Particle Physics Phenomenology
1. Introduction and Monte Carlo techniques

Torbjörn Sjöstrand

Department of Astronomy and Theoretical Physics
Lund University
Sölvegatan 14A, SE-223 62 Lund, Sweden

Lund, 16 January 2018

Course objectives

Improve understanding of how physics at the LHC works

Course objectives

Improve understanding of how physics at the LHC works

LHC is a QCD machine:

- incoming hadrons: parton distributions
- outgoing hadrons: jets
- hard processes: must involve QCD (at least for NLO)
- parton showers, beam remnants, underlying events, ...

QCD needed to understand also "new", "non-QCD" physics!

Course objectives

Improve understanding of how physics at the LHC works

LHC is a QCD machine:

- incoming hadrons: parton distributions
- outgoing hadrons: jets
- hard processes: must involve QCD (at least for NLO)
- parton showers, beam remnants, underlying events, ...

QCD needed to understand also “new”, “non-QCD” physics!

Phenomenology: the bridge between theory and experiment.
This course should complement typical courses

- on the Standard Model and BSM physics, and
- on experimental techniques.

Event generators: how things work “in real life”

Lectures

Course will run across 12 Tuesdays, tentatively.

1. Introduction and Monte Carlo techniques
2. Phase space and matrix elements
3. Evolution equations and final-state showers
4. Parton distributions and initial-state showers
5. Matching of hard matrix elements to showers
6. Multiparton interactions, minbias and underlying events
7. Hadronization: strings and clusters
8. Total and diffractive cross sections
   Event generators and other software; tunes
9. Jet physics: algorithms and properties
10. Heavy-ion physics
11. Physics Beyond the Standard Model
12. Future colliders, various applications and the road ahead

The course

My obligations

- Lecture $\sim$ 3 hours each Tuesday, with some external help
- Supervise $\sim$ 1 hour of homework exercise session
- Examine you at the end of the course
- Provide you with a certificate or send mail to your supervisor

Your obligations

- Be present, be active, ask questions
- Review lectures, read extra literature where provided
- Solve weekly exercises, be prepared to present for all
  - "paper-and-pencil" problem solving
  - computer tasks, mainly with event generators, specifically PYTHIA 8
- Individually solve hand-in exercises at the end of the course
- Take an oral exam after that
- Interact with your supervisor to have the course accepted

Assumed knowledge

Students are expected to be familiar with

- the flavour content and interactions of the Standard Model (quarks and leptons, electroweak and strong forces)
- Feynman diagrams and their relation to the Lagrangian
- the principles of matrix element calculations (but no need for detailed calculations here)
- the running of couplings, giving confinement and asymptotic freedom in QCD
- properties of the basic hadrons, notably in flavour and colour
- the basic ideas of BSM physics, e.g. in supersymmetry
- the LHC and the experiments at it
- programming in C++ and running programs

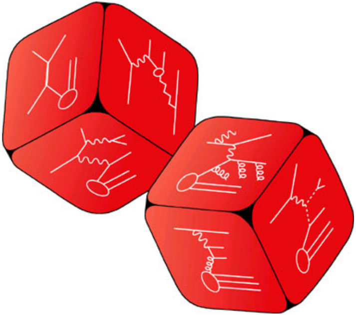
Literature

No relevant textbook, hence lectures with slides = lecture notes.

- R.K. Ellis, W.J. Stirling and B.R. Webber, QCD and Collider Physics (Cambridge University Press, 1996)
  is good but out of date, covers some aspects in too much detail and others not at all.
- J. Campbell, J. Huston and F. Krauss, The Black Book of Quantum Chromodynamics (Oxford University Press, 2018)
  only just arrived, looks like excellent reference for all details (768 pages) but too technical and lengthy for this course.

Some articles/preprints/books cover part of relevent materia, see list linked to course webpage:
`http://home.thep.lu.se/∼torbjorn/ppp2018.html`

A tour to Monte Carlo



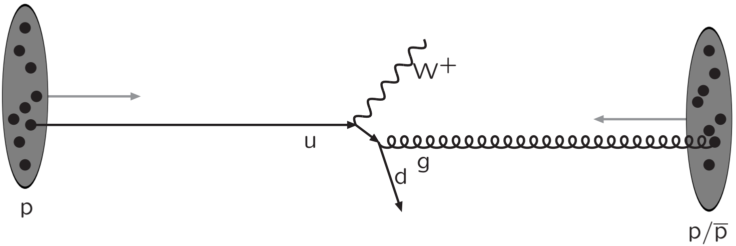
...because Einstein was wrong: God does throw dice!
Quantum mechanics: amplitudes $\Longrightarrow$ probabilities
Anything that possibly can happen, will! (but more or less often)

Event generators: trace evolution of event structure

Warning: schematic only, everything simplified, nothing to scale, . . .

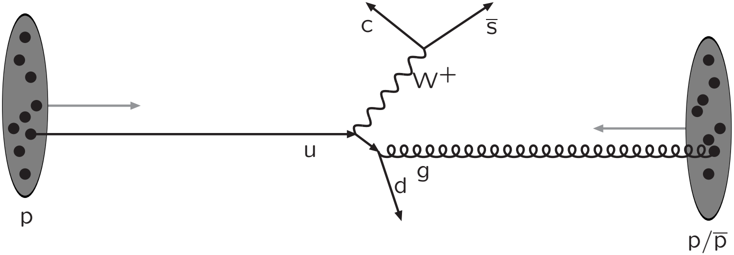
Incoming beams: parton densities

The structure of an event – 2



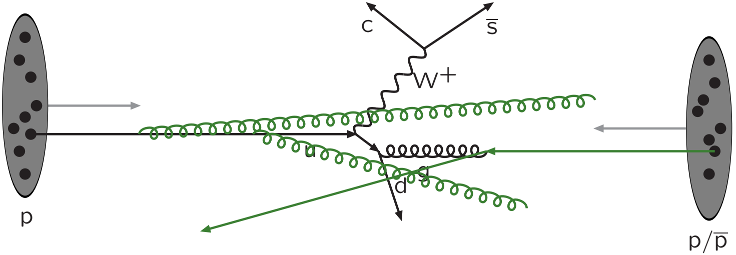
Hard subprocess: described by matrix elements

The structure of an event – 3



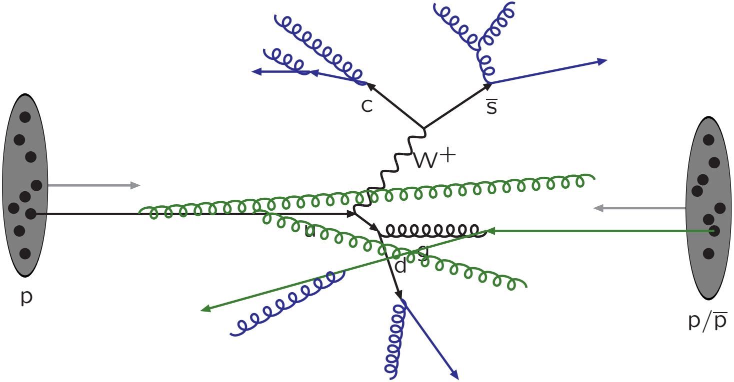
Resonance decays: correlated with hard subprocess

The structure of an event – 4



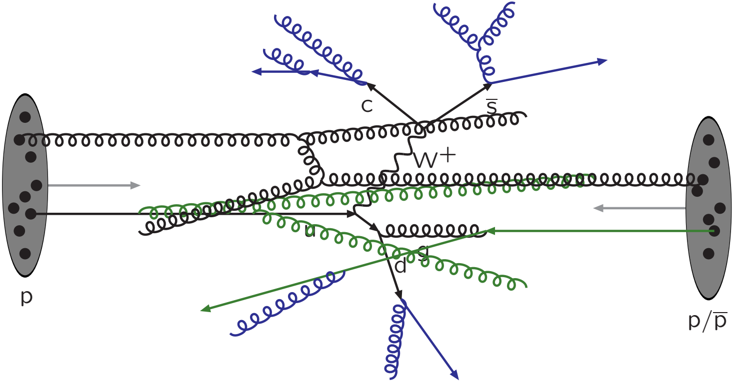
Initial-state radiation: spacelike parton showers

The structure of an event – 5



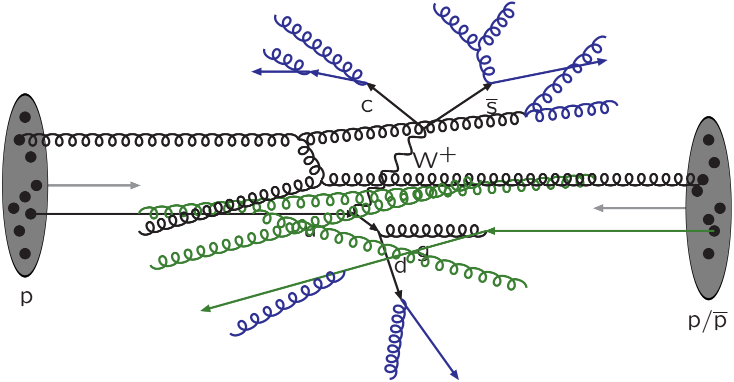
Final-state radiation: timelike parton showers

The structure of an event – 6



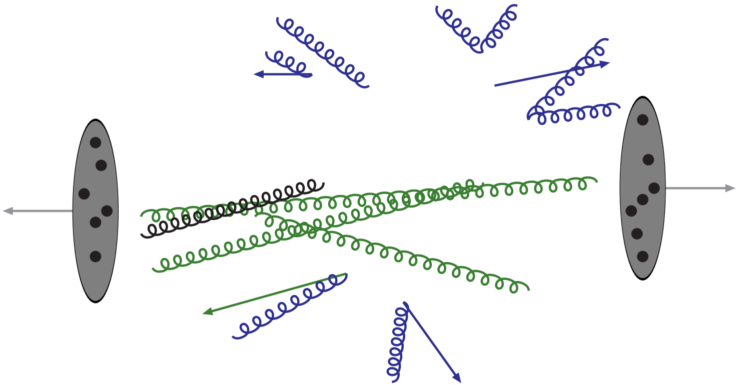
Multiple parton–parton interactions . . .

