Neutrino Generators & MC Methods

Costas Andreopoulos[1,2]

[1]University of Liverpool, [2]STFC Rutherford Appleton Laboratory

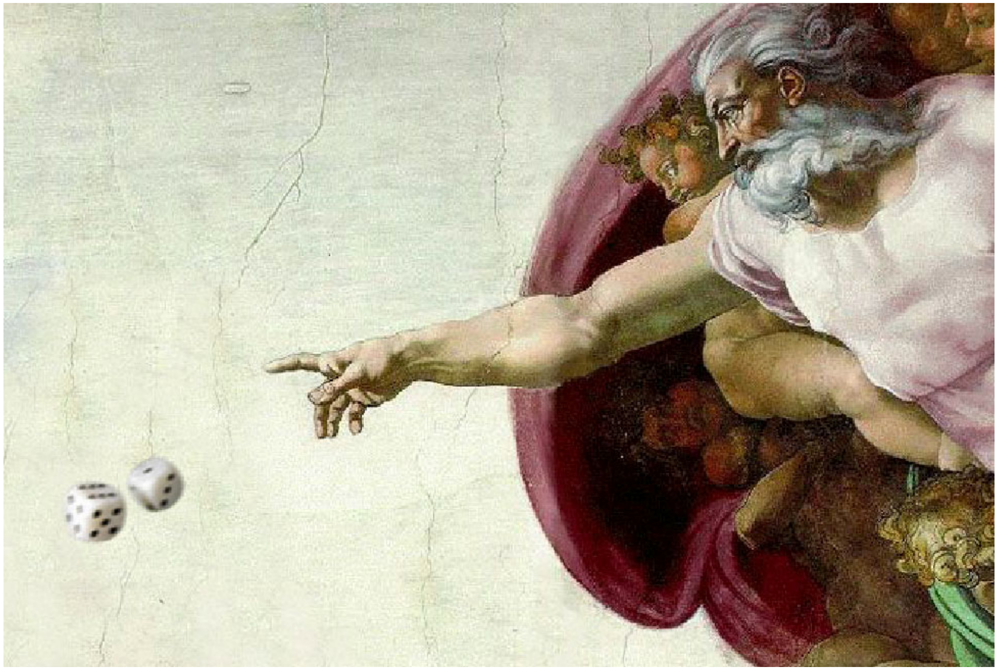
presented at the NuSTEC Neutrino Generator School, 14-16 May 2014, Liverpool

May 14, 2014

Outline

Neutrino MC Generators

- Why do we need them?
- What do they provide?
- How do they work?



Because Einstein was wrong. God does play dice!

Motivation

- Discovery of neutrino masses and mixings: New physics beyond SM!
- New physics not understood
  - What is the mass generation mechanism?
    - Why are the masses so small?
    - Could the neutrino be a Majorana particle?
  - Does it explain flavour?
    - Nearly (exactly?) maximal mixing observed in 'atmospheric' neutrino oscillations: 'μ' and 'τ' flavour interchangeable.
  - Does it provide a connection between the quark and lepton sectors?
    - Why the corresponding mixing matrices are so different?
  - What are the implications for the universe we live in?
    - Baryon asymmetry of the universe: CP violation + Majorana masses ingredients of the leptogenesis hypothesis.
    - Dark matter: Sterile neutrino is a candidate.

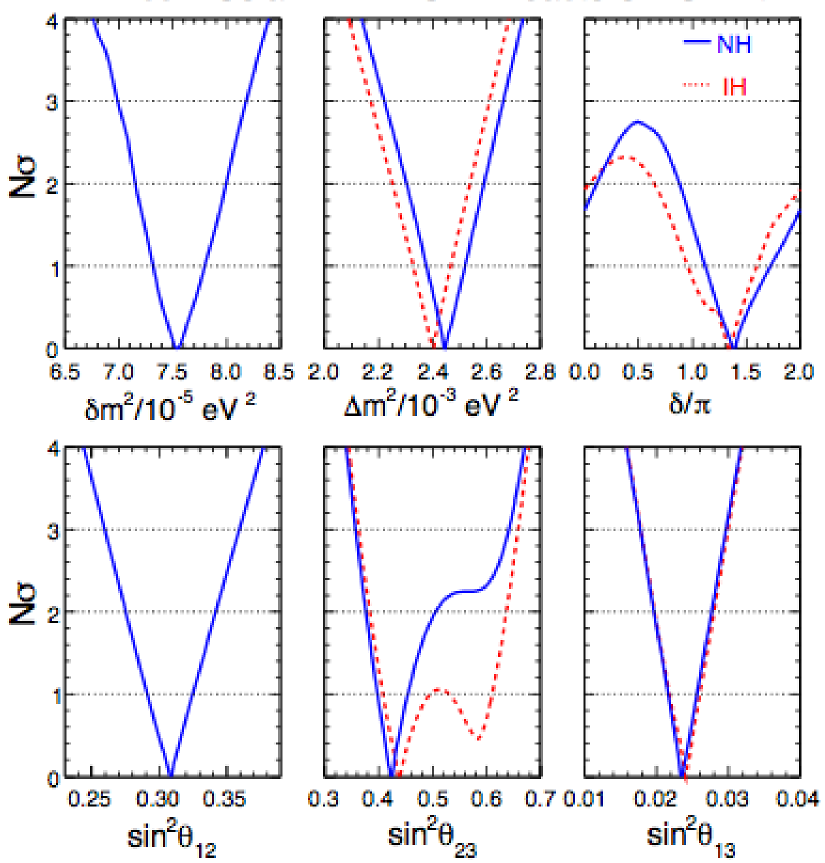
The study of ν masses and mixings the only known window to new physics.

Looking ahead

The progress over the last two decades in improving the knowledge of neutrino masses and mixings has been tremendous!

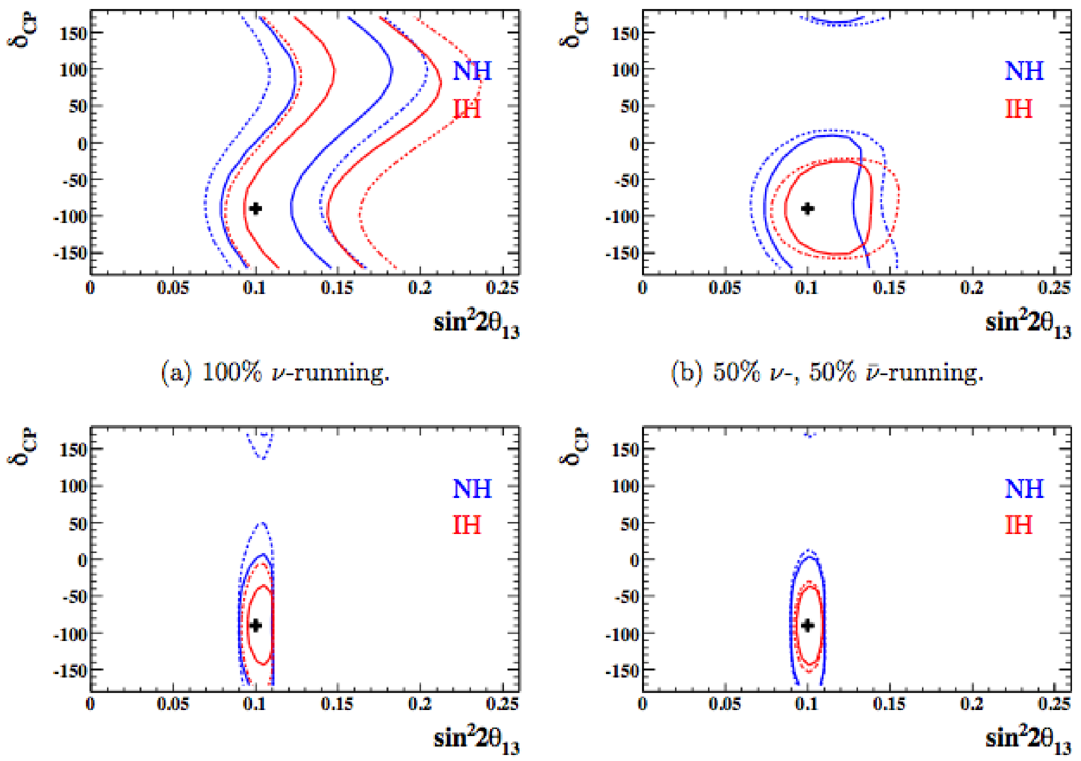
But this is not the end, just the end of the beginning!

- Could start addressing the questions of θ_{23}-octant and mass-hierarchy ambiguities by $\sim$2020, if lucky.
- Could get to 2-3σ level evidence of neurtino CP-invariance violation by $\sim$2020, if lucky.
- Should firmly establish all above by $\sim$ 2040, unless very unlucky.
- Searches for sterile neutrinos, at various mass scales, should take us to the end of this century and beyond.



F.Capozzi, G.L.Fogli, E.Lisi, A.Marrone, D.Montanino, and A.Palazzo, arXiv:1312.2878

Near future ($\sim$ 2020) sensitivities - Example from T2K



(a) 100% ν-running.

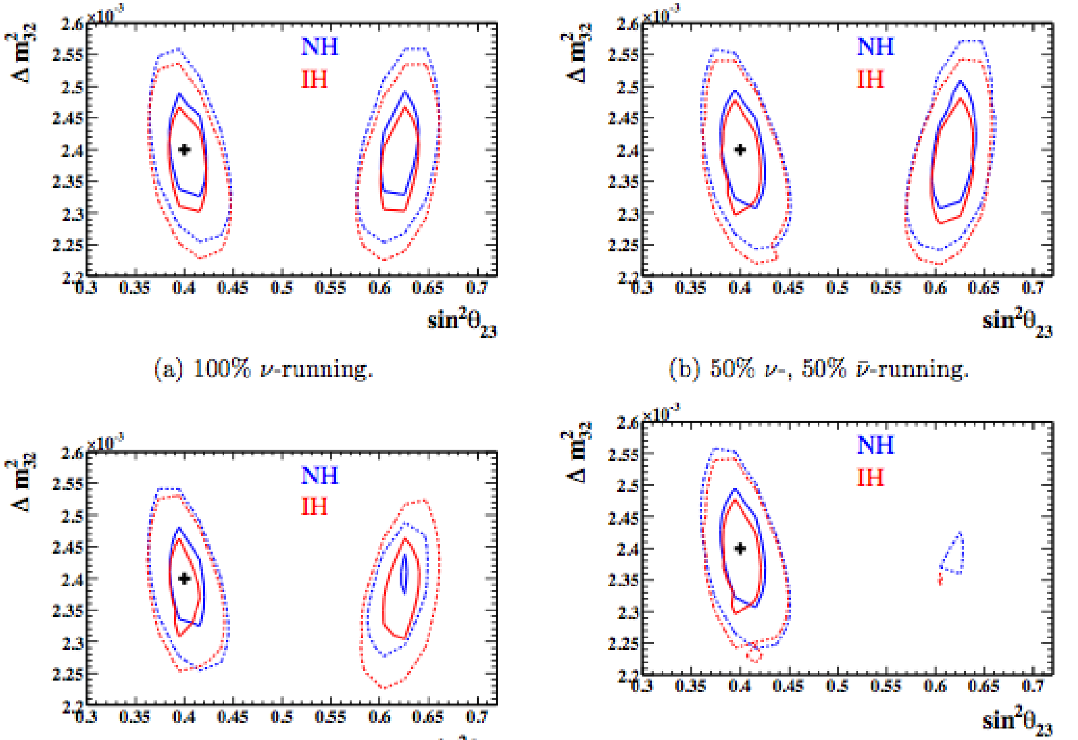
(b) 50% ν-, 50% $\bar{\nu}$-running.

(c) 100% ν-running, with ultimate reactor constraint.

(d) 50% ν-, 50% $\bar{\nu}$-running, with ultimate reactor constraint.

- Difference in δ_{CP} sensitivity with ν-enhanced and $\bar{\nu}$-enhanced beam running.
- Improved sensitivity with a combination of ν and $\bar{\nu}$ data.
- **$\sim$90% C.L. measurement for certain true values of δ_{CP}.**
- Similar δ_{CP} constraint with and without the reactor data: **Could start over-constraining the PMNS framework.**

90% C.L. intervals for true NH and true $\delta_{CP} = -\pi/2$, $sin^2 2\theta_{13}$=0.1, $sin^2\theta_{23}$=0.5, Δm^2_{32}=2.4$\times 10^{-3}$ eV^2/c^4.
Blue: Correct hierachy, Red: Incorrect hierarchy - Solid: Statistical errors only, Dashed: With 2012 systematics.
Assumed exposure: 7.8 $\times$ 10^{21} protons on target. Assumed ultimate reactor constraint: $\delta(sin^2 2\theta_{13})$=0.005.
Fully correlated ν and $\bar{\nu}$ systematic errors.



(a) 100% ν-running.

(b) 50% ν-, 50% $\bar{\nu}$-running.

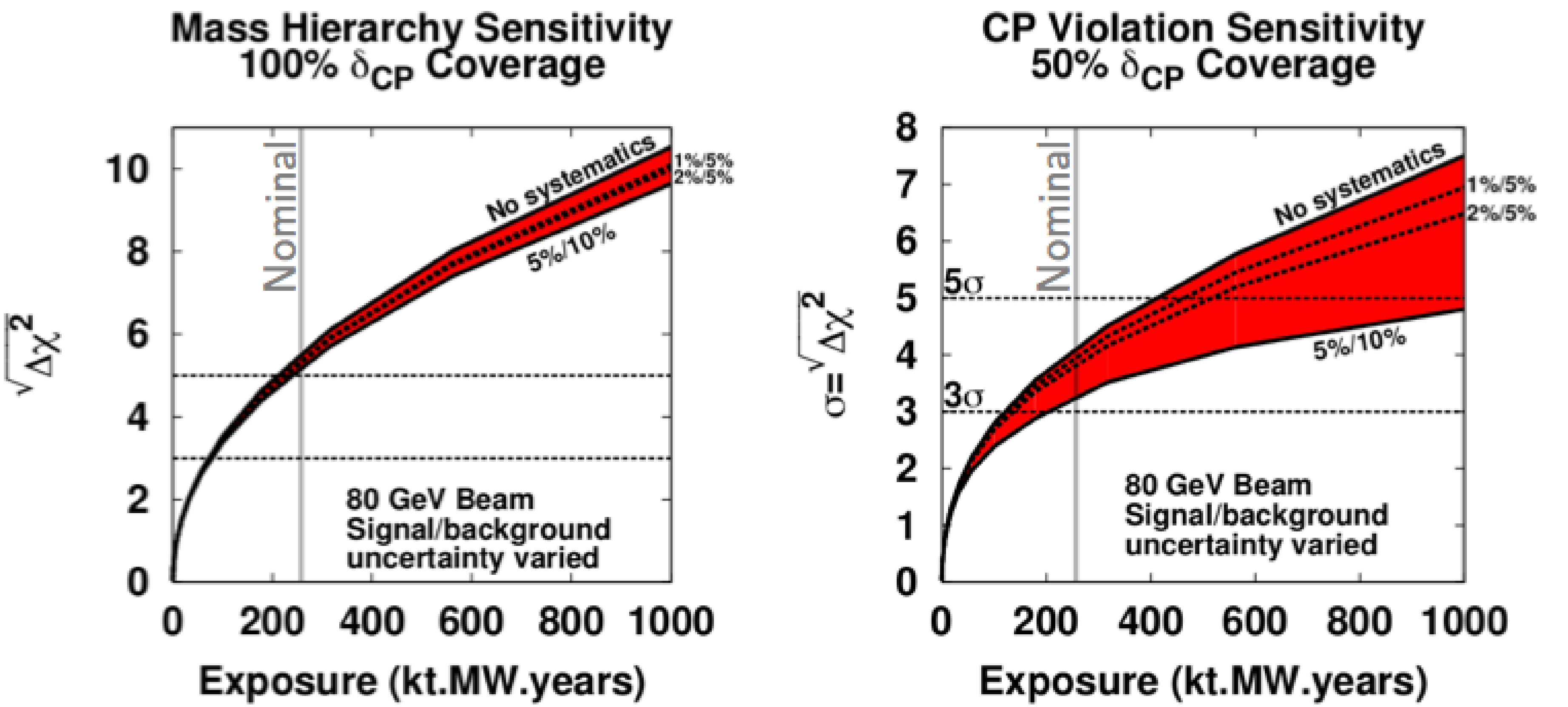
(c) 100% ν-running, with ultimate reactor error.

(d) 50% ν-, 50% $\bar{\nu}$-running, with ultimate reactor error.

- Added power from combining ν and $\bar{\nu}$ data compensates for loss of statistics in $\bar{\nu}$-enhanced beam mode. There is no effect on the disappearance measurement using T2K data alone.
- Combination of T2K ν and $\bar{\nu}$ data and reactor data could allow us to resolve the θ_{23} octant.

90% C.L. intervals for true NH and true Δm^2_{32}=2.4$\times 10^{-3}$ eV^2/c^4, $sin^2\theta_{23}$=0.4, $\delta_{CP} = 0$ and $sin^2 2\theta_{13}$=0.1.
Blue: Correct hierachy, Red: Incorrect hierarchy - Solid: Statistical errors only, Dashed: With 2012 systematics.
Assumed exposure: 7.8 $\times$ 10^{21} protons on target. Assumed ultimate reactor constraint: $\delta(sin^2 2\theta_{13})$=0.005.
Fully correlated ν and $\bar{\nu}$ systematic errors.

Looking further into the future - LBNE

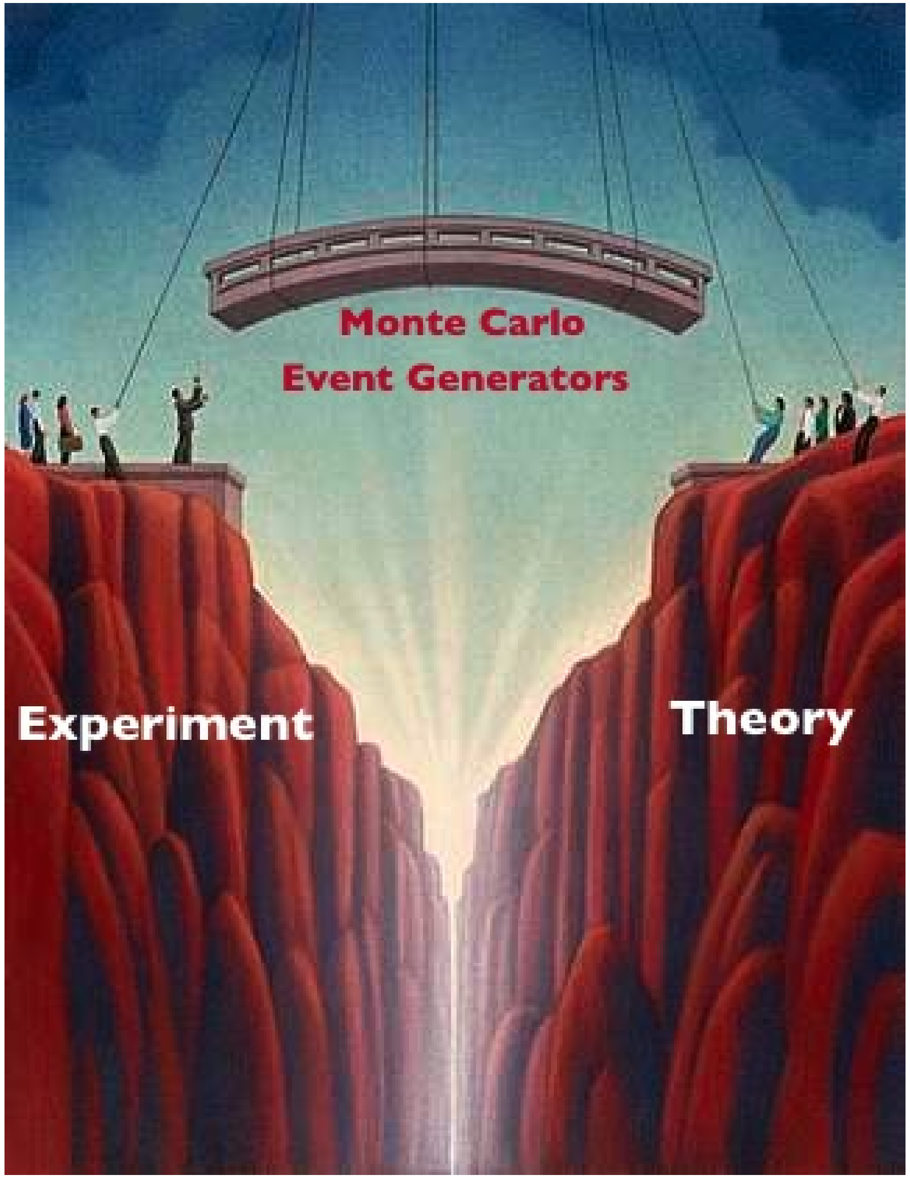


Future challenges

- Energy reconstruction?
  - modelling of nuclear dynamics
  - CCnonQE/QE
  - hadronization, rescattering $\rightarrow$ hadronic shower energy scale
- ν_e/ν_μ?
  - large θ_{13}: larger event rates but a smaller CP asymmetry
  - ν_e oscillation 'signal' systematics would be important
  - but the 'signal' interaction systematics would be poorly constrained by ND (small intrinsic ν_e contamination, with somewhat different flux systematics)
- $\nu/\bar{\nu}$?

