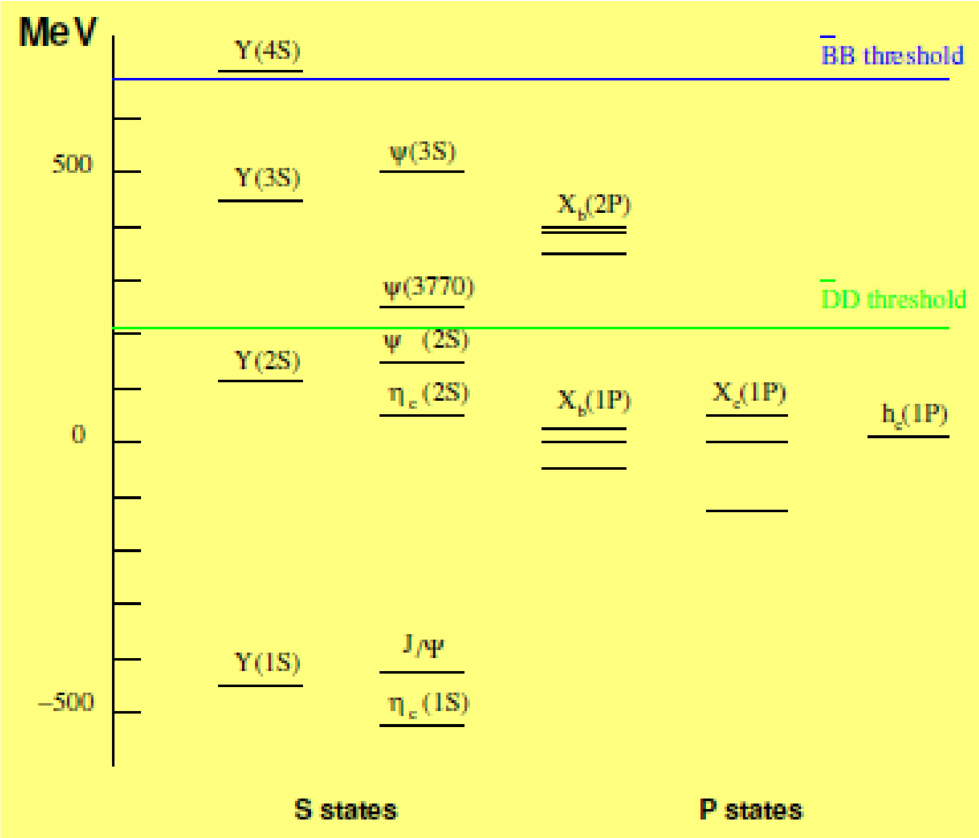
Pair quarkonia production at NLO

Andrei Onishchenko

MIPT & SINP MSU

In collaboration with T. Aushev, A. Berezhnoy and A. Likhoded

Quarkonium scales



The mass scale is perturbative:

$$m_b \approx 5 \text{ GeV},\ m_c \approx 1.5 \text{ GeV}$$

The system is non-relativistic:

$$\Delta_n E \approx mv^2,\ \Delta_{fs} E \approx mv^4$$

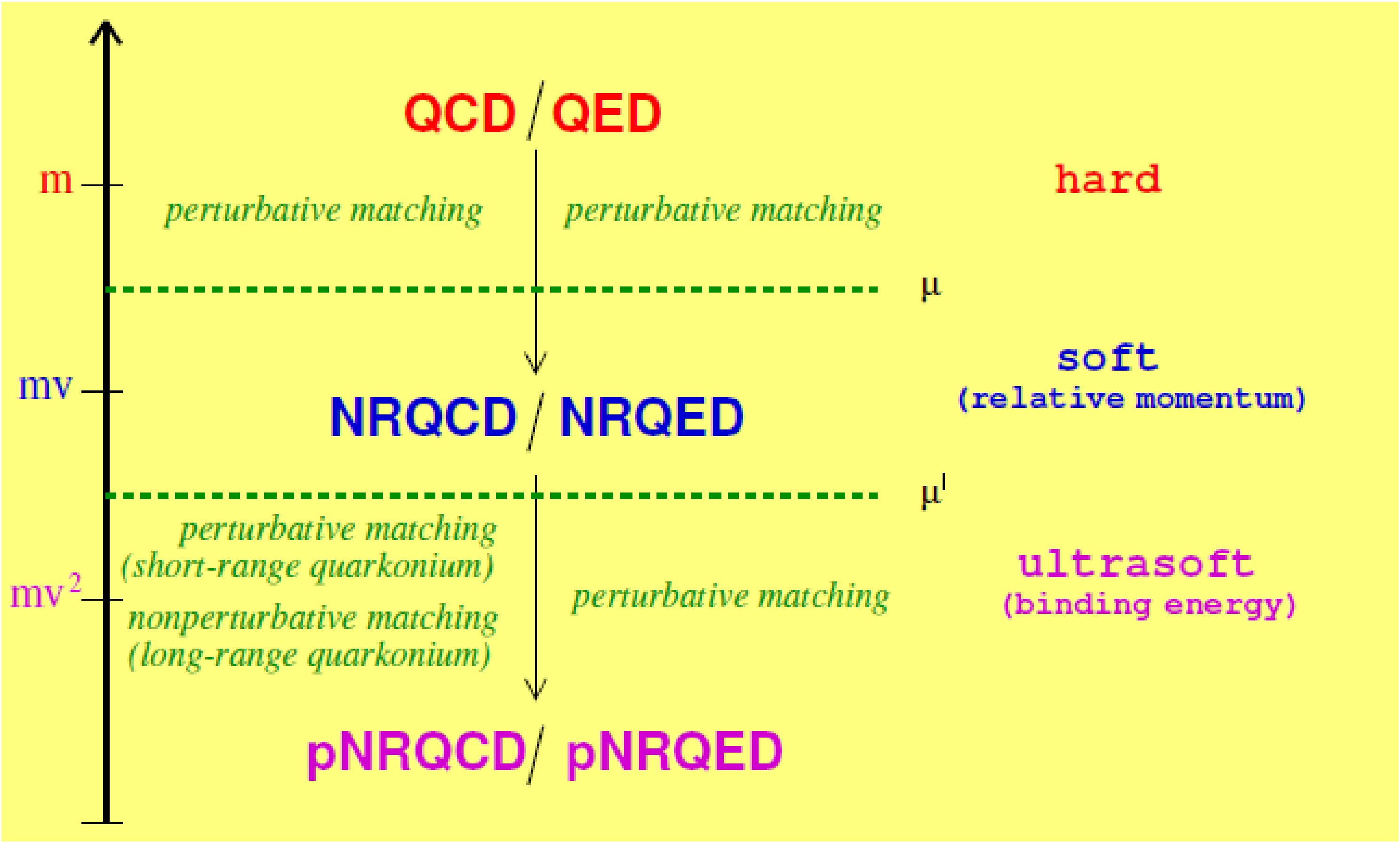
$$v_b^2 \approx 0.1,\ \ v_c^2 \approx 0.3$$

The dynamic scales are:

$$r \approx 1/mv,\ E \approx mv^2,\ v \ll 1$$

Normalized with respect to Λ_{QCD} **and** $\chi_c(1P)$

Non-relativistic EFT



In QCD another scale is relevant: Λ_{QCD}

Quarkonium and pair quarkonium production

- **The Fock state expansion for quarkonium could be organized as power expansion in v. For J/ψ we have**

$$\left|J/\psi\right\rangle = \left|c\bar{c}\left({}^3S_1,\underline{\mathbf{1}}\right)\right\rangle + O(v)\left|c\bar{c}\left({}^3P_J,\underline{\mathbf{8}}\right)g\right\rangle + O(v^2)\left|c\bar{c}\left({}^1S_0,\underline{\mathbf{8}}\right)g\right\rangle + O(v^2)\left|c\bar{c}\left({}^3S_1,\underline{\mathbf{8}}\right)gg\right\rangle + O(v^4)$$

- **Then partonic production cross-sections are given by**

$$d\sigma(a+b\to H+X) = \sum_n d\hat{\sigma}(a+b\to Q\bar{Q}[n]+X)O^H[n]$$

$$d\sigma(a+b\to H_1+H_2+X) = \sum_{n_1,n_2} d\hat{\sigma}(a+b\to Q_1\bar{Q}_1[n_1]+Q_2\bar{Q}_2[n_2]+X)O^{H_1}[n_1]O^{H_2}[n_2]$$