The structure of an event – 7



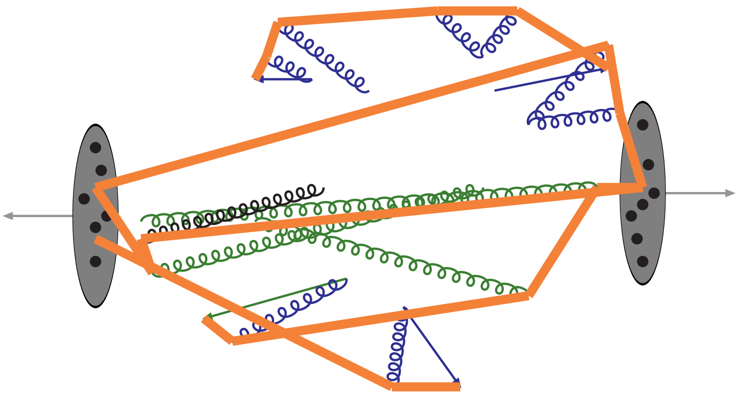
. . . with its initial- and final-state radiation

The structure of an event – 8



Beam remnants and other outgoing partons

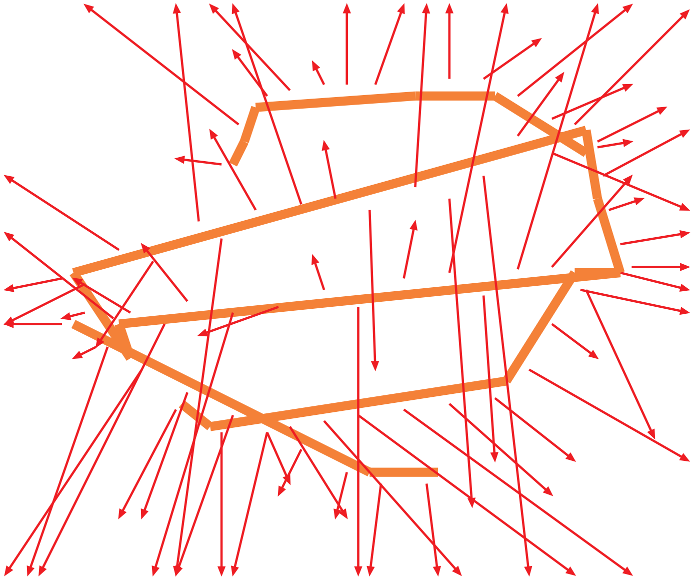
The structure of an event – 9



Everything is connected by colour confinement strings

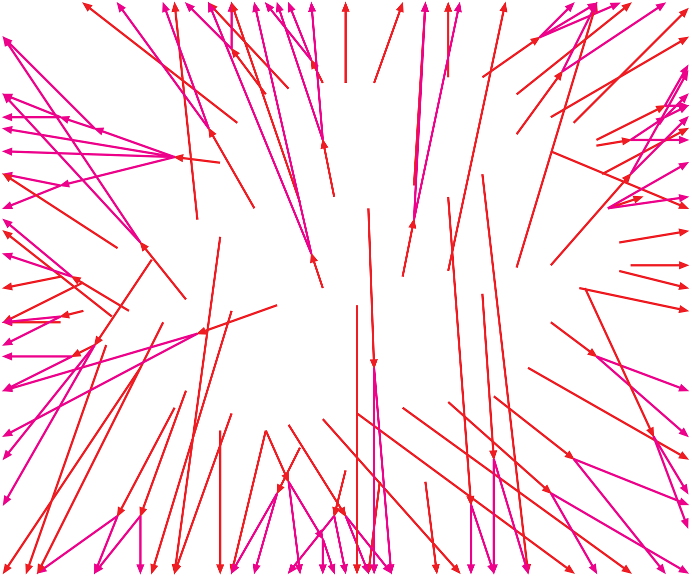
Recall! Not to scale: strings are of hadronic widths

The structure of an event – 10



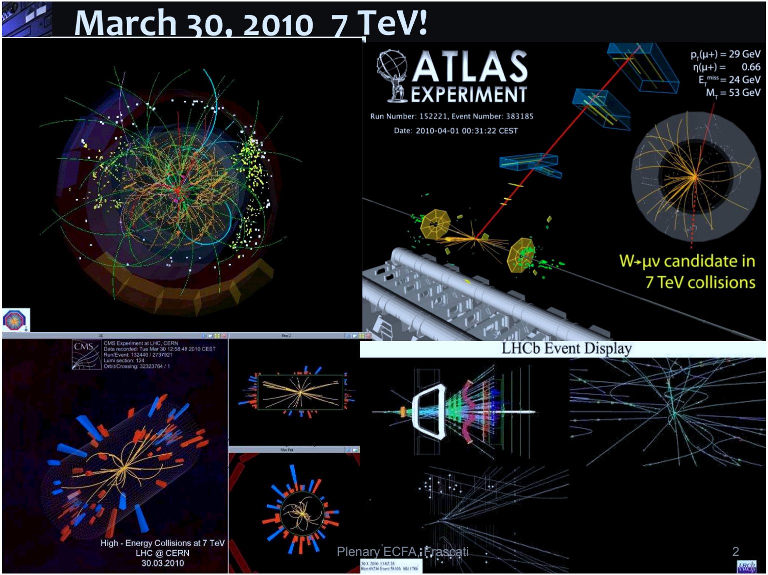
The strings fragment to produce primary hadrons

The structure of an event – 11



Many hadrons are unstable and decay further

The structure of an event – 12



These are the particles that hit the detector

The complexity of events

Theorists' folklore:
higher energies gives smaller strong coupling $\alpha_s(Q^2)$
$\Rightarrow$ less influence of nonperturbative physics $\Rightarrow$ *physics is simpler*

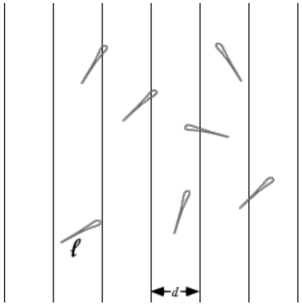
Experimental reality:
high-energy processes "cascade down" and give low-energy debris;
more cascading $\Rightarrow$ more complex and varied final states.

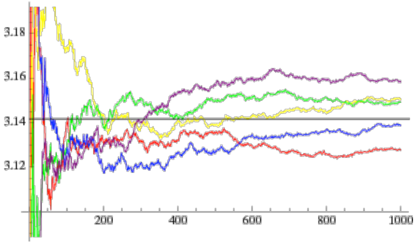
Ways back to simplicity:

- inclusive quantities, such as jets and missing $E_\perp$
- rare but clean final states, such as leptons and photons

Only partly successful:

- isolation criteria (worsened by pileup)
- detector imperfections





Buffon's needle (proposed 1733): probability for needle to cross line is related to π

. . . but gambling & odds are older

Spatial vs. temporal problems

“Spatial” problems: no memory/ordering

1. Integrate a function
2. Pick a point at random according to a probability distribution

“Temporal” problems: has memory

1. Radioactive decay: probability for a radioactive nucleus to decay at time t, given that it was created at time 0

In reality combined into multidimensional problems:

1. Random walk (variable step length and direction)
2. Charged particle propagation through matter (stepwise loss of energy by a set of processes)
3. **Parton showers** (cascade of successive branchings)
4. Multiparticle interactions (ordered multiple subcollisions)

Random numbers

For now assume algorithm that returns "random numbers" R, uniformly distributed in range $0 < R < 1$ and uncorrelated.

Step 1: find and learn how to use a random number generator on your platform of preference.

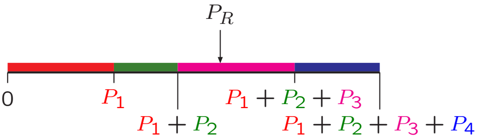
Fallback: an implementation of the Marsaglia–Zaman–Tsang algorithm is available in PYTHIA.
It allows for $\sim 900\,000\,000$ different sequences.

```
Pythia pythia;
pythia.readString("Random:setSeed = on");
pythia.readString("Random:seed = 32133");
....
pythia.init();
double rr = pythia.rndm.flat();
```

or standalone on
`http://home.thep.lu.se/~torbjorn/compute2012/rndm.cc`

Pick among discrete possibilities

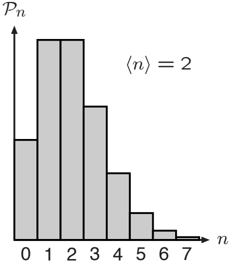
Assume n possible outcomes with (unnormalized) probabilities P_i, $1 \leq i \leq n$. Pick one of them according to



1. $i = 0$
   $P_R = R \sum_{i=1}^{n} P_i$
2. $i = i + 1$
   $P_R = P_R - P_i$
3. if $P_R > 0$ cycle to 2

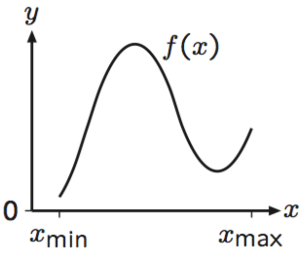
Example 1:
Poissonian $P_i = (\langle n \rangle^i / i!) \, e^{-\langle n \rangle}$, $i \geq 0$.
Note that $P_i = (\langle n \rangle / i) \, P_{i-1}$:

1. $i = -1; \quad P_R = R; \quad P_{\mathrm{now}} = e^{-\langle n \rangle}$
2. $i = i + 1; \quad$ if $(i > 0)$ $P_{\mathrm{now}} = P_{\mathrm{now}} \, (\langle n \rangle / i)$;
   $P_R = P_R - P_{\mathrm{now}}$
3. if $P_R > 0$ cycle to 2



Integration and selection

Assume function $f(x)$,
studied range $x_{\min} < x < x_{\max}$,
where $f(x) \geq 0$ everywhere



Two connected standard tasks:

1. Calculate (approximatively)

$$\int_{x_{\min}}^{x_{\max}} f(x')\,\mathrm{d}x'$$

2. Select x at random according to $f(x)$

In step 2 $f(x)$ is viewed as "probability distribution"
with implicit normalization to unit area,
and then step 1 provides overall correct normalization.

