Improved Phase Space Treatment of Massive Multi-Particle Final States

Borut Paul Krševan

Jozef Stefan Institute

Ljubljana, Slovenija

- The presentation is based on the paper **Eur. Phys. J. C 439-450 (2005)**

Overview

- With increasing energy the complexity of processes rises along with our expectations:
  - Many physics processess (signal as well as background).
  - The number of Feynman diagrams increases to large numbers.
  - We are at the threshold of (possibly new) heavy/massive particle productions.

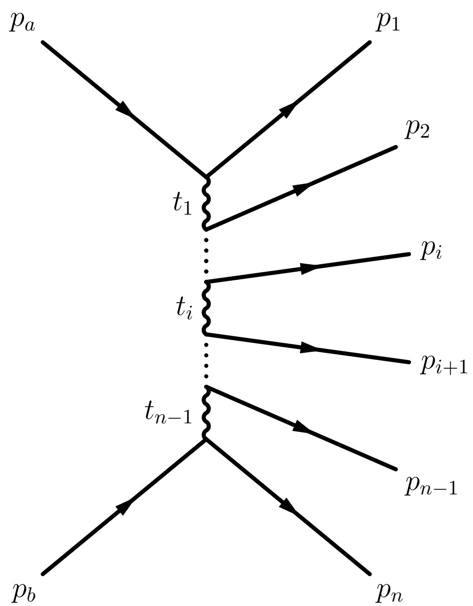
- Consequences:
  - Complex particle topologies.
  - Masses of final state particles not negligible.

- Requirements on the Monte-Carlo simulations of physics processes:
  - Automatise the MC simulation/generation for n particles in the final state.
  - Make it as efficient as possible:
    - Efficient MC unweighing procedure = event generation.
    - Every trial takes time.
    - Efficiently cover the phase space to minimise the cross-section variance and/or maximise the unweighing efficiency.

Legacy

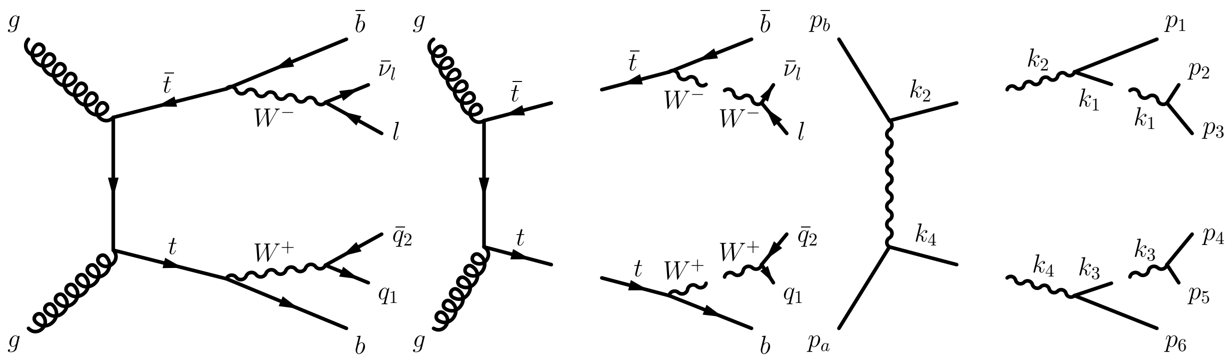
- LEP and before:
  - ➤ Matrix elements of the process calculated for particular cases.
  - ➤ Phase space sampling techniques: the buzzword is importance sampling.
    - ➔ Generate events in the n-body phase space Φ_n according to a distribution $g(\Phi_n)$ that closely matches the differential cross-section $g(\Phi_n) \sim \frac{d\sigma}{d\Phi_n}$.
    - ➔ Each event has a weight $w = \frac{1}{g(\Phi_n)} \cdot \frac{d\sigma}{d\Phi_n}$.
    - ➔ The event is accepted (unweighed) with the probability $\frac{w}{w_{\mathsf{max}}}$.
    - ➔ Ideally of course $g(\Phi_n) = \frac{d\sigma}{d\Phi_n}$, $w = \text{const}$.