Here $O^H[\mathrm{n}]$ - long distance matrix elements (LDME) and $n = {}^3S_1^{[1]},\ {}^1S_0^{[8]},\ {}^3S_1^{[8]},\ {}^3P_J^{[8]},\ \ldots$

- **In general one should consider SPS, DPS and eventually MPS**

NLO troubles

- **Large number of Feynman diagrams**
- **Reduction to scalar integrals (or sets of known integrals)**
- **Numerical instabilities (inverse Gram determinants, spurious phase-space singularities)**
- **Extraction of soft and collinear singularities (we need both virtual and real corrections)**

NLO automation

$$\sigma_{NLO} = \int_n \left(d\sigma^B + d\sigma^V + \int_1 d\sigma^A \right) + \int_{n+1} \left(d\sigma^R - d\sigma^A \right)$$

Monte Carlo Tools (MC) ➡ Tree-level contributions
Subtraction terms
Integration over phase-space

One-Loop Programs (OLP) ➡ The values of the virtual contributions (at each given phase-space point)

Strategies for full NLO automation

- **MC** controls **OLP** via Binoth Les Houches Accord Interface (BLHA)
- **OLP** is fully incorporated within **MC**

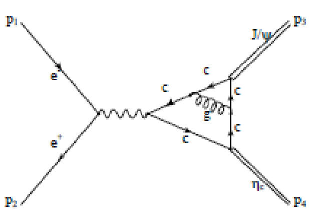
Feynman diagram generation and analysis

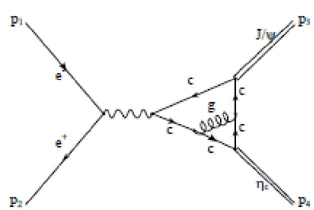
- **generation**

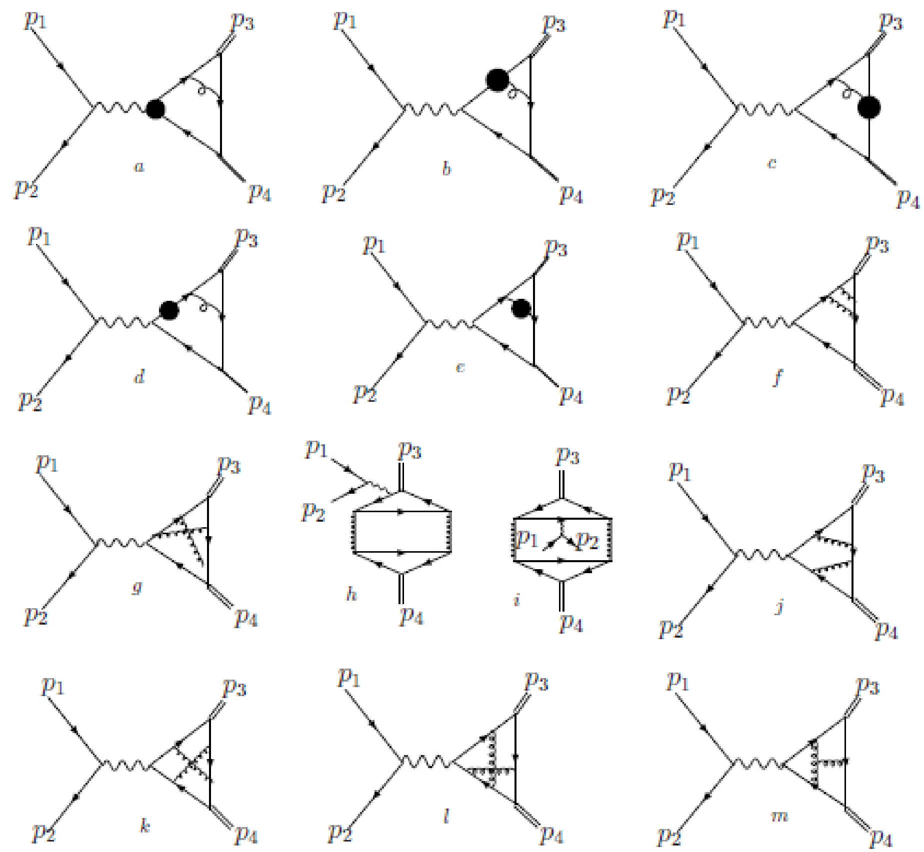
  Qgraf, Feynarts, private codes

- **analysis**

  Diana, Reduze, python code from GoSam, Feynarts, private codes







Covariant projectors

- **Color projectors**

$$C_1 = \frac{1}{\sqrt{2C_A}} \quad \text{for the singlet state}$$

$$C_8^c = \sqrt{2}T^c \quad \text{for the octet state}$$

- **Spin projectors**

$$\Pi_0 = \frac{1}{\sqrt{8m_Q^3}}\left(\frac{\not{P}}{2} - \not{q} - m_Q\right)\gamma_5\left(\frac{\not{P}}{2} + \not{q} + m_Q\right)$$

