LoopScalla: A new multiloop setup

Vladyslav Shtabovenko
University of Siegen
Center for Particle Physics Siegen (CPPS)

Young Scientists Meeting of the CRC TRR 257

Heidelberg

22nd of July 2025

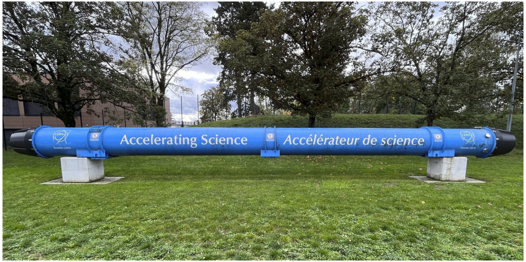
Outline

1. Motivation: Why more loops?
2. LoopScalla for multiloop calculations
3. Summary and Outlook

Motivation: Why more loops?

- Feynman's diagrammatic approach: cornerstone of modern perturbative calculations
- High luminosity LHC [Apollinari et al., 2015] + future colliders: Precise theory predictions crucial
- Higher precision on the theory side $\to$ more loops (not always, but often)
- Multiloop automation is needed but remains challenging [Campbell et al., 2024]

- Different ways to go beyond one loop:
  - New codes developed for multiloop **CARAVEL** [Abreu et al., 2021], **PYSECDEC** [Heinrich et al., 2024]
  - 1-loop code upgraded to support more loops **GOSAM** [Borowka, Heinrich, et al., 2016; Borowka, Greiner, et al., 2016], **OPENLOOPS** [Pozzorini et al., 2022; Zoller et al., 2022], **HELAC** [Canko et al., 2024], **FEYNCALC** [Mertig 1990, VS et al. 2016,2020,2021,2023]
- Practice: self-written codes tailored for the specific process, focus on $i\mathcal{M}$



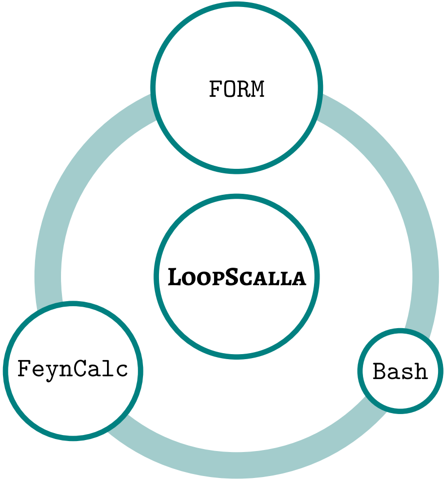
- Main steps of calculating $i\mathcal{M}$
    1. Generation of Feynman diagrams for the given process
    2. Algebraic simplification of the amplitudes (+ expansions in small parameters)
    3. IBP reduction
    4. Evaluation of the master integrals - TOUGH
    5. Assembly of the final amplitude
- Apart from (4), the automation of all other steps is well understood
- Arising problems are mostly of technical rather than conceptual nature
- But: too few codes implementing these steps in one framework are public!

- The main ingredient of most calculations - **FORM** [Vermaseren, 2000; Kuipers et al., 2013]
- **FORM** alone is not enough to cover all necessary steps!
- Need at least diagram generator, topology minimizer, IBP reducer, glue scripts, ...
- Groups specializing on multiloop calculations use private computational setups
- Some pieces are public (**TAPIR** [Gerlach et al., 2023]) or available upon request (**Q2E/EXP** [Seidensticker, 1999; Harlander et al., 1998]), but most are known only by names.
- There are also few public codes (**Alibrary** [https://magv.github.io/alibrary/], **FeAmGen.jl** [Wu & Li, 2024], **HepLib** [Feng et al., 2021, 2023]) but they are not optimal for every usage case