Integral as an area/volume

Theorem

An n-dimensional integration $\equiv$ an $n+1$-dimensional volume

$$\int f(x_1,\ldots,x_n)\,\mathrm{d}x_1\ldots\mathrm{d}x_n \equiv \int\int_0^{f(x_1,\ldots,x_n)} 1\,\mathrm{d}x_1\ldots\mathrm{d}x_n\,\mathrm{d}x_{n+1}$$

since $\int_0^{f(x)} 1\,\mathrm{d}y = f(x)$.

Integral as an area/volume

Theorem

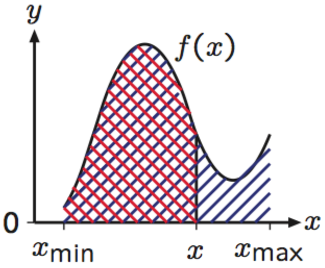
An n-dimensional integration ≡ an n + 1-dimensional volume

$$\int f(x_1,\ldots,x_n)\,\mathrm{d}x_1\ldots\mathrm{d}x_n \equiv \int\int_0^{f(x_1,\ldots,x_n)} 1\,\mathrm{d}x_1\ldots\mathrm{d}x_n\,\mathrm{d}x_{n+1}$$

since $\int_0^{f(x)} 1\,\mathrm{d}y = f(x)$.
So, for 1 + 1 dimension, selection of x according to $f(x)$ is equivalent to uniform selection of (x, y) in the area $x_{\mathrm{min}} < x < x_{\mathrm{max}}$, $0 < y < f(x)$.
Therefore

$$\int_{x_{\mathrm{min}}}^{x} f(x')\,\mathrm{d}x' = R\int_{x_{\mathrm{min}}}^{x_{\mathrm{max}}} f(x')\,\mathrm{d}x'$$

(area to left of selected x is uniformly distributed fraction of whole area)



Basic method 1: analytical solution

If **know primitive function** $F(x)$ and **know inverse** $F^{-1}(y)$ then

$$\begin{aligned} F(x) - F(x_{\min}) &= R\,(F(x_{\max}) - F(x_{\min})) = R\,A_{\mathrm{tot}} \\ \Longrightarrow x &= F^{-1}(F(x_{\min}) + R\,A_{\mathrm{tot}}) \end{aligned}$$

Proof: introduce $z = F(x_{\min}) + R\,A_{\mathrm{tot}}$. Then

$$\frac{\mathrm{d}\mathcal{P}}{\mathrm{d}x} = \frac{\mathrm{d}\mathcal{P}}{\mathrm{d}R}\,\frac{\mathrm{d}R}{\mathrm{d}x} = 1\,\frac{1}{\frac{\mathrm{d}x}{\mathrm{d}R}} = \frac{1}{\frac{\mathrm{d}x}{\mathrm{d}z}\;\frac{\mathrm{d}z}{\mathrm{d}R}} = \frac{1}{\frac{\mathrm{d}F^{-1}(z)}{\mathrm{d}z}\;\frac{\mathrm{d}z}{\mathrm{d}R}} = \frac{\frac{\mathrm{d}F(x)}{\mathrm{d}x}}{\frac{\mathrm{d}z}{\mathrm{d}R}} = \frac{f(x)}{A_{\mathrm{tot}}}$$

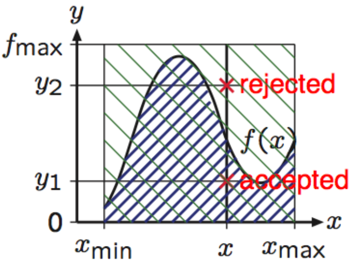
Example 2:
$f(x) = 2x,\ 0 < x < 1, \Longrightarrow F(x) = x^2$
$F(x) - F(0) = R\,(F(1) - F(0)) \Longrightarrow x^2 = R \Longrightarrow x = \sqrt{R}$

Example 3:
$f(x) = e^{-x},\ x > 0,\ F(x) = 1 - e^{-x}$
$1 - e^{-x} = R \Longrightarrow e^{-x} = 1 - R = R \Longrightarrow x = -\ln R$

Basic method 2: hit-and-miss

If $f(x) \leq f_{\max}$ in $x_{\min} < x < x_{\max}$
use interpretation as an area

1. select
$x = x_{\min} + R\,(x_{\max} - x_{\min})$
2. select $y = R\, f_{\max}$ (new R!)
3. while $y > f(x)$ cycle to 1



Integral as by-product:

$$I = \int_{x_{\min}}^{x_{\max}} f(x)\,\mathrm{d}x = f_{\max}\,(x_{\max} - x_{\min})\,\frac{N_{\mathrm{acc}}}{N_{\mathrm{try}}} = A_{\mathrm{tot}}\,\frac{N_{\mathrm{acc}}}{N_{\mathrm{try}}}$$

Binomial distribution with $p = N_{\mathrm{acc}}/N_{\mathrm{try}}$ and $q = N_{\mathrm{fail}}/N_{\mathrm{try}}$,
so error

$$\frac{\delta I}{I} = \frac{A_{\mathrm{tot}}\,\sqrt{p\,q/N_{\mathrm{try}}}}{A_{\mathrm{tot}}\,p} = \sqrt{\frac{q}{p\,N_{\mathrm{try}}}} = \sqrt{\frac{q}{N_{\mathrm{acc}}}} < \frac{1}{\sqrt{N_{\mathrm{acc}}}}$$

Hit-and-miss (2)

Example 4:

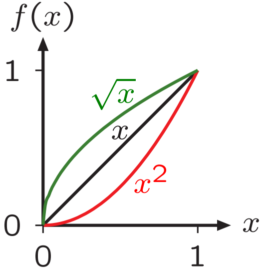
$f(x) = x^{\alpha}$, $\alpha > 0$,

$0 \leq x \leq 1 \Rightarrow 0 \leq f(x) \leq 1$

$F(x) = x^{\alpha+1}/(\alpha + 1)$

$p = I = \int_0^1 f(x)\,\mathrm{d}x = 1/(\alpha + 1)$

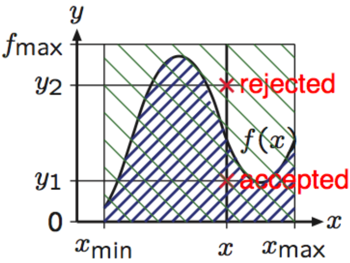
$q = 1 - I = \alpha/(\alpha + 1)$



α	I	$\sqrt{N_{\mathrm{try}}}\delta I$	$\sqrt{N_{\mathrm{try}}}(\delta I/I)$	$\sqrt{N_{\mathrm{acc}}}(\delta I/I)$
1	0.5	0.5	1	0.707
2	0.333	0.471	1.413	0.816
1/2	0.667	0.471	0.706	0.577
9	0.1	0.3	3.0	0.949
1/9	0.9	0.3	0.333	0.316

Crude Monte Carlo integration

Hit-and-miss not most efficient integration: for each x picked, and $f(x)$ evaluated, only accept/reject statistics is used.



Better use full $f(x)$ information:

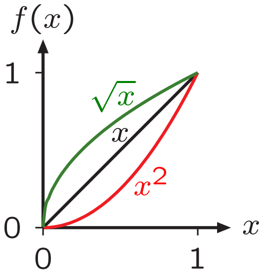
$$I = \int_{x_{\mathrm{min}}}^{x_{\mathrm{max}}} f(x)\,\mathrm{d}x = (x_{\mathrm{max}} - x_{\mathrm{min}})\,\frac{1}{N_{\mathrm{try}}}\sum_{i=1}^{N_{\mathrm{try}}} f(x_i) = \Delta x\,\langle f(x)\rangle$$

$$\delta I = \frac{1}{\sqrt{N_{\mathrm{try}}}}\,\Delta x\,\sqrt{\langle f^2(x)\rangle - \langle f(x)\rangle^2}$$

with $$\langle f^2(x)\rangle = \frac{1}{N_{\mathrm{try}}}\sum_{i=1}^{N_{\mathrm{try}}} f^2(x_i)$$

Example 4 (continued):

$f(x) = x^{\alpha}$, $\alpha > 0$, $0 \leq x \leq 1$



$$
\begin{aligned}
\langle f(x) \rangle &= \int_0^1 x^{\alpha}\,\mathrm{d}x = \frac{1}{\alpha+1} \\
\langle f^2(x) \rangle &= \int_0^1 x^{2\alpha}\,\mathrm{d}x = \frac{1}{2\alpha+1} \\
\delta f &= \sqrt{\langle f^2(x) \rangle - \langle f(x) \rangle^2} = \frac{\alpha}{(\alpha+1)\sqrt{2\alpha+1}}
\end{aligned}
$$

α	I	$\sqrt{N_{\mathrm{try}}}\delta I_{\mathrm{HaM}}$	$\sqrt{N_{\mathrm{try}}}\delta I_{\mathrm{CMCi}}$
1	0.5	0.5	0.289
2	0.333	0.471	0.298
1/2	0.667	0.471	0.236
9	0.1	0.3	0.206
1/9	0.9	0.3	0.090

Conventional integration

Plethora of "deterministic" integration formulae:
Simpson, Newton-Cotes, Gauss, Richardson, Romberg, ...

