Workshop on Radiative Corrections at Future Lepton Colliders

Next-to-soft stabilization

Sara Gündogdu
Paul Scherrer Institute & University of Zurich

CERN, 1 July 2026

Workshop on Radiative Corrections at Future Lepton Colliders

Next-to-soft ~~stabilization~~ applications

Sara Gündogdu
Paul Scherrer Institute & University of Zurich

CERN, 1 July 2026

Next-to-soft expansion

Next-to-soft stabilization

Approximate higher orders

conclusion and outlook

Next-to-soft expansion

Next-to-soft stabilization

Approximate higher orders

conclusion and outlook

The structure of (squared) QED amplitudes $\mathcal{M}_{n+1}^{(\ell)}$ in the soft photon (momentum k) limit $\lambda \sim \frac{k}{p_i} \sim \frac{k}{m_i} \ll 1$

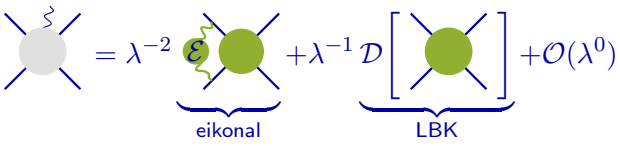
The structure of (squared) QED amplitudes $\mathcal{M}_{n+1}^{(\ell)}$ in the soft photon (momentum k) limit $\lambda \sim \frac{k}{p_i} \sim \frac{k}{m_i} \ll 1$

- Leading power (LP) given by the eikonal @all orders: $\mathcal{M}_{n+1}^{(\ell)} = \mathcal{E}\,\mathcal{M}_{n}^{(\ell)} + \mathcal{O}(\lambda^{-1})$
  [Yennie, Frautschi, Suura (1961)]

$$\mathcal{E} = -\frac{1}{\lambda^2} \sum_{ij}^{n} Q_i Q_j \frac{p_i \cdot p_j}{(k \cdot p_i)(k \cdot p_j)}$$

The structure of (squared) QED amplitudes $\mathcal{M}_{n+1}^{(\ell)}$ in the soft photon (momentum k) limit $\lambda \sim \frac{k}{p_i} \sim \frac{k}{m_i} \ll 1$

- Leading power (LP) given by the eikonal @all orders: $\mathcal{M}_{n+1}^{(\ell)} = \mathcal{E}\,\mathcal{M}_{n}^{(\ell)} + \mathcal{O}(\lambda^{-1})$
  [Yennie, Frautschi, Suura (1961)]
- NLP given by the LBK operator $\mathcal{D}$ @tree level:
  [Low (1958)] [Burnett, Kroll (1968)]

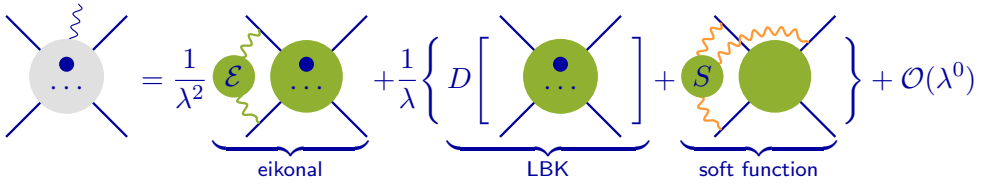


The structure of (squared) QED amplitudes $\mathcal{M}_{n+1}^{(\ell)}$ in the soft photon (momentum k) limit $\lambda \sim \frac{k}{p_i} \sim \frac{k}{m_i} \ll 1$

- Leading power (LP) given by the eikonal @all orders: $\mathcal{M}_{n+1}^{(\ell)} = \mathcal{E}\,\mathcal{M}_{n}^{(\ell)} + \mathcal{O}(\lambda^{-1})$
  [Yennie, Frautschi, Suura (1961)]
- NLP given by the LBK operator $\mathcal{D}$ @tree level: $\mathcal{M}_{n+1}^{(0)} = (\mathcal{E} + \mathcal{D})\,\mathcal{M}_{n}^{(0)} + \mathcal{O}(\lambda^{0})$
  [Low (1958)] [Burnett, Kroll (1968)]
- NLP @any order given by one-loop exact soft function $\mathcal{S}$: [Engel, Signer, Ulrich (2022)]

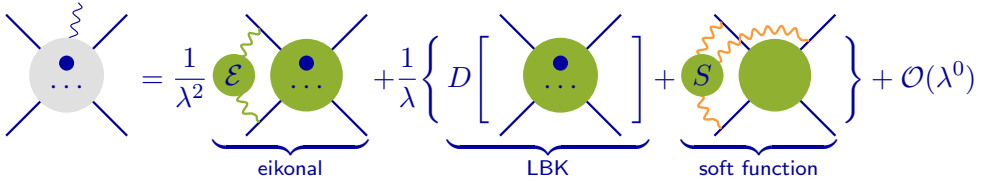
The structure of (squared) QED amplitudes $\mathcal{M}_{n+1}^{(\ell)}$ in the soft photon (momentum k) limit $\lambda \sim \frac{k}{p_i} \sim \frac{k}{m_i} \ll 1$