- There are two directions in constructing $g(\Phi_n)$:
  - ➔ Adaptive algorithms like VEGAS (choice of variables an issue).
    - ➔ G.P. Lepage, J. Comput. Phys. **27** (1978) 192.
  - ➔ multi-channel importance sampling, each channel describes a certain topology:
    (NEXT)EXCALIBUR a representative example,
    each $2 \rightarrow 4$ topology constructed by hand.
    A difficult issue multi-peripheral (t-channel) topologies.
    - ➔ F. A. Berends, C. G. Papadopoulos and R. Pittau, hep-ph/0011031.



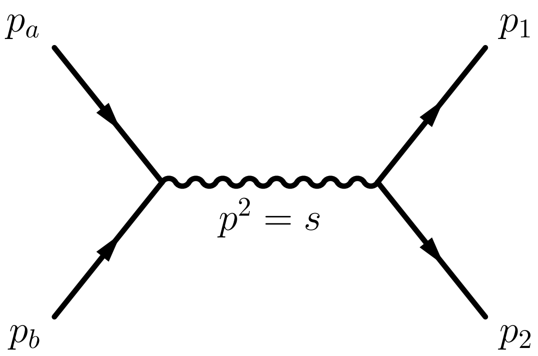
Today

- High quality automatised (leading order) matrix element calculations: MADGRAPH a good example.
  - ➔ T. Stelzer and W. F. Long, Comput.Phys.Commun. **81** (1994) 357.
- Automatised phase space description trailing behind in quality and complexity.
- Prerequisites:
  - ➤ A generic description of any topology (Feynman diag. based) with massive final state particles.
    - ➔ What one would like to do is to split the phase space sampling with any event topology into manageable pieces = modules.



- As it turns out a lot of it has already been done in the '60 (!) by K. Kajantie and E. Byckling.
  - ➔ E. Byckling and K. Kajantie, Nucl. Phys. **B9** (1969) 568.
  - ➤ Recursive expressions to split the n-body phase space into smaller subsets!
  - ➤ Just needs some modifications like adding importance sampling etc...

The s-type Branching Algorithms



- The 2 body case well known. The Phase-space integral (written in Lorentz invariant form):

$$\Phi_2(s, m_1, m_2) = \int d^4p_1 d^4p_2 \delta(p_1^2 - m_1^2)\delta(p_2^2 - m_2^2)\delta^4(p - p_1 - p_2)\Theta(p_1^0)\Theta(p_2^0)$$

- translates in the CMS of the two particles into:

$$\Phi_2(s, m_1, m_2) = \frac{\sqrt{\lambda(s, m_1^2, m_2^2)}}{8s} \int d\Omega_1^*$$

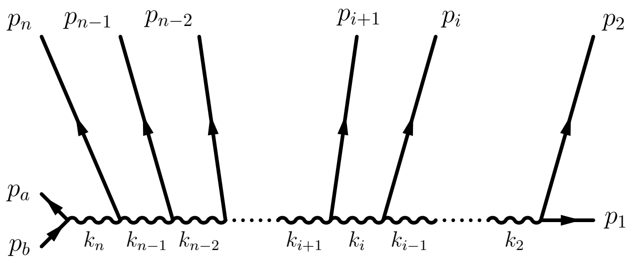
- with the Lorentz invariant function:

$$\lambda(s, m_1^2, m_2^2) = (s - (m_1 + m_2)^2)(s - (m_1 - m_2)^2)$$

- describing the threshold behavior:

$$\sqrt{s} \geq (m_1 + m_2)$$

The s-type Branching Algorithms



$$k_i = \sum_{j=1}^{i} p_j = (k_i^0, \vec{k}_i); \quad M_i^2 = k_i^2$$

- Lets start with the n-body phase space:

$$\Phi_n(\hat{s}, m_1, m_2, \ldots, m_n) = \int \delta^4\left((p_a + p_b) - \sum_{i=1}^{n} p_i\right) \prod_{i=1}^{n} d^4 p_i \delta(p_i^2 - m_i^2)\Theta(p_i^0)$$

- and insert two identities:

$$1 = \int dM_{n-1}^2 \delta(k_{n-1}^2 - M_{n-1}^2)\Theta(k_{n-1}^0) \qquad 1 = \int d^4 k_{n-1} \delta^4(p - k_{n-1} - p_n)$$

- After some integrating we get a recursion relation:

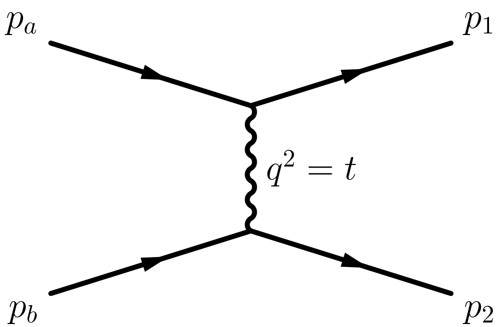
$$\Phi_n(M_n^2, m_1, m_2, \ldots, m_n) = \int_{(\sum_{i=1}^{n-1} m_i)^2}^{(M_n - m_n)^2} dM_{n-1}^2 \frac{\sqrt{\lambda(M_n^2, M_{n-1}^2, m_n^2)}}{8M_n^2} \int d\Omega_n^* \Phi_{n-1}(M_{n-1}^2, m_1, m_2, \ldots, m_{n-1})$$

- The same can be achieved by grouping particles into two arbitrary sets :

$k_l^2 = \sum_{i=1}^{l} p_i$ and $\tilde{k}_l^2 = \sum_{j=l+1}^{n} p_j$

- We can split the chain anywhere and walk in both directions

The t-type Branching Algorithms



$$t = q^2 = (p_a - p_1)^2$$

- The 2 body case also trivial but a tad more involved. In the CMS of the two particles we get:

$$\Phi_2(s, m_1, m_2) = \frac{1}{4\sqrt{\lambda(s, m_a^2, m_b^2)}} \int_{t^-}^{t^+} dt \int_0^{2\pi} d\varphi^*$$

- where the limits:

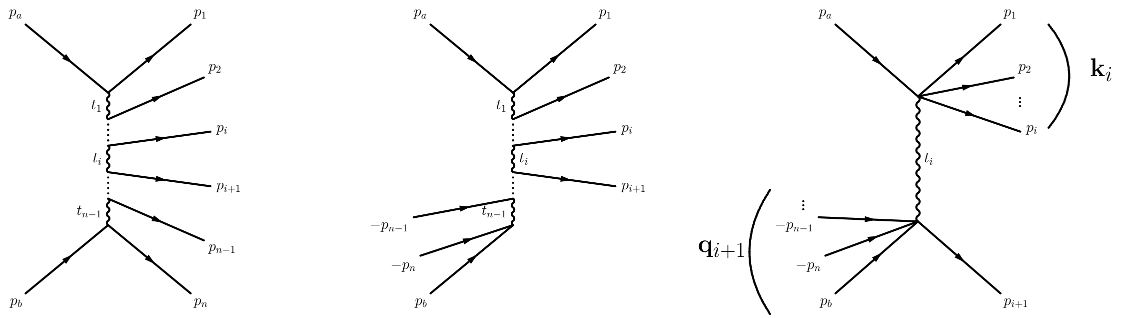
$$t^{\pm} = m_a^2 + m_1^2 - \frac{(s+m_a^2-m_b^2)(s+m_1^2-m_2^2)}{2s} \mp \frac{\sqrt{\lambda(s,m_a^2,m_b^2)\lambda(s,m_1^2,m_2^2)}}{2s}$$

- can in general be obtained from **the basic four-particle kinematic function** using the condition:

$$G(s, t, m_2^2, m_a^2, m_b^2, m_1^2) \leq 0$$

$$G(x, y, z, u, v, w) = -\frac{1}{2} \begin{vmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & v & x & z \\ 1 & v & 0 & u & y \\ 1 & x & u & 0 & w \\ 1 & z & y & w & 0 \end{vmatrix}$$

The t-type Branching Algorithms



$$k_i = \sum_{j=1}^{i} p_j \quad M_i^2 = k_i^2$$

$$q_i = p_b - \sum_{j=i+1}^{n} p_j = p_a - k_i$$

$$q_i^2 = t_i \quad q_n^2 = t_n = m_b^2$$

- In this case the n-body phase space:

$$\Phi_n(\hat{s}, m_1, m_2, \ldots, m_n) = \int \delta^4 \left((p_a + p_b) - \sum_{i=1}^{n} p_i\right) \prod_{i=1}^{n} d^4 p_i \delta(p_i^2 - m_i^2)\Theta(p_i^0)$$

