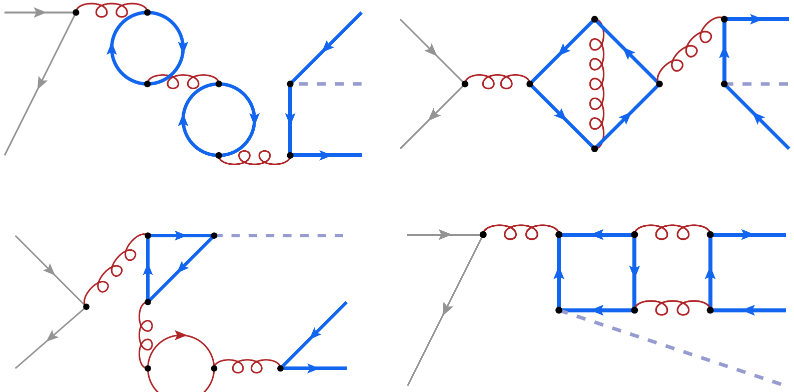
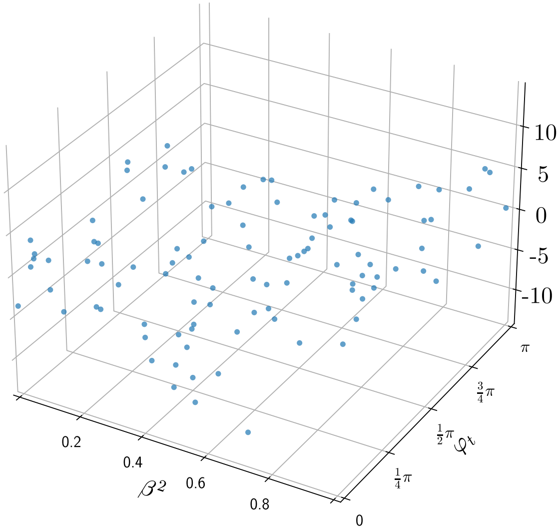
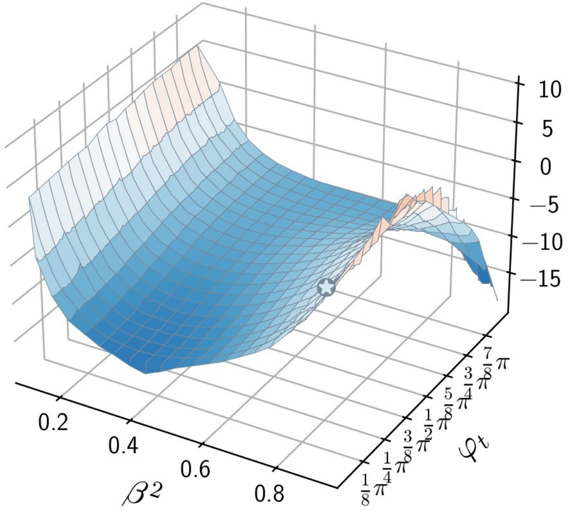
Towards a numerical evaluation of $q\bar{q} \to t\bar{t}H$ at two loops







Based on 2402.03301

Bakul Agarwal, Víctor Bresó-Pla, Gudrun Heinrich, Stephen Jones,
Matthias Kerner, Sven Yannick Klein, Jannis Lang, Vitaly Magerya, AO

Higgs 2024 – Uppsala

Anton Olsson

Outline

1. Introduction to $t\bar{t}H$ Production
2. Numerical Scattering Amplitudes
3. Constructing Amplitude Grids

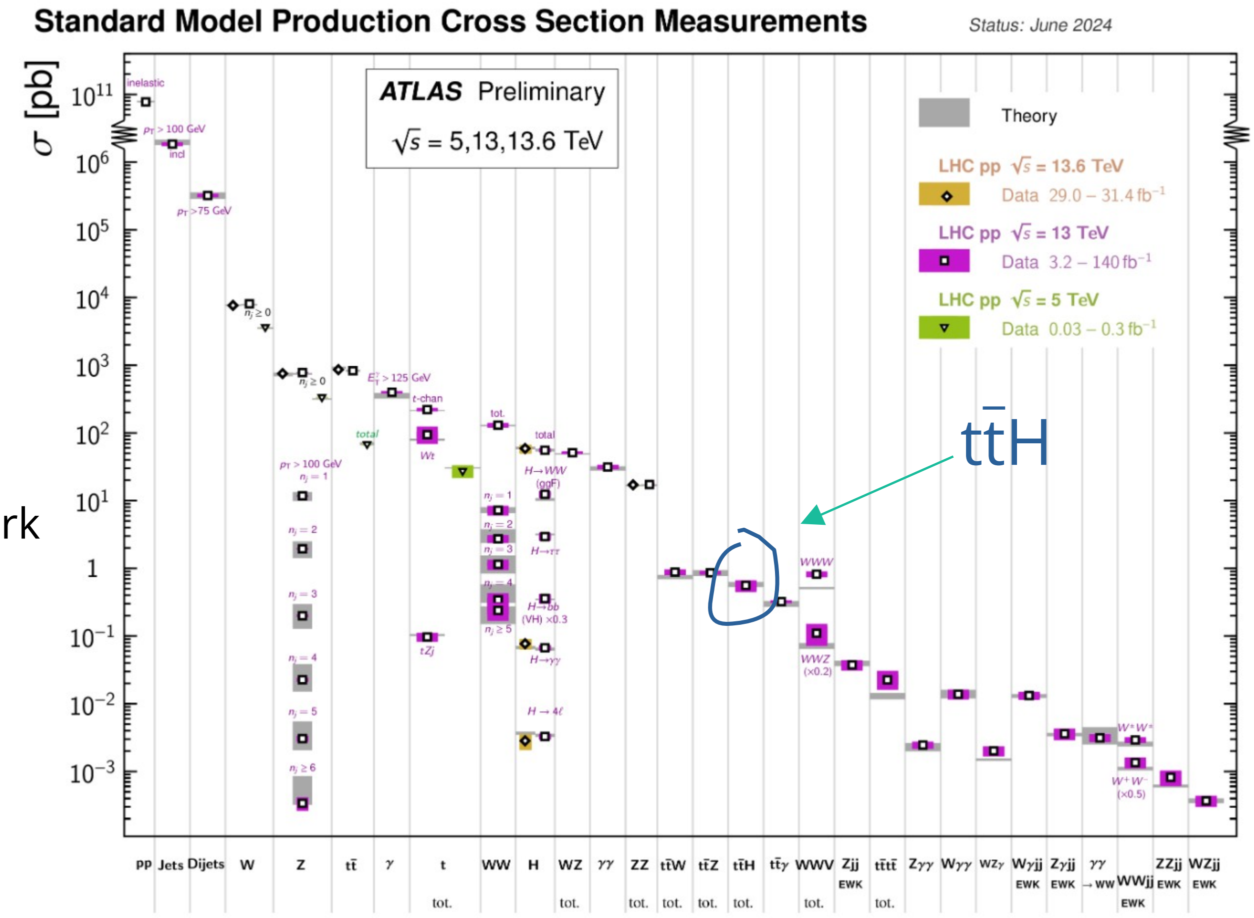
1. Introduction to $t\bar{t}H$ Production

