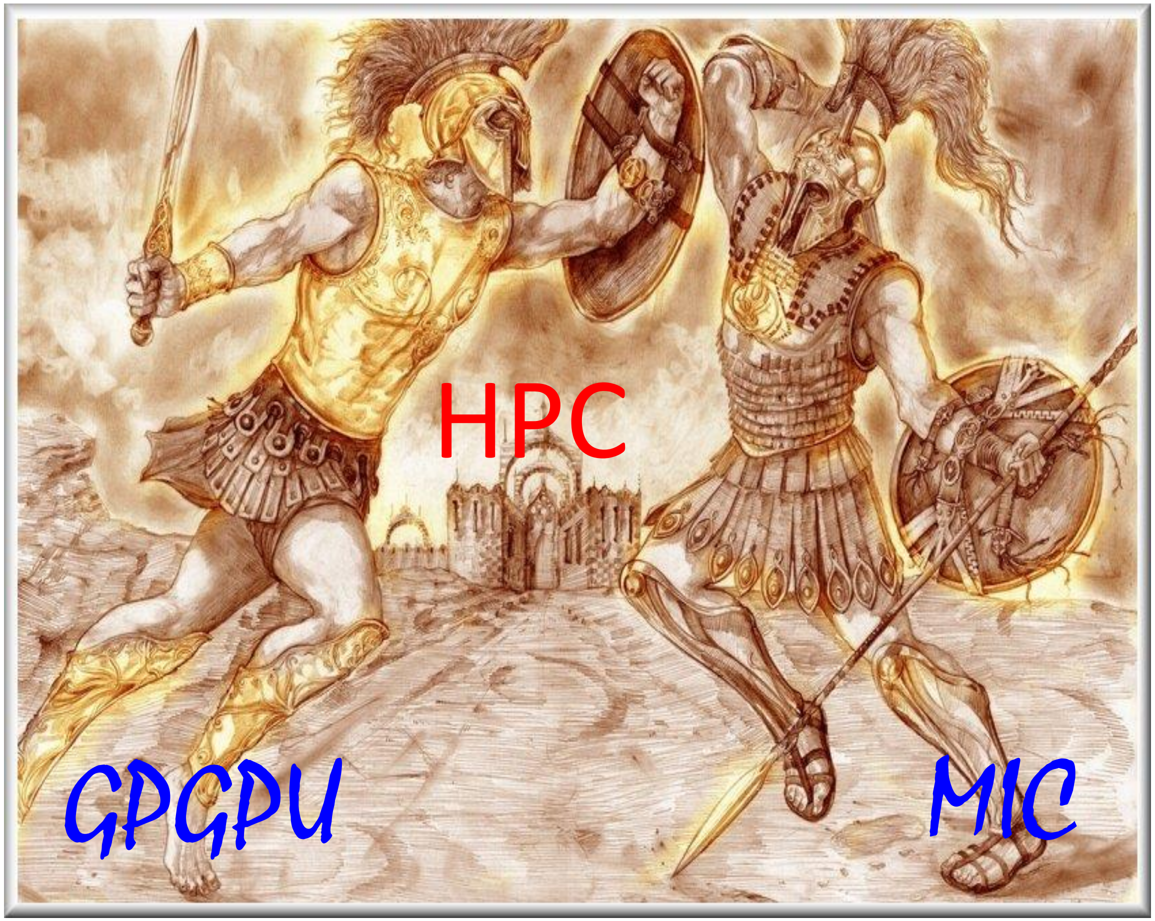
Status of Vectorized Physics

Soon Yung Jun
(FNAL)

