FEYNMAN INTEGRAL EVALUATION FOR 2-LOOP EW CORRECTIONS WITH SEASYDE

Based on: arXiv:2205.03345, arXiv:2502.14742

Tommaso Armadillo - UCLouvain & UNIMI

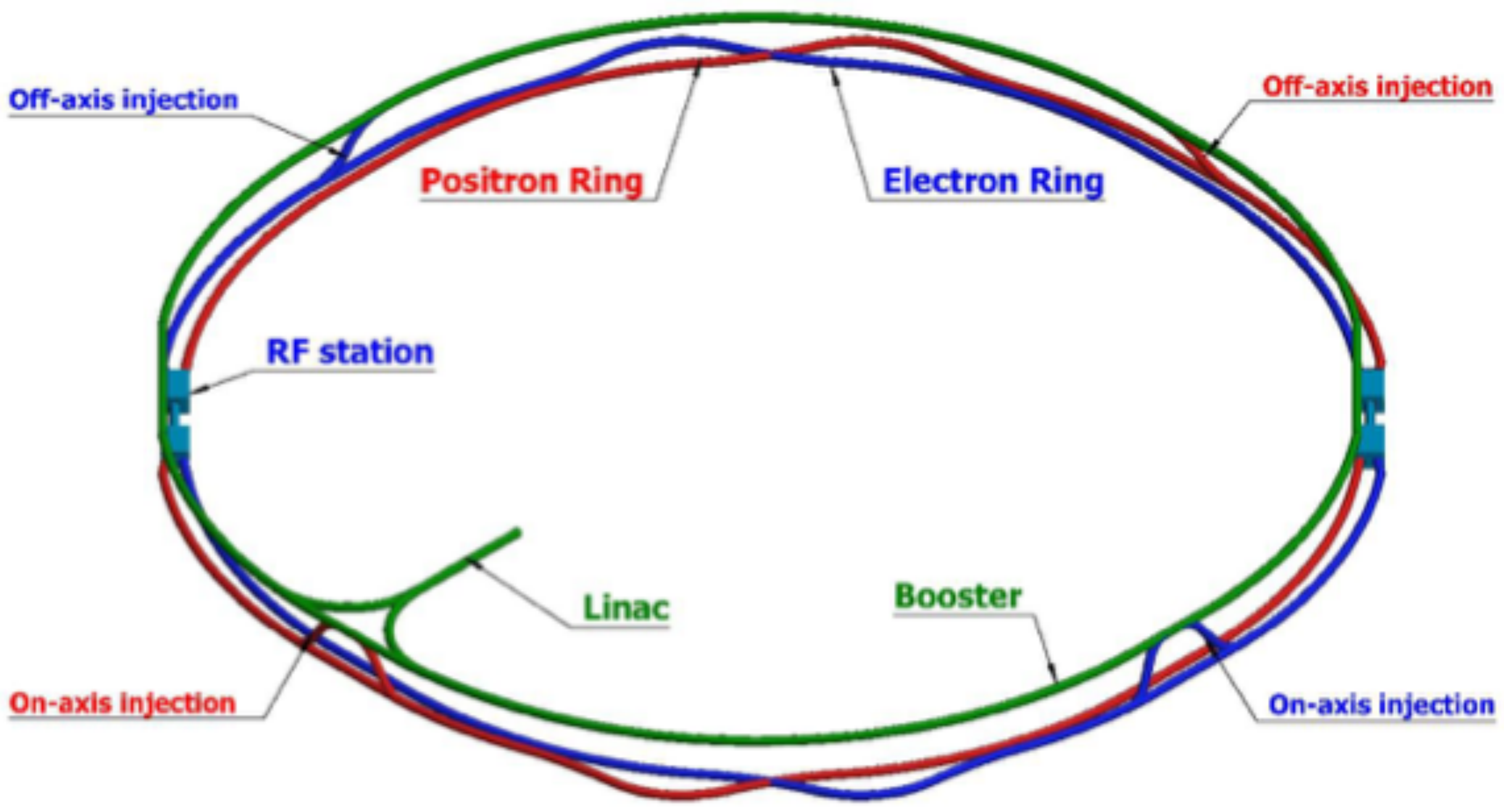
UCLouvain

CEPC Workshop - Barcelona - 16th June 2025

Motivations

- **Electron-positron colliders** are one of the main promising alternatives for the post-LHC era
- Based on the latest projections from ESPPU, CEPC will be able to provide measurements with **statistical errors of $\mathcal{O}(0.1\%)$ or better**

