Automation of Dipole Subtraction Method in MadGraph

Nicolas Greiner
in collaboration with R.Frederix,T.Gehrmann

HP2.2 Buenos Aires, 07.10.2008

Outline

LO event generator tools:

- PYTHIA [Sjoestrand,Mrenna,Skands]
- HERWIG/HERWIG++ [Marchesini,Webber],[Baehr et al.]
- MadGraph/MadEvent [Stelzer,Long],[Maltoni,Stelzer],[Alwall et al.]
- CompHep/CalcHep [Boos et al.],[Pukhov]
- SHERPA [Gleisberg et al.]
- WHIZARD [Kilian,Ohl,Reuter]
- ALPGEN [Mangano,Moretti,Piccinini,Pittau,Polosa]
- HELAC [Kanaki,Papadopoulos]
- ...

NLO calculation programs:

- MCFM [Campbell,Ellis]
- NLOJET++ [Nagy]
- MC@NLO [Frixione,Webber]
- POWHEG [Nason et al.]
- ...

Automation of loop calculations:

Enormous progress in recent years : Unitarity methods, recursion relations, generalized unitarity, OPP-method,twistor-inspired methods...

$\longrightarrow$ Packages like

- CutTools [Ossola,Papadopoulos,Pittau]
- BlackHat [Berger et al.]
- Rocket [Giele,Zanderighi]
- Golem [Binoth et al.]
- ...

Automation of subtraction methods:

Several algorithms for subtraction terms:

- Dipole subtraction [Catani,Seymour],[Catani,Dittmaier,Seymour,Trocsanyi]
- Residue subtraction [Frixione,Kunszt,Signer]
- Antenna subtraction

  [Kosower],[Campbell,Cullen,Glover],[Gehrmann-DeRidder,Gehrmann,Glover] ,[Daleo,Gehrmann,Maitre]

First automation of dipole subtraction in SHERPA [Gleisberg,Krauss] and TeVJet [Seymour,Tevlin] and attempts for external library interfaced with MadGraph. [Hasegawa,Moch,Uwer]

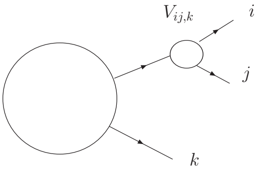
$\rightarrow$ No general tool available for arbitrary process and massive dipoles.

Dipole Subtraction Method: [Catani,Seymour] [Catani,Dittmaier,Seymour,Trocsanyi]

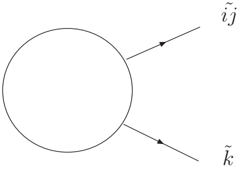
Find expressions $d\sigma^A$ for infrared singularities and subtract/add them

$$\Rightarrow \sigma^{NLO} = \int_{m+1} \left[\left(d\sigma^R \right) - \left(d\sigma^A \right) \right] + \int_m \left[d\sigma^V + \int_1 d\sigma^A \right]$$

- Dipoles contain all infrared singularities occuring in specific process.
- Cross sections for real emission and virtual corrections are finite and can be calculated independently.



$$\longrightarrow \quad \mathcal{D}_{ij,k} =$$



$$\mathcal{D}_{ij,k}\ (p_1, ..., p_{m+1}) = -\frac{1}{2p_i \cdot p_j}$$

$$\cdot\, {}_m\!< 1, .., \widetilde{ij}, .., \widetilde{k}, .., m+1 | \frac{\boldsymbol{T}_k \cdot \boldsymbol{T}_{ij}}{\boldsymbol{T}_{ij}^2}\ \boldsymbol{V}_{ij,k}\, | 1, .., \widetilde{ij}, .., \widetilde{k}, .., m+1 >_m \ .$$

with emitter i and spectator k and dipole splitting function $V_{ij,k}$.

$$\widetilde{p}_k^\mu = \frac{1}{1 - y_{ij,k}}\, p_k^\mu \ , \quad \widetilde{p}_{ij}^\mu = p_i^\mu + p_j^\mu - \frac{y_{ij,k}}{1 - y_{ij,k}}\, p_k^\mu, \quad y_{ij,k} = \frac{p_i p_j}{p_i p_j + p_j p_k + p_k p_i}.$$

Note: $p_i^\mu + p_j^\mu + p_k^\mu = \tilde{p}_{ij}^\mu + \tilde{p}_k^\mu$ and $\tilde{p}_{ij}^2 = \tilde{p}_k^2 = 0$.

