

Numerical evaluation of multi-loop integrals

Stephan Jahn*

Max-Planck-Institut für Physik, München, Germany

E-mail: sjahn@mpp.mpg.de

We present updates on the development of `pySECDEC`, a toolbox to numerically evaluate parameter integrals in the context of dimensional regularization. We discuss difficulties with loop integrals in the special kinematic condition where the squared momentum of a leg is equal to the squared mass of a propagator. We further discuss some features of a Quasi Monte Carlo (QMC) integrator that can optionally run on Graphics Processing Units (GPUs).

Loops and Legs in Quantum Field Theory (LL2018)

29 April 2018 - 04 May 2018

St. Goar, Germany

*Speaker.

1. Introduction

Current and future searches for New Physics at the LHC are focused on precision measurements. Higher order calculations are crucial for the interpretation of the data. A well established procedure for multi-loop calculations is: (i) draw all contributing Feynman diagrams, (ii) insert the Feynman rules to produce algebraic expressions, (iii) identify the loop integrals to be computed, (iv) reduce these integrals to a set of master integrals, and (v) solve the master integrals.

Sector decomposition [1, 2] provides an automated way to evaluate the master integrals numerically. In this article, we present recent developments in `pySECDDEC` [3], the successor of the program `SECDDEC` [4–6]. Other public sector decomposition tools are `sector_decomposition` [7] supplemented with `CSectors` [8] and `FIESTA` [9–12].

Given the amplitude in terms of master integrals, the major bottleneck after factorization of the poles in the dimensional regulator is the numerical integration. Suitable numerical integration algorithms are crucial to achieve reliable results in an acceptable time frame. The benefits of Quasi Monte Carlo (QMC) integration (see e.g. [13]) in combination with sector decomposition and running the numerical integration on Graphics Processing Units (GPUs) have first been pointed out in [14]. The method has also been successfully applied to phenomenological calculations, e.g. Higgs-boson pair production [15, 16] and H+jet production [17] at NLO. Considering these successes as a proof of principle, we announce an upcoming version of `pySECDDEC` with a builtin Quasi Monte Carlo integrator that can optionally run on nvidia GPUs using CUDA.

We further address problems that can occur when integrals in special kinematic configurations are considered. There are integrals that could not be dealt with by either of the public sector decomposition programs [18]. The simplest possible example that triggers the problem is a regulated $x - y$ in the denominator of the integrand, where x and y are integration variables. We therefore refer to it as the “ $x - y$ ” problem.

The structure of this article is as follows: The upcoming Quasi Monte Carlo integrator is described in section 2. We describe the previously introduced “ $x - y$ ” problem and its remedy in `pySECDDEC` in section 3 before we conclude in section 4.

2. Quasi Monte Carlo

Consider the d dimensional integral I_d ,

$$I_d[f] \equiv \int_0^1 dx_1 \dots \int_0^1 dx_d f(\mathbf{x}), \quad \mathbf{x} = (x_1, \dots, x_d) \quad (2.1)$$

of a function $f : [0, 1]^d \rightarrow \mathbb{K}$, $\mathbb{K} \in \{\mathbb{R}, \mathbb{C}\}$ over the unit hypercube. Following [14], we use shifted rank-1 lattice rules to obtain an unbiased estimate of I_d from n fixed sampling points $\{\mathbf{x}_j : \mathbf{x}_j \in [0, 1]^d, j = 1, \dots, n\}$ and m random shifts $\{\Delta_k : \Delta_k \in [0, 1]^d, k = 1, \dots, m\}$ as

$$I_d[f] \approx \bar{Q}_{d,n,m}[f] \equiv \frac{1}{m} \sum_{k=1}^m Q_{s,n}^{(k)}[f], \quad Q_{d,n}^{(k)}[f] \equiv \frac{1}{n} \sum_{j=1}^n f \left(\left\{ \frac{(j-1)\mathbf{g}}{n} + \Delta_k \right\} \right), \quad (2.2)$$

where $\mathbf{g} \in \mathbb{N}^d$ is the so-called generating vector and $\{\cdot\}$ means taking the fractional part. An unbiased error estimate is given by

$$\sigma_{d,n,m}^2[f] \equiv \text{Var}[\bar{Q}_{d,n,m}[f]] \approx \frac{1}{m(m-1)} \sum_{k=1}^m (Q_{s,n}^{(k)}[f] - \bar{Q}_{s,n,m}[f])^2. \quad (2.3)$$

An extensive review of the QMC method including proofs of the theorems we use in the following paragraphs can be found e.g. in [13].