Energy (GeV)	Integrated luminosity (ab^{-1})	% error
91	100	2 x 10^{-4}
160	6.9	0.02
240	21.6	0.23
360	1	0.14

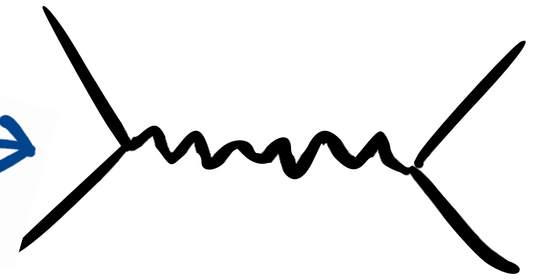


- Obtaining and interpreting such measurements will be an **incredible task** both from experimental and theoretical side.

Higher order corrections

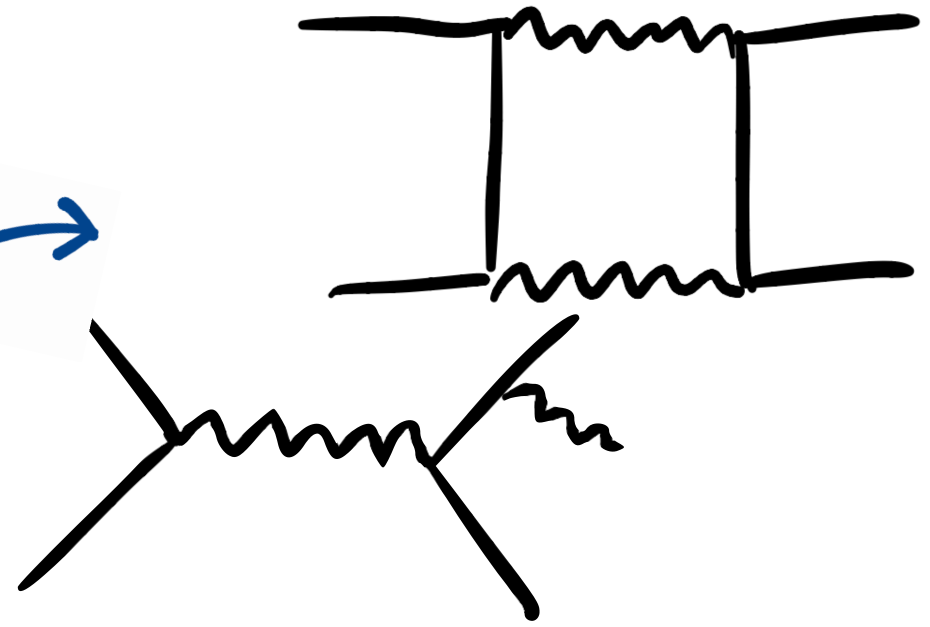
- Obtaining a theoretical prediction at this level of precision is a formidable task, which requires to have all theory systematics under control (e.g. input scheme, electron PDFs, higher order corrections, …)

