Emulation of high multiplicity NLO K-factors in $e^- e^+$ collisions

ACAT 2022

Henry Truong with Daniel Maître
Institute for Particle Physics Phenomenology
Durham University
Durham, United Kingdom

25th October 2022

Introduction

Motivation and aim

- As we have heard already event generation could be quicker
- One-loop matrix element evaluations **are slow** – $\mathcal{O}$(second) per evaluation

Introduction

Motivation and aim

- × As we have heard already event generation could be quicker
- × One-loop matrix element evaluations **are slow** – $\mathcal{O}$(second) per evaluation
- → Accelerate evaluations by building a **fast** and **accurate** emulator

Introduction

Motivation and aim

- × As we have heard already event generation could be quicker
- × One-loop matrix element evaluations **are slow** – $\mathcal{O}$(second) per evaluation
- → Accelerate evaluations by building a **fast** and **accurate** emulator
- Current emulators limited to low final-state multiplicities [1, 2] or have lower per-point accuracy [3]

[1] J. Aylett-Bullock, S. Badger, R. Moodie, *Optimising simulations for diphoton production at hadron colliders using amplitude neural networks*, **JHEP** 08 (2021), p. 066
[2] S. Badger, A. Butter, M. Luchmann, S. Pitz, T. Plehn, *Loop Amplitudes from Precision Networks*, [arXiv:2206.14831]
[3] J. Aylett-Bullock and S. Badger, *Using neural networks for efficient evaluation of high multiplicity scattering amplitudes*, **JHEP** 06 (2020), p. 114

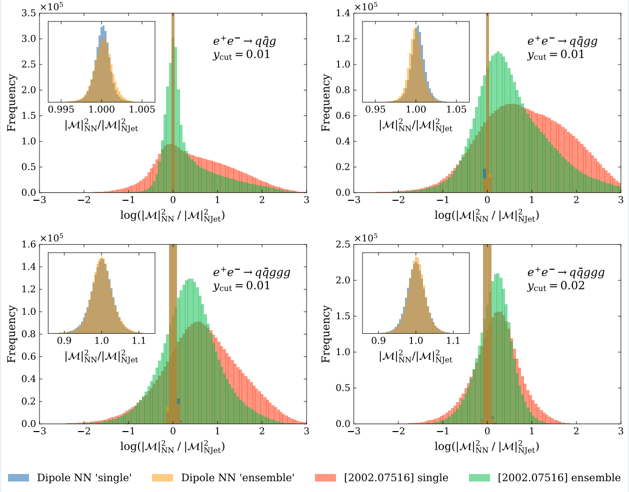
Introduction

Motivation and aim

- × As we have heard already event generation could be quicker
- × One-loop matrix element evaluations **are slow** – $\mathcal{O}$(second) per evaluation
- → Accelerate evaluations by building a **fast** and **accurate** emulator
- Current emulators limited to low final-state multiplicities [1, 2] or have lower per-point accuracy [3]
- Inclusion of **universal QCD infrared structure** in emulator enables accurate modelling, even for higher multiplicities

[1] J. Aylett-Bullock, S. Badger, R. Moodie, *Optimising simulations for diphoton production at hadron colliders using amplitude neural networks*, **JHEP** 08 (2021), p. 066
[2] S. Badger, A. Butter, M. Luchmann, S. Pitz, T. Plehn, *Loop Amplitudes from Precision Networks*, [arXiv:2206.14831]
[3] J. Aylett-Bullock and S. Badger, *Using neural networks for efficient evaluation of high multiplicity scattering amplitudes*, **JHEP** 06 (2020), p. 114

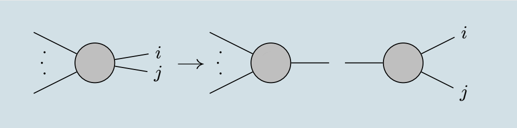
Previous work: $e^- e^+ \to q\bar{q} + \text{gluons}$ at tree-level [4]



[4] D. Maître and H. Truong, *A factorisation-aware Matrix element emulator*, **JHEP** 11 (2021), p. 066

Factorisation of matrix elements

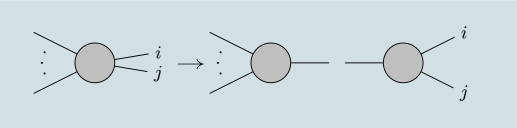
Tree-level



$$|\mathcal{M}_{n+1}^{(0)}|^2 \to X_{ijk}^0 |\mathcal{M}_n^{\text{tree}}|^2 \quad (1)$$