It is proven that the error of the QMC estimate of the integral asymptotically scales with the number of lattice points n as $\mathcal{O}(1/n^{s+1})$, where s is the smoothness of the periodically continued integrand¹

$$\tilde{f} \in C^s(\mathbb{R}^d \rightarrow \mathbb{K}), \quad \tilde{f}(\mathbf{x}) \equiv f(\{\mathbf{x}\}). \quad (2.4)$$

Sector decomposed functions are typically infinitely smooth (any derivative is well-defined and finite) but not periodic. However, there are a number of periodizing transforms implemented via a substitution of the integration variables $\mathbf{x} \rightarrow \mathbf{z}(\mathbf{x})$. In these proceedings, we consider the Korobov transform of weight α ,

$$z_j(x_j) = \int_0^{x_j} du \omega(u), \quad \omega(u) = \frac{u^\alpha (1-u)^\alpha}{\int_0^1 du' u'^\alpha (1-u')^\alpha}, \quad (2.5)$$

and the baker's transform,

$$z_j(x_j) = 1 - |2x_j - 1|. \quad (2.6)$$

The Korobov transform forces the integrand to go to zero everywhere on the boundary. Assuming a C^∞ nonperiodic integrand f and an integer weight $\alpha \in \mathbb{N}$, the Korobov transformed and periodically continued integrand $\tilde{f}^{K\alpha}$ is $C^{\alpha-1}$ for odd α and C^α for even α . This can be seen by examining the derivative of $\tilde{f}^{K\alpha}$ on the border of the unperiodized transformed integrand $f^{K\alpha}$: One finds that the periodization of a high-enough derivative becomes discontinuous. Hence, we expect the relative error of sector decomposed integrals to asymptotically scale like $\mathcal{O}(1/n^\alpha)$ (odd α) or $\mathcal{O}(1/n^{\alpha+1})$ (even α) with the number of QMC lattice points n .

The baker's transform periodizes the integrand by mirroring rather than forcing it to a particular value on the integration boundary. Note that the baker's transform is only C^0 which naively suggests an asymptotic $\mathcal{O}(1/n)$ scaling. However, the asymptotic scaling of the error can be proven to be $\mathcal{O}(1/n^2)$ by considering the baker's transform as a modification of the lattice rather than an integral transform. Further note that the derivative of the baker's transform (where it exists) is plus or minus 2, which leads to a (piecewise) constant Jacobian factor when considered as an integral transform.

As an example integral, we consider the 2loop nonplanar box depicted in Figure 1 which appears as a master integral in two Higgs boson production [15, 16]. The number of lattice points n against the relative error of the real finite part is shown in Figure 2.

It turns out that the overall scaling of the error is better than $\mathcal{O}(1/n)$ in all cases. However, the scaling is very noisy in the sense that lattices of similar size can lead to error estimates that

¹In fact, the s th derivative of \tilde{f} must be *absolutely continuous*, not only *continuous* as usually suggested by the notation C^s .

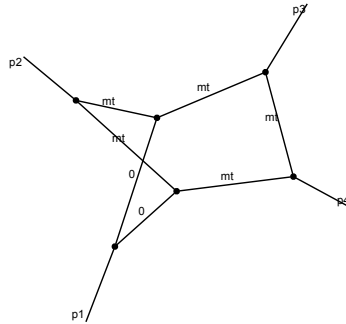


Figure 1: A massive nonplanar double box that appears as a master integral in the gluon channel of two Higgs boson production [15, 16].

differ by orders of magnitude. We checked this to be an intrinsic property of the lattice applied to a particular function, i.e. it is *not* due to statistical fluctuations when estimating the error. We further observe that the overall scaling may improve with increasing lattice size n . For example, if the double-box integral depicted in Figure 1 is computed with a Korobov transform of weight 6 then the scaling improves at $n \approx 10^6$, see Figure 1.