LO: $\sigma^{(0,0)}$



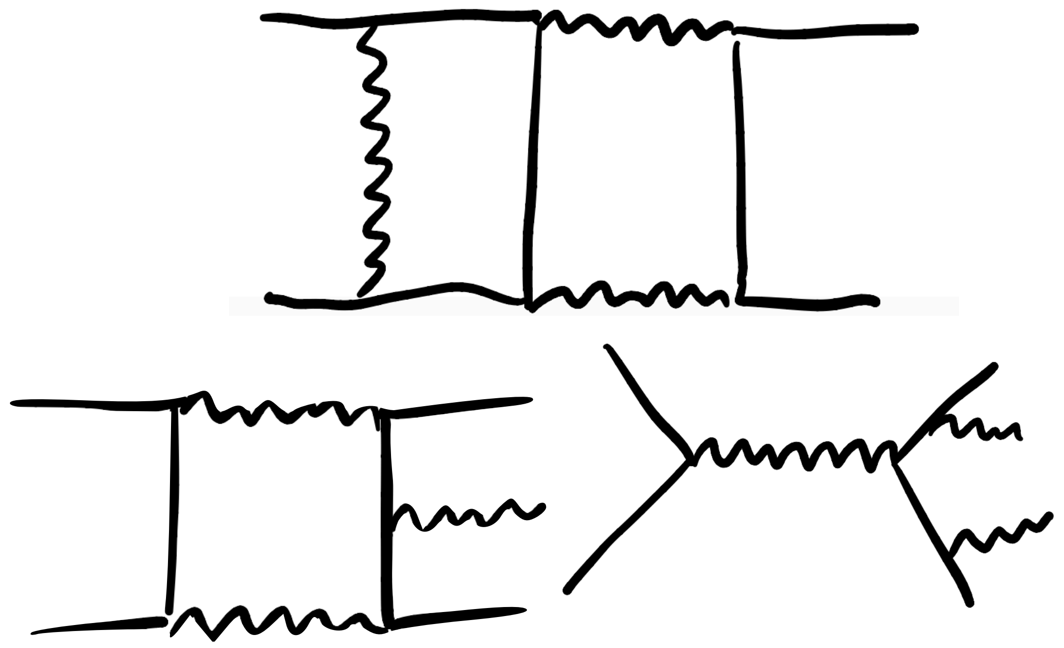
AUTOMATED

NLO: $+\alpha\,\sigma^{(1)}$



CHALLENGING ☹️

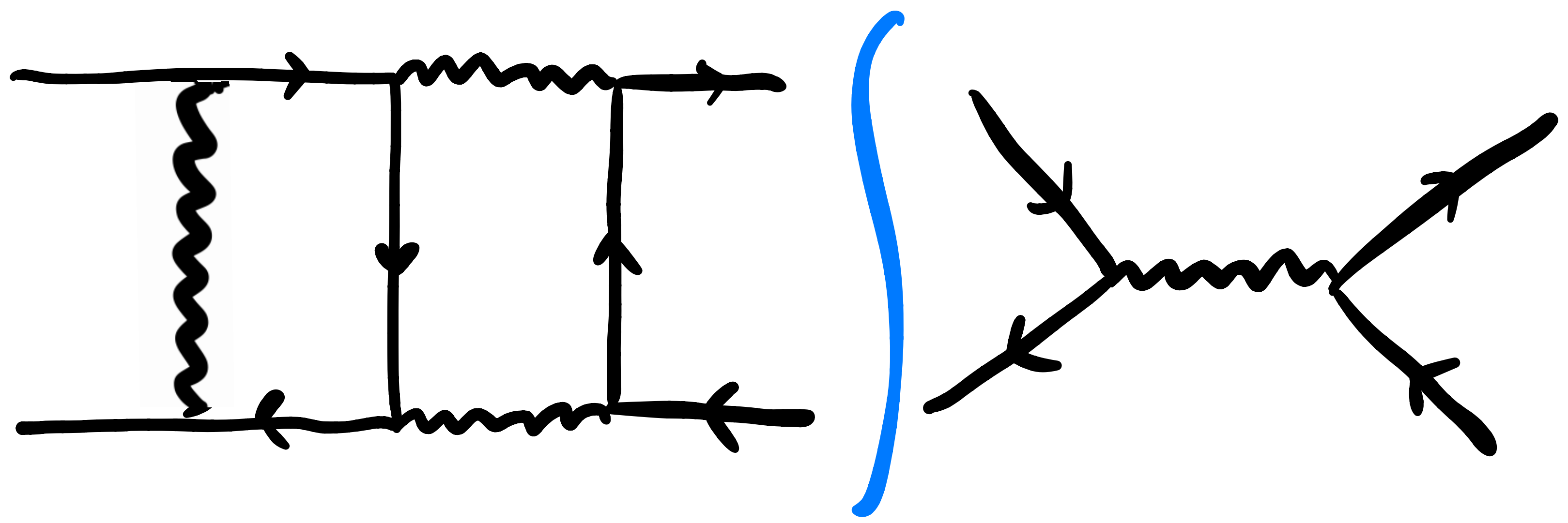
NNLO: $+\alpha^2\,\sigma^{(2)} + \ldots$



- The **state-of-the-art is NLO EW** corrections, no NNLO EW calculations for $2 \to 2$ scattering is available in literature (only some Sudakov approximations).

