NLO event generation with the $(MC)^3$ sampling algorithm

Rene Poncelet[1]

Institut für Theoretische Teilchenphysik und Kosmologie

2016-03-01

[1]in cooperation with S. Schumann (Uni Göttingen), K. Kröninger (TU Dortmund)

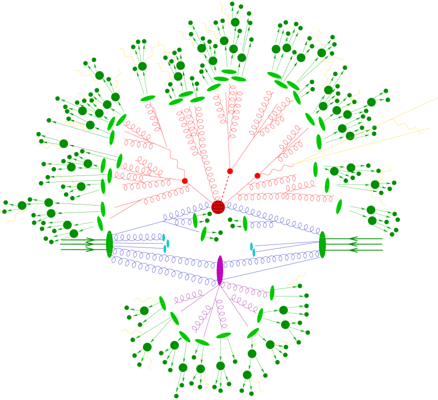
Event Generation

Factorization

- Hard-Process
- Parton-Shower, Resummation
- PDF, Hadronization, Underlying Event, ...

Phase Space Sampling

- Generation of (un)weighted events
- State of the art: Multi-Channel Importance Sampling
- Phase Space Mapping



Multi-Channel Importance Sampling

$$\int f(x)\mathrm{d}x = \int \frac{f(x)}{p(x)} p(x)\mathrm{d}x$$
$$= \sum_{i=1}^{m} \alpha_i \int \frac{f(x)}{p(x)} \mathrm{d}P_i(x)$$

Various peaks $\rightarrow$ different channels $p_i(x)$ with weight α_i: $p(x) = \sum \alpha_i p_i(x)$

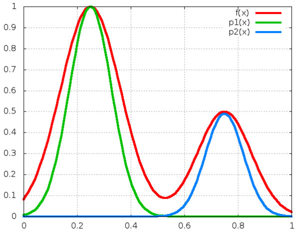
Hit and Miss algorithm

Assume: $M \cdot p(x) \geq f(x)\ \forall x$, $p(x)$ probability density

1. Draw X according to $P(X)$
2. Accept X with probability $\frac{f(X)}{Mp(X)}$
3. Return to step 1.

Mis-mappings

$f(x)/Mp(x) \ll 1$
$\rightarrow$ inefficient sampling



Markov Chains - Metropolis Hasting algorithm

Markov Property

- Given state X_0 ($\in \Omega$)
- Choose next state $X_i = K(X_{i-1})$ according to *transition kernel* K

Requirements on K:

- K is ergodic
- K is balanced, e.g. $p(X_1)K(X_1 \to X_2) = p(X_2)K(X_2 \to X_1)$
- K is independent of i (*time-homogeneous*)

Metropolis-Hasting

1. current state X_n
2. Generate randomly a proposal point Y
3. Accept the proposal point Y with probability $\alpha(X_n, Y) = \min\left[1, \frac{f(Y)}{f(X_n)}\right]$. If proposal point is accepted, $X_{n+1} = Y$, otherwise $X_{n+1} = X_n$.
4. Return to step 2.

(MC)^3 Sampling Algorithm

- New Monte Carlo Sampling algorithm [Willenberg et al., Comput.Phys.Commun. 186 (2015) 1-10]

Linear combination of two transition-kernels

$$K_{(\mathrm{MC})^3} = \beta K_{\mathrm{IS}} + (1-\beta) K_{\mathrm{MH}}$$

IS-Kernel K_{IS}:

- PS-Points from IS
- with acceptance prob. $\alpha_{IS} = \min\left[1, \frac{f(y)p(x)}{f(x)p(y)}\right]$

MH-Kernel K_{MH}:

- PS-Points via local symmetric variation
- with acceptance prob. $\alpha_{MH} = \min\left[1, \frac{f(y)}{f(x)}\right]$

- Burn-In phase $\rightarrow$ adjustment of local variation width
- Lag $\rightarrow$ reduction of autocorrelation effects

$(\mathrm{MC})^3$ Implementation in SHERPA

Implementation in SHERPA framework

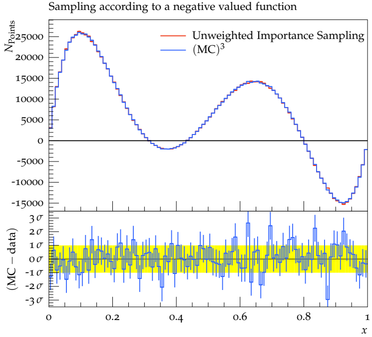
- Local phase space mapping with RamboDiet[Plätzer,arXiv:1308.2922]
- Automated Burn-In handling
- Sampling with arbitrary number of phase spaces
- Extended for NLO calculations

```
  ____
 /   /\   (MC)^3
/___/  \  Multi  Channel
\   \  /  Markov Chain
 \___\/   Monte  Carlo
```

NLO Event Generation with $(MC)^3$

- Various final-state multiplicities, i.e. σ_R and σ_V
  $\rightarrow$ multi-engine mode, selection via channel weights (only IS-kernel)
- Negative valued integrands, loops and counter-terms
  $\rightarrow$ Sampling of the modulus, bookkeeping of the sign

Example:



Validation

Markov-Chains naturally incorporate autocorrelation effects
$\rightarrow$ Idea: Compare $(\mathrm{MC})^3$ with IS samples

