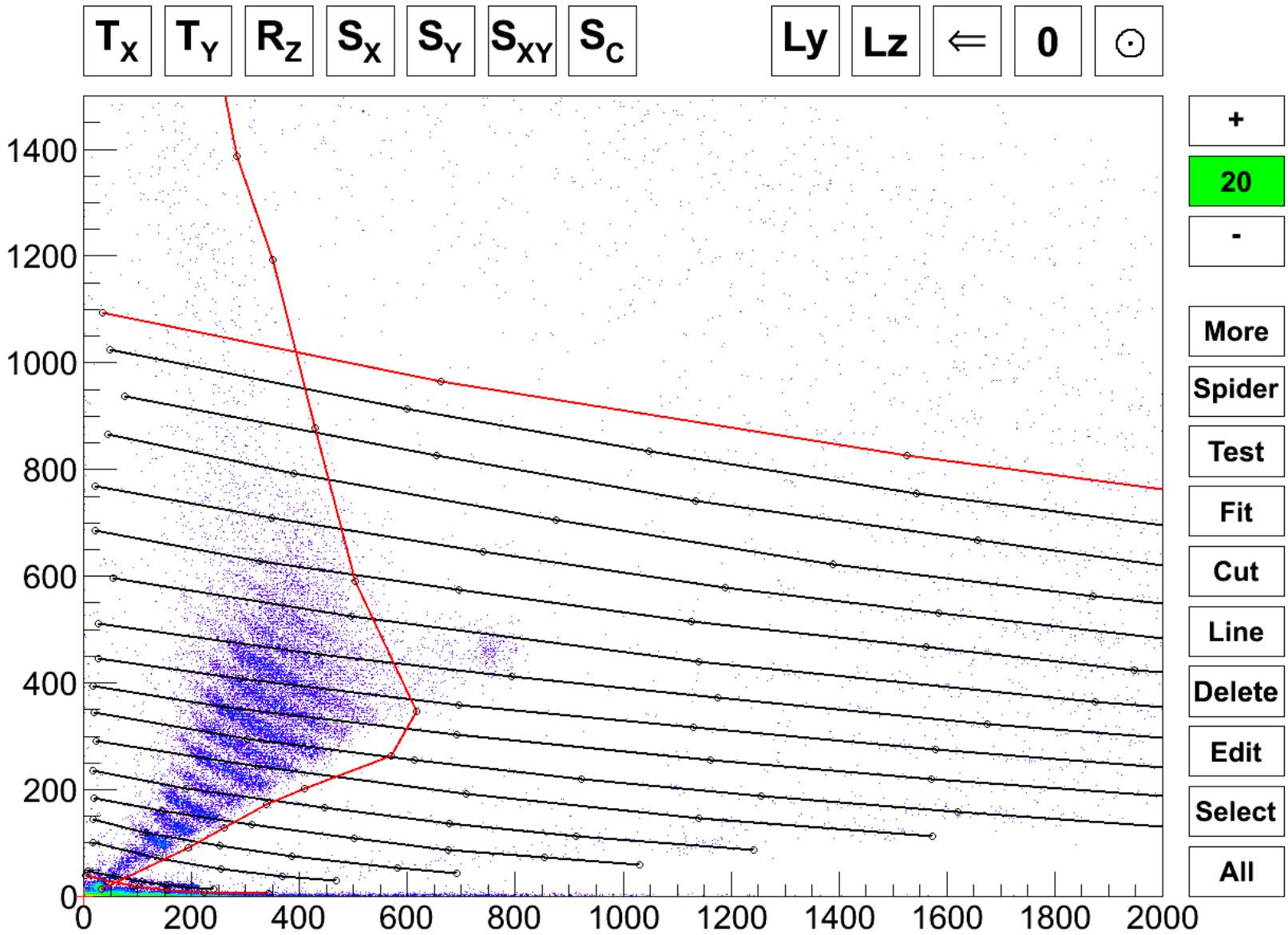
Example : CI-SI C06M03, runs 80-532



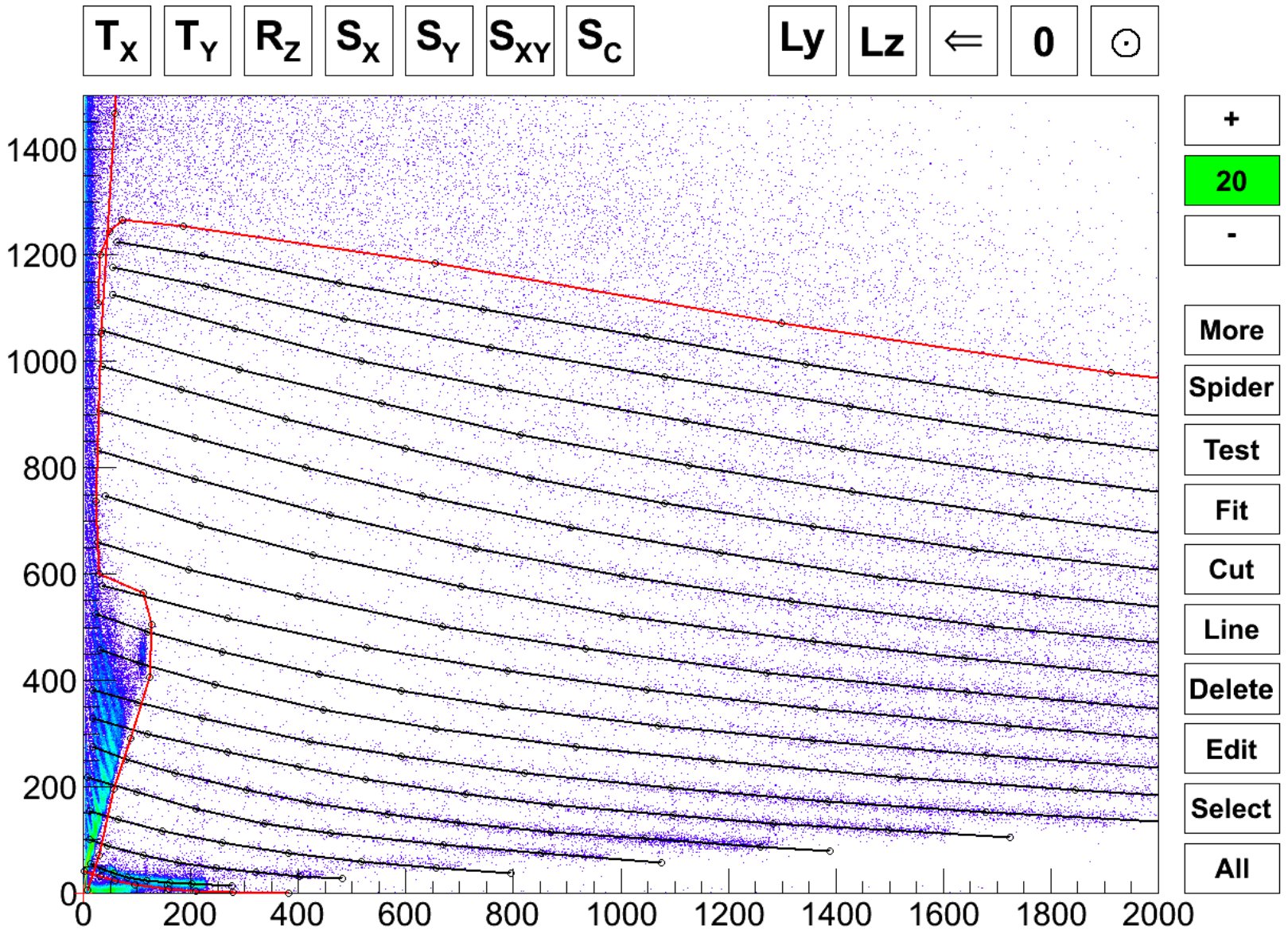
Example : CI-SI C06M04, runs 80-532



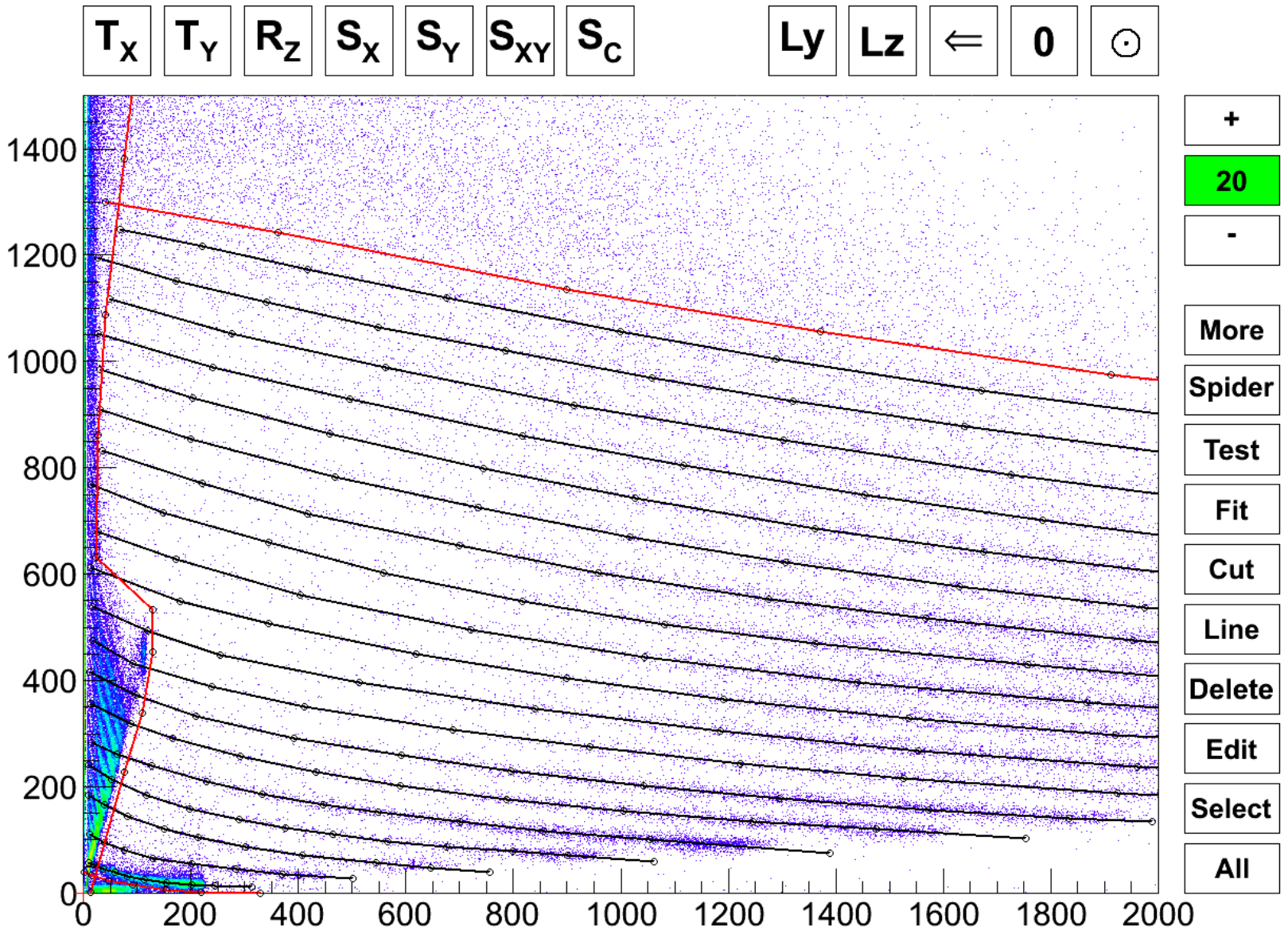
Example : CI-SI C06M15, runs 80-532



Example : CI-SI C07M03, runs 80-532



Example : CI-SI C07M04, runs 80-532



CI-CSI ID telescope

e494s

Class : KVIDChIoCorrCsI_e494s
IDMapX : CsI total light
IDMapY : $PG - pedestal_{PG}$ in ChIo, where PG is calculated from GG if GG<3900. The pedestal GG is set to zero in order to not cut the physics as the pedestal GG seems to be too high

e503

Class : KVIDChIoCorrCsI
IDMapX : CsI total light
IDMapY : $PG - pedestal_{PG}$ in ChIo, where PG is the low gain acquisition parameter

CI-CSI grids

e494s

File : grilles_ChIo_CsI.dat

Grid : KVIDGChIoSi_e494s

- "Seuil_ChIo"

e503

File : Grid_ChIoCsI_C10-17_R322-606.dat

Grid : KVIDGChIoSi

- "Punch_through"
- "Seuil_Si"
- "Bragg_line"
- "Emax_Si"

DONE :

- Correcting bad ID-grid class name in file of e494s

CI-CSI grids

TO BE DONE : same thing as for (CI-SI)

- Draw the "punch_through" line in the grids
- Use the same grid class in e494s and e503 (just change the class name in the ascii file)
  $\Rightarrow$ Draw the "Seuil_ChIo" line in the grids of e503

CI-CSI grid quality code, same as CI-SI

class KVIDGChIoSi_e494s: public KVIDGChIoSi

KVIDGChIoSi_e494s

Specific identification grid for e494s

```
Z-only identification grids for INDRA Ionisation chamber-Silicon telescopes.
To identify a particle with this grid, do:
    if( grid->IsIdentifiable(x,y) ){
          grid->Identify(x,y,nuc);
    }

After attempting identification with method Identify, GetQualityCode() method returns
one of the following status codes:

KVIDZAGrid::kICODE0,                  OK
KVIDZAGrid::kICODE1,                  slight ambiguity of Z, which could be larger
KVIDZAGrid::kICODE2,                  slight ambiguity of Z, which could be smaller
KVIDZAGrid::kICODE3,                  slight ambiguity of Z, which could be larger or smaller
KVIDZAGrid::kICODE4,                  point is in between two lines of different Z, too far from either to be considered well-identified
KVIDZAGrid::kICODE5,                  point is in between two lines of different Z, too far from either to be considered well-identified
KVIDZAGrid::kICODE6,                  (x,y) is below first line in grid
KVIDZAGrid::kICODE7,                  (x,y) is above last line in grid
KVIDZAGrid::kICODE8,                  no identification: (x,y) out of range covered by grid
KVIDGChIoSi::k_BelowPunchThrough   "warning: point below punch-through line",
KVIDGChIoSi::k_BelowSeuilSi      "point to identify left of Si threshold line (bruit/arret ChIo?)",
KVIDGChIoSi::k_LeftOfBragg       "point to identify below Bragg curve. Z given is a Zmin",
KVIDGChIoSi::k_RightOfEmaxSi         "point to identify has E_Si > Emax_Si i.e. codeur is saturated. Unidentifiable",