For $I = \int_0^1 f(x)\,\mathrm{d}x$

$$\text{Trapezoid} \qquad I = \frac{1}{2} f(0) + \frac{1}{2} f(1) + \mathcal{O}(f'')$$

$$\text{Simpson} \qquad I = \frac{1}{6} f(0) + \frac{4}{6} f\left(\frac{1}{2}\right) + \frac{1}{6} f(1) + \mathcal{O}(f^{(4)})$$

extended by split into subranges.

For comparable number of points N error then scales like

Monte Carlo	$1/\sqrt{N}$
Trapezoid	$1/N^2$
Simpson	$1/N^4$

Monte Carlo will not win for 1-dimensional integration.

Conventional integration (2)

Game changes for d dimensions:

Monte Carlo	$1/\sqrt{N}$
Trapezoid	$1/N^{2/d}$
Simpson	$1/N^{4/d}$

Also: 20 dimensions, Simpson $\Rightarrow 3^{20} \approx 3 \cdot 10^9$ points,
all except one on border.

Generally, advantages of simple Monte Carlo integration include

- proportionately faster convergence in many dimensions
- discontinuous functions no problem
- arbitrarily complex integration regions
- few points needed to get first estimate
- easy error estimate
- by-product of Monte Carlo selection of x

Detour: stratified sampling

Split integration range into subranges (adjoint, non-overlapping).
Assume n subranges, $1 \leq i \leq n$, $\Delta x_i = x_{i,\max} - x_{i,\min}$,
and N_i points in respective subrange:

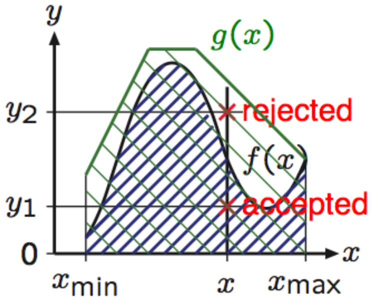
$$
\begin{aligned}
I &= \sum_{i=1}^{n} I_i = \sum_{i=1}^{n} \Delta x_i \, \langle f(x) \rangle_i \\
(\delta I)^2 &= \sum_{i=1}^{n} \frac{(\Delta x_i \, \delta f_i)^2}{N_i} = \sum_{i=1}^{n} \frac{(\Delta x_i)^2}{N_i} \left(\langle f^2(x) \rangle_i - \langle f(x) \rangle_i^2 \right)
\end{aligned}
$$

Uniform stratification: all Δx_i and N_i the same, does reduce δI.
Ultimately $N_i = 1$, n large, $\approx$ conventional integration.
Variance reduction: pick smaller ranges wherever $f'(x)$ is large, rather than $f(x)$ itself.
Wrong way to go for selection according to a distribution!
From now on only study techniques that allow (unbiased) selection.

Importance sampling

Improved version of hit-and-miss:
If $f(x) \leq g(x)$ in
$x_{\min} < x < x_{\max}$
and $G(x) = \int g(x')\,\mathrm{d}x'$ is simple
and $G^{-1}(y)$ is simple

1. select x according to $g(x)$ distribution
2. select $y = R\,g(x)$ (new R!)
3. while $y > f(x)$ cycle to 1



Example 5:
$f(x) = x\,e^{-x}$, $x > 0$
Attempt 1: $F(x) = 1 - (1 + x)\,e^{-x}$ not invertible
Attempt 2: $f(x) \leq f(1) = e^{-1}$ but $0 < x < \infty$

Attempt 3: $g(x) = N\, e^{-x/2}$

$$\frac{f(x)}{g(x)} = \frac{x\, e^{-x}}{N\, e^{-x/2}} = \frac{x\, e^{-x/2}}{N} \leq 1$$

for rejection to work, so find maximum:

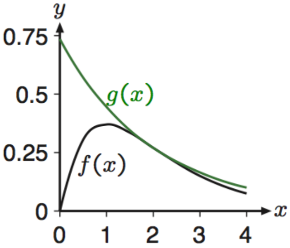
$$\frac{\mathrm{d}}{\mathrm{d}x}\left(\frac{f(x)}{g(x)}\right) = \frac{1}{N}\left(1 - \frac{x}{2}\right) e^{-x/2} = 0 \Longrightarrow x = 2$$

Normalize so $g(2) = f(2) \Rightarrow N = 2/e$

$G(x) \propto 1 - e^{-x/2} = R \Rightarrow x = -2 \ln R$

1. select $x = -2 \ln R$
2. select $y = R\, g(x) = R\, 2e^{-(1+x/2)}$
3. while $y > f(x) = x\, e^{-x}$ cycle to 1

$$\text{efficiency} = \frac{\int_0^\infty f(x)\,\mathrm{d}x}{\int_0^\infty g(x)\,\mathrm{d}x} = \frac{e}{4}$$



Variable transformation

Importance sampling can be reinterpreted as variable transformation

$$\int f(x)\,\mathrm{d}x = \int \frac{f(x)}{g(x)}\,g(x)\mathrm{d}x = \int \frac{f(x)}{g(x)}\,\mathrm{d}G(x)$$

- map to finite x range
- map away singular/peaked regions

Example 6:
$f(x) = \exp(-x^2)$, $1 \leq x < \infty$ (or $\exp(-x^\alpha)$ with $\alpha > 1$)
def. $t = \exp(-x) \Rightarrow x = -\ln t$, $0 \leq t \leq 1/e$

$$\int_1^\infty e^{-x^2}\,\mathrm{d}x = \int_1^\infty \frac{e^{-x^2}}{e^{-x}}\,e^{-x}\mathrm{d}x = \int_0^{1/e} \frac{e^{-\ln^2 t}}{t}\,\mathrm{d}t = \int_0^{1/e} t^{-1-\ln t}\,\mathrm{d}t$$

Pick t uniformly in $0 < t \leq 1/e$, repeatedly until $t^{-1-\ln t} > R$, and then obtain $x = -\ln t$

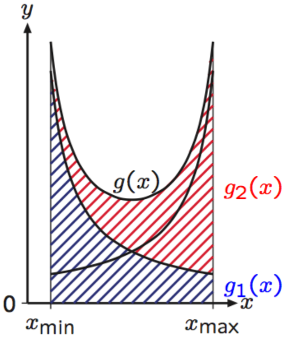
Multichannel

If $f(x) \leq g(x) = \sum_i g_i(x)$,
where all g_i "nice" ($G_i(x)$ invertible)
but $g(x)$ not

1. select i with relative probability

$$A_i = \int_{x_{\min}}^{x_{\max}} g_i(x') \, \mathrm{d}x'$$

2. select x according to $g_i(x)$
3. select $y = R \, g(x) = R \sum_i g_i(x)$
4. while $y > f(x)$ cycle to 1



Works since

$$\int f(x) \, \mathrm{d}x = \int \frac{f(x)}{g(x)} \sum_i g_i(x) \, \mathrm{d}x = \sum_i A_i \int \frac{g_i(x) \, \mathrm{d}x}{A_i} \frac{f(x)}{g(x)}$$

Multichannel (2)

Example 7:

$$f(x) = \frac{1}{\sqrt{x(1-x)}} \,, \quad 0 < x < 1$$

$$g(x) = \frac{1}{\sqrt{x}} + \frac{1}{\sqrt{1-x}} = \frac{\sqrt{x} + \sqrt{1-x}}{\sqrt{x(1-x)}} \,, \quad \frac{1}{\sqrt{2}} \leq \frac{f(x)}{g(x)} \leq 1$$

1 if $R < 1/2$ then $g_1(x)$ else $g_2(x)$

2 g_1: $G_1(x) = 2\sqrt{x} = 2R \Longrightarrow x = R^2$

g_2: $G_2(x) = 2(1 - \sqrt{1-x}) = 2R \Longrightarrow x = 1 - R^2$

3, 4 as previous page

Multichannel – alternative approach

Recall previous formula

$$\int f(x)\,\mathrm{d}x = \int \frac{f(x)}{g(x)} \sum_i g_i(x)\,\mathrm{d}x = \sum_i A_i \int \frac{g_i(x)\,\mathrm{d}x}{A_i} \, \frac{f(x)}{g(x)}$$

Now assume split $f(x) = \sum_i f_i(x)$, with $f_i(x) < g_i(x)$.

(Brute-force, always possible with $f_i(x) = (g_i(x)/g(x))\, f(x)$.)

Then

$$\int f(x)\,\mathrm{d}x = \int \sum_i \frac{f_i(x)}{g_i(x)} \, g_i(x)\,\mathrm{d}x = \sum_i A_i \int \frac{g_i(x)\,\mathrm{d}x}{A_i} \, \frac{f_i(x)}{g_i(x)}$$

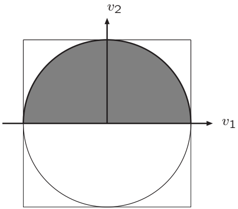
1. select i with relative probability A_i
2. select x according to $g_i(x)$
3. select $y = R\, g_i(x)$
4. while $y > f_i(x)$ cycle to 1

Sometimes special tricks exist.

Example 1:
Pick a random angle on unit sphere, to use its sine and cosine.