- translates into the recursion relation:

$$\Phi_n(M_n^2, m_1, m_2, \ldots, m_n) = \int_{(\sum_{i=1}^{n-1} m_i)^2}^{(M_n - m_n)^2} \frac{dM_{n-1}^2}{4\sqrt{\lambda(M_n^2, m_a^2, t_n)}} \int_0^{2\pi} d\varphi_n^* \int_{t_{n-1}^-}^{t_{n-1}^+} dt_{n-1} \Phi_{n-1}(M_{n-1}^2, m_1, m_2, \ldots, m_{n-1})$$

- With the $t_{n-1}^{\pm}$ limits again given by **the basic four-particle kinematic function** using the condition:

$$G(M_{i+1}^2, t_i, m_{i+1}^2, m_a^2, t_{i+1}, M_i^2) \leq 0$$

- Again, we can split the chain anywhere and walk in both directions
- Using both types of branchings we can describe/modularise any topology!

Describing the peaking behavior of the differential cross-section

- With the phase space transformed the way we want the dominant peaks come from the propagators in the (squared) Matrix element.

- In general we can describe the resonant and non-resonant propagators with:

$$f_{\mathrm{NR}}(s) \sim \frac{1}{s^{\nu}} \qquad f_{\mathrm{R}}(s) \sim \frac{\sqrt{s}}{(s-M^2)^2+M^2\Gamma^2}$$

- With the inclusion of the threshold behaviour in the s-channel topologies we get:

$$f_{\mathrm{NR}}(s) = \frac{\sqrt{\lambda(s,m_a^2,m_b^2)}}{s} \cdot \frac{1}{s^{\nu}} = \frac{\sqrt{\lambda(s,m_a^2,m_b^2)}}{s^{\nu+1}}$$

$$f_{\mathrm{R}}(s) = \frac{\sqrt{\lambda(s,m_a^2,m_b^2)}}{s} \cdot \frac{\sqrt{s}}{(s-M^2)^2+M^2\Gamma^2} = \frac{\sqrt{\lambda(s,m_a^2,m_b^2)}}{\sqrt{s}\cdot((s-M^2)^2+M^2\Gamma^2)}$$

- The goal is to produce unitary (importance) sampling algorithms according to the above functions.

- This turns out to be non-trivial!

Short reminder: Importance sampling

- If one wants to generate events distributed according to a function $f(x)$ in an **unitary way** (every trial succeeds), one has to solve for x:

$$\int_{x_-}^{x} f(x)\,dx = r \cdot \int_{x_-}^{x_+} f(x)\,dx,$$

with r a pseudo random number $r \in [0,1]$.

- In the case when the integral of the function $F(x) = \int_{x_-}^{x} f(x)\,dx$ is an analytic function and has a known inverse $\mathrm{F}^{-1}(\mathrm{x})$, one can construct explicit unitary prescriptions by:

$$x = F^{-1}\left(r \cdot [F(x_+) - F(x_-)] + F(x_-)\right)$$

- In the cases the integral can not be inverted, the prescription can directly be transformed into a zero-finding request.

- since both the integral and the first derivative (i.e. the sampling function and its cumulant) are known, the Newton-Rhapson method is chosen as the optimal one for root finding:

$$g(x) = \left\{\int_{x_-}^{x} f(x)dx - r \cdot \int_{x_-}^{x_+} f(x)dx\right\} = 0, \qquad g'(x) = \frac{d}{dx}\left\{\int_{x_-}^{x} f(x)dx - r \cdot \int_{x_-}^{x_+} f(x)dx\right\} = f(x)$$

Integral of the resonant phase-space suppressed propagator yields a rather non-trivial result:

$$\int_{(m_a+m_b)^2}^{s} f_{\mathrm{R}}(s)\, ds = \int_{(m_a+m_b)^2}^{s} \frac{\sqrt{\lambda(s, m_a^2, m_b^2)}\, ds}{\sqrt{s}\cdot((s-M^2)^2+M^2\Gamma^2)}$$

$$= \int_{a}^{s} \frac{\sqrt{(s-a)(s-b)}\, ds}{\sqrt{s}\cdot((s-M^2)^2+M^2\Gamma^2)}$$

$$= \frac{1}{\sqrt{-b}\,\Gamma\, M^2} \times \frac{-2\, i\, a\, b\, \Gamma}{(\Gamma^2+M^2)}$$

$$\times \left\{ \mathbf{F}\left[i \operatorname{arcsinh}(\frac{\sqrt{-b}}{\sqrt{a}}), \frac{a}{b}\right] - \mathbf{F}\left[i \operatorname{arcsinh}(\frac{\sqrt{-b}}{\sqrt{s}}), \frac{a}{b}\right] \right.$$