MadGraph: [Stelzer,Long]

Type in process: e.g. **e+ e- $\longrightarrow$ u u~**

$\Rightarrow$ MadGraph provides a Fortran code that calculates $|\mathcal{M}|^2$ for a given phase space point, summed over colors and helicities.

MadEvent: [Maltoni,Stelzer]

- Takes MadGraph output and integrates over phase space.
- Event generator.

MadGraph/MadEvent public available:
http://madgraph.hep.uiuc.edu/

MadDipole: [Frederix,Gehrmann,NG]

Type in real emission process: e.g. **e+ e- $\rightarrow$ u u~ g**

$\Rightarrow$ Analogous to MadGraph, MadDipole returns Fortran code for:

- Matrixelement for real emission.
- All possible dipoles for all possible born processes.

Further information and download:
http://madgraph.hep.uiuc.edu/

$$\mathcal{D}_{ij,k} \sim {}_m\!< 1,..,\widetilde{ij},..,\widetilde{k},..,m+1| \frac{\boldsymbol{T}_k \cdot \boldsymbol{T}_{ij}}{\boldsymbol{T}_{ij}^2} \boldsymbol{V}_{ij,k} |1,..,\widetilde{ij},..,\widetilde{k},..,m+1>_m$$

Splitting function $V_{ij,k}$ is tensor in helicity space. $V_{ij,k} = V_{ij,k}^{\mu\nu}$

$\Rightarrow$ Need modification of color and helicity management.

1. Color management:

- Use already existing routines $\rightarrow$ fast and correct.
- Insert additional operators in existing color calculation. Note: $\boldsymbol{T}_k \cdot \boldsymbol{T}_{ij}$ connect bra and ket $\rightarrow$ need different labelling. $\rightarrow$ new routines for squaring. $\rightarrow$ large objects.

2. Helicity management:

$V_{ij,k} = V_{ij,k}^{\mu\nu}$ combines different helicity combinations.

$$
\begin{aligned}
\mathcal{D}_{ij,k} &\sim {}_m\langle 1, ...\tilde{ij}, ..., \tilde{k}, ..., m+1|_\mu \mathbf{V}_{ij,k}^{\mu\nu}\ {}_\nu|1, ...\tilde{ij}, ..., \tilde{k}, ..., m+1\rangle_m \\
&= {}_m\langle 1, ...\tilde{ij}, ..., \tilde{k}, ..., m+1|_{\mu'} \left(-g^{\mu'}_{\ \mu}\right) \mathbf{V}_{ij,k}^{\mu\nu} \left(-g_\nu^{\ \nu'}\right)\ {}_{\nu'}|1, ...\tilde{ij}, ..., \tilde{k}, ..., m+1\rangle_m \\
&= \sum_{\lambda_a,\lambda_b} {}_m\langle \ldots |_{\mu'}\ \epsilon^{*\mu'}(\lambda_b)\epsilon_\mu(\lambda_b)\mathbf{V}_{ij,k}^{\mu\nu}\ \epsilon^*_\nu(\lambda_a)\epsilon^{\nu'}(\lambda_a)\ {}_{\nu'}| \ldots \rangle_m \\
&= \sum_{\lambda_a,\lambda_b} {}_m\langle \ldots |_{\lambda_b}\ V(\lambda_b, \lambda_a)\ {}_{\lambda_a}| \ldots \rangle_m
\end{aligned}
$$

with $V(\lambda_b, \lambda_a) = \epsilon_\mu(\lambda_b)\ \mathbf{V}_{ij,k}^{\mu\nu}\ \epsilon^*_\nu(\lambda_a)$ and $\epsilon^\mu(\lambda)\ {}_\mu| \ldots \rangle_m = {}_\lambda| \ldots \rangle_m$.