Due to the noise, it is hard to measure the scaling. Fitting a single line over the entire range does not make sense because the scaling clearly changes at certain values of n . Fitting lines to only part of the points would lead to somewhat arbitrary results depending on the considered ranges. We therefore only plot lines with predefined scaling behavior for guidance - they are *not* to be understood as fits. It is therefore hard to judge if we see the Korobov transform of weight 6 scale as the theoretically expected $\mathcal{O}(1/n^7)$ in the tail of the distribution or not. We expect to observe a $\mathcal{O}(1/n^7)$ scaling with larger lattices which are out of reach for practical reasons such as the machine precision and the available compute power.

We further observe the scaling to worsen for higher dimensional integrals. Part of the problem seems to be the periodizing transforms [19]. As a test, we integrate the constant 1 using the Korobov transform with different weights in different dimensions. The results are shown in Figure 3. The baker's transform is not shown in that plot, because it would have zero error for any lattice since it does not add variance to the original (constant) integrand. As an interesting feature, we find that Korobov transforms of higher weight or higher dimensions show a plateau where the error is approximately independent of the lattice size n for small n . However, the error starts to decrease with the lattice size for large enough lattices. We further observe lower errors for lower weight Korobovs and relatively small lattices. This suggests that "high" dimensional integrals may in practice be integrated to higher accuracy with worse scaling transforms due to available compute power.

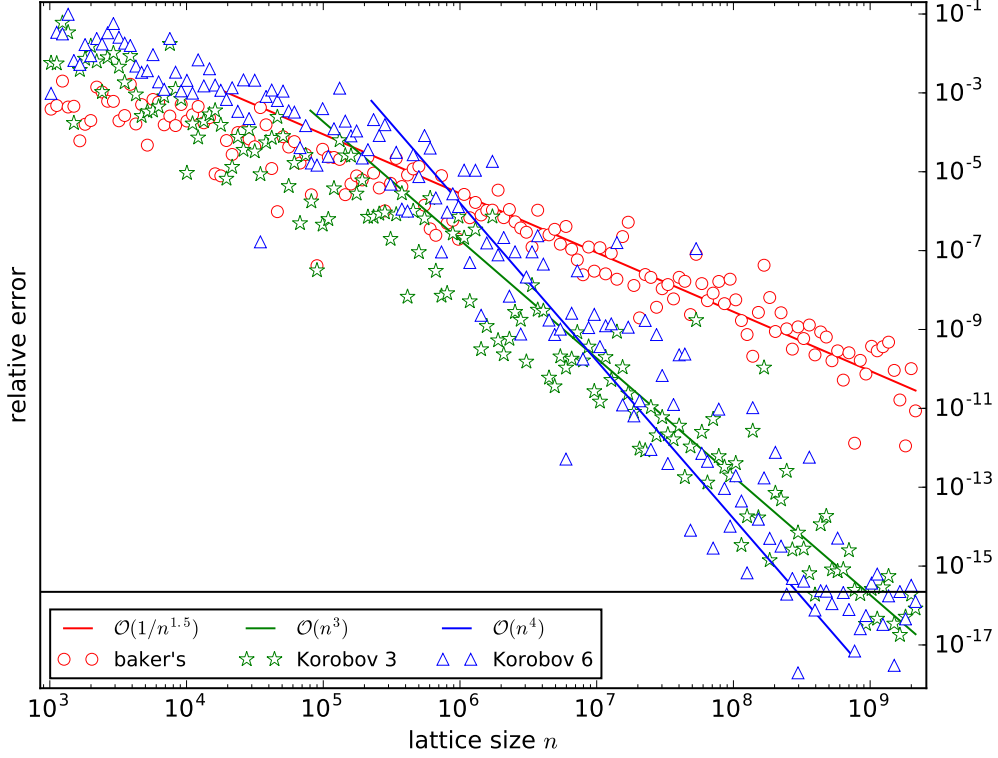


Figure 2: Scaling behavior of the real finite part of the double box integral depicted in Figure 1 integrated with the QMC method using different periodizing transforms. The red circles show the reported relative errors versus the number of points n in the QMC lattice using the baker’s periodizing transform. The observed scaling of the error roughly matches $\mathcal{O}(1/n^{1.5})$. The green stars show the reported relative errors of the same integral using a Korobov transform with weight 3. Due to the higher smoothness, a better scaling of $\mathcal{O}(1/n^3)$ is achieved. The blue triangles show the reported relative error of the integral using a Korobov transform with weight 6. The scaling matches about $\mathcal{O}(1/n)$ below $n \lesssim 10^6$ and improves to $\mathcal{O}(1/n^4)$ for $n \gtrsim 10^6$. The black line marks the precision of the floating point type (double) used in the setup to generate this plot.

