Physics Modeling and Simulation

- **Modeling and simulation of natural processes**
- **Geant4 : a tool for simulating particle interactions with matter**

Follow the course/slides from
B. Chopard et al., coursera lectures, University of Geneva
S. Paltani, Statistical Course for Astrophysicits, University of Geneva
N. Ky Phung lecture , Modeling and mathematical models
Follow presentations on Geant4 package: by M. Verdedi,

Prof. dr hab. Elżbieta Richter-Wąs

Examples of natural processes

- Physics (Fluid mechanics), astrophysics, chemistry, climatology,...
- Environmental sciences (river modeling, Volcano plume)
- Biology: (Tissue growth), pattern on animal skins, cells, organs
- Ecosystems: competition between species, ant behavior, equilibrium between forest and savanna, propagation of epidemia,...
- Finance, social sciences, traffic, pedestrian motion,..
- ...

What is a model?

- This is not an apple just its graphical representation

A model is a simplification of reality that is constructed to gain insights into select attributes of a physical, biological, economic, or social system.
A formal representation of the behaviors of system processes, often in mathematical or statistical terms.
The basis can also be physical or conceptual

What is an eviromental modeling?

- Environmental modeling involves the application of multidisciplinary knowledge to explain, explore and predict the Earth's response to environmental change, both natural and human-induced.
- Role of enviromental modeling
  - Improved understanding of environmental systems.
  - Developing scientific understanding - throughquantitative expression of current knowledge of a system (as well as displaying what we know, this may also show up what we do not know);
  - Test the effect of changes in a system;
  - Aid decision making, including (i) tactical decisions by managers; (ii) strategic decisions by planners.

Type of models

- **Physical modeling**
- **Empirical models**
- **Mathematical models**

Physical models

- Physical modeling is a way of modeling and simulating systems that consist of real physical components. A physical model is a smaller or larger physical copy of an object.
- Spatial analysis and similarity theories are used in this process to ensure that the model results can be extrapolated to the real system with high accuracy.
- Physical modeling is the main approach of scientists in developing basic theories of the natural sciences.

Water dam model

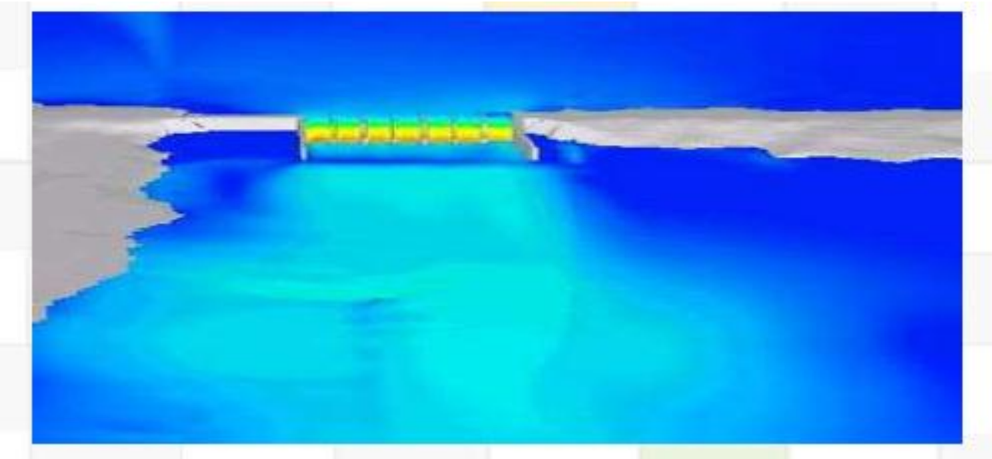
Empirical models

- Empirical models describe observed behaviour between variables on the basis of observations alone and say nothing of process.
- They are usually the simplest mathematical function, which adequately fits observed relationship between variables. No physical laws or assumptions about relationships between variables are required.
- Empirical models have high predictive power but low explanatory depth, they are thus rather specific to the conditions under which data were collected and cannot be generalized easily for application to othe conditions.

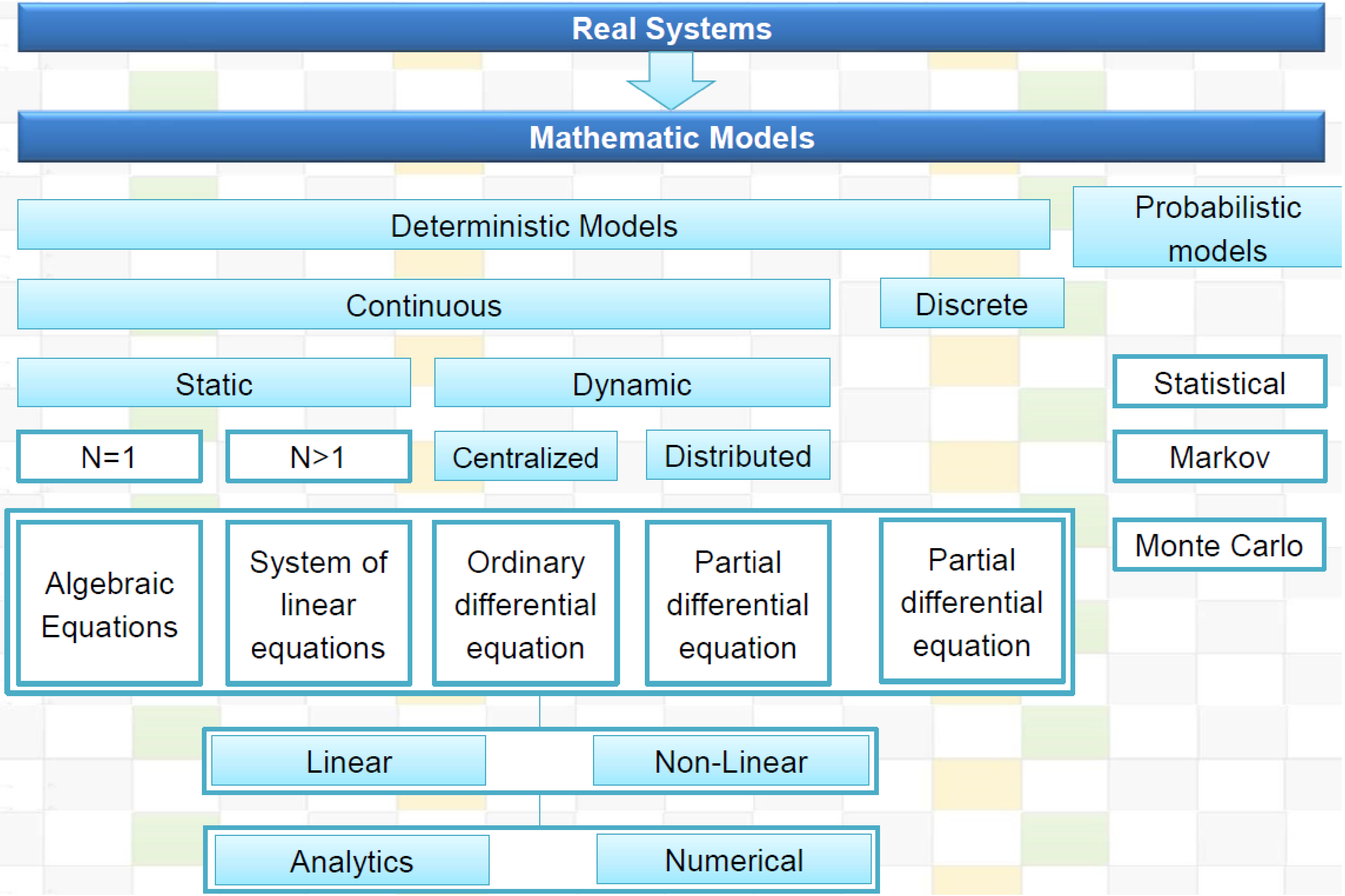
Mathematical models

- Mathematical model is representation of real world problem in mathematical form with some simplified assumptions which helps to understand it in fundamental and quantitative way.
- Are much more common and represent states and rates of change according to formally expressed mathematical rules.
- Can range from simple equations to complex software codes applying many equations and rules over time and space discretization.
- One can further define mathematical models into different types but most models are actually mixtures of many types or are transitional between types.

Dam model is described by mathematical model



Mathematical models



Mathematical models

❖ System and boundary

- A system is a set of one or more related objects, which can be a physical entity with specific properties or characteristics. The system is isolated from its surroundings by boundaries, which can be physical or virtual

❖ Open and Closed, flow/non-flow systems

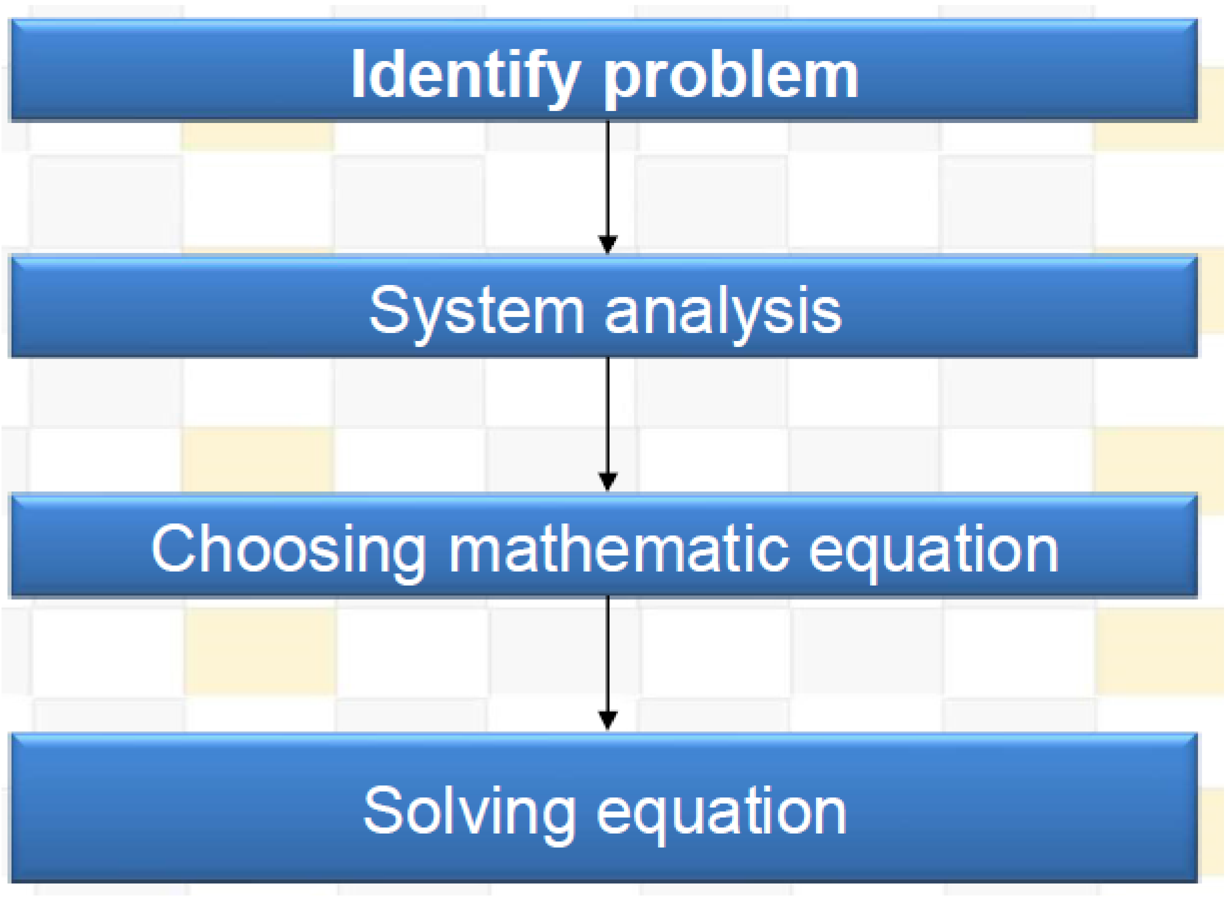
- A closed system is a system that is completely isolated from its environment.
- An open system is a system that has flows of information, energy, and/or matter between the system and its environment, and which adapts to the exchange.
- When the flow of matter does not cross the boundary (but energy can), the system is called a nonflow system. If the material flow can cross the boundary, the system is called a flow system.

Mathematical models

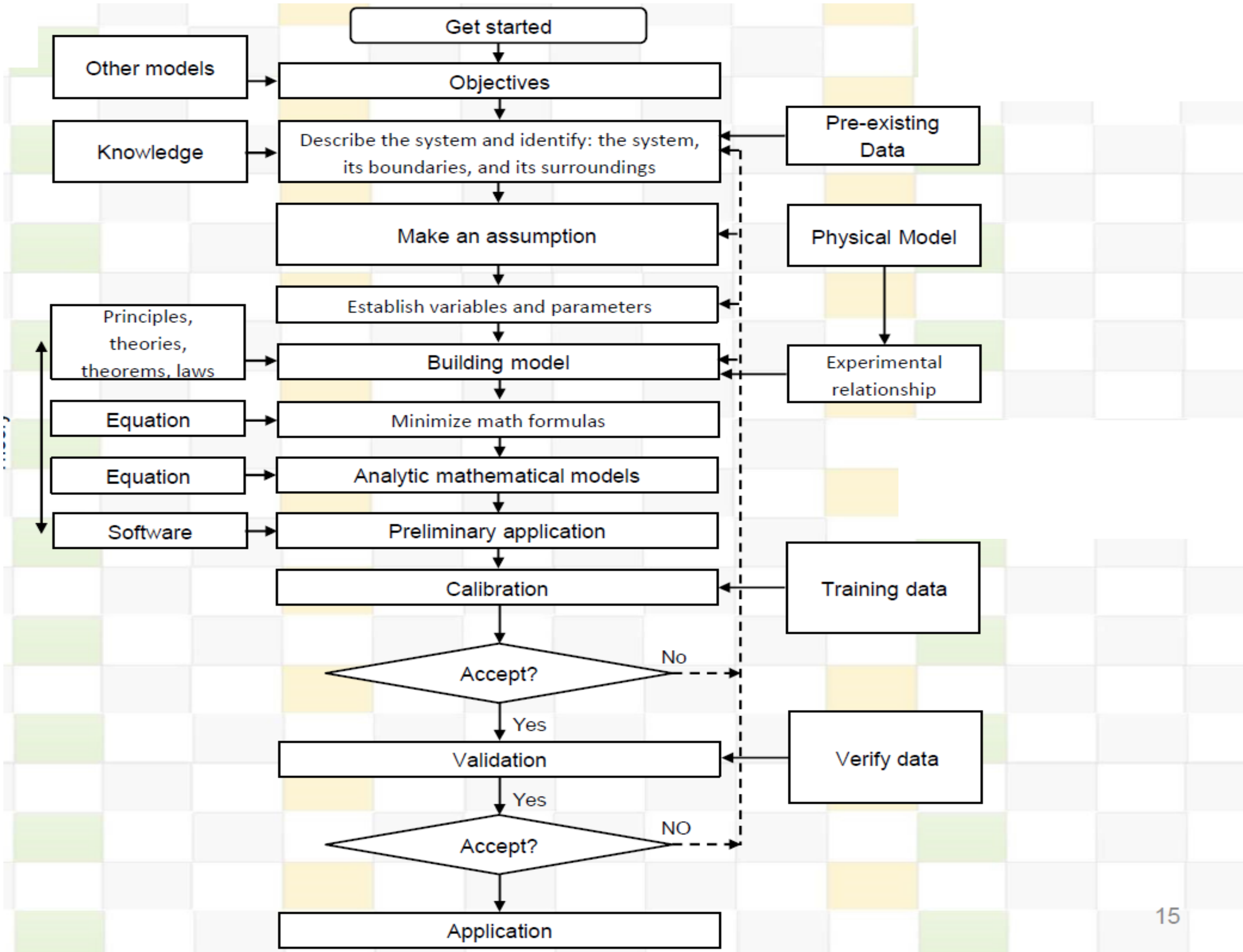
❖ Variable, parameter

- A variable is a value that changes freely in time and space (a compartment or flow) and a state variable is one which represents a state (compartment). A constant is an entity that does not vary with the system under study, for example, acceleration due to gravity is a constant in most Earth-based environmental models (but not in geophysics models looking at gravitational anomalies, for example).
- A parameter is a value which is constant in the case concerned but may vary from case to case where a case can represent a different model run or different grid cells or objects within the same model.

Steps to building mathematical models



Step to building mathematical models



Step to building mathematical models

CALIBRATION AND VALIDATION

- Calibration is **the iterative process of comparing the model with real system**, revising the model if necessary, comparing again, until a model is accepted (validated)
- Validation is a process of comparing the model and its behavior to the real system and its behavior
- Sensitivity analysis is the process of defining how changes in model input parameters affect the magnitude of changes in model output..