Trivial: $\varphi = 2\pi R$, $c = \cos\varphi$, $s = \sin\varphi$
Alternative, if slow trigonometric functions:

1. $v_1 = 2R - 1$, $v_2 = R$
2. if $r^2 = v_1^2 + v_2^2 > 1$ cycle to 1
3. $s = 2v_1v_2/r^2$, $c = (v_1^2 - v_2^2)/r^2$



If instead $v_2 = 2R - 1$, $s = v_2/\sqrt{r^2}$, $c = v_1/\sqrt{r^2}$,
but then need (expensive) square root operation.
Trick: $v_2 > 0 \Rightarrow 0 \leq \varphi \leq \pi$. Formula for double angle then gives
$\sin(2\varphi) = 2\sin\varphi\cos\varphi$, $\cos(2\varphi) = \cos^2\varphi - \sin^2\varphi$.
Cut out central hole to avoid numerical precision problems.

Example 2

e.g. $f(x) \propto e^{-x^2}$ is not integrable, but

$$\begin{aligned}
f(x)\,\mathrm{d}x\,f(y)\,\mathrm{d}y &\propto e^{-(x^2+y^2)}\,\mathrm{d}x\,\mathrm{d}y \\
&= e^{-r^2}\,r\mathrm{d}r\,\mathrm{d}\varphi \propto e^{-r^2}\,\mathrm{d}r^2\,\mathrm{d}\varphi \\
F(r^2) = 1 - e^{-r^2} &\Longrightarrow r^2 = -\ln R_1 \\
x &= \sqrt{-\ln R_1}\,\cos(2\pi R_2) \\
y &= \sqrt{-\ln R_1}\,\sin(2\pi R_2)
\end{aligned}$$

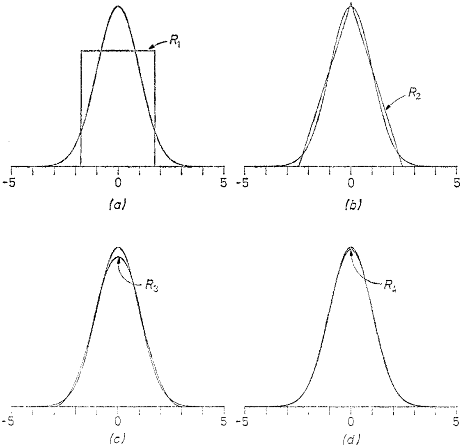
Can combine with angular selection trick of example 1.

Gaussian distribution (2)

Central limit theorem: the sum of a large number of random variables is normally distributed, i.e. a Gaussian.

$$\begin{aligned}
\langle R \rangle &= 1/2 \\
\sigma^2(R) &= 1/12 \\
\mathrm{R}_N &= \sum_{i=1}^{N} R_i \\
\langle R_N \rangle &= N/2 \\
\sigma^2(R_N) &= N/12 \\
N(0,1) &= (R_N - N/2) \\
&/ \ (N/12)
\end{aligned}$$

Figure for $N = 1, 2, 3, 12$



Forsythe–Von Neumann

$f(x) = c\,\exp(-g(x))$, $a \le x \le b$, $0 \le g(x) \le 1$

1. $x = a + R(b-a)$, $t = g(x)$
   $i = 0$, $u_0 = t = g(x)$
2. $i = i + 1$, $u_i = R$
3. if $u_i < u_{i-1}$ cycle to 2
4. if i is even cycle to 1

gives x distributed according to $f(x)$ since

$$
\begin{aligned}
P(R_1 > R_2 > \ldots > R_i) &= 1/i! \\
P(t > R_1 > \ldots > R_i) &= t^i/i! \\
P(t > R_1 > \ldots > R_{i-1} < R_i) &= P(t > R_1 > \ldots > R_{i-1}) \\
&- P(t > R_1 > \ldots > R_i) \\
&= t^{i-1}/(i-1)! - t^i/i! \\
P(i\,\mathrm{odd}) = P(1) + P(3)\ldots &= 1 - t + t^2/2! - t^3/3! + \ldots \\
&= \exp(-t) = \exp(-g(x)) \propto f(x)
\end{aligned}
$$

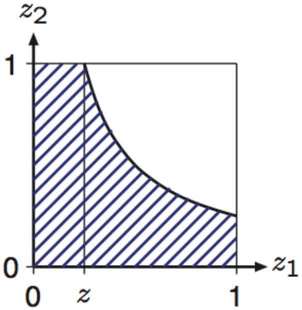
Works for Gaussian, but only over finite range

Combination of random numbers

Recall example 5 of lecture 1:
$f(x) = x\,e^{-x}$, $x > 0$
No invertible primitive function,
so use importance sampling...

...or pull the rabbit out of the hat:

$x = -\ln(R_1\,R_2)$



since with $z = z_1\,z_2 = R_1\,R_2$

$$F(z) = \int_0^z f(z')\,\mathrm{d}z' = \int_0^z 1\,\mathrm{d}z_1 + \int_z^1 \frac{z}{z_1}\,\mathrm{d}z_1 = z - z\ln z$$

and using that $x = -\ln z \iff z = e^{-x}$

$$F(x) = 1 - F(z = e^{-x}) = 1 - e^{-x} + e^{-x}(-x) \Longrightarrow f(x) = x\,e^{-x}$$

Convolutions

The probability density of a sum of random numbers is the convolution of their probability density functions.

If $x_i = g_i(R_i)$ gives probability densities $f_i(x_i)$, $1 \leq i \leq n$, then

$$f_{\mathrm{sum}}\left(x = \sum_{i=1}^{n} x_i\right) = \left(\prod_{i=1}^{n} \int f_i(x_i)\,\mathrm{d}x_i\right) \delta\left(x - \sum_{i=1}^{n} x_i\right)$$

which for $n = 2$ reads $f_{\mathrm{sum}}(x = x_1 + x_2) = \int f(x_1)\, f(x - x_1)\,\mathrm{d}x_1$.

If now $x = -\ln R$ gives $f(x) = e^{-x}\theta(x)$ then
$x = (-\ln R_1) + (-\ln R_2) = -\ln(R_1 R_2)$ gives

$$f_{\mathrm{sum}}(x) = \int_{-\infty}^{\infty} e^{-x_1}\theta(x_1)\, e^{-(x-x_1)}\theta(x-x_1)\,\mathrm{d}x_1 = e^{-x}\int_0^x 1\,\mathrm{d}x_1 = x\, e^{-x}$$

and $x = -\sum_{i=1}^{n} \ln R_i = -\ln\left(\prod_{i=1}^{n} R_i\right)$ gives

$$f_{\mathrm{sum}}(x) = e^{-x}\int_0^x \mathrm{d}x_1 \int_0^{x-x_1} \mathrm{d}x_2 \cdots \int_0^{x-x_1\cdots-x_{n-2}} \mathrm{d}x_{n-1} = \frac{x^{n-1}\, e^{-x}}{(n-1)!}$$

Convolutions (2)

Also possible to work with multiplication:

$$\begin{aligned}
f_{\mathrm{prod}}(x = x_1 x_2) &= \int f_1(x_1)\,\mathrm{d}x_1 \int f_2(x_2)\,\mathrm{d}x_2\,\delta(x - x_1 x_2) \\
&= \int f_1(x_1)\,\mathrm{d}x_1 \int f_2(x_2)\,\mathrm{d}x_2\,\frac{1}{x_1}\delta\left(x_2 - \frac{x}{x_1}\right) \\
&= \int f_1(x_1)\,f_2\left(\frac{x}{x_1}\right)\,\frac{\mathrm{d}x_1}{x_1}
\end{aligned}$$

For instance $0 < R < 1$ flat is $f(x) = \theta(x)\,\theta(1-x)$ with

$$f_{\mathrm{prod}}(R_1 R_2) = \int_0^1 \theta\left(\frac{x}{x_1}\right)\,\theta\left(1 - \frac{x}{x_1}\right)\,\frac{\mathrm{d}x_1}{x_1} = \int_x^1 \frac{\mathrm{d}x_1}{x_1} = -\ln x$$

so $f(x) = -\ln x$ for $0 < x < 1$ is simply generated by $x = R_1 R_2$.

Special tricks summary

Many smart tricks for a host of distributions.
However, usually too complicated functions to know how to start, so often fall back on generic methods like importance sampling.

Another crude approach: the histogram method.

- Initialization: subdivide range $a \leq x \leq b$ in n equal-length bins, and evaluate $f(x)$ in the middle of each bin.
- Generation: pick one of bins according to their relative weight (see lecture 1, discrete possibilities), and inside that bin uniformly between the edges.
- Possible improvement 1: guesstimate upper bound of $f(x)$ in each bin (by random sampling in bin + safety margin), use those bounds for bin selection, and then hit-and-miss
- Possible improvement 2: find bins where $f(x)$ varies rapidly, and subdivide those into smaller bins

Multidimensional spatial integrals

In practice almost always multidimensional integrals:
$\mathbf{x} = (x_1, x_2, \ldots, x_d)$, $\mathrm{d}P = f(\mathbf{x})\,\mathrm{d}\mathbf{x}$ inside region Ω.

Assume Ω can be inscribed inside hyperrectangle with volume V:

$$I = \int_\Omega f(\mathbf{x})\,\mathrm{d}\mathbf{x} = V\,\frac{1}{N_{\mathrm{try}}}\sum_i \theta(\mathbf{x} \in \Omega)\,f(\mathbf{x}_i)$$

gives error $\propto 1/\sqrt{N}$ irrespective of dimension.
It beats e.g. Simpson, $\propto 1/N^{4/d}$, as already discussed.

Selection of $\mathbf{x}$ requires e.g hit-and-miss:
if $f(\mathbf{x}) < f_{\mathrm{max}}$ for $\mathbf{x} \in \Omega$ then

1. select $\mathbf{x}$ uniformly inside V (simple since factorized)
2. if $\mathbf{x}$ outside Ω cycle to 1
3. if $f(\mathbf{x}) < R\,f_{\mathrm{max}}$ cycle to 1

which gives integration error $\approx 1/\sqrt{N_{\mathrm{acc}}}$.

