Short Guide to Choosing Your Physics Lists

Mihaly Novak (CERN, EP-SFT)

Geant4 Tutorial at Lund University, Lund (Sweden), 3-7 September 2018

Geant4.10.4

Slides based on Dennis Wright (SLAC) & Vladimir Ivantchenko (CERN) lectures

OUTLINE

- **Recapitulation**
  - Reference physics lists. Physics list naming conventions.
- **Validation**
- **Example**

RECAPITULATION

- **Recapitulation**
  - Physics list. Reference physics lists. Physics list naming conventions.
- **Validation**
  - Hadronic. Electromagnetic.
- **Example**

Physics Lists:

- **Physics List is an object that is responsible to:**
  - specify all the particles that will be used in the simulation application
  - together with the list of physics processes assigned to each individual particles
  - the user can give the list of particles and assign different set of processes to them
  - this will determine the “physics environment” of the simulation
  - the user must have a good understanding of the physics required to describe properly the given problem
  - omission of relevant particles and/or physics interactions could lead to poor modelling results
  - omission of relevant particles and/or physics interactions could lead to poor modelling results

Reference Physics Lists:

- **"Production physics lists":**
  - these physics lists are used by large user groups like ATLAS, CMS, etc.
  - because of their importance, they are well-maintained and tested physics lists
  - they are changed, updated less frequently: very stable physics lists
  - they are extensively validated by the developers and the user communities
  - FTFP_BERT, QGSP_BERT, QGSP_FTFP_BERT_EMV, FTFP_BERT_HP, QGSP_BIC_EMY, QGSP_BIC_HP, QBBC, Shielding

- **Caveats:**
  - these lists are provided as a "best guess" of the physics needed in some given use cases
  - when a user decide to use them, the user is responsible for "validating" the physics for that given application
  - it means adding (or removing) the appropriate physics, using the proper settings
  - they are intended to give a starting point or template for the user physics list

Reference Physics Lists: naming convention

- **Some Hadronic options:**
  - "QGS" Quark Gluon String model (> ~15 GeV)
  - "FTF" FRITIOF String model (> ~5 GeV)
  - "BIC" Binary Cascade model (< ~10 GeV)
  - "BERT" Bertini Cascade model (< ~10 GeV)
  - "P" `G4Precompound` model used for de-excitation
  - "HP" High Precision neutron model (< ~20 MeV)
- **Some EM options:**
  - No suffix: standard EM i.e. the default `G4EmStandardPhysics` constructor
  - "EMV" `G4EmStandardPhysics_option1` CTR: HEP, fast but less precise
  - "EMY" `G4EmStandardPhysics_option3` CTR: medical, space sci., precise
  - "EMZ" `G4EmStandardPhysics_option4` CTR: most precise EM physics
- **Name decoding:** String(s)_Cascade_Neutron_EM
- **The complete list of pre-packaged physics list with detailed description can be found in the documentation ("*Guide for Physics Lists*"):**
  - http://geant4-userdoc.web.cern.ch/geant4-userdoc/UsersGuides/PhysicsListGuide/html/index.html

Reference Physics Lists: naming convention (example)

- **FTFP_BERT:**
  - Recommended by Geant4 developers for HEP applications
  - Includes the standard EM physics i.e. `G4EmStandardPhysics` CTR
  - "FTF" FRITIOF string model (> 4 GeV)
  - "BERT" Bertini Cascade model (< 5 GeV)
  - "P" `G4Precompound` model used for de-excitation

- **QGSP_BIC_HP(_EMZ):**
  - Recommended for medical applications (experimental QGSP_BIC_AllHP)
  - "QGS" Quark Gluon String model (> 12 GeV)
  - "FTF" FRITIOF String model (9.5 - 25 GeV)
  - "P" `G4Precompound` model used for de-excitation
  - "BIC" Binary Cascade model (200 MeV - 9.9 GeV)
  - "HP" High Precision neutron model (< ~20 MeV)
  - "EMZ" `G4EmStandardPhysics_option4` CTR (or EMY that's a bit less precise)