$$\Pi_1 = \frac{1}{\sqrt{8m_Q^3}}\left(\frac{\not{P}}{2} - \not{q} - m_Q\right)\gamma^\alpha\left(\frac{\not{P}}{2} + \not{q} + m_Q\right)$$

- **Matrix elements**

$$M_{Q\bar{Q}}\left[{}^1S_0^{[8]}\right] = Tr\left[C_8^c\,\Pi_0\,A_{Q\bar{Q}[n]}\right]\Big|_{q=0} \qquad M_{Q\bar{Q}}\left[{}^3S_1^{[1]}\right] = \xi_\alpha Tr\left[C_1\,\Pi_1^\alpha\,A_{Q\bar{Q}[n]}\right]\Big|_{q=0}$$

$$M_{Q\bar{Q}}\left[{}^3S_1^{[8]}\right] = \xi_\alpha Tr\left[C_8^c\,\Pi_1^\alpha\,A_{Q\bar{Q}[n]}\right]\Big|_{q=0} \qquad M_{Q\bar{Q}}\left[{}^3P_J^{[8]}\right] = \xi_{\alpha\beta}^{(J)}\frac{d}{d\,q_\beta}Tr\left[C_8^c\,\Pi_1^\alpha\,A_{Q\bar{Q}[n]}\right]\Big|_{q=0} \qquad (J = 0,1,2)$$

ξ_α and $\xi_{\alpha\beta}^{(J)}$ – polarization vectors of $Q\bar{Q}$ state

Note: one can also use helicity projection method

Renormalization

$$m_Q^0 = Z_m m_Q$$

$$\psi^0 = \sqrt{Z_\psi}\psi$$

$$A_\mu^0 = \sqrt{Z_A} A_\mu$$

$$g_s^0 = Z_g g_s$$

$$\delta Z_m^{OS} = -\frac{3g_s^2}{16\pi^2} C_F C_\varepsilon \left[\frac{1}{\varepsilon_{UV}} + \frac{4}{3} \right] + O(\alpha_s^2)$$

$$\delta Z_\psi^{OS} = -\frac{g_s^2}{16\pi^2} C_F C_\varepsilon \left[\frac{1}{\varepsilon_{UV}} + \frac{2}{\varepsilon_{IR}} + 4 \right] + O(\alpha_s^2)$$

$$\delta Z_A^{OS} = \frac{g_s^2}{48\pi^2} (5C_A - 2n_{lf}) C_\varepsilon \left[\frac{1}{\varepsilon_{UV}} - \frac{1}{\varepsilon_{IR}} \right] + O(\alpha_s^2)$$

$$\delta Z_g^{\overline{MS}} = \frac{g_s^2}{16\pi^2} \left(-\frac{11}{6} C_A + \frac{1}{3} n_{lf}\right) C_\varepsilon \left[\frac{1}{\varepsilon_{UV}} + \ln\frac{\mu^2}{m_Q^2} \right] + O(\alpha_s^2)$$