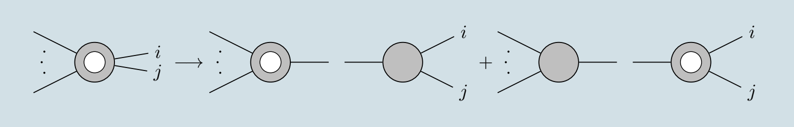
Factorisation of matrix elements

Tree-level



$$|\mathcal{M}_{n+1}^{(0)}|^2 \to X_{ijk}^0 |\mathcal{M}_n^{\text{tree}}|^2 \tag{1}$$

One-loop



$$|\mathcal{M}_{n+1}^{(1)}|^2 \equiv 2\mathrm{Re}(\mathcal{M}_{n+1}^{1-\text{loop}} \mathcal{M}_{n+1}^{\text{tree},*}) \to X_{ijk}^0 |\mathcal{M}_n^{(1)}|^2 + X_{ijk}^1 |\mathcal{M}_n^{(0)}|^2 \tag{2}$$

Antenna functions [5]

X^0_{ijk} and X^1_{ijk} are derived from physical matrix elements and so by construction have the **correct infrared behaviour in the collinear and soft regions**.

Class	Radiation	Antenna functions	
		Tree-level	One-loop
Quark-antiquark	$q\bar{q} \to qg\bar{q}$	A_3^0	$A_3^1,\ \tilde{A}_3^1,\ \hat{A}_3^1$
Quark-gluon	$qg \to qgg$	D_3^0	$D_3^1,\ \hat{D}_3^1$
	$qg \to qQ\bar{Q}$	E_3^0	$E_3^1,\ \tilde{E}_3^1,\ \hat{E}_3^1$
Gluon-gluon	$gg \to ggg$	F_3^0	$F_3^1,\ \hat{F}_3^1$
	$gg \to gq\bar{q}$	G_3^0	$G_3^1,\ \tilde{G}_3^1,\ \hat{G}_3^1$

[5] A. Gehrmann–De Ridder, T. Gehrmann, E.W.N. Glover, *Antenna Subtraction at NNLO*, **JHEP** 09 (2005), p. 056

Antenna functions [5]

X^0_{ijk} and X^1_{ijk} are ... by construction have the **correct** ... **t regions**.

Qu	$\hat{A}^1_3$
Q	$^1_3$
	$\hat{E}^1_3$
G	$^1_3$
	$\hat{G}^1_3$

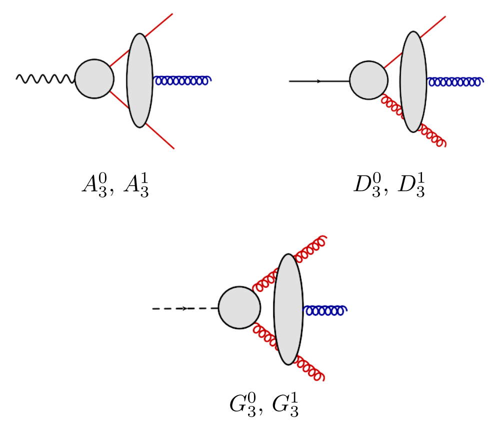


image credit to James Currie

[5] A. Gehrmann–De Ridder, T. Gehrmann, E.W.N. Glover, *Antenna Subtraction at NNLO*, **JHEP** 09 (2005), p. 056

NLO K-factor ansatz

NLO K-factor ansatz

- ✓ K-factors **naturally of order unity**

NLO K-factor ansatz

- ✓ K-factors **naturally of order unity**
- ✓ **Avoid multi-peaked** distribution of $|\mathcal{M}^{(1)}|^2$

NLO K-factor ansatz

- K-factors **naturally of order unity**
- **Avoid multi-peaked** distribution of $|\mathcal{M}^{(1)}|^2$

Factorisation of K-factor

$$\begin{aligned} K_{n+1} &= \frac{|\mathcal{M}^{(1)}_{n+1}|^2}{|\mathcal{M}^{(0)}_{n+1}|^2} \\ &\to \frac{X^0_{ijk}|\mathcal{M}^{(1)}_n|^2 + X^1_{ijk}|\mathcal{M}^{(0)}_n|^2}{X^0_{ijk}|\mathcal{M}^{(0)}_n|^2} = K_n + \frac{X^1_{ijk}}{X^0_{ijk}} \end{aligned} \tag{3}$$