Analytical solutions

Analytical solutions require possibility to successively integrate out one dimension at a time, with proper boundaries, and invert primitive functions. Unless dimensions factorize this is not trivial.

Example 3:

$f(x_1, x_2) = 1 + x_1 x_2$, $0 \leq x_1 \leq 1$, $0 \leq x_2 \leq 1$

$$\begin{aligned}
f_1(x_1) &= \int_0^1 f(x_1, x_2)\,\mathrm{d}x_2 = \left[x_2 + x_1 \frac{x_2^2}{2} \right]_0^1 = 1 + \frac{x_1}{2} \\
F_1(x_1) &= \int_0^{x_1} f_1(x_1')\,\mathrm{d}x_1' = x_1 + \frac{x_1^2}{4} = R(F_1(1) - F_1(0)) = \frac{5}{4} R \\
x_1 &= \sqrt{4 + 5R} - 2 \\
F_2(x_2) &= \int_0^{x_2} f(x_1, x_2')\,\mathrm{d}x_2' = x_2 + x_1 \frac{x_2^2}{2} = R\left(1 + \frac{x_1}{2}\right) \\
x_2 &= \frac{1}{x_1}\left(\sqrt{1 + R(x_1^2 + 2x_1)} - 1\right)
\end{aligned}$$

Multichannel importance sampling

If $f(\mathbf{x}) \leq g(\mathbf{x}) = g^{(1)}(x_1)\, g^{(2)}(x_2) \cdots g^{(d)}(x_d)$
and each $g^{(j)}(x_j) = \sum_i g_i^{(j)}(x_j)$ then

1. for each dimension j select i according to integral of $g_i^{(j)}(x_j)$
2. for each dimension j select x_j according to $g_i^{(j)}(x_j)$
3. if $\mathbf{x}$ outside Ω cycle to 1
4. if $f(\mathbf{x}) < R\, g(\mathbf{x})$ cycle to 1

Example 3':

$f(x_1, x_2) = 1 + x_1 x_2$, $0 \leq x_1 \leq 10$, $0 \leq x_2 \leq 10$.
$g(x_1, x_2) = 101$ gives efficiency $2600/10100 \approx 0.26$, while
$g(x_1, x_2) = (1 + x_1)(1 + x_2)$ gives efficiency $2600/3600 \approx 0.72$.

Of course also allowed to have sum of terms $g(\mathbf{x}) = \sum_k g_k(\mathbf{x})$
where each g_k in its turn may (or may not) factorize,
e.g. $f(\mathbf{x}) \leq g(\mathbf{x}) = \sum_k \prod_{j=1}^{d} \sum_i g_{ki}^{(j)}(x_j)$

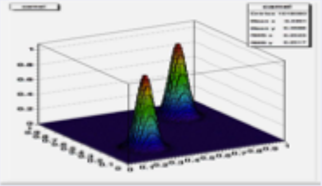
Stratified sampling

Multidimensional integration e.g. with VEGAS program:
(classic; later alternatives exist, like Bases/Spring)

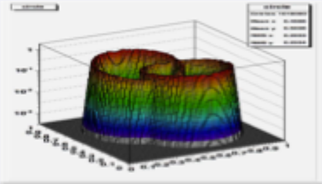
- assumes separable upper estimate
  $g(\mathbf{x}) = g^{(1)}(x_1)\, g^{(2)}(x_2) \cdots g^{(d)}(x_d)$
- assumes hyperrectangle volume in $\mathbf{x}$
- learning points at random inside volume
- $g^{(j)}(x_1)$ obtained by projection onto respective axis
- originally equal-length "histogram" bins along each axis,
  but then 5 - 10 rebinning iterations
  so that important regions are better sampled
- provides integral value, but also Monte Carlo generation
- user responsibility: carefully consider axes rotations
  that may align peaks/ridges

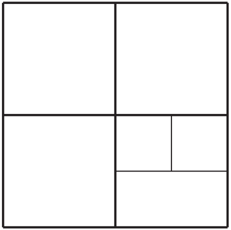
Stratified sampling (2)

- Good for Vegas: Singularity "parallel" to integration axes



- Bad for Vegas: Singularity forms ridge along integration axes

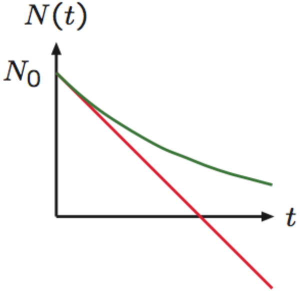




Possible improvement:
subsplittings along only part of axis.
OK for low dimensions, but generally too complicated.

Temporal methods: radioactive decays

Consider "radioactive decay":
$N(t)$ = number of remaining nuclei at time t
but normalized to $N(0) = N_0 = 1$ instead, so equivalently
$N(t)$ = probability that (single) nucleus has not decayed by time t
$P(t) = -\mathrm{d}N(t)/\mathrm{d}t$ = probability for it to decay at time t



Naively $P(t) = c \Longrightarrow N(t) = 1 - ct$.
Wrong! Conservation of probability driven by depletion:
a given nucleus can only decay once

Correctly
$P(t) = cN(t) \Longrightarrow N(t) = \exp(-ct)$
i.e. exponential dampening
$P(t) = c\exp(-ct)$

There is memory in time!

Temporal methods: radioactive decays (2)

For radioactive decays $P(t) = cN(t)$, with c constant, but now generalize to time-dependence:

$$P(t) = -\frac{\mathrm{d}N(t)}{\mathrm{d}t} = f(t)\,N(t)\ ; \quad f(t) \geq 0$$

Standard solution:

$$\frac{\mathrm{d}N(t)}{\mathrm{d}t} = -f(t)N(t) \iff \frac{\mathrm{d}N}{N} = \mathrm{d}(\ln N) = -f(t)\,\mathrm{d}t$$

$$\ln N(t) - \ln N(0) = -\int_0^t f(t')\,\mathrm{d}t' \implies N(t) = \exp\left(-\int_0^t f(t')\,\mathrm{d}t'\right)$$

$$F(t) = \int^t f(t')\,\mathrm{d}t' \implies N(t) = \exp\left(-(F(t) - F(0))\right)$$

Assuming $F(\infty) = \infty$, i.e. always decay, sooner or later:

$$N(t) = R \implies t = F^{-1}(F(0) - \ln R)$$

The veto algorithm: problem

What now if $f(t)$ has no simple $F(t)$ or F^{-1}?
Hit-and-miss not good enough, since for $f(t) \leq g(t)$, g "nice",

$$t = G^{-1}(G(0) - \ln R) \implies N(t) = \exp\left(- \int_0^t g(t')\,\mathrm{d}t' \right)$$

$$P(t) = -\frac{\mathrm{d}N(t)}{\mathrm{d}t} = g(t)\, \exp\left(- \int_0^t g(t')\,\mathrm{d}t' \right)$$

and hit-or-miss provides rejection factor $f(t)/g(t)$, so that

$$P(t) = f(t)\, \exp\left(- \int_0^t g(t')\,\mathrm{d}t' \right)$$

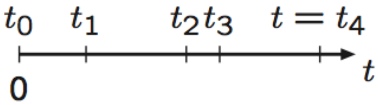
(modulo overall normalization), where it ought to have been

$$P(t) = f(t)\, \exp\left(- \int_0^t f(t')\,\mathrm{d}t' \right)$$

The veto algorithm: solution

The veto algorithm

1. start with $i = 0$ and $t_0 = 0$
2. $i = i + 1$
3. $t_i = G^{-1}(G(t_{i-1}) - \ln R)$, i.e $t_i > t_{i-1}$
4. $y = R\, g(t)$
5. while $y > f(t)$ cycle to 2



That is, when you fail, you keep on going from the time when you failed, and *do not* restart at time $t = 0$. (Memory!)

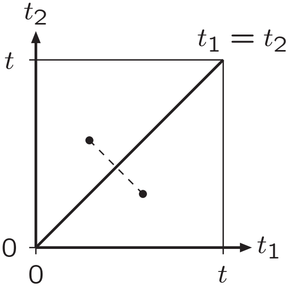
The veto algorithm: proof (1)

Study probability to have i intermediate failures before success:

Define $S_g(t_a, t_b) = \exp\left(-\int_{t_a}^{t_b} g(t')\,\mathrm{d}t'\right)$ ("Sudakov factor")

$$P_0(t) = P(t = t_1) = g(t)\,S_g(0,t)\,\frac{f(t)}{g(t)} = f(t)\,S_g(0,t)$$

$$\begin{aligned}
P_1(t) &= P(t = t_2) \\
&= \int_0^t \mathrm{d}t_1\, g(t_1) S_g(0,t_1) \left(1 - \frac{f(t_1)}{g(t_1)}\right) g(t)\,S_g(t_1,t)\,\frac{f(t)}{g(t)} \\
&= f(t)\,S_g(0,t) \int_0^t \mathrm{d}t_1\,(g(t_1) - f(t_1)) = P_0(t)\,I_{g-f} \\
P_2(t) &= \cdots = P_0(t) \int_0^t \mathrm{d}t_1\,(g(t_1) - f(t_1)) \int_{t_1}^t \mathrm{d}t_2\,(g(t_2) - f(t_2)) \\
&= P_0(t) \int_0^t \mathrm{d}t_1\,(g(t_1) - f(t_1)) \int_0^t \mathrm{d}t_2\,(g(t_2) - f(t_2))\,\theta(t_2 - t_1) \\
&= P_0(t)\,\frac{1}{2}\left(\int_0^t \mathrm{d}t_1\,(g(t_1) - f(t_1))\right)^2 = P_0(t)\,\frac{1}{2}\,I_{g-f}^2
\end{aligned}$$



Generally, i intermediate times corresponds to $i!$ equivalent ordering regions.