3. The “ $x - y$ ” problem

Consider an integral of the form

$$\int_0^1 dx_1 \dots \int_0^1 dx_d \prod_{j=1}^m P_j(\{x\}, \{u\})^{a_j + b_j \varepsilon}, \quad (3.1)$$

where the P_j are polynomials of the integration variables x_j and additional parameters $\{u\}$ (e.g. Mandelstam invariants or masses) raised to a regulated power $a_j + b_j \varepsilon$ with numeric constants a_j and b_j . We particularly focus on Feynman parameterized loop integrals where the P_j correspond to

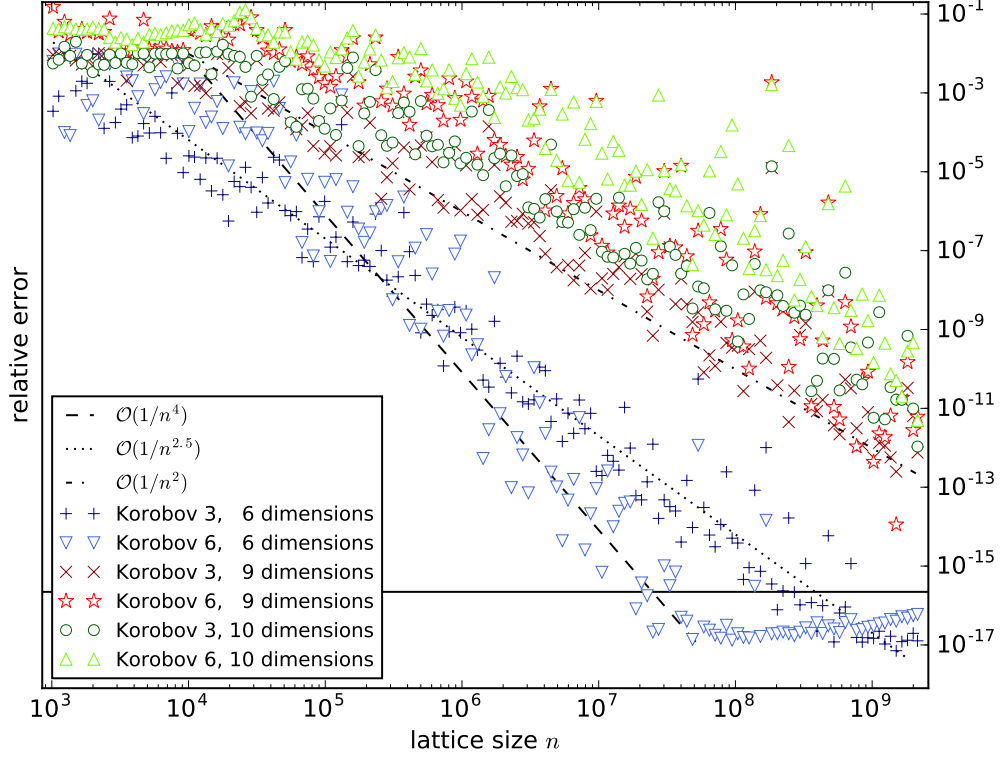


Figure 3: Scaling behavior of the Korobov transforms with weight 3 and 6 in different dimensions. The dashed, dotted, and dash-dotted lines show $\mathcal{O}(1/n^4)$, $\mathcal{O}(1/n^{2.5})$, and $\mathcal{O}(1/n^2)$ scaling normalized to arbitrary starting points. The blue pluses (triangles) show the reported error when integrating a constant using a Korobov with weight 3 (6) against the number of QMC lattice points n in 6 dimensions. The Korobov transform of weight 3 roughly shows a $\mathcal{O}(1/n^{2.5})$ to $\mathcal{O}(1/n^3)$ scaling starting from the smallest lattices $n \approx 10^3$ used. The Korobov transform of weight 6 exhibits a better $\mathcal{O}(1/n^4)$ scaling for $n \gtrsim 10^4$ but yields an error that is roughly independent of the lattice size for smaller lattices. A similar behavior is observed for both Korobov transforms, with weights 3 and 6, in 9 and 10 dimensions.

the Symanzik polynomials \mathcal{U} and \mathcal{F} [2]. Remember that singularities of the integrand originating from the borders of the integration domain, i.e. where any of the x_j is either zero or one², lead to $1/\varepsilon$ poles while singularities of the integrand in the interior of the integration domain are avoided by the Feynman prescription which is implemented via a deformation of the integration contour into the complex plane [5, 20–23].