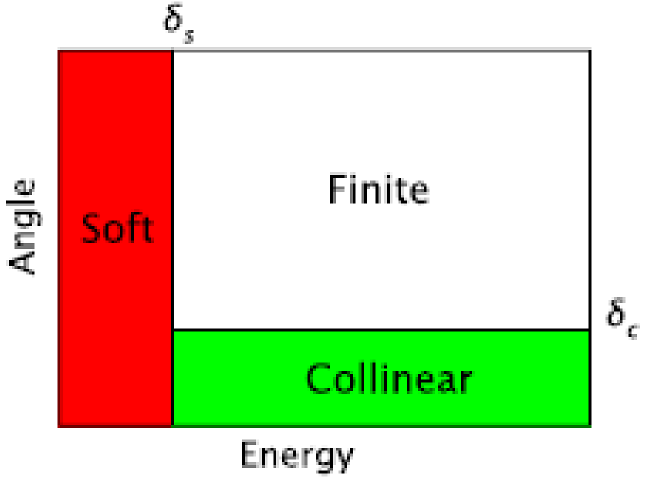
$$C_\varepsilon = \left(\frac{4\pi\mu^2}{m_Q^2} e^{-\gamma_E} \right)^\varepsilon$$

γ_5 is treated in t'Hooft-Veltman scheme: $\gamma_5 := \frac{i}{4!}\varepsilon_{\mu\nu\rho\sigma}\gamma^\mu\gamma^\nu\gamma^\rho\gamma^\sigma$

ε tensors are treated in the end in terms of 4-dimensional metric tensors $g^{\mu\nu}$

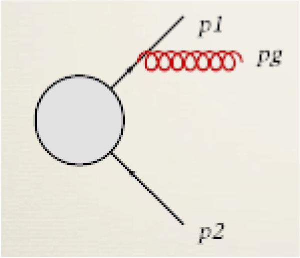
Infrared divergences

Dimensional regularization is used to regulate infrared divergences both in virtual and real emission corrections



The two cutoff phase space slicing method is used [B. Harris, J. Owens, 2002]

- **Eikonal approximation**



$$= \bar{u}^i(p_1)\left\{ i g_s \not{\varepsilon}_g^a T_{ij}^a \frac{i(\not{p}_1 + \not{p}_g)}{(p_1 + p_g)^2} \left[i M_0^{jk} \right] \right\} v^k(p_2)$$

$$\approx -g_s \frac{p_1 \cdot \varepsilon_g^a}{p_1 \cdot p_g} \bar{u}^i(p_1)\left\{ T_{ij}^a \left[i M_0^{jk} \right] \right\} v^k(p_2)$$

- **Collinear approximation** $p_1 \| p_g$

$$\left| M_1(p_1, p_2, p_g) \right|^2 \approx \frac{2}{s_{1g}} g_s^2 \mu^{2\varepsilon} P_{qq}(z, \varepsilon) \left| M_0(p_1 + p_g, p_2) \right|^2$$

$$P_{qq}(z, \varepsilon) = C_F \left[\frac{1 + z^2}{1 - z} - \varepsilon(1 - z) \right]$$

Diagram evaluation: mostly used methods

- **Passarino-Veltman reduction for tensor integrals**
- **Golem95 workout for vanishing Gramm determinants**
- **Partial fractioning and IBP reduction**
- **Integrand-level reduction**

Passarino-Veltman reduction

The generic one-loop integral is

$$T^{\mu_1\ldots\mu_p} \equiv \frac{(2\pi\mu)^{4-d}}{i\pi^2}\int d^dk\,\frac{k^{\mu_1}\ldots k^{\mu_p}}{D_0 D_1 D_2\ldots D_{n-1}}$$

Generally, these n-point tensor integrals can be reduced to simpler scalar integrals

$$B^{\mu}(r_{10}^2,m_0^2,m_1^2) = \frac{(2\pi\mu)^{4-d}}{i\pi^2}\int d^dk\,k^{\mu}\prod_{i=0}^{1}\frac{1}{(k+r_i)^2-m_i^2}$$

$$C^{\mu\nu}(r_{10}^2,r_{12}^2,r_{20}^2,m_0^2,m_1^2,m_2^2) = \frac{(2\pi\mu)^{4-d}}{i\pi^2}\int d^dk\,k^{\mu}k^{\nu}\prod_{i=0}^{2}\frac{1}{(k+r_i)^2-m_i^2}$$

$$\mathrm{A}_0(m_0^2) = \frac{(2\pi\mu)^{4-d}}{i\pi^2}\int d^dk\,\frac{1}{k^2-m_0^2}$$

$$B_0(r_{10}^2,m_0^2,m_1^2) = \frac{(2\pi\mu)^{4-d}}{i\pi^2}\int d^dk\,\prod_{i=0}^{1}\frac{1}{(k+r_i)^2-m_i^2}$$

$$C_0(r_{10}^2,r_{12}^2,r_{20}^2,m_0^2,m_1^2,m_2^2) = \frac{(2\pi\mu)^{4-d}}{i\pi^2}\int d^dk\,\prod_{i=0}^{2}\frac{1}{(k+r_i)^2-m_i^2}$$

