# SingularPhasespace and helicity recycling in MadGraph

Kiran Ostrolenk
kiran.ostrolenk@manchester.ac.uk

University of Manchester




20th MCnet Meeting, my house, April 16$^{th}$, 2020

# Outline

# Matching algorithms

$$\mathrm{d}\sigma^{NLO} = \mathrm{d}\phi_n \left[B + V\right] + \mathrm{d}\phi_{n+1}\, R$$

- The terms labelled $S$ are called the Subtraction Dipoles
- They are defined to mimic the singular behaviour of $R$ and $V$
- When all integrands are finite a Monte Carlo simulation can be performed

# Matching algorithms

$$\begin{aligned} \mathrm{d}\sigma^{NLO} &= \mathrm{d}\phi_n \left[B + V\right] + \mathrm{d}\phi_{n+1}\, R \\ &= \mathrm{d}\phi_n \left[B + V\right] + \mathrm{d}\phi_{n+1}\, R + \mathrm{d}\phi_{n+1}\, S - \mathrm{d}\phi_{n+1}\, S \end{aligned}$$

- The terms labelled $S$ are called the Subtraction Dipoles
- They are defined to mimic the singular behaviour of $R$ and $V$
- When all integrands are finite a Monte Carlo simulation can be performed

# Matching algorithms

$$
\begin{aligned}
\mathrm{d}\sigma^{NLO} &= \mathrm{d}\phi_n \left[B + V\right] + \mathrm{d}\phi_{n+1}\, R \\
&= \mathrm{d}\phi_n \left[B + V\right] + \mathrm{d}\phi_{n+1}\, R + \mathrm{d}\phi_{n+1}\, S - \mathrm{d}\phi_{n+1}\, S \\
&= \mathrm{d}\phi_n \left[B + V + \int_{\phi_1} S\right] + \mathrm{d}\phi_{n+1} \left[R - S\right]
\end{aligned}
$$

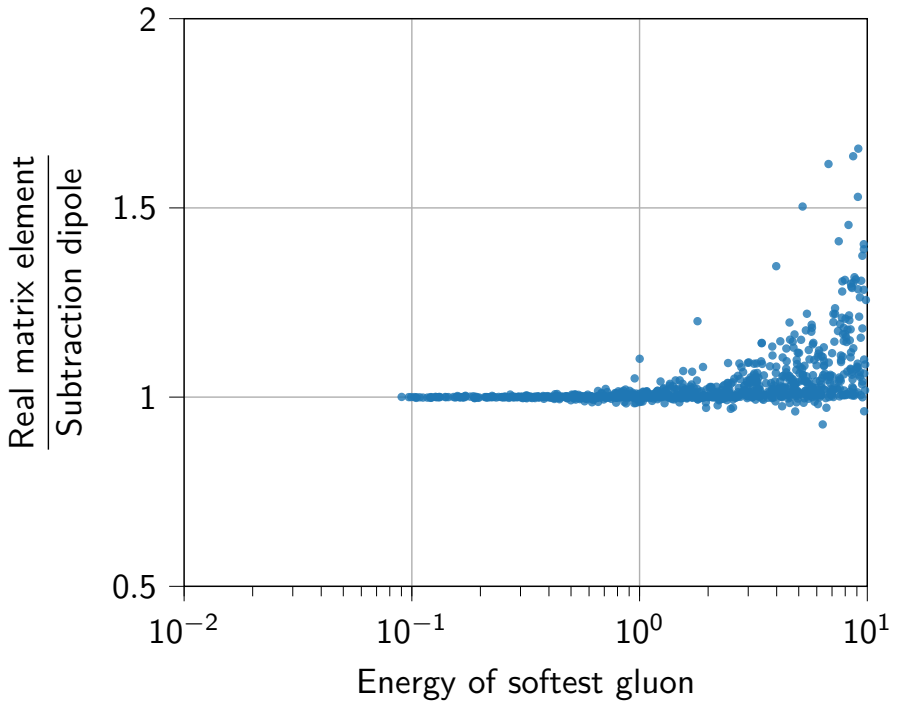
- The terms labelled $S$ are called the Subtraction Dipoles
- They are defined to mimic the singular behaviour of $R$ and $V$
- When all integrands are finite a Monte Carlo simulation can be performed

# Matching algorithms

$$\begin{aligned} \mathrm{d}\sigma^{Matched} =& PS_1 \otimes \mathrm{d}\phi_n \left[ B + V + \int_{\phi_1} S \right] \\ &+ PS_1 \otimes \mathrm{d}\phi_{n+1} \left[ P - S \right] \\ &+ PS_2 \otimes \mathrm{d}\phi_{n+1} \left[ R - P \right] \end{aligned}$$