Step to building mathematical models

■ Evaluation of simulation results

There are two methods to evaluate model performance

- Graphical method (qualitative)
- Statistical methods (quantitative)

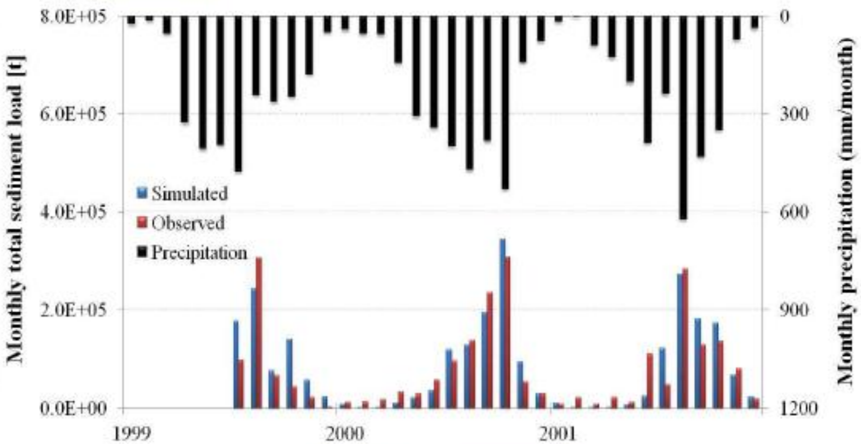
➢ **The calibration model does not represent accurately possibly due to the factors multiply as below:**

- The model is used incorrectly or the model setting is incorrect
- The model is not suitable for this application
- Lack of data to describe the real world
- Measurement data is not reliable

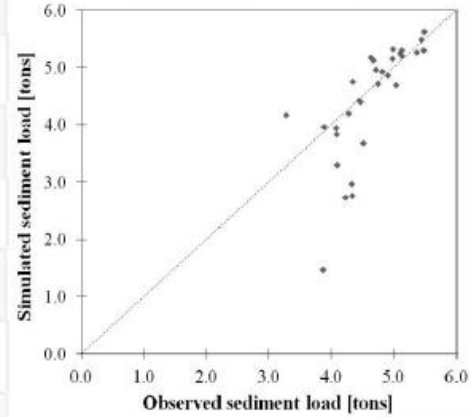
Building mathematical models

■ Evaluation of simulation results

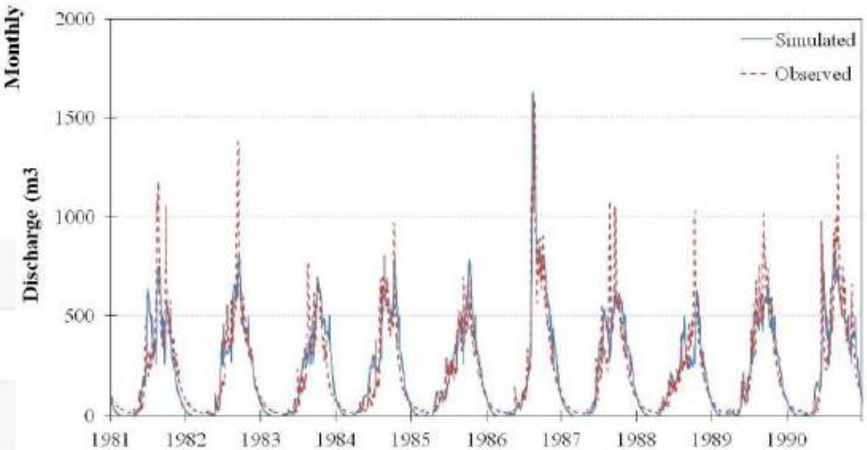
- **Graphical Method**



A. Bar graph



B. Scatter graph



C. Line graph

Building mathematical models

■ Evaluation of simulation results (cont.)

- **Statistical method**

Percent bias – PBIAS : measures the average tendency of the simulated values to be larger or smaller than their observed ones.

PBIAS → 0: indicating accurate model simulation.

$$PBIAS = \left[\frac{\sum_{i=1}^{n}\left(Y_i^{obs} - Y_i^{sim}\right) \times 100}{\sum_{i=1}^{n}\left(Y_i^{obs}\right)}\right]$$

Y_i^{obs} : observed value

Y_i^{sim}: Simulate value

n: total observed value / Simulate

Building mathematical models

Evaluation of simulation results (cont.)

- **Statistical method**

Correlation coefficient formulas R^2 are used to find how strong a relationship is between observed and simulate data.

$$R^2 = \frac{\sum_{i=1}^{n}\left(Y_i^{obs} - \bar{Y}^{obs}\right)\left(Y_i^{sim} - \bar{Y}^{sim}\right)}{\sqrt{\sum_{i=1}^{n}(Y_i^{obs} - \bar{Y}^{obs})^2} \times \sqrt{\sum_{i=1}^{n}(Y_i^{sim} - \bar{Y}^{sim})^2}}$$

$\bar{Y}^{obs}$: average value of the series of observed data

$\bar{Y}^{sim}$: average value of the series of simulated data

Building mathematical models

- **Statistical method**

Nash – Sutcliffe (NSE): is a normalized statistic that determines the relative magnitude of the residual variance compared to the measured data variance (Nash and Sutcliffe, 1970). Nash-Sutcliffe efficiency indicates how well the plot of observed versus simulated data fits the 1:1 line. NSE = 1, corresponds to a perfect match of the model to the observed data. NSE = 0, indicates that the model predictions are as accurate as the mean of the observed data, Inf < NSE < 0, indicates that the observed mean is a better predictor than the model.

$$NSE = 1 - \left[\frac{\sum_{i=1}^{n}(Y_i^{obs} - Y_i^{sim})^2}{\sum_{i=1}^{n}(Y_i^{obs} - \bar{Y}^{obs})^2}\right]$$

Properties	**NSE, R^2**	**PBIAS**	
		Flow	***Water - quality***
Very Good	0.75 → 1.00	< ± 10 %	<± 25 %
Good	0.65→ 0.75	± 10 % → ± 15 %	± 25 % → ± 40 %
	0.50→0.65	± 15 % → ± 25 %	± 40 % → ± 70 %
Unsatisfactory	<0.50	> ± 25 %	> ± 70 %

(Moriasi et al., 2007)

Several models/ different language of description

Partial differential equation for a fluid;

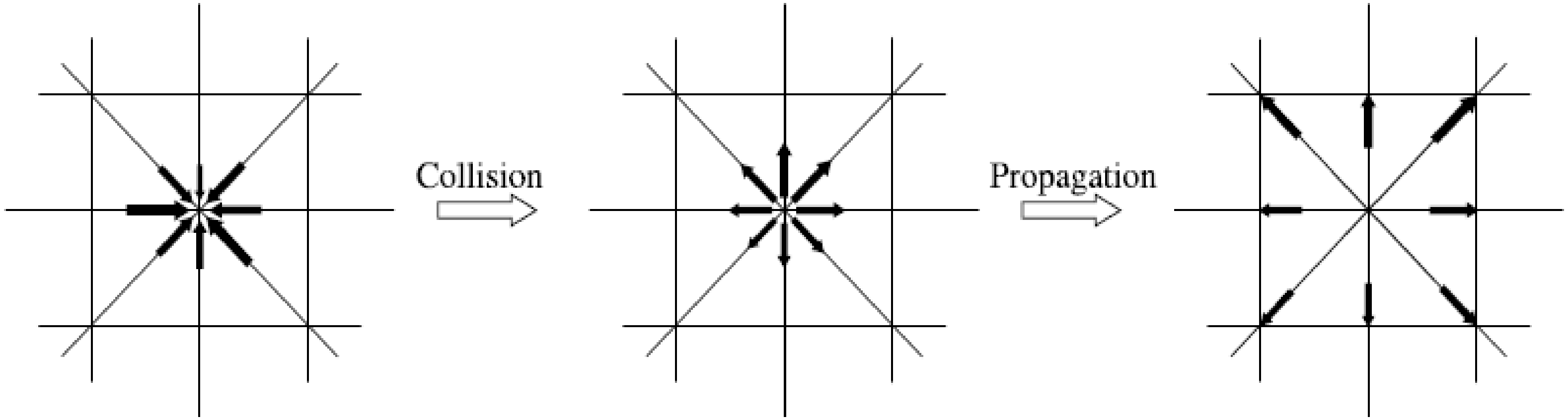
$$\partial_t \mathbf{u} + (\mathbf{u} \cdot \nabla)\mathbf{u} = -\frac{1}{\rho}\nabla p + \nu \nabla^2 \mathbf{u}$$

$\boxed{\textbf{phenomena} \rightarrow \textbf{PDE} \rightarrow \textbf{discretisation} \rightarrow \textbf{numerical solution}}$

…to a virtual model of reality

One considers a discrete universe as an abstraction of the real word

$$\boxed{\textbf{phenomena} \rightarrow \textbf{computer model}}$$



- Mesoscopic *Rule* describing the phenomena

Example of modeling method

- N-body systems, molecular dynamics
- Mathematical equations, ODE, PDE
- Monte-Carlo methods (equilibrium, dynamic, kinetic)
- Cellular Automata and Lattice Boltzmann methods
- Multi-agents systems
- Discrete Events simulation
- Complex networks

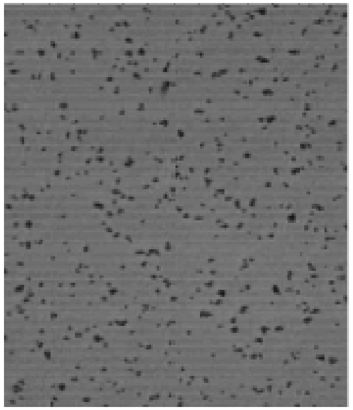
From a model to a simulation

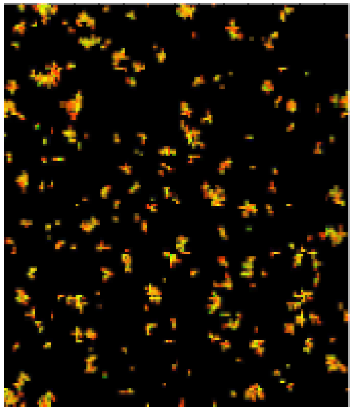
- Once a model is specified, one need to program it, to run it (many times) and to study the results.
- It is a numerical experiment in **computer based virtual universe**
- One need to understand computer programs, software engineering, algorithms, data-structures, hardware (parallel machines, GPUs), code optimization, data-analysis.

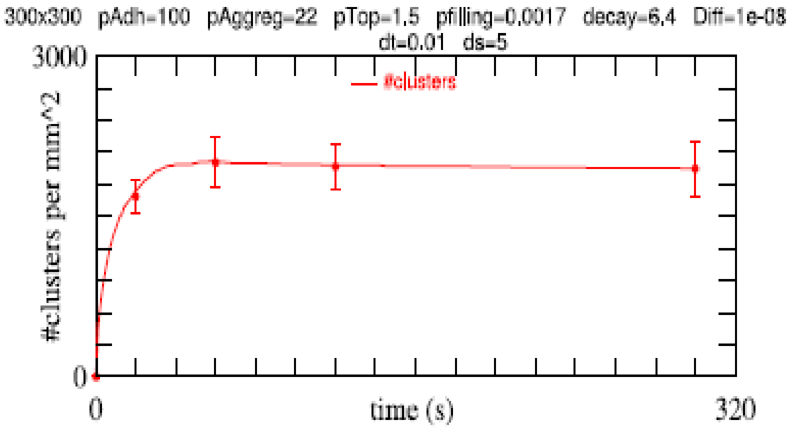
From a model to a simulation

- The program needs to be verified (did we really implemented the model?)
- The model should be validated (run benchmarks with known results).
- One need enough knowledge of the phenomena to judge if its predictions are acceptable in new situations.

From a model to a simulation: illustration







Space and time

- Natural processes occurs in space and evolve over time (spatially extended dynamical systems).
- For instance the atmospheric temperature is different from one place to another, and changes over time.
- Also, a car on a road changes position as time goes on.
- Sometime one is only interested in the time evolution of a quantity, regardless of the spatial location (e.g the number of individuals) in a population
- Sometime, a process is stationary (no time evolution). Then only the spatial variations are of interest (e.g temperature in a room, in the middle or near the windows).

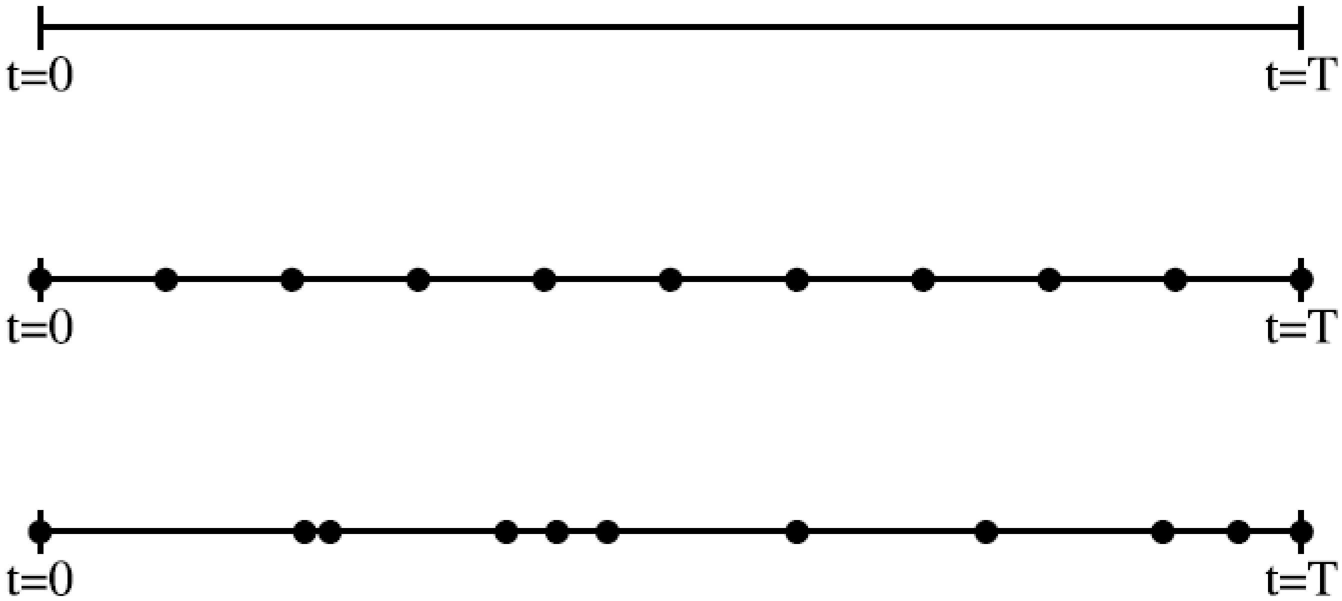
Time evolution

- To capture the temporal dimensions in a model, there are several ways:
- Time takes any real values (as physics suggests). Only mathematical models can deal with this approach (differential equations)
- Otherwise, the duration of the process is broken up in small time intervals Δt and one describes the state of the system at each of these **time-step** $t_0 = 0, t_1 = \Delta t, \ldots, t_n = n\Delta t \ldots$.
- The time is discretized, but the process is followed **continuously** over its duration.

Time evolution

- Alternatively, we can only focus on the interesting moments of a process
- In a queue in front of a post office booth, one can simply consider the time at which a remarkable event occurs. For instance a new customer enters, or a previous one is done.
- The time t at which an event occurs can be any real value.
- The time is not discretized but the evolution of the system is broken up according to events.
- This is the so-called Discrete-Event-Simulation (DES) approach

Time evolution



Modeling space: Eurelian approach

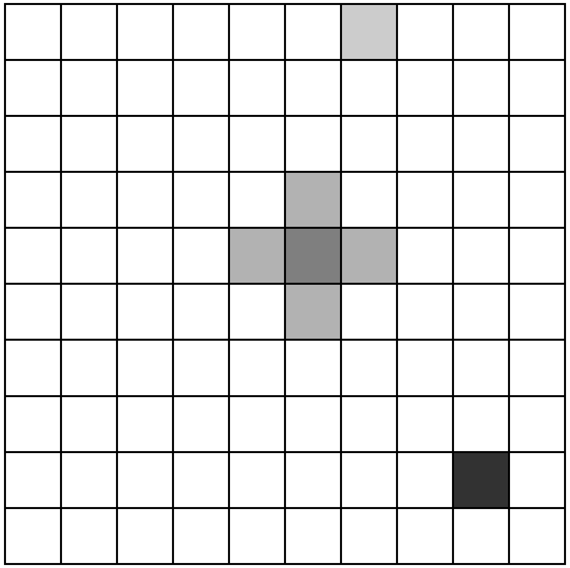
To include the spatial dimensions in a model, there are also different ways.

- One can take the point of view of an observer who sits at a fixed position $\vec{x}$ in space and records what he sees.
- For instance the local atmospheric pressure $p(\vec{x}, t)$.
- Or the number of cars that passed by every minute.
- This is the so-called **Eulerian** approach: attach a property of the system at each spatial locations.
- Space can be continuous (mathematical models) or discretized in cells, forming a mesh covering the region of interest.

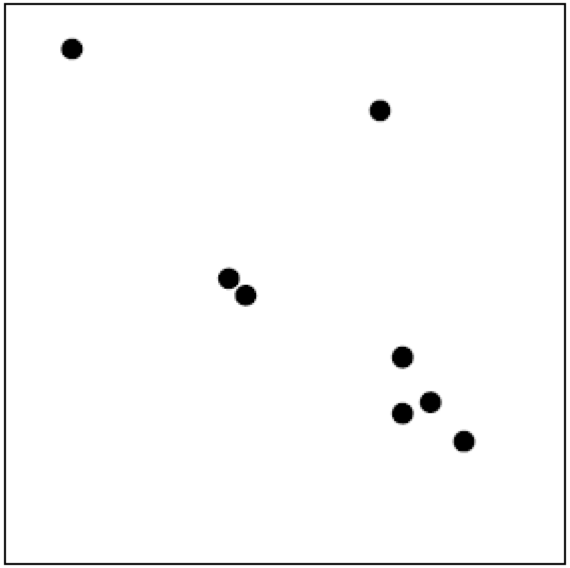
Modeling space: Lagrangian approach

- Alternatively, one can give the position of all the objects of interest, as a function of time.
- For instance the movement of the Moon is described by its trajectory $\vec{x}(t)$, where $\vec{x}$ is a continuous variable.
- In a traffic model, one can give the positions over time of all the cars.
- This is the so-called **Lagrangian** approach: the observer take the point of view of the moving objects.