- Leading power (LP) given by the eikonal @all orders: $\mathcal{M}_{n+1}^{(\ell)} = \mathcal{E}\,\mathcal{M}_{n}^{(\ell)} + \mathcal{O}(\lambda^{-1})$
  [Yennie, Frautschi, Suura (1961)]
- NLP given by the LBK operator $\mathcal{D}$ @tree level: $\mathcal{M}_{n+1}^{(0)} = (\mathcal{E} + \mathcal{D})\,\mathcal{M}_{n}^{(0)} + \mathcal{O}(\lambda^{0})$
  [Low (1958)] [Burnett, Kroll (1968)]
- NLP @any order given by one-loop exact soft function $\mathcal{S}$: [Engel, Signer, Ulrich (2022)]



The structure of (squared) QED amplitudes $\mathcal{M}_{n+1}^{(\ell)}$ in the soft photon (momentum k) limit $\lambda \sim \frac{k}{p_i} \sim \frac{k}{m_i} \ll 1$

- Leading power (LP) given by the eikonal @all orders: $\mathcal{M}_{n+1}^{(\ell)} = \mathcal{E}\,\mathcal{M}_n^{(\ell)} + \mathcal{O}(\lambda^{-1})$
  [Yennie, Frautschi, Suura (1961)]
- NLP given by the LBK operator $\mathcal{D}$ @tree level: $\mathcal{M}_{n+1}^{(0)} = (\mathcal{E} + \mathcal{D})\,\mathcal{M}_n^{(0)} + \mathcal{O}(\lambda^0)$
  [Low (1958)] [Burnett, Kroll (1968)]
- NLP @any order given by one-loop exact soft function $\mathcal{S}$: [Engel, Signer, Ulrich (2022)]

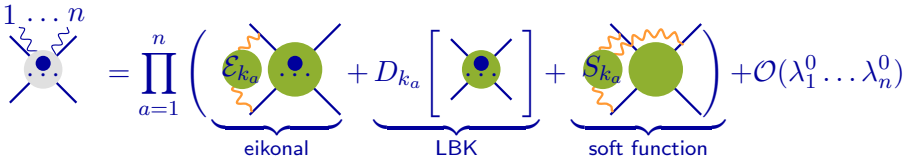


Loop-level extension via heavy-quark effective theory:

- Hard region: LBK operator — valid to all orders
- Soft region: one-loop exact

- NLP multiple ordered photons: $\lambda_1 \sim \frac{k_1}{p_i} \sim \frac{k_1}{m_i} \ll \ldots \ll \lambda_n \sim \frac{k_n}{p_i} \sim \frac{k_n}{m_i} \ll 1$

  [Engel (2024)]



- NLP multiple ordered photons: $\lambda_1 \sim \frac{k_1}{p_i} \sim \frac{k_1}{m_i} \ll \ldots \ll \lambda_n \sim \frac{k_n}{p_i} \sim \frac{k_n}{m_i} \ll 1$
  [Engel (2024)]

$$\mathcal{M}_{m+n} = \prod_{a=1}^{n} \left(\mathcal{E}_a(k_a) + \mathcal{D}_a(k_a) + \mathcal{S}_a^{(1)}(k_a) \right) \mathcal{M}_m(\{s\}, \{m^2\}) + \mathcal{O}(\lambda_1^0 \ldots \lambda_n^0)$$

- NLP multiple ordered photons: $\lambda_1 \sim \frac{k_1}{p_i} \sim \frac{k_1}{m_i} \ll \ldots \ll \lambda_n \sim \frac{k_n}{p_i} \sim \frac{k_n}{m_i} \ll 1$ [Engel (2024)]

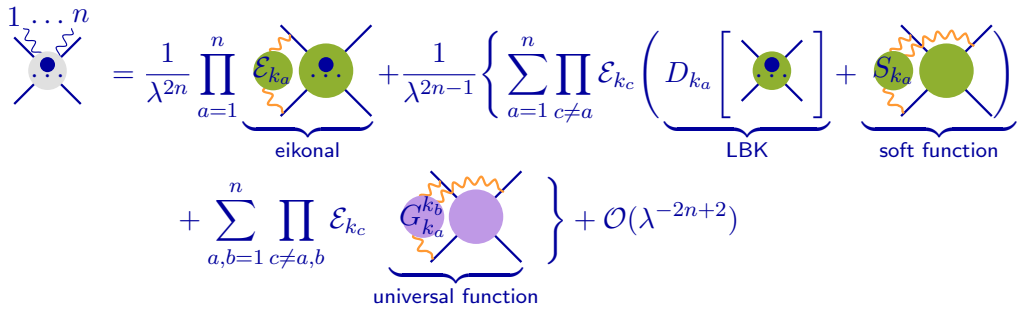
$$\mathcal{M}_{m+n} = \prod_{a=1}^{n} \left(\mathcal{E}_a(k_a) + \mathcal{D}_a(k_a) + \mathcal{S}_a^{(1)}(k_a) \right) \mathcal{M}_m(\{s\}, \{m^2\}) + \mathcal{O}(\lambda_1^0 \ldots \lambda_n^0)$$

- NLP multiple unordered photons: $\lambda \sim \frac{k_a}{p_i} \sim \frac{k_a}{m_i} \ll 1$ [Engel (2024)]

- NLP multiple ordered photons: $\lambda_1 \sim \frac{k_1}{p_i} \sim \frac{k_1}{m_i} \ll \ldots \ll \lambda_n \sim \frac{k_n}{p_i} \sim \frac{k_n}{m_i} \ll 1$ [Engel (2024)]

$$\mathcal{M}_{m+n} = \prod_{a=1}^{n} \left(\mathcal{E}_a(k_a) + \mathcal{D}_a(k_a) + \mathcal{S}_a^{(1)}(k_a) \right) \mathcal{M}_m(\{s\}, \{m^2\}) + \mathcal{O}(\lambda_1^0 \ldots \lambda_n^0)$$

- NLP multiple unordered photons: $\lambda \sim \frac{k_a}{p_i} \sim \frac{k_a}{m_i} \ll 1$ [Engel (2024)]



How can we make LBK suitable for numerical implementation?