Introduction to $t\bar{t}H$

- First observed in 2018
  - 1804.0261 (CMS), 1806.00425 (ATLAS)

- Only 1% of all Higgs bosons are produced with a top-quark pair

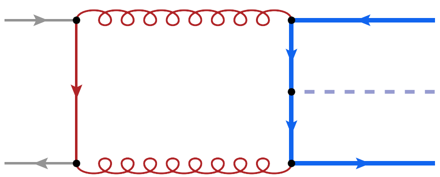
- Why is this an interesting process?
  - Direct sensitivity to the top-quark Yukawa coupling y_t
  - Probe CP properties of y_t
    2208.02686 (CMS), 2303.05974 (ATLAS)



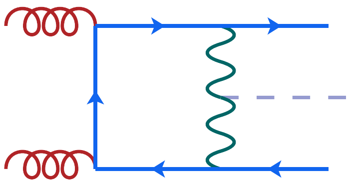
Plot: Ana Rosario Cueto Gómez; HP2 2024

Precision in $t\bar{t}H$

- Experimental uncertainty is currently dominated by statistics ~10-15%
  - But: projected statistical uncertainties for HL-LHC **~2%** [Les Houches '21; Snowmass '22]
- Calculation complete up to **NLO**
  - QCD @NLO [Dittmaier et al. '02, Dawson et al. '02]
  - EW @NLO [Frixione et al. '14]
  - QCD + EW @NLO [Denner et al. '17]
  - QCD @(NLO+NNLL) [Broggio et al. '15, Kulesza et al. '17]
  - QCD + EW @(NLO+NNLL) [Broggio et al. '19]
  - t→H fragmentation functions [Czakon et al. '21]
  - QCD @NLO to $\mathcal{O}(\varepsilon^2)$ [Tancredi et al. '23]