Loop integrals usually have no singularities originating from $x_j = 1$: The \mathcal{U} polynomial is always positive semidefinite in the integration domain while the kinematic dependent \mathcal{F} polynomial

²Singularities at one can be introduced by integrating out the delta function or by the remappings of the sector decomposition procedure.

often has a so-called Euclidean region (all Mandelstam variables negative and all masses positive), where it is positive semidefinite as well. However, integrals without Euclidean region or a special choice of kinematics can lead to a vanishing \mathcal{F} polynomial after sector decomposition when an $x_j = 1$.

Let us first consider a simple toy integral that reproduces the problem,

$$\lim_{\delta \rightarrow 0} \int_0^1 dx \int_0^1 dy (x - y - i\delta)^{-2+\varepsilon}, \quad (3.2)$$

where the $-i\delta$ prescription is to be understood analogous to the Feynman prescription in loop integrals. The integrand has a regulated singularity along the line $x = y$. The $-i\delta$ shifts the singularity away from the real axis which regulates the integral in the interior of the integration domain. On the border, however, the limit $\delta \rightarrow 0$ forces the endpoints of the integral back to the real axis and thus into the singularity. Therefore, the singular endpoints $x = y = 0$ and $x = y = 1$ are purely regulated by the ε which becomes manifest as $1/\varepsilon$ poles.

Versions one to three of `SECDDEC` split the integration domain as

$$\int_0^1 dx_j = \int_0^{1/2} dx_j + \int_{1/2}^1 dx_j, \quad (3.3)$$

and then map the integration boundaries back to the unit interval by the substitutions $x_j = z_j/2$ and $x_j = 1 - z_j/2$, respectively. More details about the previous split can be found in the `SECDDEC-1` paper [4].

Although splitting all integration variables at $1/2$ for all integration variables seems a natural choice, that is exactly what causes unregulated singularities to remain even after the whole sector decomposition procedure. The reason is that the singular point $x = y = 1/2$ in the interior of the integration domain is mapped to an endpoint (all x_j equal to either one or zero) of the resulting integrals. However, the contour deformation vanishes at the endpoints by construction. The singular point $x = y = 1/2$ that should be avoided by a deformed contour is therefore left unregulated. Note that the contour deformation has to vanish on the endpoints of the original integral only. Consequently, one could in principle track which new endpoints are introduced by the split and allow for a nonzero deformation at those. One would then, however, have to match the nonzero deformation at these artificial endpoints between all split sectors. In particular, the split sectors could then no longer be considered independent of each other.

Instead, we can simply avoid mapping singular points to the border by splitting x and y elsewhere. In `pySECDDEC`, we generate random integers $r_j \in [1, 19]$ and split the integration domain as

$$\int_0^1 dx_j = \int_0^{r_j/20} dx_j + \int_{r_j/20}^1 dx_j \quad (3.4)$$

and remap the resulting integration borders back to the unit interval substituting $z_j = \frac{r_j}{20}x_j$ and $z_j = 1 - \frac{r_j}{20}x_j$, respectively. The singularity of the integrand at $x = y$ and possible resulting integral domains from the two different splitting procedures are depicted in Figure 4.

To summarize, unregulated singularities can remain in the sector decomposed integrals if the integration domain is split such that singular points of the integrand are mapped to endpoints of the integration. Consider for example the three-point function depicted in Figure 5 which is discussed

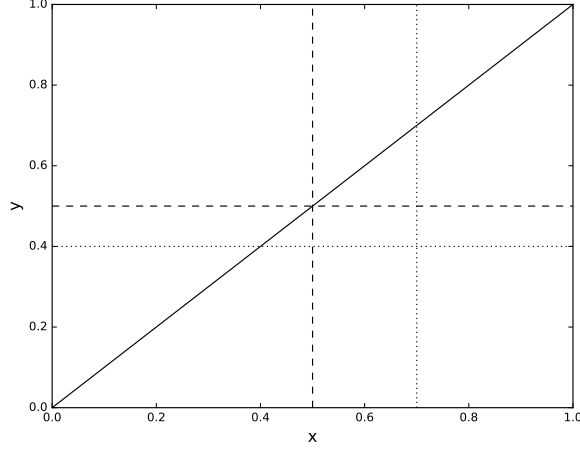


Figure 4: Graphical representation of the “ $x - y$ ” problem. The black line shows the singularity of then integrand at $x = y$. The dashed line shows a split at $x = y = 1/2$ which maps a singular point in the interior of the integration region to end points of the resulting regions. The dotted line shows how the integrand can be split without mapping a point of the singularity to the border of the resulting integrals.