Example: using reference physics lists with EM option

- **QGSP_BIC_HP_EMZ:**
  - a QGSP_BIC_HP reference physics list, including all the above mentioned CTRs is available (but with the standard EM physics)
  - the `G4PhysListFactory` knows everything about the available reference lists
  - moreover, it makes possible to replace their EM option with a new one

```
// IM YOUR MAIN APPLICATION
//
  // create your run manager
#ifdef G4MULTITHREADED
  G4MTRunManager* runManager = new G4MTRunManager;
  // number of threads can be defined via macro command
  runManager->SetNumberOfThreads(4);
#else
  G4RunManager* runManager = new G4RunManager;
#endif
  //
  // create a physics list factory object that knows
  // everything about the available reference physics lists
  // and can replace their default EM option
  G4PhysListFactory physListFactory;
  // obtain the QGSP_BIC_HP_EMZ reference physics lists
  // which is the QGSP_BIC_HP refrence list with opt4 EM
  const G4String plName = "QGSP_BIC_HP_EMZ";
  G4VModularPhysicsList* pList = physListFactory.GetReferencePhysList(plName);
  // (check that pList is not nullptr, that I skipp now)
  // register your physics list in the run manager
  runManager->SetUserInitialization(pList);
  // register further mandatory objects i.e. Detector and Primary-generator
  ...
```

Choosing your physics list:

- **Recommendation:**
  - Ideal situation: the user(s) have a good understanding of the physics relevant for a given application
    - **the user can either build its own physics list or decide to use a pre-defined one**
    - **the chosen physics list needs to be validated for the given application**
    - **can be done either by the user or by someone else in case of some reference lists**
    - **during the validation procedure, some parts of the physics list might be changed add physics, remove physics, change settings, etc.**
  - The given application belongs to a well defined application area (e.g. medical applications)
    - **the user can choose the reference physics list recommended for the given application area as a staring point**
    - **the chosen physics list needs to be validated for the given application (same as above)**
  - Something that always works (but time consuming):
    - **the user can take the most accurate physics settings (e.g. opt4 for EM)**
    - **run some simulation with lower statistics to obtain the most accurate result**
    - **then the user can take a less accurate but fast physics setting (e.g. opt0 for EM) as a starting point and obtain some simulation results**
    - **then granularly extend the initial physics list by using the accurate results as reference**

VALIDATION

- Recapitulation
  - Reference physics lists. Physics list naming conventions.
- Validation
- Example

Validation:

- **Using the Geant4 validation results:**
  - you must choose a physics list based on how well its component processes and models perform in your specific case:
    - physics accuracy versus CPU performance
  - Geant4 provides validation (i.e. comparison to data) for most of its physics codes
    - validation is a continuing task, performed at least as often as each release
    - more validation tests added as time goes on
  - To access these comparisons, go to Geant4 website:
    - Geant4 website: **https://geant4.web.cern.ch**
    - Click: **Validation of Geant4**
    - Then choose **Validation and Testing** from the menu
    - We will use the **Geant4 GRID-based testing results portal** today