The generic procedure is to write first form factor representation

$$C^{\mu\nu} = g^{\mu\nu}C_{00} + \sum_{i,j=1}^{2} r_i^{\mu} r_j^{\nu} C_{ij}$$

Then taking different convolutions get equations for C_{00} **and** C_{ij}

Tensor reduction with Golem95

Reduction to scalar integrals sensitive to $\det G$ [Binoth, Guillet, Heinrich, Pilon, Schubert, (2005)]

$$\int \frac{d^d k}{i\pi^{d/2}} \frac{p\cdot k}{(k+r_1)^2(k+r_2)^2(k+r_3)^2 k^2}$$

- If $\{r_1, r_2, r_3\}$ linearly independent: $p^\mu = \alpha_1 r_1^\mu + \alpha_2 r_2^\mu + \alpha_3 r_3^\mu + \alpha_\perp \varepsilon^{\mu\nu\rho\sigma} r_{1\nu} r_{2\rho} r_{3\sigma}$
- Since $2r_i \cdot k = (k+r_i)^2 - k^2 - r_i^2 \;\Rightarrow$ decomposition into scalar integrals
- Need to solve: (what introduces Gram determinant)

$$\begin{pmatrix} r_1\cdot r_1 & r_1\cdot r_2 & r_1\cdot r_3 & 0 \\ r_2\cdot r_1 & r_2\cdot r_2 & r_2\cdot r_3 & 0 \\ r_3\cdot r_1 & r_3\cdot r_2 & r_3\cdot r_3 & 0 \\ 0 & 0 & 0 & \det G \end{pmatrix} \cdot \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_\perp \end{pmatrix} = \begin{pmatrix} p\cdot r_1 \\ p\cdot r_2 \\ p\cdot r_3 \\ \varepsilon^{p r_1 r_2 r_3} \end{pmatrix}$$

Golem method: compute $\det G$ **and avoid reduction if too small**

- non-scalar integrals can become end-points of reduction
- Fast and stable numerical integration of these integrals (one-dimensional integrals in Feynman parameter representation)

Partial fractioning and IBP reduction

- **Partial fractioning of numerator and linear dependent denominators**
- **IBP relations**

$$\int \ldots \int d^d k_1 d^d k_2 \ldots \frac{\partial}{\partial k_i}\left[\frac{p_j\{k_j\}}{D_1^{a_1}\ldots D_n^{a_n}}\right] = 0$$

so that

$$\sum \alpha_i F(a_1 + b_{i,1}, \cdots, a_n + b_{i,n}) = 0, \qquad b_{i,j} = \{-1, 0, 1\}$$

where

$$F(a_1, \cdots, a_n) = \int \ldots \int \frac{d^d k_1 \cdots d^d k_h}{D_1^{a_1} \cdots D_n^{a_n}}$$

- **Codes: AIR, FIRE, Reduze, LiteRed, private codes**

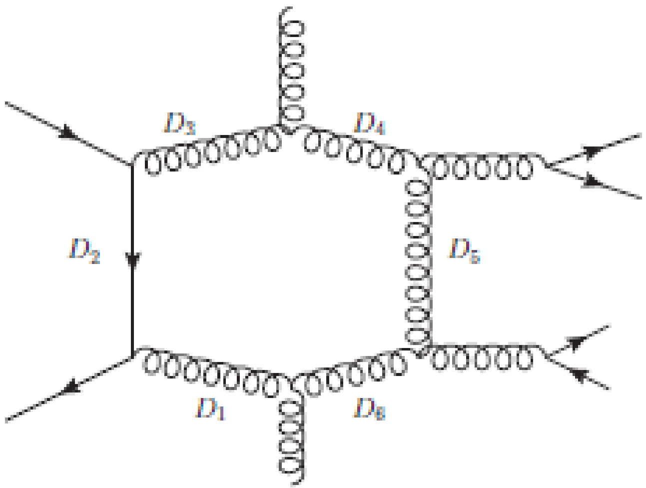
Integrand-level reduction

The integrand of a generic n-point one-loop integral :