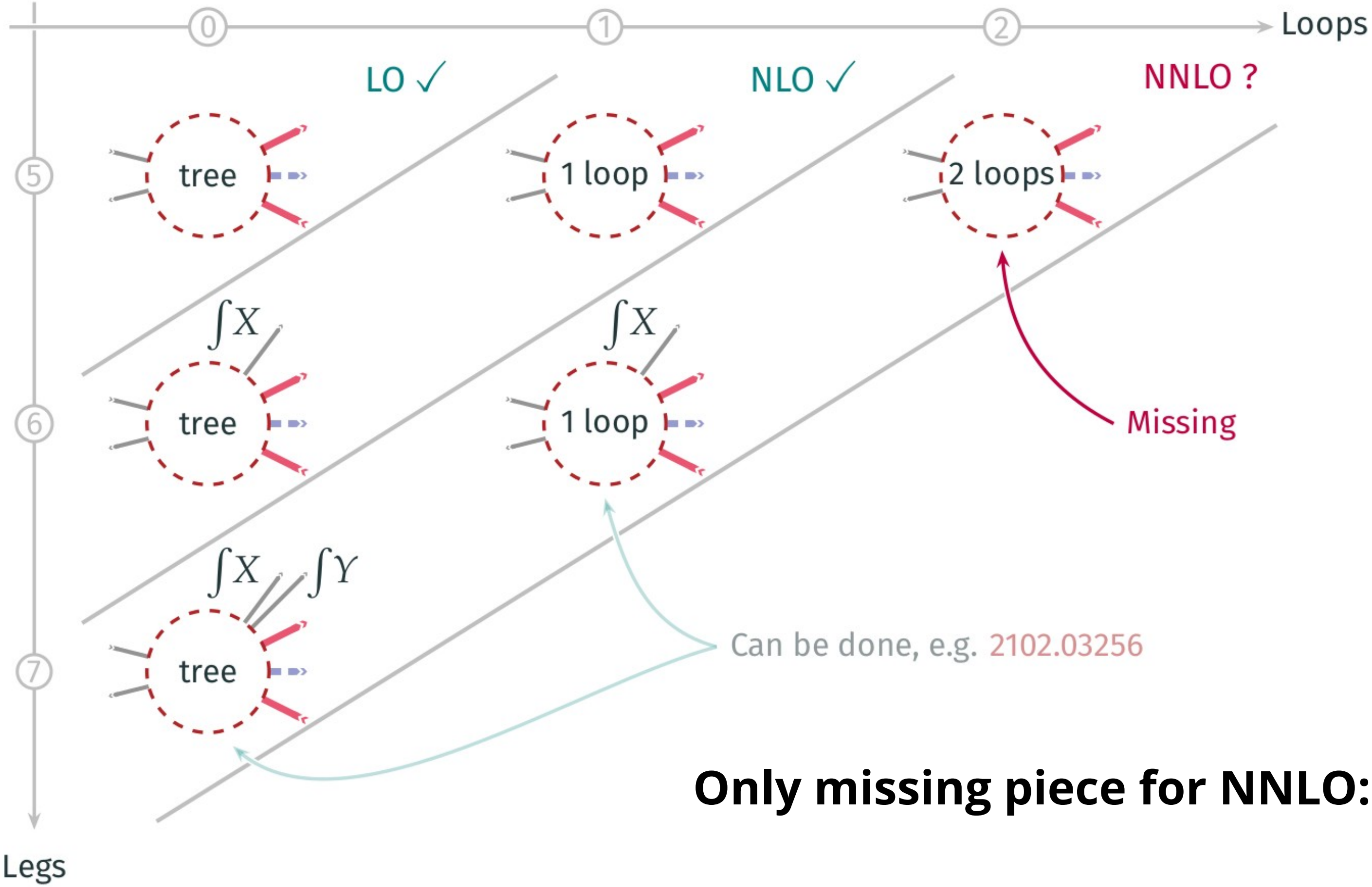
NLO QCD



NLO EW

- NLO QCD scale uncertainties **~10-15%:** NNLO calculation needed!

2 → 3 at NNLO

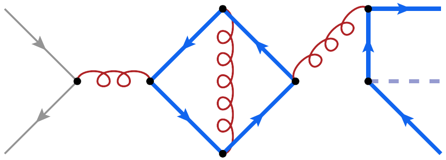


Only missing piece for NNLO: *double-virtual amplitudes*

Figure: Magerya; QCD@LHC 2024

Progress at NNLO

- **NNLO calculation** using **approximations** for the 2L amplitude
  - Soft Higgs approximation [Catani et al. '22] and massification [talk by Savoini at HP2 '24]
  - Highly accurate for total cross section ~1%
  - Expected to be less accurate in some kinematic regions
  - We want to validate this!



NNLO QCD

- IR-pole coefficients of the 2L amplitude [Wang et al. '22]
- Master integrals analytically with light quark-loops at leading color [Reina et al. '23]
- 2L amplitude in the high-energy boosted limit [Wang et al. '24]
- Quark-initiated 2L amplitude with light and heavy quark-loops, **numerically** [Heinrich et al. '24]
  - **Next step:** full quark-initiated amplitude

This talk

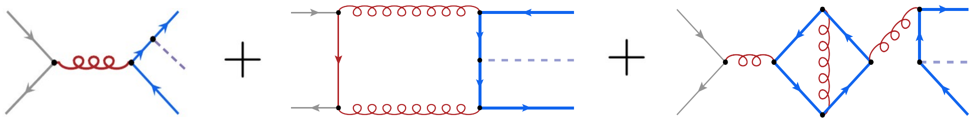
2. Numerical Scattering Amplitudes

Calculation Workflow I

1. Amplitude Generation:

- Generate all Feynman diagrams (702 2-loop quark-initiated QCD diagrams) [QGRAF]

$$\mathcal{A}_{\mathcal{V}} = \mathcal{A}_{\mathcal{V}}^{\rm LO} + \mathcal{A}_{\mathcal{V}}^{\rm NLO} + \mathcal{A}_{\mathcal{V}}^{\rm NNLO} + \cdots \sim$$



- Insert Feynman rules and apply projector [ALIBRARY], sum over tensor structures [COLOR.H, FORM]

$$\langle \mathcal{A}^0 | \mathcal{A} \rangle = \langle \mathcal{A}^0 | \mathcal{A}^0 \rangle + \left(\frac{\alpha_s}{2\pi}\right) \langle \mathcal{A}^0 | \mathcal{A}^1 \rangle + \left(\frac{\alpha_s}{2\pi}\right)^2 \langle \mathcal{A}^0 | \mathcal{A}^2 \rangle + \mathcal{O}(\alpha_s^3)$$

- The $\langle \mathcal{A}^0 | \mathcal{A}^2 \rangle$ piece contains **~90 000** scalar integrals

2. Reduction:

- Most integrals are **linearly dependent**: find relations through **Integration-By-Parts (IBP)** reduction
- Generate IBP system [KIRA], with **7 scales**: symbolic solutions *not feasible* currently
  - Resort to *numerical strategy*

Calculation Workflow II

- Substitute numbers for all 7 kinematics: $s_{12}, s_{23}, s_{34}, s_{45}, s_{51},\ m_H^2, m_t^2$
- Record an *execution trace* for the solution to the linear system of IBP relations [RATRACER]
  - We now have a recipe of arithmetic operations that solves our linear system
  - Replay the execution trace for each *phase space* point we need; O(minutes)
- **90 000** scalar integrals reduced to linear combination of **3005 master integrals (MIs)**
- Basis *not* unique: selecting a 'good' basis of MIs is a non-trivial trial and error process

3. Evaluation:

- Evaluate MIs numerically with pySecDec [Heinrich et al. '17, '18, '21, '23]
- Two step process:
  1. Sector decomposition: Isolate and extract singularities as expansion in the regulator, **do once**
  2. Quasi-Monte Carlo (QMC) integration, **do once for each phase space point**; O(minutes)

3. Constructing Amplitude Grids

Amplitude Grids

- Evaluate difficult amplitudes on a **point-by-point** basis
- Amplitudes are used to compute physical observables

$$\langle \mathcal{O} \rangle_\Phi = \int_{\bar{x} \in \Phi} \mathcal{O}(\bar{x}) \mathrm{d}\sigma(\bar{x})$$

$$\mathrm{d}\sigma = \frac{1}{2\hat{s}} |\mathcal{A}|^2 \mathrm{d}\Phi \, \mathrm{d}\rho_{a,b}(\hat{s})$$