The study of ν masses and mixings the only known window to new physics. Looking through this window requires accurate and reliable simulations, anchored both on the best available theory and all relevant scattering data both from neutrino and non-neutrino probes.

Neutrino MC Generators: A Theory/Experiment Interface



Neutrino MC Generators connect the true and observed event topologies and kinematics.

Every observable a convolution of flux, interaction physics and detector effects. Neutrino MC Generators allow experimentalists to access, improve, validate, assess the uncertainty of and tune the *physics* models that drive the result of that convolution.

Neutrino MC Generators: What do they provide?

- Front-ends for fast neutrino event generation / 4-vector level studies.
- Back-ends in experimental Monte Carlo simulation chains.
- Libraries of calculated neutrino cross-sections.
- Event re-weighting engines, used for evaluating systematic errors and propagating them in physics analyses.
- Data-bases of experimental neutrino, electron and hadron scattering data used for model tuning and systematic error evaluation.
- Toolkits for model tuning, data/MC comparisons, ...
- Electron-Nucleus and Hadron-Nucleus generators working in the same physics framework as the Neutrino-Nucleus generator and used to extract model constraints from complementary non-neutrino scattering data.
- Nucleon decay generators.
- ...

Neutrino MC Generators

MC Generators are used from the time an experiment gets proposed

- prediction of event rates and feasibility studies
- physics-driven requirements for the experiment design
- ...

till the final publication

- evaluation and propagation of systematic uncertainties
- evaluation of acceptances and backgrounds
- ...

Neutrino MC Generators commonly used

Generators developed primarily by experimental groups:

- **GENIE** (http://www.genie-mc.org) :
  Activelly developed by a large group of authors. Written in C++. Used primarily in the Fermilab experiments (LBNE, MINERvA, MicroBooNE, LAr1-ND) but also from T2K, LAGUNA-LBNO, IceCuBE and sevaral others.
- **NEUT**:
  Activelly developed by Yoshinari Hayato and collaborators. Written in Fortran-77. Used primarily within T2K and Super-Kamiokande.
- Nuance:
  Was developed by David Casper and collaborators. Written in Fortran-77. Unsupported? A modified version? still used by MiniBooNE.
- Neugen:
  Was developed by Hugh Gallagher and collaborators. Written in Fortran-77. Development discontinued several years ago. (GENIE v2.0.0 equivalent with last version of Neugen) Still used by MINOS.

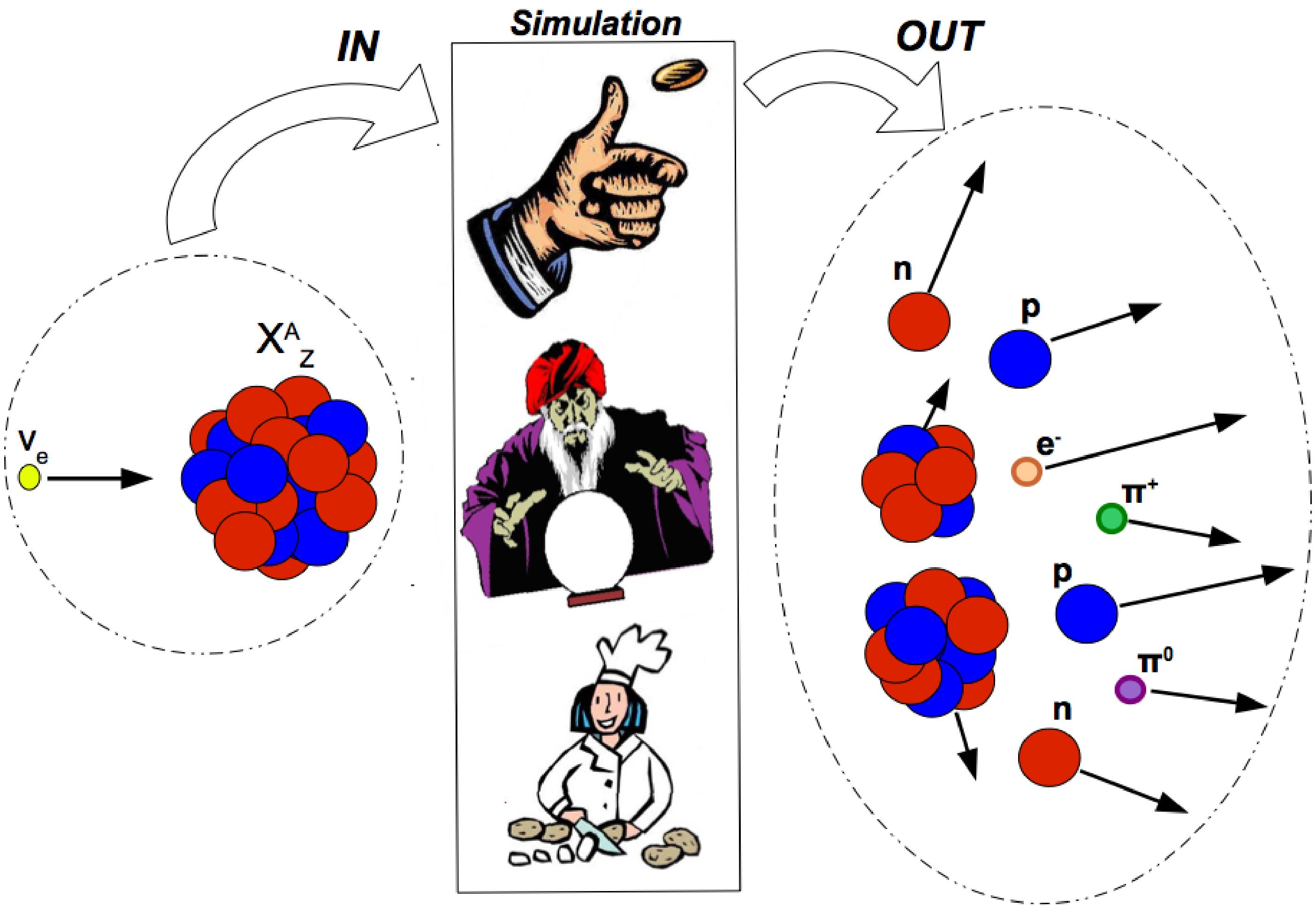
Generators developed primarily by theory groups:

- **NuWro** (http://borg.ift.uni.wroc.pl/nuwro) :
  Developed by Jan Sobczyk and collaborators at Wroclaw. Written in C++. Used by several experiments (T2K, MINERvA).
- **GiBUU:** (https://gibuu.hepforge.org/trac/wiki) :
  Developed by Ulrich Mosel and collaborators at Giessen. Written in Fortran-2003.

A survey of neutrino generators is presented by J.Sobczyk (Lecture T2).
The GENIE and NuWro generators will be used in Tutorials 1-7.

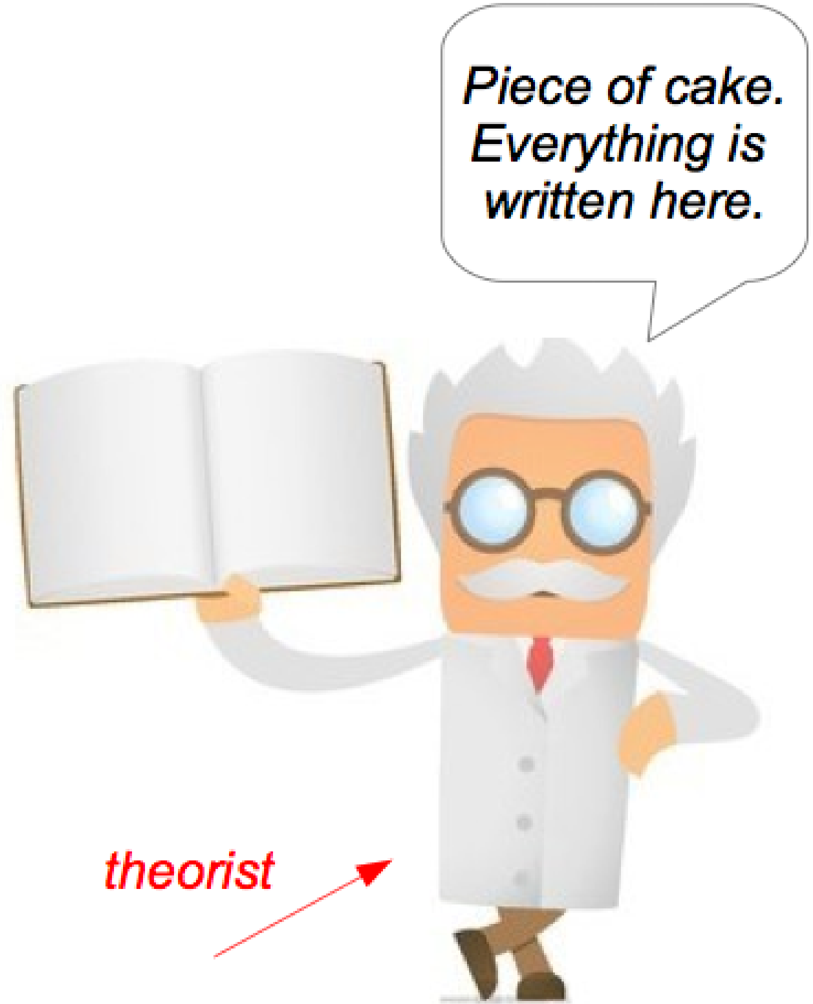
What do we want to simulate?

To boost the event rate, experiments use nuclear targets. Dozens of different isotopes present in our detectors, but typically the bulk of detector fiducial mass is made of C12, O16, Ar40, Fe56.
A Neutrino MC Generator needs to simulate, on an event-by-event basis, the types and 4-momenta of *all* final state particles produced in interactions of neutrinos with nuclei.



What does the theory tell us?

A Neutrino MC Generator needs to simulate, on an event-by-event basis, the types and 4-momenta of *all* final state particles produced in interactions of neutrinos with nuclei.



What does the theory tell us?

A Neutrino MC Generator needs to simulate, on an event-by-event basis, the types and 4-momenta of *all* final state particles produced in interactions of neutrinos with nuclei.

The input theory should be internally consistent and provide a neutrino-**nucleus** cross-section model which is:

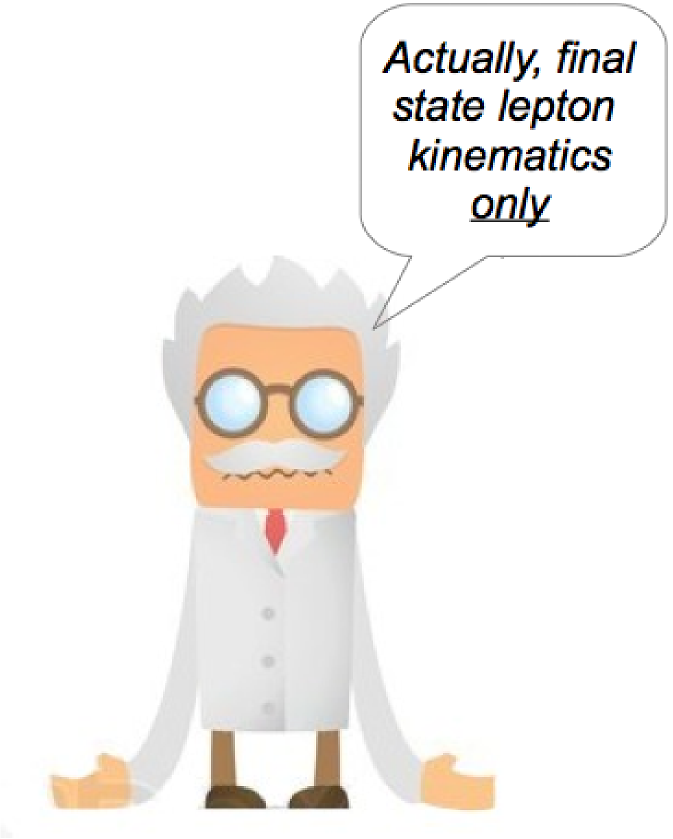
- fully differential over the kinematics of all final-state particles;
- taking into account all reaction mechanisms;
- valid in the full kinematical phase space accessible to experiments;
- valid for all combinations of $\nu/\bar{\nu}$, flavour (e,μ,τ), CC/NC and A (from Hydrogen to Lead).

What does the theory tell us?

A Neutrino MC Generator needs to simulate, on an event-by-event basis, the types and 4-momenta of *all* final state particles produced in interactions of neutrinos with nuclei.

Need a neutrino-**nucleus** cross-section model which is:

- fully differential over the kinematics of all final-state particles;
  - depending on the detector technology and analysis methods, all particles matter for energy reconstruction, event ID, signal efficiency and bkg contamination evaluations,...
  - theory models better at specifying the lepton system than specifying the hadronic system.
- taking into account all reaction mechanisms;
- valid in the full kinematical phase space accessible to experiments;
- valid for all combinations of $\nu/\bar{\nu}$, flavour (e,μ,τ), CC/NC and A (from Hydrogen to Lead).



What does the theory tell us?

A Neutrino MC Generator needs to simulate, on an event-by-event basis, the types and 4-momenta of *all* final state particles produced in interactions of neutrinos with nuclei.

Need a neutrino-**nucleus** cross-section model which is:

- fully differential over the kinematics of all final-state particles;
- taking into account all reaction mechanisms;
  - expts need to simulate *every process* that contributes to their observable distributions
  - e.g. higher multiplicity events will feed down if particles are misreconstructed or absorbed in the nucleus
  - best microphysical models not fully inclusive (?)
- valid in the full kinematical phase space accessible to experiments;
- valid for all combinations of $\nu/\bar{\nu}$, flavour (e,μ,τ), CC/NC and A (from Hydrogen to Lead).

What does the theory tell us?

A Neutrino MC Generator needs to simulate, on an event-by-event basis, the types and 4-momenta of *all* final state particles produced in interactions of neutrinos with nuclei.

Need a neutrino-**nucleus** cross-section model which is:

- fully differential over the kinematics of all final-state particles;
- taking into account all reaction mechanisms;
- valid in the full kinematical phase space accessible to experiments;
  - most current and future accelerator experiments have fluxes that peak before 5 GeV, but with tails that extend to many tens of GeV
  - backgrounds from high-energy event feed downs
  - single theory models usually not valid over the entire energy spectrum of experiments
- valid for all combinations of $\nu/\bar{\nu}$, flavour (e,μ,τ), CC/NC and A (from Hydrogen to Lead).

What does the theory tell us?

A Neutrino MC Generator needs to simulate, on an event-by-event basis, the types and 4-momenta of *all* final state particles produced in interactions of neutrinos with nuclei.

Need a neutrino-**nucleus** cross-section model which is:

- fully differential over the kinematics of all final-state particles;
- taking into account all reaction mechanisms;
- valid in the full kinematical phase space accessible to experiments;
- valid for all combinations of $\nu/\bar{\nu}$, flavour (e,μ,τ), CC/NC and A (from Hydrogen to Lead).
  - neutrino beams not pure
  - even if they were pure to start with, oscillations will bring other flavours in
  - ν_e/ν_μ, $\nu/\bar{\nu}$ important for appearance and CP
  - usually several targets in the same experiment
  - poor theory coverage

Quick discussion point

Now, more seriously, the nuclear theory community is strongly engaged and is providing important calculations that help the experimental community address some of the issues we are confronted with.

Need more of that!

Any comments from the theorists in the audience?

What is reasonable to expect from microphysical models in the future?

Can we get guidance from neutrino scattering data...

... to build effective models anchored on that data?

Indeed, many of our models are effective ones, grounded on solid but simple physics assumptions, and relying heavily on data. An iterative method is used (use model $\rightarrow$ extract constraints from new data $\rightarrow$ improve model).

However, the experiment has some of the same issues as theory.

- Very poor coverage in $\nu/\bar{\nu}$, flavour, A, kinematics, multiplicities...
- Important exclusive samples have a handful of measured events.

Be careful with embedding neutrino data directly in simulations: Using observables (mapped from unknown true quantities) from experiment X, to create the mapping between true and observable quantities in experiment Y (dog chasing tall)? Difficult with observables that have strong energy dependence.
And be careful with model-dependent corrections applied to data.

A survey of neutrino scattering data is presented by S.Boyd (Lecture E1).

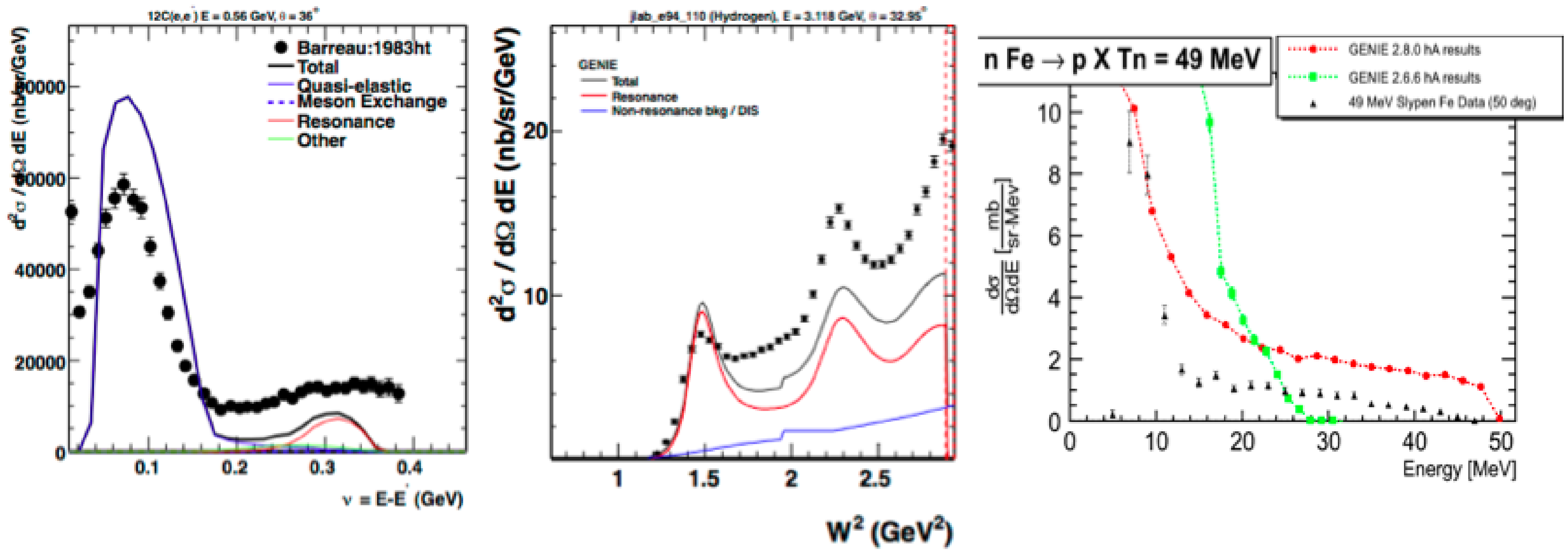
A lecture on unfolding techniques and their application producing model-independent neutrino cross-section measurements is given by M.Wascko (Lecture E3).

How about non-neutrino scattering data?

MC generators use several inputs from electron-nucleus and hadron-nucleus scattering.

Electron-nucleus scattering data can be used to constrain the vector part of the interaction, the nuclear model and the hadronic simulations. Hadron-nucleus scattering data can constrain the hadronic transport simulations.

Generator comparisons with electron and hadron scattering data reveal some striking model deficiencies. Much more work is needed to integrate data constraints from non-neutrino probes into our simulations.

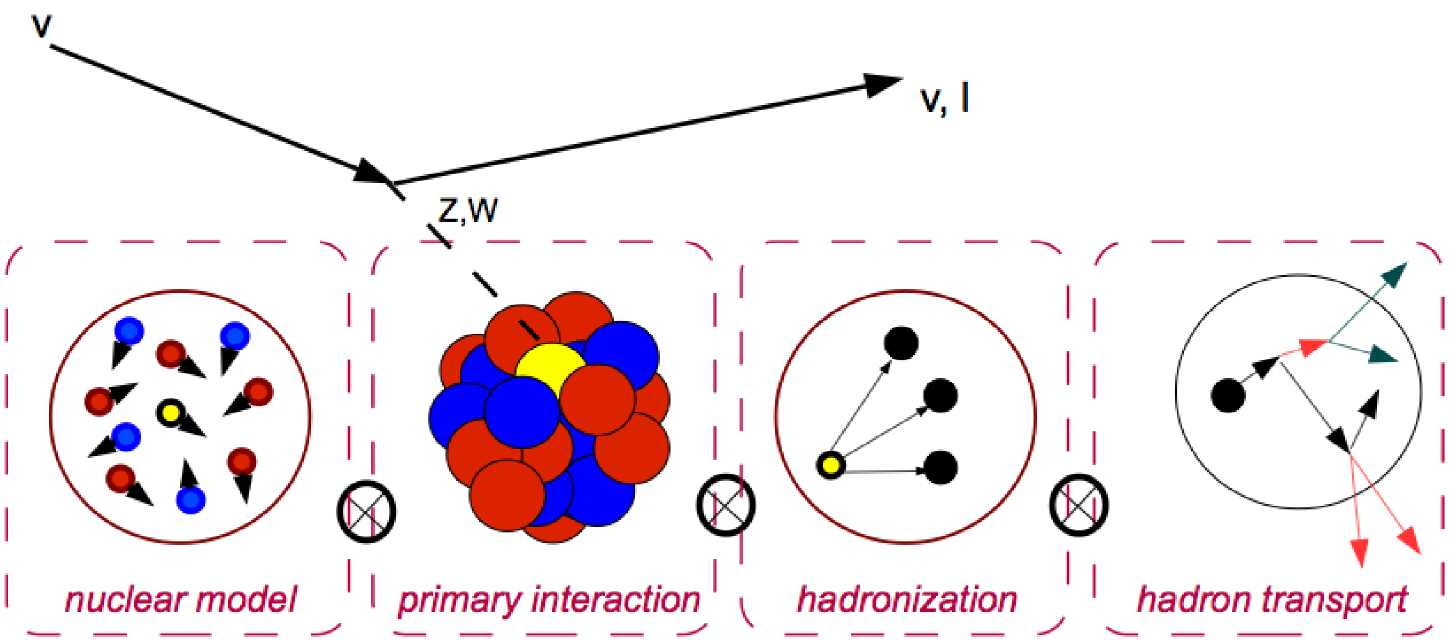


A lecture on electron-scattering data and its use in constraining neutrino interaction models is given by J.Nowak (Lecture E2).