Problem: non-radiative amplitude is evaluated with unphysical momenta

$$\sum_i p_i = k \neq 0$$

- ambiguities are NNLP, but unsatisfactory for numerical use

How can we make LBK suitable for numerical implementation?

Problem: non-radiative amplitude is evaluated with unphysical momenta

$$\sum_i p_i = k \neq 0$$

- ambiguities are NNLP, but unsatisfactory for numerical use

Solution: use that derivatives are generators of translations
[Balsach et al. (2023)]

$$\begin{aligned}\mathcal{M}_{n+1}^{(0)} &= \left(\mathcal{E} + \mathcal{D} + \mathcal{O}(\lambda^0)\right) \mathcal{M}_n^{(0)} \\ &\stackrel{!}{=} \mathcal{E} \left(1 - \sum_i \delta p_i \cdot \frac{\partial}{\partial p_i}\right) \mathcal{M}_n^{(0)} \\ &\quad + \mathcal{O}(\lambda^0) \\ &= \mathcal{E}\, \mathcal{M}_n^{(0)}(p') + \mathcal{O}(\lambda^0)\end{aligned}$$

How can we make LBK suitable for numerical implementation?

Problem: non-radiative amplitude is evaluated with unphysical momenta

$$\sum_i p_i = k \neq 0$$

- ambiguities are NNLP, but unsatisfactory for numerical use

Solution: use that derivatives are generators of translations
[Balsach et al. (2023)]

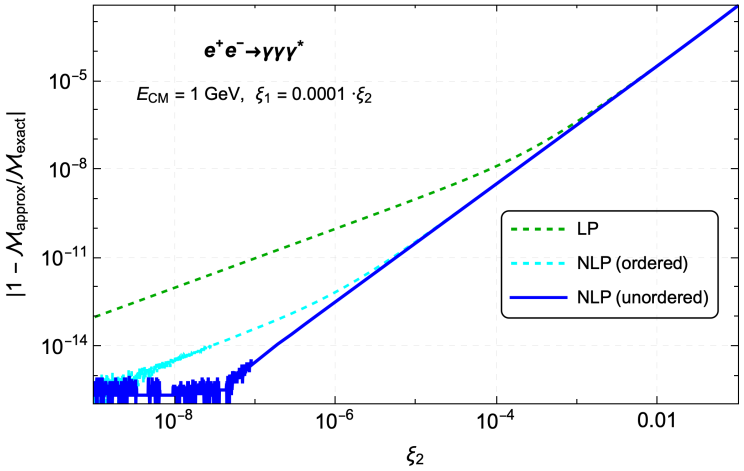
$$\begin{aligned}\mathcal{M}_{n+1}^{(0)} &= \left(\mathcal{E} + \mathcal{D} + \mathcal{O}(\lambda^0)\right) \mathcal{M}_n^{(0)} \\ &\stackrel{!}{=} \mathcal{E}\left(1 - \sum_i \delta p_i \cdot \frac{\partial}{\partial p_i}\right) \mathcal{M}_n^{(0)} \\ &\quad + \mathcal{O}(\lambda^0) \\ &= \mathcal{E}\, \mathcal{M}_n^{(0)}(p') + \mathcal{O}(\lambda^0)\end{aligned}$$

Shifted momenta $p_i' = p_i + \delta p_i$

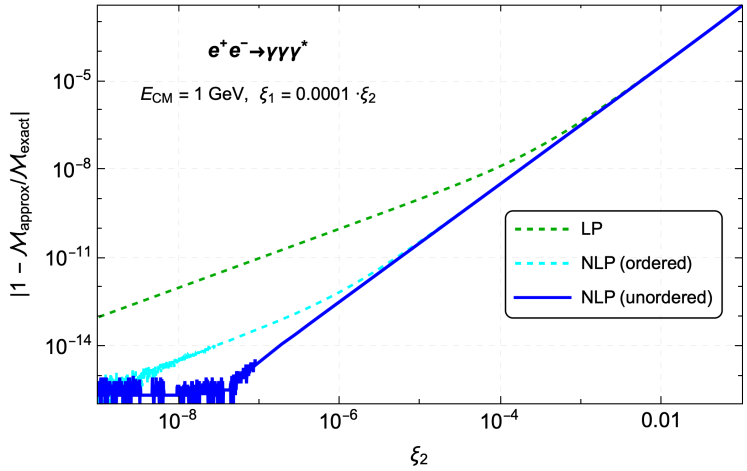
$\delta p_i \sim \mathcal{O}(k)$ such that:

- on-shell: $p_i'^2 = m_i^2$
- mom. cons.: $\sum_i \delta p_i = -k$
- LBK validity preserved

Numerical validation of the shifts for the 2 photon case : $k_1 \ll k_2$



Numerical validation of the shifts for the 2 photon case : $k_1 \ll k_2$



- **Unordered NLP works in all regimes** ✓
- Expected: it retains all k_1/k_2 power corrections