Amplitude

PDFs

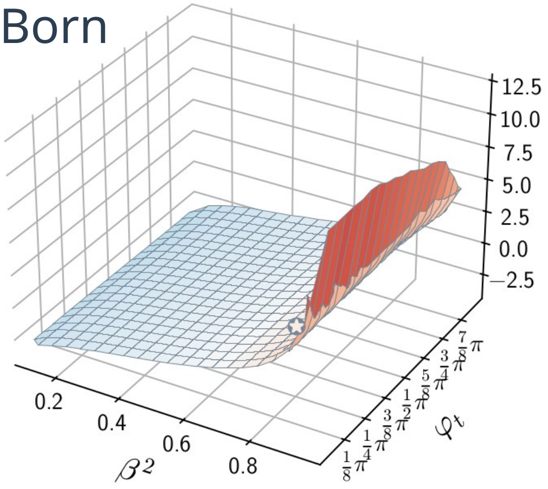
Phase space differential

- **Problem:** Integrals require millions of MC samples
  - Evaluation time of amplitude at 1 point ~ O(minutes)
  - Evaluation time of 1 observable ~ O(years)
- **Solution:** Evaluate amplitude at a few points and **interpolate** for values in between
  - This implies there will be interpolation/grid uncertainties
  - How do these uncertainties propagate to observables?

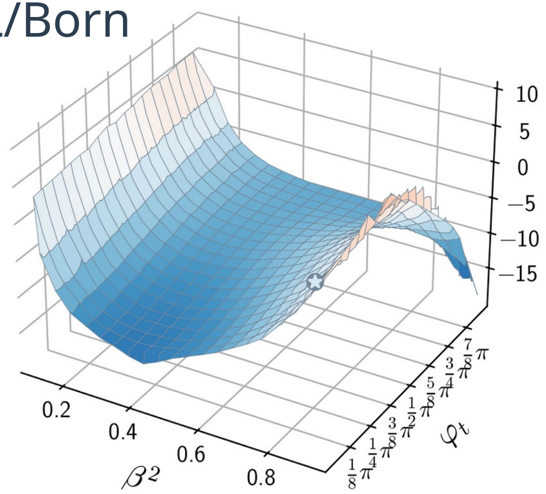
Why is this difficult?

- $t\bar{t}H$ amplitudes are:
  - High-dimensional
  - Complicated at higher loops
- Points with increasing dimension:
  - 2 final-state particles: 2 variables
  - 3 final-state particles: **5 variables**

1L/Born



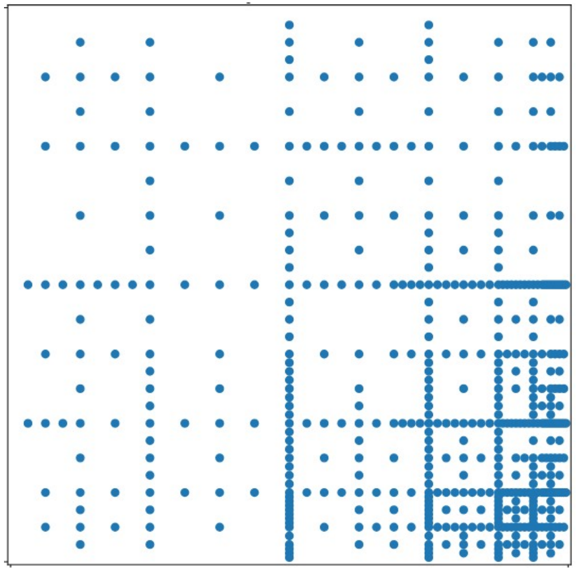
2L/Born



	$q\bar{q} \to t\bar{t}$	$q\bar{q} \to t\bar{t}H$
Points per dim \ Dimension	$\mathbf{d} = 2$	$\mathbf{d} = 5$
10	100	100000
20	400	3200000
30	900	24300000

The Interpolation Problem

- Our goal: approximate $a = |\mathcal{A}|^2$ with some $\tilde{a}$ defined on the whole phase space, based on the knowledge of a at some data points
  - When the approximation error of $\tilde{a}$ is “small enough”, the calculation is finished
- Two main questions:
  1. What should $\tilde{a}$ be? (polynomial, spline, neural network etc.)
  2. Where to evaluate a? (selecting interpolation nodes)
- Promising approximation methods we investigated:
  - Chebyshev polynomials
  - Spherical harmonics
  - Sparse grids
  - Spline interpolation
  - Neural networks (GATr) [2405.14806]