Neutrino MC Generator factorization

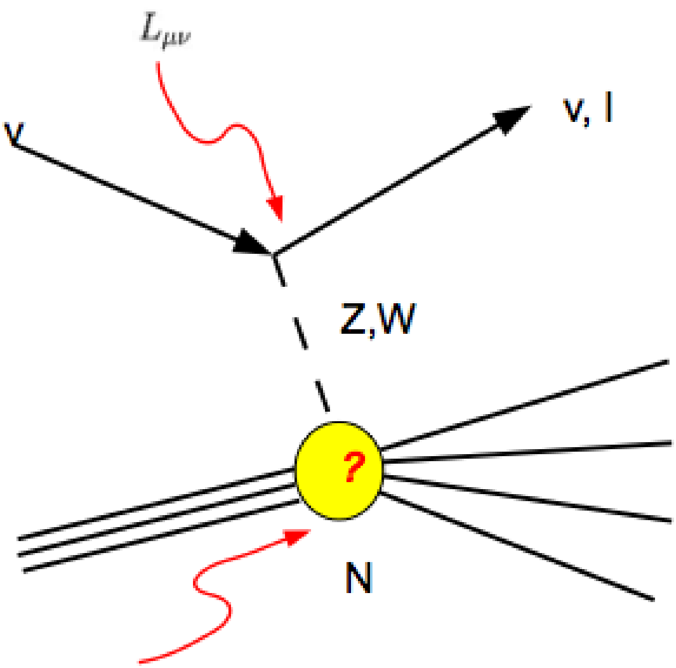
Since we do not have a complete theory of particle production in neutrino scattering off nuclear targets, simulation of exclusive final states proceeds in a *bottom-up fashion*, using models of:

- the initial nuclear state dynamics
- cross-sections at the neutrino-nucleon level (+ a model of how to sum-up the nucleon-level contributions)
- the process by which hadrons emerge from the primary interaction (hadronization)
- intranuclear hadron transport



Can the physics really be factorized this way? Unlikely!

Primary interaction



But the nucleon is not a simple object either!

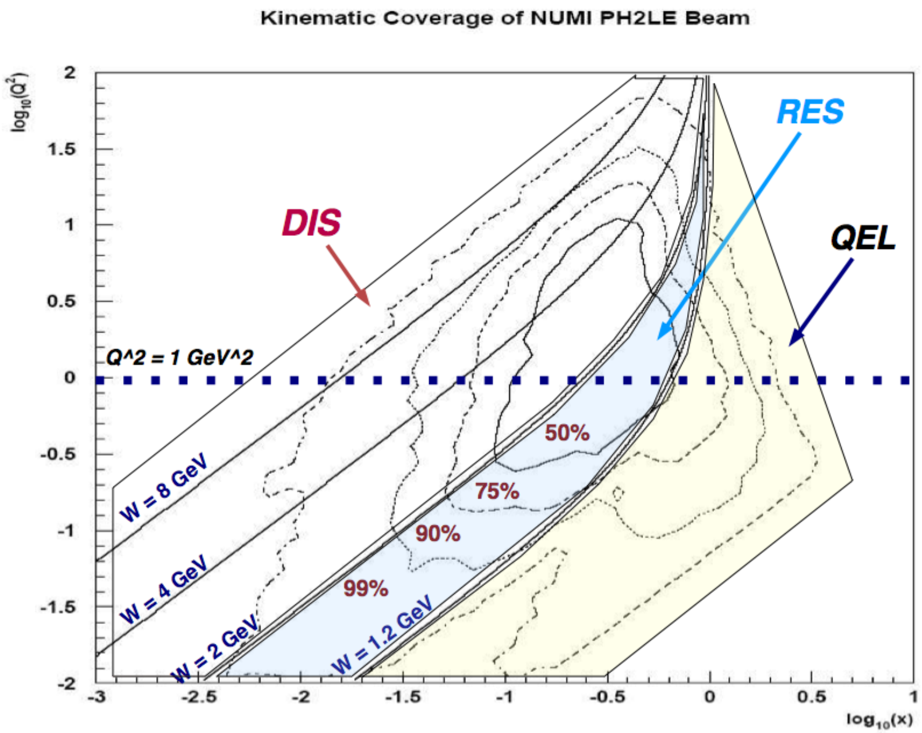
Process dynamics described by the invariant amplitude $|M|^2 = L_{\mu\nu} W^{\mu\nu}$

Knowledge of W_1, W_2, ...?
In a kinematical regime that bridges the non-perturbative and perturbative pictures of the nucleon.

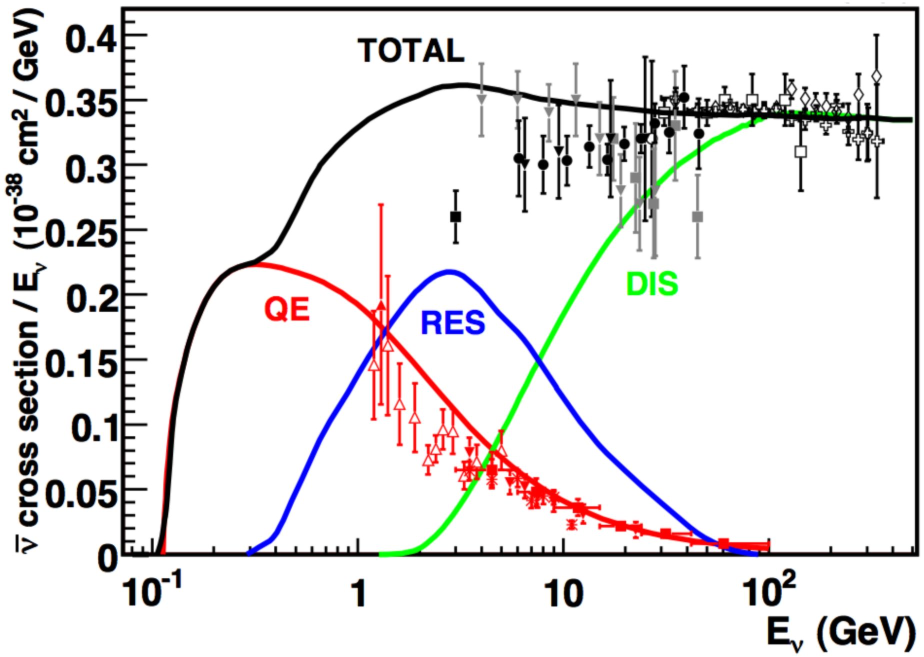
A lecture on the basics of electro-weak interactions by L.Alvarez-Ruso (Lecture T3).

$$W_{\mu\nu} = W_1 \delta_{\mu\nu} + W_2 p_\mu p_\nu + W_3 \epsilon_{\mu\nu\alpha\beta} p^\alpha p^\beta + W_4 q_\mu q_\nu + W_5 (p_\mu q_\nu + p_\nu q_\mu) + W_6 (p_\mu q_\nu - p_\nu q_\mu)$$

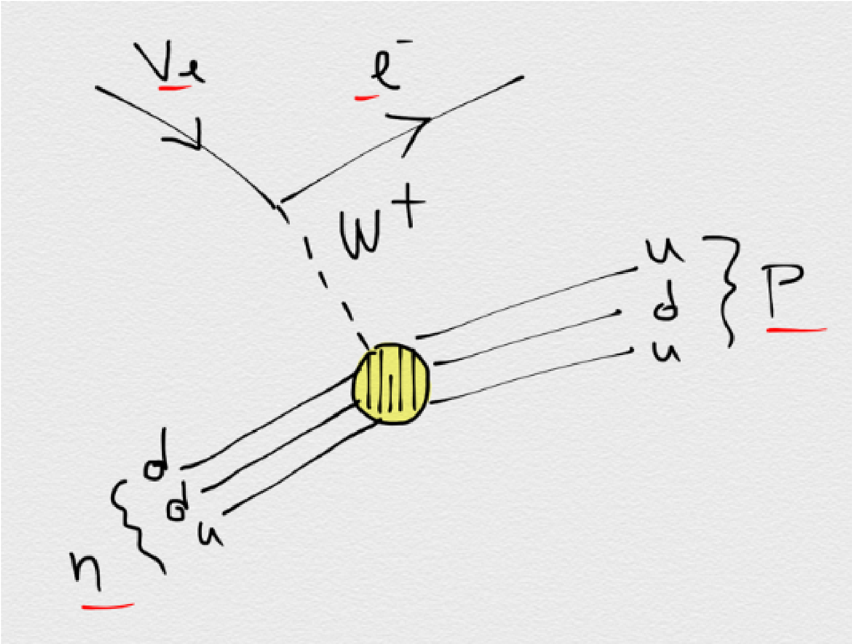
In the broad energy range relevant to current and near future experiments, several scattering mechanisms are important.

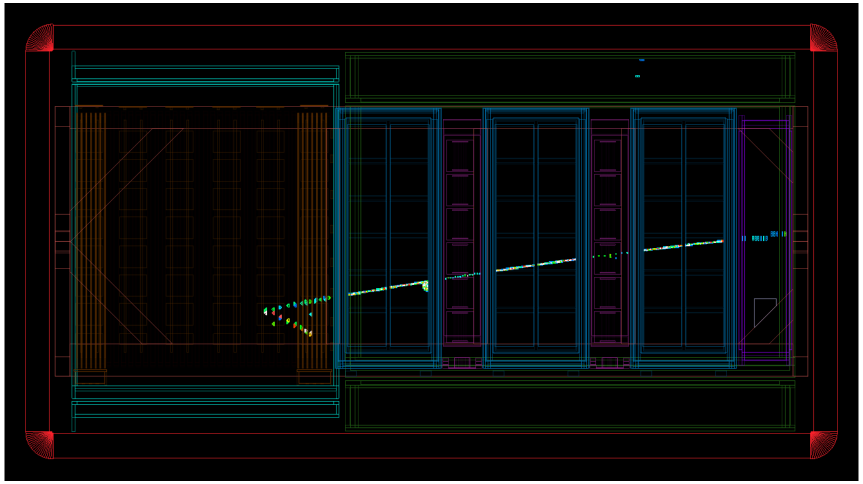


[Hugh Gallagher]



Quasielastic scattering





The "easiest" important process at $\sim$ 1 GeV.
The neutrino scatters elastically off the nucleon (does not excite or break it up).
A clean and distinct final state topology.

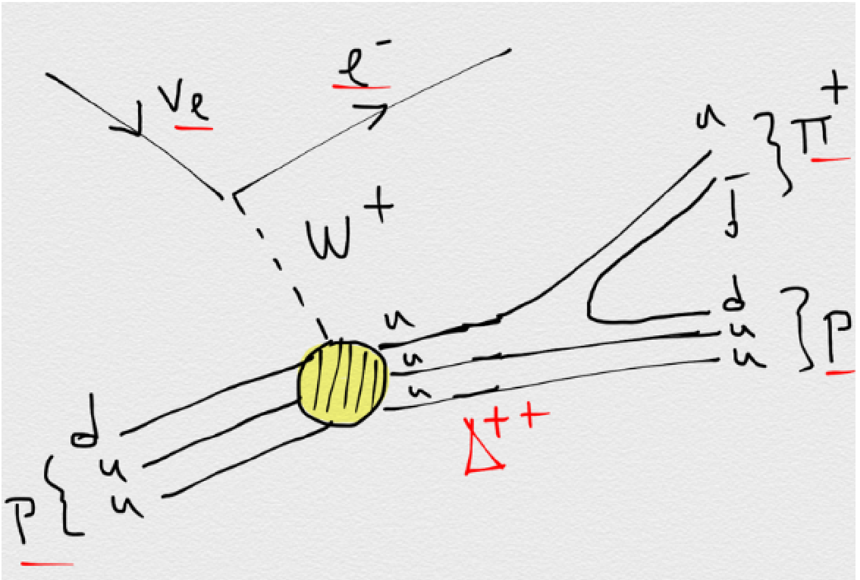
However, significant complications exist in the nuclear medium (Pauli blocking, offshelness of hit nucleon, the role of other nucleons and nucleon-nucleon correlations).

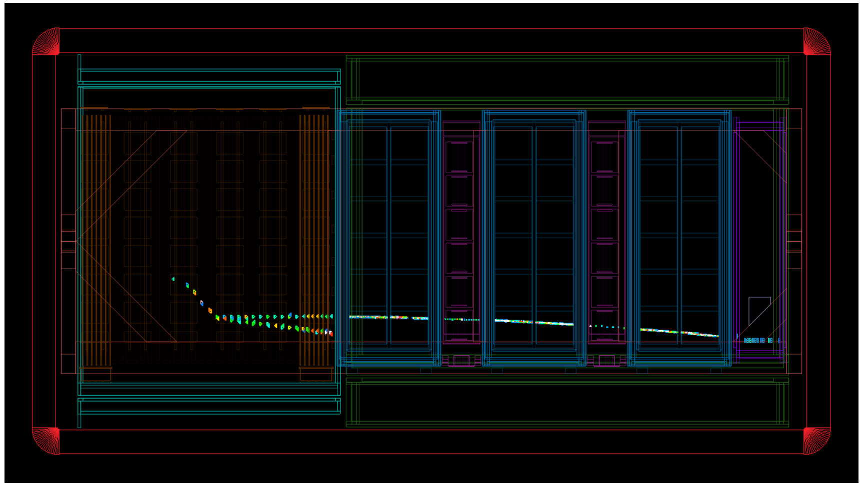
Lecture on quasielastic neutrino scattering by L.Alvarez-Ruso (Lecture T5).

Lecture on Np-Nh contributions to quasielastic-like neutrino scattering by J.Sobczyk (Lecture T6).

Event display from http://homepages.warwick.ac.uk/$\sim$phrfba/

Neutrino production of resonances

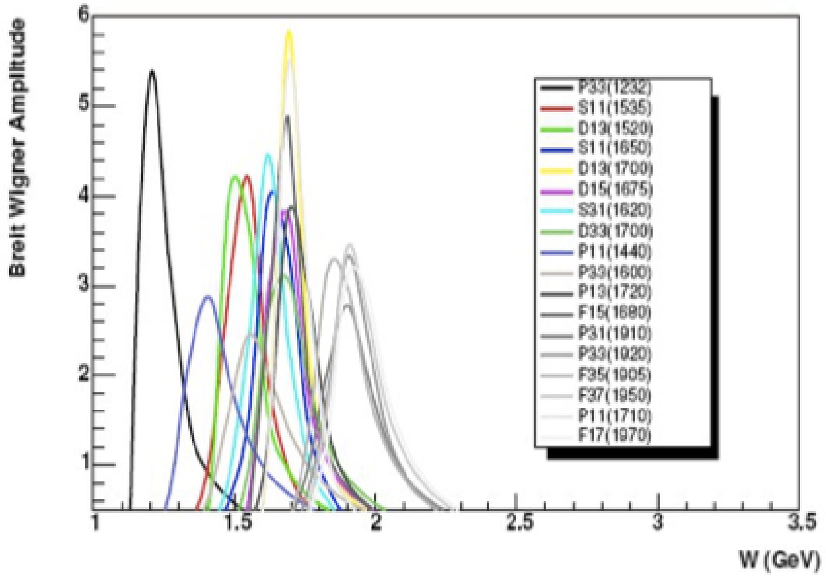




Event display from http://homepages.warwick.ac.uk/∼phrfba/

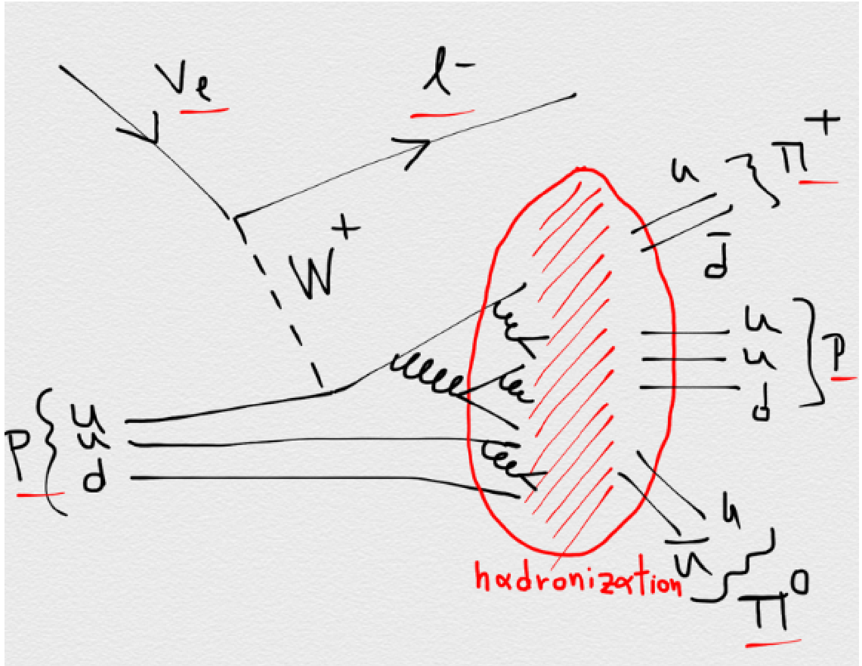
Things get more difficult. Now, the neutrino excites the hit nucleon to a resonance with complex dynamics.

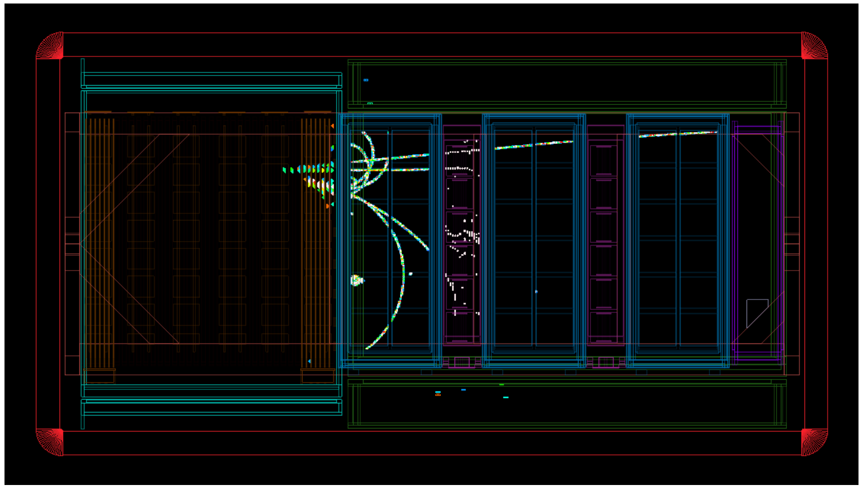
About 18 baryon resonances exist between 1.2 and 2 GeV. Resonances can interfere.



Lecture on neutrino-production of resonances by L.Alvarez-Ruso (Lecture T7).

Deep inelastic neutrino scattering





Event display from http://homepages.warwick.ac.uk/∼phrfba/

Things get easy again: Elastic scattering off a quark.

However, our deep inelastic scattering is not all that deep. Large fraction of DIS events at our experiments ar low Q^2 and high x, where structure functions are not well known.

Also complications exist for nuclear targets (shadowing, anti-shadowing, EMC) with virtually no neutrino data to constrain these effects (these effects were measured with charged lepton probes, but for charged lepton scattering the structure functions are made up from different combinations of quark PDFs.)

Neutrino-induced hadronization for invariant masses below ∼5 GeV, also quite difficult to model.

Lecture on deep inelastic scattering and neutrino-induced hadronization by J.Sobczyk (Lecture T8).

Other processes

The generators also handle many additional processes, and handle separately processes with very distinct topologies which may be of interest to the experimental community:

- charm production
- inelastic single-Kaon
- $\Delta S=1$ QE processes
- coherent elastic
- coherent production of π, ρ,...
- neutrino-electron elastic
- inverse muon decay
- ...