$$+ \ (i\,\Gamma + M)\,(a + i\,(\Gamma + i\,M)\,M)\,(b + i\,(\Gamma + i\,M)\,M)\,\mathbf{\Pi}\left[\frac{M\,(-i\,\Gamma + M)}{b}, i \operatorname{arcsinh}(\frac{\sqrt{-b}}{\sqrt{a}}), \frac{a}{b}\right]$$

$$+ \ (\Gamma + i\,M)\,(b + (-i\,\Gamma - M)\,M)\,(i\,a + (\Gamma - i\,M)\,M)\,\mathbf{\Pi}\left[\frac{M\,(i\,\Gamma + M)}{b}, i \operatorname{arcsinh}(\frac{\sqrt{-b}}{\sqrt{a}}), \frac{a}{b}\right]$$

$$- \ (i\,\Gamma + M)\,(a + i\,(\Gamma + i\,M)\,M)\,(b + i\,(\Gamma + i\,M)\,M)\,\mathbf{\Pi}\left[\frac{M\,(-i\,\Gamma + M)}{b}, i \operatorname{arcsinh}(\frac{\sqrt{-b}}{\sqrt{s}}), \frac{a}{b}\right]$$

$$\left. - \ (\Gamma + i\,M)\,(b + (-i\,\Gamma - M)\,M)\,(i\,a + (\Gamma - i\,M)\,M)\,\mathbf{\Pi}\left[\frac{M\,(i\,\Gamma + M)}{b}, i \operatorname{arcsinh}(\frac{\sqrt{-b}}{\sqrt{s}}), \frac{a}{b}\right] \right\}$$

- $\mathbf{F}[\varphi, \mathrm{k}]$ and $\mathbf{\Pi}[\varphi, \mathrm{k}, \mathrm{n}]$ are the Legendre's incomplete elliptic integrals of the second and third kind with complex arguments. Had to be coded from scratch!

Integral of the non-resonant phase-space suppressed propagator yields a similarly non-trivial result:

$$\int_{(m_a+m_b)^2}^{s} f_{\mathrm{NR}}(s)\, ds = \int_{(m_a+m_b)^2}^{s} \frac{\sqrt{\lambda(s, m_a^2, m_b^2)}\, ds}{s^{\nu+1}}$$

$$= \frac{1}{2\sqrt{1-\frac{s}{a}}\,\nu}\left\{ \frac{-2\sqrt{(a-s)\ (b-s)}\,\mathbf{F_1}\left[-\nu, -\left(\frac{1}{2}\right), -\left(\frac{1}{2}\right), 1-\nu, \frac{s}{a}, \frac{s}{b}\right]}{s^{\nu}\sqrt{1-\frac{s}{b}}} + \frac{\sqrt{\pi}\sqrt{(-a+b)\ (a-s)}\,\mathbf{\Gamma}\left[1-\nu\right]\,\mathbf{F}\left[-\nu, -\left(\frac{1}{2}\right), \frac{3}{2}-\nu, \frac{a}{b}\right]}{a^{\nu}\sqrt{1-\frac{a}{b}}\,\mathbf{\Gamma}\left[\frac{3}{2}-\nu\right]} \right\}$$

- The function $\mathrm{F}[\alpha, \beta, \gamma, \mathrm{x}]$ is the Gauss Hypergeometric function and the $\mathrm{F}_1[\alpha, \beta, \beta', \gamma, \mathrm{x}, \mathrm{y}]$ is the two-parameter (Appell) Hypergeometric function.

- Explicit numerical calculation of the integral turns out to be faster, a 50-point Gauss-Legendre quadrature with $\sqrt{s}$ weight function was used.

Example of the implementation: AcerMC 2.x Monte-Carlo generator

- A Monte-Carlo generator of background processes for searches at **ATLAS/LHC**.
- Matrix element coded by **MADGRAPH/HELAS**
  - ➔ T. Stelzer and W. F. Long, Comput.Phys.Commun. **81** (1994) 357.
- Phase space sampling done by native **AcerMC** routines:

  ⊕ Each channel topology constructed from the t-type and s-type modules and sampling functions described in this talk. The event topologies derived from modified **MADGRAPH/HELAS** code.