Estimators

$$\chi_i = \frac{\omega_{1,i} - \omega_{2,i}}{\sqrt{\sigma_{1,i}^2 + \sigma_{2,i}^2}}$$

$$\bar{\chi} = \frac{1}{N_{\mathrm{Bins}}} \sum_i^{N_{\mathrm{Bins}}} \chi_i$$

$$RMS = \sqrt{\frac{1}{N_{\mathrm{Bins}}} \sum_i^{N_{\mathrm{Bins}}} \chi_i^2 - \bar{\chi}^2}$$

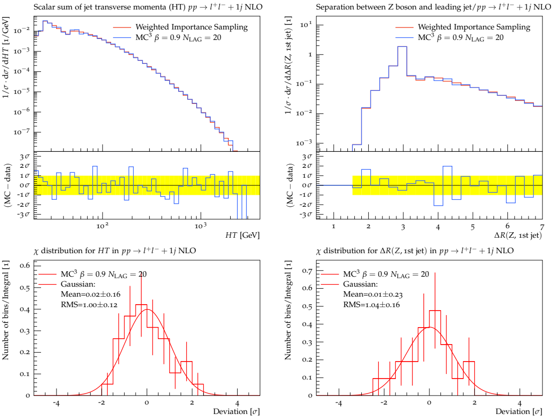
$\chi_i \sim \mathcal{N}(0,1)$ for statistical independent events

Criteria

- $\bar{\chi}$ compatible with 0 within 1.5 standard deviations
- *RMS* compatible with 1 within 1.5 standard deviations

$\rightarrow$ "Statistically compatible"

Validation
Example



"Quality" strongly process (multiplicity) dependent: A priori choice of N_{LAG} not ideal

↓

Dynamic lag choice

based on the correlation length for $\beta = 1$

Performance Measurement

$$\text{Time Gain} = \frac{\text{Time to generate sample with unweighted IS}}{\text{Time to generate sample with } (\text{MC})^3}$$

Process $pp \to$	Gain factor LO	NLO
$l^+l^- + 1j$		19.1
$l^+l^- + 2j$	2.4	106.0
$l^+l^- + 3j$	17.2	
$l^+l^- + 4j$	8.6	
$t\bar{t}$	1.0	1.3
$t\bar{t} + j$	3.1	86.9
$t\bar{t} + 2j$	7.7	
$t\bar{t}H$	1.0	1.7
$t\bar{t}H + j$	9.4	36
$t\bar{t}H + 2j$	20.0	

Process $pp \to$	Gain factor LO	NLO
W^+W^-	1	1.8
$W^+W^- + j$	1.8	13.1
$W^+W^- + 2j$	15.9	
WZ	0.4	0.7
$WZ + j$	1.0	36
$WZ + 2j$	5.1	
ZZ	1.8	0.9
$ZZ + j$	2.0	6.0
$ZZ + 2j$	7.6	
$H + j$	3.3	43.5
$H + 2j$	2.2	31.6
$H + 3j$	4.9	

Conclusion and Outlook

Conclusion

- Validated and tested within SHERPA framework for LO and NLO processes
- Setup dependent time-gain
  - $\mathcal{O}(10)$ for high multiplicities (>3 particles)
  - up to $\mathcal{O}(100)$ for NLO

Outlook

- Future SHERPA release
- Paper

Thank you for your attention.

Backup

NLO Event Generation

$$\sigma = \sigma_{LO} + \sigma_{NLO} \text{ with } \sigma_{NLO} = \sigma_R + \sigma_V = \int_{n+1} \mathrm{d}\sigma_R + \int_n \mathrm{d}\sigma_V$$

- σ_R & σ_V separately IR-divergent (after Renormalization) (for $\epsilon \to 0$ in dimensional regulation with $d = 4 - 2\epsilon$)
- $\to$ Subtraction-Method (Catani-Seymour-scheme)

$$\sigma^{NLO} = \int_{m+1} \left[\mathrm{d}\sigma^R - \mathrm{d}\sigma^A\right] + \int_{m+1} \mathrm{d}\sigma^A + \int_m \mathrm{d}\sigma^V$$

$$\sigma^{NLO} = \int_{m+1} \left[\mathrm{d}\sigma^R - \mathrm{d}\sigma^A\right]_{\epsilon=0} + \int_m \left[\mathrm{d}\sigma^V + \int_1 \mathrm{d}\sigma^A\right]_{\epsilon=0}$$

- $\int_{m+1}$ and $\int_m$ finite in $d = 4$ dimensions

Processes:

- $pp \rightarrow l^+ l^- + 0, 1, 2(, 3, 4)j$
- $pp \rightarrow VV + 0, 1(, 2)j$
- $pp \rightarrow t\bar{t} + 0, 1(, 2)j$

Parameter:

- $\beta \in [0.5, 0.9]$
- $N_{\text{LAG}} = 1, 10, 20$

Result:

- $\bar{\chi} \approx 0$ always
- $RMS \nearrow$ for $\beta \searrow$
- $RMS \nearrow$ for $N_{\text{LAG}} \searrow$
- $RMS \nearrow$ for number of $j \nearrow$

"Quality" strongly process (multiplicity) dependent: A priori choice of N_{LAG} not ideal

$\downarrow$

Dynamic lag choice

based on the correlation length for $\beta = 1$