NLO K-factor ansatz

- ✓ K-factors **naturally of order unity**
- ✓ **Avoid multi-peaked** distribution of $|\mathcal{M}^{(1)}|^2$

Factorisation of K-factor

$$\begin{aligned} K_{n+1} &= \frac{|\mathcal{M}^{(1)}_{n+1}|^2}{|\mathcal{M}^{(0)}_{n+1}|^2} \\ &\to \frac{X^0_{ijk}|\mathcal{M}^{(1)}_n|^2 + X^1_{ijk}|\mathcal{M}^{(0)}_n|^2}{X^0_{ijk}|\mathcal{M}^{(0)}_n|^2} = K_n + \frac{X^1_{ijk}}{X^0_{ijk}} \end{aligned} \tag{3}$$

Ansatz

$$\to \quad K_{n+1} = C_0 + \sum_{\{ijk\}} C_{ijk} \frac{X^1_{ijk}}{X^0_{ijk}} \tag{4}$$

Neural network is algorithm of choice

- ✓ Neural networks are good function approximators

Neural network is algorithm of choice

- ✓ Neural networks are good function approximators
- ✓ Shown to work well at tree-level

Neural network is algorithm of choice

- ✓ Neural networks are good function approximators
- ✓ Shown to work well at tree-level
- ✓ Accelerated naturally on GPUs

Neural network is algorithm of choice

- ✓ Neural networks are good function approximators
- ✓ Shown to work well at tree-level
- ✓ Accelerated naturally on GPUs
- ✓ Generic interfaces available for HEP applications

Model inputs and outputs

Inputs (100k datapoints)

$$\begin{aligned}
p &= [E, p_x, p_y, p_z]\,, \quad \sqrt{s} = 1000\text{ GeV} \\
r_{ijk} &= \frac{s_{jk}}{s_{ij} + s_{jk}} \\
\rho_{ijk} &= \sqrt{1 + \frac{4r_{ijk}(1 - r_{ijk})s_{ij}s_{jk}}{s_{ijk}s_{ik}}} \\
s_{ij} &= (p_i + p_j)^2 \\
&\mu_R
\end{aligned} \tag{5}$$

Model inputs and outputs

Inputs (100k datapoints)

$$
\begin{aligned}
p &= [E, p_x, p_y, p_z]\,, \quad \sqrt{s} = 1000\ \mathrm{GeV} \\
r_{ijk} &= \frac{s_{jk}}{s_{ij} + s_{jk}} \\
\rho_{ijk} &= \sqrt{1 + \frac{4 r_{ijk}(1 - r_{ijk}) s_{ij} s_{jk}}{s_{ijk} s_{ik}}} \\
s_{ij} &= (p_i + p_j)^2 \\
&\mu_R
\end{aligned}
\tag{5}
$$

Outputs

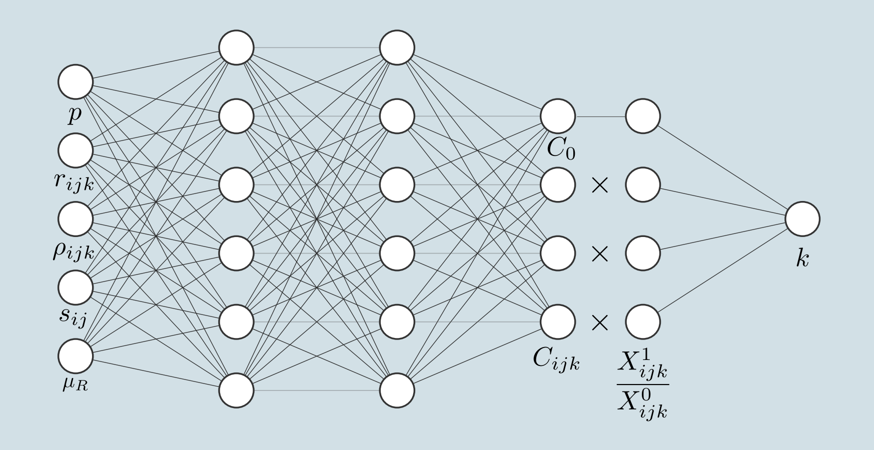
$\{C_0, C_{ijk}\} \quad \rightarrow \quad K$ with ansatz (4)

Neural network hyperparameters

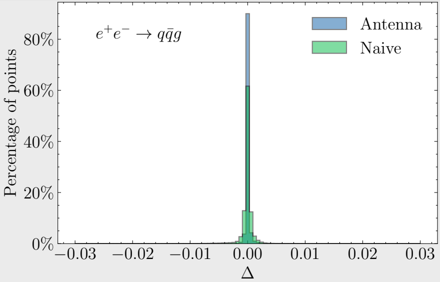
Table: Hyperparameters of the neural network and their values.