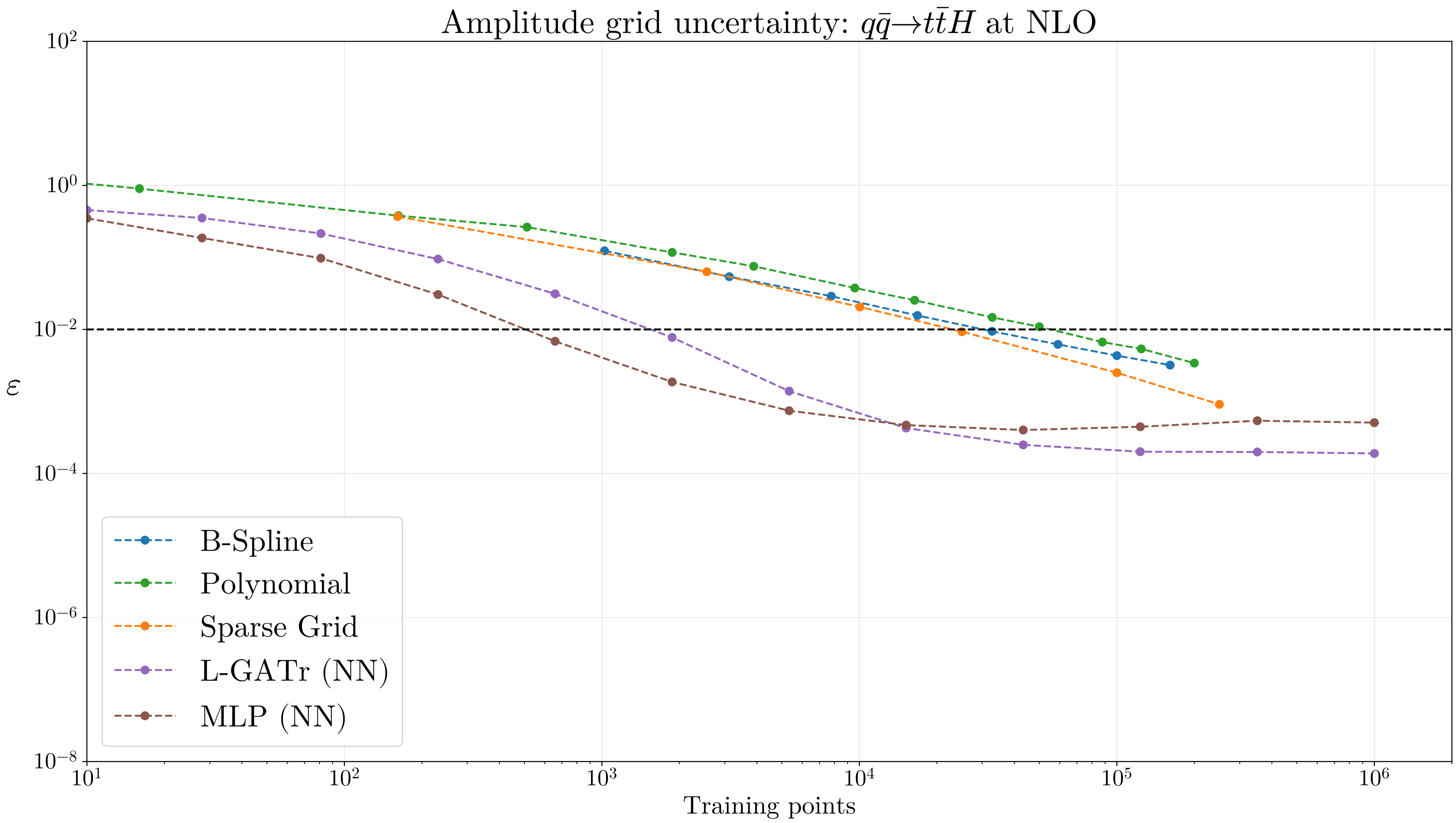


Spatially adaptive sparse grid

Error Estimation

- Validation should consider the desired use-case, i.e. *observables*
  - Some phase space points are more important
  - Ignore large errors in physically insignificant regions
- Take physically relevant samples: $x \sim w$ and $\varepsilon = \frac{1}{N} \sum_i \varepsilon_i$
- Also possible: sample uniformly and suppress errors: $x \sim 1$ and $\varepsilon = \frac{1}{N} \sum_i w_i \, \varepsilon_i$
- Incorporate into interpolation procedure
  - Target function $f = w \cdot |\mathcal{A}|^2$
  - Any weight is allowed, try to flatten f
  - Normalized error metric $\varepsilon = \frac{||f - \tilde{f}||_1}{||f||_1} = \frac{\sum_i |a(x_i) - \tilde{a}(x_i)|}{\sum_i |a(x_i)|}$ with $x \sim w$