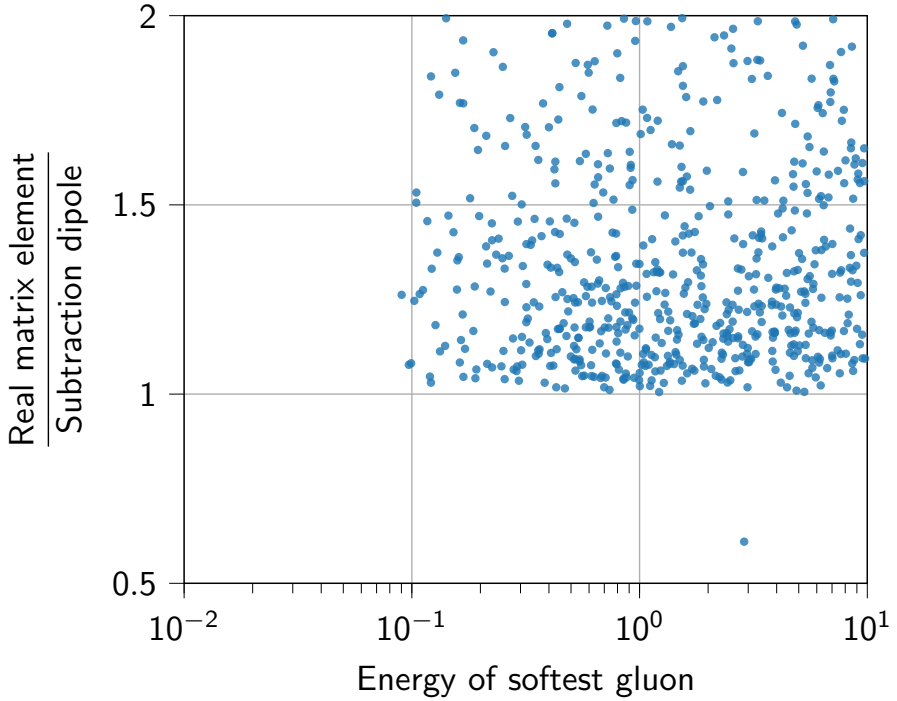
- $V$, $R$, $S$ and $P$ are made of many terms
- Need to consistently handle all of them
- What happens when something goes wrong?

# Current diagnosis tools




- Lower values on $x$-axis means more singular events
- The more singular an event the closer $\frac{S}{R}$ should be to 1
- Points are smeared because of randomised momenta

# Example issue




- No longer seeing convergence
- Difficult to extract any further information though
- Kinematic information about each point is lost

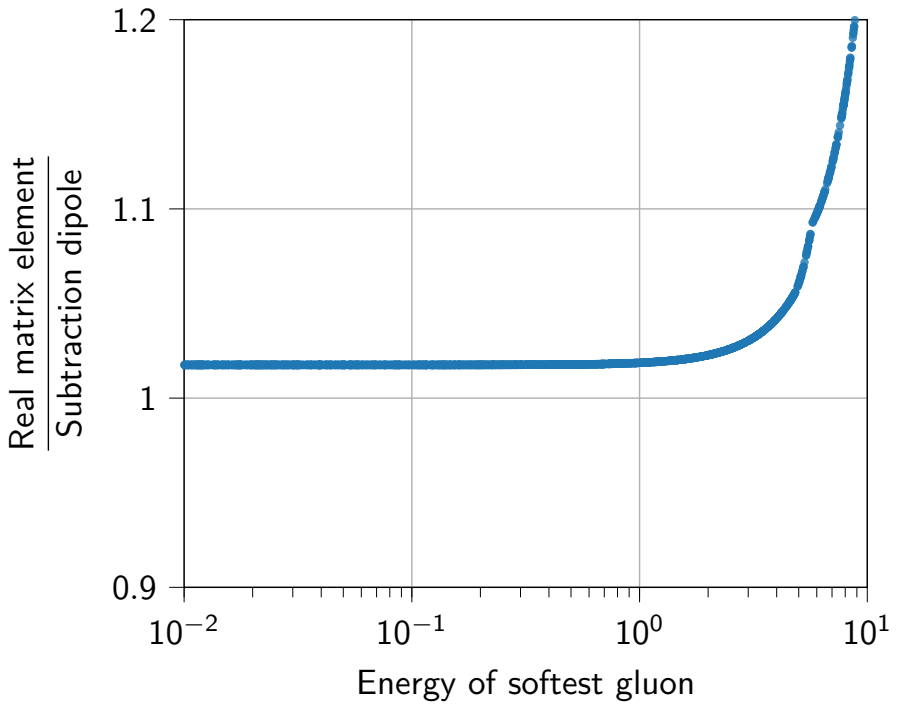
# The SingularPhasespace approach

- Take a momentum configuration
- Approach singular limit, keeping as much the same as possible
- Still need to conserve momentum and keep everything on-shell
- Use tilde transformations (TT):