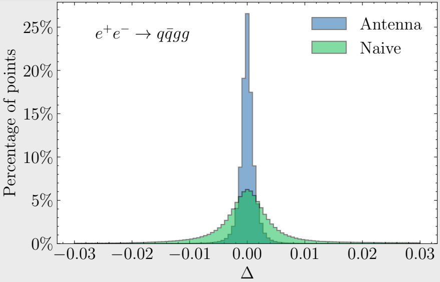
Parameter	Value
Hidden layers	3
Nodes in hidden layers	[64, 64, 64]
Activation function	swish
Weight initialiser	Glorot uniform
Loss function	MAE (k-factor), MSE (one-loop matrix element)
Batch size	256
Optimiser	Adam
Learning rate	10^{-3}
Callbacks	`EarlyStopping`, `RatioEarlyStopping`, `ReduceLROnPlateau`

Neural network architecture

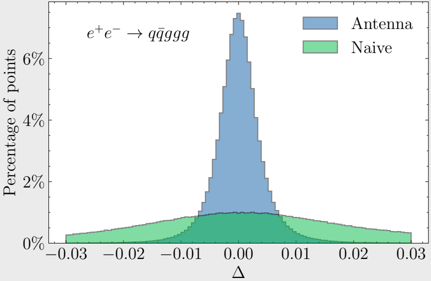


Comparison to naive model

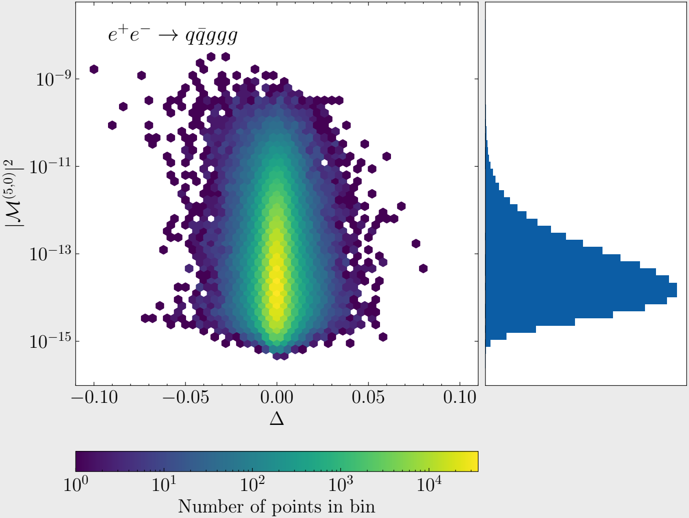




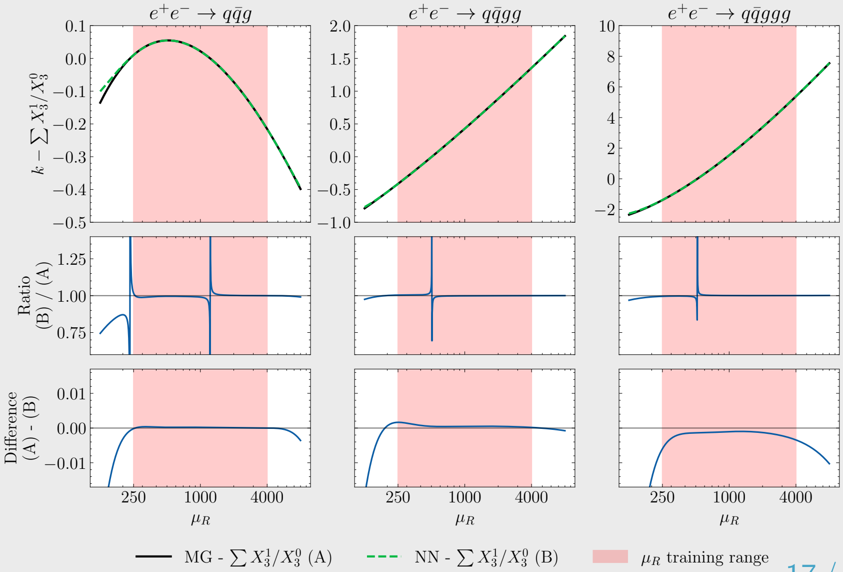
$$\Delta = K_{\text{true}} - K_{\text{pred}} = \frac{|\mathcal{M}^{(1)}_{\text{true}}|^2 - |\mathcal{M}^{(1)}_{\text{pred}}|^2}{|\mathcal{M}^{(0)}_{\text{true}}|^2}$$