Phase space restrictions

Subtraction only needed when approaching divergency. $\Rightarrow$ Cut away non-singular parts of phase space by additional parameter $\alpha \in [0,1]$.
[Nagy,Trocsanyi]

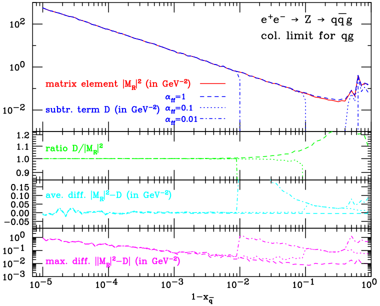
$$
\begin{aligned}
d\sigma_{ab}^{A} = \sum_{\{n+1\}} d\Gamma^{(n+1)}(p_a, p_b, p_1, \ldots, p_n + 1)\frac{1}{S_{\{n+1\}}} \\
\times \Bigg\{ \sum_{\substack{\text{pairs}\\ i,j}} \sum_{k \neq i,j} \mathcal{D}_{ij,k}(p_a, p_b, p_1, \ldots, p_{n+1}) F_J^{(n)}(p_a, p_b, p_1, .., \tilde{p}_{ij}, \tilde{p}_k, .., p_{n+1}) \Theta(y_{ij,k} < \alpha) \\
+ \sum_{\substack{\text{pairs}\\ i,j}} \Big[\mathcal{D}_{ij}^{a}(p_a, p_b, p_1, \ldots, p_{n+1}) F_J^{(n)}(\tilde{p}_a, p_b, p_1, .., \tilde{p}_{ij}, .., p_{n+1}) \Theta(1 - x_{ij,a} < \alpha) \quad + (a \leftrightarrow b) \Big] \\
+ \sum_{i \neq k} \Big[\mathcal{D}_{k}^{ai}(p_a, p_b, p_1, \ldots, p_{n+1}) F_J^{(n)}(\tilde{p}_a, p_b, p_1, .., \tilde{p}_k, .., p_{n+1}) \Theta(u_i < \alpha) + (a \leftrightarrow b) \Big] \\
+ \sum_{i} \Big[\mathcal{D}^{ai,b}(p_a, p_b, p_1, \ldots, p_{n+1}) F_J^{(n)}(\tilde{p}_a, p_b, \tilde{p}_1, \ldots, \tilde{p}_{n+1}) \Theta(\tilde{v}_i < \alpha) + (a \leftrightarrow b) \Big] \Bigg\} \ .
\end{aligned}
$$

$\rightarrow$ 4 parameters: `alpha_ff`, `alpha_fi`, `alpha_if`, `alpha_ii`, adjustable by user.

Massive particles

- Motivation: collinear radiation off massive particle finite, but source of possibly large logs.
- Finite dipoles put in separate routine `dipolsumfinite(...)`.
  Not evaluated by default but can be switched on if needed.
- Recover massless results in the limit of vanishing masses.

Check: In the limit $s_{ij} = p_i \cdot p_j \to 0$ dipoles approach matrixelement.



Ratio $|\mathcal{M}|^2 / \sum_{dipoles} \to 1$.
Difference integrable.

Package contains routine that checks all limits.

Further checks against MCFM: [Campbell,Ellis]

process	subprocesses
Drell-Yan (W)	$q\bar{q}' \to W^+(\to e^+\nu_e)g$ $qg \to W^+(\to e^+\nu_e)q'$
Drell-Yan (Z)	$q\bar{q} \to Z(\to e^+e^-)g$ $qg \to Z(\to e^+e^-)q$
Drell-Yan (Z+jet)	$q\bar{q} \to Z(\to e^+e^-)q'\bar{q}'$ $q\bar{q} \to Z(\to e^+e^-)q\bar{q}$ $q\bar{q} \to Z(\to e^+e^-)gg$ $q\bar{g} \to Z(\to e^+e^-)qg$ $g\bar{g} \to Z(\to e^+e^-)q\bar{q}$
top quark pair ($t\bar{t}$)	$q\bar{q} \to t(\to bl^+\nu_l)\bar{t}(\to \bar{b}l^-\bar{\nu}_l)g$ $qg \to t(\to bl^+\nu_l)\bar{t}(\to \bar{b}l^-\bar{\nu}_l)q$ $gg \to t(\to bl^+\nu_l)\bar{t}(\to \bar{b}l^-\bar{\nu}_l)g$
t-channel single top with massive b-quark	$gg \to t\bar{b}q\bar{q}'$ $qq' \to t\bar{b}q'q''$ $qq' \to t\bar{b}q'q''$ $qg \to t\bar{b}q'g$