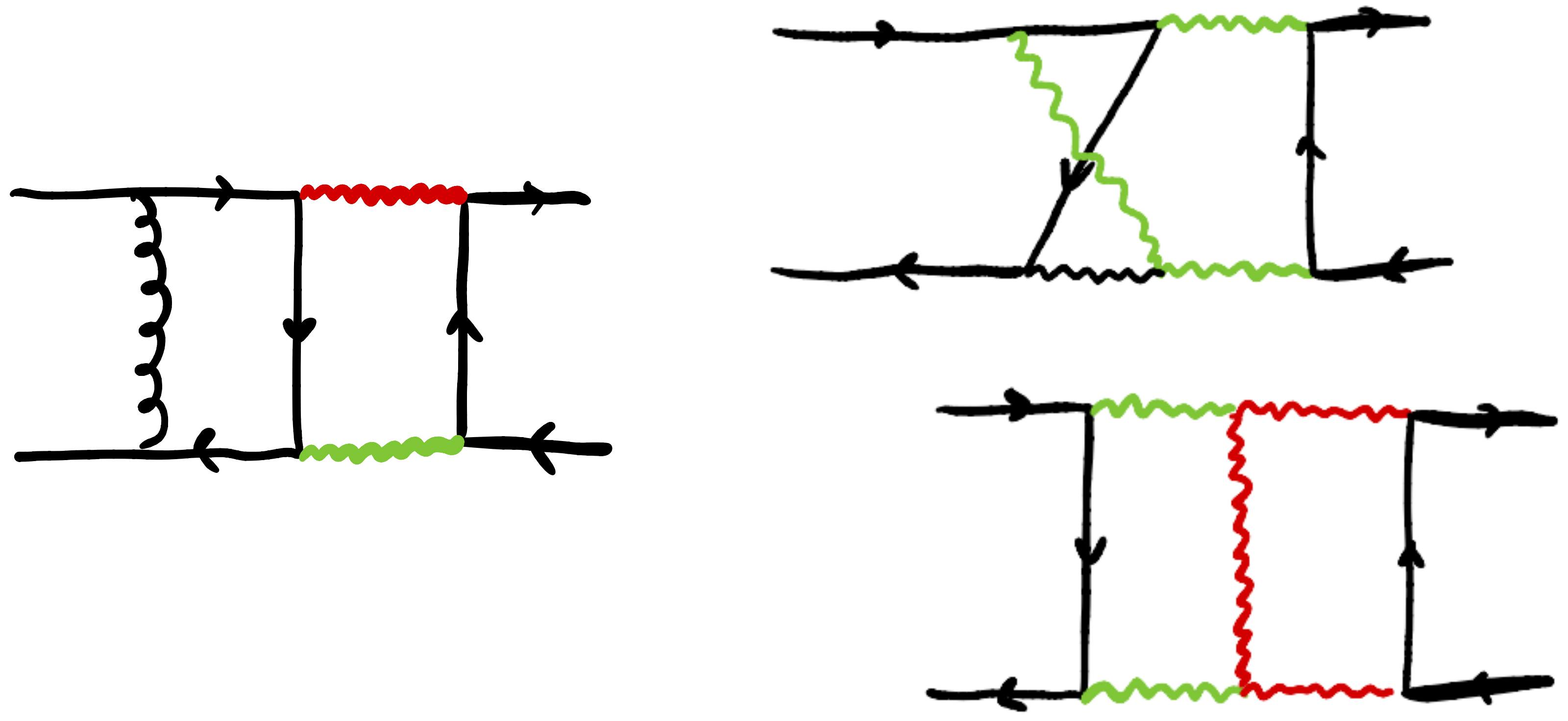
NNLO EW corrections

Pure Virtual



- The **pure virtual contributions are usually the main bottleneck**;
- Can we use re-use the same technology we developed for **NNLO QCD** or **NNLO QCD-EW** corrections? Yes, especially result for Drell-Yan, however there some additional complications:
  - Gamma-5
  - Renormalisation
  - Feynman integrals

This talk!