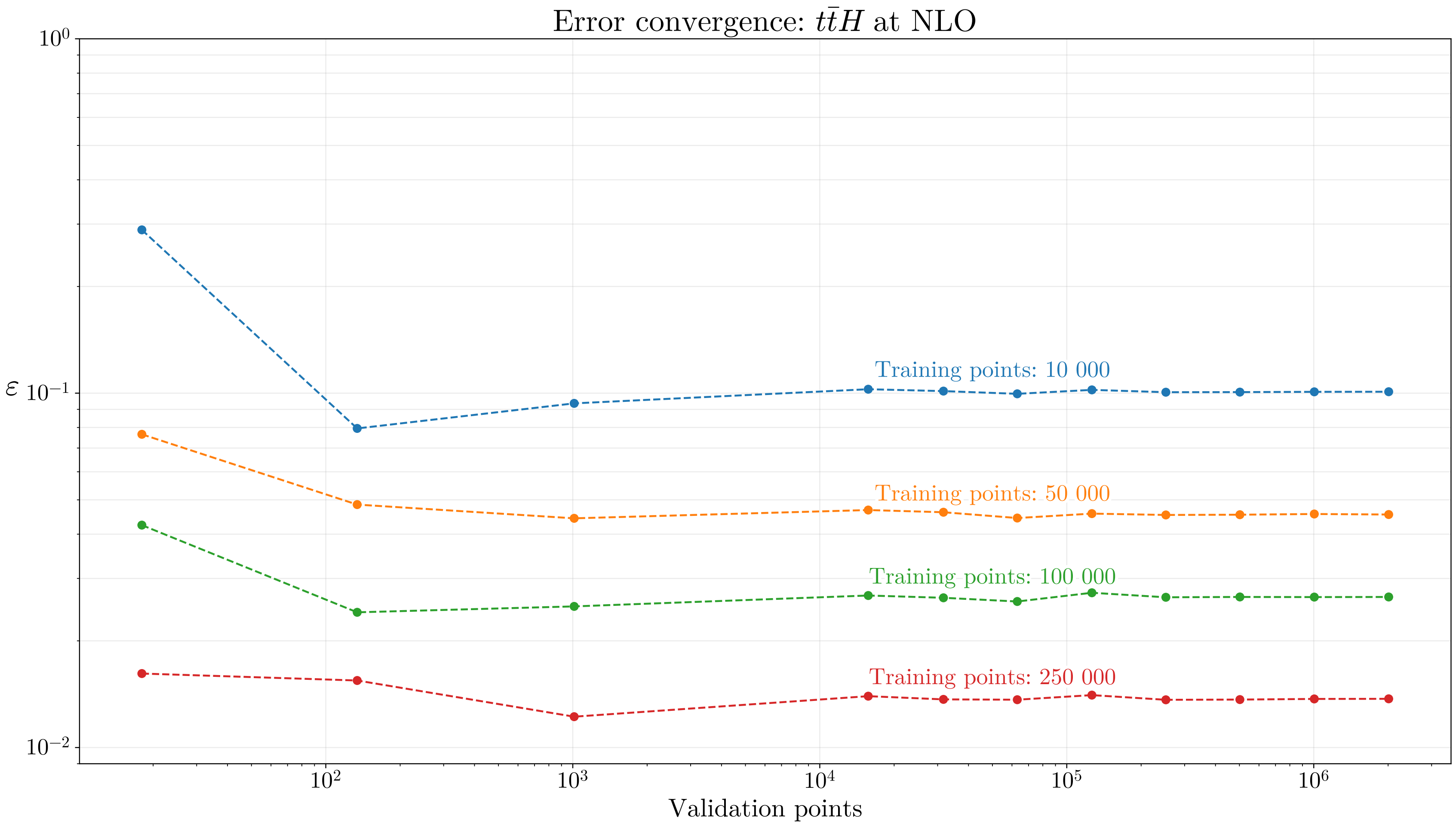
Comparison of Interpolation Methods



Summary

- We have a fully numerical setup to calculate complicated amplitudes ($t\bar{t}H$)
- Grids are required to include results into MC event generators
  - Especially for 2 → 3 processes with slow numerical evaluation times
  - This requires careful consideration of grid uncertainties
  - Grids can be optimized towards observables through physical arguments
- **Currently:** Finalize full $q\bar{q}$-channel and construct an amplitude grid
- **Future:** Include gg-channel

Backup: How many validation points are required?



Backup: Selecting an IBP basis

- An IBP basis is not unique, which master integrals should be selected?
- Four criteria: **1.** finiteness, **2.** D-factorising, **3.** fast to evaluate with pySecDec, **4.** simple denominators in IBP coefficients
- Finiteness and D-factorisation is achieved by dimension shifts and dotted propagators
  - d=4,6,8 for most integrals (d=2 for some easy ones with 4 propagators)
  - 2 dots in most sectors, in some lower sectors there are more dots (up to 6 for a three propagator integral)
- Fast evaluation and simple denominators is done through trial and error
  - Generate a basis that fulfills finiteness and D-factorisation
  - Perform reductions while neglecting sub-sectors for sets of master integrals, select the set with smallest denominator factors
  - Benchmark which master integrals in this set are fast to evaluate with pySecDec
  - Repeat the process while restricting the basis to include the fast to evaluate masters

Backup: Sector Decomposition

- Targets dimensionally regulated Feynman integrals

$$I \sim \int_0^\infty \prod_{j=1}^{N} \mathrm{d}x_j x_j^{\nu_j - 1} \delta(1 - \sum_{i=1}^{N} x_i) \frac{\mathcal{U}(\bar{x})^{N_\nu - (L+1)D/2}}{\mathcal{F}(\bar{x}, p^2, m^2)^{N_\nu - LD/2}}$$

L loops

$$D = 4 - 2\varepsilon$$

$$N_\nu = \sum_{j=1}^{N} \nu_j$$

- Transforms integral into sums of expansions in the regulator

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

Parameter integrals

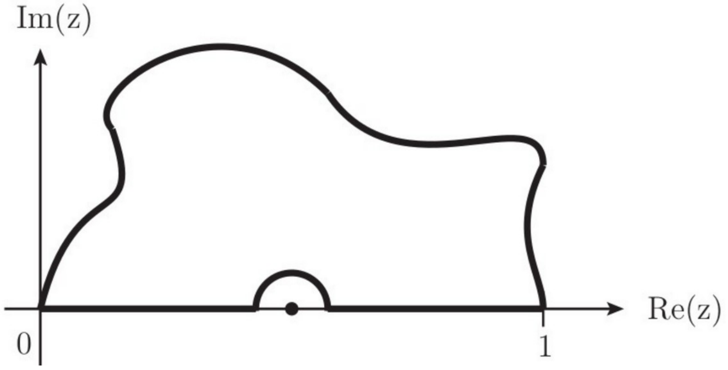
- Singularities are extracted as poles in regulator with simple subtraction terms

$$\int_0^1 \mathrm{d}x\, x^{-1-\varepsilon} \mathcal{I}(x, \varepsilon) = \frac{-1}{\varepsilon} \mathcal{I}(0, \varepsilon) + \underbrace{\int_0^1 \mathrm{d}x\, x^{-1-\varepsilon} [\mathcal{I}(x, \varepsilon) - \mathcal{I}(0, \varepsilon)]}_{\text{Finite piece, integrate numerically!}} = \frac{C_1}{\varepsilon} + C_0$$

Backup: Contour Deformation

- Sector decomposition extracts endpoint singularities only
  - Bulk singularities are avoided with contour deformation
- Contour deformation works in most cases, but is computationally very expensive
  - New ideas to avoid having to use contour deformation [2407.06973]