$$3 \text{ particles} \xrightarrow{\text{TT}} 2 \text{ particles}$$

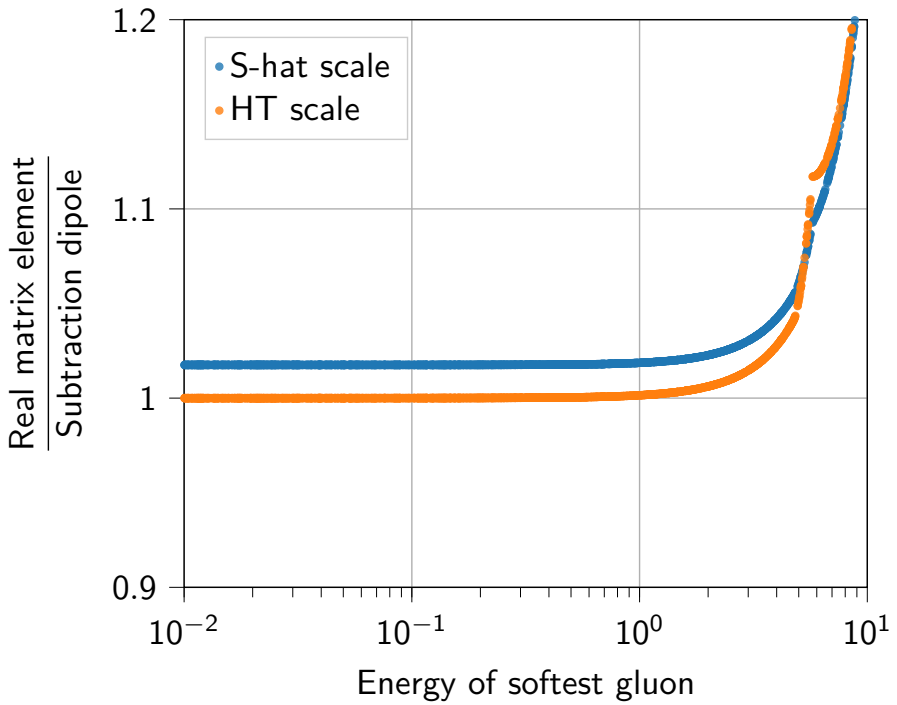
$$2 \text{ particles} \xrightarrow{\text{Inverted TT } (z, k_\perp, \phi)} 3 \text{ particles}$$

# Example issue again




- Converges to a constant factor off one
- Scale choice: $\hat{s}$
- $\hat{s}$ is also constant across the simulation 🤔

# Example issue again




- Converges to a constant factor off one
- Scale choice: $\hat{s}$
- $\hat{s}$ is also constant across the simulation 🤔

# SingularPhasespace

SingularPhasespace can also be used to check everything is working:

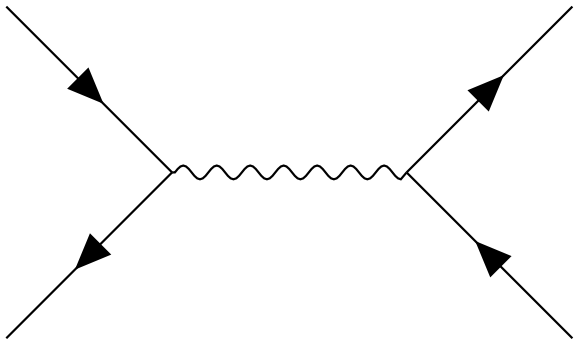
1. Generate random momenta
2. Step into singular limit
3. Check for suspicious data
4. Repeat

Lower CPU and disk space cost than just waiting for a random generation to reach interesting phasespace.

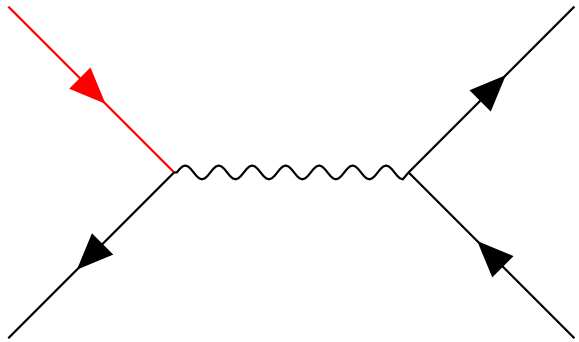
# Matrix elements in MadGraph

- MadGraph uses the helicity amplitude basis, which expresses amplitudes in terms of products of spinors instead of momenta.
- This simplifies the number of terms to calculate and makes it easier to handle interference between diagrams.
- Final result in terms of helicity of external particles, so need to loop over all possible combinations.

# Loop over helicity



# Loop over helicity



- The amplitude is the sum of all possible helicity combinations
- Each leg can be helicity $+1$ or $-1$
- There are 8 possible combinations where red leg has helicity $+1$
- Aim of project: Only calculate spinors/wavefunctions once and store them to RAM
- Already seen $\sim 40\%$ speed increase for $2 \to 2$

# Summary

## SingularPhasespace

- Take an event and push it into a singular limit
- Can be used to search for and diagnose issue in matching algorithms

## Helicity recycling

- Store spinors and wavefunctions to RAM instead of repeatedly calculating them
- Hope to apply to both LO and NLO calculations

Thanks!