Modeling space



Eulerian point of view



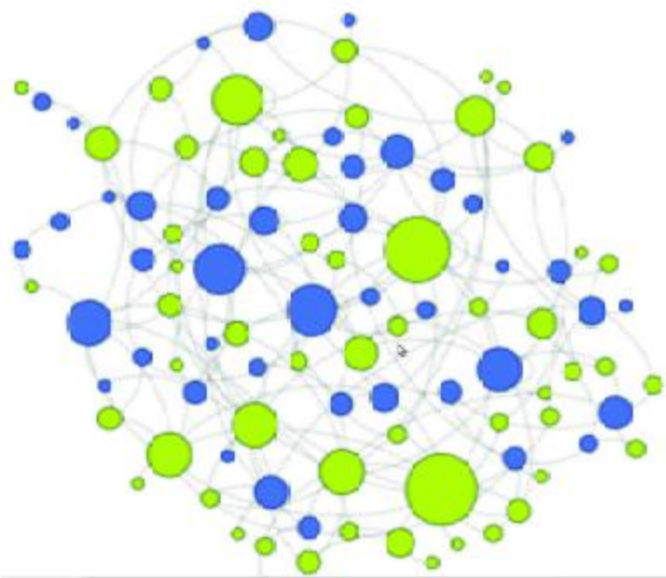
Lagrangian point of view

Beyond the physical space: complex networks

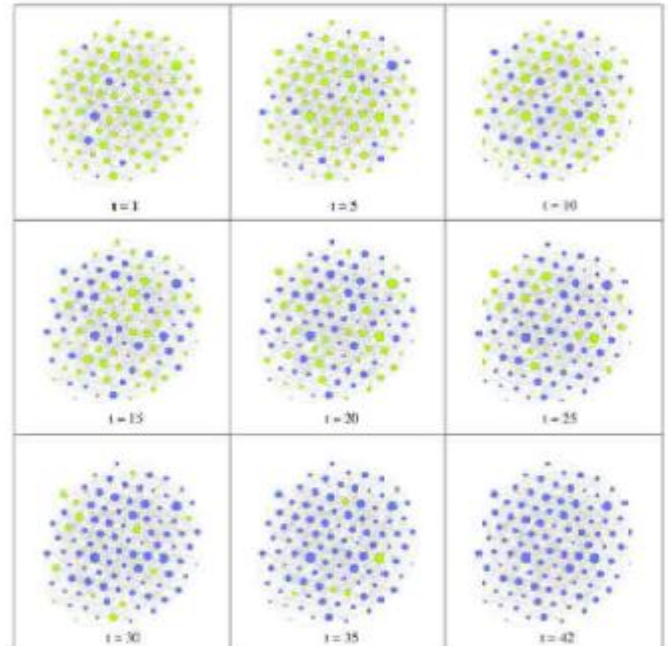
- In many systems, it is not so much the exact spatial positions of the components of a system that matters
- It is rather whether these components see each others, or can interact.
- This is typically the case in social systems. Two persons can be very far away but still interact a lot by phone or other means
- For instance, the agent in an economical model can be represented as a **graph** (or a complex network): an edge connects pairs of agents that exchange information, money, goods,..
- Obviously, such a graph can be dynamical: creation of new links or destruction of old ones.

Example

A model of opinion propagation in a social network



(Lino Velasquez, UNIGE)



Réseau aléatoire, $N = 100$, $p = 0.05$, $\epsilon = 0.3$

Complex networks

- Dynamical systems on complex networks is a fast developing field
- Graph topology imposes a rich "spatial" structure which constrains the dynamics
- Many quantities characterize the graph topology and can be related to some global properties of the system: degree distribution, clustering coefficient, centrality measures, assortativity, etc.

Monte Carlo methods

- The goal of Monte-Carlo methods is the sampling of a process in order to determine some statistical properties
- For instance, we toss a coin 4 times. What is the probability to obtain 3 tail and 1 head?
- Mathematics gives us the solution:

$$P(3\text{ head}) = \binom{4}{3}\left(\frac{1}{2}\right)^3\left(1-\frac{1}{2}\right)^1 = \frac{1}{4}$$

- But we could also do a simulation

A Monte Carlo computer simulation

```
from random import randint

success=0

attempts=10000
for i in range(attempts):
    if randint(0,1)+randint(0,1)+randint(0,1)+randint(0,1)==3:
        success+=1

print "Number of attempts=", attempts
print "Number of success=", success
```

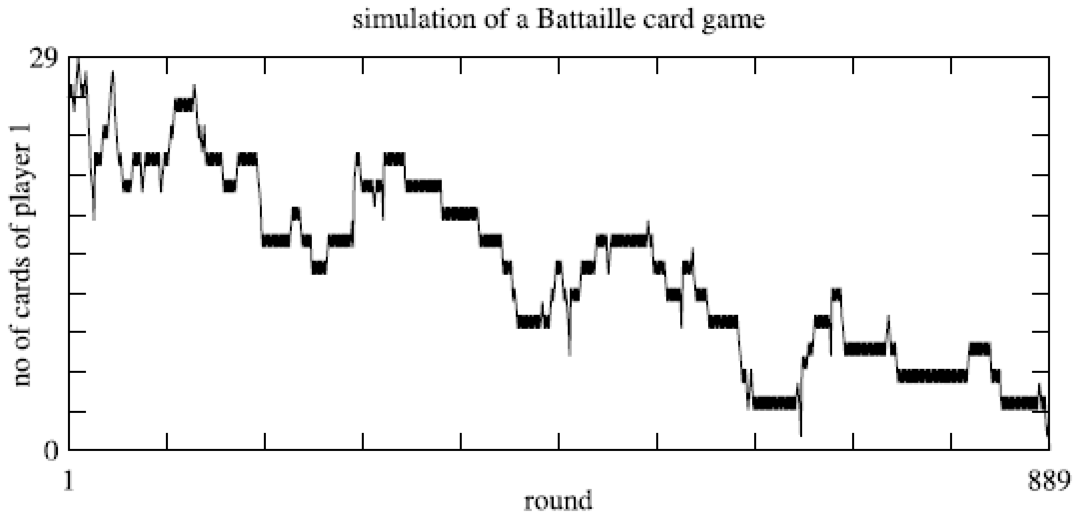
We get for instance:

```
Number of attempts= 10000
Number of success= 2559
```

A more difficult problem

- For the coin tossing problem, no need for a simulation
- But we can think of other problems for which probability theory could hardly be applied
- For instance: what is the average duration of the card game called "war" (or battle)?

The *war* card game with 52 cards

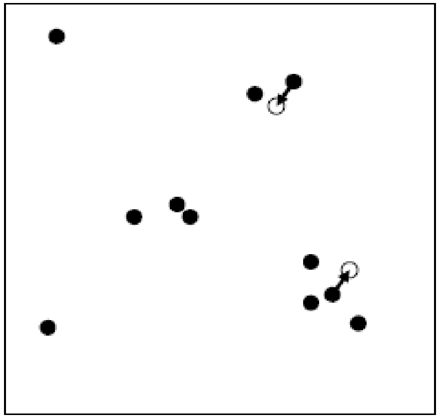


Historical note

- The method was name in the 1940s by **John von Neumann**, **Stanislaw Ulam** and **Nicholas Metropolis** after the name of the *Monte-Carlo casino*, where Ulam's uncle used to gamble ...and loose his money
- The motivation was to find out the probability that a Canfield solitaire will finish sucessfully.
- Ulam found it easier to play many Canfield solitaires and estimate the number of successes, rather than trying to apply combinatorics and probability theory.
- Then the Monte-Carlo methods was successfully applied to the *Manhattan project* (nuclear weapon) in the Los Alamos National Laboratory.

Markov-chain Monte Carlo (MCMC)

- We consider a stochastic process whose goal is to explore the state space of a system of interest.
- Let x be a point in this state space. Let us asssume that this point moves across the space by jumping randomly to another point x'.
- The jump from location x to location x' takes place with probability $W_{x \to x'}$. This advanced the system time from t to $t+1$ (Markov chain)



- We want this process to sample a prescribed probability $\rho(t, x)$. This stochastic process should be at point x at time t with a probability $\rho(t, x)$.
- How do we choose $W_{x \to x'}$?

$$\rho \propto \exp(-E(x)/k_B T)$$

Sampling the diffusion equation in 1D

The probability that our random exportation is at location x at time t is

$$p(t+1,x) = \sum_{x'} p(t,x')W_{x'\to x}$$

- Let us consider a 1D discrete space: $x \in \mathbf{Z}$.
- where one can move to the right with probability W_+, to the left with probability W_- and stay still with probability W_0.
- The equation for $p(t,x)$ simplifies to

$$p(t+1,x) = p(t,x-1)W_+ + p(t,x)W_0 + p(t,x+1)W_-$$

Diffusion equation in 1D

- The diffusion equation is $\partial_t \rho = D \partial_x^2 \rho$
- Which can be discretized as

$$\rho(t + \Delta t, x) = \rho(t, x) + \frac{\Delta t D}{\Delta x^2} \left(\rho(t, x - 1) - 2\rho(t, x) + \rho(t, x + 1) \right)$$

- to be compared with

$$p(t + 1, x) = p(t, x - 1) W_+ + p(t, x) W_0 + p(t, x + 1) W_-$$

- In order to have $p = \rho$, one need $W_+ = W_- = \Delta t D / (\Delta x)^2$ and $W_0 = 1 - 2\Delta t D / (\Delta x)^2 = 1 - W_+ - W_-$, and thus $\Delta t D / (\Delta x)^2 \leq 1/2$
- Therefore a random walk is a way to sample a density ρ that obeys the diffusion equation.
- With a random walk, it is easy to add obstacles, or aggregation processes, hard to include in the differential equation.

More general case: Master equation

The probability to find the random exploration at location x at time t is $p(t,x)$ given by

$$
\begin{aligned}
p(t+1,x) &= \sum_{x'} p(t,x')W_{x'\to x} \\
&= \sum_{x'\neq x} p(t,x')W_{x'\to x} + p(t,x)W_{x\to x} \\
&= \sum_{x'\neq x} p(t,x')W_{x'\to x} + p(t,x)(1-\sum_{x'\neq x} W_{x\to x'}) \\
&= p(t,x) + \sum_{x'\neq x} [p(t,x')W_{x'\to x} - p(t,x)W_{x\to x'}]
\end{aligned}
$$

Detailed balance

In a steady state, the condition $p(x) = \rho(x)$ requires that

$$\sum_{x' \neq x} [\rho(x')W_{x' \to x} - \rho(x)W_{x \to x'}] = 0$$

We can then choose $W_{x \to x'}$ according to the **detailed balance** condition

$$\rho(x')W_{x' \to x} - \rho(x)W_{x \to x'} = 0$$

Metropolis Rule

Let us consider a physical system at equilibrium whose probability to be in state x is given by the Maxwell-Boltzmann distribution

$$\rho(x) = \Gamma \exp(-E(x)/kT)$$

We can sample this distribution with a stochastic process by choosing $W_{x \to x'}$ according to the **Metropolis rule**:

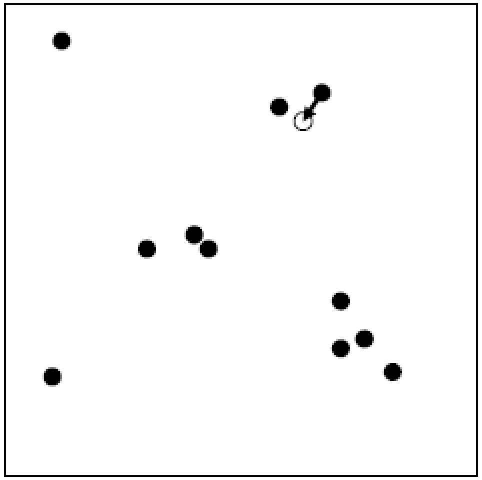
$$W_{x \to x'} = \begin{cases} 1 & \text{si } E' < E \\ \exp[-(E' - E)/kT] & \text{si } E' > E \end{cases}$$

Metropolis Rule in practice

- In a gas, one selects one particle at random.
- One moves it by an amount Δx.
- One computes the energy E' of the gas with this new position.
- One accepts this change if

$$\mathrm{rand}(0,1) < \min(1, \exp[-(E' - E)/kT])$$

- By sampling ρ with $W_{x \to x'}$, one can compute average physical properties, such as for instance the pressure in the gas.



Metropolis Rule in practice

The Metropolis obeys the detailed balance

Let us assume that $E' > E$. Detailed balance is obeyed because

$$\begin{aligned}
\rho(x)W_{x\to x'} &= \Gamma \exp(-E/kT)\exp[-(E'-E)/kT] \\
&= \Gamma \exp(-E'/kT) \\
&= \rho(x') \times 1 \\
&= \rho(x')W_{x'\to x}
\end{aligned}$$

And similarly if $E' \leq E$

Glauber Rule

This is an alternative to the Metropolis rule. $W_{x \to x'}$ is given by

$$W_{x \to x'} = \frac{\rho(x')}{\rho(x) + \rho(x')}$$

which also clearly obeys detailed balance
With $\rho = \Gamma \exp(-E(x)/kT)$, one obtains

$$W_{x \to x'} = \frac{\exp(-E'/kT))}{\exp(-E/kT) + \exp(-E'/kT)}$$

Kinetic/Dynamic Mote Carlo

Let us consider the chemical equations

$$A \xrightarrow{k_1} B \qquad B \xrightarrow{k_2} A$$

They can be written as an ordinary equation

$$\frac{d}{dt}\begin{pmatrix} A \\ B \end{pmatrix} = \begin{pmatrix} -k_1 & k_2 \\ k_1 & -k_2 \end{pmatrix}\begin{pmatrix} A \\ B \end{pmatrix}$$

Analytical solution

$$A(t) = \frac{k_2}{k_1+k_2}(A_0+B_0) + \frac{A_0k_1 - B_0k_2}{k_1+k_2}e^{-(k_1+k_2)t}$$

$$B(t) = \frac{k_1}{k_1+k_2}(A_0+B_0) - \frac{A_0k_1 - B_0k_2}{k_1+k_2}e^{-(k_1+k_2)t}$$

where A_0 and B_0 are the initial concentration of A and B.
When $t \to \infty$,

$$A \to A_\infty = \frac{k_2}{k_1+k_2}(A_0+B_0) \qquad B \to B_\infty = \frac{k_1}{k_1+k_2}(A_0+B_0)$$

Monte Carlo simulation

1 One defines a time step Δt, small enough so that $k_1 \Delta t$ et $k_2 \Delta t$ are smaller than 1. They are the **probabilities** that, during Δt, one A particle get transformed into one B particle, or conversely.

2 One chooses randomly a particle among the $N = A(t) + B(t) = \text{const}$ of them. (In practice one chooses a A particle $\text{rand}(0,1) < A/(A+B)$, and a B particle otherwise.

3a If a A particle was chosen, it is transformed into a B particle provided $\text{rand}(0,1) < k_1 \Delta t$. Then $A = A - 1$, $B = B + 1$.

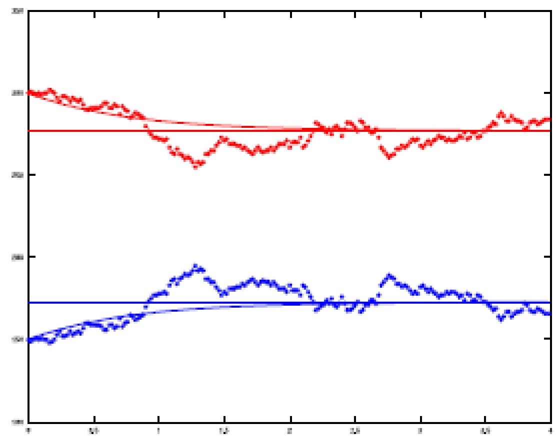
3b If a B particle was chosen, it is transformed into a A particle, provided $\text{rand}(0,1) < k_2 \Delta t$. Then $A = A + 1$, $B = B - 1$.

4 (2) and (3) are repeated N times and the physical time t is incremented by Δt: $t = t + \Delta t$

5 One repeats (2)-(4) until $t = t_{max}$

Monte Carlo simulation

Results



$\Delta t = 0.02$ and $k_1 = 0.5,\ k_2 = 0.8.$

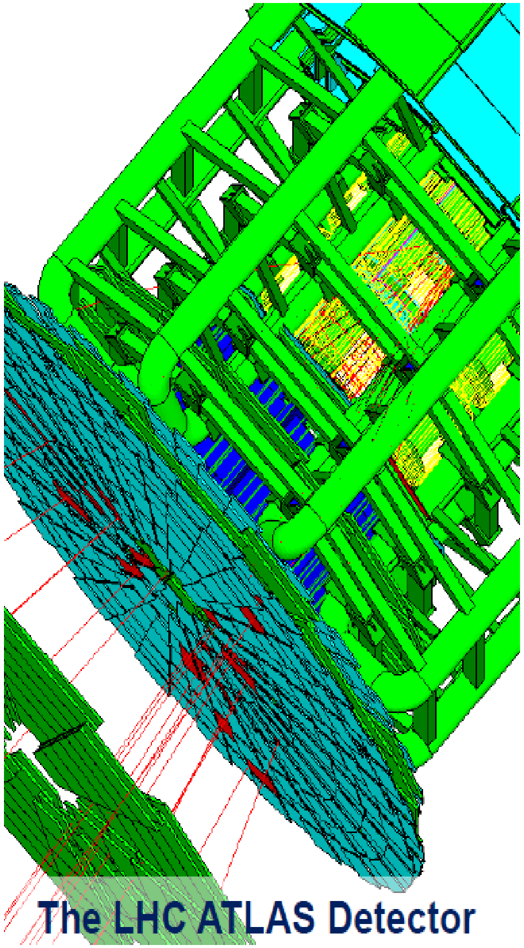
The Monte-Carlo simulation fluctuate around analytic solution.