KVIDGChIoSi_e494s::k_BelowSeuilChIo         "point to identify was below of "ChIo threshold" line,

Correctly-identified particles with no ambiguity have quality codes < KVIDZAGrid::kICODE4.
Particles with quality code KVIDGChIoSi::k_BelowPunchThrough are also in principle correctly identified,
but we give a warning that the point is below the line 'Punch_through' delimiting the region in which the
identification lines can be mixed with particles punching through the silicon if no additional condition
is placed on e.g. the CsI detector behind the silicon not firing.
Particles with quality code KVIDGChIoSi::k_LeftOfBragg are below the Bragg curve and the Z attributed
is a minimum value.
Particles with code KVIDZAGrid::kICODE7 are (far) above the last line of the grid, their Z is also a minimum.
```

CI-SI example

SI-CsI ID telescope

e494s and e503

Class : KVIDSiCorrCsI

IDMapX : CsI total light

IDMapY : $PG - pedestal_{PG}$ in Si, where PG is calculated from GG if GG<3900

SI-CSI grids

e494s

File : grilles_fit_Si_CsI.dat
Fit : KVTGID

- No ID cut lines
- No A_{min}-A_{max}
- Run validity range not taken into account

e503

File : SI_CSI_C4-9_autofit.dat
Fit : KVTGID

- idem e494s

DONE :

- Initializing IDcode and IDquality in the method KVIDSiCorrCsI : :Identify(...)

TO BE DONE :

- Add a validity range for A-identification
- Take into account the run validity range of the fits. IN PROGRESS

SI-CSI grid quality code

class KVTGIDManager

```
KVTGIDManager

Handles a set of Tassan-Got functional-based identifications
(KVTGID objects)  for use by a KVIDTelescope.

IdentZ and IdentA are, respectively, Z and A identification procedures.
Their purpose is to retrieve the correct KVTGID identification object
from the manager's list, and then use KVTGID::GetIdentification.

Status code for last identification with IdentZ and IdentA can be
obtained using GetStatus/GetStatusString.

 0 KVTGIDManager::kStatus_OK            "ok",
 1 KVTGIDManager::kStatus_noTGID        "no KVTGID for requested identification",
 2 KVTGIDManager::kStatus_OutOfIDRange  "point to identify outside of identification range of function",
 3 KVTGIDManager::kStatus_ZtooSmall     "IdentA called with Z<1",
 4 KVTGIDManager::kStatus_ZtooLarge     "IdentA called with Z larger than max Z defined for KVTGIDZA isotopic identification object"
```

SI-CSI example

CSI ID telescope

e494s and e503

Class : KVIDCsI

IDMapX : Low component of CsI

IDMapY : Fast component of CsI

CSI grids

e494s

File : grilles_CsI_RL_e494s.dat
Grid : KVIDGCsI

- "gamma_line"
- "IMF_line" not considered by KVIDGCsI

e503

File : IDGridManager_E503_CSI_RL.dat
Grid : KVIDGCsI

- idem e494s

TO BE DONE :

- Extend the "IMF_line" up to the lower signal (R-L)
  IN PROGRESS (Nicolas)

CSI grid quality code

class KVIDGCsI: public KVIDZAGrid

KVIDGCsI

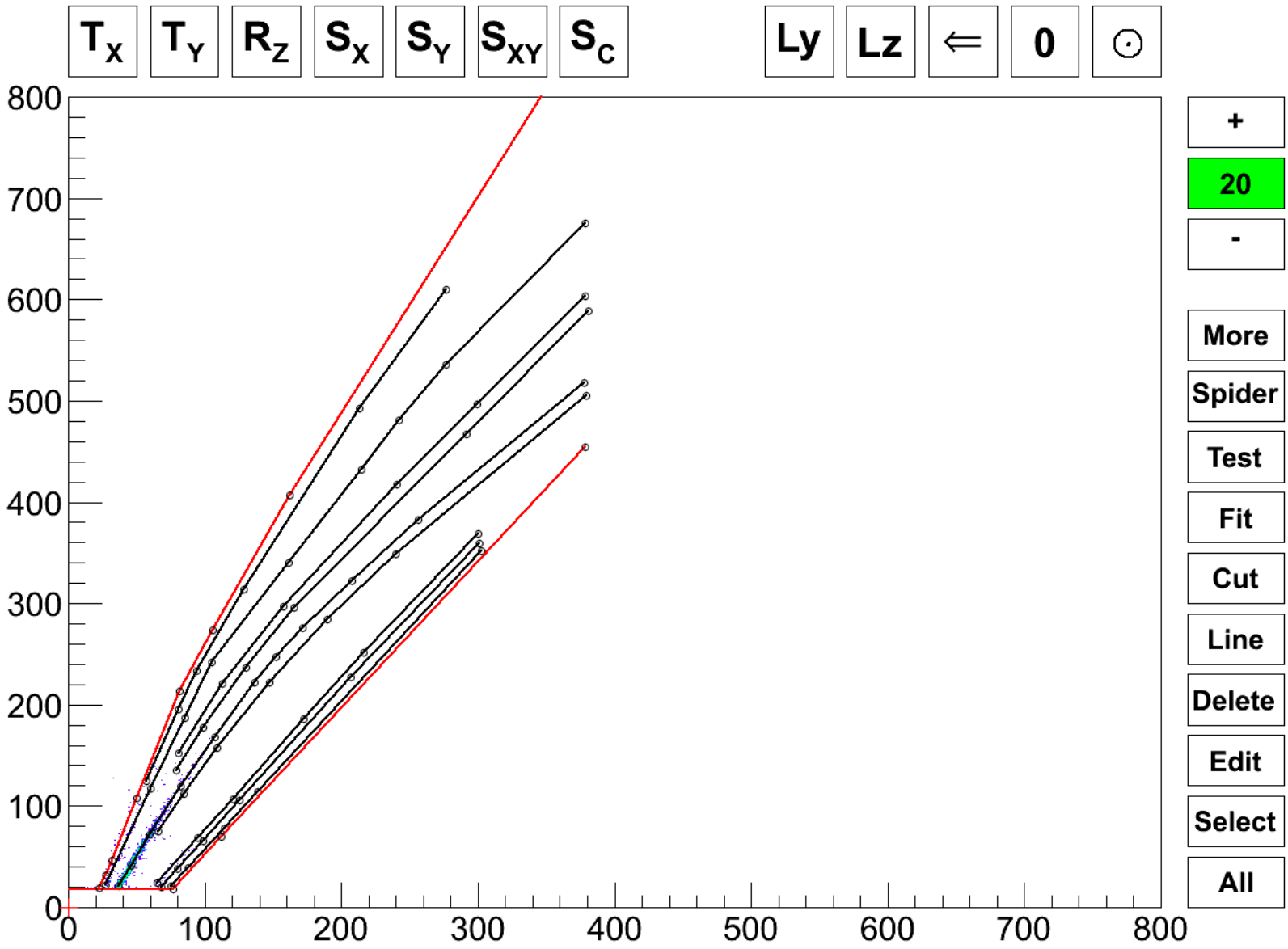
Identification grids for CsI R-L matrices.