Compare dipoles in single phase space points. No inconsistencies found.

$$\Rightarrow \sigma^{NLO} = \int_{m+1} \left[\left(d\sigma^R \right) - \left(d\sigma^A \right) \right] + \int_m \left[d\sigma^V + \int_1 d\sigma^A \right]$$

Fully automated integration over one-particle phase space would be more convient for user.
Phase space factorization:

$$d\phi(p_i, p_j, p_k; Q) = d\phi(\widetilde{p}_{ij}, \widetilde{p}_k; Q) \; \left[dp_i(\widetilde{p}_{ij}, \widetilde{p}_k) \right]$$

Integration over dipole:

$$\int \left[dp_i(\widetilde{p}_{ij}, \widetilde{p}_k) \right] \; \mathcal{D}_{ij,k}(p_1, \ldots, p_{m+1})$$

$$= -\, \mathcal{V}_{ij,k} \;\; {}_m\!< 1, .., \widetilde{ij}, .., \widetilde{k}, .., m+1 | \, \frac{\boldsymbol{T}_k \cdot \boldsymbol{T}_{ij}}{\boldsymbol{T}_{ij}^2} \, | 1, .., \widetilde{ij}, .., \widetilde{k}, .., m+1 >_m \; ,$$

with $\mathcal{V}_{ij,k} = \int \left[dp_i(\widetilde{p}_{ij}, \widetilde{p}_k) \right] \; \frac{1}{2p_i \cdot p_j} \;\; < \boldsymbol{V}_{ij,k} > \equiv \; \frac{\alpha_S}{2\pi} \frac{1}{\Gamma(1-\epsilon)} \left(\frac{4\pi\mu^2}{2\widetilde{p}_{ij}\widetilde{p}_k} \right)^{\epsilon} \mathcal{V}_{ij}(\epsilon)$

Several non-trivial details:

- Integrated splitting function with initial state particles contains distributions ,e.g.

$$\mathcal{V}_{qg}(x;\epsilon) = C_F\left[\left(\frac{2}{1-x}\ln\frac{1}{1-x}\right)_+ - \frac{3}{2}\left(\frac{1}{1-x}\right)_+ + \frac{2}{1-x}\ln(2-x)\right] + \delta(1-x)\left[\mathcal{V}_{qg}(\epsilon) - \frac{3}{2}C_F\right] + \mathcal{O}(\epsilon) \;,$$

with $\int_0^1 dx\, g(x)\,[\mathcal{V}(x)]_+ \equiv \int_0^1 dx\,[g(x)-g(1)]\,\mathcal{V}(x)$.

- $\Rightarrow$ Need to calculate $|\mathcal{M}|^2$ at x and at $x=1$.
- For massive particles also $\delta(x_+-x)$ and $(..)_{x_+}$ contributions with $x_+ = 1-4\frac{m_f^2}{Q^2}$ and $\int_0^1 \mathrm{d}x \left(f(x)\right)_{x_+} g(x) \equiv \int_0^1 \mathrm{d}x\, f(x)\Theta(x_+-x)\,[g(x)-g(x_+)]$.

- Inclusion of α-parameter leads to nontrivial dependence on α [Nagy,Trocsanyi],[Campbell,Ellis].
  New integrals required.
- Assume only one mass scale.
- Inclusion of pdf's.
- Checks involve sampling over pdf $\rightarrow$ more involved

Status:
Implementation in principle finished.
To do:
Extensive testing (e.g. with MCFM,α independence of result)

Conclusions

- Dipole subtraction formalism ensures finiteness of real emission terms and virtual corrections.
- MadDipole: Fully automated implementation of dipole formalism.
- Numerous checks to ensure correctness.
- Automated integration over one particle phase space desirable.
  $\longrightarrow$ In progress.
- Principle implementation done.
  Test against existing results.