Next-to-soft expansion

Next-to-soft stabilization

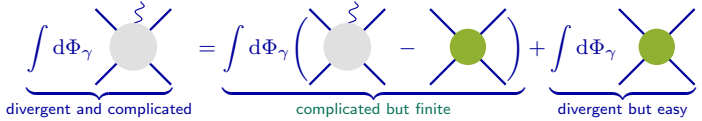
Approximate higher orders

conclusion and outlook

Monte **C**arlo for **MU**ons and other **LE**ptons

- integrator for higher-order predictions in low-energy ($\sqrt{s} \sim$ few GeV) QED
- **input**: matrix elements provided by us or others
- **output**: cross sections for any physical observable at fixed order
- exploit the exponentiation of soft poles: $e^{\hat{\mathcal{E}}} \sum_{\ell=0}^{\infty} \mathcal{M}_n^{(\ell)} = \text{finite}$

use this to construct an all-order subtraction scheme FKS$^\ell$ [Engel, Signer, Ulrich (2019)]



Real-virtual matrix elements suffer from numerical instabilities in the soft photon region

- if $E_\gamma < E_{\text{NTS}}$: switch to NTS expansion
- use OpenLoops for the bulk of the phase space

  [Buccioni, Pozzorini, Zoller (2018)]

  [Buccioni et al. (2019)]

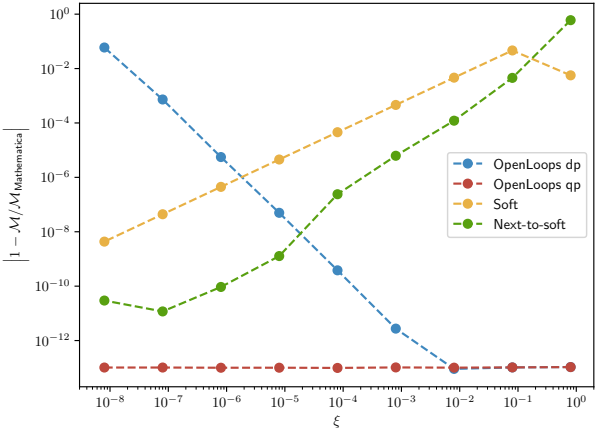
Real-virtual matrix elements suffer from numerical instabilities in the soft photon region

- if $E_\gamma < E_{\text{NTS}}$: switch to NTS expansion
- use OpenLoops for the bulk of the phase space

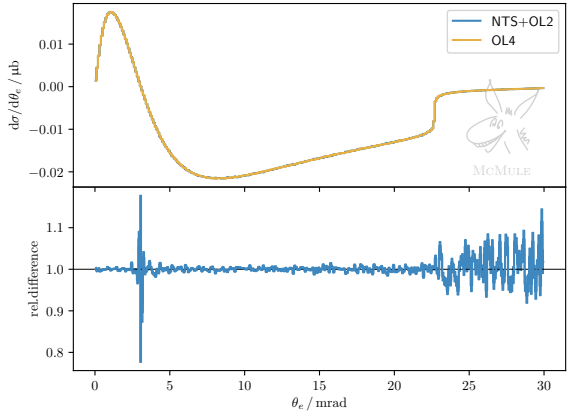
  [Buccioni, Pozzorini, Zoller (2018)]

  [Buccioni et al. (2019)]

Determine E_{NTS} through pointwise tests for different phase space configurations



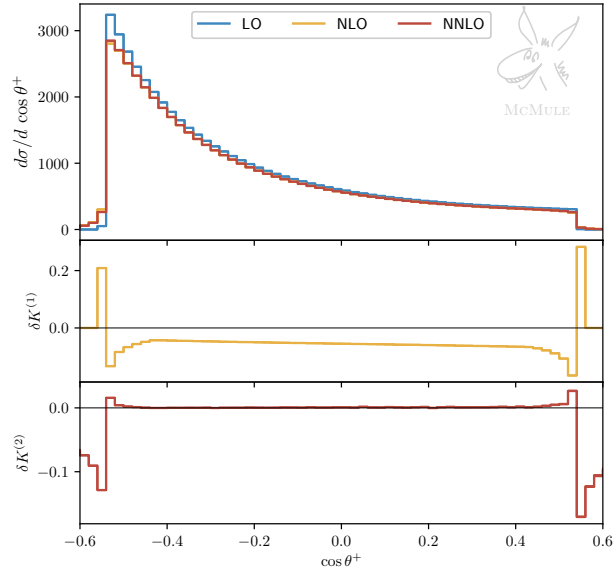
Comparing OpenLoops quad precision (OL4) against double precision (OL2) + NTS



- $\mu\, e \to \mu\, e$ runtime 70 days vs 4 days
- integrated for different cuts

NTS	OL4
$-0.29268(4)$	$-0.29267(4)$
$-0.44789(6)$	$-0.44778(6)$
$-0.64662(9)$	$-0.64649(9)$

$\Rightarrow$ This is not an approximation but a numerical tool!