  ⊕ **multi-channel approach**
  - ➔ J.Hilgart, R. Kleiss, F. Le Dibider, Comp. Phys. Comm. **75** (1993) 191.
  - ➔ F. A. Berends, C. G. Papadopoulos and R. Pittau, hep-ph/0011031.

  ⊕ additional **ac-VEGAS** smoothing
  - ➔ G.P. Lepage, J. Comput. Phys. **27** (1978) 192.

- ac−VEGAS Cell splitting in view of maximal weight reduction based on function:

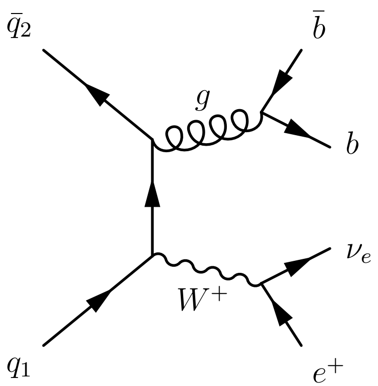
$$< F >_{\rm cell} = \left(\Delta_{\rm cell} \cdot {\rm wt}_{\rm cell}^{\rm max}\right) \cdot \left\{ 1 - \frac{<{\rm wt}_{\rm cell}>}{{\rm wt}_{\rm cell}^{\rm max}} \right\}$$

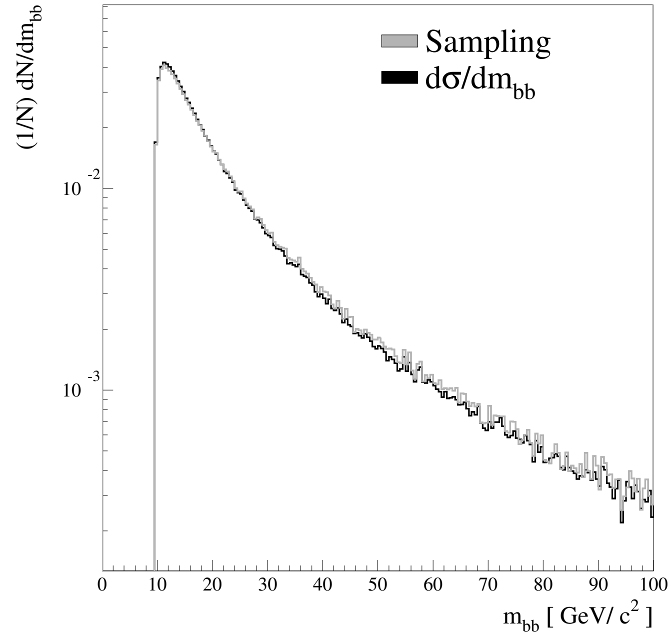
- ac−VEGAS logic in this respect analogous to FOAM:

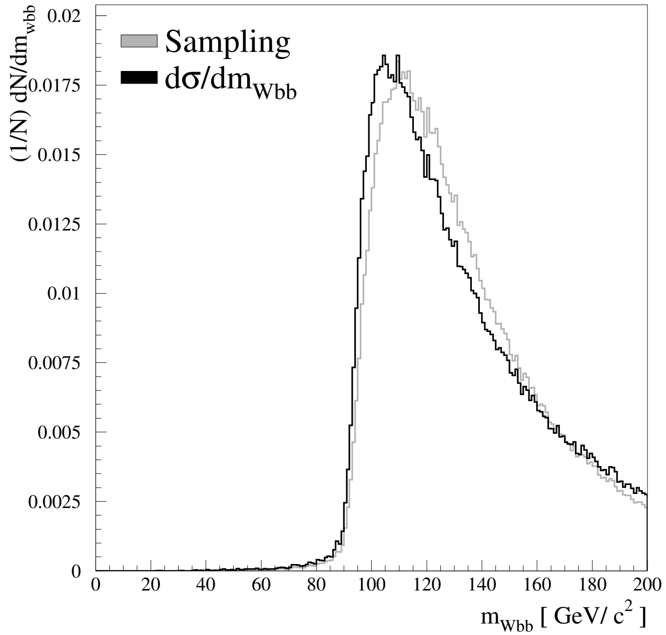
➔ S. Jadach, Comput. Phys. Commun. **130** (2000) 244.