in [18]. Note that the squared mass of the single massive propagator is equal to the Mandelstam invariant s . The \mathcal{F} polynomial can be expressed as,

$$\begin{aligned} \mathcal{F}/m_Z^2 = & x_3^2 x_5 + x_3^2 x_4 + x_2 x_3 x_5 + x_2 x_3 x_4 + x_1 x_3 x_5 + x_1 x_3 x_4 \\ & + x_1 x_3^2 + x_1 x_2 x_3 + x_0 x_3 x_4 + x_0 x_3^2 + x_0 x_2 x_3 \\ & - x_1 x_2 x_4 - x_0 x_1 x_5 - x_0 x_1 x_4 - x_0 x_1 x_2 - x_0 x_1 x_3. \end{aligned} \quad (3.5)$$

where m_Z is the mass of the Z-boson and the x_j are Feynman parameters.

It can easily be verified that \mathcal{F} vanishes for e.g. $x_0 = x_1 = x_2 = x_4 = x_5 = 1$ and $x_3 = 1/2$. A split of x_3 at $1/2$ maps that singularity to the endpoints of the split integrals as described above, which explains why this integral could not be evaluated with earlier versions of `SECDEC`.

4. Conclusion

We have presented the latest updates on the development of `pySECDEC`, a toolbox to numerically evaluate parameter integrals in the context of dimensional regularization.

We have resolved issues with integrals where the same kinematic invariant appears with opposite sign in the second Symanzik polynomial \mathcal{F} .

Several tests have shown that the numerical integration of sector decomposed functions can be dramatically sped up by Quasi Monte Carlo (QMC) integration and by the use of Graphics Processing Units (GPUs). It has been shown that evaluating the master integrals numerically is a viable option for phenomenological studies. Since the approach is general to loop integrals (and even beyond), numerically evaluating the sector decomposed master integrals may become an essential part of automated multi-loop amplitude generators.

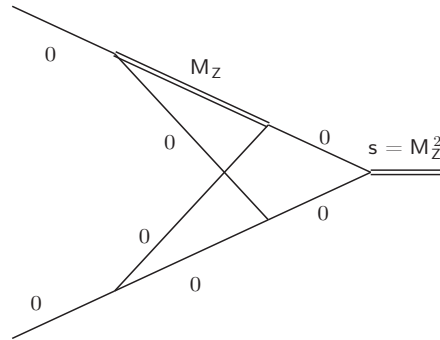


Figure 5: A vertex diagram where splitting the integration domain as denoted in equation 3.3 maps a singularity in the interior of the integration domain to an end point of the integration. Figure taken from [18].

Acknowledgments

I thank my fellow `SECDEC` team Sophia Borowka, Gudrun Heinrich, Stephen Jones, Matthias Kerner, Johannes Schlenk, and Tom Zirke for a successful collaboration.

References

- [1] T. Binoth and G. Heinrich, *An automatized algorithm to compute infrared divergent multiloop integrals*, *Nucl. Phys.* **B585** (2000) 741 [hep-ph/0004013].
- [2] G. Heinrich, *Sector Decomposition*, *Int. J. Mod. Phys.* **A23** (2008) 1457 [0803.4177].
- [3] S. Borowka, G. Heinrich, S. Jahn, S. P. Jones, M. Kerner, J. Schlenk et al., *pySecDec: a toolbox for the numerical evaluation of multi-scale integrals*, *Comput. Phys. Commun.* **222** (2018) 313 [1703.09692].
- [4] J. Carter and G. Heinrich, *SecDec: A general program for sector decomposition*, *Comput. Phys. Commun.* **182** (2011) 1566 [1011.5493].
- [5] S. Borowka, J. Carter and G. Heinrich, *Numerical Evaluation of Multi-Loop Integrals for Arbitrary Kinematics with SecDec 2.0*, *Comput. Phys. Commun.* **184** (2013) 396 [1204.4152].
- [6] S. Borowka, G. Heinrich, S. P. Jones, M. Kerner, J. Schlenk and T. Zirke, *SecDec-3.0: numerical evaluation of multi-scale integrals beyond one loop*, *Comput. Phys. Commun.* **196** (2015) 470 [1502.06595].
- [7] C. Bogner and S. Weinzierl, *Resolution of singularities for multi-loop integrals*, *Comput. Phys. Commun.* **178** (2008) 596 [0709.4092].
- [8] J. Gluza, K. Kajda, T. Riemann and V. Yundin, *Numerical Evaluation of Tensor Feynman Integrals in Euclidean Kinematics*, *Eur. Phys. J.* **C71** (2011) 1516 [1010.1667].

