Numerical Multi-Loop Calculations with pySecDec

Anton Olsson

Institute for Theoretical Physics,
Karlsruhe Institute of Technology

Recent pySecDec release: 2305.19768 (01.06.23)

The pySecDec collaboration:
Gudrun Heinrich, Stephen Jones, Matthias Kerner, Vitaly Magerya, AO, Johannes Schlenk

Previous pySecDec papers:

Comput. Phys. Commun. 273 (2022) 108267, [2108.10807]
Comput. Phys. Commun. 240 (2019) 120–137, [1811.11720]
Comput. Phys. Commun. 222 (2018) 313–326, [1703.09692]

Outline

Karlsruhe Institute of Technology

- Calculating scattering amplitudes - when and why are loop integrals evaluated?
- Where does pySecDec fit in?
- Sector decomposition and contour deformation
- Latest release
  - Results and new features

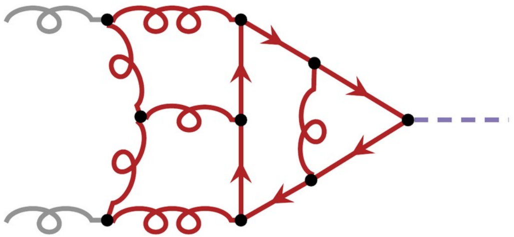
Motivation

- The standard model is tested experimentally to a high level of precision
- For several processes, predictions are or will be limited by theoretical uncertainties

$$\mathcal{A}^{\text{tot}} = \mathcal{A}^{\text{LO}} + \mathcal{A}^{\text{NLO}} + \mathcal{A}^{\text{NNLO}} + \mathcal{A}^{\text{N}^3\text{LO}} + \dots$$

- For Higgs production through gluon fusion (scale uncertainties): [CERN YR4]
  - NLO ~ 39%
  - NNLO ~ 20%
  - N^3LO (HTL) ~ 3%

N^3LO diagram for Higgs production



- If the (HL-)LHC reaches percent level precision, higher order contributions will be important

Calculating A Scattering Amplitude

- **Standard Workflow**
  - **Generation:** Write amplitude as sum of Feynman diagrams
  - **Projection:** Project tensor structures onto scalar integrals
  - **Reduction:** Use IBP relations to find a basis of master integrals
  - **Evaluation:** Evaluate the master integrals
    - Analytical methods struggle with many loops and kinematic scales, resort to numerical methods
- Numerical methods include
  - Sector Decomposition (**pySecDec**, Fiesta5)
  - Numerical Differential Equations (DiffExp, AMFlow)
  - Tropical Sampling (Feyntrop)

What Is pySecDec?

Karlsruhe Institute of Technology

- Documentation: ‘Toolbox for numerical evaluation of dimensionally regularized parameter integrals’
- Mainly: Sector decomposition and produces an integral library
  - Several sector decomposition routines (iterative and geometric)
  - Several integrators available (Monte Carlo methods)
- Also:
  - Expansion By Regions (EBR) handles difficult kinematic limits
  - Optimized for amplitude evaluation (weighted sums)

Sector Decomposition – Basic Example

- 2-dimensional parameter integral with overlapping singularities

$$I = \int_0^1 dx \int_0^1 dy x^{-1-a\varepsilon} y^{-b\varepsilon} (x+y)^{-1}$$

- Integration range is split up into two sectors where the variables are ordered

$$I = \int_0^1 dx \int_0^1 dy x^{-1-a\varepsilon} y^{-b\varepsilon} (x+y)^{-1} [\Theta(x-y) + \Theta(y-x)]$$

- The integration range is remapped to the unit square (unit hypercube, in general)
  - 1^{st} sector: $y = xt$
  - 2^{nd} sector: $x = yt$

Sector Decomposition – Basic Example

- Substituting $y = xt$ and $x = yt$ in the respective sectors:

$$I = \int_0^1 dx x^{-1-(a+b)\varepsilon} \int_0^1 dt t^{-b\varepsilon} (1+t)^{-1} +$$
$$\int_0^1 dy y^{-1-(a+b)\varepsilon} \int_0^1 dt t^{-1-a\varepsilon} (1+t)^{-1}$$