CMD: $\sqrt{s} = 0.7\,\text{GeV}$

$$K^{(i)} = \frac{\mathrm{d}\sigma_i}{\mathrm{d}\sigma_{i-1}} = 1 + \delta K^{(i)}$$

Comparison to BabaYaga (NLO+PS)

	σ / μb		δK [%]
	McMule	BabaYaga	McMule
LO	+0.99053236(9)	+0.99053131083(3)	
dNLO	−0.0731903(9)	−0.073190(1)	−7.38899(9)
dNNLO vs. d(NLO+PS)	+0.002689(8)	+0.001969(2)	+0.2931(8)

no vacuum polarization in this table, but they are included in plot

Next-to-soft expansion

Next-to-soft stabilization

Approximate higher orders

conclusion and outlook

- @NNLO: use NTS stabilization as a numerical tool ✓

- @NNLO: use NTS stabilization as a numerical tool ✓

Challenge @N3LO: the real-virtual-virtual matrix element $\mathcal{M}_{n+1}^{(2)}$ is known only in the massless case! [Badger et al. (2023)]

- @NNLO: use NTS stabilization as a numerical tool ✓

Challenge @N3LO: the real-virtual-virtual matrix element $\mathcal{M}_{n+1}^{(2)}$ is known only in the massless case! [Badger et al. (2023)]

- **Massification:** expand in m_ℓ^2/Q^2 since $m_\ell^2 \ll Q^2 \sim s$
  [Penin 06; Mitov, Moch 06; Becher, Melnikov 07; Engel et al. 18]

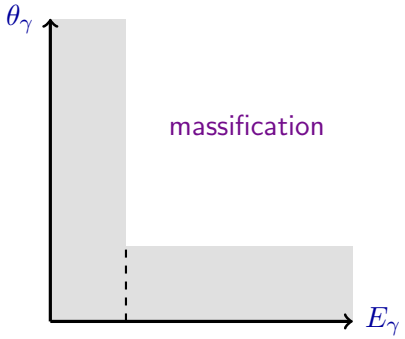
$$\mathcal{A}(m_\ell) = \underbrace{\mathcal{S}}_{\text{soft}} \times \underbrace{Z \times Z}_{\text{collinear}} \times \mathcal{A}(m_\ell = 0) + \mathcal{O}\left(\frac{m_\ell}{Q}\right)$$

$\rightarrow$ not valid where θ_γ is small

- @NNLO: use NTS stabilization as a numerical tool ✓

Challenge @N3LO: the real-virtual-virtual matrix element $\mathcal{M}^{(2)}_{n+1}$ is known only in the massless case! [Badger et al. (2023)]

Massification: where θ_γ not small



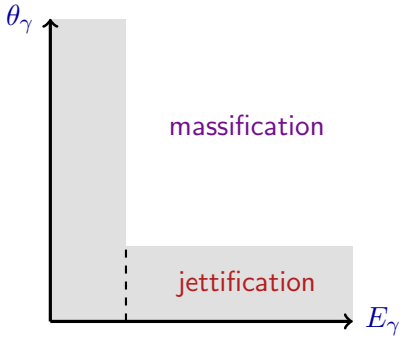
- @NNLO: use NTS stabilization as a numerical tool ✓

Challenge @N3LO: the real-virtual-virtual matrix element $\mathcal{M}^{(2)}_{n+1}$ is known only in the massless case! [Badger et al. (2023)]

Massification: where θ_γ not small

Jettification: massive J unknown at 2-loop

$$\mathcal{M}_{n+1}(m) \xrightarrow[\theta_\gamma \to 0]{m \to 0} \mathcal{M}_n(0) \times Z \times J$$



- @NNLO: use NTS stabilization as a numerical tool ✓

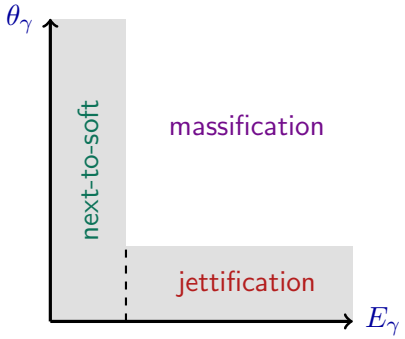
Challenge @N3LO: the real-virtual-virtual matrix element $\mathcal{M}^{(2)}_{n+1}$ is known only in the massless case! [Badger et al. (2023)]

Massification: where θ_γ not small

Jettification: massive J unknown at 2-loop

$$\mathcal{M}_{n+1}(m) \xrightarrow[\theta_\gamma \to 0]{m \to 0} \mathcal{M}_n(0) \times Z \times J$$

NTS stabilisation: $E_\gamma \to 0$



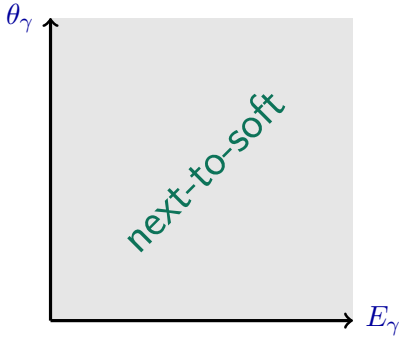
Question: How well can NTS approximate higher orders?

Question: How well can NTS approximate higher orders?

- **YFS (LP) resummation** is already used in Monte Carlo codes
- this will be dominated by $\alpha^3 L^3$ cmp. to NNLO

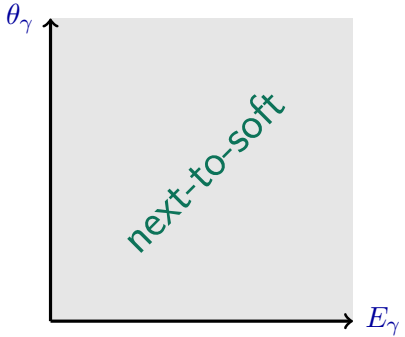
Question: How well can NTS approximate higher orders?

