Geant4 for external beam therapy

Brad Oborn$^{1,2}$

$^1$Illawarra Cancer Care Centre (ICCC), Wollongong Hospital, NSW 2500 Australia
$^2$Centre for Medical Radiation Physics (CMRP), University of Wollongong, NSW 2522 Australia

April 24, 2013
External Beam Radiotherapy Simulations

- Simple, just model this thing... well just the key parts
**Why use Geant4 for external beam RT simulations??**

- Most people in the past have used BEAMnrc/DOSXYZnrc to successfully simulate linac treatments
- So why do this in geant4?
- Geant4 can do more stuff...
  - simulate neutrons + other particles
  - simulate electric/magnetic fields
  - break down dose calculations into various components
  - essentially unlimited geometry flexibility
- However some cons of using Geant4:
  - no default user interface to build a linac (however see GATE, GAMOS)
  - user needs to write most code, or search/share code..
  - usually slower then BEAMnrc/DOSXYZnrc for most applications
- Interestingly, Varian use Geant4 to model their TrueBeam - soon to go to cloud based simulations built by them where you can change geometry and do testing...
Welcome to the home page of the GAMOS Project

GAMOS is a GEANT4-based framework that is at the same time easy-to-use and flexible.

The comprehensive scripting language makes it easy to implement the most common requirements of a Medical Physics application, without any need of C++ coding.

The plug-in technology together with a careful modular design, a detailed documentation and a set of examples and tutorials that explain in detail how to extend the framework in different directions allows to exploit the full flexibility of GEANT4, by creating new user code or by reusing any piece of GEANT4 code and mixing it seamlessly with the existing GAMOS components.

Thanks to its big flexibility, already a sensible fraction of GAMOS users work in other fields than medical physics. If this is your case we recommend you to have a look at the ‘Histogram and Scorer tutorial’

In summary, by using GAMOS you will be able to carry your GEANT4-based simulation in an easy way and at the same time you will have the flexibility of using any of the GEANT4 components and mix with or substitute the GAMOS components.
Try GATE?

GATE for Radiation Therapy applications

GATE was initially developed for PET and SPECT applications. From version 6.0, some specific tools have been added for radiation therapy (RT) applications. The following examples illustrate the use of GATE for various photon, proton and carbon ion applications.

The tools dedicated to radiation therapy simulations provided in this GATE release are provided “as is” and on an “as available” basis without any representation or endorsement made and without warranty of any kind. In clear, it is a first attempt in this field and the authors are pretty sure that there are bugs. We will be happy if you can report bugs and hence improve this collaborative software.

You can find a list of the systems that have already been modeled by the collaboration members here.

Used by permission, CREAB (CNRS UMR 5220 – INSERM U1044 – Université Lyon 1 – INSA Lyon), and Leon Berard cancer center.
This one's ready to go right now...
Still not happy...

- So you want to build your own?
Still not happy...

- So you want to build your own?
Still not happy...

- So you want to build your own?

- *HINT* at least play with the medlinac2 example for a while first...
Where to start (IMO)...

- Breakdown the simulation into its basic components:

  **Stage 0** - conversion of incident electron beam on target to x-ray beam: only required once for each energy/mode

  **Stage 1** - simulation through patient dependent section: jaws

  **Stage 2** - simulation through patient dependent section: MLCs

  **Stage 3** - patient/phantom dose calculation:

  At the end of each stage you are best to create a "phase space file", and then use this as the particle source into the next stage
Phase space files (in case you don’t know)

- These are binary files which hold the properties of particles terminated from a previous simulation at some position in space.
- E.G. stop all x-ray particles after they escape the linac head, save the properties to a file.
- Each line of file holds one particle that crossed that plane/surface/condition: x, y, z, px, py, pz, E, type, latch, other stuff that you want...
- Format varies - up to you what to do...
- IAEA standard format is becoming more popular: Varian Truebeam will have 50GB of data available soon.
- Also IAEA people have Geant4 ready package to add to your code to read IAEA format...
- Personally, I built my own simple format before IAEA and so stick to that...
What do I need to include in the model of a Linac head??