Evaluating Feynman integrals

- What we would like to compute are objects like this:

$$d = 4 - 2\epsilon$$

$$I(\alpha_i; s_j, d) = \int \prod_{k=1}^{l} \frac{d^d q_k}{i\pi^{d/2}} \frac{1}{\mathcal{D}_1^{\alpha_1} \dots \mathcal{D}_n^{\alpha_n}}$$

kinematic variables

$e.g.\ \ (p_1 - q_1)^2 - m^2 + i\delta$

- Using **Integration by Parts** (IBP) identities, we can express all the integrals in our problem in terms a smaller subset, the so-called **Master Integrals**.

$\mathcal{O}(10^{5-7})$ Integrals $\xrightarrow{\text{IBPs}}$ $\mathcal{O}(10^{2-3})$ Master Integrals

How to compute the Master Integrals?

- **Many techniques** have been developed during the years, each with pros and cons. Here I will focus on the **method of differential equations**.

- The idea is that by differentiating a master w.r.t. a kinematical invariants we obtain a **first order linear differential equations**, whose solution is the master integrals we are interested in.

$$\frac{\partial}{\partial s_k} I(\alpha_i; s_j, d) = \sum \textit{scalar integrals} = \sum \textit{master integrals}$$

$$\textit{IBPs}$$

- By repeating the same process for every master integral we obtain a **system of first order linear and homogeneous differential equations**.

What are we looking for?

- So we just have to solve a system of first order differential equations… HOW?
- Ideally, we would like:
  - A method **easy to automate**

```
ImportMasterIntegrals["my_master_integrals"]
ABISS: Succesfully imported 380 master integrals
```

  - A solution **compact and easy to handle** to allow for simplifications

$$\mathrm{Li}_2(x) + \mathrm{Li}_2\left(\frac{1}{x}\right) = -\frac{\pi^2}{6} - \frac{\log^2(-x)}{2}$$

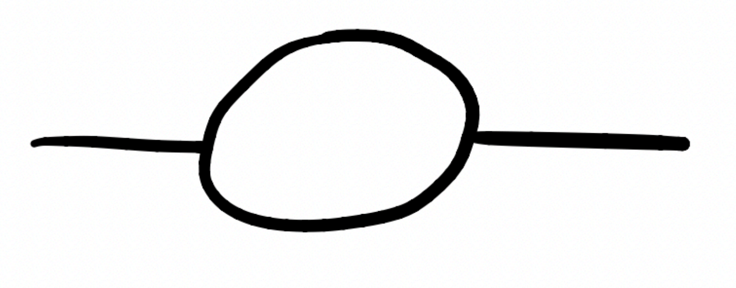
$$\mathrm{Li}_2(x) + \mathrm{Li}_2(1-x) = \frac{\pi^2}{6} + \log(x)\log(1-x)$$

  - A solution **fast to evaluate** to be implemented in a Monte-Carlo
  - To have **high control on numerical precision**

$$\mathcal{O}(10-10^2) \qquad 2\mathrm{Re}\mathcal{M}^{(2)}\mathcal{M}^{(0)*} = \sum_i c_i\, MI_i \qquad \mathcal{O}(10^{-10}-10^{10})$$

Semi-analytical solution

- A possibility could be to use a **semi-analytical approach**. The result is provided as a power series which can be easily evaluated in every point of the domain.