- **YFS (LP) resummation** is already used in Monte Carlo codes
- this will be dominated by $\alpha^3 L^3$ cmp. to NNLO
- What happens if we do **fixed order (NNLO) + next-to-soft (NLP)**?



Question: How well can NTS approximate higher orders?

- **YFS (LP) resummation** is already used in Monte Carlo codes
- this will be dominated by $\alpha^3 L^3$ cmp. to NNLO
- What happens if we do **fixed order (NNLO) + next-to-soft (NLP)**?



↪ Test this @NNLO: Compare NLO+NTS against exact result

Individual matrix elements depend on an unphysical parameter $0 < \xi_c < 1$

$$\boldsymbol{\sigma_n^{(2)}(\xi_c)} \quad \mathcal{M}_n^{(2)} + \hat{\mathcal{E}}(\xi_c)\mathcal{M}_n^{(1)} + \tfrac{1}{2}\hat{\mathcal{E}}(\xi_c)^2\mathcal{M}_n^{(0)}$$

$$\boldsymbol{\sigma_{n+1}^{(2)}(\xi_c)} \quad \mathcal{M}_{n+1}^{(1)} + \hat{\mathcal{E}}(\xi_c)\mathcal{M}_{n+1}^{(0)} - \left(\mathcal{S}(\xi_c)@\mathcal{M}_{n+1}^{(1)} - \hat{\mathcal{E}}(\xi_c)\mathcal{S}(\xi_c)@\mathcal{M}_{n+1}^{(0)}\right)$$

$$\boldsymbol{\sigma_{n+2}^{(2)}(\xi_c)} \quad \tfrac{1}{2}\mathcal{M}_{n+2}^{(0)} - \tfrac{1}{2}\left(\mathcal{S}_1(\xi_c) + \mathcal{S}_2(\xi_c) - \mathcal{S}_1(\xi_c)\mathcal{S}_2(\xi_c)\right)@\mathcal{M}_{n+2}^{(0)}$$

Individual matrix elements depend on an unphysical parameter $0 < \xi_c < 1$

$$\boldsymbol{\sigma_n^{(2)}(\xi_c)} \quad \mathcal{M}_n^{(2)} + \hat{\mathcal{E}}(\xi_c)\mathcal{M}_n^{(1)} + \tfrac{1}{2}\hat{\mathcal{E}}(\xi_c)^2\mathcal{M}_n^{(0)}$$

$$\boldsymbol{\sigma_{n+1}^{(2)}(\xi_c)} \quad \mathcal{M}_{n+1}^{(1)} + \hat{\mathcal{E}}(\xi_c)\mathcal{M}_{n+1}^{(0)} - \left(\mathcal{S}(\xi_c)@\mathcal{M}_{n+1}^{(1)} - \hat{\mathcal{E}}(\xi_c)\mathcal{S}(\xi_c)@\mathcal{M}_{n+1}^{(0)}\right)$$

$$\boldsymbol{\sigma_{n+2}^{(2)}(\xi_c)} \quad \tfrac{1}{2}\mathcal{M}_{n+2}^{(0)} - \tfrac{1}{2}\left(\mathcal{S}_1(\xi_c) + \mathcal{S}_2(\xi_c) - \mathcal{S}_1(\xi_c)\mathcal{S}_2(\xi_c)\right)@\mathcal{M}_{n+2}^{(0)}$$

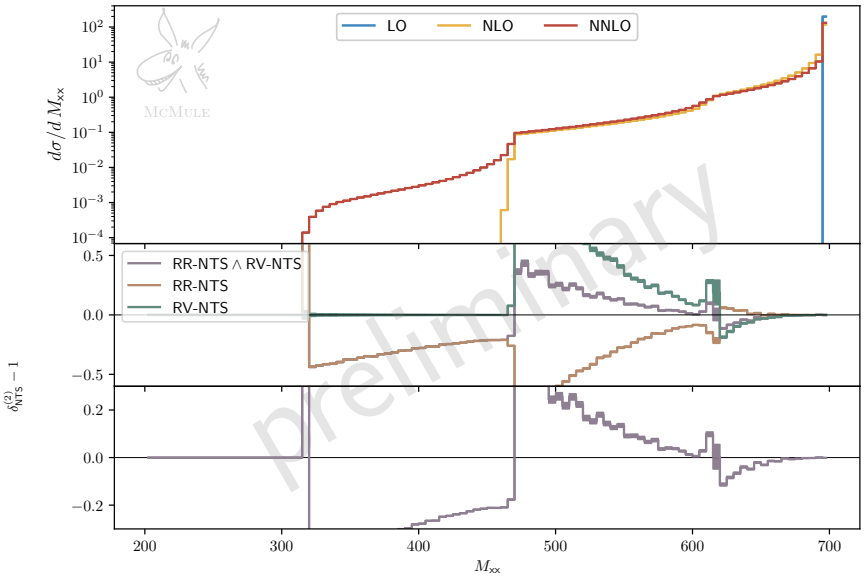
Individual matrix elements depend on an unphysical parameter $0 < \xi_c < 1$

$$\boldsymbol{\sigma_n^{(2)}(\xi_c)} \quad \mathcal{M}_n^{(2)} + \hat{\mathcal{E}}(\xi_c)\mathcal{M}_n^{(1)} + \tfrac{1}{2}\hat{\mathcal{E}}(\xi_c)^2\mathcal{M}_n^{(0)}$$

$$\boldsymbol{\sigma_{n+1}^{(2)}(\xi_c)} \quad \mathcal{M}_{n+1}^{(1)} + \hat{\mathcal{E}}(\xi_c)\mathcal{M}_{n+1}^{(0)} - \big(\mathcal{S}(\xi_c)@\mathcal{M}_{n+1}^{(1)} - \hat{\mathcal{E}}(\xi_c)\mathcal{S}(\xi_c)@\mathcal{M}_{n+1}^{(0)}\big)$$