We have no time to cover all that during the school, but we will cover a process which is both very interesting theoretically and of interest to the experimental community due to its distinct topology and role as a ν_e appearance background.

Lecture on weak coherent meson production by L.Alvarez-Ruso (Lecture T9).

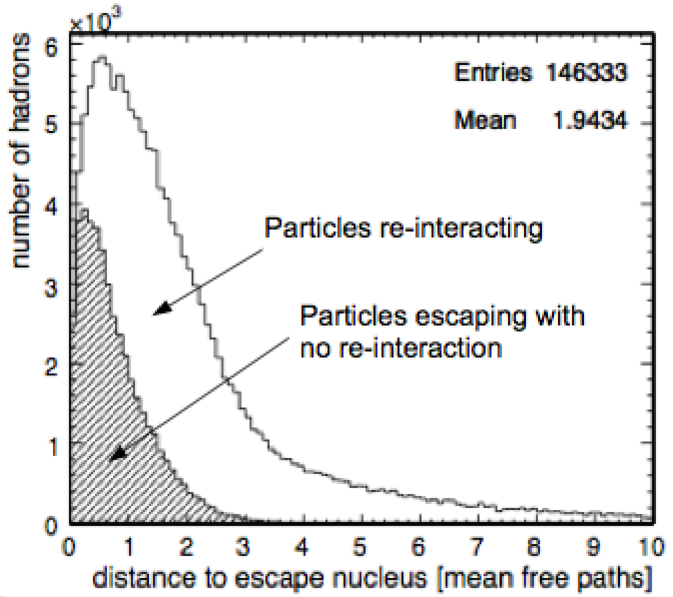
Nuclear modelling

An important modelling aspect, depending on the kinematics of the reaction (i.e. important for QE and less important for DIS).

It introduces coupling between several aspects of the simulation (cross-section calculation, intranuclear rescattering).

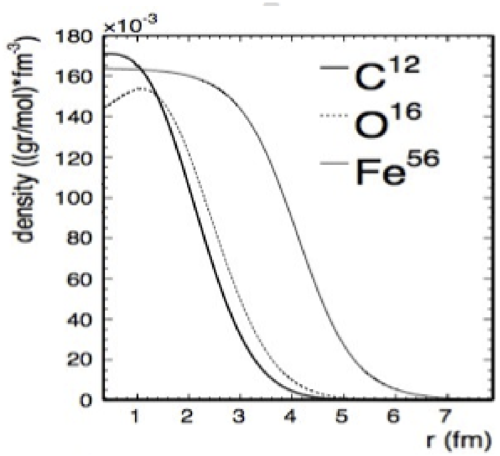
Lecture on the nuclear initial state & the basics of many-body theory by L.Alvarez-Ruso (Lecture T4).

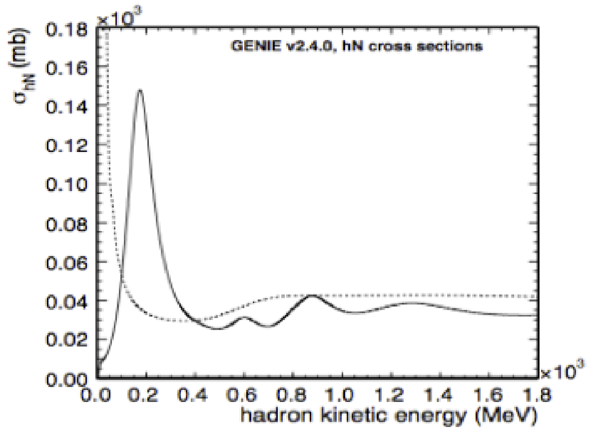
At ∼1 GeV, 2/3 of all primary hadrons re-interact in the nucleus.



$$P^h_{rescat} = 1 - \int dr e^{-r/\lambda^h(\vec{r},E_h)}$$

$$\lambda^h(\vec{r}, E_h) = 1/(\rho_{nucl}(r) \cdot \sigma^{hN}(E_h))$$

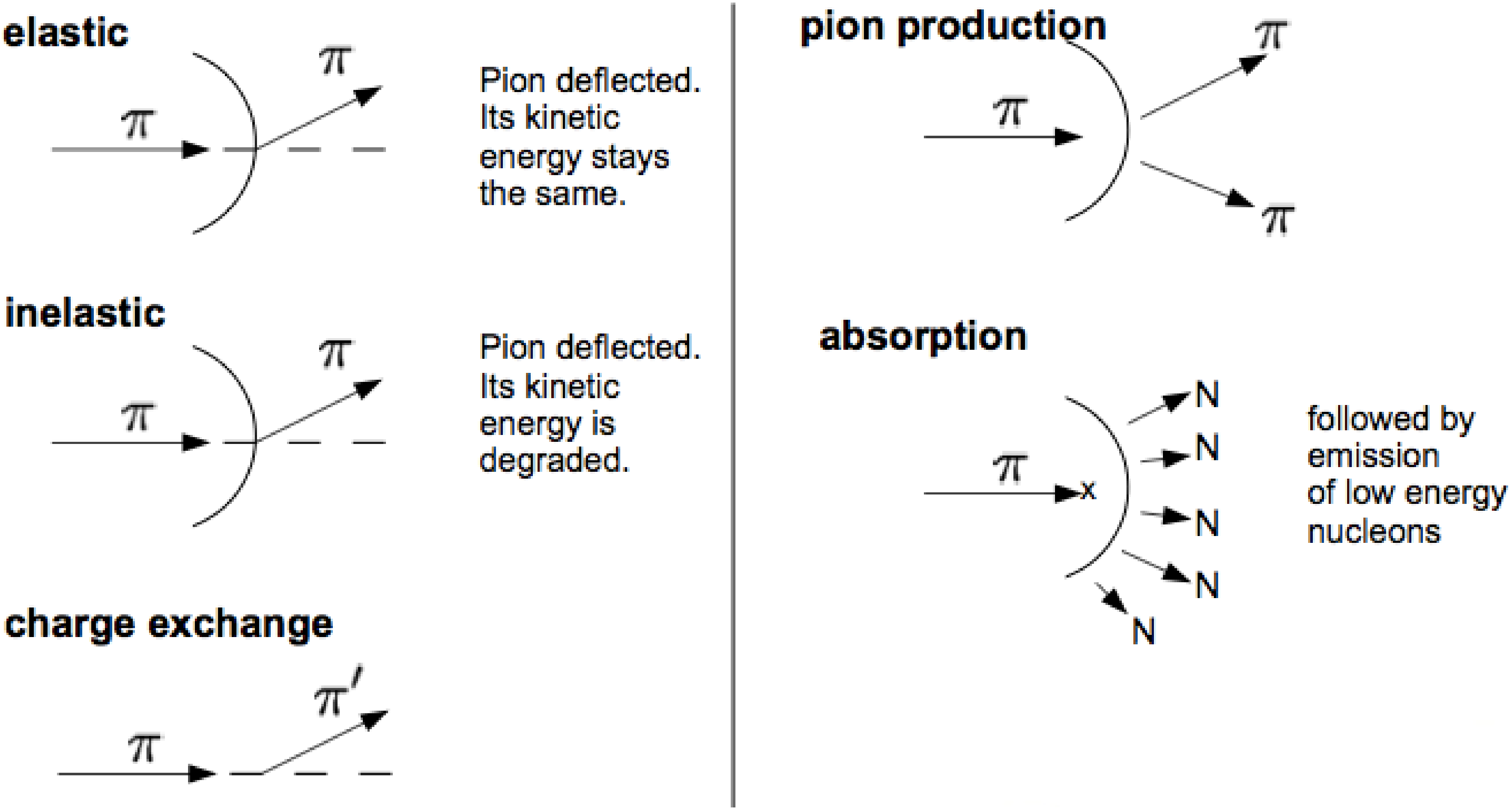




$\nu_\mu + C^{12}$, 1 GeV

Intranuclear hadron transport

A number of scattering mechanisms are possible.



Intranuclear hadron transport

Intranuclear rescattering degrades the hadron energy, smears their angular distributions and mixes the event topologies.

A 1-pion event, where the pion is absorbed in the nucleus, will appear as a QE-like event.
On the other hand, a QE event with a recoil nucleon that re-interacts and produces a pion, may appear 1-pion-like.

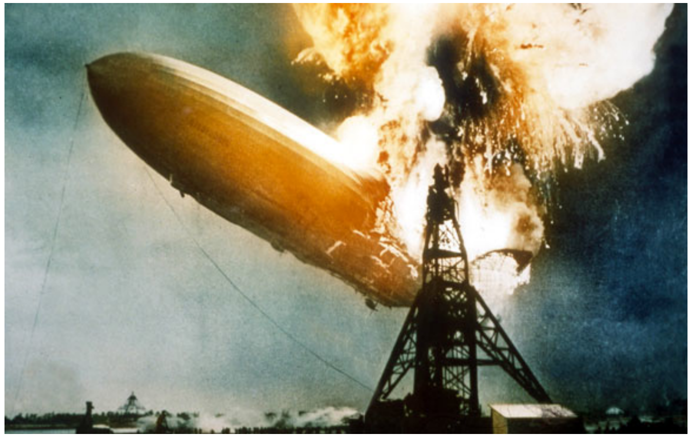
$\nu_\mu + O^{16}$, T2K spectrum

Final-State	Primary Hadronic System									
	$0\pi X$	$1\pi^0 X$	$1\pi^+ X$	$1\pi^- X$	$2\pi^0 X$	$2\pi^+ X$	$2\pi^- X$	$\pi^0\pi^+ X$	$\pi^0\pi^- X$	$\pi^+\pi^- X$
$0\pi X$	**293446**	12710	22033	3038	113	51	5	350	57	193
$1\pi^0 X$	1744	**44643**	3836	491	1002	25	1	1622	307	59
$1\pi^+ X$	2590	1065	**82459**	23	14	660	0	1746	5	997
$1\pi^- X$	298	1127	1	**12090**	16	0	46	34	318	1001
$2\pi^0 X$	0	0	0	0	**2761**	2	0	260	40	7
$2\pi^+ X$	57	5	411	0	1	**1999**	0	136	0	12
$2\pi^- X$	0	0	0	1	0	0	**134**	0	31	0
$\pi^0\pi^+ X$	412	869	1128	232	109	106	0	**9837**	15	183
$\pi^0\pi^- X$	0	0	1	0	73	0	8	5	**1808**	154
$\pi^+\pi^- X$	799	7	10	65	0	0	0	139	20	**5643**

Lecture on final state interactions & a survey of intranuclear cascade codes by J.Sobczyk (Lecture T10).

Lack of reliable nucleon level data

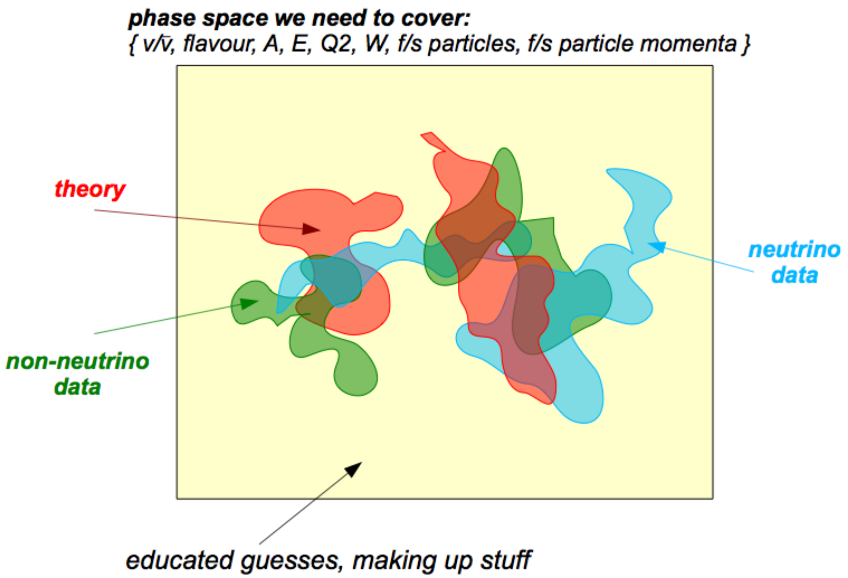
- Free-nucleon cross-sections are a key ingrendient in how MCs build nuclear cross-sections.
- Precise knowledge of cross-section at nucleon level is required.
- Unfortunately we do not have such data either.
- Existing nucleon level data are mainly from the bubble chamber era.
- No neutrino data with Hydrogen or Deuterium targets from any recent experiment.



- Such data would have been extremely usefull in resolving some of modelling degeneracies and understanding some of recent and seemingly odd results.
- We need to get over Hindenburg

The art of developing Neutrino Monte Carlos

Developing a neutrino Monte Carlos Generator is the *art* of:



- Using the best theory models, each one in the region where it is valid.
- Determining how to best extrapolate and merge models outside of their stated validity range.
- Developing empirical models to bridge the gaps and cover the full required phase space.
- Validating and tuning each specific model using a variety of complementary data.
- Validating and tuning the overall comprehensive model, testing how specific models get extrapolated and merged, giving special care to avoid double-counting and keeping, to the extend that is possible, a consistent view of physics.
- Maintaining CPU-efficiency, as experiments require high-statistics samples.
- Fully evaluating all sources of model uncertainty (to do physics, knowledge of the error on the model is as necessary as the model itself)

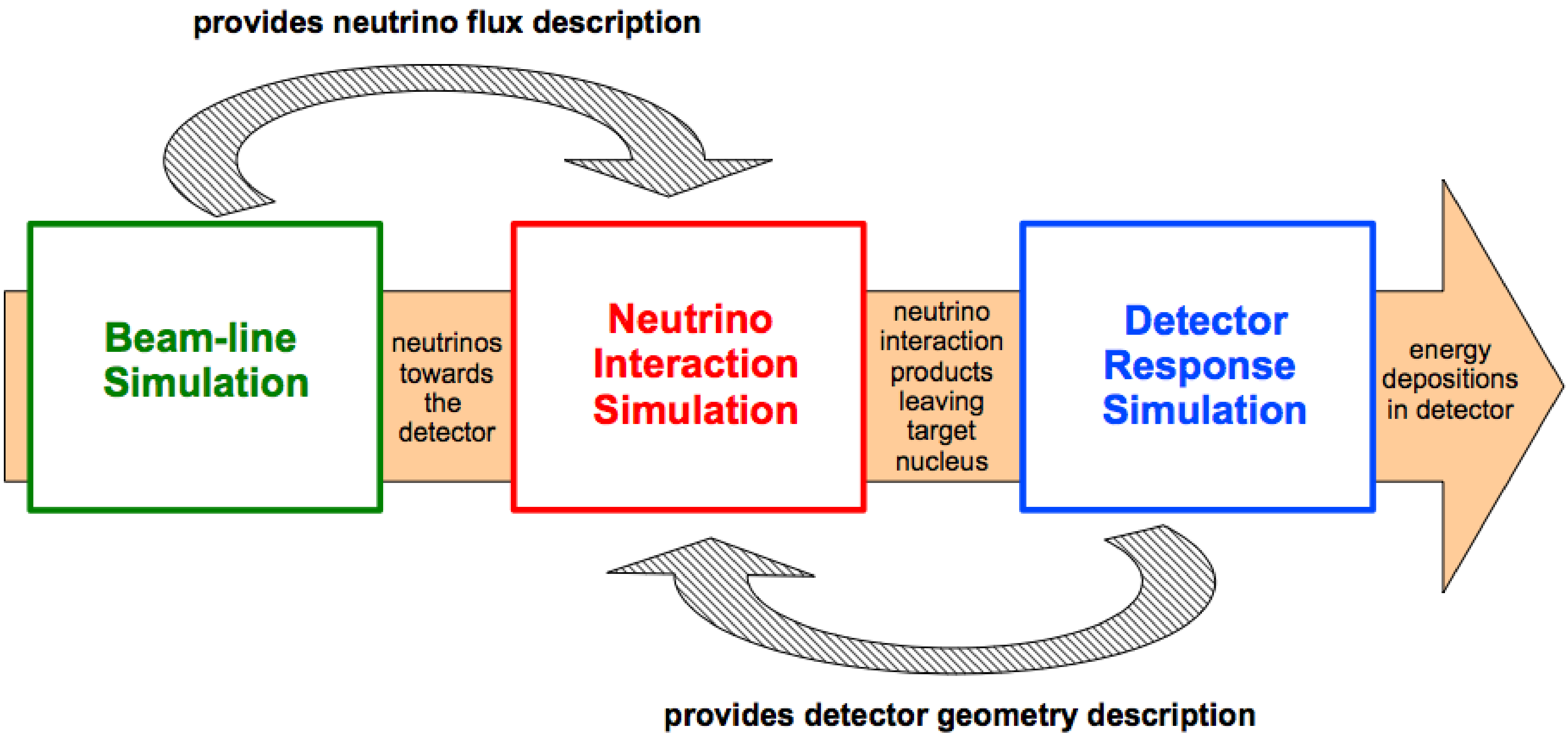
The future of Neutrino Generators

Probably another good point for a quick discussion.

Also a good discussion for all of us to have over a wine or beer this evening, following Steve's talk.

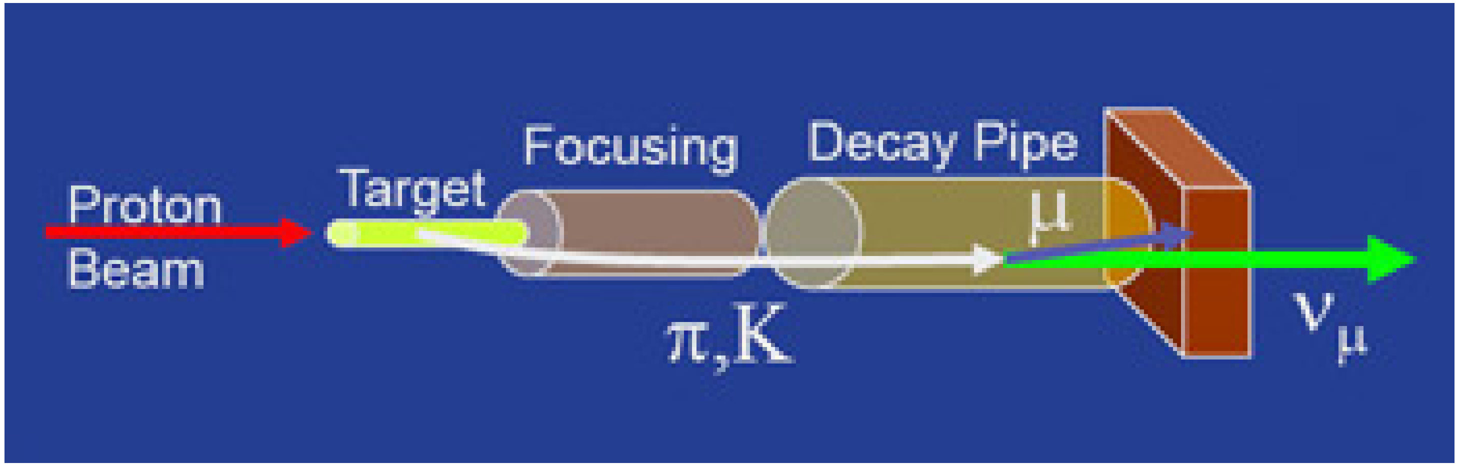
Generating a neutrino interaction event: A detailed walkthrough

A typical accelerator neutrino experiment simulation chain



Simulation chain upstream the Neutrino Generator

Experiments run detailed FLUKA/Geant simulations of their neutrino beamlines. These simulations start from the primary proton beam interactions in the target and track all daughter particles simulating their subsequent interactions and decays. The beam-line simulations typically produce "neutrino flux rays" that are fed into the neutrino generator.



Where do our ν's come from?

$\pi^+ \to \nu_\mu + \mu^+$
$\pi^- \to \bar{\nu}_\mu + \mu^-$
$\mu^+ \to \bar{\nu}_\mu + \nu_e + e^+$
$\mu^- \to \bar{\nu}_e + \nu_\mu + e^-$
$K^+ \to \nu_\mu + \mu^+$
$K^+ \to \nu_e + \pi^0 + e^+$
$K^+ \to \nu_\mu + \pi^0 + \mu^+$
$K^- \to \bar{\nu}_\mu + \mu^-$
$K^- \to \bar{\nu}_e + \pi^0 + e^-$
$K^- \to \bar{\nu}_\mu + \pi^0 + \mu^-$
$K_L^0 \to \bar{\nu}_\mu + \pi^+ + \mu^-$
$K_L^0 \to \nu_\mu + \pi^- + \mu^+$
$K_L^0 \to \bar{\nu}_e + \pi^+ + e^-$
$K_L^0 \to \nu_e + \pi^- + e^+$

Generator tasks

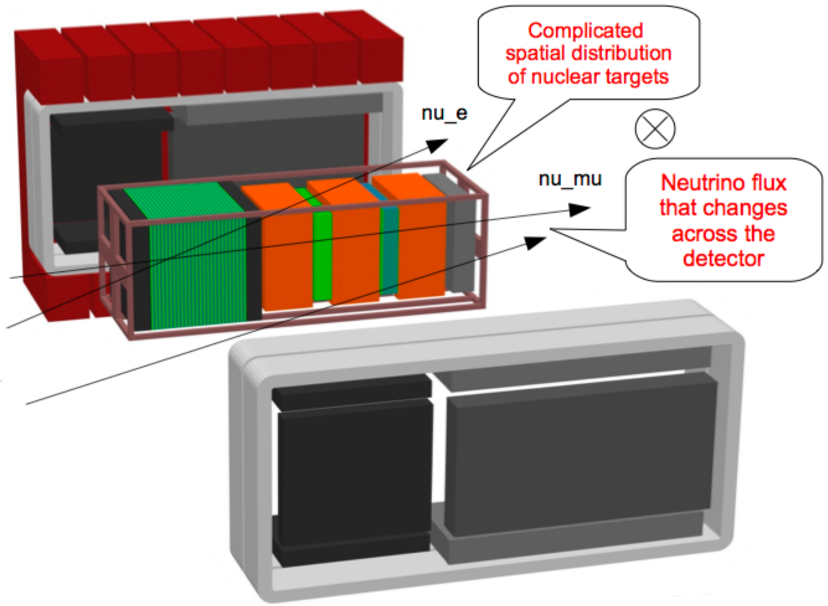
The beam-line simulations typically produce "neutrino flux rays" that are fed into the neutrino generator.