The identification procedure is supposed to be identical to that of the FORTRAN algorithm IdnCsOr developed by Laurent Tassan-Got and used by the INDRA collaboration since 1993.

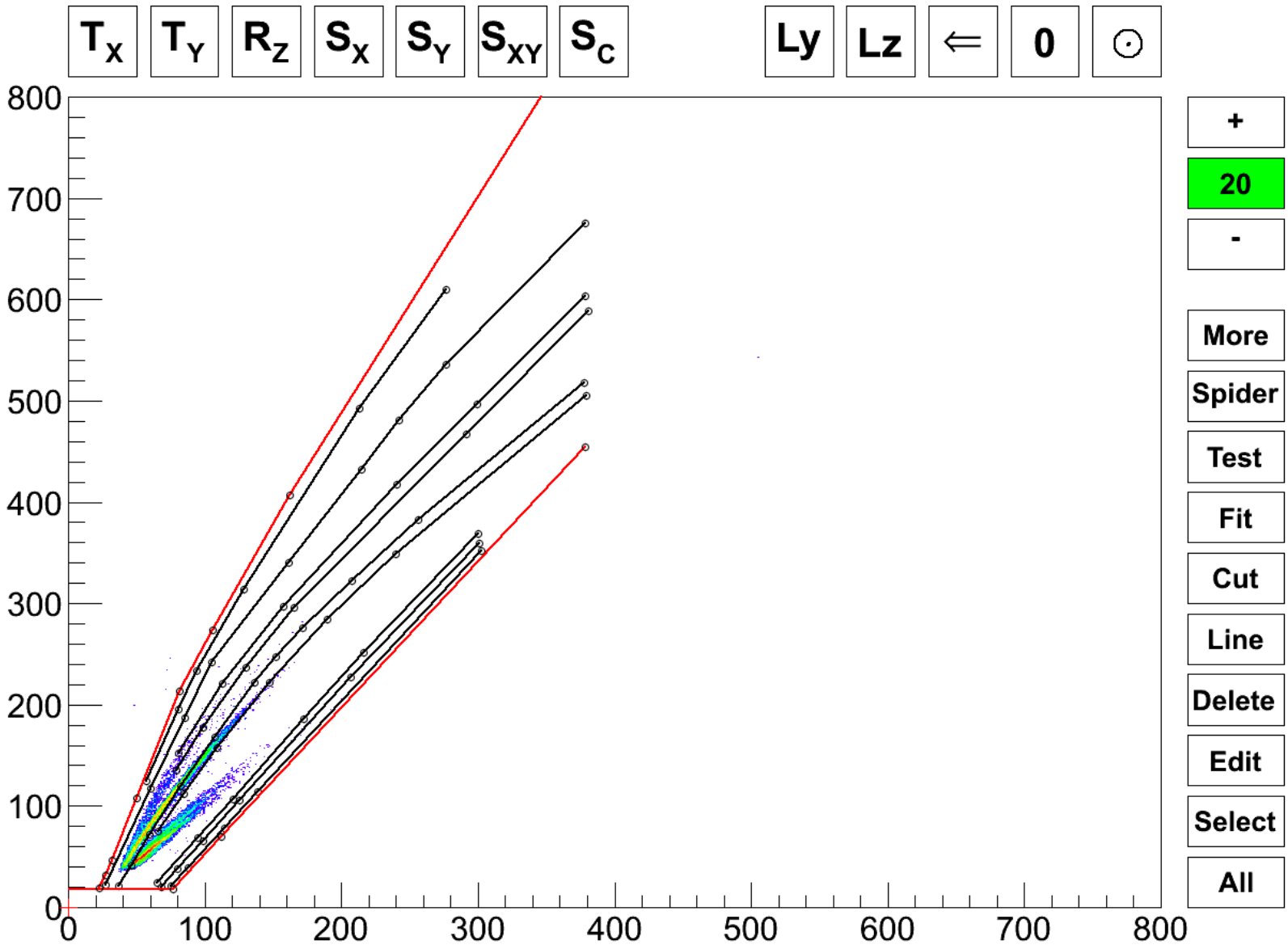
The status codes returned by GetQualityCode() are the same as IdnCsOr, with the addition of kICODE10 for identification of gammas:

```
KVIDGCsI::kICODE0  ok
KVIDGCsI::kICODE1  Z ok, mais les masses superieures a A sont possibles
KVIDGCsI::kICODE2  Z ok, mais les masses inferieures a A sont possibles
KVIDGCsI::kICODE3  Z ok, mais les masses superieures ou inferieures a A sont possibles
KVIDGCsI::kICODE4  Z ok, masse hors limite superieure ou egale a A
KVIDGCsI::kICODE5  Z ok, masse hors limite inferieure ou egale a A
KVIDGCsI::kICODE6  au-dessus de la ligne fragment, Z est alors un Zmin
KVIDGCsI::kICODE7  a gauche de la ligne fragment, Z est alors un Zmin et le plus probable
KVIDGCsI::kICODE8  Z indetermine ou (x,y) hors limites
KVIDGCsI::kICODE9  pas de lignes pour ce module
KVIDGCsI::kICODE10 gamma
```