- Basically everything detailed in your geometry package provided by the linac vendor
- target, primary collimator, exit window, flattening filter, ion chamber, mirror, jaws, MLCs
- best to put more in up front rather than later...
Geometry components: G4 solids

1. Target: simple square slab on slab (G4Box is fine, same as BEAMnrc)
2. Primary Collimator: G4Cons minus a G4Con (cone shaped)
3. Exit window: G4Box
4. Flattening Filter: G4Cons \( \times \) many (on top of each other)
5. Ion chamber: layers of G4Cons (discs), add walls if you want..
6. Jaws: G4Box, however move with field size and rotate with collimator... see medLinac example
7. MLC’s: fairly complex boolean solid, leaves move/shift and rotate with coll..
8. Water block phantom: G4Box
9. CT-data based patient phantoms: G4NestedVolumes stuff!

Brad Oborn (ICCC, CMRP)
Basic model benchmarking

- The first task to perform is to determine if your 6MV beam matches real linac data...
- Need to setup a watertank phantom and compare with real measured data
- mmm, block of water, extract dose inside it....how?
- default dose scorers in G4 should do a good job
- compare plots with exp. data
- change properties of electron beam hitting target to get required match
- energy peak, energy FWHM, spot size FWHM, angular divergence....a bit out there in the literature
- e.g. 6.00 MeV peak, 0.01 MeV FWHM, 0.9x1.2 mm spot size FWHM, 0-deg divergence for my 6MV work
Watertank simulation

1. G4Box = watertank!

2. Decide on voxel sizes to score in: 1-2 mm cubes
   - Careful: 1mm³ will need > 1GB RAM for 1 50x50x50cm watertank

3. Set up parallel world mesh scoring: built-in or custom
   - a. inside SteppingAction access the G4Step
   - b. ask if dE occurred in current step
   - c. if so, then find x,y,z and bin
   - d. oh and initialize dE arrays (constructor) and dump (destructor)

4. Examine results vs measured data

5. Repeat and tinker with source parameters until you get the match you want...
1. G4Box = watertank!
2. Decide on voxel sizes to score in: 1-2 mm cubes
   Careful: 1mm³ will need > 1GB RAM for 1 50x50x50cm watertank
3. Set up parallel world mesh scoring: built-in or custom
   a. inside SteppingAction access the G4Step
   b. ask if dE occured in current step
   c. if so, then find x,y,z and bin
   d. oh and initialize dE arrays (constructor) and dump (destructor)
4. Examine results vs measured data
5. Repeat and tinker with source parameters until you get the match you want...
   or just download a phase space file from the IAEA website
Calibration of doses...

- You can actually simulate the dose to the linac ion chamber
- Reference is 10x10 cm field size
- All doses relate back to the dose at 10x10 cm
- Advanced stuff really
IMRT plans consist of say up to 10 beam, each with up 20 segments, i.e
10 different stage 1 sims (through jaws)
200 different stage 2 sims (through MLCs)
200 different stage 3/phantom sims
That's a lot of different parts....
In BEAMnrc/DOSXYZnrc the default approach is to use dynamic particle sources
- Each segment is simulated in the same run
- Num of Particles is proportional to the MU delivered for that segment
- At the end you just get back a single dose cube
Discussion/Conclusions

- Geant4 can quite successfully be used to model a linac head for radiotherapy.
- Fairly steep learning curve.
- Best not to reinvent the wheel...
- Several “user friendly” system exist where you can develop Linac head models exist: GATE, GAMOS, TOPAS (soon!)
- Building your own application (based on MedLinac2 or similar) would require more input, however I feel you would get more of an understanding of Geant4 and the Monte Carlo method.
Happy coding...