We should average over several runs

What is Geant4?

Geant4 is a Software Toolkit

- **Geant4 is an Object Oriented (using C++17) Monte Carlo particle transport software toolkit for simulating the passage of elementary particles through matter and interacting with it.**
- It started in **1994** as the CERN **RD44** project :
  - **Goal of RD44 : assess the benefit of OO technologies for detector simulation for LHC era** (LHC yet to come at that time ! FORTRAN was the programming language !)
    - Medical and space domains requests included since the beginning !
  - **Geant4 v1.0 released in Dec 1998**
    - After alpha release in Apr 1997 and beta one in Jul 1998
- **Key functionalities:**
  - Kernel → to **manage & animate** the system
  - Geometry + navigation & materials → to **describe** the setup
  - Physics processes & tracking → to **generate** the series of **physics** interactions
    - EM (O(100 eV) – PeV), special extensions (O(eV) & O(mK)), hadronic (rest - multi-TeV)
  - Scoring → to **collect data** from the simulation
  - GUI and Visualization drivers → to **pilot** the application and **visualize**
- **"Toolkit"** because users **select** components and **build** their application
  - Not an application like ROOT, or Powerpoint, etc.
- **Users can extend the toolkit !**



The LHC ATLAS Detector

What is Geant4?

Geant4 is a Collaboration


Some of the members, at last Collaboration Meeting, in Rennes (2022)

- **Geant4 is also the name of the Collaboration maintaining, developing and validating the software**
  - ~130 members + O(10) "contributors" = new light status
  - ~30 FTE
  - ~30 institutes, worldwide
    - (Map of collaborative institutes after)
  - 16 working groups
- **Web site:**
  - http://geant4.cern.ch/
  - **Download area,** documentation, news, announcement of releases, meetings (**Technical Forum**, etc.)

- **Distributed development model:**
  - Based on **GitLab** (geant4-dev repo.)
    - Reserved to members & contributors
  - About 1000 Merge Requests / year
- **Distribution through:**
  - Geant4 Web site
  - **GitHub** instance
    - GitLab mirror for public releases & patches
    - **Open to public for Pull Requests**
  - Special way, CVMFS, for LHC experiments (monthly tag)
- **One public –major or minor– release/year, in December**
  - **+ patches**, as they come
  - Latest release: **Geant4-11.1.2**

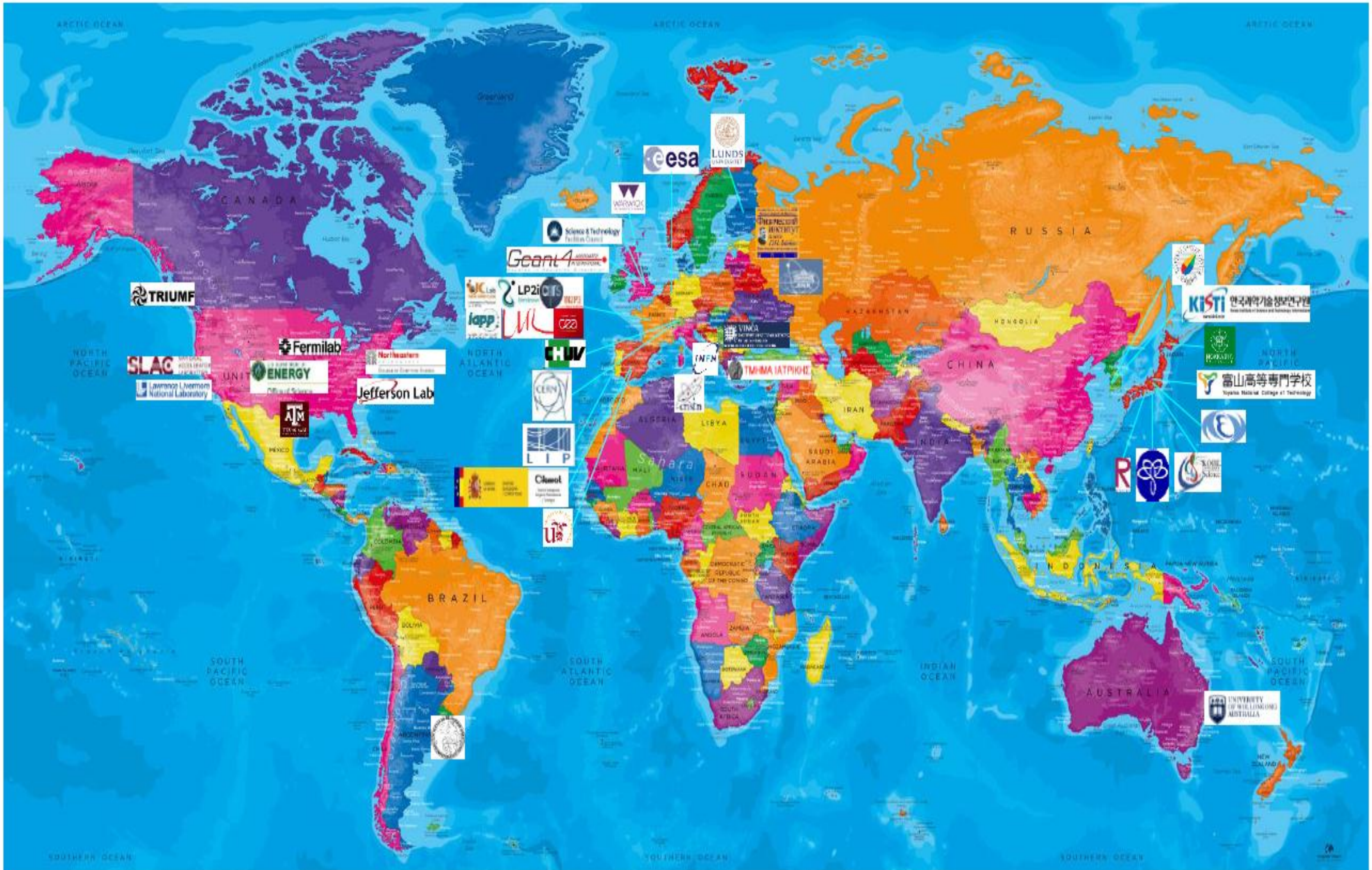
Major release #

Minor release #

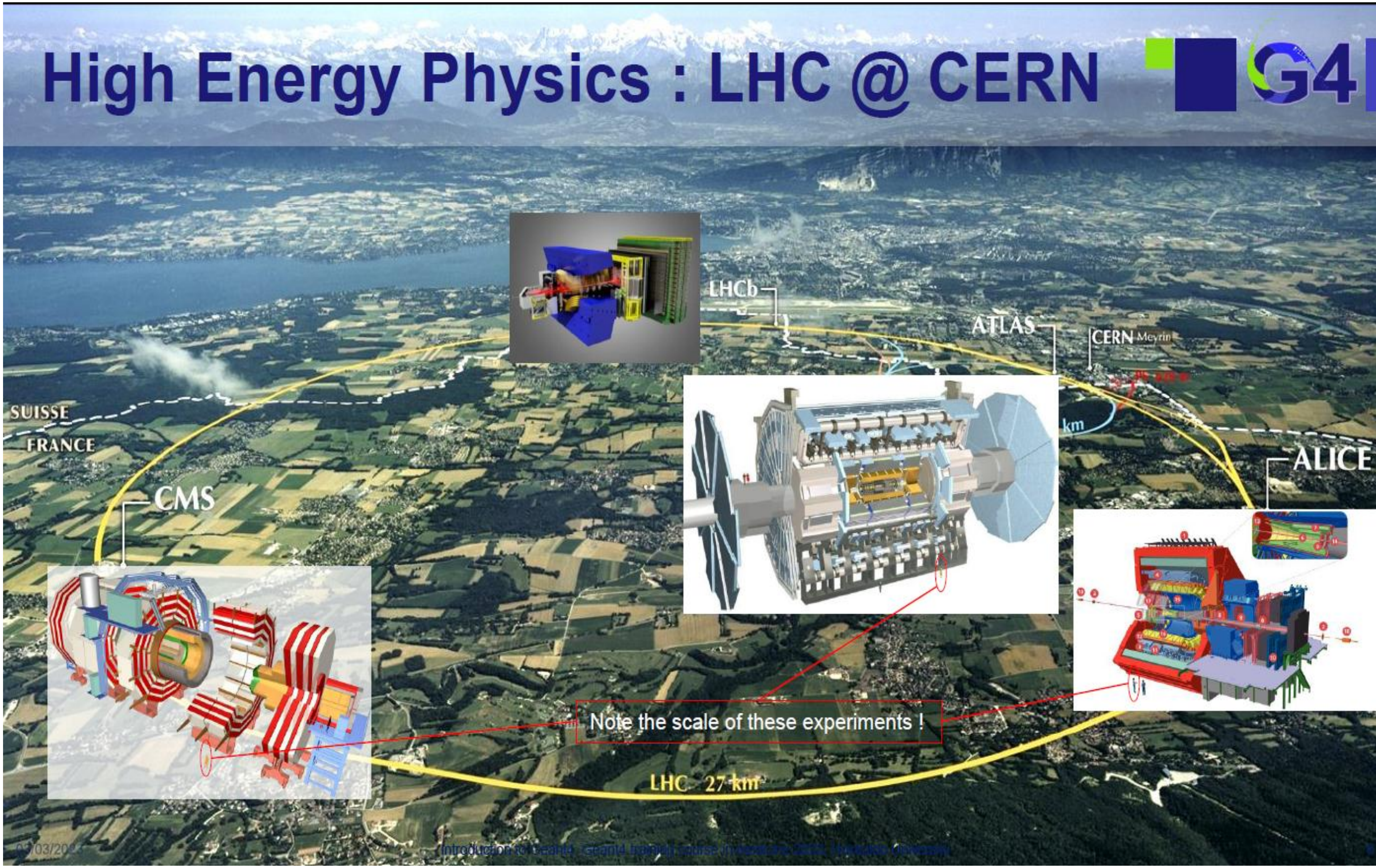
Patch #

- **Three general papers:**
  - **"Geant4: a simulation toolkit"**, S. Agostinelli *et al.*, NIM A, vol. 506, no. 3, pp. 250-303, 2003
  - **"Geant4 Developments and Applications"**, J. Allison *et al.*, IEEE TNS, vol. 53, no. 1, pp. 270-278, 2006
  - **"Recent Developments in Geant4"**, J. Allison *et al.*, NIM A, vol. 835, pp. 186-225, 2016

Map of collaborative Institutes



Geant4 Application Domains

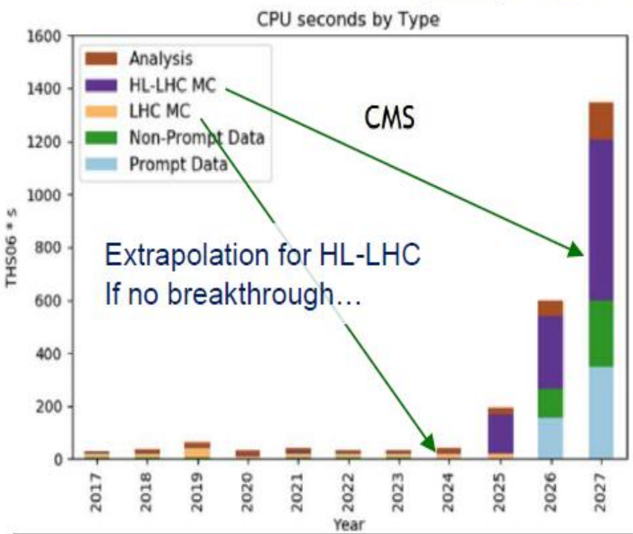


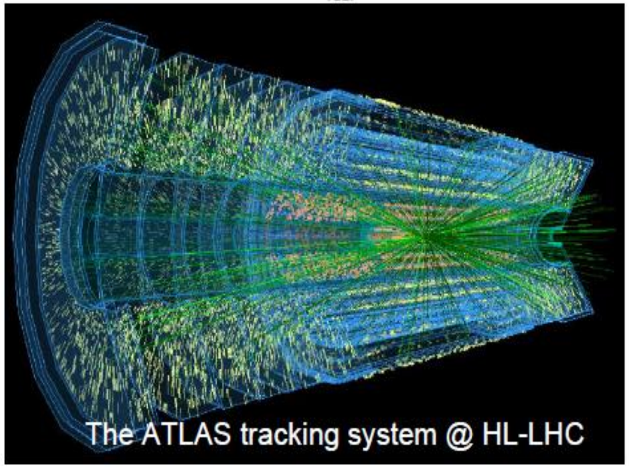
Geant4 Application Domains

High Energy Physics

- LHC experiments are **very demanding** in terms of simulation
  - **Large detectors :**
    - O(1 – 10) millions of volumes
  - **High energy O(10 TeV) in center of mass:**
    - Lead to MANY tracks per event (O(10 k) lead-lead collisions !)
  - **Long processing time and huge production volume:**
    - From O(1 s) to O(1 mn) per event !
    - O(10^9) events processed / experiment !
    - **Each % CPU improvement saves a lot of money…**

- **New phase HL-LHC (~2027) even more challenging !**
  - **Request for O(10) times higher throughput !**
    - With **better physics** (to not inflate syst. errors wrt to stat. errors !)
    - With **more+++ complex detectors** (high granularity calorimetry)
  - **Triggers quite R&Ds activities:**
    - Geant4 on GPU : not trivial at all !
    - ML-based fast simulation
    - Etc.



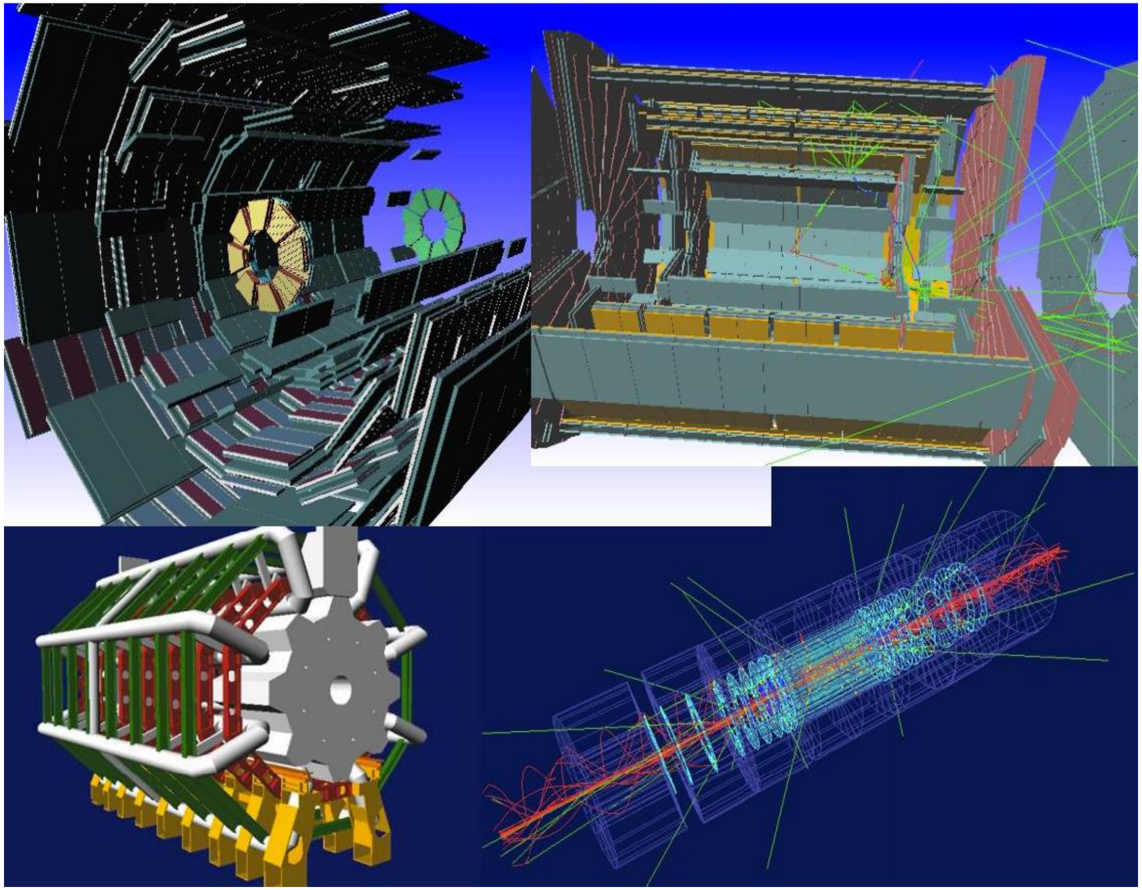


Geant4 Application Domains

Geant4 has been successfully employed in many HEP experiments

- Detector design
- Calibration / alignment
- First analyses

Geant4 in High Energy Physics (ATLAS at LHC)



Simulation

- Simulation plays a fundamental role in various domains and phases of an experimental physics project
  - design of the experimental set-up
  - evaluation and definition of the potential physics output of the project
  - evaluation of potential risks to the project
  - assessment of the performance of the experiment
  - development, test and optimization of reconstruction and physics analysis software
  - contribution to the calculation and validation of physics results
- The scope of Geant4 encompasses the simulation of the passage of particles through matter
- There are other kinds of simulation components, such as physics event generators, electronics response generation, etc.
- Often the simulation of a complex experiment consists of several of these components interfaced to one another