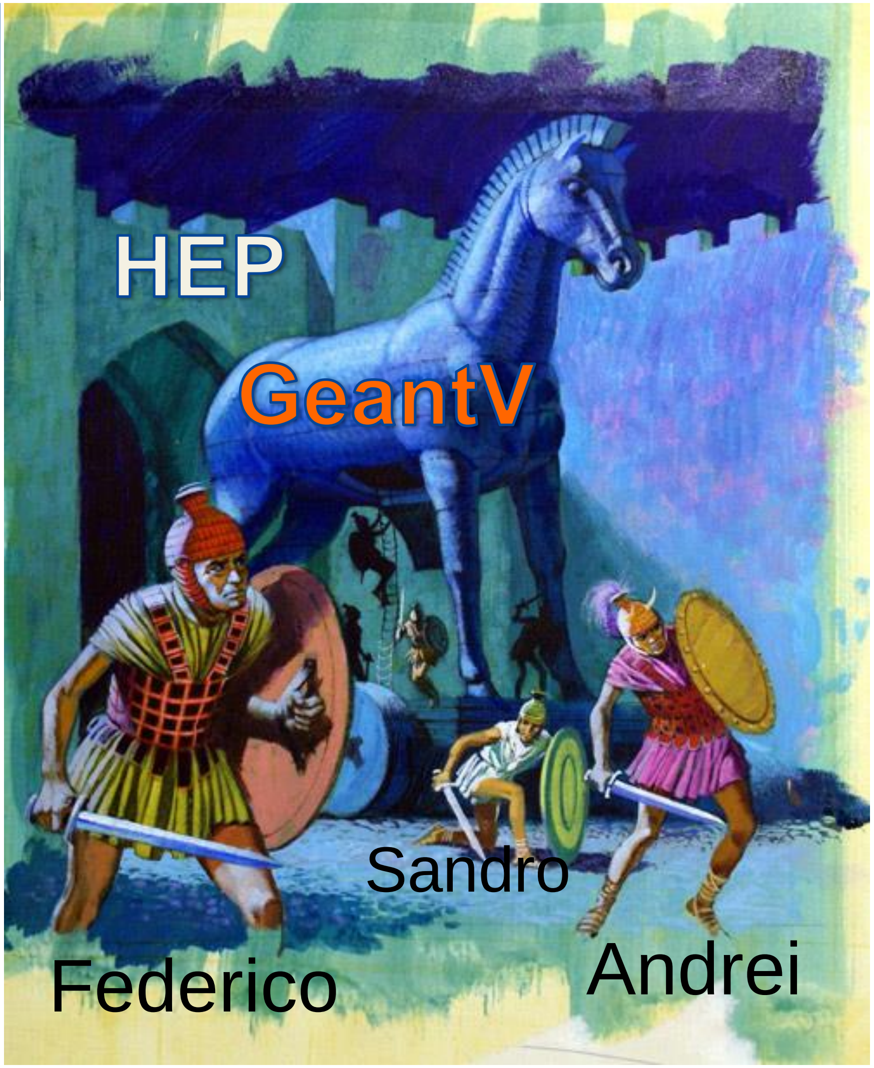
25C Ago: Death of Hector

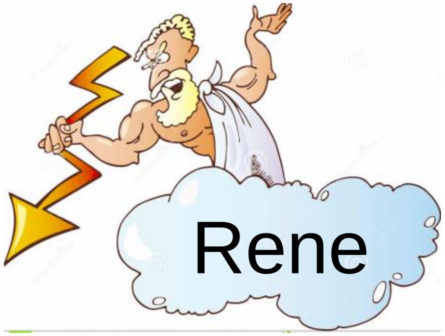
2 Decades Ago: Death of Vector

Today: Vector is still on HPC Battle Field



Allies of GeantV





Specialized Applications

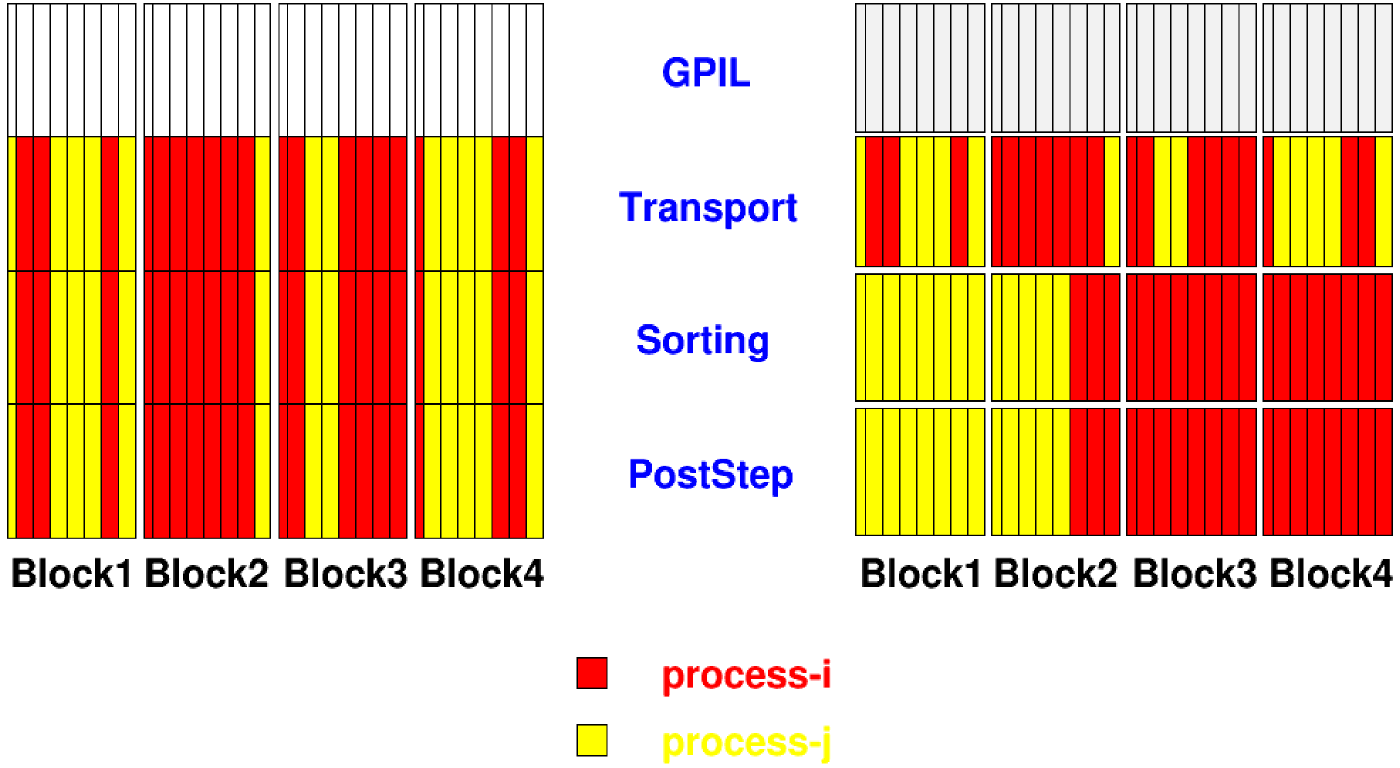
- Vector: no good for convention physics model!
  - pipelined instruction/data throughput
  - locality (spatial, temporal)
- GPU : no good for convention physics model!
  - arithmetic intensity
  - substantial parallelism
  - throughput over latency
- Common strategies or potential solutions?
  - decompose tasks
  - minimize branches and data dependencies
  - share or reuse data

High Performance Physics Models

- Physics processes and models cannot be easily vectorized or parallelized: Why?
  - stochastic process (random-ness kills parallelism)
  - decision trees (branches and switches)
- Paradigm swift
  - task level parallelism
  - algorithmic changes (performance over …)
  - tabulated physics
  - alias method (replace loops with conditional breaks)