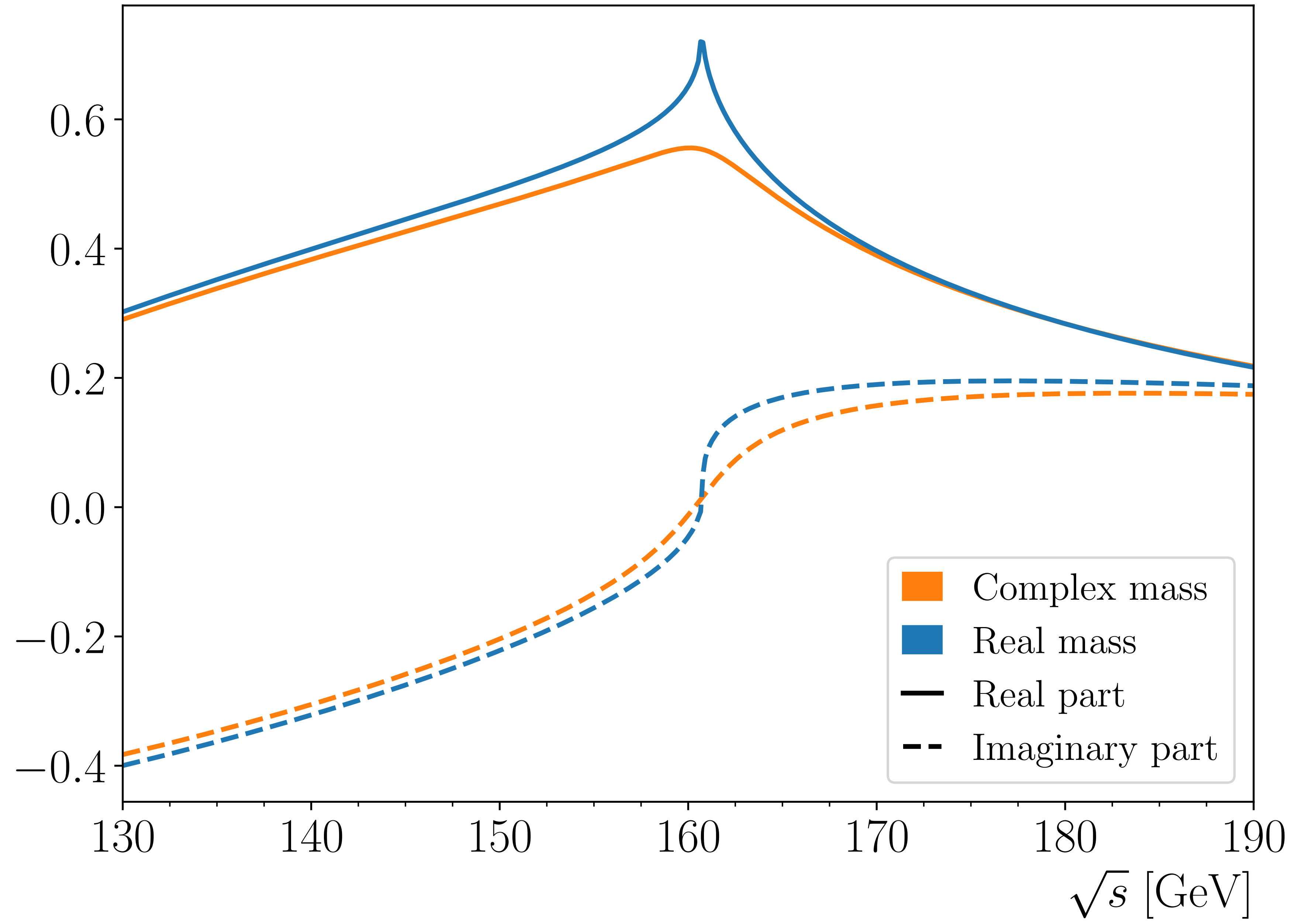
$$= -\gamma_E + \frac{1}{6}p^2 + \frac{1}{60}(p^2)^2 + \frac{1}{420}(p^2)^3 + \frac{1}{2520}(p^2)^4 + \frac{1}{13860}(p^2)^5 + \dots$$

- The method has been firstly implemented in the Mathematica package **DiffExp** for a **real kinematic variable** *[F.Moriello, arXiv:1907.13234], [M.Hidding, arXiv:2006.05510]*
- The main advantage is that all the calculations can be carried out analytically.
- This method is quite **easy to automate**. Provided that we have infinite time and space, we could achieve **arbitrary precision**. Moreover, once we have the solution, it can be evaluated numerically in a negligible amount of time.
- However, DiffExp cannot handle **complex kinematic variables.**

Complex Mass Scheme

- In EW calculations, we have to deal with intermediate unstable particles, such as W and Z. We have to use a gauge invariant definition of the mass, which is given by the **complex-mass scheme**;
- For these particles we consider their mass to be complex-valued:

$$\mu_V^2 = m_V^2 - i\Gamma_V m_V$$



- The complex mass scheme **regularises** the divergences, while preserving gauge invariance.