Basic requirements for simulation system

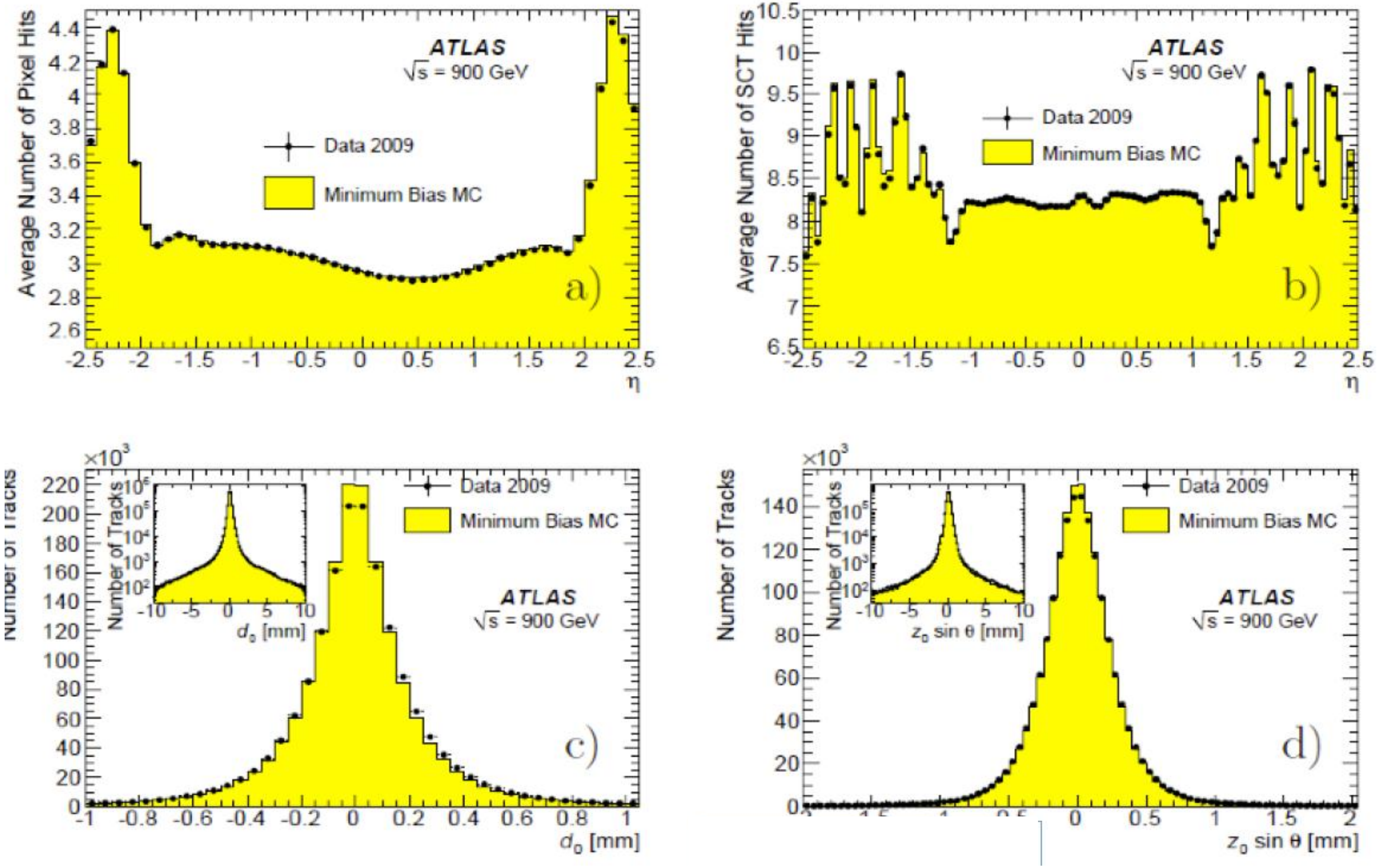
- Modeling the experimental set-up
- Tracking particles through matter
- Interaction of particles with matter
- Modeling the detector response
- Run and event control
- Accessory utilities (random number generators, PDG particle information etc.)
- Interface to event generators
- Visualisation of the set-up, tracks and hits
- User interface
- Persistency

Detector simulation

- General characteristics of a detector simulation system
  - You specify **the geometry** of a particle detector
  - Then the software system automatically transports the particle you shoot into the detector by simulating the particle interactions in matter based on the Monte Carlo method

- The heart of the simulation: the **Monte Carlo method**
  - A method to search for solutions to a mathematical problem using a **statistical sampling with random numbers**

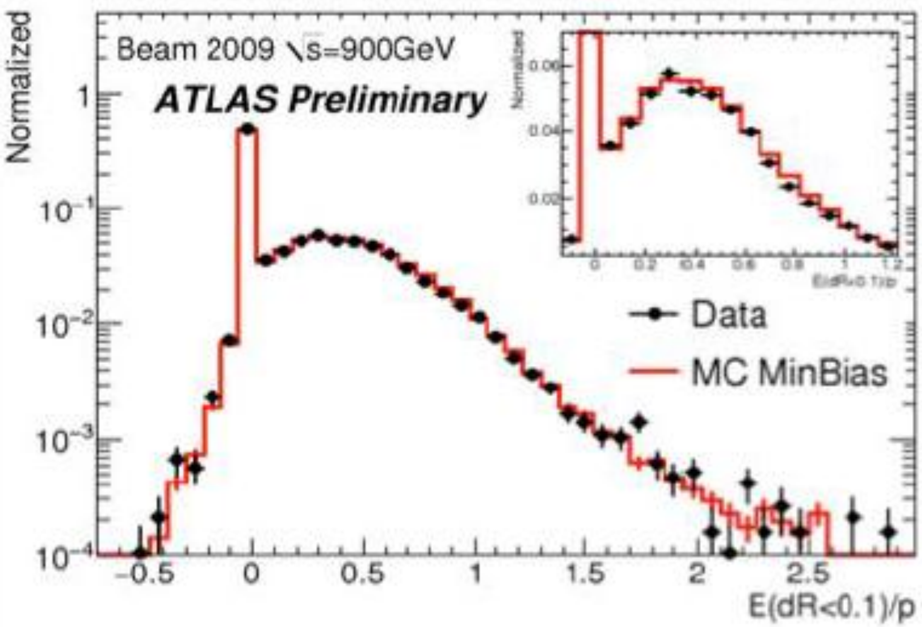
Data and Simulation agreement

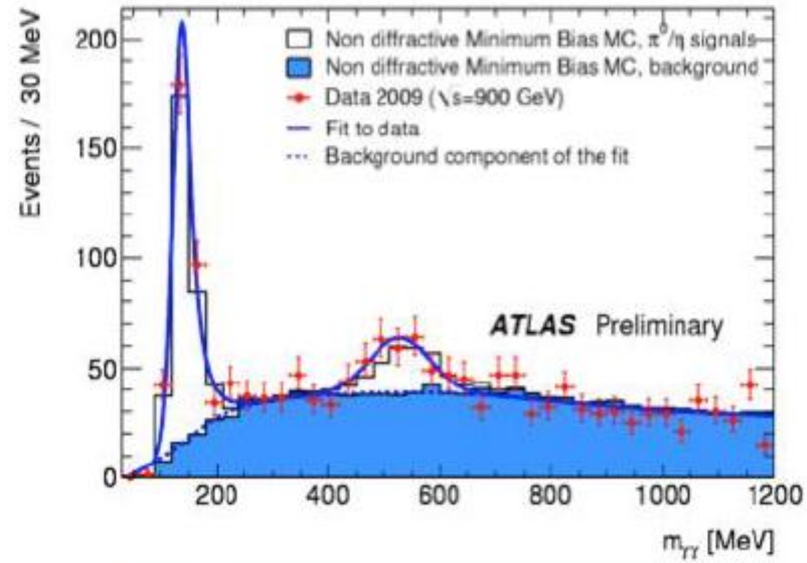


Geant4 Comparison with Calorimeters

Response of the calorimeter to single isolated tracks. To reduce the effect of noise, topological clusters are used in summing the energy.

This plot agreed better than we ever expected. (I sent the student who made it back to make sure that they didn't accidentally compare G4 with G4.





Invariant mass of pairs of well-isolated electromagnetic clusters.

The π^0 mass is within 0.8 ± 0.6% of expectations.

The η^0 mass is within 3 ± 2% of expectations.

The detector uniformity is better than 2%.

Physics in Geant4

- It is rather unrealistic to develop a uniform **physics model** to cover wide variety of particles and/or wide energy range.
- Much wider coverage of physics comes from mixture of **theory-driven, parameterized, and empirical formulae**. Thanks to polymorphism mechanism, both cross-sections and models (final state generation) can be combined in arbitrary manners into one particular process.
  - Standard EM processes
  - Low energy EM processes
  - Hadronic processes
  - Photon/lepton-hadron processes
  - Optical photon processes
  - Decay processes
  - Shower parameterization
  - Event biasing technique

Physics in Geant4

- Each cross-section table or physics model (final state generation) has its own applicable energy range. Combining more than one tables / models, one physics process can have enough coverage of energy range for wide variety of simulation applications.
- Geant4 provides **sets of alternative physics models** so that the user can freely choose appropriate models according to the type of his/her application.
- Several individual universities / physicists groups are contributing their physics models to Geant4. Given the modular structure of Geant4, developers of each physics model are well recognized and credited.

What Geant4 can do for you?

- Transports a particle step-by-step by taking into account the interactions with materials and external electromagnetic fields until the particle
  - loses its kinetic energy to zero,
  - disappears by an interaction,
  - comes to the end of the simulation volume

- Provides a way for the user to access the transportation process and grab the simulation results
  - at the beginning and end of transportation,
  - at the end of each stepping in transportation,
  - at the time when the particle is going into the sensitive volume of the detector
  - etc.
  - These are called "User Actions"

What you have to do for Geant4?

- Three essential information you have to provide:
  - **Geometrical information** of the detector
  - Choice of **physics processes**
  - **Kinematical information of particles** going into the detector

- Auxiliary you have to prepare:
  - Magnetic and electric field
  - Actions you want to take when you access the particle transportation
  - Actions you want to take when a particle goes into a sensitive volume of the detector
  - etc.

Tools for input preparation

- Geant4 provides standard tools to help you to prepare input information
- Multiple choices to describe the detector geometry
  - Combining basic geometry elements (box, cylinder, trapezoid, etc)
  - Representation by surface planes
  - Representation by boolean operation, etc.
- Standard way to define materials in the detector
  - A large collection of examples to define various materials
- A set of wide variety of particles
  - Standard elementary particles (electron, muon, proton,....)
  - Unstable particles (resonances, quarks, ...)
  - Ions
  - Exotic particles (geantino, charged geantino)

Minimum software knowledge to use Geant4

- C++
  - Geant4 is implemented in C++, therefore a basic knowledge of C++ is mandatory
  - C++ is a complex language, but you are not required to be a C++ expert to use Geant4
- Object Oriented Technology
  - basic concepts
  - in-depth knowledge needed only for the development of complex applications
- Unix/Linux
  - Unix/Linux is a standard working environment for Geant4, therefore a minimum knowledge/experience is required
    - How to use basic Unix command
    - How to compile a C++ code
  - Windows
    - You can use Visual C++
    - Though still you need some knowledge of Unix (cygwin) for installation

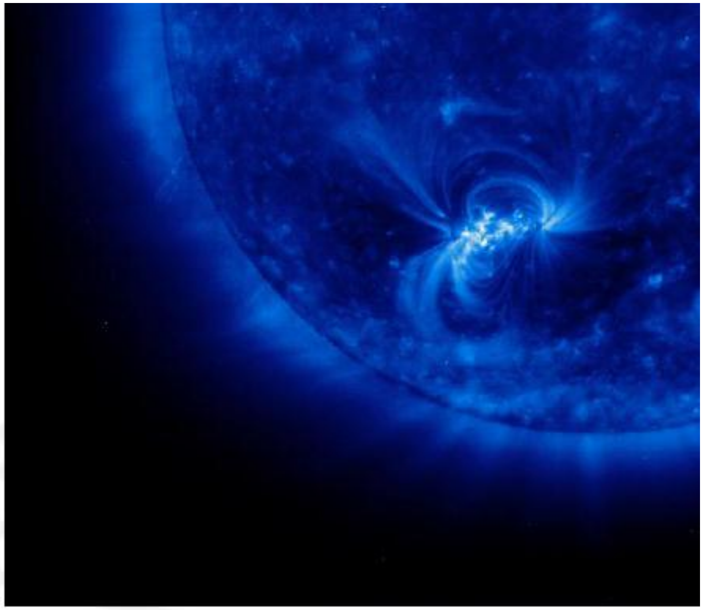
Tools to help your simulation with Geant4

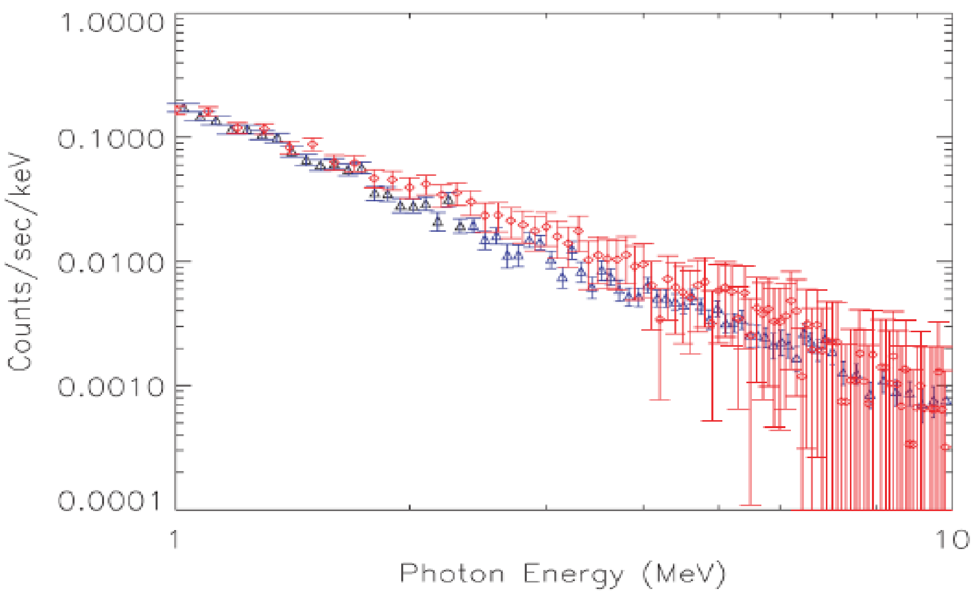
- User interface
  - Interactive mode with terminal or GUI
  - Batch mode
- Visualisation
  - Trajectory of a particle and its all secondaries
  - Detector geometry
- Debugging
  - Controllable verbose outputs from the kernel during transportation
  - Errors in the geometry definition, etc.
- Data analysis

Solar events gamma-rays

- Electron Bremsstrahlung – induced gammas in solar flares
- Compton back-scattering

  → observable gamma-ray spectrum





Effects of Compton scattering on the Gamma Ray Spectra of Solar flares

Jun'ichi Kotoku

National Astronomical Observatory of Japan, 2-21-1 Osawa, Mitaka, Tokyo 181-8588, JAPAN

junichi.kotoku@nao.ac.jp

Kazuo Makishima[1] and Yukari Matsumoto[2]

Department of Physics, University of Tokyo, Bunkyo-ku, Tokyo, 113-0022

and

Mitsuhiro Kohama, Yukikatsu Terada and Toru Tamagawa

RIKEN (Institute of Physical and Chemical research), Wako-shi, Saitama

[1] *Also at RIKEN*

[2] *Present address: Mitsubishi Electric Co., Ltd.*

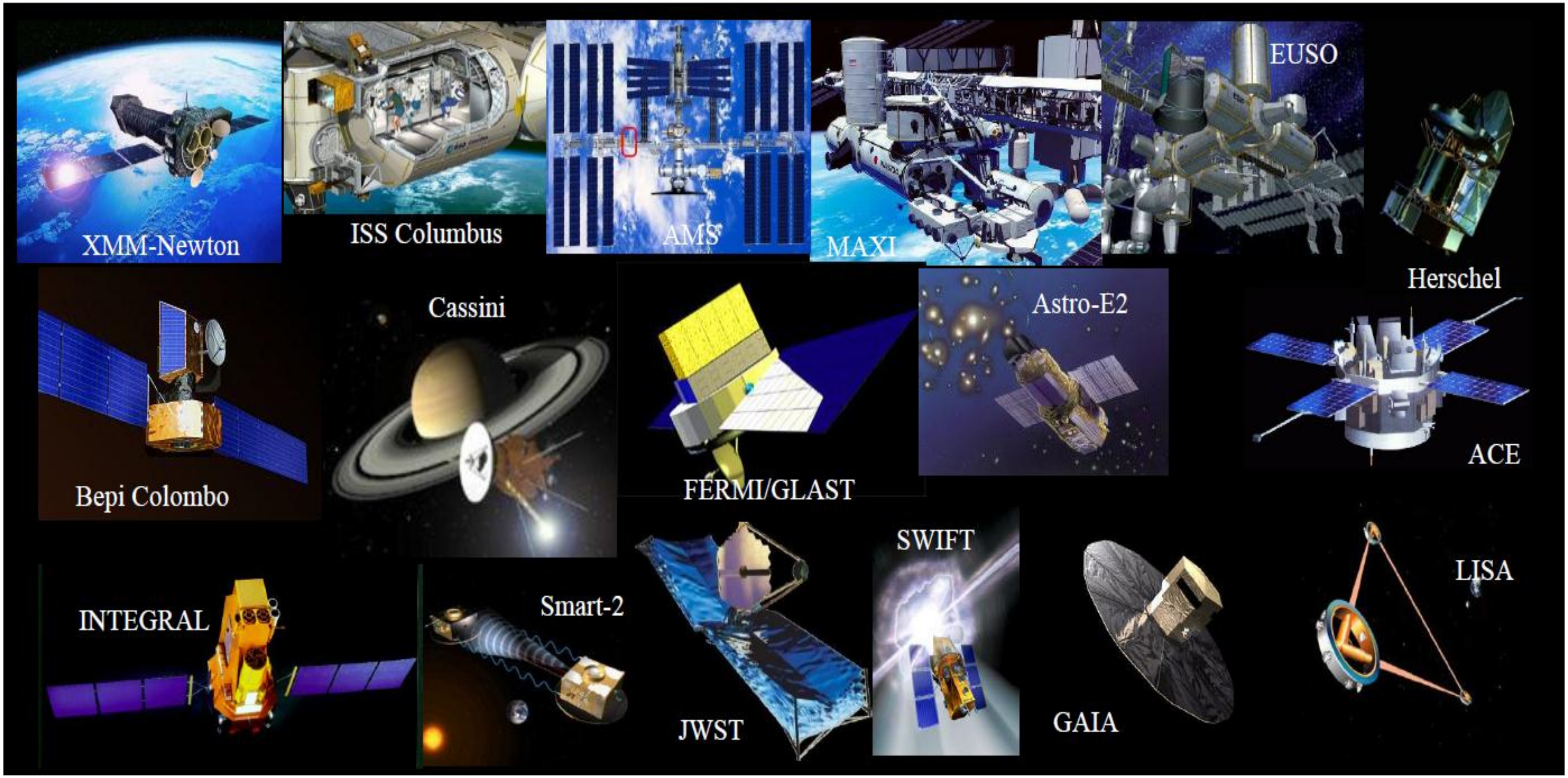
(Received ; accepted)