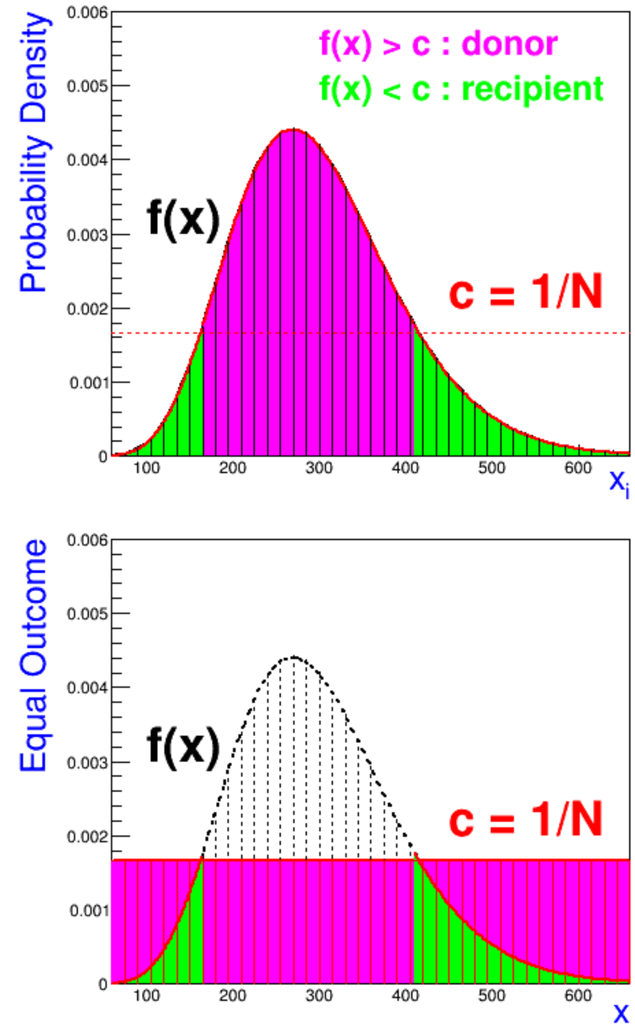
Task Decomposition

- Task by the particle type (e^-, γ, *h*) or geometry
- Regroup particles (tracks) by the same task



Alias Method: Sampling

- Recast p.d.f f(x) to N equal probable events, each with likelihood 1/N=c
- Alias table
  - a[recipient]=donor
  - q[N] = non-alias probability
- Sampling x_j : random u_1, u_2
  - bin index: $N \times u_1 = i + \alpha$
  - If $(q[i] < u_2)$: take j=i
    else take j = a[i]
  - $x_j = [\alpha j + (1-\alpha)(j+1)]\delta x$



Particle Tracking

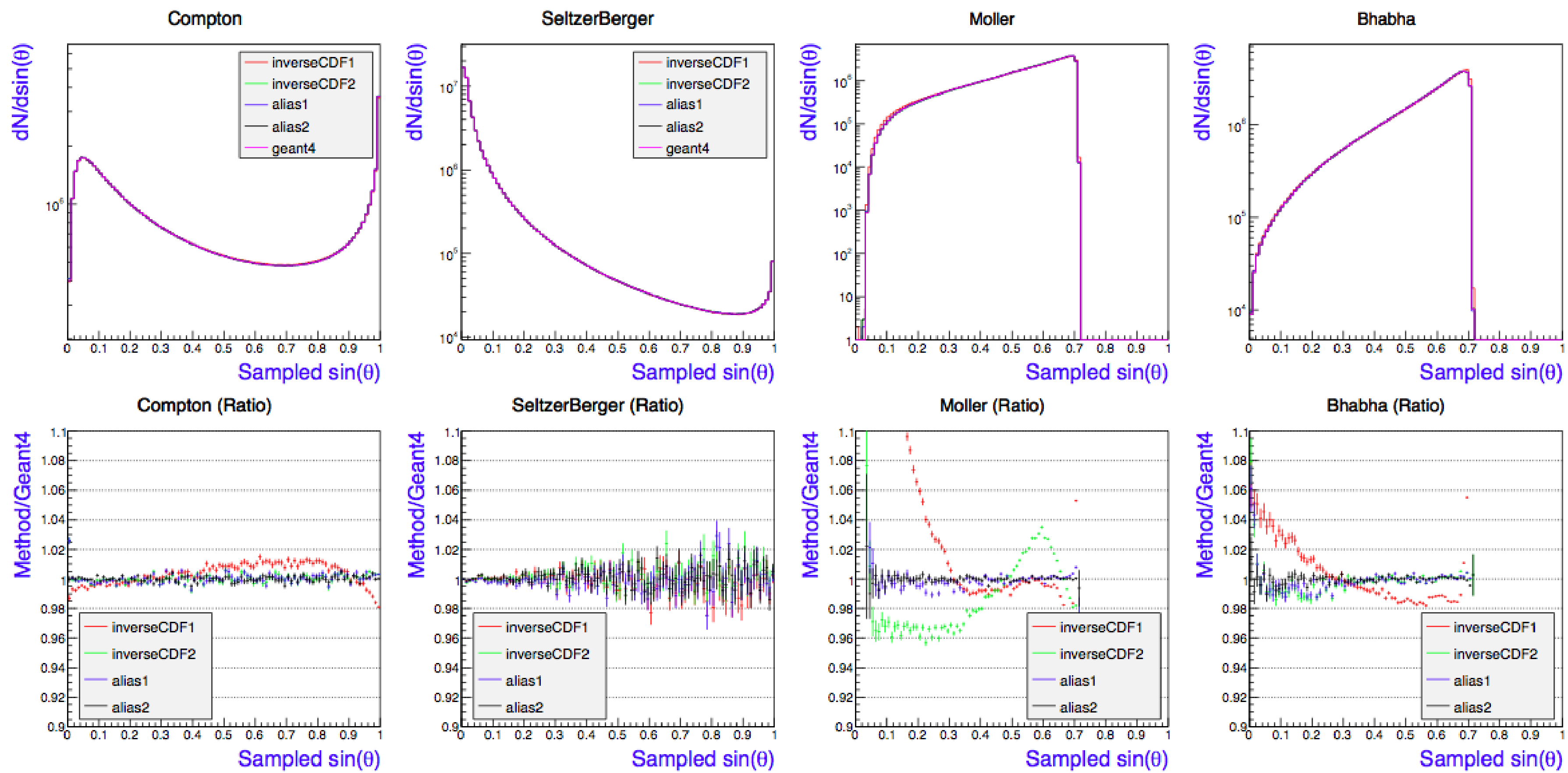
- Process
  - mean free path analysis (step length)
  - sampling physics process
- Associated
  - multiple scattering (true path length)
  - transport (geometry limited step)
- Interaction
  - sampling secondary particles
  - other collision analysis