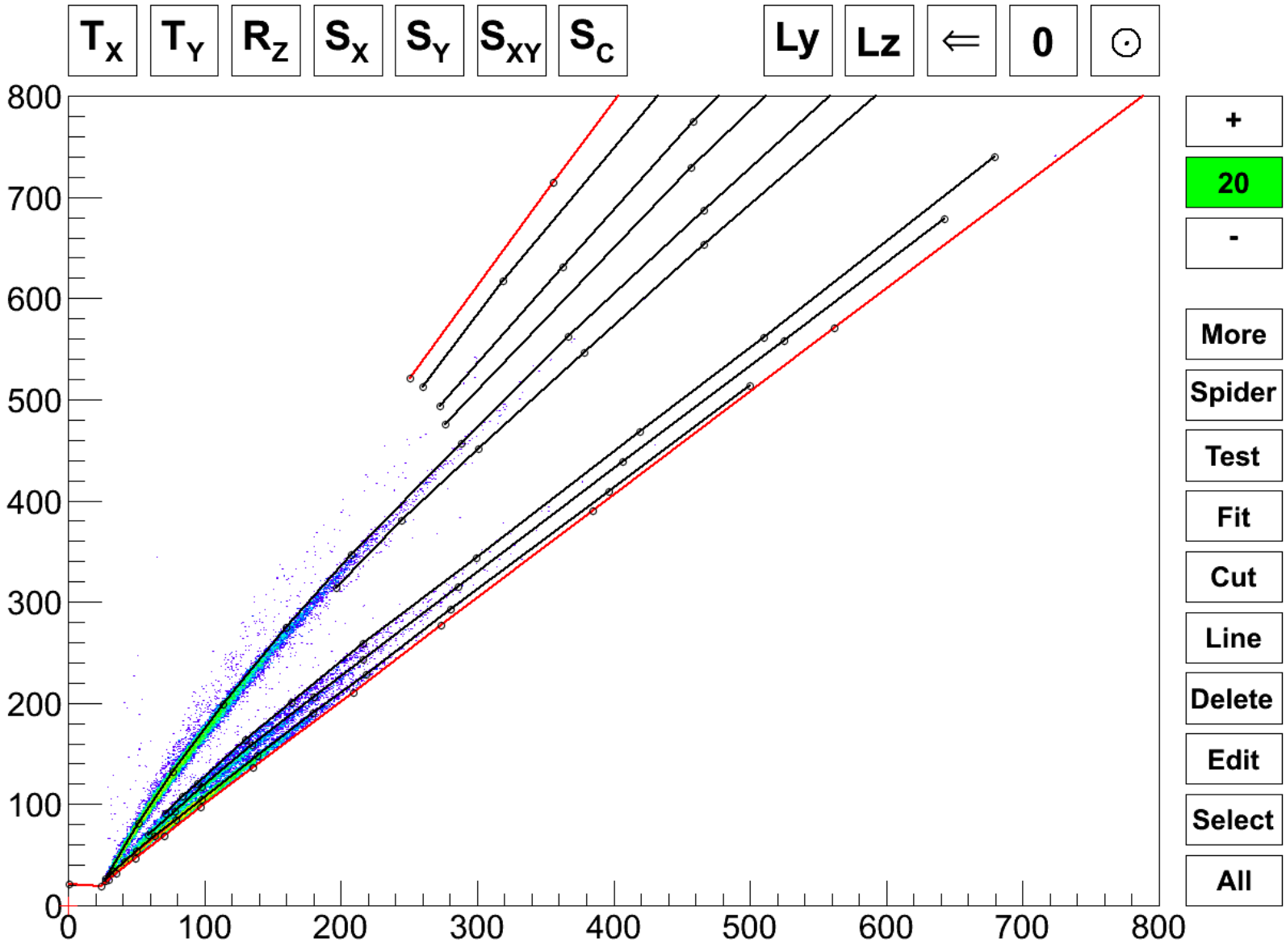
Example : CSI C04M03, runs 80-150



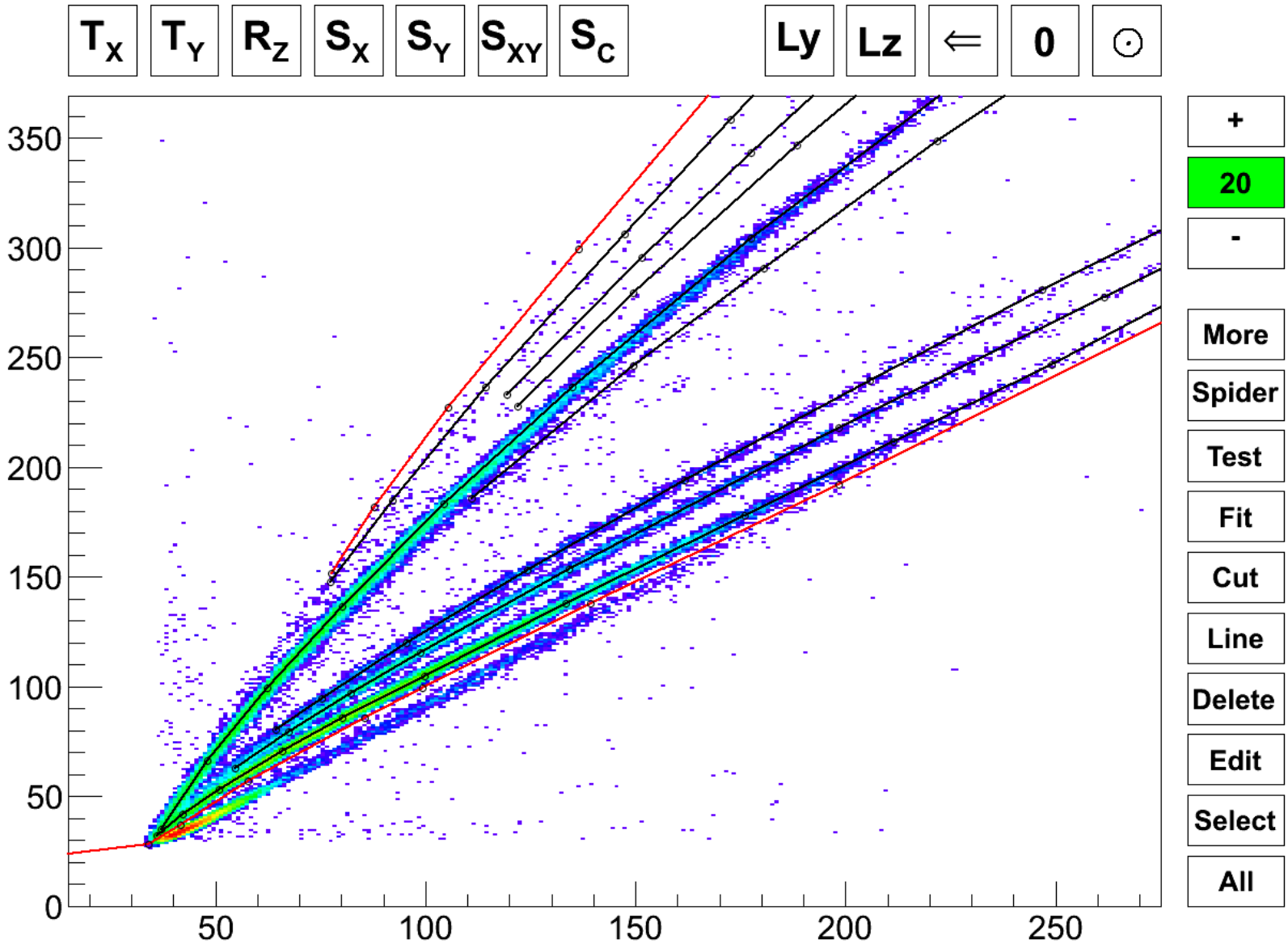
Example : CSI C04M03, runs 460-530



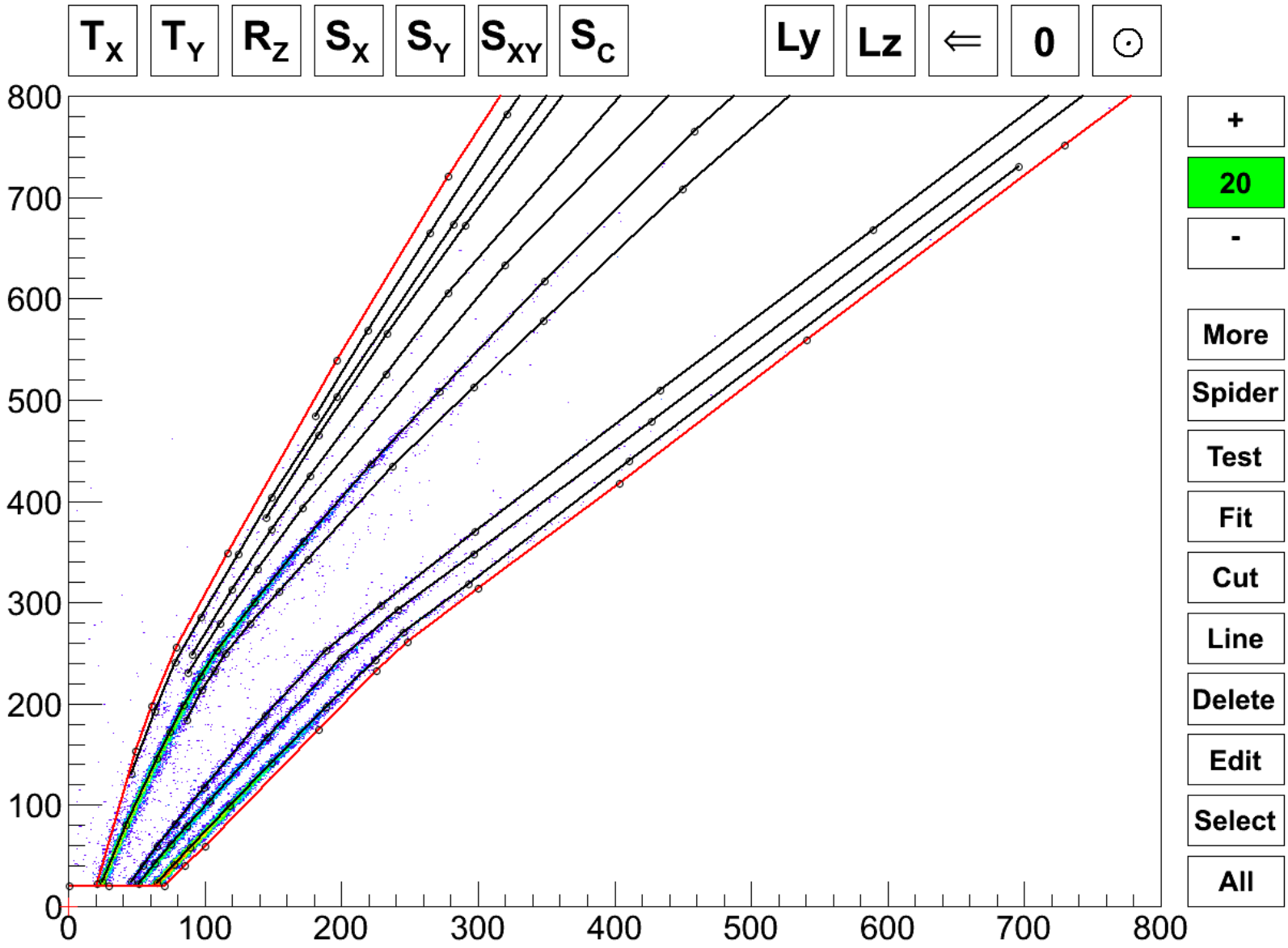
Example : CSI C08M24, runs 80-150



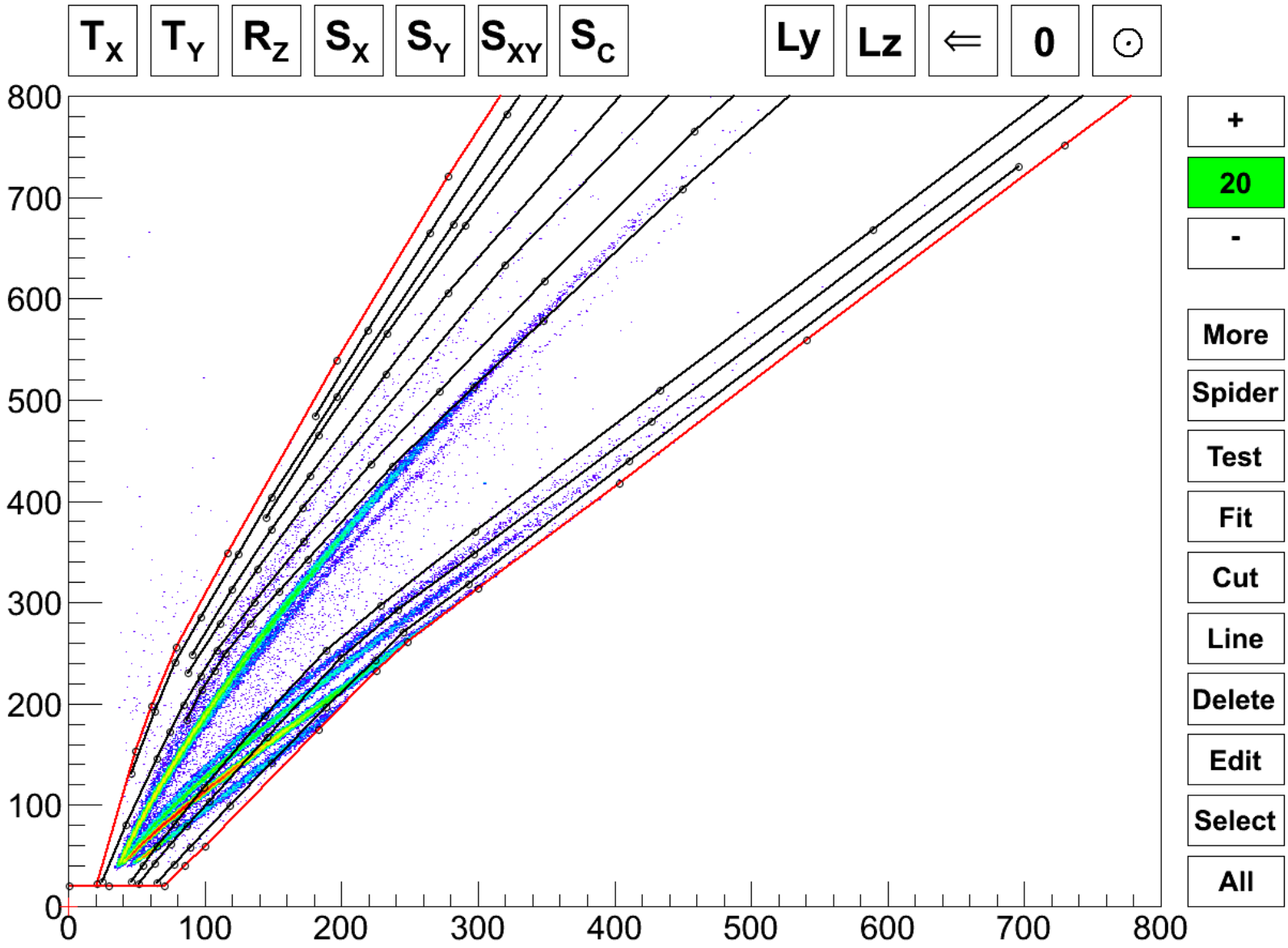
Example : CSI C09M07, runs 80-150



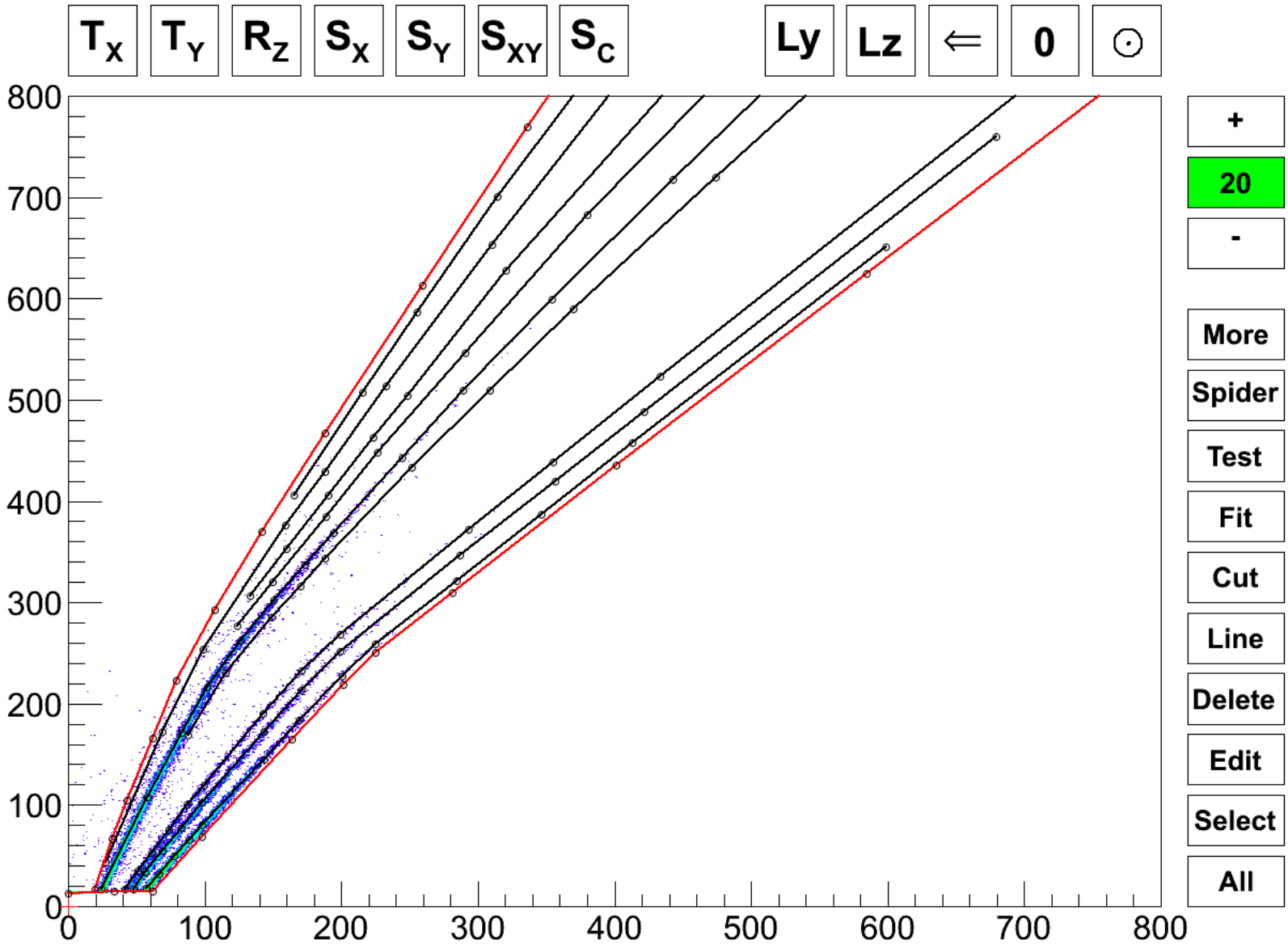
Example : CSI C09M08, runs 80-150