- Is a rational function in the components of the loop momentum $\overline{q}$
- polynomial numerator N

$$M_n = \int d^d \overline{q}\, I_n, \qquad I_n \equiv \frac{N(\overline{q})}{D_1 \cdots D_n}$$

- quadratic polynomial denominators D_i
  (they correspond to Feynman loop propagators)



$$D_i = (\overline{q} + p_i)^2 - m_i^2$$

Integrand-level reduction

- **Every one-loop integrand can be decomposed as**

  [Ossola, Papadopoulos, Pittau (2007); Ellis, Giele, Kunst, Melnikov (2008)]

$$I_n = \frac{N}{D_1 \cdots D_n} = \sum_{j_1 \cdots j_5} \frac{\Delta_{j_1 j_2 j_3 j_4 j_5}}{D_{j_1} D_{j_2} D_{j_3} D_{j_4} D_{j_5}} + \sum_{j_1 \cdots j_4} \frac{\Delta_{j_1 j_2 j_3 j_4}}{D_{j_1} D_{j_2} D_{j_3} D_{j_4}}$$

$$+ \sum_{j_1 j_2 j_3} \frac{\Delta_{j_1 j_2 j_3}}{D_{j_1} D_{j_2} D_{j_3}} + \sum_{j_1 j_2} \frac{\Delta_{j_1 j_2}}{D_{j_1} D_{j_2}} + \sum_{j_1 j_2} \frac{\Delta_{j_1}}{D_{j_1}}$$

- **The residues** $\Delta_{j_1 \dots j_k}$
  - are polynomials in the components of $\bar{q}$
  - have a known universal parametric form
  - are parameterized by unknown process-dependent coefficients

! can be completely determined from a polynomial fit

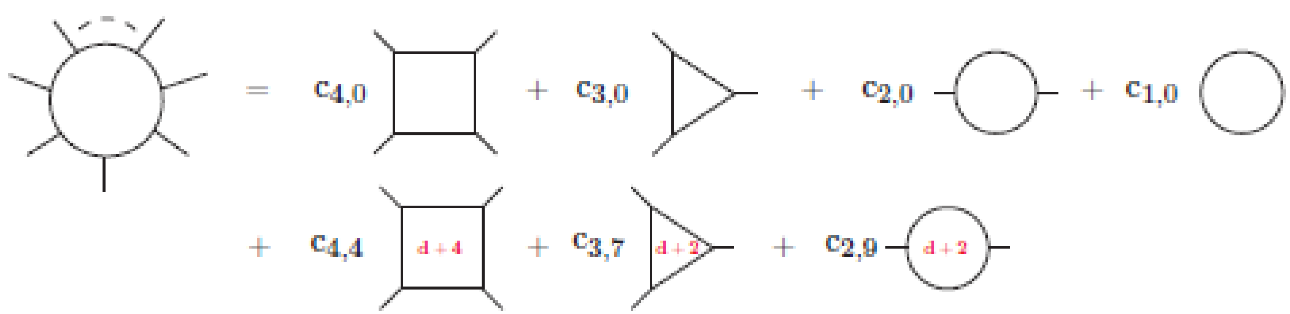
- **There is an extension to higher loops using multivariate polynomial division techniques** [Y.Zhang (2012), P.Mastrolia, E.Mirabella, G.Ossola, T.Peraro]

Integrand-level reduction

- **After integration**
  - Some terms vanish and do not contribute to the amplitude (spurious terms)
  - non-vanishing terms give master integrals (MI)
  - the amplitude is a linear combination of known MIs
- **The coefficients of this linear combination**
  - can be identified with some of the coefficients which parameterize the polynomial residues