Applications of Alias Methods

- Replace non-vectorizable conditional branches
  - composition and rejection methods (do-while)
    - no bias compared with Geant4 in secondary particle sampling (see the next page)
  - conditional exit to select (if-break)
    - physics process (relative weight of cross sections)
    - random element of composite material
    - (can have pre-built alias tables from xsec data table?)
- To vectorize alias method effectively
  - coalesced random array
  - gather scatter memory accesses

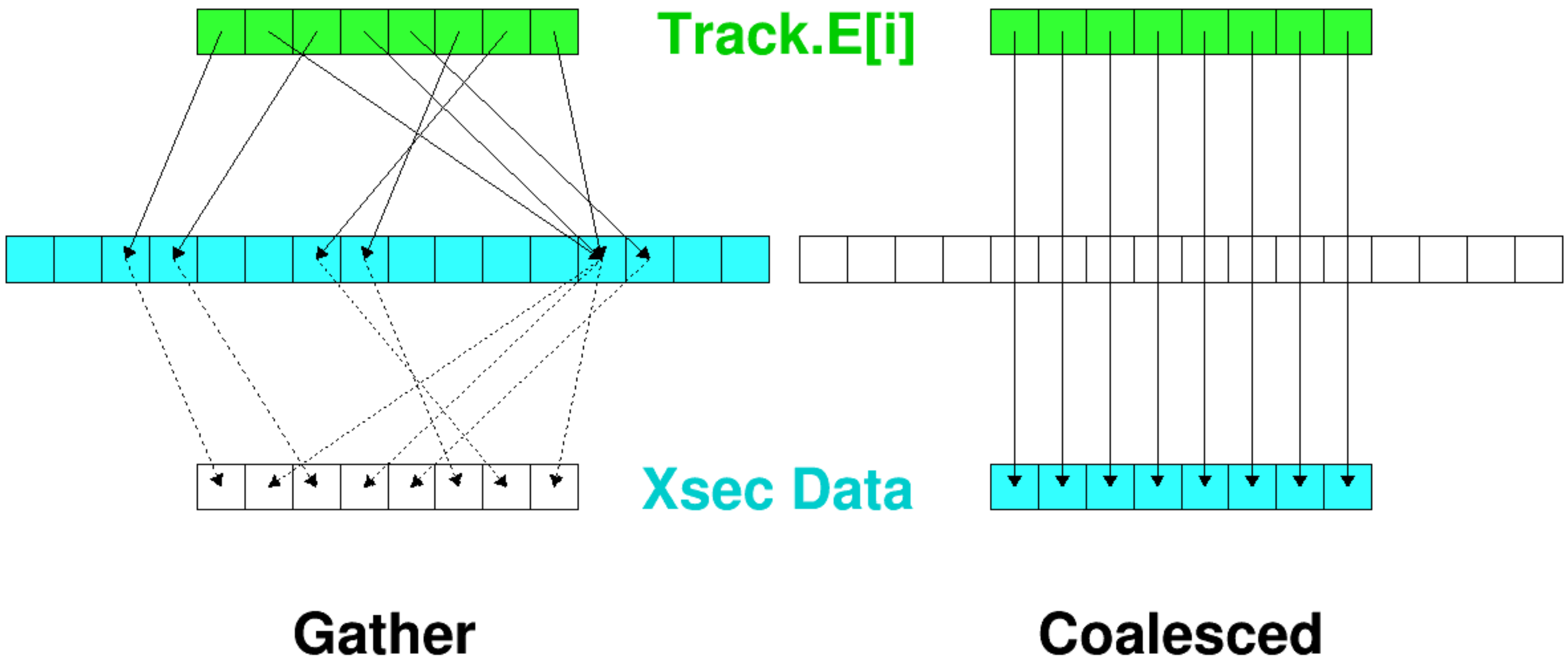
Alias Method: Validation

- Angles of scatter particles



Gather

- Gather operations may be very expensive if associated arithmetic intensity is light



Sampling Physics Process

- Inclusive sampling vs. explicit sampling

	Inclusive	**Explicit**
step length (d) physics process	$d=\min(d_t,d_g)$ $d_t=-\log(u)/\sigma_t$ sampling by xsec weights	$d=\min(d_i,d_g)$ $d_i=-\log(u)/\sigma_i$ Process for the smallest d_i
N_{int} left	process independent	material independent
N random N log(rand)	less than nv + (np-1) nv nv	nv x np nv x np
Consideration for vectorization	conditional exit to select a process $\Sigma wgt[i] < u$	if test to select d and a physics process
implementation	holistic approach with a fixed number of processes	modularized (OO) design

Remarks

- Statically, explicit sampling = inclusive sampling
  - validation for passage through inhomogeneous materials (Geant4 vs. GeantV inclusive sampling?)
  - Independent of process history?
- Cons and Pros
  - performance-wise: implicit
  - flexibility-wise: explicit
  - vectorization-wise: explicit
- Performance, accuracy, portability

Status

- Implemented alternative secondary particle sampling (alias, inverse CDF) for
  - electron models
    - Bremstraahlung (Seltzer Berger)
    - Ionization (Moller Bhabha)
  - photon models
