A first magnetic field

First steps in field propagation in Geant4

John Apostolakis

11th November 2002
Contents

1. What is involved in propagating in a field
   • calculating the motions
   • intersecting the volume boundaries

2. A first example
   • Defining a simple field in Geant4

3. More capabilities
   • Using your own field
   • Many fields,
   • Tomorrow: tuning for performance, ..
Magnetic field: overview

• In order to *propagate* a particle inside a field (e.g. magnetic, electric or both), we *solve* the equation of motion of the particle in the field.

• We use a Runge-Kutta method for the integration of the ordinary differential equations of motion.
  – Several Runge-Kutta ‘steppers’ are available.

• In specific cases other solvers can also be used:
  – In a uniform field, using the analytical solution.
  – In a nearly uniform field (BgsTransportation/future)
  – In a smooth but varying field, with new RK+helix.
Magnetic field: overview (cont)

• Using the method to calculate the track's motion in a field, Geant4 breaks up this curved path into linear chord segments.

• We determine the chord segments so that they closely approximate the curved path.

• We use the chords to interrogate the Navigator, to see whether the track has crossed a volume boundary.
Stepping and accuracy

- You can set the accuracy of the volume intersection,
  - by setting a parameter called the “miss distance”
    - it is a measure of the error in whether the approximate track intersects a volume.
    - Default “miss distance” is 3 mm.
- One physics/tracking step can create several chords.
  - In some cases, one step consists of several helix turns.
Magnetic field: a first example

Create your Magnetic field class

– Uniform field:
  • Use an object of the G4UniformMagField class

```cpp
#include "G4UniformMagField.hh"
#include "G4FieldManager.hh"
#include "G4TransportationManager.hh"

G4MagneticField* magField = new G4UniformMagField(
    G4ThreeVector(1.0*Tesla, 0.0, 0.0) );
```

– Non-uniform field:
  • Create your concrete class derived from G4MagneticField
Magnetic field: a first example

Tell Geant4 to use your field

• Find the global Field Manager

  G4FieldManager* globalFieldMgr =
  G4TransportationManager::
    GetTransportationManager()
  ->GetFieldManager();

• Set the field for this FieldManager,

  globalFieldMgr->SetDetectorField(magField);

• and create a Chord Finder.

  globalFieldMgr->CreateChordFinder(magField);
In practice: exampleN04

From geant4/examples/novice/N04/src/ExN04DetectorConstruction.cc

```cpp
G4VPhysicalVolume* ExN04DetectorConstruction::Construct()
{
    //-------------------------------------------------------------------------
    // Magnetic field
    //-------------------------------------------------------------------------

    static G4bool fieldIsInitialized = false;
    if(!fieldIsInitialized)
    {
        ExN04Field* myField = new ExN04Field;
        G4FieldManager* fieldMgr = G4TransportationManager::GetTransportationManager()->GetFieldManager();
        fieldMgr->SetDetectorField(myField);
        fieldMgr->CreateChordFinder(myField);
        fieldIsInitialized = true;
    }

    return myField;
}
```
Beyond your first field

• Create your own field class
  – To describe your setup’s EM field

• Global field and local fields
  – The world or detector field manager
  – An alternative field manager can be associated with any logical volume
    • Currently the field must accept position global coordinates and return field in global coordinates

• Customizing the field propagation classes
  – Choosing an appropriate stepper for your field
  – Setting precision parameters
Creating your own field

Create a class, with one key method – that calculates the value of the field at a Point

```cpp
void ExN04Field::GetFieldValue(
    const double Point[4],
    double *field) const
{
    field[0] = 0.;
    field[1] = 0.;
    if(abs(Point[2])<zmax &&
        (sqr(Point[0])+sqr(Point[1]))<rmax_sq)
    { field[2] = Bz; }
    else
    { field[2] = 0.; }
}
```
Global and local fields

• One field manager is associated with the ‘world’
  – Set in G4TransportationManager

• Other volumes can override this
  – By associating a field manager with any logical volume
    • By default this is propagated to all its daughter volumes
      G4FieldManager* localFieldMgr =
      new G4FieldManager(magField);
      logVolume->setFieldManager(localFieldMgr, true);
      where ‘true’ makes it push the field to all the volumes it
      contains.