Sector Decomposition – Pole Extraction

- Simple logarithmic divergence

$$\int_0^1 dx\, x^{-1-(a+b)\varepsilon} = \frac{-1}{(a+b)\varepsilon}$$

- More general case: $I = \int_0^1 dt\, t^{-1-\varepsilon}\mathcal{I}(t,\varepsilon)$ Expand around t = 0:

$$I = \frac{-1}{\varepsilon}\mathcal{I}(0,\varepsilon) + \int_0^1 dt\, t^{-1-\varepsilon}[\mathcal{I}(t,\varepsilon) - \mathcal{I}(0,\varepsilon)]$$

Sector Decomposition – In General

- **Operates on**: Dimensionally regularized integrals in Feynman parameter space:

$$G(p^2, m^2, D) \sim \int_0^\infty \prod_{j=1}^{N} dx_j \delta(1 - \sum_{i=1}^{N} x_i) \frac{\mathcal{U}^{N-(L+1)D/2}}{\mathcal{F}(p^2, m^2)^{N-LD/2}}$$

- **Produces:** Laurent series in the regularization parameter:

$$G(p^2, m^2, D) \sim \sum_{\text{Sectors}} \sum_{n=-r}^{2L} C_n(p^2, m^2) \frac{1}{\varepsilon^n} + \mathcal{O}(\varepsilon^{r+1})$$

- Singularity structure completely factorized in the Laurent series!

Evaluating The Integrals

Karlsruhe Institute of Technology

- Need to integrate the coefficients in the epsilon-expansion

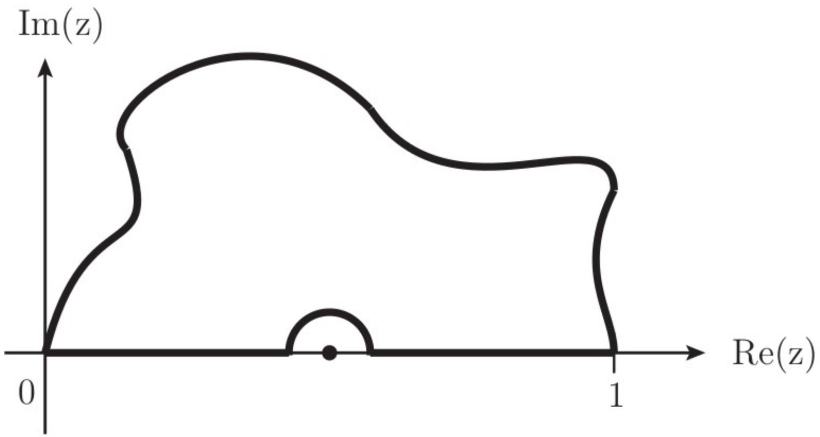
$$\sim \int_0^1 \prod_{j=1}^{N} dx_j \frac{1}{\mathcal{F}(\vec{x}, p^2, m^2)} = \int_0^1 \prod_{j=1}^{N} dx_j \mathcal{I}(\vec{x})$$

- Singularities at 0 are isolated by sector decomposition
- What if F vanishes inside the integration region? Contour deformation!
  - Cauchy’s theorem allows avoiding singularities by closing a contour

Contour Deformation

- Cauchy's theorem:

$$0 = \oint_c \prod_{j=1}^{N} \mathrm{d}z_j \mathcal{I}(\vec{z}) = \int_0^1 \prod_{j=1}^{N} \mathrm{d}x_j \, \mathcal{I}(\vec{x}) + \int_\gamma \prod_{j=1}^{N} \mathrm{d}z_j \, \mathcal{I}(\vec{z}(\vec{x}))$$



Contour Deformation

- Transform back to integration over the Feynman parameters

$$\int_0^1 \prod_{j=1}^{N} \mathrm{d}x_j \, \mathcal{I}(\vec{x}) = -\int_\gamma \prod_{j=1}^{N} \mathrm{d}z_j \, \mathcal{I}(\vec{z}) = \int_0^1 \prod_{j=1}^{N} \mathrm{d}x_j \; \det\left(\frac{\partial \vec{z}(\vec{x})}{\partial \vec{x}}\right) \mathcal{I}(\vec{z}(\vec{x}))$$