$$P_i(t) = P_0(t)\,\frac{1}{i!}\,I_{g-f}^i$$

$$\begin{aligned} P(t) &= \sum_{i=0}^{\infty} P_i(t) = P_0(t) \sum_{i=0}^{\infty} \frac{I_{g-f}^i}{i!} = P_0(t)\exp(I_{g-f}) \\ &= f(t)\,\exp\left(-\int_0^t g(t')\,\mathrm{d}t'\right)\,\exp\left(\int_0^t (g(t') - f(t'))\,\mathrm{d}t'\right) \\ &= f(t)\,\exp\left(-\int_0^t f(t')\,\mathrm{d}t'\right) \end{aligned}$$

Poissonian resummed

Assume infinite chain of sequential "radioactive decays",
$(A, Z) \to (A, Z+1) \to (A, Z+2) \to \cdots$,
each with same decay rate $f(t)$.
How many steps in chain between 0 and time t?

$$\begin{aligned}
P_0(t) &= S_f(0,t) = \exp\left(-\int_0^t f(t')\,\mathrm{d}t'\right) = \exp(-I_f(t)) \\
P_1(t) &= \int_0^t \mathrm{d}t_1\, S_f(0,t_1)\, f(t_1)\, S_f(t_1,t) = S_f(0,t) I_f(t) \\
P_n(t) &= S_f(0,t) \frac{I_f(t)^n}{n!} = \frac{I_f(t)^n}{n!} \exp(-I_f(t))
\end{aligned}$$

i.e. Poissonian with $\langle n \rangle = I_f(t)$.
For upper estimate $g(t)$ correspondingly $\langle n' \rangle = I_g(t)$,
so efficiency $\langle n_{\mathrm{acc}} \rangle / \langle n_{\mathrm{try}} \rangle = \langle n \rangle / \langle n' \rangle = I_f(t)/I_g(t)$.
Probability for decay at time t is

$$P(t) = \sum_{i=0}^{\infty} P_n(t)\, f(t) = f(t)$$

Radioactive decay as perturbation theory

Assume we don't know about exponential function.
Recall wrong solution, starting from $N(t) = N_0(t) = 1$:

$$\frac{\mathrm{d}N}{\mathrm{d}t} = -cN = -cN_0(t) = -c \Rightarrow N(t) = N_1(t) = 1 - ct$$

Now plug in $N_1(t)$, hoping to find better approximation:

$$\frac{\mathrm{d}N}{\mathrm{d}t} = -cN_1(t) \Rightarrow N(t) = N_2(t) = 1 - c \int_0^t (1 - ct')\mathrm{d}t' = 1 - ct + \frac{(ct)^2}{2}$$

and generalize to

$$N_{i+1}(t) = 1 - c \int_0^t N_i(t')\,\mathrm{d}t' \Rightarrow N_{i+1}(t) = \sum_{k=0}^{i+1} \frac{(-ct)^k}{k!}$$

which recovers exponential e^{-ct} for $i \to \infty$.

Reminiscent of (QED, QCD) perturbation theory with $c \to \alpha f$.

Combining time and space

Radioactive decay $n \to pe^- \overline{\nu}_e$ has continuous energy spectrum E_e.
Want to study both time and $x = E_e$, i.e.

$$P(t, x) = f(t, x)\, N(t)\ ; \quad f(t, x) \geq 0$$

A. When $f(t) = \int f(t, x)\, \mathrm{d}x$ easily found

1. Find simple $g(t) \geq f(t)$
2. Use veto algorithm to pick t
3. Use $f(t, x)$ to pick x for chosen t

B. When $\int f(t, x)\, \mathrm{d}x$ not easily found

1. Find simple $g(t, x) \geq f(t, x)$
2. Use $g(t) = \int g(t, x)$ to pick t
3. Use $g(t, x)$ to pick x for chosen t
4. Accept with probability $f(t, x)/g(t, x)$, else evolve further in t

Combining time and space (2)

Example 1

$f(t,x) = 1 - 4x(1-x)e^{-t}, \quad t \geq 0, \quad 0 \leq x \leq 1$

Alternative A

$f(t) = \int_0^1 f(t,x)\,\mathrm{d}x = 1 - (2/3)e^{-t}$,
$g(t) = 1 \Rightarrow G(t) = \int_0^t g(t')\,\mathrm{d}t' = t \Rightarrow G^{-1}(t) = t$.

1. $i = 0,\ t_0 = 0$
2. $i = i + 1$
3. $t_i = t_{i-1} - \ln R$ (note: memory!)
4. if $1 - (2/3)e^{-t} < R$ cycle to 2
5. $x = R$ (note: **no** memory!)
6. if $f(t,x) < R$ cycle to 5

i.e. selection of t and x separated.

Note: 5 + 6 can be replaced by solving third-degree equation.

Combining time and space (3)

Example 1 (continued)
$f(t,x) = 1 - 4x(1-x)e^{-t}, \quad t \geq 0, \quad 0 \leq x \leq 1$

Alternative B
$f(t,x) \leq g(t,x) = 1, \quad g(t) = \int_0^1 g(t,x)\,\mathrm{d}x = 1,$
$\Rightarrow G(t) = \int_0^t g(t')\,\mathrm{d}t' = t \Rightarrow G^{-1}(t) = t.$

1. $i = 0,\ t_0 = 0$
2. $i = i + 1$
3. $t_i = t_{i-1} - \ln R$
4. $x = R$
5. if $f(t,x) < R$ cycle to 2

i.e. selection of t and x integrated.

Both alternatives can be proven correct with simple extensions of the veto algorithm proof.
Usually B is more convenient, even if possible loss of efficiency.

The winner takes it all

Assume "radioactive decay" with two possible decay channels 1&2

$$P(t) = -\frac{\mathrm{d}N(t)}{\mathrm{d}t} = f_1(t)N(t) + f_2(t)N(t)$$

Alternative 1:
use normal veto algorithm with $f(t) = f_1(t) + f_2(t)$.
Once t selected, pick decays 1 or 2 in proportions $f_1(t) : f_2(t)$.

Alternative 2:

The winner takes it all

select t_1 according to $P_1(t_1) = f_1(t_1)N_1(t_1)$
and t_2 according to $P_2(t_2) = f_2(t_2)N_2(t_2)$,
i.e. as if the other channel did not exist.
If $t_1 < t_2$ then pick decay 1, while if $t_2 < t_1$ pick decay 2.

The winner takes it all: proof

$$
\begin{aligned}
P_1(t) &= (f_1(t) + f_2(t)) \; \exp\left(- \int_0^t (f_1(t') + f_2(t')) \, \mathrm{d}t' \right) \; \frac{f_1(t)}{f_1(t) + f_2(t)} \\
&= f_1(t) \; \exp\left(- \int_0^t (f_1(t') + f_2(t')) \, \mathrm{d}t' \right) \\
&= f_1(t) \; \exp\left(- \int_0^t f_1(t') \, \mathrm{d}t' \right) \; \exp\left(- \int_0^t f_2(t') \, \mathrm{d}t' \right)
\end{aligned}
$$

i.e. probability to pick decay kind 1 at t, times no kind 2 before t.
Algorithm especially convenient when temporal and/or spatial dependence of f_1 and f_2 are rather different.

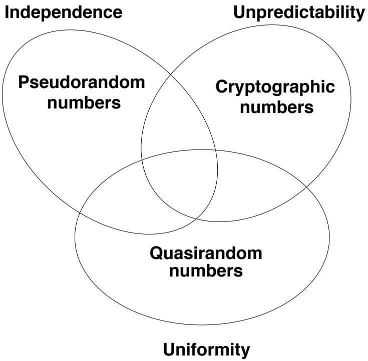
Efficiency trick, when veto algorithm used for both channels:

- order processes so that the most likely is considered first
- interrupt veto-algorithm evolution of the second process if its trial time t_2 exceeds the previously selected t_1

Random numbers

Three kinds:

- Truly random = "uncorrelated" = "impossible to guess"
- Pseudorandom: approximation to truly random
  = deterministic sequence that passes tests of randomness
- Quasirandom: well distributed, i.e. less-than-random



but overlap
not obvious

Truly random numbers

Classical example:

- radioactive source
- adjust time slot so that ~ 1 decay/slot (alternatively $\gg 1$)
- record stream 1/0 for decay/not
  (alternatively record 1/0 for odd/even)
- combine two and two, $10 \to 1$, $01 \to 0$, throw 11 and 00
  (so insensitive to unknown "true" decay rate)
- use to build binary representation of number between 0 and 1

Technically not simple to avoid any biases, but 2.5M such numbers were produced in 1978, and made available on tape.
Today corresponds to a fraction of a second of MC usage.
Nowadays there exist hardwre random number generators, based e.g. on (Schottky) noise in electronic circuits, but not common.
Main usage: cryptography (need unpredictability; small bias OK).

Quasirandom numbers

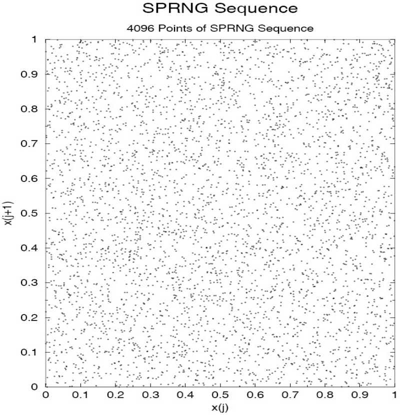
Objective: distribute numbers more evenly than random

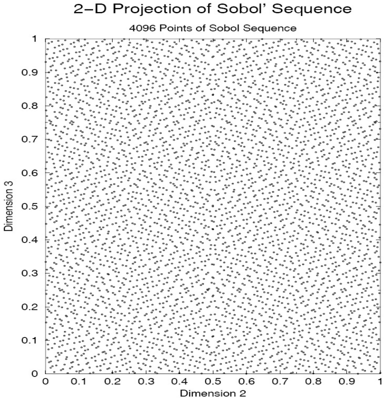
Simple example: van der Corput

- count up integers in base P, where P is a prime
- reverse the digits, precede by a point and interpret number as a fraction in base P, e.g. for $P = 2$:

decimal	binary	binary fraction	rational fraction	decimal fraction
1	1	0.1	1/2	0.5
2	10	0.01	1/4	0.25
3	11	0.11	3/4	0.75
4	100	0.001	1/8	0.125
5	101	0.101	5/8	0.625
6	110	0.011	3/8	0.375
7	111	0.111	7/8	0.875
8	1000	0.0001	1/16	0.0625

Pseudorandom vs. quasirandom





Sobol sequence: more complicated but similar in spirit to van der Corput.