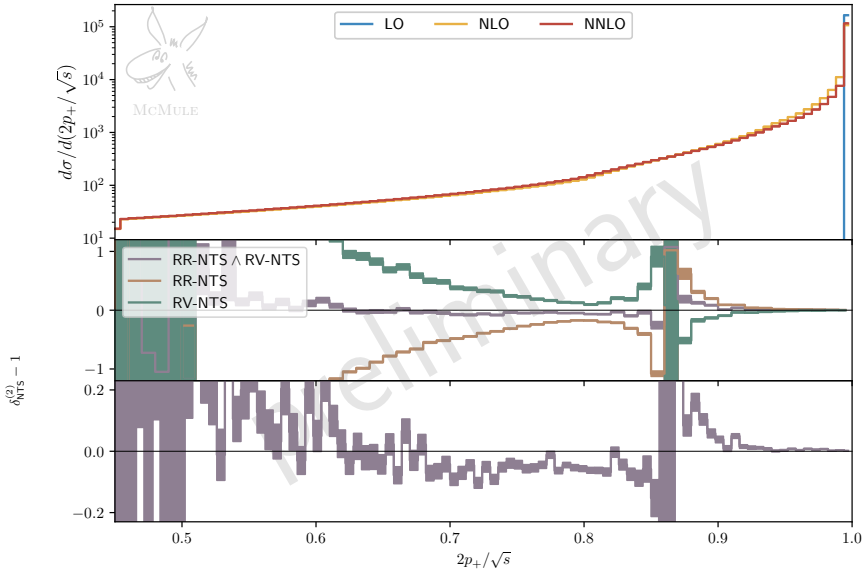
$$\boldsymbol{\sigma_{n+2}^{(2)}(\xi_c)} \quad \tfrac{1}{2}\mathcal{M}_{n+2}^{(0)} - \tfrac{1}{2}\big(\mathcal{S}_1(\xi_c) + \mathcal{S}_2(\xi_c) - \mathcal{S}_1(\xi_c)\mathcal{S}_2(\xi_c)\big)@\mathcal{M}_{n+2}^{(0)}$$

$\hookrightarrow$ ξ_c-independence requires NTS on *all* colored matrix elements at once



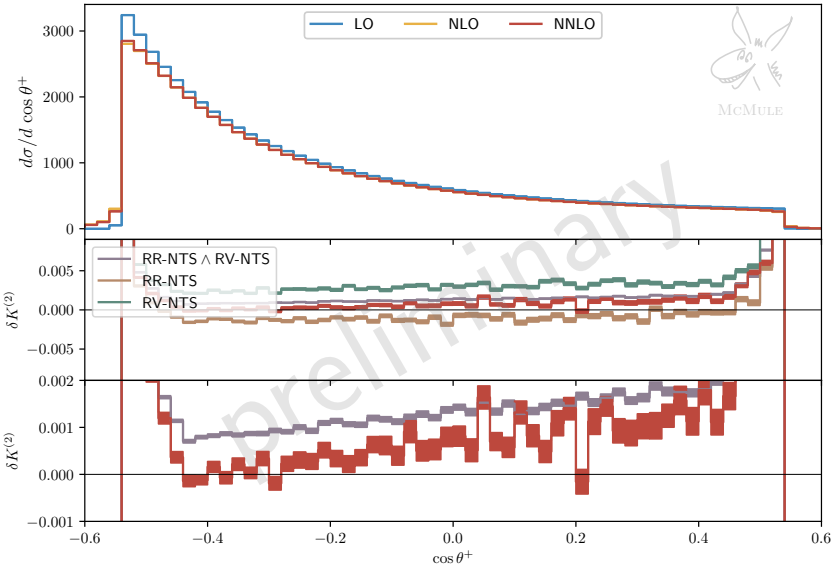
$$\delta^{(i)}_{\rm NTS} = \frac{\delta K^{(i)}_{\rm NTS}}{\delta K^{(i)}}$$

$$M_{XX} = \sqrt{(|\vec{p}_+| + |\vec{p}_-|)^2}$$



$$\delta^{(i)}_{\mathrm{NTS}} = \frac{\delta K^{(i)}_{\mathrm{NTS}}}{\delta K^{(i)}}$$

$$p_+ = |\vec{p}_+| > 0.45 \times \sqrt{s}/2$$


LO
NLO
NNLO
McMule
$d\sigma/d\cos\theta^+$
3000
2000
1000
0
RR-NTS ∧ RV-NTS
RR-NTS
RV-NTS
$\delta K^{(2)}$
0.005
0.000
−0.005
0.002
0.001
0.000
−0.001
−0.6
−0.4
−0.2
0.0
0.2
0.4
0.6
$\cos\theta^+$
preliminary

Next-to-soft expansion

Next-to-soft stabilization

Approximate higher orders

conclusion and outlook

status :

- NTS is a useful numerical tool $\rightarrow$ reach numerical stability in the soft photon region & gain computational speed
- NTS stabilization can (and needs to) be used @high energies and @higher orders
- replacing unknown higher-order matrix elements by a NTS expansion quantifies the impact of missing orders

future progress :

- apply NTS @N3LO and cmp. to exact result: RRR, RRV, RVV with cuts (on θ_γ)
- jettification: needs to be calculated for a complete N^{3}LO prediction
- extend YFS+resummation $\rightarrow$ NTS+resummation, to achieve NLP accuracy

Thank you!