- How to chose the parametrization $\vec{z}(\vec{x})$?
  - Preserve the causal $i\delta$ prescription
  - Make sure no poles are enclosed by the contour
  - Some choices yield better numerical stability – not fully understood yet

Parametrization Example

- Simple choice that fulfills these requirement (returning to F)

$$z_k(\vec{x}) = x_k - i\,\tau_k(\vec{x})\;,\;\;\tau_k = \lambda\, x_k(1-x_k)\,\frac{\partial \mathcal{F}(\vec{x})}{\partial x_k}$$

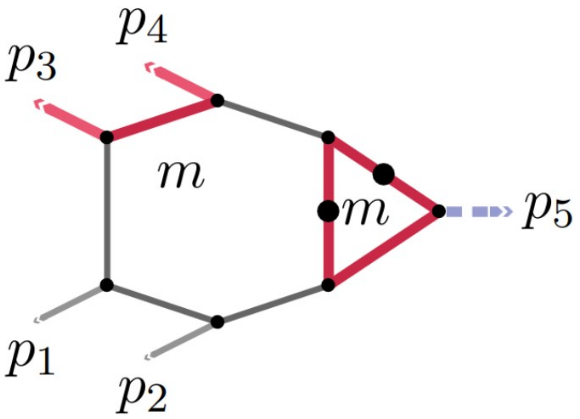
- Expanding around $\lambda = 0$

$$\mathcal{F}(\vec{z}(\vec{x})) = \mathcal{F}(\vec{x}) - i\,\lambda \sum_j x_j(1-x_j)\,\left(\frac{\partial \mathcal{F}}{\partial x_j}\right)^2 + \mathcal{O}(\lambda^2)$$