    - Compton (Klein Nishina)
- Working on vectorization schema for the mean free path analysis: need to converge
  - tracking strategies (inclusive vs. explicit)
  - simd layers (outer vs. inner loop)

Toward HP Physics Library

- Efficient random number handling and data layout
- More task decomposition
- Adopt the template approach (portability)
- Need more troops

Backup

What to vectorize?

- Ex1: do-while loop

```
xo = f(Z);                       // common data
for(i = 0; i < ntracks ; ++i) {  // for the problem-size
  yo = g((track.E)[i],xo);       // for a given constraint
\end{verbatim}
{\color{magenta} \begin{verbatim}
  do {                           // sample x and y
    x[i] = h(r1);
    y[i] = s(x[i],r2);
  } while (y[i] < yo);
}
```

- Alternative: alias method for sampling from p.d.f (differential cross sections)

What to vectorize?

- Ex2: conditional exit to select a physics process

```
// np = number of process
// w[np] = relative weight of cross sections
int ip = np;
double w = 0;
double fu = rand()
for (int i=0 ; i < np-1 ; ++i) {
  w += w[i];
  if (w < fu) { ip = i;  break; }
}
```

- Alternative: alias method with the normalized weight of cross sections (physics processes)

What to vectorize?

- Ex3: conditional exit to select a physics process

```
// np = number of process
// w[np] = relative weight of cross sections
int ip = np;
double w = 0;
double fu = rand()
for (int i=0 ; i < np-1 ; ++i) {
  w += w[i];
  if (w < fu) { ip = i;  break; }
}
```

- Alternative: alias method with the normalized weight of cross sections (physics processes)

What to vectorize?

- Ex4: if-test to select a random atom

```
// np = number of processes
 // s[np] = proposed step lengths
 int ip;
 double s = 0;
 double tmp = MAX;
 for(int i=0 ; i < np ; ++i) {
   if(s[i] < tmp) { s = s[i];  tmp = s; }
 }
```

- Alternative: mask