- [9] A. V. Smirnov and M. N. Tentyukov, *Feynman Integral Evaluation by a Sector decomposition Approach (FIESTA)*, *Comput. Phys. Commun.* **180** (2009) 735 [0807.4129].
- [10] A. V. Smirnov, V. A. Smirnov and M. Tentyukov, *FIESTA 2: Parallelizeable multiloop numerical calculations*, *Comput. Phys. Commun.* **182** (2011) 790 [0912.0158].
- [11] A. V. Smirnov, *FIESTA 3: cluster-parallelizable multiloop numerical calculations in physical regions*, *Comput. Phys. Commun.* **185** (2014) 2090 [1312.3186].
- [12] A. V. Smirnov, *FIESTA4: Optimized Feynman integral calculations with GPU support*, *Comput. Phys. Commun.* **204** (2016) 189 [1511.03614].
- [13] J. Dick, F. Y. Kuo and I. H. Sloan, *High-dimensional integration: The quasi-monte carlo way*, *Acta Numerica* **22** (2013) 133–288.
- [14] Z. Li, J. Wang, Q.-S. Yan and X. Zhao, *Efficient numerical evaluation of Feynman integrals*, *Chin. Phys.* **C40** (2016) 033103 [1508.02512].
- [15] S. Borowka, N. Greiner, G. Heinrich, S. Jones, M. Kerner, J. Schlenk et al., *Higgs Boson Pair Production in Gluon Fusion at Next-to-Leading Order with Full Top-Quark Mass Dependence*, *Phys. Rev. Lett.* **117** (2016) 012001 [1604.06447].
- [16] S. Borowka, N. Greiner, G. Heinrich, S. P. Jones, M. Kerner, J. Schlenk et al., *Full top quark mass dependence in Higgs boson pair production at NLO*, *JHEP* **10** (2016) 107 [1608.04798].
- [17] S. P. Jones, M. Kerner and G. Luisoni, *Next-to-Leading-Order QCD Corrections to Higgs Boson Plus Jet Production with Full Top-Quark Mass Dependence*, *Phys. Rev. Lett.* **120** (2018) 162001 [1802.00349].
- [18] I. Dubovyk, A. Freitas, J. Gluza, T. Riemann and J. Usovitsch, *30 years, some 700 integrals, and 1 dessert, or: Electroweak two-loop corrections to the $Z\bar{b}b$ vertex*, *PoS LL2016* (2016) 075 [1610.07059].
- [19] F. Y. Kuo, I. H. Sloan and H. Woźniakowski, *Periodization strategy may fail in high dimensions*, *Numerical Algorithms* **46** (2007) 369.
- [20] D. E. Soper, *Techniques for QCD calculations by numerical integration*, *Phys. Rev.* **D62** (2000) 014009 [hep-ph/9910292].
- [21] T. Binoth, J. P. Guillet, G. Heinrich, E. Pilon and C. Schubert, *An Algebraic/numerical formalism for one-loop multi-leg amplitudes*, *JHEP* **10** (2005) 015 [hep-ph/0504267].
- [22] C. Anastasiou, S. Beerli and A. Daleo, *Evaluating multi-loop Feynman diagrams with infrared and threshold singularities numerically*, *JHEP* **05** (2007) 071 [hep-ph/0703282].
- [23] J. Schlenk and T. Zirke, *Calculation of Multi-Loop Integrals with SecDec-3.0*, *PoS RADCOR2015* (2016) 106 [1601.03982].