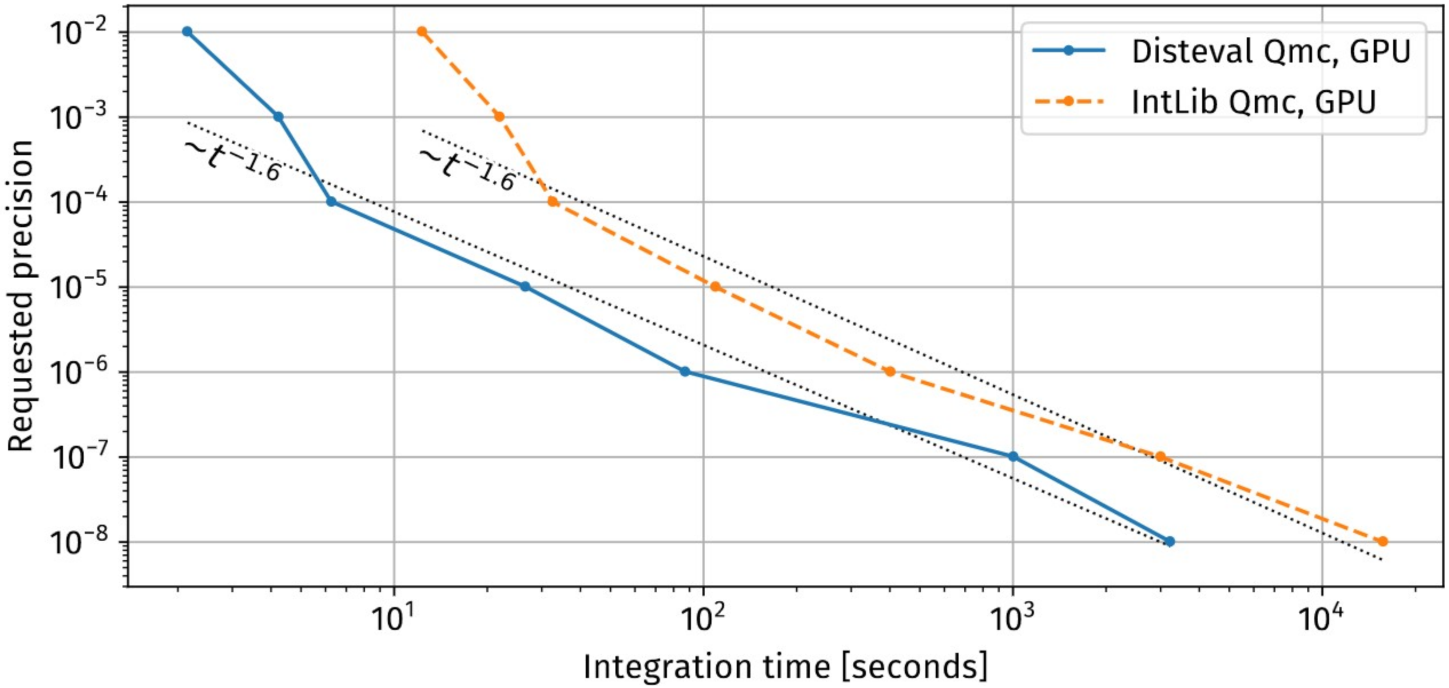
Latest Updates In pySecDec

- New QMC integrator called Disteval (distributed evaluation)
  - Up to an order of magnitude performance increase
- Median generating vectors for the QMC lattice rules
  - Avoid ‘unlucky lattices’
- Automated detection and insertion of necessary extra regulators in EBR
  - Resolves spurious singularities introduced by EBR
- More practical amplitude evaluation

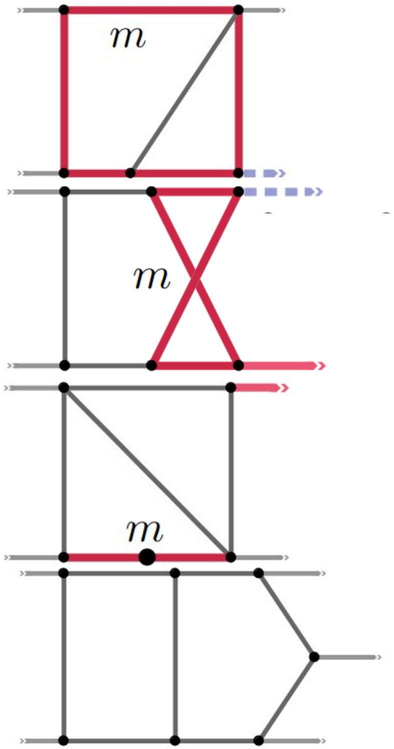
Performance Improvements



- 2-loop ‘Hexa-triangle’
- 5-point
- Massive propagators
- Massive legs



More Benchmarks



Accuracy:	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}
DISTEVAL	1.6 s	1.5 s	1.7 s	1.9 s	4.0 s	19 s	7.6 m
INTLIB	3.1 s	4.8 s	4.9 s	7.3 s	13.8 s	53 s	4.3 m
Ratio	1.9	3.1	2.8	3.9	3.4	2.9	0.6
DISTEVAL	5 s	5 s	9 s	37 s	2.3 m	5.4 m	27.1 m
INTLIB	9 s	17 s	41 s	163 s	9.6 m	16.0 m	27.3 m
Ratio	1.8	3.4	4.6	4.4	4.2	3.0	1.0
DISTEVAL	8 s	16 s	23 s	40 s	2.4 m	9.1 m	19.9 m
INTLIB	24 s	73 s	223 s	6.6 m	25.6 m	43.3 m	92.8 m
Ratio	3.0	4.6	9.7	9.9	10.5	4.8	4.7
DISTEVAL	5 s	8 s	11 s	0.71 m	3.7 m	18.5 m	1.1 h
INTLIB	45 s	65 s	88 s	3.2 m	11.3 m	74.8 m	4.6 h
Ratio	8.6	7.9	7.7	4.5	3.1	4.0	4.2

Summary

- pySecDec evaluates dimensionally regularized parameter integrals
  - Needed for higher order scattering amplitudes
- Integrates numerically by implementing
  - Sector decomposition and contour deformation
  - Integration libraries based on QMC methods
- Latest release has brought performance increases up to an order of magnitude, and several other features

Summary

- pySecDec evaluates dimensionally regularized parameter integrals
  - Needed for higher order scattering amplitudes
- Integrates numerically by implementing
  - Sector decomposition and contour deformation
  - Integration libraries based on QMC methods
- Latest release has brought performance increases up to an order of magnitude, and several other features

Thank You!