The neutrino generator task is then to:

- Propagate the neutrino rays towards the detectors.
- Establish whether the neutrino interacts and where exactly.
- Select an interaction type for the given neutrino and target.
- Generate all final state particles for the selected interaction.

Does the flux neutrino interact?

An easy question in principle. In practise, this is one of the most complex and CPU-intensive calculations generators do.



- Detectors not uniform, with a complex distribution of nuclear targets
- Neutrino fluxes also not uniform for detector locations close to the neutrino target, changing across the detector
- For CPU efficiency, experiments need to be able to consider events in specific volumes only, sometimes not related with any actual geometry volume. Need to be able to mask out arbitrarily defined parts of the geometry.
- Experiments also want to consider interactions in large external volumes (eg surrounding rock) to estimate beam-related backgrounds.
- Avoid oddities from recycling flux rays.
- Keep track of the absolute sample normalization in terms of protons on target.

Does the flux neutrino interact?

The fundamental problem here is to compute a complex multi-dimensional integral

$$N_{ev} \propto \int dE_\nu dcos\theta_\nu d\phi_\nu dxdydt \sum_{f(lavor)} \frac{d^6\Phi_f(E_\nu, cos\theta_\nu, \phi_\nu, x, y, t)}{dE_\nu dcos\theta_\nu d\phi_\nu dxdydt} \int_0^\infty ds \sum_{i(sotope)} \frac{\rho(\vec{r}) w_i(\vec{r}) \sigma_{f;i}^{tot}(E_\nu)}{A_i(\vec{r})}$$

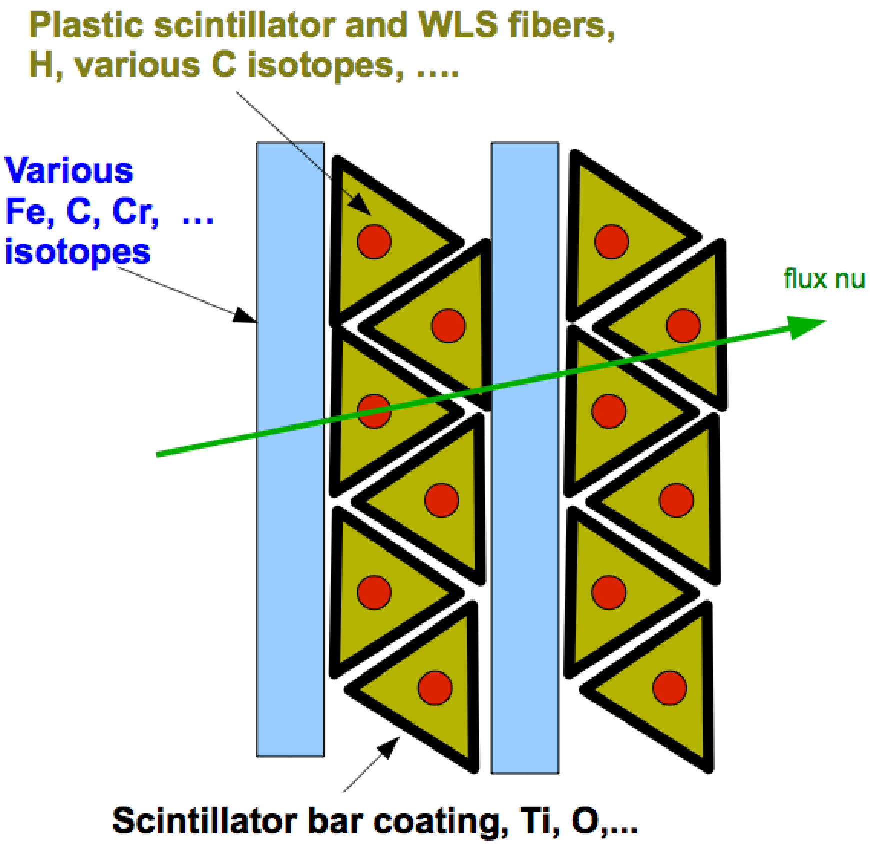
for which an analytical answer is not possible.

- E_ν: neutrino energy
- $cos\theta_\nu$: flux neutrino zenith angle
- ϕ_ν: flux neutrino azimuthal angle
- $dxdy$: unit area in some surface upstream of any volume where we consider interactions
- t: time
- ds: infinitesimal step along the neutrino direction
- $\vec{r} = \vec{r}(s, cos\theta_\nu, \phi_\nu, x, y)$: position along the neutrino ray in the detector coordinate system
- $\frac{d^6\Phi_f}{dE_\nu dcos\theta_\nu d\phi_\nu dxdydt}$: differential number of neutrinos of flavour f
- $\rho(\vec{r})$: detector density at position $\vec{r}$ (typically a mixture)
- $w_i(\vec{r})$: weight fraction for isotope i, in the mixture at position $\vec{r}$
- $A_i(\vec{r})$: mass number for isotope i, in the mixture at position $\vec{r}$
- $\sigma_{f;i}^{tot}(E_\nu)$: total cross-section for interactions of neutrinos of flavour f with isotope i, at energy E_ν.

Does the flux neutrino interact?

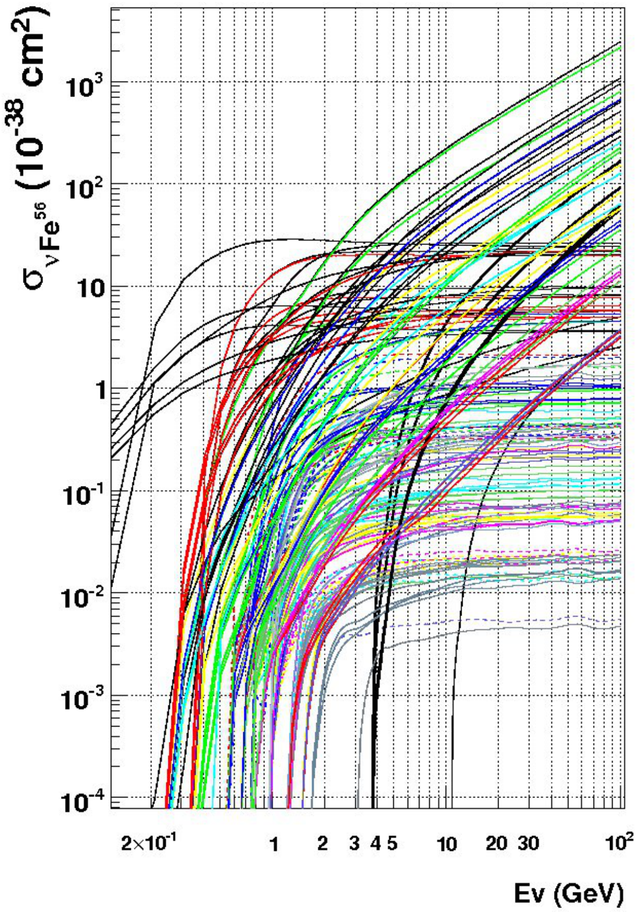
The integration involves:

- Pull random flux rays from the neutrino beam-line simulation and "throw" towards the detector.
- Follow the ray, stepping through the geometry and, in the absence of external fiducial cuts, getting all segments described by the entry and exit points of the flux ray in every single volume it passes through.
- For each segment, which is purely within a volume, figure out the corresponding mixture information (density, isotopic composition, weight fraction for each isotope) and calculate the density-weighted path length for each isotope seen along the trajectory.



Does the flux neutrino interact?

$\nu_\mu, \bar{\nu_\mu} + Fe$, all processes



- The generator needs to calculate the total cross-section, at the given energy, for the given neutrino flavour and target isotope.
- The generator then needs to look at all the physics processes enabled, and sum-up their cross-section.
- Calculation of each cross-section requires numerical integration of the corresponding differential cross-section model.
  - $\sim 10^2$ isotopes in typical detector geometries
  - $\sim 10^2$ interaction modes per given initial state (neutrino+isotope)
  - $\sim 10^4$ differential cross-section evaluations per numerical integration
  - $\sim 10^8$ **differential cross-section evaluations to decide whether a neutrino interacts**
- All generators, in one way or another, pre-compute the numerical integrals for a series of neutrino energies and then intrepolate.

Does the flux neutrino interact?

The interaction probability for each isotope is calculated using the **integrated density-weighted path length**, along the neutrino trajectory, and the **total interaction cross-section**, at the given energy, as:

$$P \propto \frac{\rho \cdot L \cdot \sigma^{tot}}{A}$$

As it is well known, these interaction probabilities are exceedingly small.

But the absolute number does not matter that much for the generator.

One needs to find the maximum possible interaction probability P_{max} for the current MC run (obtained with a neutrino that has the highest allowed energy and which travels along the trajectory that sees the most detector mass).

P_{max} can then be used to scale all interaction probabilities and improve CPU efficiency (there is an associated weight, but all events have that same weight so, effectively, the sample is still unweighted - effectively one just redefines the sample normalization)

Does the flux neutrino interact?

However there is **fundamental source of inefficiency**:

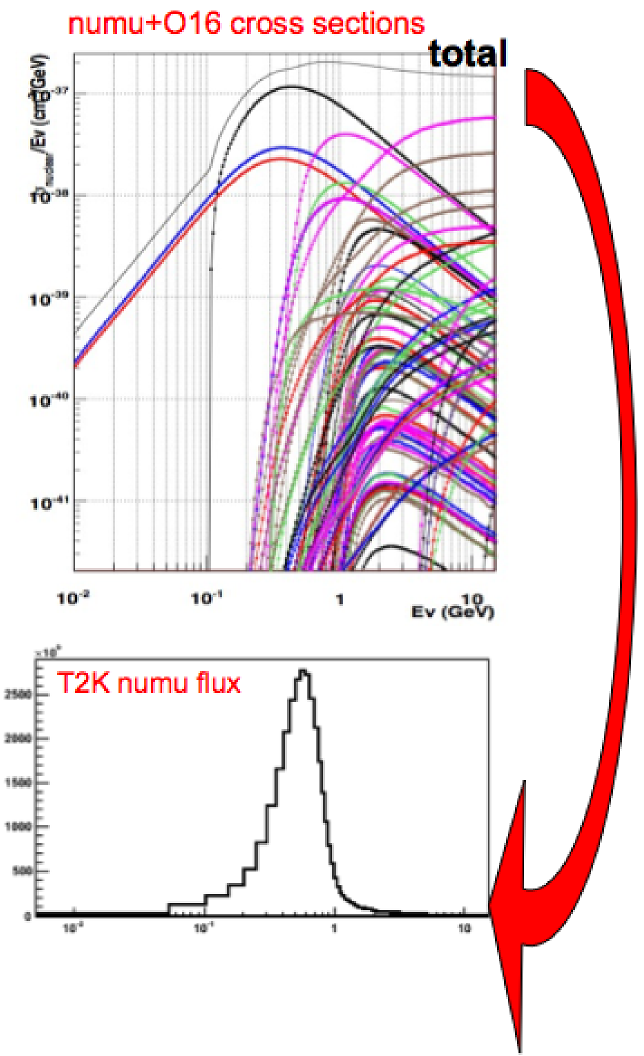
P_{max} is obtained for an event at the tail of the energy spectrum ($\sim$100 Gev) for a flux that typically peaks around $\sim$ 1 GeV.

Since the cross-section increases 100-fold between 1 and 100 GeV, the maximum interaction probability will be more than 100 times larger than the values most likely obtained with typical neutrinos. This means that, even after re-normalizing the interaction probabilities, hundreds of flux ν's need to be to be thrown before one interacts.

Several optimizations are typically implemented to make that work efficiently.

A very efficient work-around would have been to generate weighted events, but it is difficult to combine multiple weighted events into a single simulated beam spill (multiple events are seen in the detector during each beam spill).

Remember all above next time you wonder why the simulation speed dropped as soon as you included a flux and geometry description

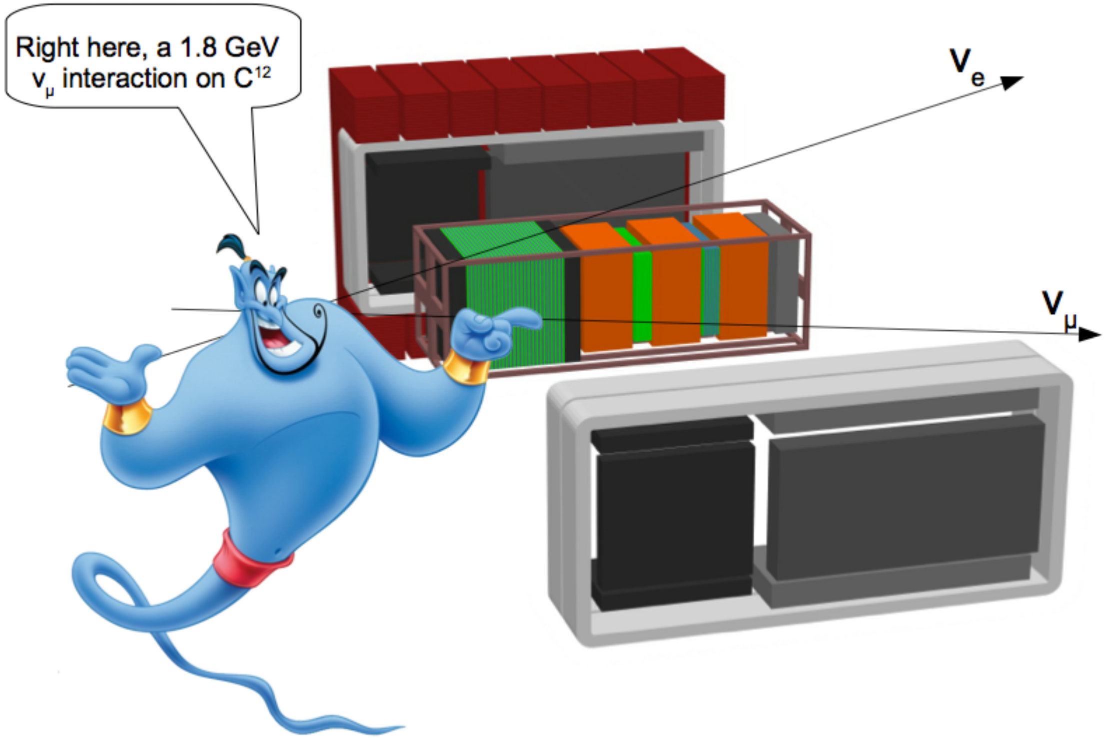


Does the flux neutrino interact?

Eventually the generator decides that the neutrino interacted with isotope i.

It also generates a vertex position along the neutrino trajectory, by generating a random density-weighted path length between 0 and the maximum one computed for the given trajectory and isotope.
The generator steps through the geometry once again following the neutrino trajectory so as to make sure that the vertex is placed in the appropriate position, in a volume that does contain isotope i.

Then, the next generator task is to select an actual interaction type and generate the final state particles.



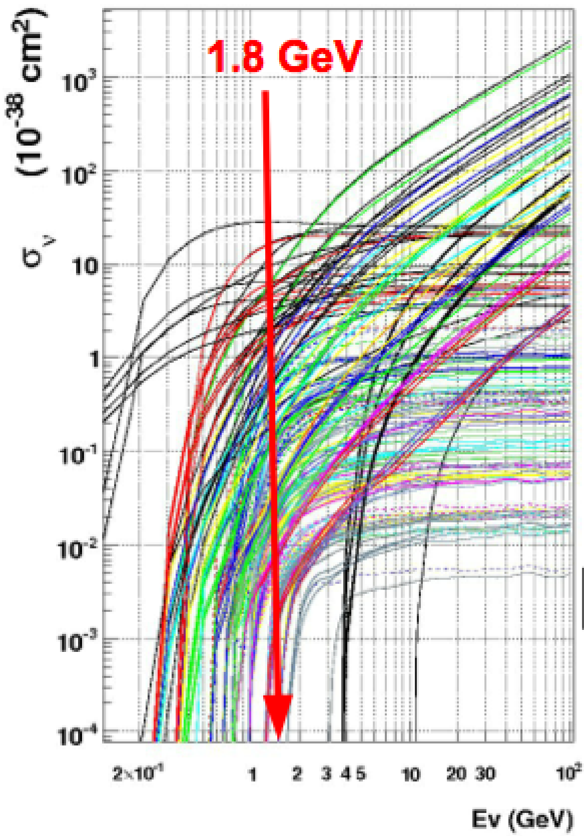
Tutorial by G.Perdue on using fluxes and detector geometries and customizing GENIE for a complex experimental setup (Tutorial 7).

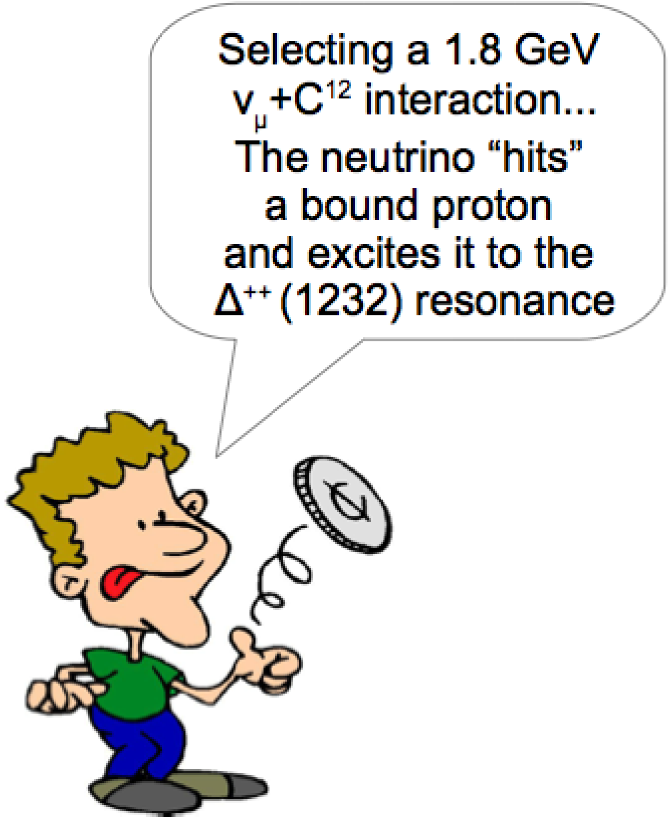
Generating event kinematics

```
|------------------------------------------------------------------------------------------------------------------|
|GENIE GHEP Event Record [print level:   3]                                                                        |
|------------------------------------------------------------------------------------------------------------------|
| Idx |          Name | Ist |        PDG |   Mother  | Daughter  |       Px |       Py |       Pz |        E |        m  |
|------------------------------------------------------------------------------------------------------------------|
|   0 |         nu_mu |   0 |         14 |  -1 |  -1 |           |    0.000 |    0.000 |    1.800 |    1.800 |    0.000 |
|   1 |           C12 |   0 | 1000060120 |  -1 |  -1 |           |    0.000 |    0.000 |    0.000 |   11.179 |   11.179 |
```

Generating event kinematics

```
|--------------------------------------------------------------------------------------------------------------|
|GENIE GHEP Event Record [print level:   3]                                                                   |
|--------------------------------------------------------------------------------------------------------------|
| Idx |           Name | Ist |        PDG |   Mother   | Daughter   |      Px |      Py |      Pz |       E |       m |
|--------------------------------------------------------------------------------------------------------------|
|   0 |          nu_mu |   0 |         14 |  -1 |  -1 |            |   0.000 |   0.000 |   1.800 |   1.800 |   0.000 |
|   1 |            C12 |   0 | 1000060120 |  -1 |  -1 |            |   0.000 |   0.000 |   0.000 |  11.179 |  11.179 |
```

