The STAR FaSt TPC Simulation
(A User's Guide)

P.G. Jones and I. Sakrejda
Lawrence Berkeley Laboratory

Abstract:

The STAR FaSt TPC Simulation program (FST) has been designed to simulate the response of the STAR TPC and the performance of the hit reconstruction algorithm. Input to the program are the hit banks created by the GEANT Monte-Carlo package [1]. FST assigns each native Monte-Carlo hit a spatial width and a spatial resolution according to a parameterization which is dependent on the track trajectory and it's position in the detector. The hits are smeared about the known track position according to their calculated spatial resolution. Those hits whose spatial width are then found to overlap in both the pad plane projection and in the drift direction are merged to form larger single clusters. The output of FST forms the input to the STAR pattern recognition algorithm.

Introduction:

The Fast TPC simulation seeks to provide a model for the STAR TPC hit-finder. This is achieved by parameterizing the pad response [2] and the pad resolution [3] as a function of the transverse diffusion and the track crossing angle normal to the padrow. In the drift direction the spatial extent of the pulse is determined by the track dip angle and by longitudinal diffusion of the primary ionization as it is drifted to the pad plane. An additional factor contributing to the spatial width of the pulse measured in the drift direction is the effect of the electronics shaping time, which has also been included into the parameterization.

In the current implementation of GEANT each track crossing is given at the exit point of a 2 mm volume centered on each pad. A correction is calculated to find the crossing at the center of each pad according to local trajectory of the track in the pad plane and the equation of the line describing the padrow. The track crossing angle and the dip angle are then calculated.

Spatial width:

Two estimates of the induced signal width on the pad plane are calculated. The first is obtained from the parameterization of the pad response [2]. Previous measurements have demonstrated that the parameterization of the pad response is good for crossing angles less than approximately 45°. For large crossing angle tracks, a large number of pads may be traversed in a single padrow. In this case the pulse in the pad plane is calculated from the projected arc length of track subtended by the row or by a simple projection of the tangent to the track using the momentum vector at the center of the pad. The smaller of the two values is chosen and an additional term is added to account for transverse diffusion. Because the pulse width calculated from the track projection onto the pad plane is always larger than the pad response calculated at the same value of the crossing angle, the pad response is scaled to meet to the projected width for crossing angles between 45° and 72°. In the drift direction the spatial extent of the charge distribution is calculated according to the track length projected over each pad, the longitudinal diffusion and the electronics shaping time.

Position Resolution:

The spatial resolution is similarly parameterized as a function of the expected signal-to-noise ratio, the transverse diffusion and the track crossing angle relative to normal to the padrow [3]. This parameterization is employed for all but the largest crossing angle tracks (below 89°). All hits are smeared about the known pad crossing using a gaussian distribution function whose width (sigma) is given by the calculated position resolution. For larger crossing angle tracks, the spatial resolution is determined by the sigma of the pad response divided by $\sqrt{12}$. In the drift direction the spatial resolution is fixed at an average of 700 μm; a value determined by the electronics shaping time [4].

Hit Merging:

Hits are merged according to their separation and calculated spatial width in both the padrow plane and the drift direction. First the hits are sorted row-by-row according to their position in the drift direction. A maximum cut-off is imposed on the spatial width of the charge distribution in the drift direction (also in the pad plane), in order to minimize the time spent looping over neighbouring hits in any given padrow. In this way, when neighbouring hits are separated by greater than twice the cutoff width, no more hits can merge with the hit in question. Hits in the same padrow are merged when their separation Δ is given by,

$$\Delta pad < \sigma_1 + \sigma_2 + 0.8 \text{ cm}$$

and

$$\Delta z < \sigma_1 + \sigma_2 + 1.0 \text{ cm}$$

where, σ is the calculated width of the charge distribution. The constant factors relate to one pad width and two time bins respectively, in the initial design specification of the TPC (namely 8 x 20 mm^2 pads and v_{drift} = 5 cm/μs with 512 time samples).

General Features:

The FST simulation program has been written with flexibility in mind. The program is parameter file controlled (see Appendix i) for easy modification of the TPC running parameters. The current version of the program allows for two different pad sizes to be investigated and two different pad geometries; namely straight or circular padrows. Since GEANT has only been set-up to cater for straight padrows, FST calculates internally the approximate pad crossings in the circular padrow geometry. This involves some approximation, as well as in the process of hit merging where neighbouring hits are assumed to distributed along a straight padrow. However, for the majority of hits the approximate assignment of the center of the padrow should not dominate the observed spatial resolution.

The program has also been written in such a way, that events prior to and following the triggered event may be read-in to form a single 'overlapped' event. This is particularly important in the high luminosity environment of proton-proton collisions where non-triggered events may be registered during the time that the TPC volume is active. This is the subject of a separate report [5], where the likely event rates and analysis of such events have been outlined in more detail. Overlapping events from the non-triggered beam crossings are displaced in time, according to the time elapsed either before or since the triggered event. This creates quite a different event topology to that created by a central silicon-silicon event, which has a similar total multiplicity. This feature of FST may be selected by a logical switch contained in the parameter file. The average number of events per crossing are calculated from the beam bunch separation, the beam luminosity, and the inelastic cross section. The actual number of events in any given beam crossing is determined according to a poisson distribution whose mean is the calculated average number of events per crossing.