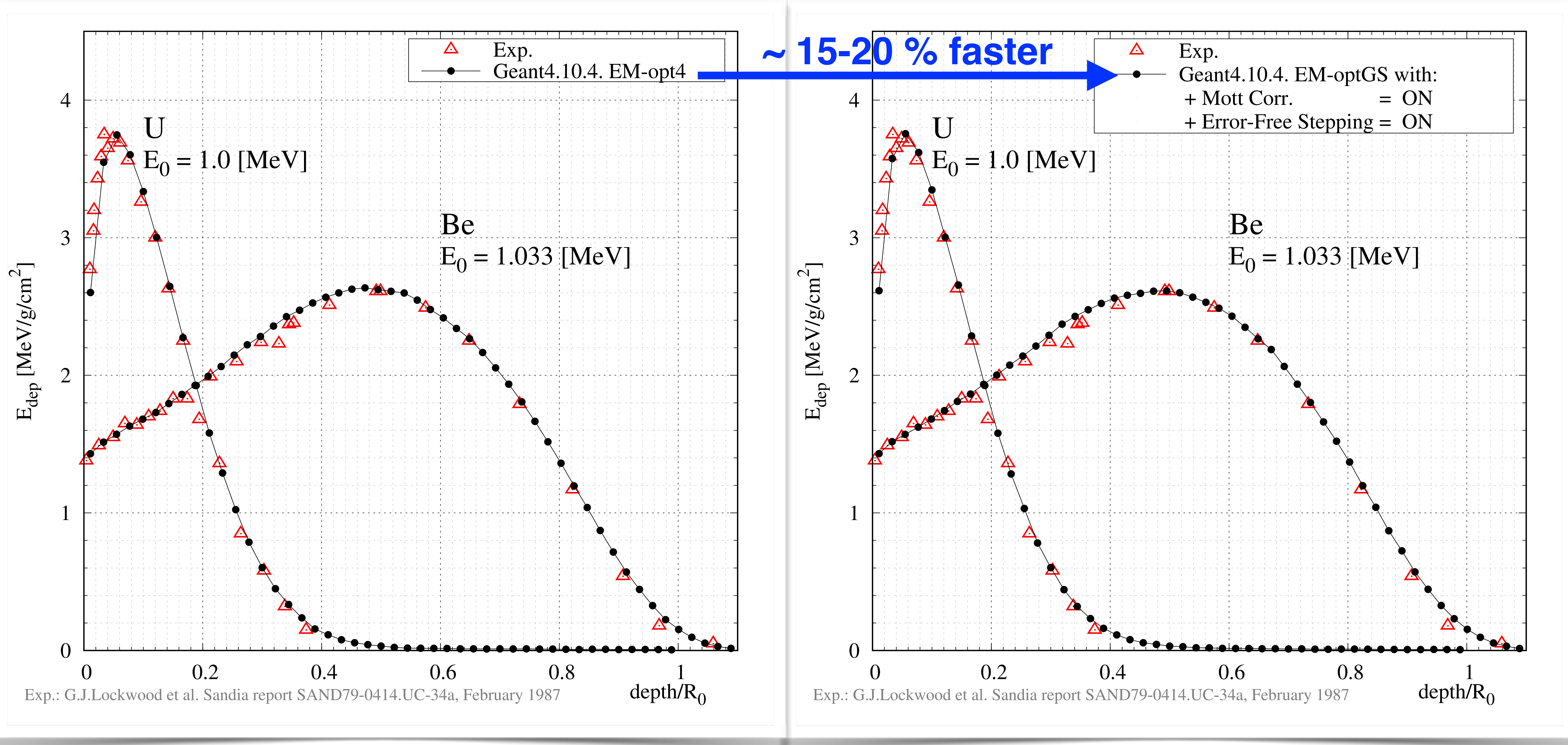
EXAMPLE

- Recapitulation
  - Reference physics lists. Physics list naming conventions.
- Validation
- Example

Example:

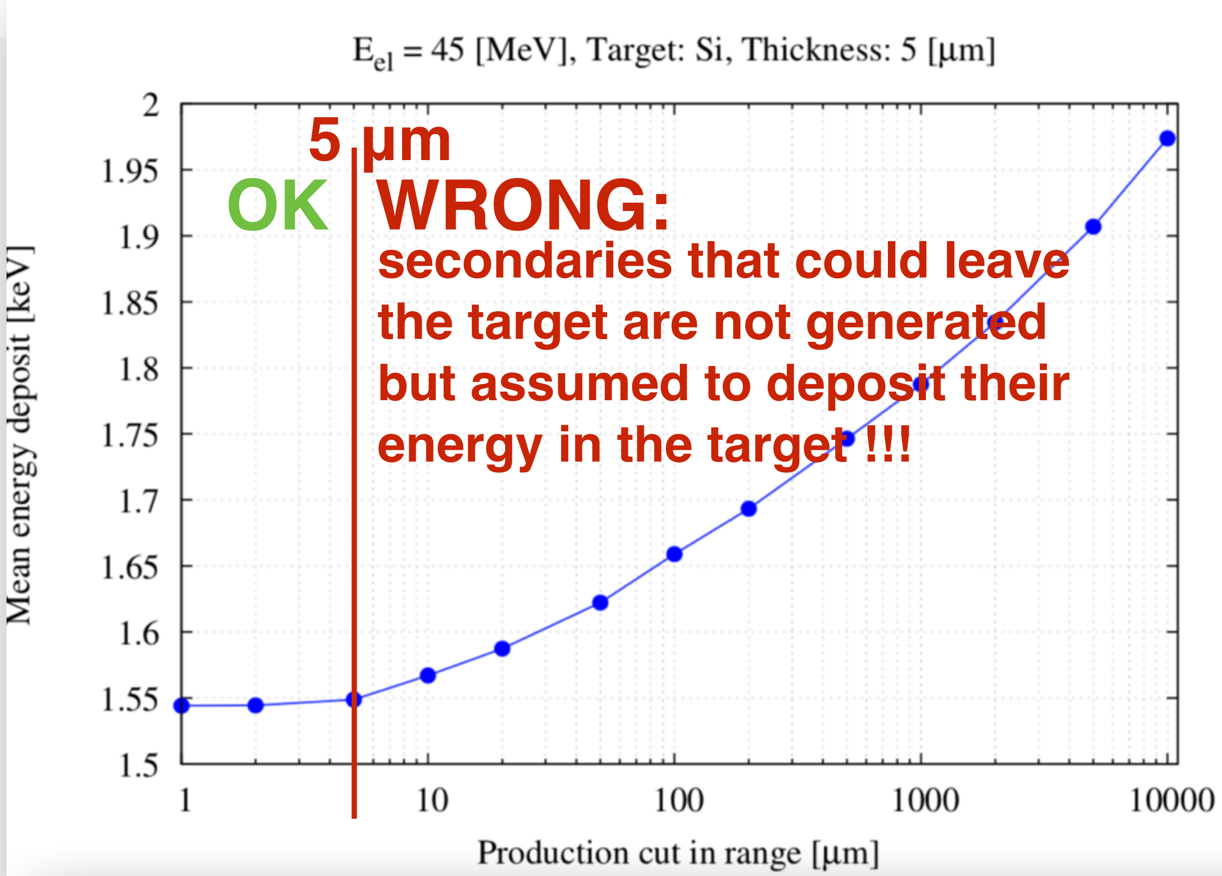
- **Suppose you want to simulate (EM) depth dose profile:**
  - simulation of energy deposit by energetic electrons as a function of the penetration depth (both lighter and heavier materials)
  - we will use the Geant4 validation results from **Geant4 GRID-based testing results portal**, especially **test37** to choose our initial physics list to start with
  - then we will adjust our initial reference physics list to achieve maximum physics performance while improving the computation efficiency

Example:



~ 15-20 % faster
Exp.
Geant4.10.4. EM-opt4
Geant4.10.4. EM-optGS with:
+ Mott Corr. = ON
+ Error-Free Stepping = ON
U
E0 = 1.0 [MeV]
Be
E0 = 1.033 [MeV]
Edep [MeV/g/cm2]
depth/R0
0
0.2
0.4
0.6
0.8
1
2
3
4
Exp.: G.J.Lockwood et al. Sandia report SAND79-0414.UC-34a, February 1987

Secondary production threshold: never forget!



cut [μm]	mean E_{dep}	rms E_{dep}	prod. thres. [keV]		mean num. sec.	
			γ	e^-	γ	e^-
1	1.54423	0.000573911	0.99	0.99	0.0006811	0.1018230
2	1.54443	0.000583879	0.99	2.9547	0.0006843	0.0316897
5	1.54882	0.000605834	0.99	13.1884	0.0006857	0.0068261
10	1.56717	0.000665733	0.99	31.9516	0.0006730	0.0028232
20	1.58734	0.000743473	1.08038	47.8191	0.0006651	0.0018811
50	1.62223	0.000912408	1.67216	80.7687	0.0006557	0.0011304
100	1.65893	0.001108240	2.32425	121.694	0.0006518	0.0007536
200	1.69338	0.001342180	3.2198	187.091	0.0006465	0.000477
500	1.74642	0.001774670	5.00023	337.972	0.0006184	0.0002617
1000	1.78751	0.002219870	6.95018	548.291	0.0006054	0.0001622
2000	1.83440	0.002861020	9.66055	926.09	0.0005786	9.3e-05
5000	1.90700	0.004243030	14.9521	2074.3	0.0005427	4.07e-05
10000	1.97378	0.006036600	20.6438	4007.59	0.000521	2.22e-05

Compute the mean of the energy deposit (E_f-E_0) in the target