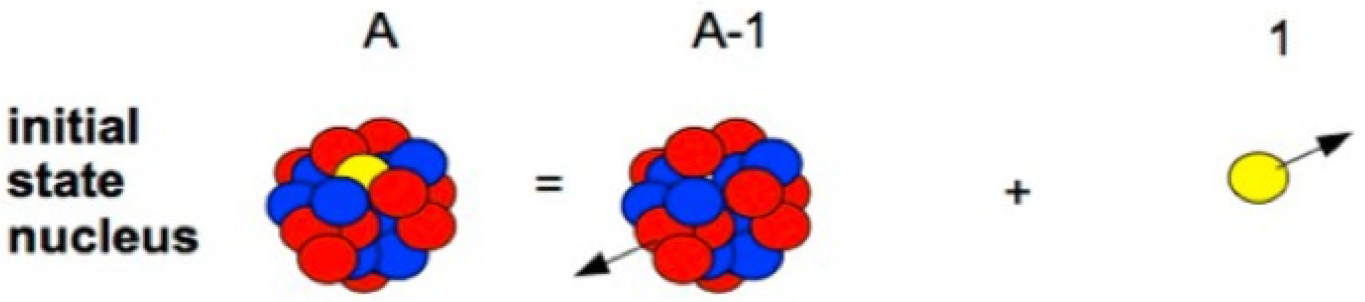




Generating event kinematics

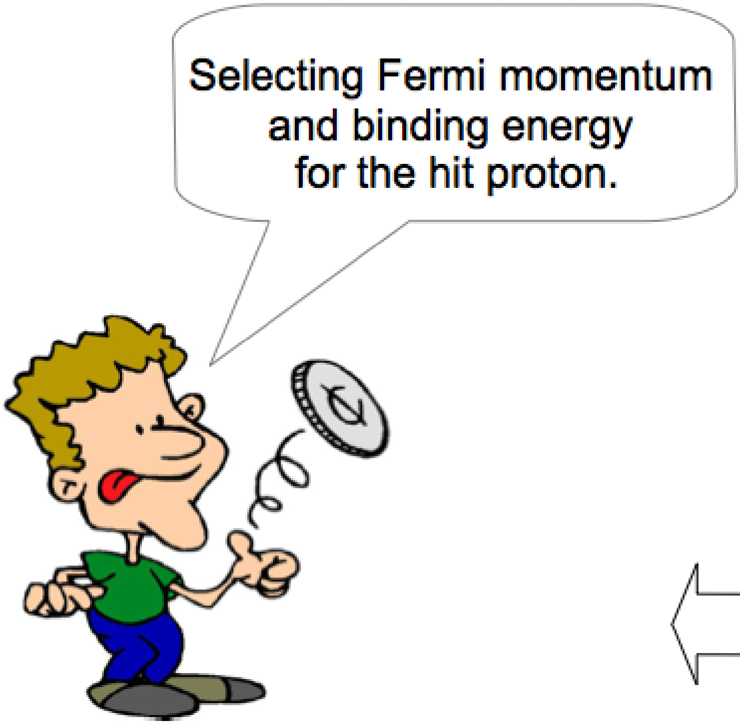
GENIE GHEP Event Record [print level: 3]

Idx	Name	Ist	PDG	Mother		Daughter		Px	Py	Pz	E	m
0	nu_mu	0	14	-1	-1			0.000	0.000	1.800	1.800	0.000
1	C12	0	1000060120	-1	-1	2	3	0.000	0.000	0.000	11.179	11.179
2	proton	11	2212	1	-1							
3	B11	2	1000050110	1	-1							

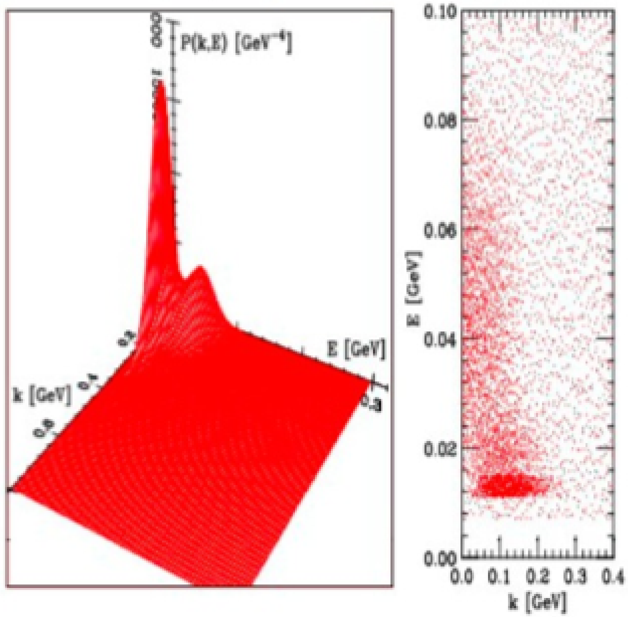


GENIE GHEP Event Record [print level: 3]

Idx	Name	Ist	PDG	Mother		Daughter		Px	Py	Pz	E	m	
0	nu_mu	0	14	-1	-1			0.000	0.000	1.800	1.800	0.000	
1	C12	0	1000060120	-1	-1	2	3	0.000	0.000	0.000	11.179	11.179	
2	proton	11	2212	1	-1			-0.050	-0.055	0.135	0.923	**0.938	M = 0.910
3	B11	2	1000050110	1	-1			0.050	0.055	-0.135	10.256	10.255	



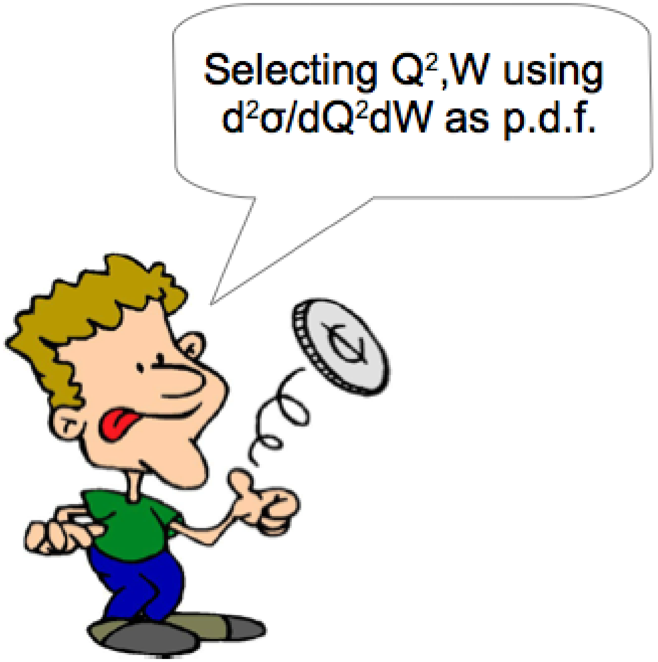
O.Benhar et,al., Phys.Rev.D72:053005,2005.

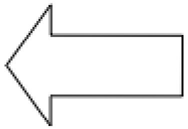


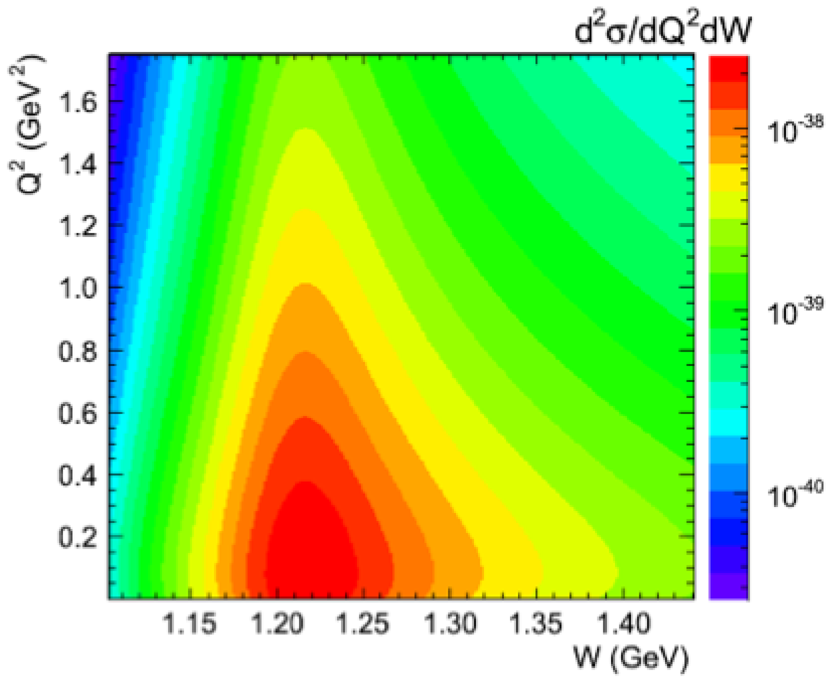
Generating event kinematics

GENIE GHEP Event Record [print level: 3]

Idx	Name	Ist	PDG	Mother		Daughter		Px	Py	Pz	E	m	
0	nu_mu	0	14	-1	-1	4	4	0.000	0.000	1.800	1.800	0.000	
1	C12	0	1000060120	-1	-1	2	3	0.000	0.000	0.000	11.179	11.179	
2	proton	11	2212	1	-1	5	5	-0.050	-0.055	0.135	0.923	**0.938	M = 0.910
3	B11	2	1000050110	1	-1			0.050	0.055	-0.135	10.256	10.255	
4	mu-	1	13	0	-1	-1	-1	0.151	0.140	0.987	1.014	0.106	P = (-0.149,-0.138,-0.979)
5	Delta++	3	2224	2	-1	6	7	-0.201	-0.195	0.947	1.709	**1.231	M = 1.394



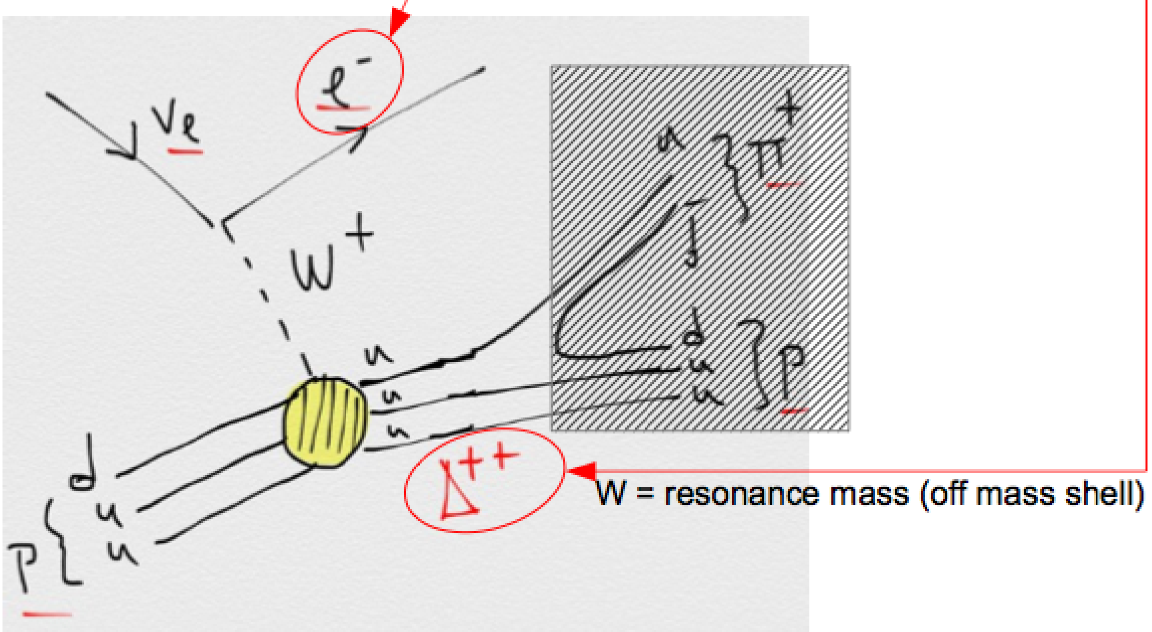




Generating event kinematics

GENIE GHEP Event Record [print level: 3]

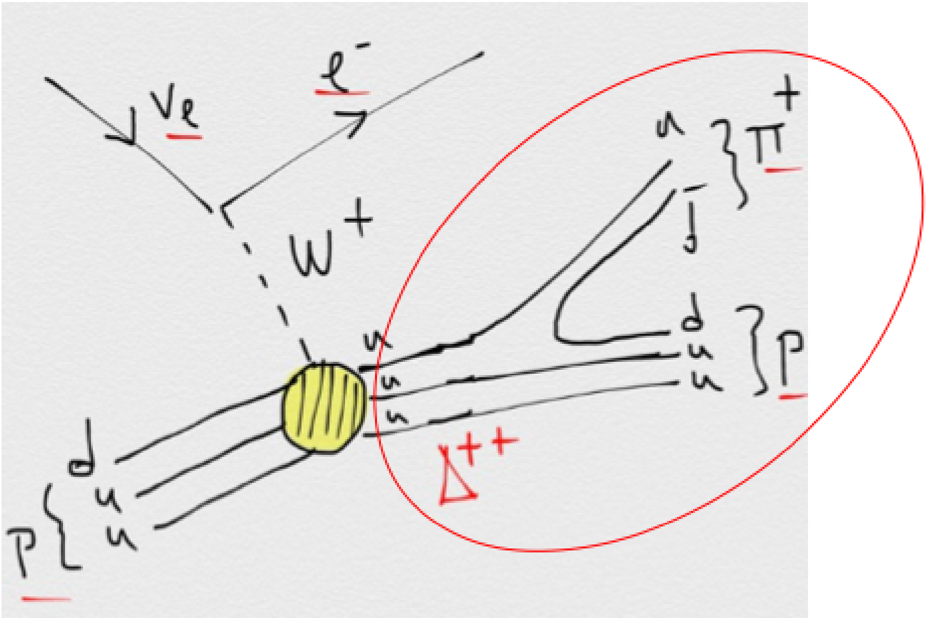
Idx	Name	Ist	PDG	Mother		Daughter		Px	Py	Pz	E	m	
0	nu_mu	0	14	-1	-1	4	4	0.000	0.000	1.800	1.800	0.000	
1	C12	0	1000060120	-1	-1	2	3	0.000	0.000	0.000	11.179	11.179	
2	proton	11	2212	1	-1	5	5	-0.050	-0.055	0.135	0.923	**0.938	M = 0.910
3	B11	2	1000050110	1	-1			0.050	0.055	-0.135	10.256	10.255	
4	mu-	1	13	0	-1	-1	-1	0.151	0.140	0.987	1.014	0.106	P = (-0.149,-0.138,-0.979)
5	Delta++	3	2224	2	-1	6	7	-0.201	-0.195	0.947	1.709	**1.231	M = 1.394



Generating event kinematics

GENIE GHEP Event Record [print level: 3]

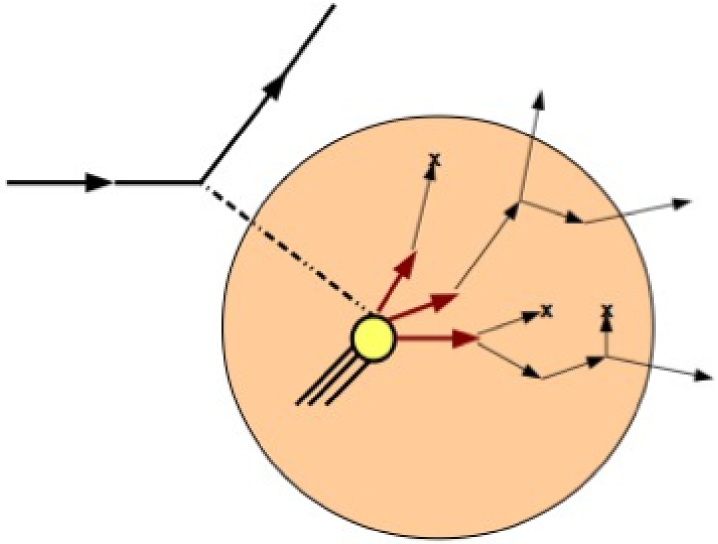
Idx	Name	Ist	PDG	Mother		Daughter		Px	Py	Pz	E	m	
0	nu_mu	0	14	-1	-1	4	4	0.000	0.000	1.800	1.800	0.000	
1	C12	0	1000060120	-1	-1	2	3	0.000	0.000	0.000	11.179	11.179	
2	proton	11	2212	1	-1	5	5	-0.050	-0.055	0.135	0.923	**0.938	M = 0.910
3	B11	2	1000050110	1	-1			0.050	0.055	-0.135	10.256	10.255	
4	mu-	1	13	0	-1	-1	-1	0.151	0.140	0.987	1.014	0.106	P = (-0.149,-0.138,-0.979)
5	Delta++	3	2224	2	-1	6	7	-0.201	-0.195	0.947	1.709	**1.231	M = 1.394
6	proton	14	2212	5	-1			-0.059	-0.010	0.269	0.978	0.938	
7	pi+	14	211	5	-1			-0.142	-0.184	0.679	0.731	0.140	



Generating event kinematics

GENIE GHEP Event Record [print level: 3]

Idx	Name	Ist	PDG	Mother		Daughter		Px	Py	Pz	E	m	
0	nu_mu	0	14	-1	-1	4	4	0.000	0.000	1.800	1.800	0.000	
1	C12	0	1000060120	-1	-1	2	3	0.000	0.000	0.000	11.179	11.179	
2	proton	11	2212	1	-1	5	5	-0.050	-0.055	0.135	0.923	**0.938	M = 0.910
3	B11	2	1000050110	1	-1			0.050	0.055	-0.135	10.256	10.255	
4	mu-	1	13	0	-1	-1	-1	0.151	0.140	0.987	1.014	0.106	P = (-0.149,-0.138,-0.979)
5	Delta++	3	2224	2	-1	6	7	-0.201	-0.195	0.947	1.709	**1.231	M = 1.394
6	proton	14	2212	5	-1	8	9	-0.059	-0.010	0.269	0.978	0.938	FSI = 4
7	pi+	14	211	5	-1	10	11	-0.142	-0.184	0.679	0.731	0.140	FSI = 4
8	proton	1	2212	6	-1	-1	-1	0.030	0.054	0.150	0.952	0.938	
9	proton	1	2212	6	-1	-1	-1	0.095	-0.023	0.173	0.959	0.938	
10	pi+	1	211	7	-1	-1	-1	-0.411	-0.142	0.325	0.561	0.140	
11	neutron	1	2112	7	-1	-1	-1	0.363	0.022	0.428	1.095	0.940	

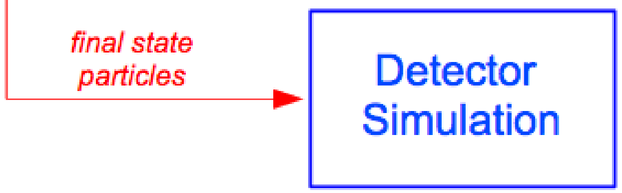


Generating event kinematics

```
|-----------------------------------------------------------------------------------------------------------------------|
|GENIE GHEP Event Record [print level:   3]                                                                              |
|-----------------------------------------------------------------------------------------------------------------------|
| Idx |          Name | Ist |        PDG |   Mother  | Daughter  |      Px |      Py |      Pz |       E |        m  |
|-----------------------------------------------------------------------------------------------------------------------|
|   0 |         nu_mu |   0 |         14 |  -1 |  -1 |   4 |   4 |   0.000 |   0.000 |   1.800 |   1.800 |   0.000 |
|   1 |           C12 |   0 | 1000060120 |  -1 |  -1 |   2 |   3 |   0.000 |   0.000 |   0.000 |  11.179 |  11.179 |
|   2 |        proton |  11 |       2212 |   1 |  -1 |   5 |   5 |  -0.050 |  -0.055 |   0.135 |   0.923 | **0.938 | M = 0.910
|   3 |           B11 |   2 | 1000050110 |   1 |  -1 |  12 |  12 |   0.050 |   0.055 |  -0.135 |  10.256 |  10.255 |
|   4 |           mu- |   1 |         13 |   0 |  -1 |  -1 |  -1 |   0.151 |   0.140 |   0.987 |   1.014 |   0.106 | P = (-0.149,-0.138,-0.979)
|   5 |       Delta++ |   3 |       2224 |   2 |  -1 |   6 |   7 |  -0.201 |  -0.195 |   0.947 |   1.709 | **1.231 | M = 1.394
|   6 |        proton |  14 |       2212 |   5 |  -1 |   8 |   9 |  -0.059 |  -0.010 |   0.269 |   0.978 |   0.938 | FSI = 4
|   7 |           pi+ |  14 |        211 |   5 |  -1 |  10 |  11 |  -0.142 |  -0.184 |   0.679 |   0.731 |   0.140 | FSI = 4
|   8 |        proton |   1 |       2212 |   6 |  -1 |  -1 |  -1 |   0.030 |   0.054 |   0.150 |   0.952 |   0.938 |
|   9 |        proton |   1 |       2212 |   6 |  -1 |  -1 |  -1 |   0.095 |  -0.023 |   0.173 |   0.959 |   0.938 |
|  10 |           pi+ |   1 |        211 |   7 |  -1 |  -1 |  -1 |  -0.411 |  -0.142 |   0.325 |   0.561 |   0.140 |
|  11 |       neutron |   1 |       2112 |   7 |  -1 |  -1 |  -1 |   0.363 |   0.022 |   0.428 |   1.095 |   0.940 |
|  12 |      HadrBlob |  15 | 2000000002 |   3 |  -1 |  -1 |  -1 |  -0.228 |  -0.050 |  -0.264 |   8.398 | **0.000 | M = 8.391
|-----------------------------------------------------------------------------------------------------------------------|
|       Fin-Init:                                                 |  -0.000 |   0.000 |  -0.000 |   0.000 |           |
|-----------------------------------------------------------------------------------------------------------------------|
|       Vertex:         nu_mu @ (x =     0.00000 m, y =     0.00000 m, z =     0.00000 m, t =              0 s) |
|-----------------------------------------------------------------------------------------------------------------------|
| Err flag [bits:15->0] : 0000000000000000     |  1st set:                                                      none |
| Err mask [bits:15->0] : 1111111111111111     |  Is unphysical:    NO |   Accepted:   YES                          |
|-----------------------------------------------------------------------------------------------------------------------|
| sig(Ev) =       4.3934e-38 cm^2  | d2sig(W,Q2;E)/dWdQ2 =      5.5985e-38 cm^2/GeV^3 | Weight =                 1 |
|-----------------------------------------------------------------------------------------------------------------------|
```