Abstract

Using fully relativistic GEANT4 simulation tool kit, the transport of energetic electrons generated in solar flares was Monte-Carlo simulated, and resultant bremsstrahlung gamma-ray spectra were calculated. The solar atmosphere was ap-

Geant4 Application Domains

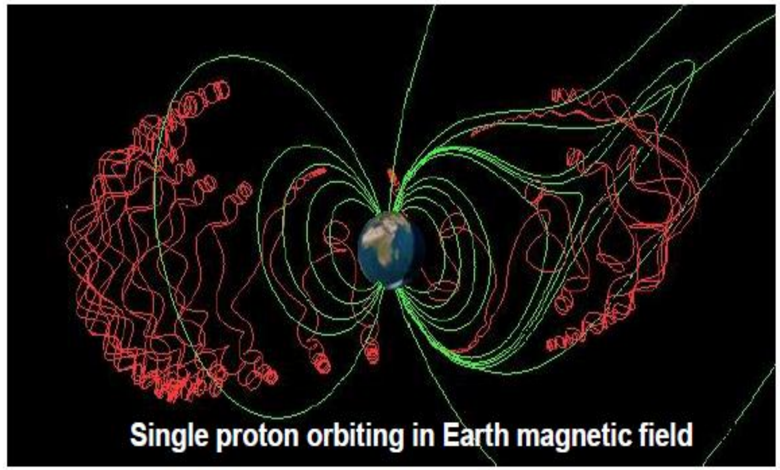
Geant4 in Space

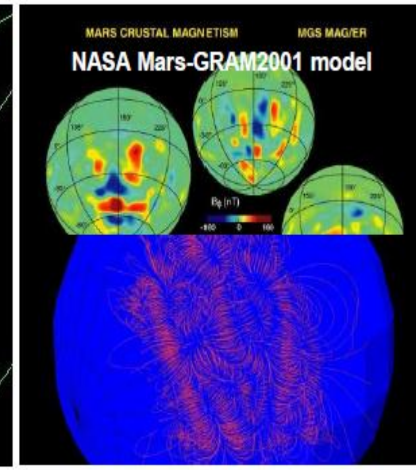


Geant4 Application Domains

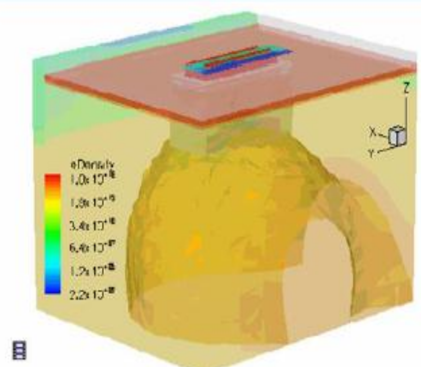
Planetary scale simulation, dosimetry

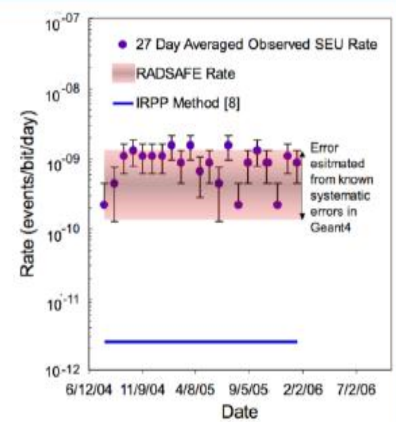
- **Planetocosmic:**
  - Geant4 simulation of Cosmic Rays in planetary Atmo-/Magneto- spheres
  - Laurent Desorgher *et al.* (Now at ICHUV, Switzerland)





- **Single event effect rate:**
  - RADSAFE / MRED project
  - Robert A. Weller *et al.* (Vanderbilt University, Nashville, TN, USA)

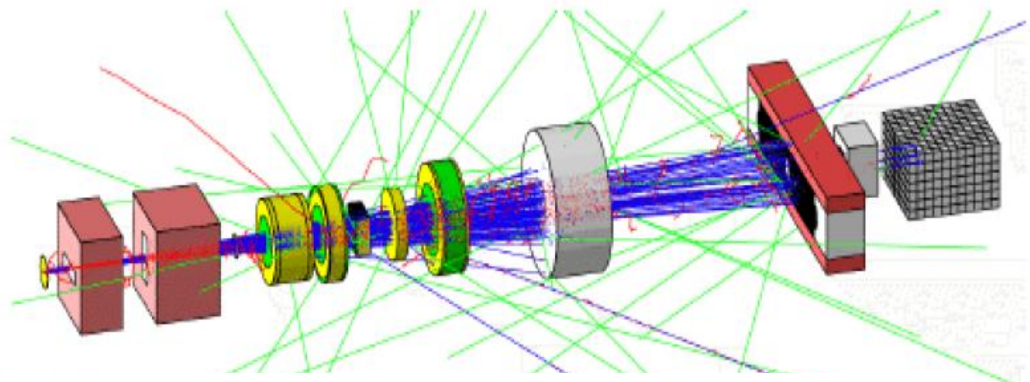


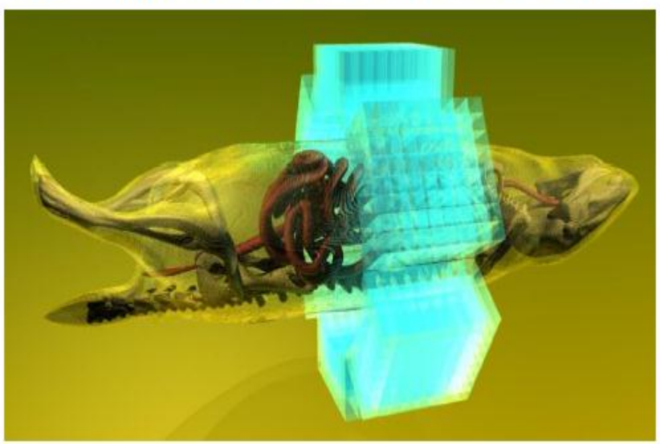


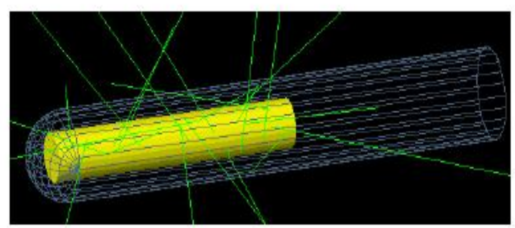
Geant4 Application Domains

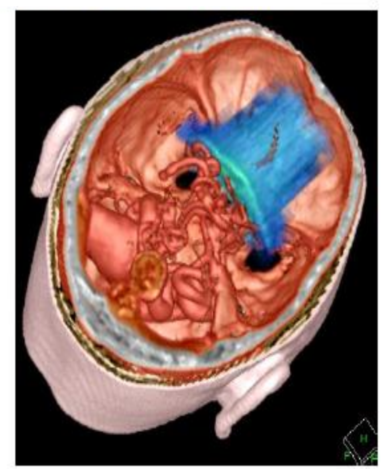
Geant4 in Medical Science

- Main use cases:
  - Beam therapy
  - Brachytherapy
  - Imaging
  - Irradiation study





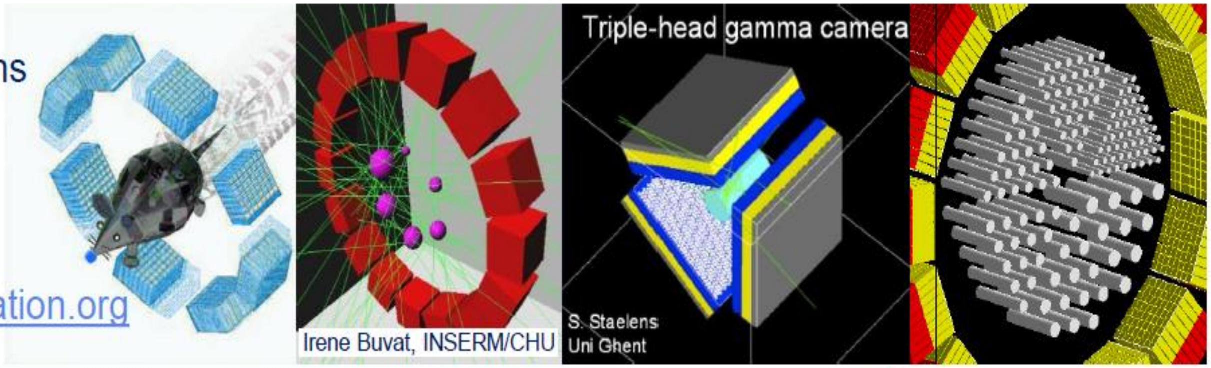




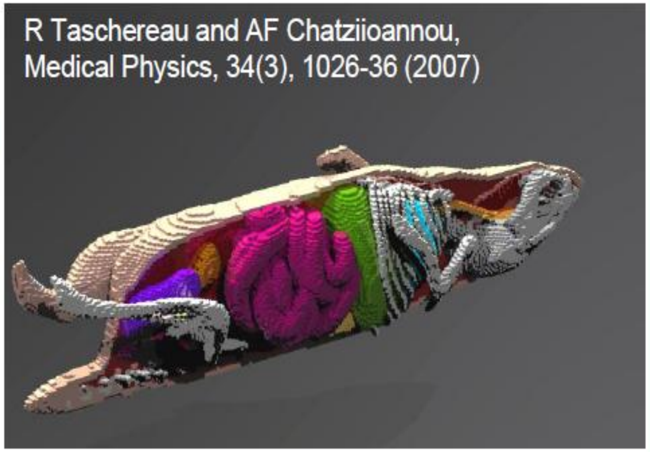
Geant4 Application Domains

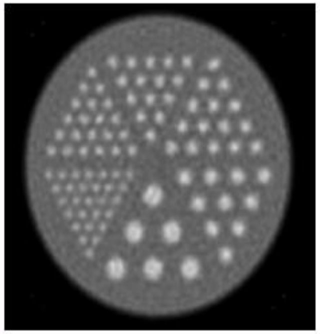
Imaging

- GATE
  - Toolkit for Imaging applications
  - based on the Geant4 toolkit
  - easier to use for Imaging applications
  - http://www.opengatecollaboration.org



- Ex of High resolution phantoms
  - (400 µm)^3 voxelized mouse phantom
  - Simulated map of 18-fluorine absorbed dose

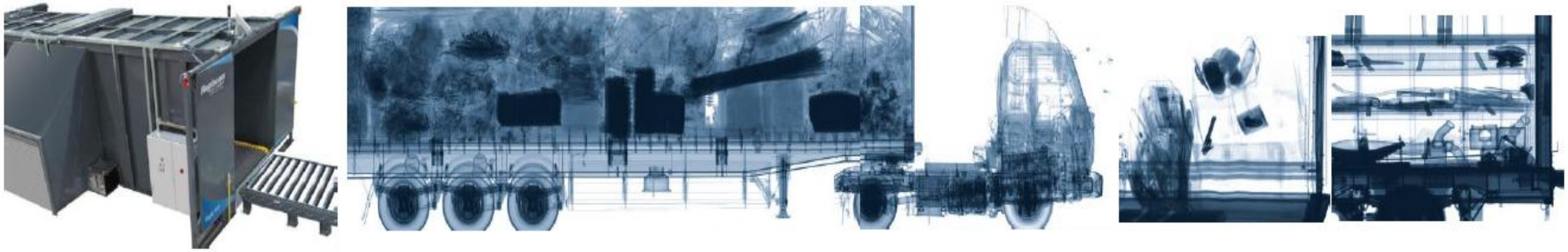




One reconstruction example, extracted from https://doi.org/10.1186/s40658-020-00309-8

Geant4 Application Domains

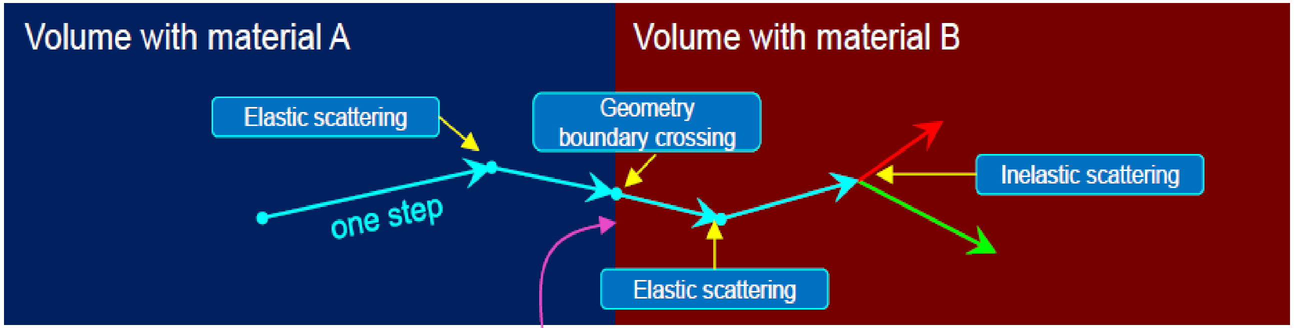
Geant4 in Homeland Security : simulating X-ray cargo radiography



Physics modeling

Monte-Carlo Particle Transport

- In a Monte-Carlo transport code, particles are moved by **steps**:
  - Ie : particles are not moved "continuously" but with "finite displacements", the steps
  - During which **physics calculations** are made and applied to the particle to **modify its state**



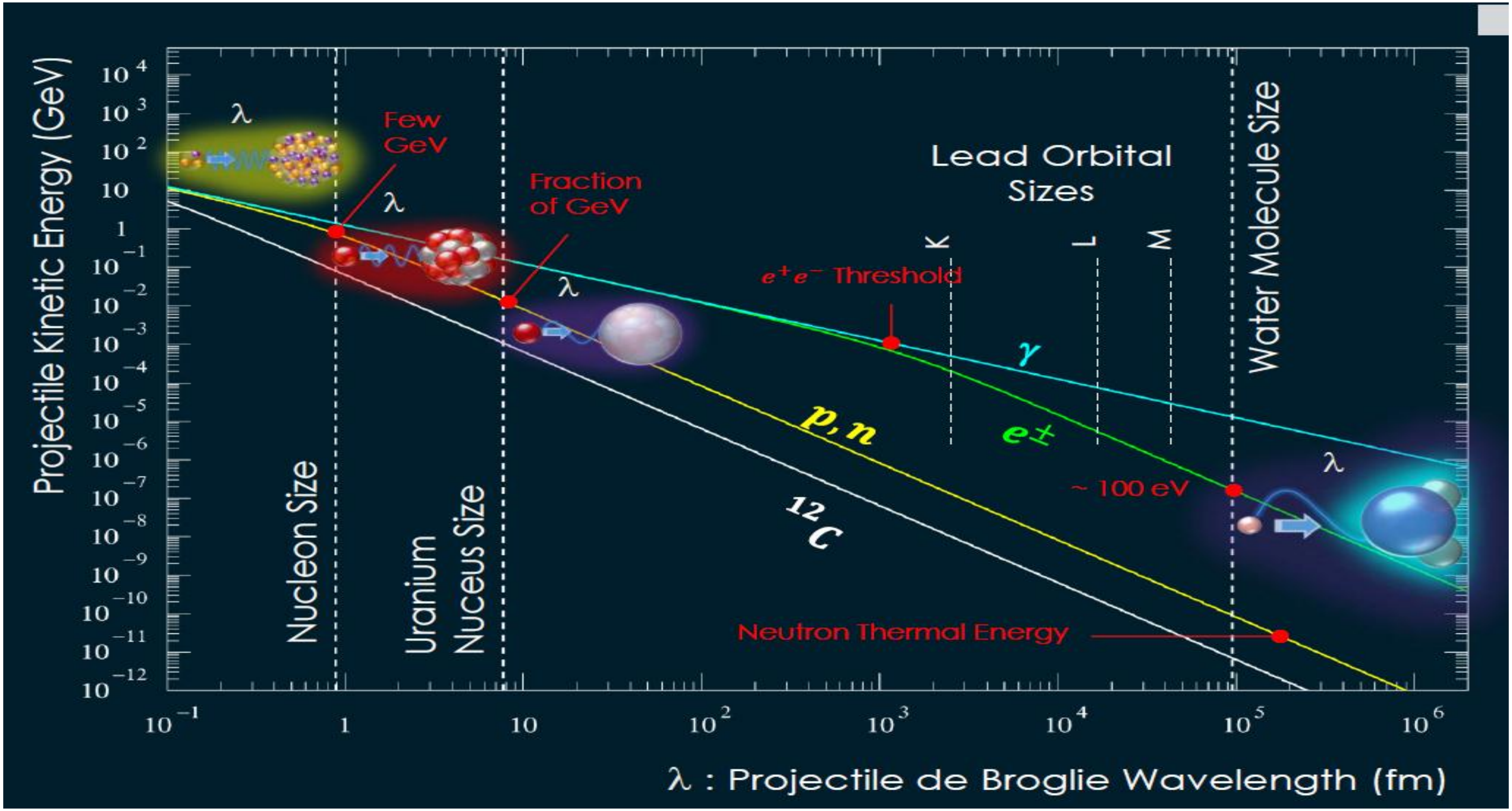
- **How a step proceeds ?**
  1. The **step length** is determined at the beginning of the step
     - **Step length** = minimum of {**physics limit**, geometry limit, (user limit also possible)}
     - The **physics limit**, is the minimum of the {interaction length ℓ_i}, $i = 1, \ldots, n_{processes}$
       - Each ℓ_i is sampled from $p_i(\ell_i) = \sigma_i \cdot exp(-\ell_i \cdot \sigma_i)$ with σ_i = cross-section for process i.
  2. The particle is **moved to the end** of the step
     - If the particle is charged, **continuous looses are applied** along the way of the step
  3. If the step is limited by the physics, **the process with minimum ℓ_i is applied**

For example for a γ, processes are:
- Photo-electric
- Compton
- Conversion
- Gamma-nuclear

- **If the particle is still alive** (case for geometry limit or (quasi-)elastic process), **we go back to "1.".**
- **Details** of physics calculation in a step **can greatly differ** if simulating, eg, **HEP or medical** problems !