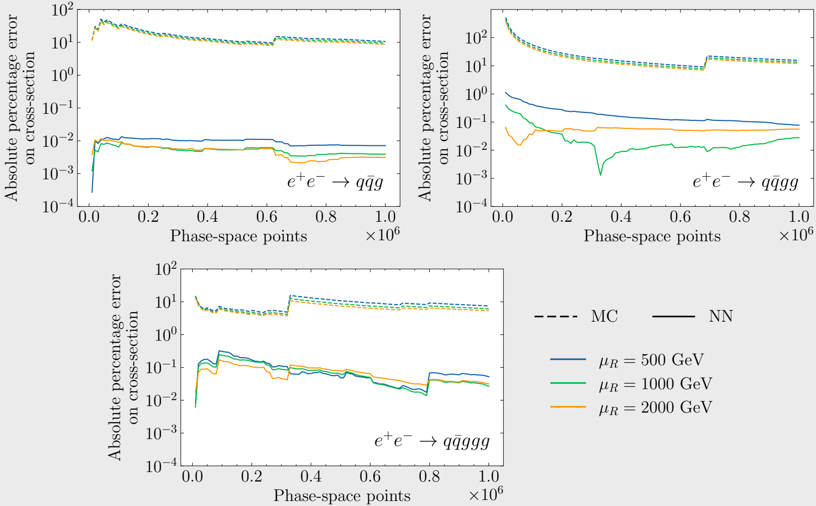
Distribution of $5j$ errors



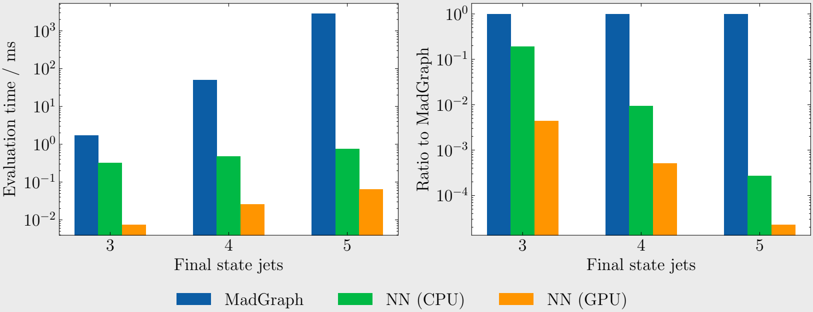
Renormalisation scale dependence



Total cross-section predictions



Evaluation time



Summary

- Possible to **accurately** emulate NLO K-factors for up to $2 \to 5$ process

Summary

- Possible to **accurately** emulate NLO K-factors for up to $2 \to 5$ process
- Building in **universal QCD IR structure** into model enables soft and collinear region predictions to be well behaved

Summary

- Possible to **accurately** emulate NLO K-factors for up to $2 \to 5$ process
- Building in **universal QCD IR structure** into model enables soft and collinear region predictions to be well behaved
- Accumulated error in total cross-section **much lower** than statistical Monte Carlo error

Summary

- Possible to **accurately** emulate NLO K-factors for up to $2 \to 5$ process
- Building in **universal QCD IR structure** into model enables soft and collinear region predictions to be well behaved
- Accumulated error in total cross-section **much lower** than statistical Monte Carlo error
- **Orders of magnitude speed up** whilst keeping errors to the **1%** level

Outlook

- Extend methodology to pp collisions at NLO QCD
- Bridge gap between proof of concept and actual usage in event generators
  - again see Timo's talk on Wednesday afternoon for a possible application in event unweighting

References

[1] Simon Badger et al. "Loop Amplitudes from Precision Networks". In: (June 2022).

[2] Joseph Aylett-Bullock et al. "Optimising simulations for diphoton production at hadron colliders using amplitude neural networks". In: **JHEP** 08 (2021), p. 066.

[3] Simon Badger and Joseph Bullock. "Using neural networks for efficient evaluation of high multiplicity scattering amplitudes". In: **JHEP** 06 (2020), p. 114.

[4] Daniel Maître and Henry Truong. "A factorisation-aware Matrix element emulator". In: **JHEP** 11 (2021), p. 066.

[5] A. Gehrmann-De Ridder et al. "Antenna subtraction at NNLO". In: **JHEP** 09 (2005), p. 056.

Evaluation time breakdown