Early pseudorandom number generators

Linear congruential (or multiplicative congruential):

$$R_i = aR_{i-1} \pmod{m}$$

Mixed congruential:

$$R_i = aR_{i-1} + b \pmod{m}$$

Specified by integers a, b, m and seed R_0.
For computer: $m = 2^t$, where t is number of bits in representation of integers, say 32 (or 31 if $\pm$ bit cannot be used).
Then do multiplication in 64 bit registers and throw away all but the last t bits. Convert to real $R' = R/m$ with $0 \leq R' < 1$.
Has to repeat; maximum period $m/2$ (from odd/even).
Period $2^{30} \approx 10^9$ not good enough today.
$t = 64$ requires special tricks to avoid uncontrolled overflow.

Randomness

What is "good randomness"?

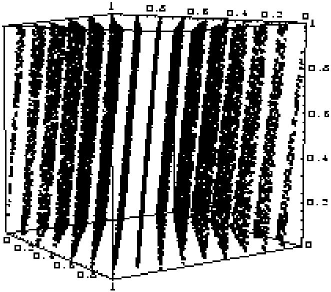
- mean $1/2$, variance $1/12$, ...
- histogram: subdivide range $0 \leq x \leq 1$ in n bins.
  The number in each bin should be $\sim$Poissonian,
  i.e. for large N Gaussian with mean N/n and width $\sqrt{N/n}$.
- Monte Carlo results where analytical answer known
- run tests, poker tests, ...

Randomness

What is "good randomness"?

- mean $1/2$, variance $1/12$, ...
- histogram: subdivide range $0 \leq x \leq 1$ in n bins.
  The number in each bin should be $\sim$Poissonian,
  i.e. for large N Gaussian with mean N/n and width $\sqrt{N/n}$.
- Monte Carlo results where analytical answer known
- run tests, poker tests, ...

...and then came the
Marsaglia effect (1968):
random numbers fall in planes!



The Marsaglia Effect

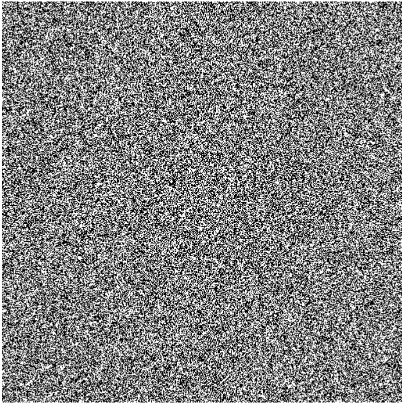
Take successive d-tuplets from a congruential generator with t bits.
Interpret them as point coordinates in a d-dimensional hypercube.
Then they all fall on at most $(d!2^t)^{1/d}$ parallel hyperplanes.

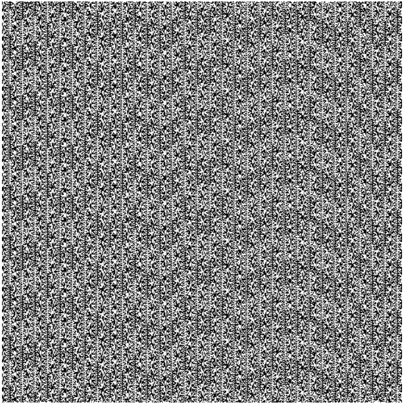
t	$d=3$	$d=4$	$d=6$	$d=10$
16	73	35	19	13
32	2 953	566	120	41
48	119 086	9 065	766	126
64	4 801 280	145 055	4 866	382

Disastrous for any "repetitive" application (fixed-length cycle).
Has lead to explosion of new tests and new generators.

The Marsaglia Effect (2)

512×512 2D-array: white if $R < 0.5$, black if $R > 0.5$:





Generator examples

1. Linear congruential:
   $R_i = aR_{i-1} + b \pmod{m}$
   Period $= m$
2. Implicit inversive congruential:
   $R_i = a\overline{R_{i-1}} + b \pmod{p}$
   Period $= p$
3. Explicit inversive congruential:
   $R_i = a\overline{i} + b \pmod{p}$
   Period $= p$
4. Shift register: (one bit at a time!)
   $R_i = R_{i-q} + R_{i-p} \pmod{2}, p > q$
   Period $= 2^p - 1$
5. Additive lagged Fibonacci (can use real numbers!)
   $R_i = R_{i-q} + R_{i-p} \pmod{2^t}, p > q$
   Period $= (2^p - 1)2^{t-1}$

Generator examples (2)

6. Combined
   $R_i = R_i^{(I)} + R_i^{(II)} \pmod{p}$
   Period = least common multiple of two periods
7. Multiplicative lagged Fibonacci
   $R_i = R_{i-q} \times R_{i-p} \pmod{2^t}, p > q$
   Period $= (2^p - 1)2^{t-3}$
8. l'Ecuyer: combined multiplicative linear congruential
9. Mersenne twister: uses Mersenne primes ($M_p = 2^p - 1$ with p and M_p primes), bit masks, bit shifts
10. WELL: well equidistributed long-period linear
11. MIXMAX: multiplicative (under development)
12. ...

Some theory

View a set $\mathbf{R} = (R_1, R_2, \ldots, R_n)$ as a vector in n-dimensional space.
Define a norm, i.e. a distance between two $\mathbf{R}$ vectors, e.g. as

$$||\mathbf{R}' - \mathbf{R}|| = \min \left(|R'_1 - R_1|, |R'_2 - R_2|, \ldots, |R'_n - R_n| \right)$$

Define a $n \times n$ matrix $\mathbf{A}$ such that $\mathbf{R}^{(k+1)} = \mathbf{A}\mathbf{R}^{(k)}$,
i.e. $R_i^{(k+1)} = \sum_j A_{ij} R_j^{(k)}$ (with periodic boundary for unit box).
Require determinant, i.e. product of eigenvalues, to be unity.
Plus a number of other "goodness" criteria.

Lyapunov coefficient p: $||\mathbf{R}'^{(k+1)} - \mathbf{R}^{(k+1)}|| = e^p \, ||\mathbf{R}'^{(k)} - \mathbf{R}^{(k)}||$.
p large $\Rightarrow$ vectors initially different in last bit diverge.

Luxury level L: from initial $\mathbf{R}^{(0)}$, generate and throw away
$L - 1$ sets of new vectors before accepting next, i.e. only use
$\mathbf{R}^{(0)}$, $\mathbf{R}^{(L)}$, $\mathbf{R}^{(2L)}$, etc., with L such that $pL > 1$.

MIXMAX: find matrix $\mathbf{A}$ with large Lyapunov p.

Marsaglia–Zaman–Tsang

- Portable and machine-independent, using either 24 or 48 bit single or double precision real numbers.
- One input number $[1, > 900\,000\,000]$ to choose sequence.
- Simple random number generator (lagged Fibonacci + linear congruential) used to initialize 97 numbers u_i, $0 \leq u_i < 1$, bit by bit. Unique setup for each sequence.
- Lagged Fibonacci: $u_i = u_{i-97} - u_{i-33}$, add 1 if < 0. (Do not move random numbers, only update two indices.)
- $R_i = u_i + c$ where c is "arithmetic sequence" (very simple stepping, not good on its own, but adds further randomness).
- Explicitly skip when $R \equiv 0$.
- RANLUX: view as steps of 97-dimensional $\mathbf{R}$, and skip according to suitable Luxury level L. Tradeoff between improved randomness and time consumption.

Generator desiderata and example periods

- Totally reproducibe sequence
- Purely periodic sequence
- Long period
- Moderate cost per bit
- Theoretically solid
- Empirically tested (!)
- Portable

Some algorithms with $R_i = f(R_{i-1}, R_{i-2}, \ldots, R_{i-p})$

name	p	period
linear congruential	1	$\sim 10^9$
L'Ecuyer	3	$\sim 10^{26}$
Marsaglia-Zaman	97	$\sim 10^{171}$
Mersenne twister	623	$\sim 10^{600}$

Some rules of thumb

- Recursions modulo a power of two are cheap, but have simple structure.
- Recursions modulo a prime (like Mersenne) are more costly, but have higher quality.
- Shift-registers (Mersenne twisters) are efficient and were supposed to have good quality, but some flaws found.
- Lagged-Fibonacci generators are efficient and easily portable, but have structural flaws.
- Combining generators is "provably good".
- All linear recursions fall in planes.
- Inversive (nonlinear) recursions fall on hyperbolae.

Random number prospects

- Has become active research field
- Combining several generators, e.g. additively or more complicated (never worse than worst)
- Rewrite for parallelization: not trivial, e.g. one generator to initialize each new other generator for each thread
- Procedures for grid: how ensure nonidentical run in several copies?
- Rewrite for GPU's
- Rewrite for more languages (used to be Fortran)
- Portability vs. power/speed
- Package and distribute generator libraries, preferably with uniform interface, e.g. `http://sprng.org/`
- Documentation and "user experience"