Example of 2 → 4 processes: $u\bar{d} \rightarrow W^{+}g^{*} \rightarrow l^{+}\nu_{l}b\bar{b}$, pp @ 14 TeV

- Examples of invariant mass distributions obtained with **AcerMC**







- Some variances and unweighing efficiencies obtained using standard **AcerMC 1.4** and new **AcerMC 2.0** phase space sampling.

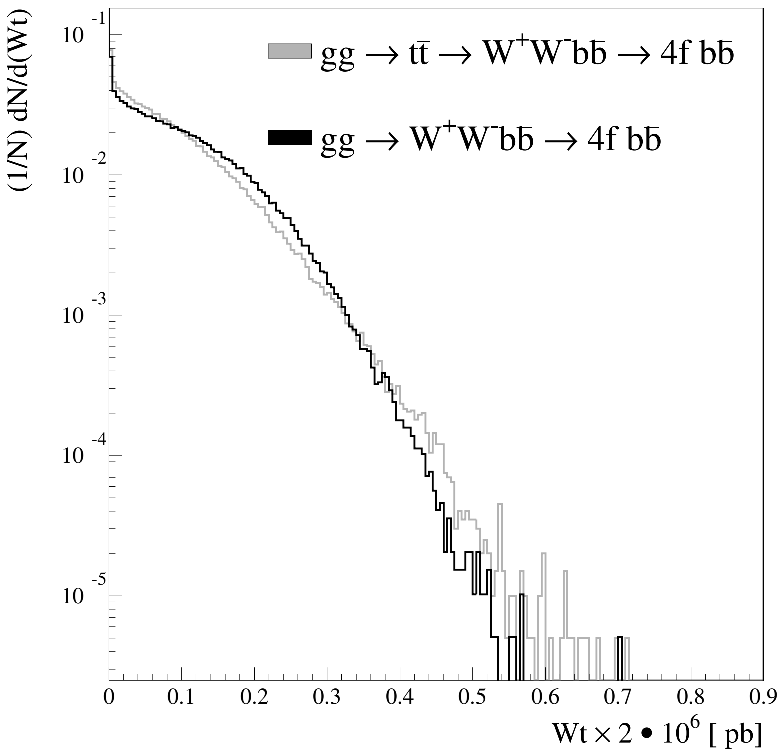
Process	`AcerMC 2.0` V_σ [pb^2]	`AcerMC 1.4` V_σ [pb^2]	`AcerMC 2.0` ϵ	`AcerMC 1.4` ϵ
$gg \rightarrow Z/(\rightarrow \ell\ell)b\bar{b}$	$0.129 \cdot 10^{-2} \pm 0.52 \cdot 10^{-5}$	$0.159 \cdot 10^{-2} \pm 0.61 \cdot 10^{-5}$	37%	33%
$q\bar{q} \rightarrow W(\rightarrow \ell\nu)b\bar{b}$	$0.390 \cdot 10^{-2} \pm 0.15 \cdot 10^{-4}$	$0.533 \cdot 10^{-2} \pm 0.18 \cdot 10^{-4}$	35%	33%
$gg \rightarrow t\bar{t}b\bar{b}$	$0.522 \cdot 10^{-4} \pm 0.19 \cdot 10^{-6}$	$0.972 \cdot 10^{-4} \pm 0.44 \cdot 10^{-6}$	36%	20%

Example of 2 → 6 processes: $gg \to b\bar{b}W^+W^- \to b\bar{b}\ell\bar{\nu}_\ell\bar{\ell}\nu_\ell$

- The process cross-sections and variances with their uncertainties and unweighing efficiencies as obtained for two sample $2 \to 6$ processes implemented in **AcerMC 2.0** Monte–Carlo generator.

AcerMC 2.0 Process		σ [pb]	V_σ [pb^2]	ϵ
$gg \to t\bar{t} \to b\bar{b}W^+W^- \to b\bar{b}\ell\bar{\nu}_\ell\bar{\ell}\nu_\ell$	(3 Feyn./2 sampl. chan.)	4.49	$0.80 \cdot 10^{-4} \pm 0.39 \cdot 10^{-6}$	14%
$gg \to b\bar{b}W^+W^- \to b\bar{b}\ell\bar{\nu}_\ell\bar{\ell}\nu_\ell$	(31 Feyn./13 sampl. chan.)	4.77	$0.77 \cdot 10^{-4} \pm 0.29 \cdot 10^{-5}$	17%

- Example of the weight distributions obtained with the two processess.



- Bottom line is: It Works!