Simulation chain downstream the Neutrino Generator

```
|----------------------------------------------------------------------------------------------------------------|
|GENIE GHEP Event Record [print level:   3]                                                                      |
|----------------------------------------------------------------------------------------------------------------|
| Idx |          Name | Ist |        PDG |   Mother  | Daughter  |      Px |      Py |      Pz |       E |       m |
|----------------------------------------------------------------------------------------------------------------|
|   0 |         nu_mu |   0 |         14 |  -1 |  -1 |   4 |   4 |   0.000 |   0.000 |   1.800 |   1.800 |   0.000 |
|   1 |           C12 |   0 | 1000060120 |  -1 |  -1 |   2 |   3 |   0.000 |   0.000 |   0.000 |  11.179 |  11.179 |
|   2 |        proton |  11 |       2212 |   1 |  -1 |   5 |   5 |  -0.050 |  -0.055 |   0.135 |   0.923 | **0.938 | M = 0.910
|   3 |           B11 |   2 | 1000050110 |   1 |  -1 |  12 |  12 |   0.050 |   0.055 |  -0.135 |  10.256 |  10.255 |
|   4 |           mu- |   1 |         13 |   0 |  -1 |  -1 |  -1 |   0.151 |   0.140 |   0.987 |   1.014 |   0.106 | P = (-0.149,-0.138,-0.979)
|   5 |       Delta++ |   3 |       2224 |   2 |  -1 |   6 |   7 |  -0.201 |  -0.195 |   0.947 |   1.709 | **1.231 | M = 1.394
|   6 |        proton |  14 |       2212 |   5 |  -1 |   8 |   9 |  -0.059 |  -0.010 |   0.269 |   0.978 |   0.938 | FSI = 4
|   7 |           pi+ |  14 |        211 |   5 |  -1 |  10 |  11 |  -0.142 |  -0.184 |   0.679 |   0.731 |   0.140 | FSI = 4
|   8 |        proton |   1 |       2212 |   6 |  -1 |  -1 |  -1 |   0.030 |   0.054 |   0.150 |   0.952 |   0.938 |
|   9 |        proton |   1 |       2212 |   6 |  -1 |  -1 |  -1 |   0.095 |  -0.023 |   0.173 |   0.959 |   0.938 |
|  10 |           pi+ |   1 |        211 |   7 |  -1 |  -1 |  -1 |  -0.411 |  -0.142 |   0.325 |   0.561 |   0.140 |
|  11 |       neutron |   1 |       2112 |   7 |  -1 |  -1 |  -1 |   0.363 |   0.022 |   0.428 |   1.095 |   0.940 |
|  12 |      HadrBlob |  15 | 2000000002 |   3 |  -1 |  -1 |  -1 |  -0.228 |  -0.050 |  -0.264 |   8.398 | **0.000 | M = 8.391
|----------------------------------------------------------------------------------------------------------------|
|       Fin-Init:                                                 |  -0.000 |   0.000 |  -0.000 |   0.000 |         |
|----------------------------------------------------------------------------------------------------------------|
|       Vertex:         nu_mu @ (x =      0.00000 m, y =      0.00000 m, z =      0.00000 m, t =             0 s) |
|----------------------------------------------------------------------------------------------------------------|
| Err flag [bits:15->0] : 0000000000000000    |  1st set:                                                  none |
| Err mask [bits:15->0] : 1111111111111111    |  Is unphysical:    NO |   Accepted:   YES                       |
|----------------------------------------------------------------------------------------------------------------|
| sig(Ev) =        4.3934e-38 cm^2  | d2sig(W,Q2;E)/dWdQ2 =      5.5985e-38 cm^2/GeV^3 | Weight =                1 |
|----------------------------------------------------------------------------------------------------------------|
```



Neutrino Generators' definition of final state particles

Whatever leaves the nucleus and could, in principle, be detected.

- Much of what could be detected, will not be detected in practise.
- But the generator, by default, makes no assumption about the capabilities of your detector technology and the sophistication of your analysis methods.

For example, Λ_c^+ and D_s have $c\tau_0$ which is much much larger than the nuclear radius. If C^{12} was as big as the Earth, those particles would decay a light years away. From the generator point of view, these are 'final state particles'. Similarly, most detectors will not track a τ lepton, but some did. The τ lepton is also a 'final state particle' which is not decayed be default.

Many detector technologies and analyses will not pick them some of the produced particles and one is really interested in their decay products.

Make sure who understand which particles get produced by the neutrino generator and define who has the responsibility to decay the unstable ones. Both the neutrino generator and the detector simulation would decay these particles, if asked. A common mistake though is that neither is asked to.

Another example event

Can you guess what is happening here?

```
|-----------------------------------------------------------------------------------------------------------------------|
|GENIE GHEP Event Record [print level:   3]                                                                           |
|-----------------------------------------------------------------------------------------------------------------------|
| Idx |          Name | Ist |        PDG |   Mother  | Daughter  |      Px |      Py |      Pz |       E |        m  |
|-----------------------------------------------------------------------------------------------------------------------|
|   0 |         nu_mu |   0 |         14 |  -1 |  -1 |   4 |   4 |   0.000 |   0.000 |   1.800 |   1.800 |    0.000  |
|   1 |           C12 |   0 | 1000060120 |  -1 |  -1 |   2 |   3 |   0.000 |   0.000 |   0.000 |  11.179 |   11.179  |
|   2 |       neutron |  11 |       2112 |   1 |  -1 |   5 |   5 |  -0.057 |  -0.005 |  -0.158 |   0.934 | **0.940   | M = 0.919
|   3 |           C11 |   2 | 1000060110 |   1 |  -1 |  13 |  13 |   0.057 |   0.005 |   0.158 |  10.245 |   10.243  |
|   4 |           mu- |   1 |         13 |   0 |  -1 |  -1 |  -1 |   0.172 |  -0.463 |   0.896 |   1.028 |    0.106  | P = (-0.168,0.452,-0.876)
|   5 |     N+(1520)  |   3 |       2124 |   2 |  -1 |   6 |   7 |  -0.229 |   0.458 |   0.747 |   1.706 | **1.520   | M = 1.446
|   6 |       Delta++ |   3 |       2224 |   5 |  -1 |   8 |   9 |  -0.092 |   0.501 |   0.623 |   1.471 |    1.231  |
|   7 |           pi- |  14 |       -211 |   5 |  -1 |  10 |  10 |  -0.136 |  -0.044 |   0.124 |   0.235 |    0.140  | FSI = 1
|   8 |        proton |  14 |       2212 |   6 |  -1 |  11 |  11 |  -0.268 |   0.341 |   0.374 |   1.099 |    0.938  | FSI = 1
|   9 |           pi+ |  14 |        211 |   6 |  -1 |  12 |  12 |   0.176 |   0.160 |   0.249 |   0.371 |    0.140  | FSI = 3
|  10 |           pi- |   1 |       -211 |   7 |  -1 |  -1 |  -1 |  -0.136 |  -0.044 |   0.124 |   0.235 |    0.140  |
|  11 |        proton |   1 |       2212 |   8 |  -1 |  -1 |  -1 |  -0.268 |   0.341 |   0.374 |   1.099 |    0.938  |
|  12 |           pi+ |   1 |        211 |   9 |  -1 |  -1 |  -1 |   0.036 |   0.152 |   0.305 |   0.370 |    0.140  |
|  13 |      HadrBlob |  15 | 2000000002 |   3 |  -1 |  -1 |  -1 |   0.197 |   0.013 |   0.101 |  10.246 | **0.000   | M = 10.243
|-----------------------------------------------------------------------------------------------------------------------|
|       Fin-Init:                                                |   0.000 |   0.000 |  -0.000 |   0.000 |             |
|-----------------------------------------------------------------------------------------------------------------------|
|       Vertex:         nu_mu @ (x =     0.00000 m, y =     0.00000 m, z =     0.00000 m, t =             0 s)  |
|-----------------------------------------------------------------------------------------------------------------------|
| Err flag [bits:15->0] : 0000000000000000     |  1st set:                                                  none |
| Err mask [bits:15->0] : 1111111111111111     |  Is unphysical:    NO |    Accepted:   YES                        |
|-----------------------------------------------------------------------------------------------------------------------|
| sig(Ev) =        6.1729e-39 cm^2  | d2sig(W,Q2;E)/dWdQ2 =      1.2875e-38 cm^2/GeV^3 | Weight =                    1 |
|-----------------------------------------------------------------------------------------------------------------------|
```

Another example event

Can you guess what is happening here?

```
|---------------------------------------------------------------------------------------------------------------------|
|GENIE GHEP Event Record [print level:   3]                                                                          |
|---------------------------------------------------------------------------------------------------------------------|
| Idx |          Name | Ist |        PDG |   Mother  | Daughter  |      Px |      Py |      Pz |       E |        m  |
|---------------------------------------------------------------------------------------------------------------------|
|   0 |         nu_mu |   0 |         14 |  -1 |  -1 |   4 |   4 |   0.000 |   0.000 |   1.800 |   1.800 |    0.000 |
|   1 |           C12 |   0 | 1000060120 |  -1 |  -1 |   2 |   3 |   0.000 |   0.000 |   0.000 |  11.179 |   11.179 |
|   2 |        proton |  11 |       2212 |   1 |  -1 |   5 |   5 |   0.032 |   0.031 |   0.035 |   0.924 | **0.938 | M = 0.922
|   3 |           B11 |   2 | 1000050110 |   1 |  -1 |  12 |  12 |  -0.032 |  -0.031 |  -0.035 |  10.255 |   10.255 |
|   4 |           mu- |   1 |         13 |   0 |  -1 |  -1 |  -1 |   0.213 |  -0.242 |   0.412 |   0.534 |    0.106 | P = (-0.407,0.462,-0.788)
|   5 |      HadrSyst |  12 | 2000000001 |   2 |  -1 |   6 |   8 |  -0.181 |   0.273 |   1.423 |   2.190 | **0.000 | M = 1.632
|   6 |        proton |  14 |       2212 |   5 |  -1 |   9 |   9 |  -0.397 |   0.301 |   0.700 |   1.273 |    0.938 | FSI = 3
|   7 |           pi+ |  14 |        211 |   5 |  -1 |  10 |  10 |   0.200 |   0.062 |   0.710 |   0.754 |    0.140 | FSI = 1
|   8 |           pi0 |  14 |        111 |   5 |  -1 |  11 |  11 |   0.017 |  -0.090 |   0.012 |   0.164 |    0.135 | FSI = 1
|   9 |        proton |   1 |       2212 |   6 |  -1 |  -1 |  -1 |  -0.462 |   0.235 |   0.686 |   1.273 |    0.938 |
|  10 |           pi+ |   1 |        211 |   7 |  -1 |  -1 |  -1 |   0.200 |   0.062 |   0.710 |   0.754 |    0.140 |
|  11 |           pi0 |   1 |        111 |   8 |  -1 |  -1 |  -1 |   0.017 |  -0.090 |   0.012 |   0.164 |    0.135 |
|  12 |      HadrBlob |  15 | 2000000002 |   3 |  -1 |  -1 |  -1 |   0.033 |   0.035 |  -0.021 |  10.255 | **0.000 | M = 10.255
|---------------------------------------------------------------------------------------------------------------------|
|       Fin-Init:                                                |   0.000 |  -0.000 |  -0.000 |  -0.000 |           |
|---------------------------------------------------------------------------------------------------------------------|
|       Vertex:         nu_mu @ (x =      0.00000 m, y =      0.00000 m, z =      0.00000 m, t =            0 s) |
|---------------------------------------------------------------------------------------------------------------------|
| Err flag [bits:15->0] : 0000000000000000    |  1st set:                                                     none |
| Err mask [bits:15->0] : 1111111111111111    |  Is unphysical:    NO |   Accepted:   YES                           |
|---------------------------------------------------------------------------------------------------------------------|
| sig(Ev) =        1.2824e-38 cm^2  | d2sig(x,y;E)/dxdy =        5.355e-38 cm^2       | Weight =                  1 |
|---------------------------------------------------------------------------------------------------------------------|
```

Another example event

Can you guess what is happening here?

```
|---------------------------------------------------------------------------------------------------------------------|
|GENIE GHEP Event Record [print level:   3]                                                                           |
|---------------------------------------------------------------------------------------------------------------------|
| Idx |           Name | Ist |         PDG |  Mother  | Daughter |      Px |      Py |      Pz |       E |        m  |
|---------------------------------------------------------------------------------------------------------------------|
|   0 |          nu_mu |   0 |          14 |  -1 |  -1 |   4 |   4 |   0.000 |   0.000 |   1.800 |   1.800 |    0.000 |
|   1 |            C12 |   0 |  1000060120 |  -1 |  -1 |   2 |   3 |   0.000 |   0.000 |   0.000 |  11.179 |   11.179 |
|   2 |         proton |  11 |        2212 |   1 |  -1 |   5 |   5 |   0.096 |  -0.053 |  -0.135 |   0.922 | **0.938 | M = 0.906
|   3 |            B11 |   2 |  1000050110 |   1 |  -1 |  13 |  13 |  -0.096 |   0.053 |   0.135 |  10.257 |   10.255 |
|   4 |            mu- |   1 |          13 |   0 |  -1 |  -1 |  -1 |  -0.210 |   0.073 |   0.213 |   0.326 |    0.106 | P = (0.682,-0.238,-0.691)
|   5 |       HadrSyst |  12 |  2000000001 |   2 |  -1 |   6 |   8 |   0.306 |  -0.126 |   1.452 |   2.397 | **0.000 | M = 1.878
|   6 |         Sigma+ |  14 |        3222 |   5 |  -1 |   9 |   9 |   0.332 |   0.008 |   0.981 |   1.577 |    1.189 |
|   7 |             K+ |  14 |         321 |   5 |  -1 |  10 |  10 |  -0.050 |  -0.033 |   0.317 |   0.589 |    0.494 | FSI = 1
|   8 |            pi0 |  14 |         111 |   5 |  -1 |  11 |  12 |   0.025 |  -0.102 |   0.155 |   0.230 |    0.135 | FSI = 4
|   9 |         Sigma+ |   1 |        3222 |   6 |  -1 |  -1 |  -1 |   0.332 |   0.008 |   0.981 |   1.577 |    1.189 |
|  10 |             K+ |   1 |         321 |   7 |  -1 |  -1 |  -1 |  -0.050 |  -0.033 |   0.317 |   0.589 |    0.494 |
|  11 |            pi0 |   1 |         111 |   8 |  -1 |  -1 |  -1 |   0.114 |   0.079 |   0.005 |   0.193 |    0.135 |
|  12 |        neutron |   1 |        2112 |   8 |  -1 |  -1 |  -1 |  -0.001 |  -0.179 |   0.069 |   0.959 |    0.940 |
|  13 |       HadrBlob |  15 |  2000000002 |   3 |  -1 |  -1 |  -1 |  -0.184 |   0.052 |   0.216 |   9.335 | **0.000 | M = 9.330
|---------------------------------------------------------------------------------------------------------------------|
|       Fin-Init:                                                 |   0.000 |  -0.000 |   0.000 |   0.000 |           |
|---------------------------------------------------------------------------------------------------------------------|
|       Vertex:         nu_mu @ (x =     0.00000 m, y =     0.00000 m, z =     0.00000 m, t =             0 s)   |
|---------------------------------------------------------------------------------------------------------------------|
| Err flag [bits:15->0] : 0000000000000000    |  1st set:                                                      none |
| Err mask [bits:15->0] : 1111111111111111    |  Is unphysical:    NO |    Accepted:    YES                          |
|---------------------------------------------------------------------------------------------------------------------|
| sig(Ev) =        1.2824e-38 cm^2  | d2sig(x,y;E)/dxdy =        8.8411e-38 cm^2       | Weight =                  1 |
|---------------------------------------------------------------------------------------------------------------------|
```

Supplement: MC methods

Monte Carlo methods

For the time being, let's neglect neutrino interaction specific issues and and difficulties we are confronted with in developing the physics models.

Seen from a different perpective, what a Monte Carlo generator does most of the time is:

- Generating random numbers from multi-dimensional p.d.fs.
- Performing numerical integrations

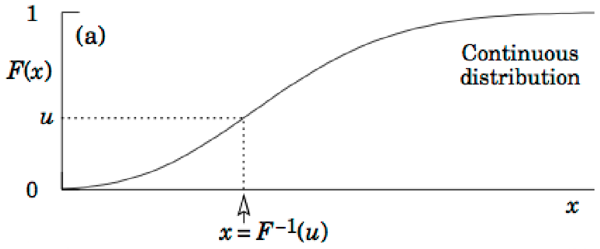
These numerical tasks are a major consideration for the CPU efficiency of the generator.

If you plan to work, and hopefully improve, neutrino generators you need to be conversant enough with the relevant numerical methods.

Generating random numbers from a p.d.f.

- The inverse transform method
- The acceptance-rejection method
- Markov Chain Monte Carlo
  - The Metropolis-Hastings algorithm

The inverse transform sampling method



[image from J. Beringer et al.(PDG), PRD 86, 010001 (2012)]

- **Basic idea:** You want to generate random numbers according to the f(x) p.d.f. The value of the cumulative distribution function

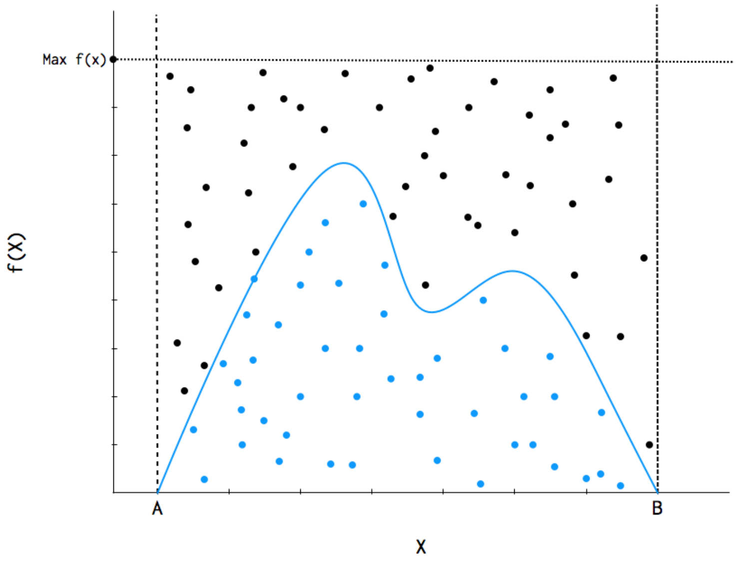
$$F(x) = \int_{-\infty}^{x} dt \cdot f(t)$$

is itself a random variable uniformly distributed in [0,1].

- **Details:**
  - Generate a random number u with uniform distribution in [0,1].
  - Find the value of x for which F(x) = u
  - x is a number drawn from f(x)
- **Caveats:** You need to know (fx) analytically and be able to invert F, i.e. $x = F^{-1}(u)$. So this method is never used (except, perhaps, to implement an importance sampling method - see next)

The acceptance-rejection (von Neumann) method

Assume you can evaluate the p.d.f. f(x) for any x, but you do not know it analytically and you can not invert its cumulative distribution function. Can draw numbers from f(x) as follows:



[image from http://pymc-devs.github.io/pymc/]

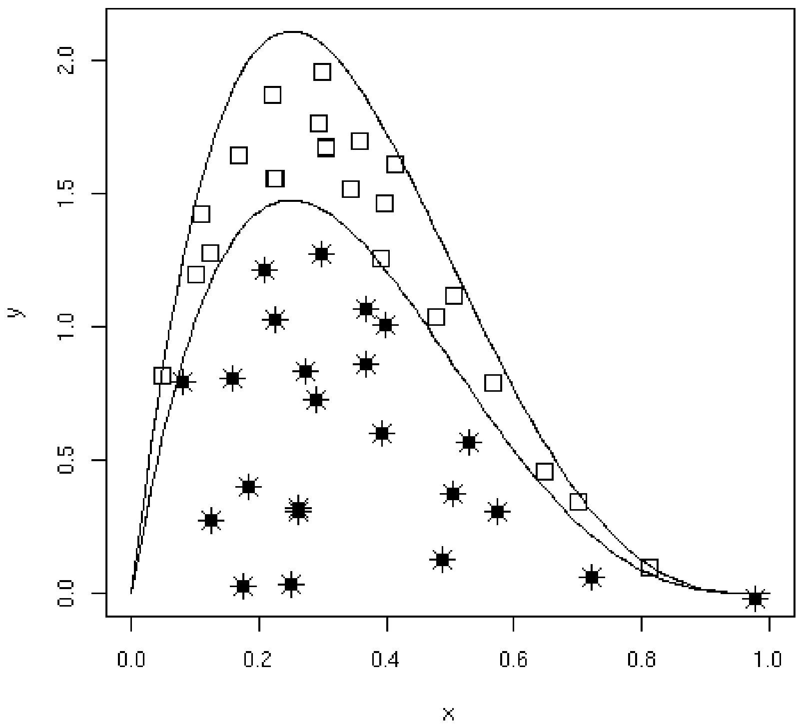
- Evaluate the maximum value of f(x), f_{max}.
  - It doesn't matter if you overshoot, except in terms of CPU efficiency
  - You will have made a mistake if you undershoot the maximum, since you will be clipping the p.d.f. peaks off
- Throw x in its allowed range (uniformly)
- Throw another number u distributed uniformly between 0 and f_{max}.
- Evaluate f(x)
- Is $u < f(x)$? Keep x, otherwise reject and repeat.