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



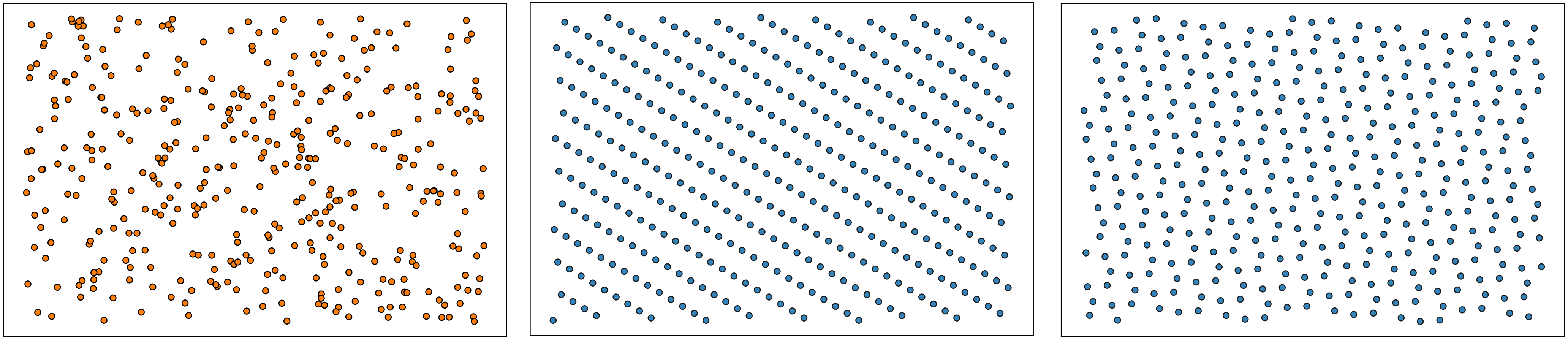
$$\mathcal{F}(\vec{z}(\vec{x})) = \mathcal{F}(\vec{x}) - \underbrace{i\,\lambda \sum_j x_j(1-x_j) \left(\frac{\partial \mathcal{F}}{\partial x_j}\right)^2}_{\text{Complicates integrand massively}} + \mathcal{O}(\lambda^2)$$

Backup: QMC Integration I

- Observation: $\varepsilon \leq \mathrm{Discrepancy}\{x_i\} \cdot V[\mathcal{I}]$, take $\{x_i\}$ from a low discrepancy sequence (R1SL-rule)
- Estimate of integral is achieved through random shifts
- Error convergence: $\varepsilon_{MC} \sim \mathcal{O}\left(\frac{1}{\sqrt{n}}\right)$ and $\varepsilon_{RQMC} \sim \mathcal{O}\left(\frac{1}{n}\right)$

$$I[\mathcal{I}] \approx \bar{Q}_{n,m}[\mathcal{I}] \equiv \frac{1}{m}\sum_{k=0}^{m-1} Q_n^{(k)}[\mathcal{I}], \quad Q_n^{(k)}[\mathcal{I}] \equiv \frac{1}{n}\sum_{i=0}^{n-1} \mathcal{I}\left(\left(\frac{i\mathbf{z}}{n} + \Delta_k\right) \mathbf{mod}\, 1\right)$$

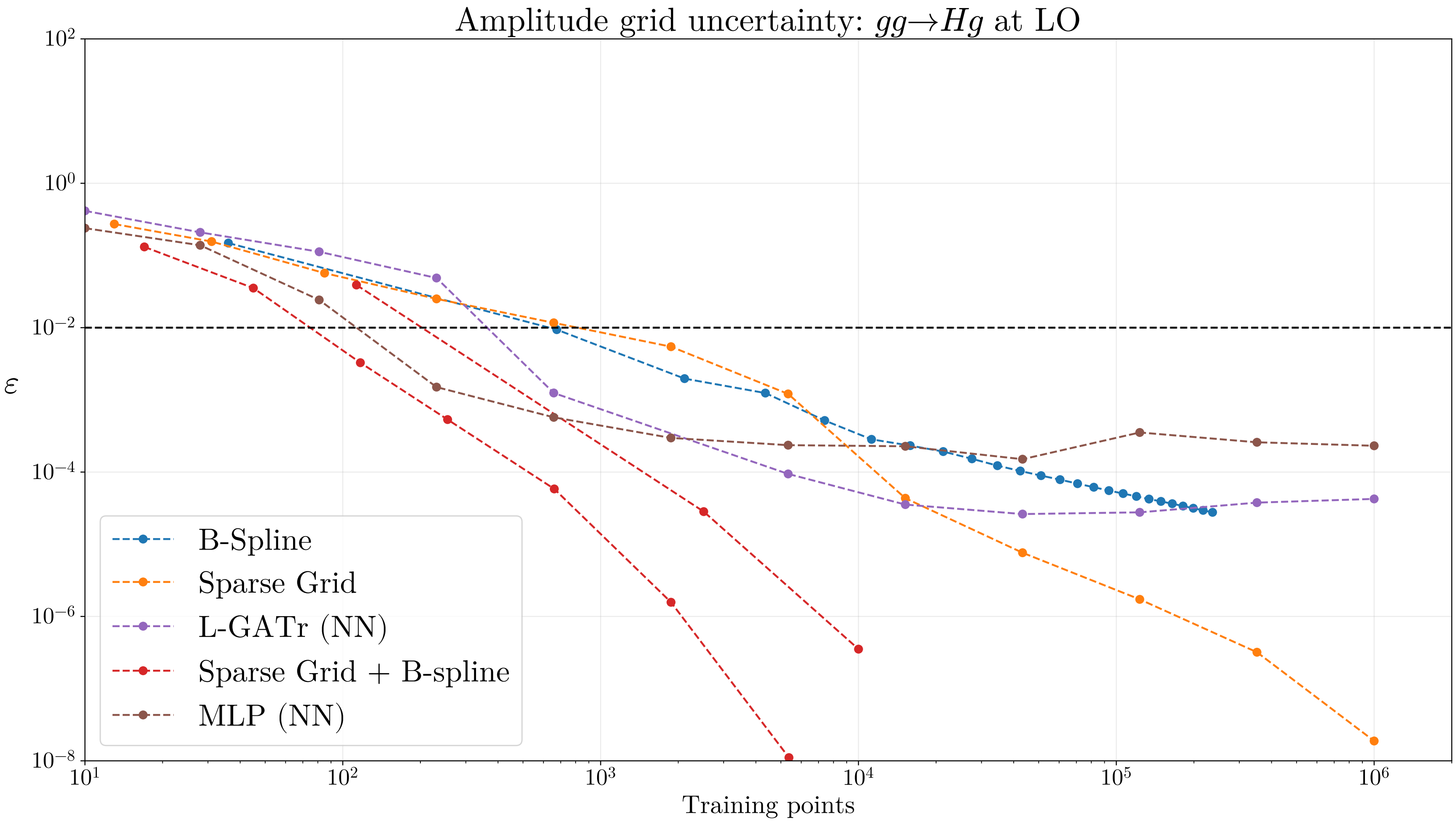
Random samples (MC) | Quasi-random samples (QMC) | Random shifts (RQMC)