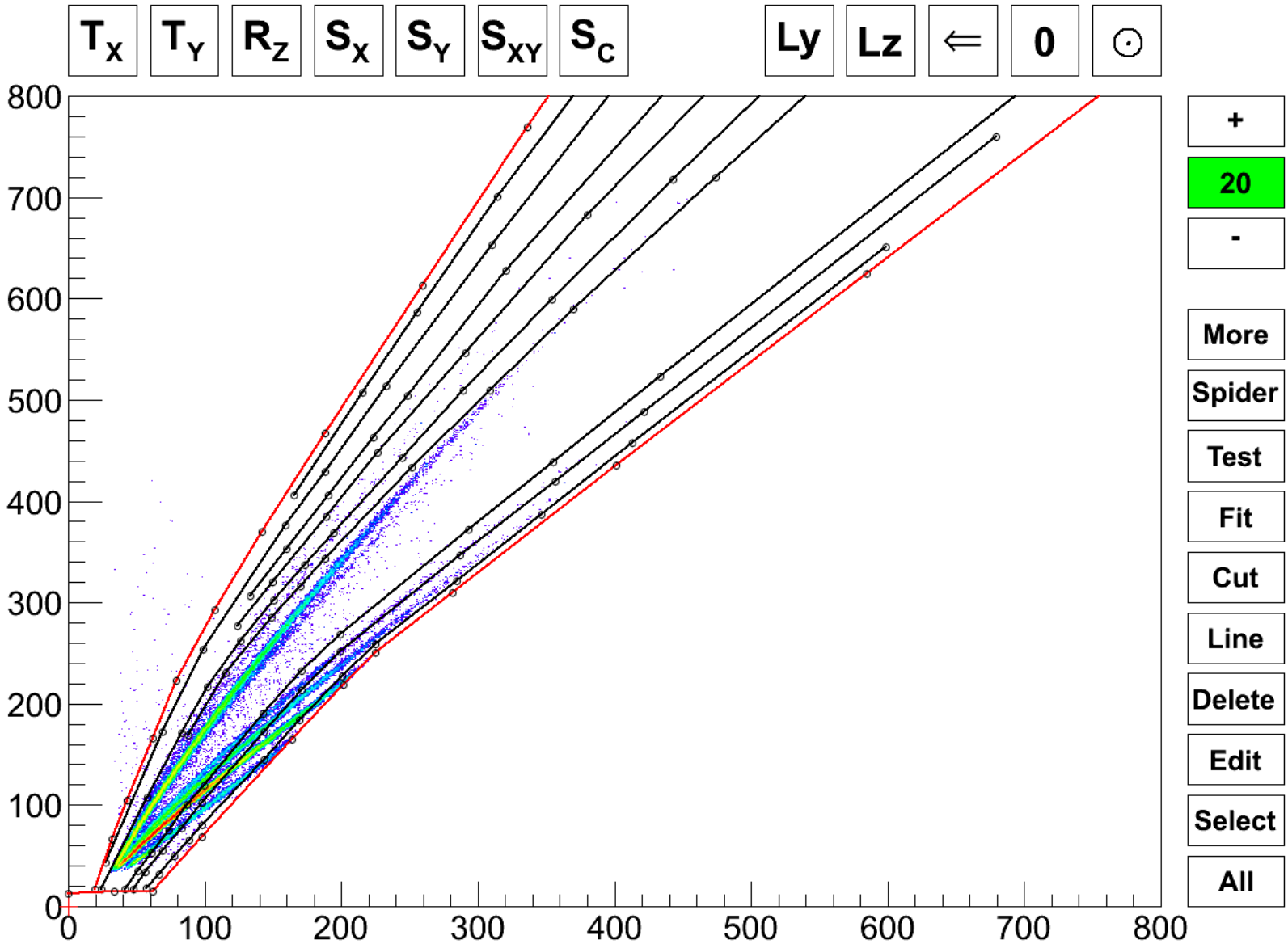
Example : CSI C09M08, runs 460-530



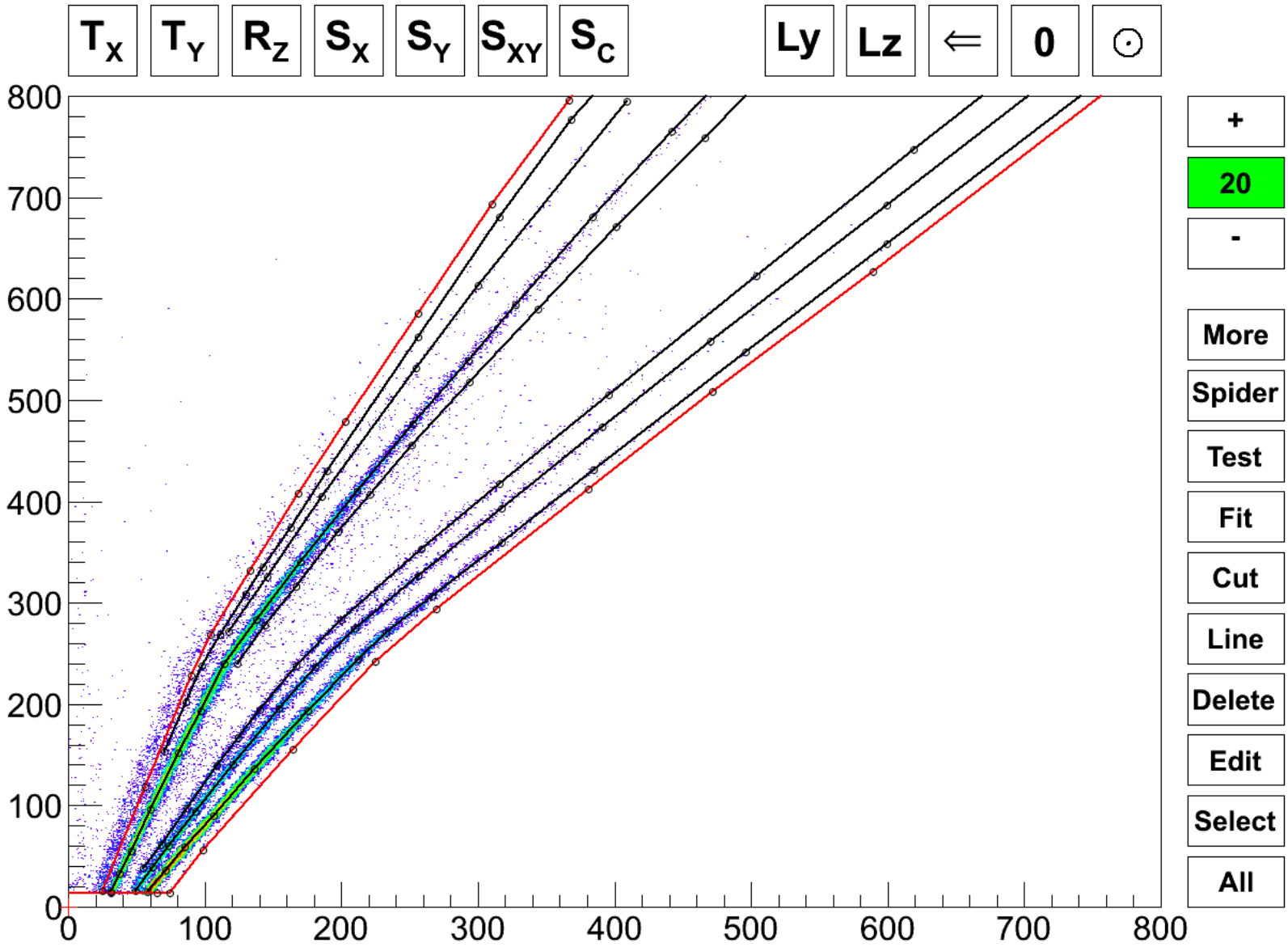
Example : CSI C09M16, runs 80-150



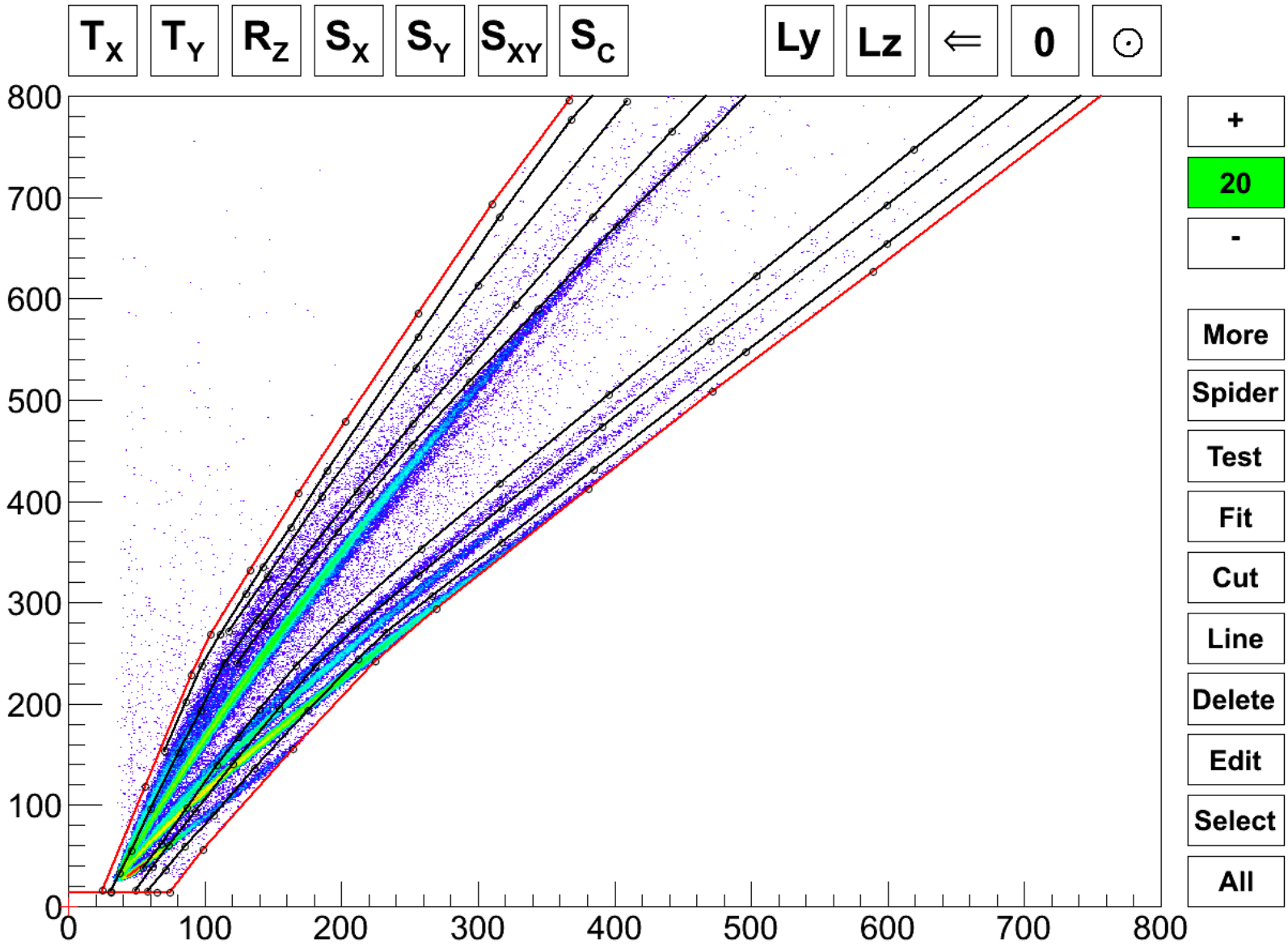
Example : CSI C09M16, runs 460-530



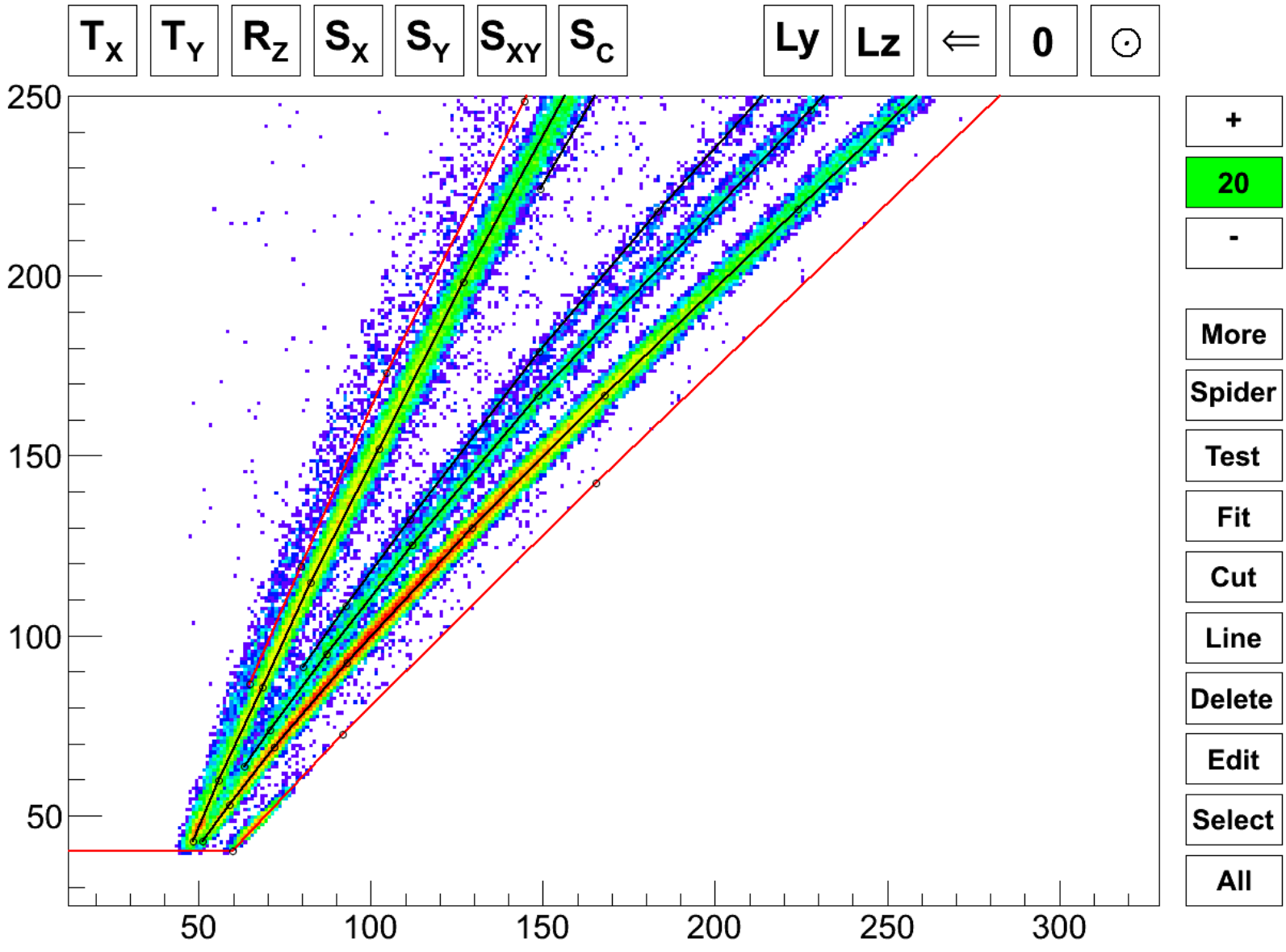
Example : CSI C10M06, runs 80-150



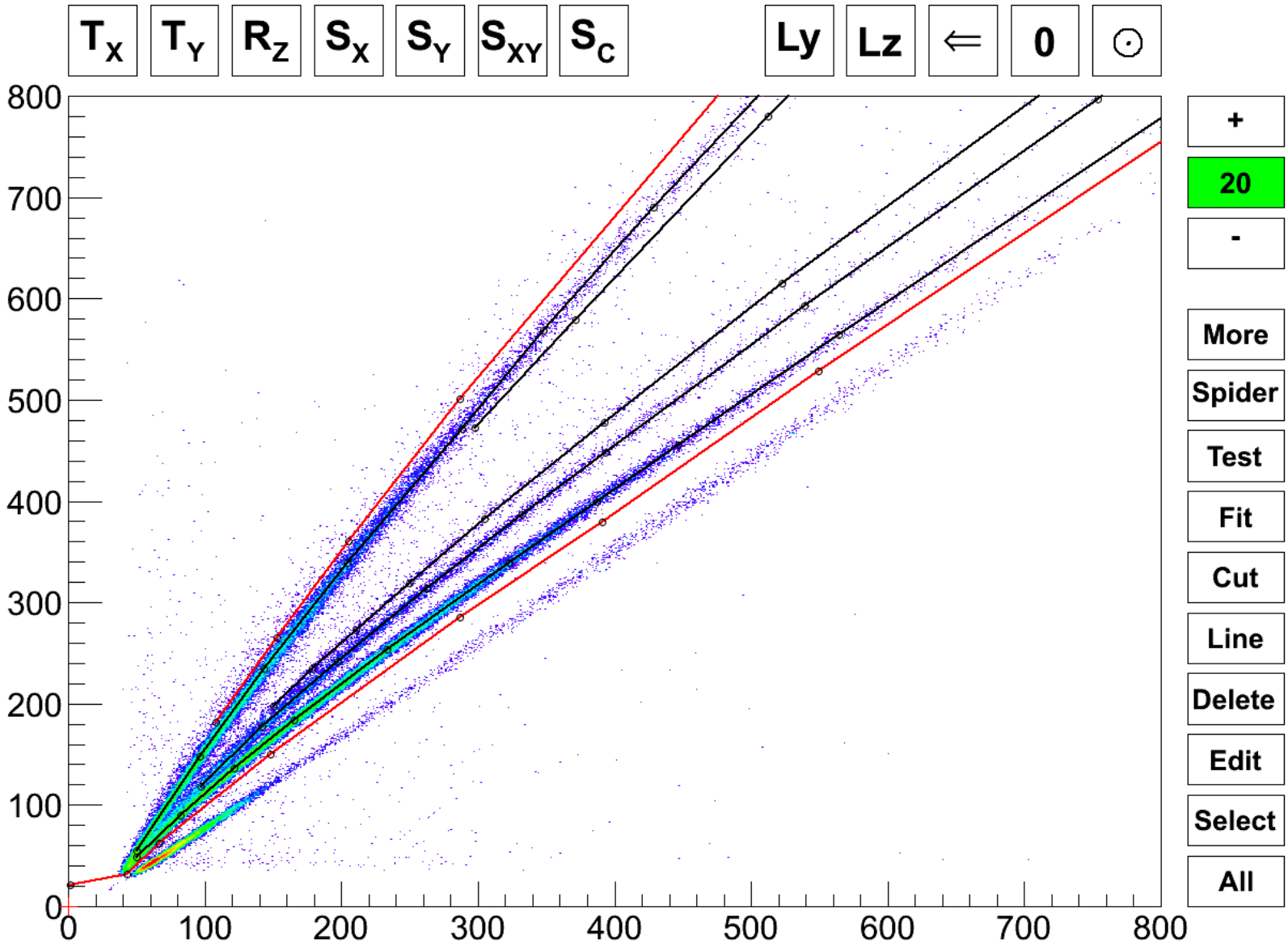
Example : CSI C10M06, runs 460-530



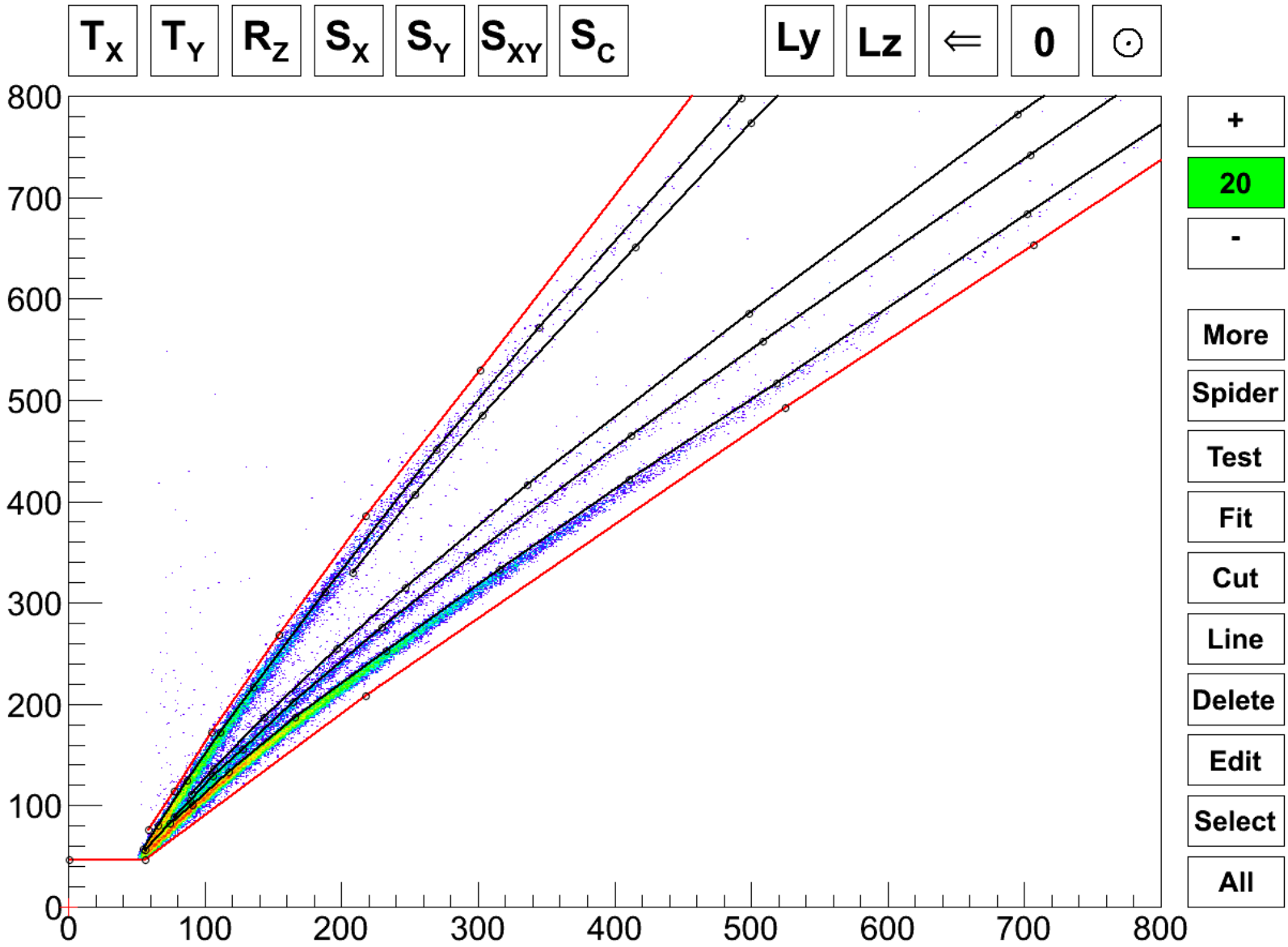