Basically, generate random points in the rectangular region (see on the left) and keep the point it is under the f(x) curve. For each throw at x, the probability of accepting the point is $f(x)/f_{max}$. Thus, x is a number drawn from f(x).

The acceptance-rejection method

The acceptance-rejection method is easy to implement (as a test, use it to calculate π). However, the method can be inefficient for functions which are strongly peaked in small regions.

A measure of inefficiency is the integral of f(x) relative to the size of the entire area where random points can be thrown. This figure is the average random number "acceptance" rate.



[image from http://pymc-devs.github.io/pymc/]

A common method to improve the efficiency of the acceptance-rejection method is to employ an importance sampling scheme:

Sample more frequently in areas where the acceptance probability is higher.

Can be implemented as follows:

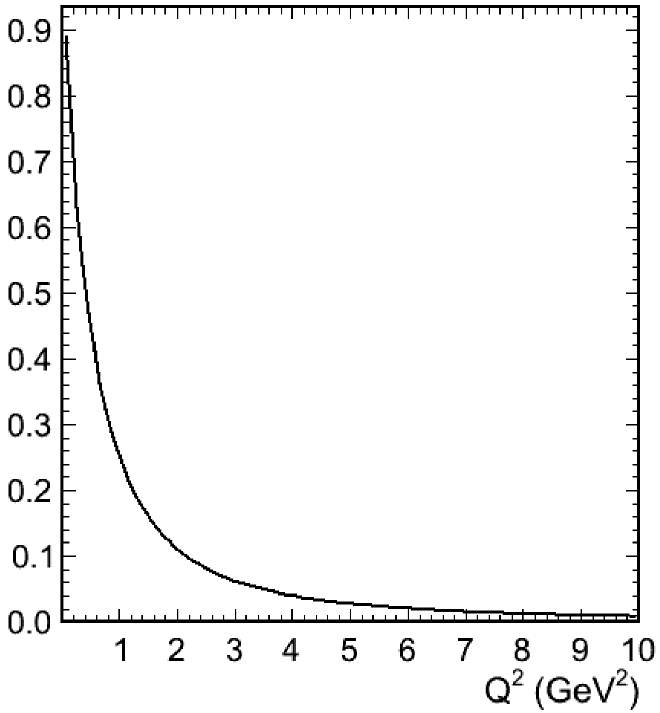
- Find a function g(x) that envelopes f(x) and which can be evaluated much much faster than f(x).
- Draw numbers from g(x).
- Keep those numbers with probability f(x)/g(x).

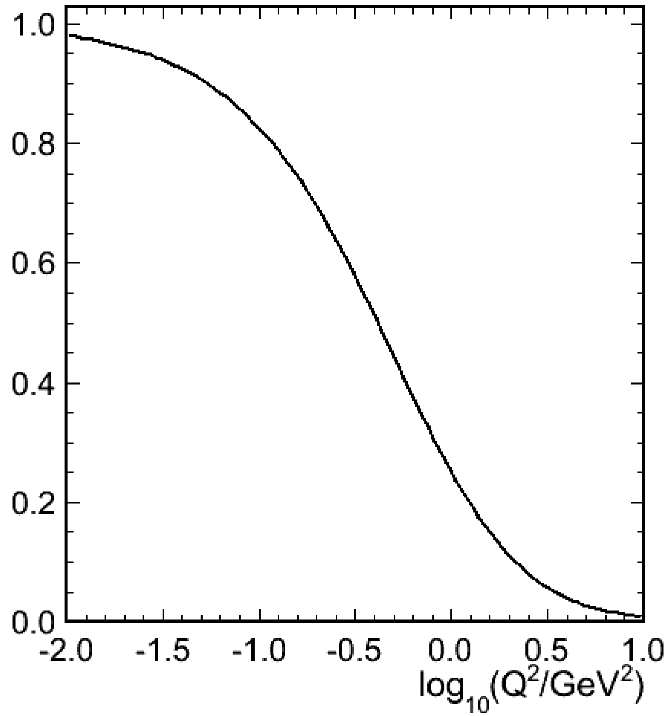
The acceptance-rejection method

Another similar and straightforward trick to improve the efficiency of the acceptance - rejection method, is variable transformation.

If the p.d.f, is strongly peaked in small region of x, can you find a tranformation $x \rightarrow x\prime$ that flattens out the distribution? Then, use the transformed space. Once a $x\prime$ value is accepted, map it back to the corresponding x value (Do not forget the Jacobian).

Example, a dipole form factor in Q^2 and $logQ^2$:





Generating random numbers from n-dimensional p.d.fs

That's all we do. We do not typically have p.d.f's of high dimensionality, so we are fine.

But this is changing, with new microscopic calculations providing cross-section models which are differential over many kinematical variables.
Simulation efficiency issues are one of the main issues preventing us deploying model implementations within the generators.

- But we can not always be the ones who complain.
- We complained that theorists do not provide models which are fully differential over all kinematical variables...
- ...and again when they started doing so, because we can not run these models fast enough.

What to do with these models is a good discussion point.

People in this room have either provided such theory calculations (Luis) or spent significant effort to integrate them in MC (Steve B., Steve D., David H.). Any comments?

Markov Chain Monte Carlo

A method commonly used in our field, though not in our generators, is a Markov Chain Monte Carlo.
Behind the Bayesian hokum (excuse my ingrained Frequentism), there is a generally useful method for the efficient sampling of multi-dimensional p.d.fs.

The multi-dimensional distribution f($\vec{x}$) is **sampled by a Markov chain**, not independent points*.

Although points are locally correlated, over a long run, each point $\vec{x}$ is visited with a frequency in proportion to f($\vec{x}$) (*ergodic* property).
f($\vec{x}$) is sampled ergodically if the probability p($\vec{x_{i+1}}|\vec{x_i}$) for the Markov Chain to transition between points $\vec{x_i}$ and $\vec{x_{i+1}}$ satisfies the detailed balance equation:

$$f(\vec{x_i})p(\vec{x_{i+1}}|\vec{x_i}) = f(\vec{x_{i+1}})p(\vec{x_i}|\vec{x_{i+1}})$$

---

*Local correlation may be an issue or may not be, depending on how the generator is used. Will need to test and design a fool-proof system. Points can be de-correlated at the expense of some CPU efficiency.

Metropolis-Hastings algorithm

The Metropolis algorithm[†] gives a recipe for finding a transition probability $p(\vec{x_{i+1}}|\vec{x_i})$ that satisfies the detailed balance equation:

- Select a proposal function $q(\vec{x_{i+1}}|\vec{x_i})$, anything that, over many steps, could cover all space. For example, a multi-dimensional gaussian centered at $\vec{x_i}$.
- Start at $\vec{x_0}$.
- Generate a candidate point $\vec{x_{1;c}}$.
- Calculate an acceptance probability for the candidate point given by:

$$\alpha = min 1, \frac{f(\vec{x_{1;c}})q(\vec{x_0}|\vec{x_{1;c}})}{f(\vec{x_0})q(\vec{x_{1;c}}|\vec{x_0})}$$

- Throw a random number u in [0,1]. If u$<\alpha$, accept the candidate point ($\vec{x_1} = \vec{x_{1;c}}$). Otherwise, reject the candidate point and stay where you are ($\vec{x_1} = \vec{x_0}$).

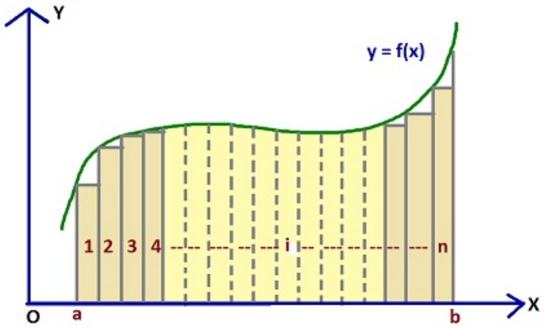
[†] N. Metropolis et al., J. Chem. Phys 21, (1953), 1087

Random number generators

At the heart of everything discussed above, is a **uniform** (pseudo)random number generator:

- **Linear congruential** generator (TRandom in ROOT)
- **RANLUX** generator (TRandom1 in ROOT)
  Luscher, Computer Physics Communications, 79 (1994) 100110.
- **Tausworthe** generator (TRandom2 in ROOT)
  L'Ecuyer, Mathematics of Computation, 65, 213 (1996), 203213.
- **Mersenne twister** generator (TRandom3 in ROOT)
  Matsumoto and Nishimura, Vol. 8, No. 1, January 1998, pp 3–30
  Periodicity of 10^{6000}

Numerical integration methods



If I give you an arbitrary function you are all able to calculate its derivative analytically. However, chances are you can not integrate it analytically. Numerical integration methods were developed from the early stages in the development of calculus.

Numerical integrations are the bread and butter of Monte Carlo generators, where complex multi-dimensional integrations need to be performed and where the integrand is usually not known analytically. Numerous methods are used, both:

- Classical methods (numerical quadrature rules), and
- Monte Carlo methods

Classical quadrature rules

Unsurprisingly, the method consists of choosing a set of abscissas and summing-up the computed integrand values.

The goal is to achieve the smallest error with as few integrand evaluations as possible.

Several methods, open and closed, of various orders, exist, both with

- Equally-spaced abscissas (Newton-Cotes type formulas)
  - Trapezoidal rule
  - Simpson's rule
  - Simpson's 3/8 rule
  - Bode's rule
  - ...
- Non-equally-spaced abscissas
  - Gaussian quadratures

Simple examples of Newton-Cotes formulae

They usually have the the form

$$\int_{x_0}^{x_{N-1}} f(x)dx = \sum_{0}^{N-1} w_i f(x_i) = \sum_{0}^{N-1} w_i f_i$$

where w_i are a set of 'weights' and $f_i \equiv f(x_i)$.

Example
Simpson's rule (2-interval/3-point formula, exact for polynomials up to order 3):

$$\int_{x_0}^{x_2} f(x)dx = h\left(\frac{1}{3}f_0 + \frac{4}{3}f_1 + \frac{1}{3}f_2)\right) + O(h^5 f^{''''})$$

Extended Simpson's rule (using Simpson's rule N-1 times to cover the $[x_0, x_{N-1}]$ range:

$$\int_{x_0}^{x_{N-1}} f(x)dx = h\left(\frac{1}{3}f_0 + \frac{4}{3}f_1 + \frac{2}{3}f_2 + \frac{4}{3}f_3 + ... + \frac{2}{3}f_{N-3} + \frac{4}{3}f_{N-2} + \frac{1}{3}f_{N-1}\right) + O(\frac{1}{N^4})$$

Limitations of classical integration methods

- The curse of dimensionality ($\sim$100 function evaluations for 1-D $\rightarrow$ $\sim$10000 for 2-D, $\rightarrow$ $\sim$1000000 for 3-D $\rightarrow$ $\sim$100000000 for 4-D, ...)
- Complicated integration boundaries
- These methods (a repeated 1-D integration) could still be your best bet if
  - you can reduce the dimensionality analyticaly, or
  - the integrand is smooth and the integration boundary simple
- Generally, complex multi-dimensional integrals call for MC integration methods

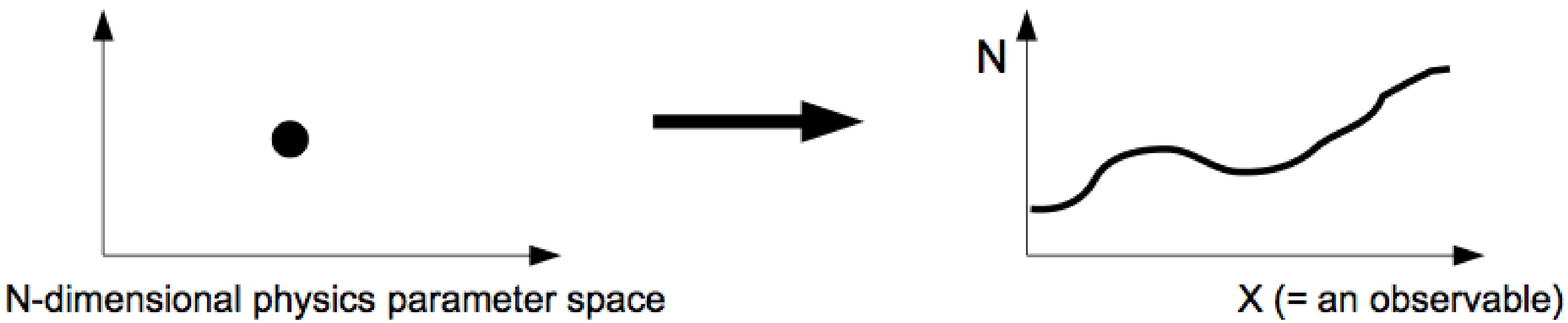
Monte Carlo methods

- Naive MC methods can be trivially implemented and run for multi-dimensional integrals.
- **Accuracy improves as** $1/\sqrt{N}$
  - the good: $1/\sqrt{N}$ does not depend on the dimensionality
  - and the bad: $1/\sqrt{N}$ is not that fast...
- Several improved methods have been developed (and excellent references exist) using
  - stratified sampling (e.g MISER algorithm)
  - importance sampling (e.g VEGAS algorithm)
- The above methods are typically used in generators for the more CPU-intensive integrations

Event Reweighting

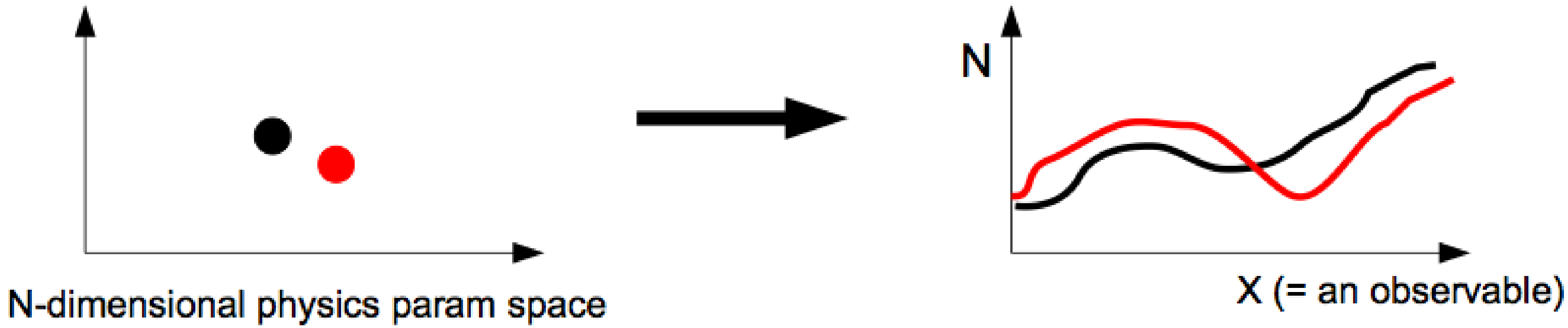
Event Reweighting

What we have discussed so far connects a vast amount of inputs (physics models, assumptions, external data and model parameter tunes) to distributions of observable quantities (e.g. muon momentum - angle distributions for ν_μCC events, pion momentum distributions for ν_μCC$1\pi^+$ events).



Event Reweighting

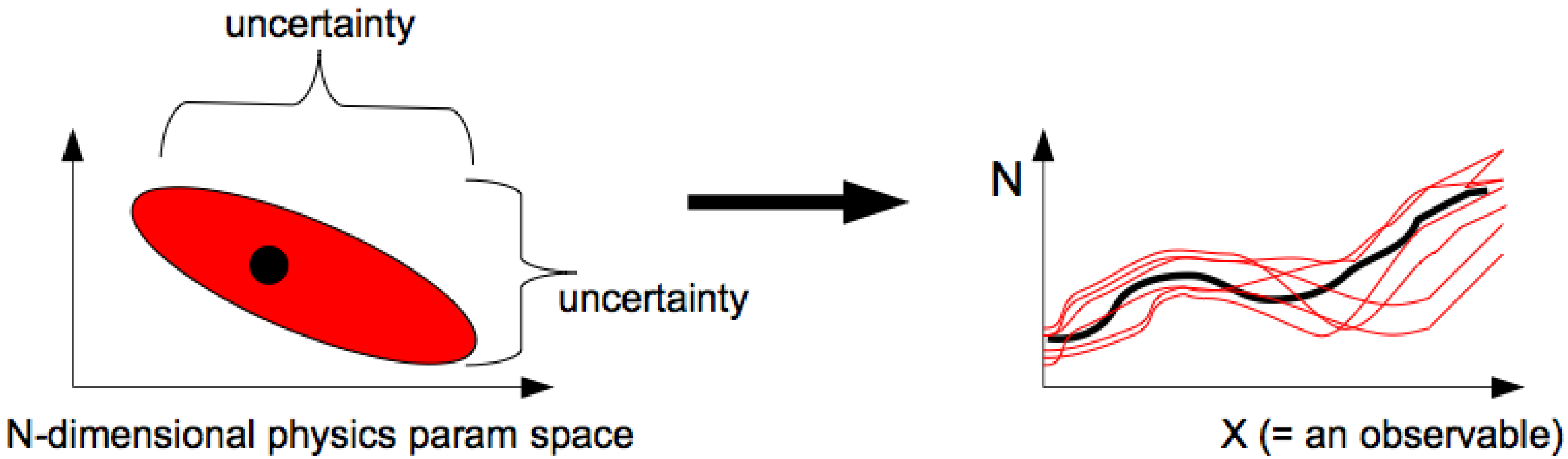
However, our inputs, for a specific choice of models, have uncertainties. And, of course, there are many different model choices. Experiments need to investigate the effect that simulation choices have on physics observables (model uncertainties $\rightarrow$ oscillation systematics).



T2K/ND280: Generating a sample corresponding to 5 $\times$ 10^{21} POT, takes $\sim$ 600 CPU $\times$ weeks.

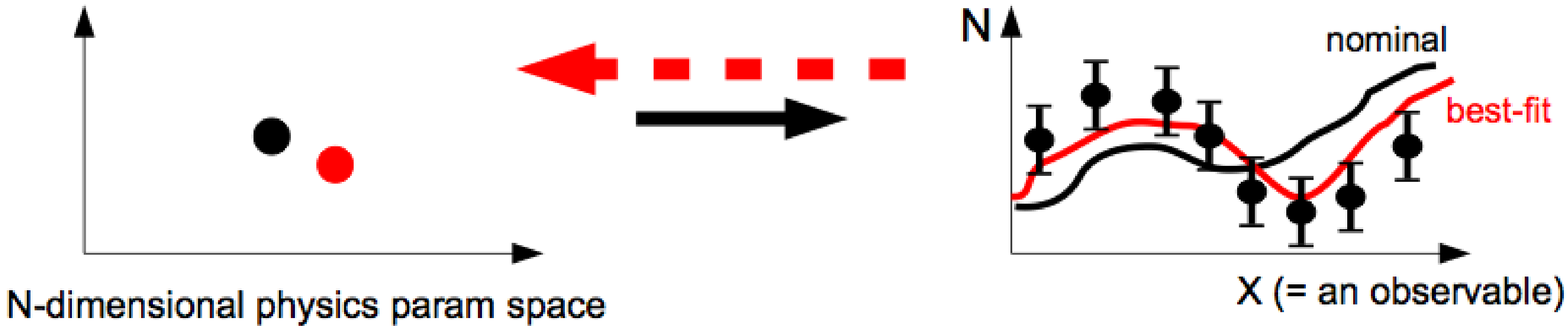
Event Reweighting

We need a faster procedure to "emulate" the simulation output, under different input physics assumptions, without rerunning the full simulation.



Event Reweighting

If this procedure we could inverted, we could also use it to fit observable distributions from our experiments, to improve the level of data/MC agreement, constrain uncertainties and obtain the correlations imposed by our experimental constraint:



Event Reweighting

Neutrino Monte Carlo Generators usually provide a extensive toolkit to allow physics models to be *reweighted*.

We will explain event reweighting in detail later in this school.

Tutorial by C.A on Event Reweighting and Systematics (Tutorial 5).

Summary

I hope that you have started to appreciate the art of Neutrino Monte Carlo generators.

Enjoy the Liverpool NuSTEC Generator School.