Physics modeling

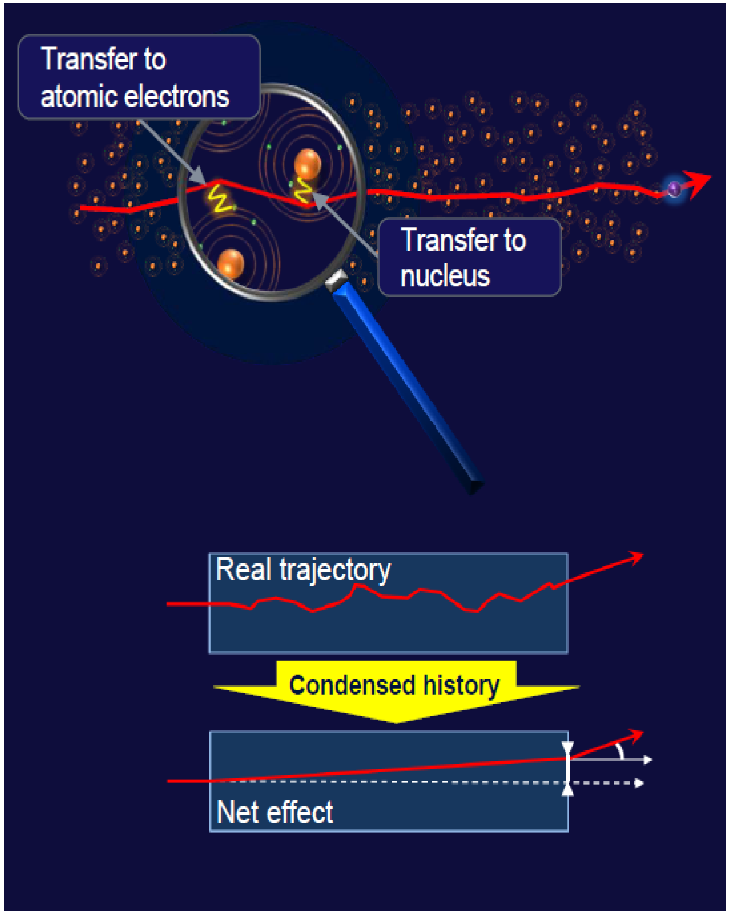
Monte-Carlo Particle Transport



Physics modeling

Charged Particles & Condensed History

- What means "If the particle is charged, **continuous looses are applied** along the way" ?
- In Nature, particles –neutral of charged- travel only through **discrete physical interactions**
- But for **charged particles** this involves **MANY interactions:**
  - Ionization generates $O(10^6)$ interactions/mm
  - Bremsstrahlung has infrared divergence
  - → Both generate **very numerous** but **very little interactions**
    - With tiny energy particles, **which almost don't travel**
  - → ***Lot of CPU time*** needed to simulated all these !
- **Condensed History** approach:
  - **Theoretically sum-up** the numerous tiny interactions
    - To generate their **net effect in one single calculation**
    - Tiny energy particles are accounted as **"local energy deposit"** –as they don't travel– and **are not created**
  - User defines a **threshold** : the **so-called "cut"**
    - Only ionization e^-'s –i.e. "δ-ray"'s–, or a bremsstrahlung γ's **above the cut are produced**, and then tracked
    - So "cut" = **limit between continuous and discrete energy losses**
- **The "cut" is a question each user has to care about !**

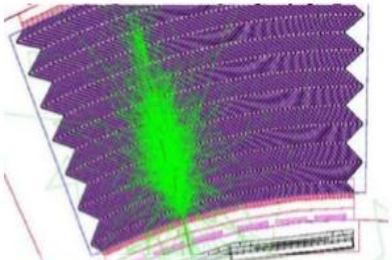


Physics modeling

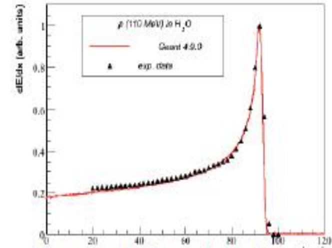
Electromagnetic Physics Overview

- **"Standard" Electromagnetic:**
  - Energy range 1 keV – O(100 TeV)
  - Processes for e-, e+, γ
  - Charged hadrons ionization up to 100 TeV
- **Muon, up to PeV**
- **"Low energy" Electromagnetic:**
  - More precise description:
    - PENELOPE 2008 reimplementation
    - LIVERMORE data for cross-sections and final states
    - Energy range down to ~250 eV / ~100 eV
  - Charged hadron ionization
    - ICRU' 49 & 73 & 90, NIST
  - Material relaxation (PIXE, Auger e-, …)
- **DNA & MuElec:**
  - For microdosimetry studies in DNA and Silicon
  - Processes down to a few eV
  - Chemistry stage for DNA
    - Water radical scattering
- **Optical photon: long wavelength** γ (X-ray, UV, visible)
  - Reflection, refraction, absorption, wavelength shifts, Rayleigh
- **Phonons:**
  - Suited for very low-temperature detectors (tens of mK)

Of most interest for medical applications

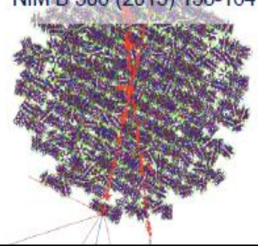


EM shower in ATLAS calorimeter

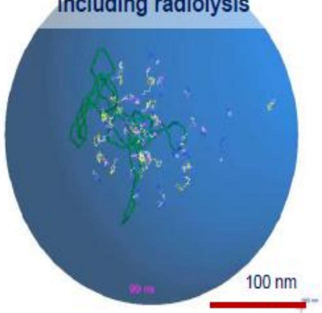


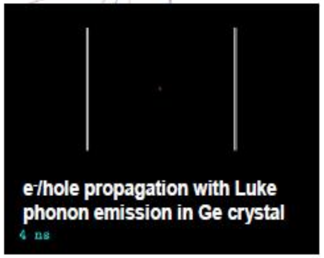
Bragg peak for p in water

Cell nucleus (15 µm diameter) with 6×10^9 base pairs of DNA
NIM B 306 (2013) 158-164

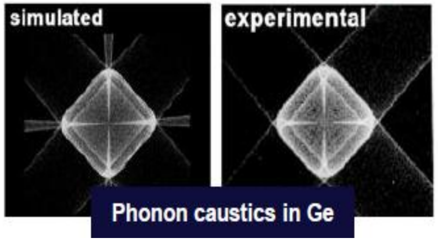


pBR322 plasmid irradiation, including radiolysis



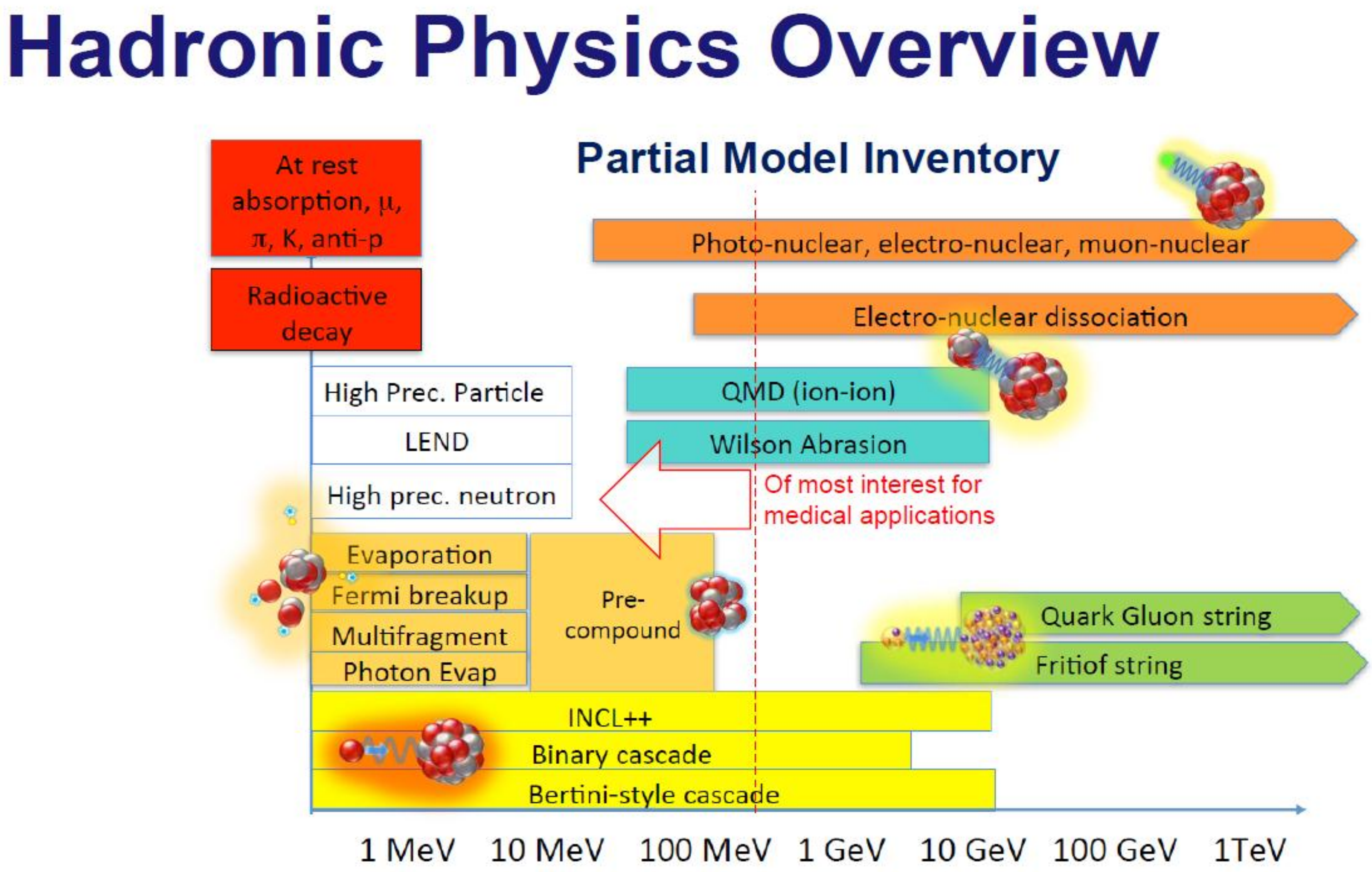


e-/hole propagation with Luke phonon emission in Ge crystal



Phonon caustics in Ge

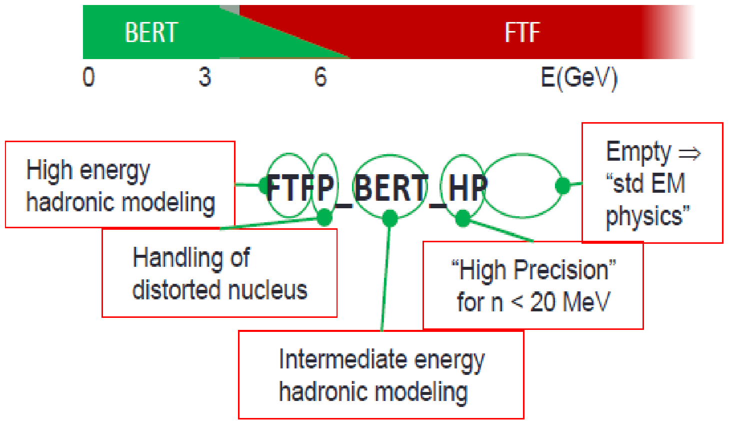
Physics modeling



Physics modeling

The Physics List Concept

- **There are <u>many</u> physics models in Geant4 !**
  - electromagnetic & hadronic, but also radioactive decay, options for low energy neutrons, low energy electromagnetic, etc.) available in Geant4
  - plus some options like fast simulation, variance reduction (not discussed today)
- Some physics models are:
  - **complementary** (valid on ≠ energy domains)
  - **competitive** (valid on the same energy domain)
- A "physics process" –eg "hadron inelastic" – is **often composed of several models**
  - Each model serving one energy domain
- **The "physics list" is an object that gathers consistent set of "physics processes" = physics configuration of the app.**
  - And that configures their underneath "models", parameters, etc
- Geant4 provides **"ready to go"** physics lists, meant to **respond to different use-cases**, eg:
  - **High Energy Physics**
    - With for example LPM effect activated, by no details on atomic structure
  - **Medical**
    - With accurate description of Bragg peak
  - **DNA**
    - With ultra-low energy processes activated, but no precision on high energy side
- **These are continuously monitored**
- They can served as a **basis for more specialized physics**



Example of FTFP_BERT_HP physics list

Physics list

- Since different (hadronic) models exists with different performances (quality of results and computing requirements) at different energy ranges, multiple choices are available:
  - Models are assembled in “physics lists”
- Can be built from scratch or use one of the provided “educated” physics lists, for applications in:
  - HEP calorimetry, tracking, low-E dosimeter with neutrons, shielding, medical applications, air shower applications, low background experiments, space applications

Physics list

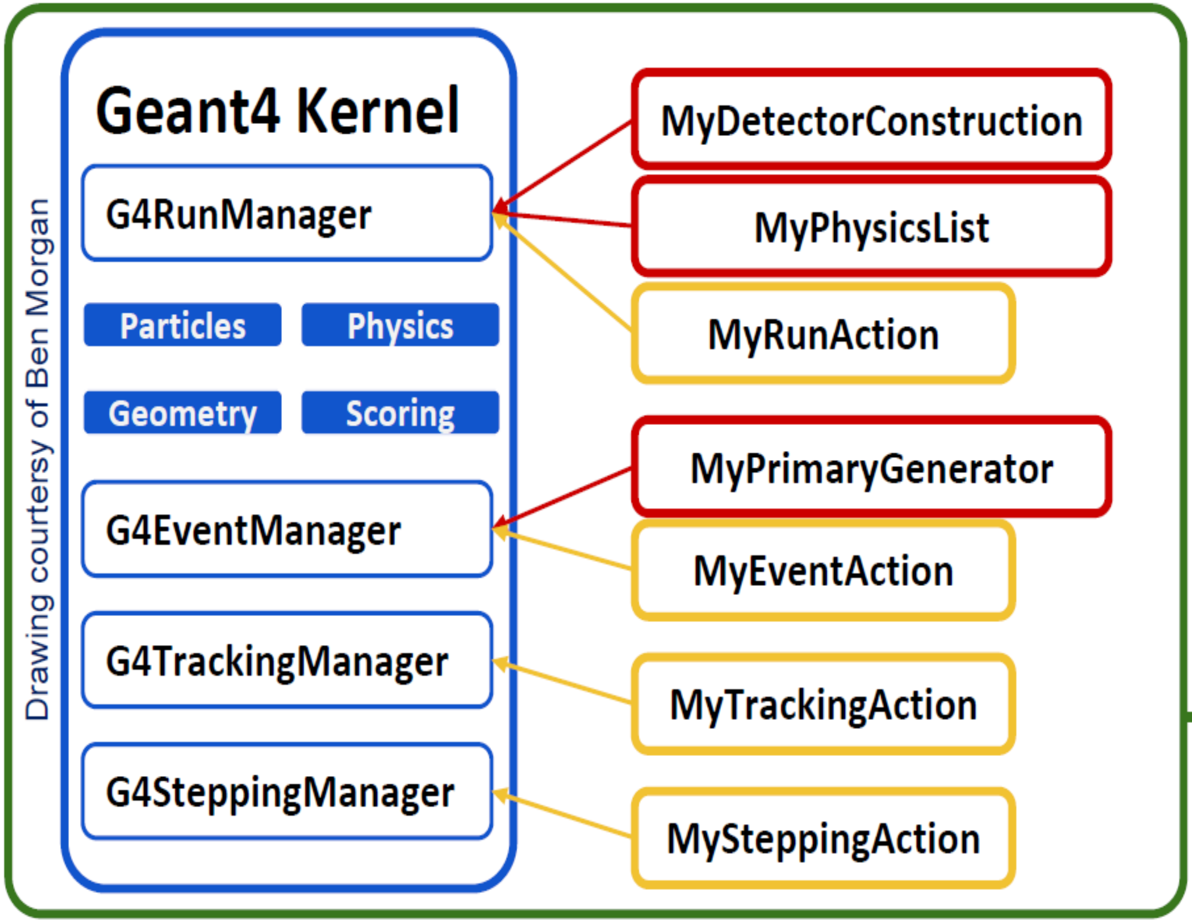
Test-beam summary (G4 9.4.p01)