Example : CSI C11M09, runs 80-150



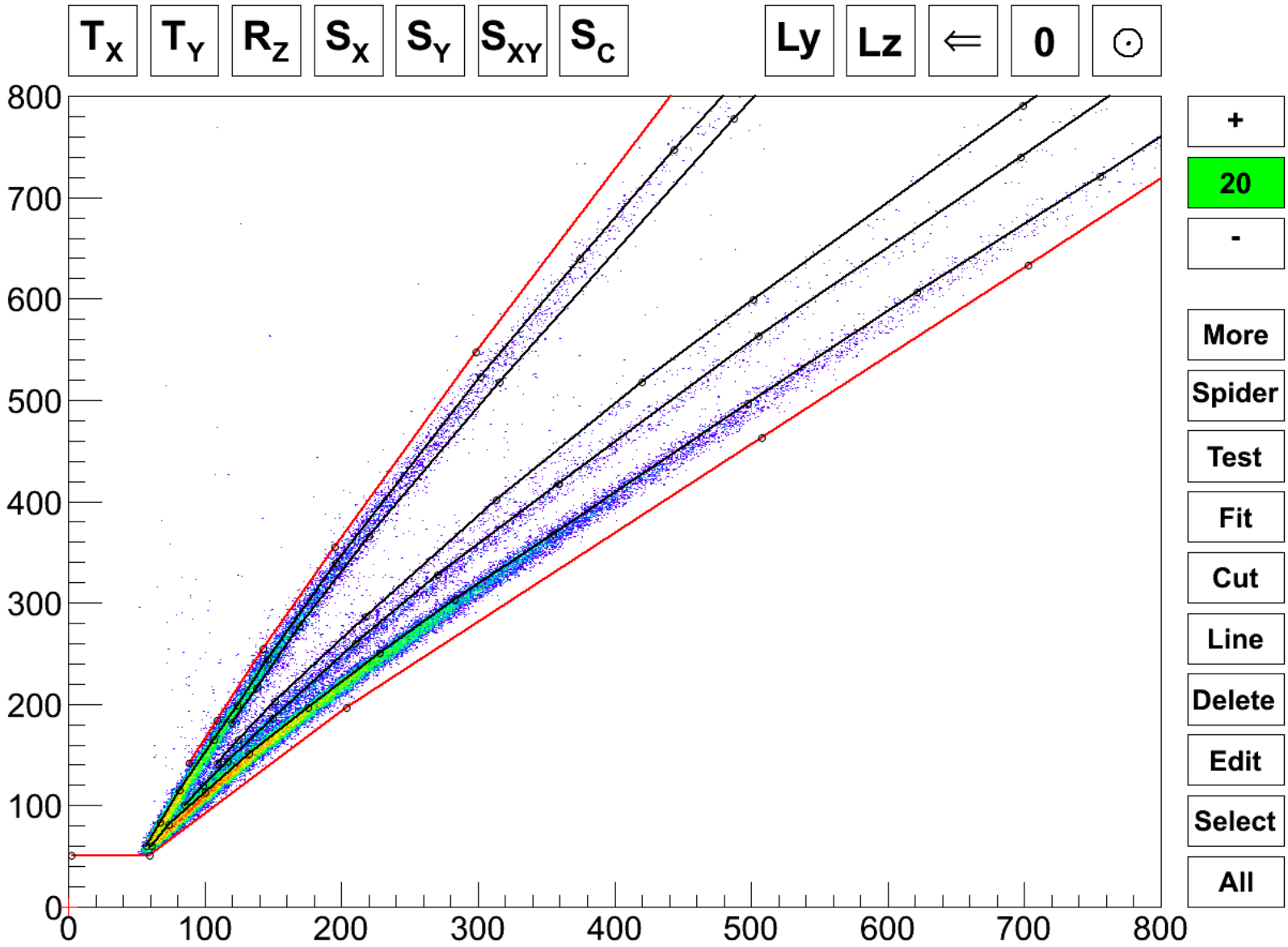
Example : CSI C12M23, runs 80-150



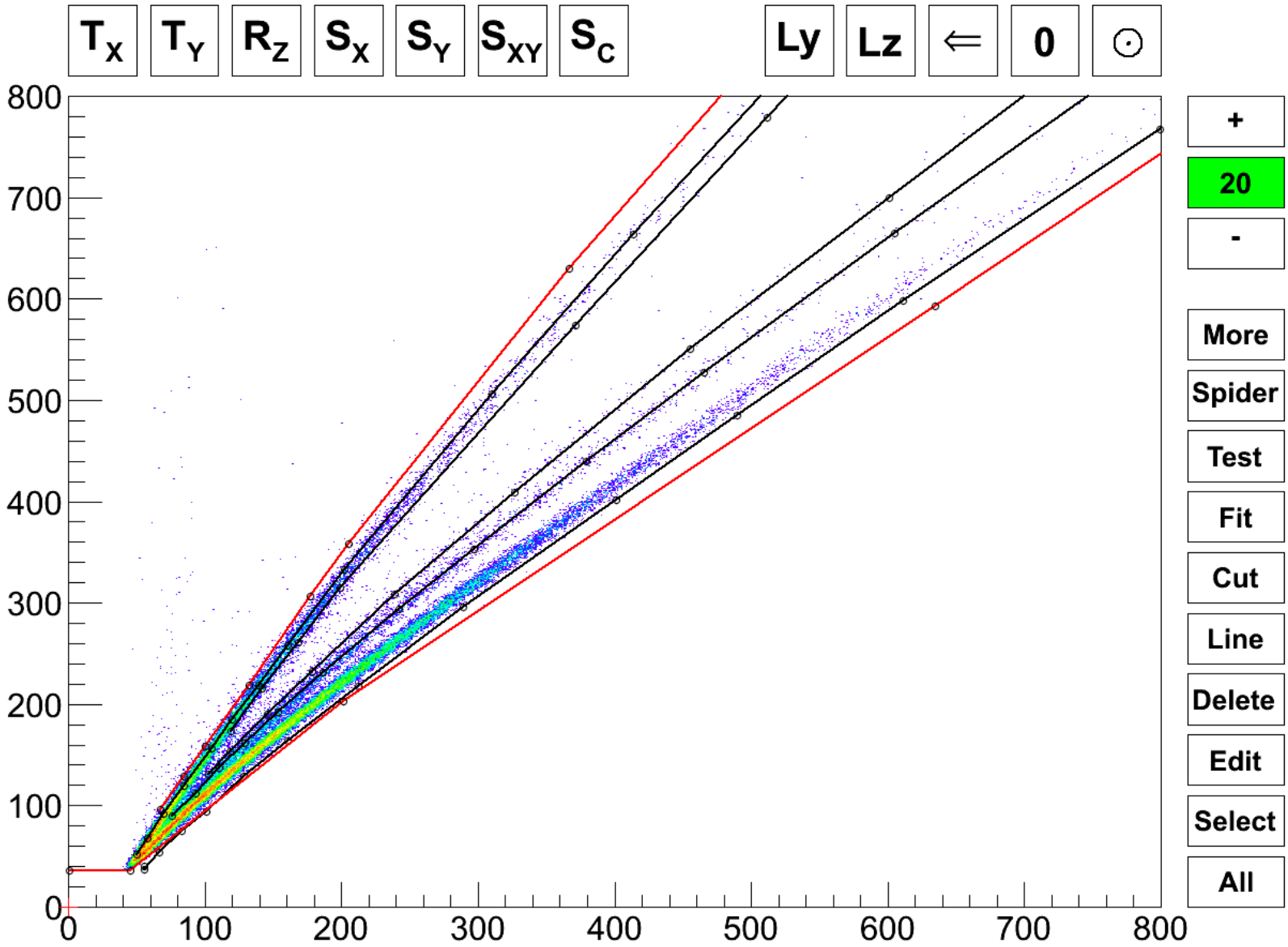
Example : CSI C13M07, runs 80-150



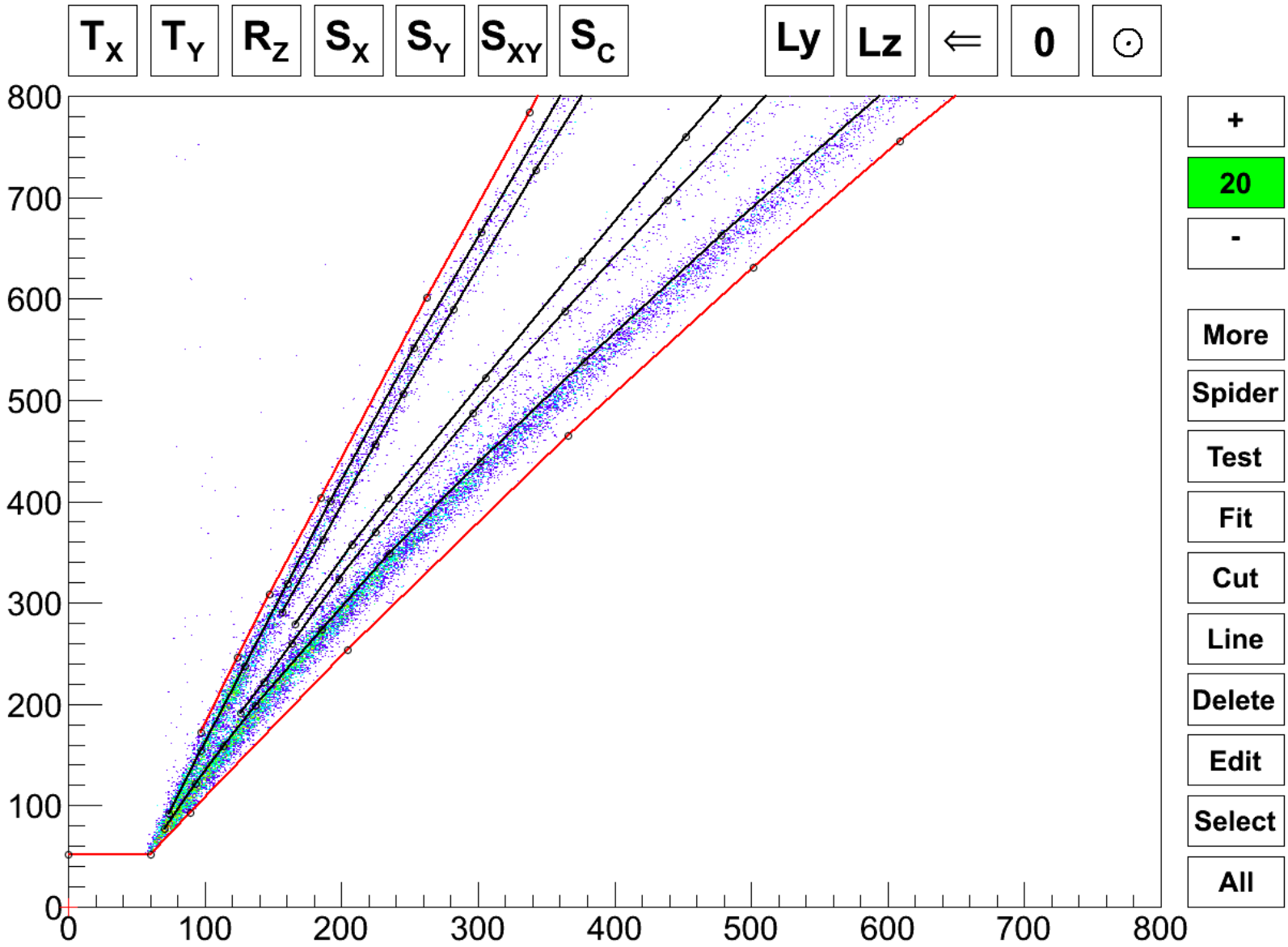
Example : CSI C13M13, runs 80-150



Example : CSI C13M23, runs 80-150



Example : CSI C16M03, runs 80-150



- Branche sur launchpad : `lp :kalibeda/e494s-dev`
- branche installée au centre de calcul de Lyon ⇒ SetKaliVedaVersion e494s

CI-SI

- Ajout d'une "ligne de Bragg" pour éliminer les faux contacts sur c6m3, c6m4, c7m3, c7m4, c6m15

CSI R-L

En partant des grilles CsI R-L faites pour e503 on apporte les modifications suivantes : **Runs 80-150 :**

- Détecteurs manquants : C6m17, c7m6, c7m7, c7m8, c7m23, c7m24
- Détecteurs repris : C8m24, c9m7 (ligne gamma), c10m22 (pas d'3He), c11m1, c11m4, c11m9, c11m20 (3H au lieu d'3He), c12m23 (ligne gamma), c13m7, c13m13, c16m3
- Plus toutes les directions acceptées pour les cuts gamma et IMF "above et below"
- Prolongation des lignes IMF : ok pour c10-17, fallait-il le faire aussi pour c4-9 ?

Runs 460-530 :

- A revoir : c4m3, c9m8, c9m16, c10m6 plusieurs modes de fonctionnement au cours du temps.