The one-loop radiative amplitude in the soft limit decomposes as:

$$\mathcal{M}_{n+1}^{(1)} \overset{k\sim\xi}{=} \mathcal{M}_{n+1}^{(1),\text{hard}} + \mathcal{M}_{n+1}^{(1),\text{soft}}$$

with

$$\mathcal{M}_{n+1}^{(1),\text{hard}} = \sum_l \sum_i Q_i Q_l \left(-\frac{1}{\xi^2} \frac{p_i \cdot p_l}{(k\cdot p_i)(k\cdot p_l)} + \frac{1}{\xi} \frac{p_l \cdot \tilde{D}_i}{k \cdot p_l} \right) \mathcal{M}_n^{(1)}(\{s\},\{m^2\})$$

$$\mathcal{M}_{n+1}^{(1),\text{soft}} = \frac{1}{\xi} \sum_l \sum_{i\neq j} Q_i^2 Q_j Q_l \left(\frac{p_i \cdot p_l}{(k\cdot p_i)(k\cdot p_l)} - \frac{p_j \cdot p_l}{(k\cdot p_j)(k\cdot p_l)} \right) 2S(p_i,p_j,k) \times \mathcal{M}_n^{(0)}(\{s\},\{m^2\})$$

- hard part: LBK operator $\tilde{D}_i$
- soft part: universal soft function $S(p_i,p_j,k)$ $\times$ tree-level amplitude

How do we get a massive two-loop amplitude ?

- masses are small: $m_\ell^2 \ll s \sim Q^2 \to$ expand in m_ℓ^2/Q^2
- based on SCET factorisation & method of regions

  [Penin 06; Mitov, Moch 06; Becher, Melnikov 07; Engel, Gnendiger, Signer, Ulrich 18]

$$\sim A \log^2 \frac{m_\ell^2}{Q^2} + B \log \frac{m_\ell^2}{Q^2} + C + \mathcal{O}\left(\frac{m_\ell^2}{Q^2}\right)$$

- up to three-loop level: $\mathcal{A}(m_\ell) = \mathcal{S} \times Z \times Z \times \mathcal{A}(0) + \mathcal{O}(m_\ell)$
- soft: process-dependent $\mathcal{S} = 1 +$ fermion loops
- collinear: universal Z, converts $1/\epsilon \to \log(m_\ell^2/Q^2)$
- theory error: $\mathcal{O}(10^{-2})$ @ NNLO $\sim \mathcal{O}(10^{-5})$ for $e\mu \to e\mu$ [Bonciani et al. 21]

- **VVV:** known for massless fermions [Crisanti et al.(2026)]
- **RRR** and **RRV:** from OpenLoops with full mass dependence
- **RVV:** only known in the massless case

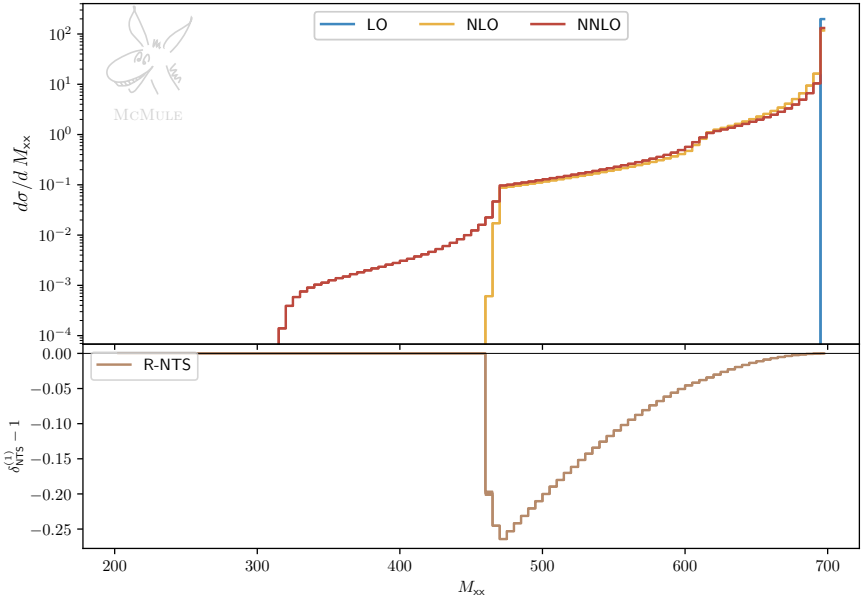
$\sqrt{s} = 0.7\,\text{GeV}$

$1\,\text{rad} \leq (\theta^- - \theta^+ + \pi)/2 \leq \pi - 1\,\text{rad}$

$|\vec{p}_\pm| > 0.45\sqrt{s}/2$

$||\phi^+ - \phi^-| - \pi| < 0.15\,\text{rad}$

$|\theta^+ + \theta^- - \pi| < 0.25\,\text{rad}$



$$\delta^{(i)}_{\mathsf{NTS}} = \frac{\delta K^{(i)}_{\mathsf{NTS}}}{\delta K^{(i)}}$$

$$M_{XX} = \sqrt{(|\vec{p}_+| + |\vec{p}_-|)^2}$$