Status Sept-Oct 2011

	Response	Resolution	Smoothness	Lateral Shape	Longitudinal Shape @10λ	Peculiarities, comments
QGSP_BERT	+(1-3)%	-(5-10)%	Δ~5%@10GeV	π,p: -(10-20)%	π: -10% p: -20%	Extensive use of LHEP
FTFP_BERT QGSP_FTFP_BERT	+(0-5)% (***)	-(3-7)%	Δ~0	π: -(10-20)% p: -(3-10)%	π: +10% p: +(10-20)%	anti-nucleons, hyperons via CHIPS(*), no LHEP
CHIPS	+(5-10)%	-(10-20)%	Δ~0	π: -(3-10)% p: -(10-20)%	π: -10% p: -20%	anti-nucleons, hyperons, single model
FTF_BIC(**)	+(3-5)%	-(2-6)%	Several irregularities	-	π: +10%	Implements re-scattering at high E, Extensive use of LHEP

Software aspects

Geant4 Main Components



Mandatory *Description*
- System Geometry
  - Scoring
- Particles and Processes
- Primary Particle Source(s)

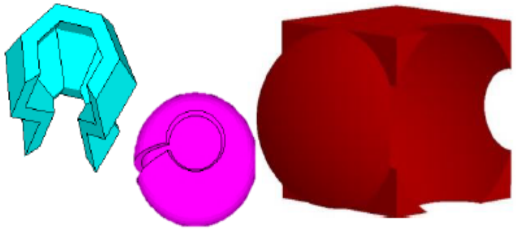
Optional *Actions*
- Custom control of steps, tracks, events, runs

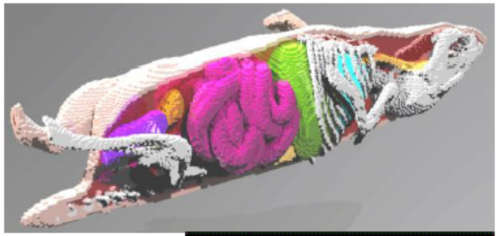
MyG4App.exe
main(){...}

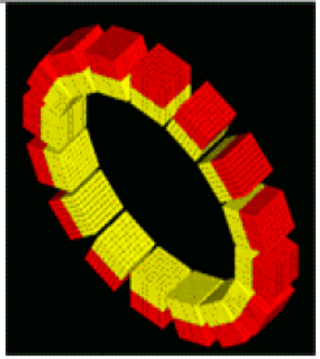
Software aspects

Key geometry capabilities

- Richest collection of shapes
  - CSG (Constructed Solid Geometry), Boolean operation, Tessellated solid, etc.
  - The user can extend
- Describing a setup as hierarchy or 'flat' structure
  - Describing setups up to billions of volumes
  - Tools for creating & checking complex structures
  - Interface to CAD
- Navigating fast in complex geometry model
  - Automatic optimization
  - By subdivision of geometry in "voxels" containing a few volumes, with fast navigation between neighbor voxels
- Geometry models can be 'dynamic'
  - Changing the setup at run-time
    - e.g. "moving objects"







Citations

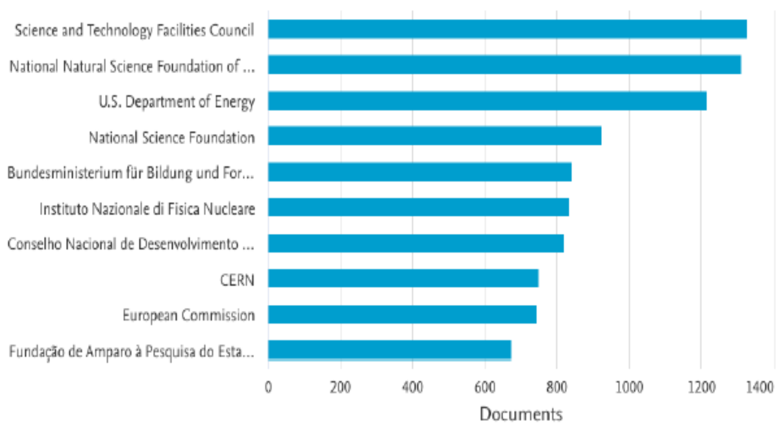
Scopus **As of 02/2020** Search Sources Lists SciVal Create account

12,644 documents have cited:

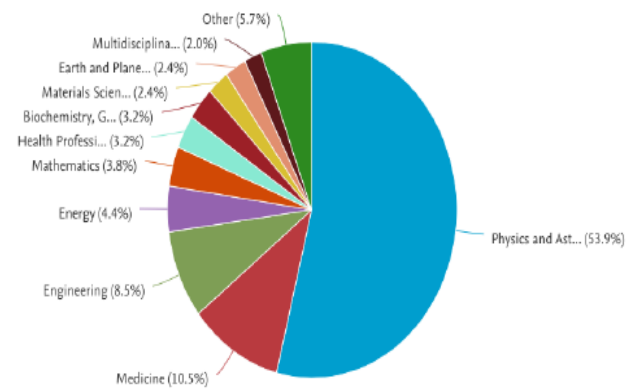
GEANT4 - A simulation toolkit
Agostinelli S., Allison J., Amako K., Apostolakis J., Araujo H., Arce P., Asai M., (...), Zschiesche D.
(2003) Nuclear Instruments and Methods in Physics Research, Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 506 (3) , pp. 250-303.

Set feed

Documents by funding sponsor
Compare the document counts for up to 15 funding sponsors.



Documents by subject area



Geant4 - A simulation toolkit
[NIM A, vol 506(3), pp250-303, 2003]

Significant use across **many** research areas, **considered mission critical for HEP**

Physics processs and their clasification

- Particles undergo various interactions as they move through matters
- Each interaction is called *Process* and described by a process class

  For example:

  - G4ComptonScattering ← photon Compton scattering
  - G4eBremsstrahlung ← e^+/e^- bremsstrahlung
  - ....

- Processes are classified as:
  - Electromagnetic
  - Hadronic
  - Decay
  - Parameterized
  - Transportation
  - Optical

Geant4 EM sub-libraries

- Standard
  - γ, e^+, e up to 100 TeV
  - hadrons up to 100 TeV
  - ions up to 100 TeV
- Muons
  - up to 1 PeV
  - energy loss propagator
- X-rays
  - X-ray and optical photon production processes
- High-energy
  - process at high energy (E>10GeV)
  - physics for exotic particles
- Polarization
  - simulation of polarized beams
- Optical
  - optical photon interactions
- Low-energy
  - Livermore library γ,e- from 10 eV up to 1 GeV
  - Livermore library based polarized processes
  - PENELOPE 2008 code rewrite γ,e-,e+ from 250 eV up to 6 GeV
  - hadrons and ions up to 1 GeV
  - atomic de-excitation (fluorescence + Auger)
- DNA
  - Geant4 DNA models and processes
  - Micro-dosimetry models for radiobiology from 0.025 eV to 10 MeV
  - many of them material specific (water)
  - Chemistry in liquid water

Geant4 Physics: Hadronic

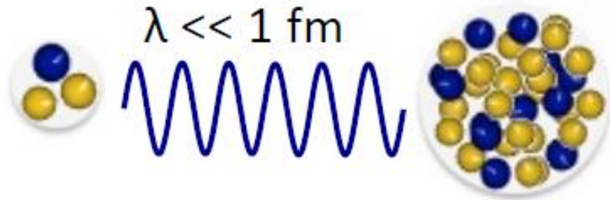
- Pure hadronic interactions for 0 to 100 TeV
  - elastic, inelastic, capture, fission
- Radioactive decay:
  - both at-rest and in-flight
- Photo-nuclear interaction from ~1 MeV to 100 TeV
- Lepto-nuclear interaction from ~100 MeV up to 100 TeV
  - e- and e+ induced nuclear reactions
  - muon induced nuclear reactions

Hadronic interation from TeV to MeV

- High energy
  - Quark-gluon interaction
  - String model
    - Quark-Gluon string model
    - Fritiof

TeV hadron

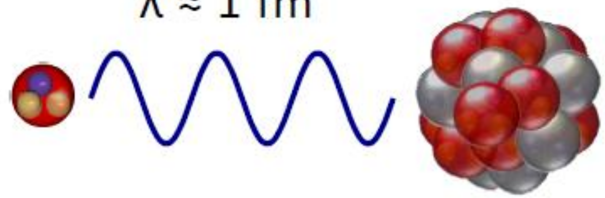
$\lambda << 1$ fm



- Intermediate energy
  - Hadron-nucleon interaction
  - Intra-nuclear cascade model
    - Binary cascade model
    - Bertini model
    - INCL++

~GeV - ~100 MeV

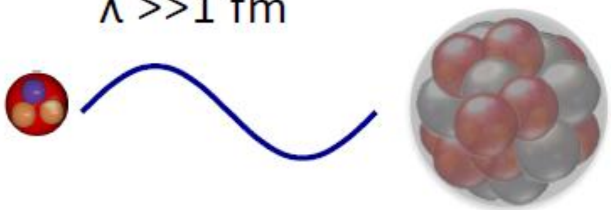
$\lambda \approx 1$ fm



- Low energy
  - Nucleon-nucleus interaction
  - Pre-compound model
    - Fission/Evaporation
    - Gamma Evaporation
    - Radioactive Decay
    - Capture at rest

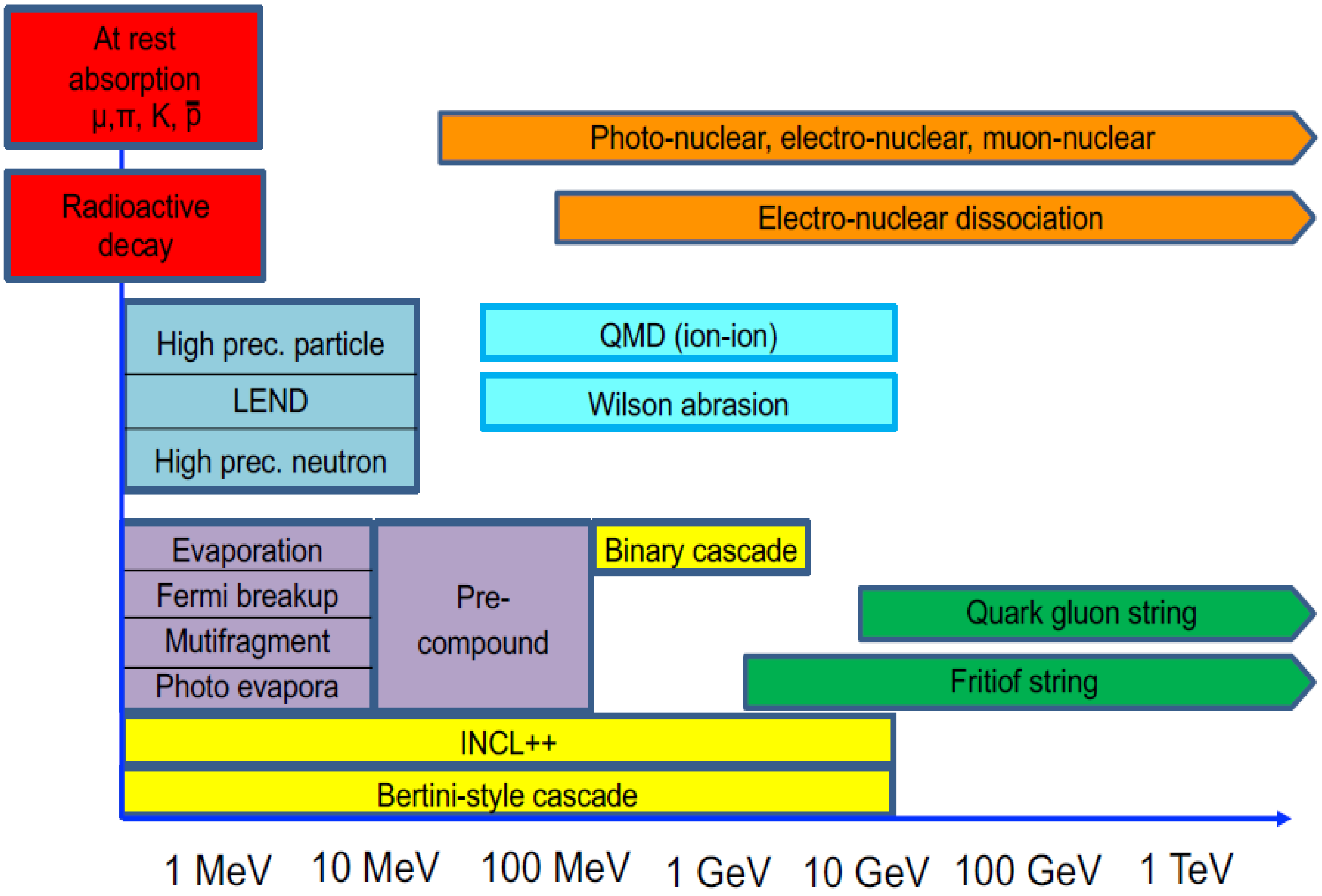
~100 MeV - ~10 MeV

$\lambda >> 1$ fm



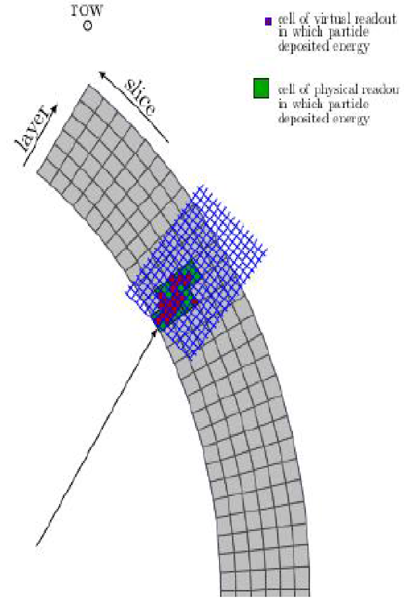
~10 MeV to thermal

Hadronic interation from TeV to MeV



Geant4 example Par04

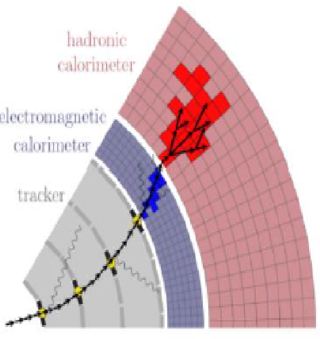
- **A new example demonstrates how to build a shower model using ML:**
  - Extended example Par04 shows how to use Machine Learning (ML) models within GEANT4.
  - Distributed with a Variational AutoEncoder (VAE) model of showers used in fast simulation.
  - Demonstrates how to incorporate inference libraries: ONNX runtime, pyTorch, lwtnn.
  - Ability to run it on GPU (a choice done by UI command).
  - It scores energy along shower axis, performs validation of shower observables.
  - Recent additions of physical detector readout for performance benchmarking.

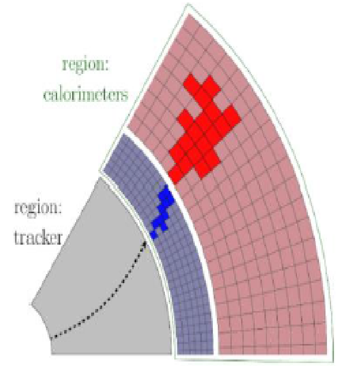


- Note : example **Par04 is not limited to showers**
  - For example, will be used to speed-up tracking in channeling process

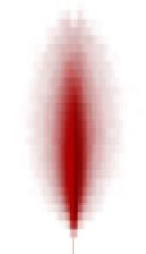
Fast simulation and ML in Geant4

- **What is "fast simulation" ? (FS)**
  - In HEP "fast simulation" classically refers to "fast shower simulation"
  - And is even more restricted in practice to "fast electromagnetic shower simulation" (FS of hadronic showers are rare [1]).
- FS replaces the detailed tracking with an approximate model of energy deposition in the calorimeter(s)
  - Model is in general detector dependent [2]





- **On-going work in Geant4:**
  - Investigate usage of Machine-Learning (ML) techniques in replacement of the classical analytical & detector-specific models
    - It may happen that ML techniques resolve [1] & [2] !
  - Focus on electromagnetic showers for now
  - **Exploit Variational AutoEncoder (VAE) model**
    - Used by ChatGPT, DALL-E, …
    - A neural-network with probabilistic connections → naturally varies the generation, request after request
- **Aggregates quite intense activities with LHC experiments !**



Longitudinal & lateral VAE generated shower profiles