! reduction to MI's ≡ polynomial fit of the residues

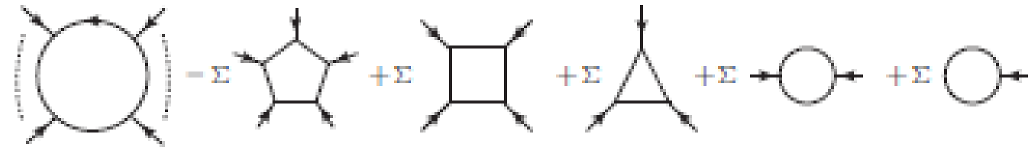
- **Any one-loop amplitude can be computed with a polynomial fit**

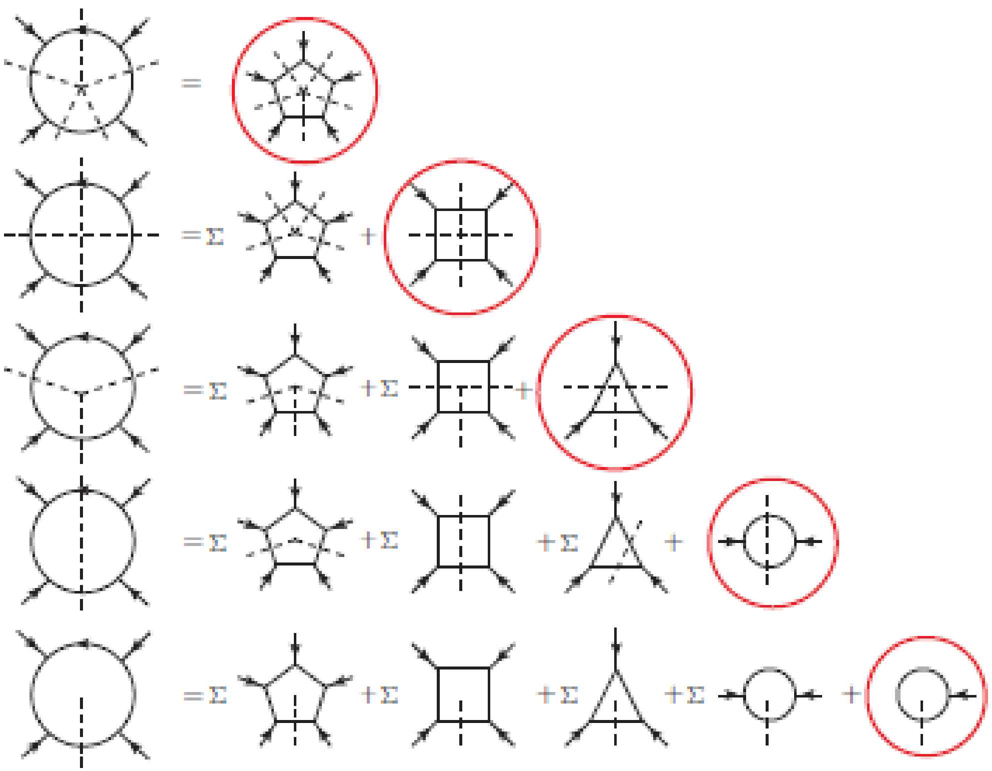


Fit on the cut at one-loop

[Ossola, Papadopoulos, Pittau (2007)]

Integrand decomposition





Fit-on-the cut

- fit m-point residues on m-ple cuts
- Cutting a loop propagator means

$$\frac{1}{D_i} \to \delta(D_i)$$

i.e. putting it on-shell

Integrand reduction via Laurent expansion

[P.Mastrolia, E.Mirabella, T.Peraro (2012)]

The integrand reduction via Laurent expansion:

- **fits residues by taking their asymptotic expansions on the cuts**
  - elaborating ideas first proposed by Forde and Badger
- **simplifications**
  - fewer coefficients needed
  - diagonal systems of equations for the coefficients
  - higher-point subtractions as corrections at the coefficient level
- **implemented in the semi-numerical C++ library Ninja**
  - Laurent expansion via simplified polynomial division algorithm
  - is a faster and more stable integrand reduction algorithm compared to Samurai or CutTools for example

Summary

- **The NLO code which implements both IBP and integrand reduction strategies for quarkonium production is under development**

- **It had passed some tests for $e^+ e^-$ pair quarkonium production**

- **Our wishlist includes production of single and pair quarkonia, B_c mesons and doubly heavy baryons both for $e^+ e^-$ collisions and hadroproduction with NLO accuracy within NRQCD framework**