As well as generating an output file of hits for the subsequent pattern-recognition process, FST may also output a NTUPLE for the study of hit merging as a function of (for example) various pseudorapidity intervals. The content of the FST NTUPLE is outlined in Appendix iii. Please note that the FST NTUPLE has a logical record length of 2048 words (1024 is the default). This must be specified as an additional argument when reading in the histogram using PAW [6].

Standard versions of the code (PATCHY cards file), makefiles, command files and parameter files may be found in the STAR library on both CSA and the SGI [7].

References:

[1] The GEANT interface to FST was written by R.J. Morse.

[2] P.G. Jones and G. Rai, The STAR TPC Pad Response Function, STAR note 33.

[3] P.G. Jones and G. Rai, The STAR TPC Pad Resolution Function, STAR note 34.

[4] Aleph Handbook (W. Blum, Ed.), ALEPH 89-77, p. V.62.

[5] M.A. Bloomer, P.G. Jones and I.M. Sakrejda,
Evaluation of the STAR TPC performance for detecting p+p collisions at RHIC, STAR note 57.

[6] R. Brun, O. Couet, C. Vandoni, P. Zanarini,
PAW The Complete Reference, V1.07, (1989).

[7] D. Olson, STARR:[doc]library_guide.doc.
R.J. Morse and W.A. Love,
The Current Implementation of the STAR Library at RHIC, STAR note 42.

Appendix i, Example parameter file:

```
! Parameter file to control the fast simulator
!
! Select pad geometry: T = circular, F = normal
  F
! Select pad length: T = long, F = short
  T
!
! Short pad length, Long pad length, Pad Pitch
   1.2   2.0   2.0
!
! Pad response: prf_0, prf_dr, prf_ta, ldiff
   .36200   .02330   .44600   .04000   ! 8 x 12 mm
   .36200   .02330   .63700   .04000   ! 8 x 20 mm
!
! Spatial resolution: sigma0, sigmad, sigmat, sigmal
   .04400   .00213   .10000   .07000   ! 8 x 12 mm
   .04400   .00166   .16650   .07000   ! 8 x 20 mm
!
! Limits for scaling pad response to projected width
   45.00   72.00   ! degrees
!
! Min sep for deconv.- drmin0, dzmin0: dz = sigz1 + sigz2 + dzmin0
    0.80    1.00   ! cm
!
! Drift velocity (cm/microsecond), Shaping time (nanoseconds)
    5.00   180.0
!
! Max merged size in r-phi and z; Max cut-off single hit
    8.00    9.00   20.00   ! cm
!
! *** The following parameters deal with overlapping events ***
!
! Select overlapping events
   F
!
! Spacing between bunch crossings (nanoseconds)
    220.
!
! Luminosity (cm^-2 per second)
   0.1400E+32
!
! Inelastic cross-section (mbarns)
        45.00
```

Appendix ii, Example command file:

```
$ass/nolog sys$input for005
$ass/nolog sys$command dbg$input
$run STARL:FST TPC_SIMULATION          ! Standard library version
FST_REF:FST_TPC_SIMULATION.PAR         ! Parameter file
RM:TEST_SISI.FZD                       ! Input file (GEANT output)
A                                      ! Zebra format
1                                      ! Events to process
Y                                      ! Save NTUPLE ?
TEST_SISI.NTUPLE                       ! NTUPLE filename
Y                                      ! Save output (PATREC input)
TEST_SISI.HITS                         ! Output filename
```

Appendix iii, NTUPLE word definitions:

```
************************
*  Var numb *     Name  *
************************
*      1    * x         *     !
*      2    * y         *     ! Spatial coordinates
*      3    * z         *     !
*      4    * dx        *     !
*      5    * dy        *     ! Spatial resolution
*      6    * dz        *     !
*      7    * itr       *     ! MC track number
*      8    * ires      *     ! Hit flag                       (a)
*      9    * trsiz     *     ! Spatial width - pad plane
*     10    * lsiz      *     ! Spatial width - drift direction
*     11    * alpha     *     ! Pad crossing angle (degrees)
*     12    * resx      *     !
*     13    * resy      *     ! Residuals (eg. yhit-ymc)
*     14    * resz      *     !
*     15    * eta       *     ! Pseudorapidity
*     16    * row       *     ! Row number
*     17    * dedx      *     ! dE/dx value
*     18    * iev       *     ! Event number                   (b)
*     19    * drm       *     ! Pad plane separation (merged)
*     20    * dzm       *     ! Drift separation (merged)
*     21    * num       *     ! Hits per cluster (merged)
************************
```

Notes:

```
(a) IRES = 0        normal hit
    IRES = 1,4,5    flag special case hits, ignored to avoid computational
                    overflows etc.
    IRES = 8        is hit discarded because it was merged with another
                    (larger) hit.
    IRES = 9        is a hit discarded in displaced events (i.e.those
                    other than the triggered event), when the displacement
                    took the hit outside the TPC.
    IRES = 11       a merged hit, rejected because it was too large.
    IRES = 12       a single hit, rejected because it was too large.
    IRKS = -'ve     is a merged hit that survived to the output. The
                    absolute value of the flag is the MC track number of
                    the other hit (or the last other hit - in the case of
                    multiple overlapping hits).

(b) iev = event_numberx1OOO + partition_number
```