LoopScalla for multiloop calculations

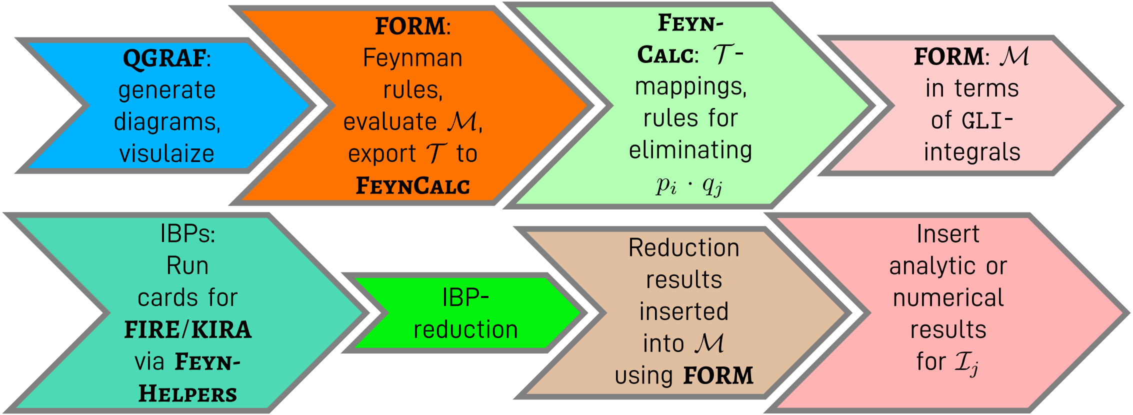
LOOPSCALLA: an attempt to create a public multiloop framework that is

- well-designed (borrowing good ideas from other codes, especially **CALC** from KIT)
- easy-to-use (for people that know **FORM**)
- extensible to cover a wide range of possible calculations
- built-in interfaces to essential tools (diagrams, IBPs, numerics)
- native support for calculations on clusters (**SLURM**)



- **LoopScalla** covers main multiloop amplitude evaluation steps
- Performance-critical operations (heavy algebra) implemented directly in **FORM**
- **Mathematica** for steps that are too cumbersome using **FORM** alone
- Parallelization to work around **Mathematica** bottlenecks
- Can add new models (**QGRAF** [Nogueira, 1993]) and Feynman rules by hand
- Diagram visualization using **GraphViz** or **TikZ-Feynman** [Ellis, 2017]
- Tensor reduction built-in
- Topology minimization including basis completion and partial fractioning (**FeynCalc**)

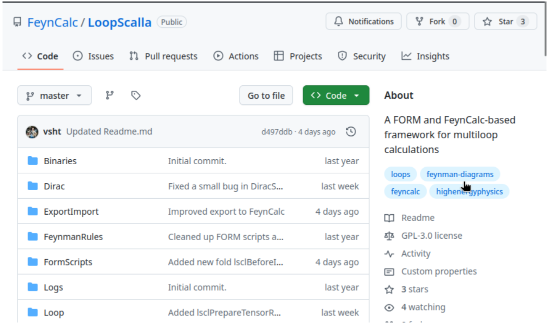
Example of using a **LoopScalla**-based setup (locally or on a cluster)



LoopScalla is used through the terminal

- Extensive collection of shell scripts to control the progress of the calculation
- Every step started by a dedicated script
- Input: range of dias/topos or single items
- Every step is parallelizable with **GNU parallel**
- Every step can be run on a **SLURM** cluster

- Not published yet, but publicly available
- https://github.com/FeynCalc/LoopScalla
- Free and Open Source
- Documentation is still WIP
- Only one example so far (QCD $g \to g$ @ 2L)
- Bench-tested in calculations of the purely hard-collinear coefficient $F_{hc}(\gamma)$ for $B_c \to \eta_c$ form factors in NR approximation ($m_b \gg m_c \gg \Lambda_{\text{QCD}}$) at 2- and 3-loops [Boer, Bell, Feldmann, Horstmann, VS 2023,2024]



Summary and Outlook

Summary

- **LOOPSCALLA**: get into loop calculations without spending months on writing your own setup
- Ready to run on a **SLURM** cluster
- The knowledge of **FORM** is still necessary
- Reliance on **FEYNCALC** and **MATHEMATICA** might not be everyone's cup of tea

Outlook

- More examples and proper documentation
- More expansion options beyond naive Taylor would be useful