BOOST Introduction

Z.Y. Deng     M. He     H.M. Liu
Y.J. Mao     Z.Y. You    Y.Yuan
BOOST Project
(BESIII Object Oriented Simulation Tool)

Event GENERBES Generator → Geometry Geant4 Tracking → Detector Digitization Response

HepEvt format → Hit objects → Raw data MC truth
BOOST Project

(BESIII Object Oriented Simulation Tool)

- **Components**
  - Generator
  - Particles and physics processes
  - Magnetic field
  - Material
  - MC truth
  - Data I/O
  - User interface
  - Geometry
  - Hits recording
  - Digitization

- common parts

- sub-detector parts
BOOST code structure

boost/

bes/

src/ source c++ codes
include/ header files

gen/ event generators
phy/ physics processes
tru/ MC truth
gmk/ common makefiles
mac/ common card files
dat/ geometry data files
exe/ main program for execution
doc/ documentations
mdc/ main drift chamber
tof/ time of flight
emc/ electromagnetic calorimeter
muc/ muon chamber
BOOST Project
(BESIII Object Oriented Simulation Tool)

Developers

- Liu Huaimin: common parts
- Yuan Ye: MDC
- Deng Ziyan: TOF & common parts
- Fu Chengdong/He Miao: EMC
- Peking University: MUC & xml geometry
BOOST Project

(BESIII Object Oriented Simulation Tool)

✓ Phase 1: Framework prototype, simple geometry with hit
  ✓ Finished, BOOST prototype realized in May, 2003

✓ Phase 2: Detailed detector with simple digitization
  ✓ Finished, now digits and MC truth output can be used for reconstruction

➢ Phase 3: Detailed digitization, data/MC comparisons
  ➢ In progress, need information from beam test and data. A long-term work!
Geant4 and BOOST releases

◆ Geant4
  ➢ Release 6.1  March, 2004
  ➢ Release 6.2  June, 2004
  ➢ Release 7.0  December, 2004
  ➢ Patch of 7.0  February, 2005

◆ BOOST (mantained by cvs)
  ➢ 'boost-1-0'  (March, 2004)  ----Geant4.6.1
  ➢ 'boost-1-1'  (June, 2004)  ----Geant4.6.2
  ➢ ......
  ➢ 'boost-2-0'  (February, 2005)  ----Geant4.7.0
  ➢ 'boost-2-1'  (May, 2005)  ---- Geant4.7.0+p01
BOOST Current Status

**geometry**

From boost-2-1

**barrel:** MDC, TOF, EMC, SCM, MUC

**end-cap:** TOF, EMC, MUC
BOOST Current Status

**generator**

**genbes:** (BESII generator system)
- Generate events in an ASCII file which supports HEPEVT interface with Geant4
- The same user interface as BESII
  - genbes.cards, genbes.user
- All BESII generators can be used in BOOST

**tester:** (single-particle generator for debugging)
- Shoot particles of given type
- to a given direction
- with given kinetic energy
- Users can randomize above quantities
BOOST Current Status

generator: rhopi

\[ J/\psi \rightarrow \rho \pi \rightarrow \pi^+\pi^- \gamma\gamma \]

Hits in barrel MUC
(\(\pi\) - punch through)

Shower in EMC
BOOST Current Status
processes & magnetic field & material

Physics processes
- BesPhysicsList constructed with Geant4 classes
- Many hadronic models installed, LHEP, LHEP-GN, QGSP-GN, …
- More research on them needed to choose suitable ones for BESIII physics

Magnetic field
- 1 Tesla uniform magnetic field defined inside SCM

Material
- All materials needed in BOOST defined in XML
BOOST Current Status

hits and digitization

**Hits recording**
- Finished last year

**digitization**
- **MDC**
  - Drift distance $\rightarrow$ drift time
  - dE/dx
  - Wire resolution added
  - Simple background added
- **TOF**
  - Detailed digitization
  - Light emission & light propagation
  - PMT response & threshold discrimination
- **EMC**
  - waveform added
  - Simulate main amplifier output
  - Provide risetime to eliminate noise
  - No noise, so not used now
- **MUC**
  - Digit information added
BOOST Current Status

MC truth

Purpose

- To provide truth information for reconstruction debugging
- Similar to MCMADE in BESII

What to save

- True state of the particles and their association with detector response
  - Particle
    - particles from generator
    - uninterrupted decayed daughters
  - Detector response
    - hits in sub-detectors associated with particles
**BOOST Current Status**

**MC truth**

- **Track and vertex** (associated with particle)
  - Track
    - PDGcode
    - charge
    - original vertex
    - terminal vertex
    - four-momentum
    - decayed daughters
  - Vertex
    - parent track
    - position
    - time

- **truth hits** (associated with track)
  - Considering demand of each reconstruction developer
    - MDC
      - Hit history
    - TOF
      - First-hit
    - EMC
      - Total energy loss
    - MUC
      - Hit history
**BOOST Current Status**

**Data I/O**

◆ **Ascii data I/O**

- It's temporary, but useful in software (SIM + REC) development
- Ascii data format fixed for the moment
- MC truth (TRUTH) & Raw data (DIGI) included in one file
  - EVHEAD
  - TRACKTRUTH, VERTEXTRUTH
  - MDCTRUTH, MDCDIGI
  - TOFTRUTH, TOFDIGI
  - EMCTRUTH, EMCDIGI
  - MUCTRUTH, MUCDIGI

- Users can turn on/off TRUTH and DIGI part of each sub-detector in run cards
**BOOST Current Status**

**User interface**

- **Friendly to both users and developers**
  - source ~dengzy/.boostenv
  - cvs co boost
  - edit boost.cards & boost.user
  - more controls in run cards

- **Run background/terminal/interactive/PBS job freely**
  - boost -b/-t/-i/-q
  - use 'boost -h' for detail
From LHC experience
- G4 direct comparison with G3
- G4 is slower (1.5~2) than G3
- G4 tracking particle to 0 energy

Compared with SIMBES
- SIMBES (G3/BESII), BOOST (G4/BESIII)

How to compare
- Similar cuts
  - CUTE CUTM (10KeV), range cuts (1mm)
- Same platform (Besfarm7)
- Physics events used (dimuon, rhopi, bhabha)
## BOOST Performance

### Speed

<table>
<thead>
<tr>
<th>Event</th>
<th>SIMBES</th>
<th>BOOST</th>
</tr>
</thead>
<tbody>
<tr>
<td>ee -&gt; uu</td>
<td>0.122</td>
<td>0.189</td>
</tr>
<tr>
<td>J/ψ -&gt; ρ π</td>
<td>0.471</td>
<td>0.763</td>
</tr>
<tr>
<td>ee -&gt; ee</td>
<td>0.736</td>
<td>1.224</td>
</tr>
</tbody>
</table>

- BOOST 1.5 times slower than SIMBES
- More to do about speed improvement
  - Threshold cut in digitization
  - Speed of hadronic models to be compared
- Stable (no crash for 100,000 events)
BOOST Integration to BOSS framework

Different developing environment

- **BOOST**
  - On besfarm
  - Configured with gmake
  - Based on GEANT4

- **BOSS**
  - On koala
  - Configured with CMT
  - Based on GAUDI

Integration Status

- BOOSTalg algorithm/packages created in BOSS

- Key functions of BOOST realized in BOSS

- Interface of saving data to TDS is ok

- Codes imported into cvs