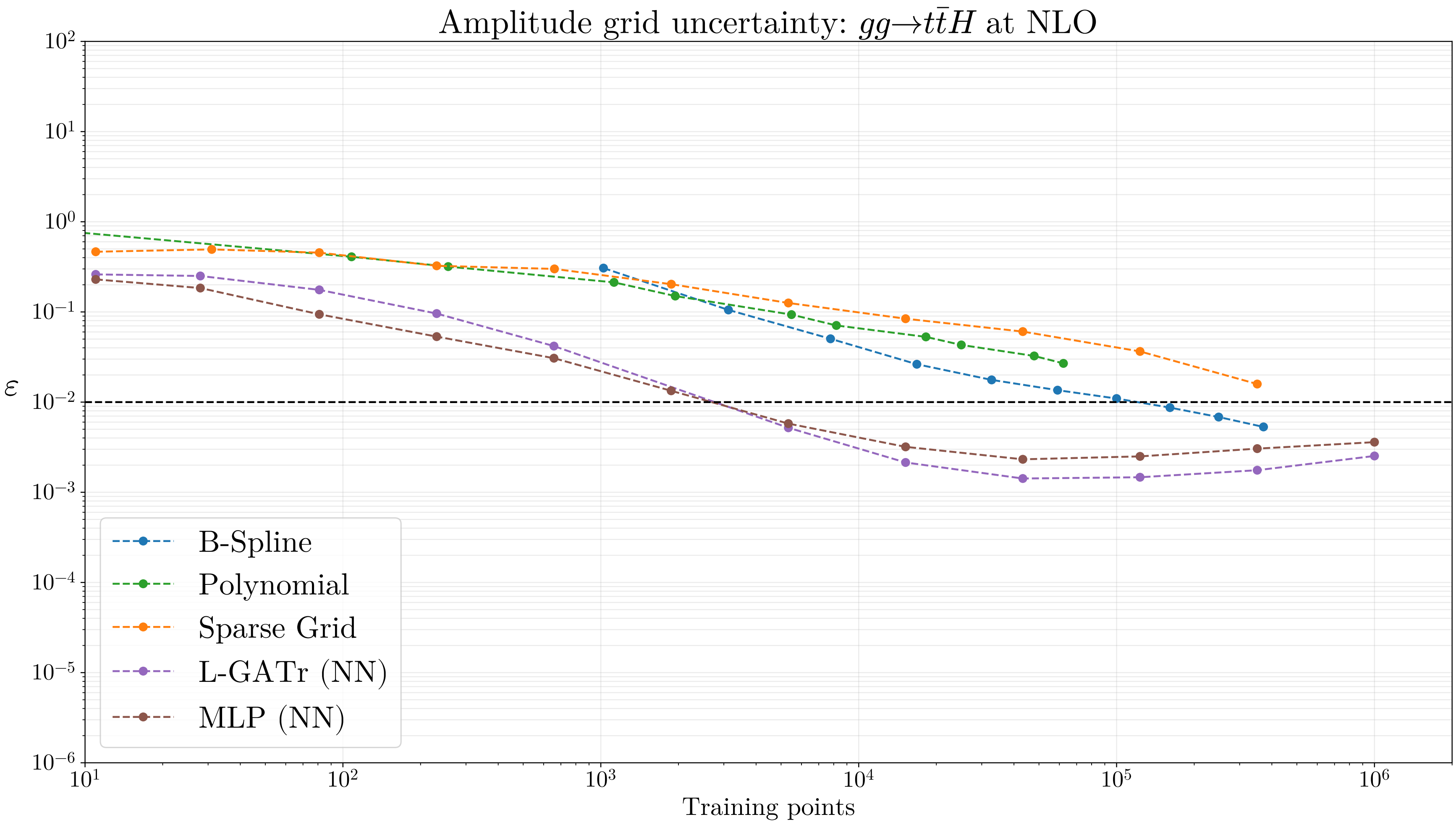
Backup: QMC Integration II

- Classical QMC error bound: $\varepsilon \leq \mathrm{Discrepancy}\{x_i\} \cdot V[\mathcal{I}]$ (Star disc. ; Hardy and Krause variance)
- Smooth integrands have: $\mathcal{O}((\log n)^d/n)$ (dimension dependent?!)
- In certain *weighted function spaces* convergence becomes independent of dimension
- Example: Korobov space of periodic smooth functions
  - Our integrands are usually smooth but not periodic: Apply Korobov transformation
  - Differentiable integrands after Korobov transformation have error scaling: $\mathcal{O}\left(\frac{1}{n}\right)$
  - Convergence is independent of dimension

Backup: 2-Dimensional Case (H+j)



Backup: Interpolation in gg-channel



Backup: Interpolation at LO vs NLO