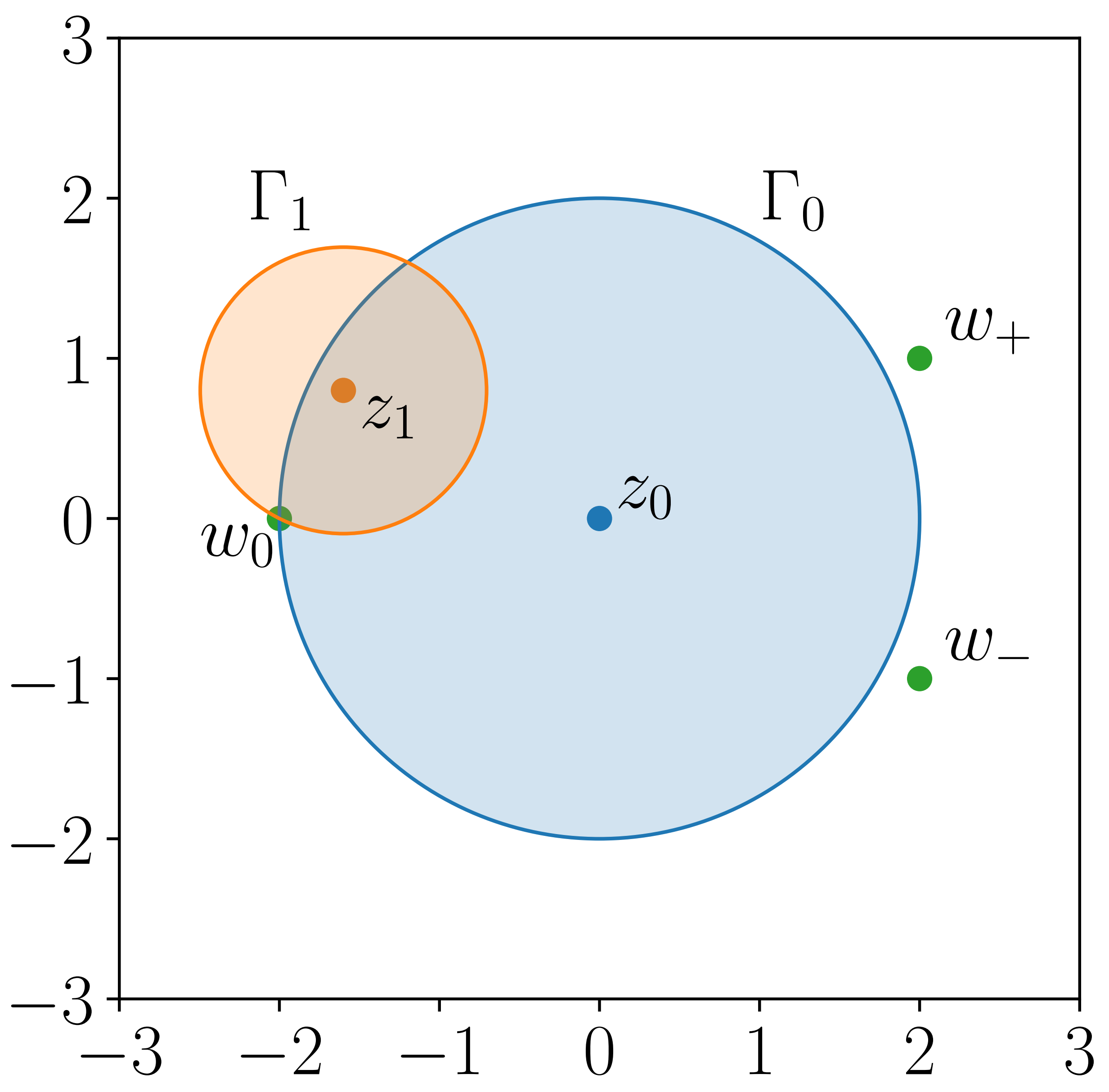
$$\frac{1}{s - m_V^2 + i\delta}$$

- However, it requires all the masses to be complex-valued, included the ones in the Feynman integrals. If we utilise **adimensional variables**, they become complex-valued as well:

$$x = \frac{s}{m_V^2} \to \frac{s}{\mu_V^2}$$

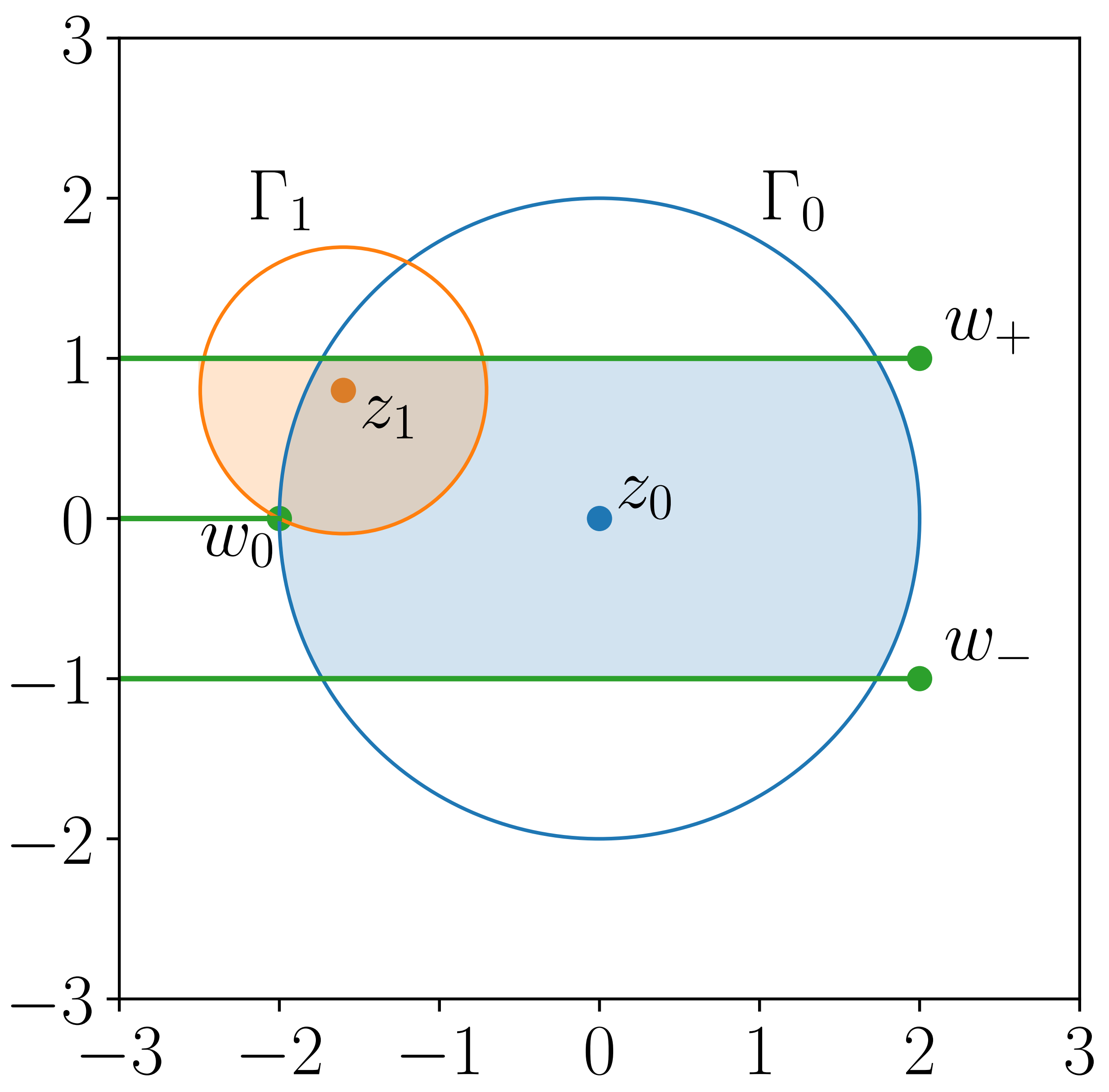
Analytic continuation

- As we saw, the analytic continuation must be discussed in the entire complex plane
- Power series have a limited **radius of convergence**.
- The radius is determined by the position of the **nearest singularity**.



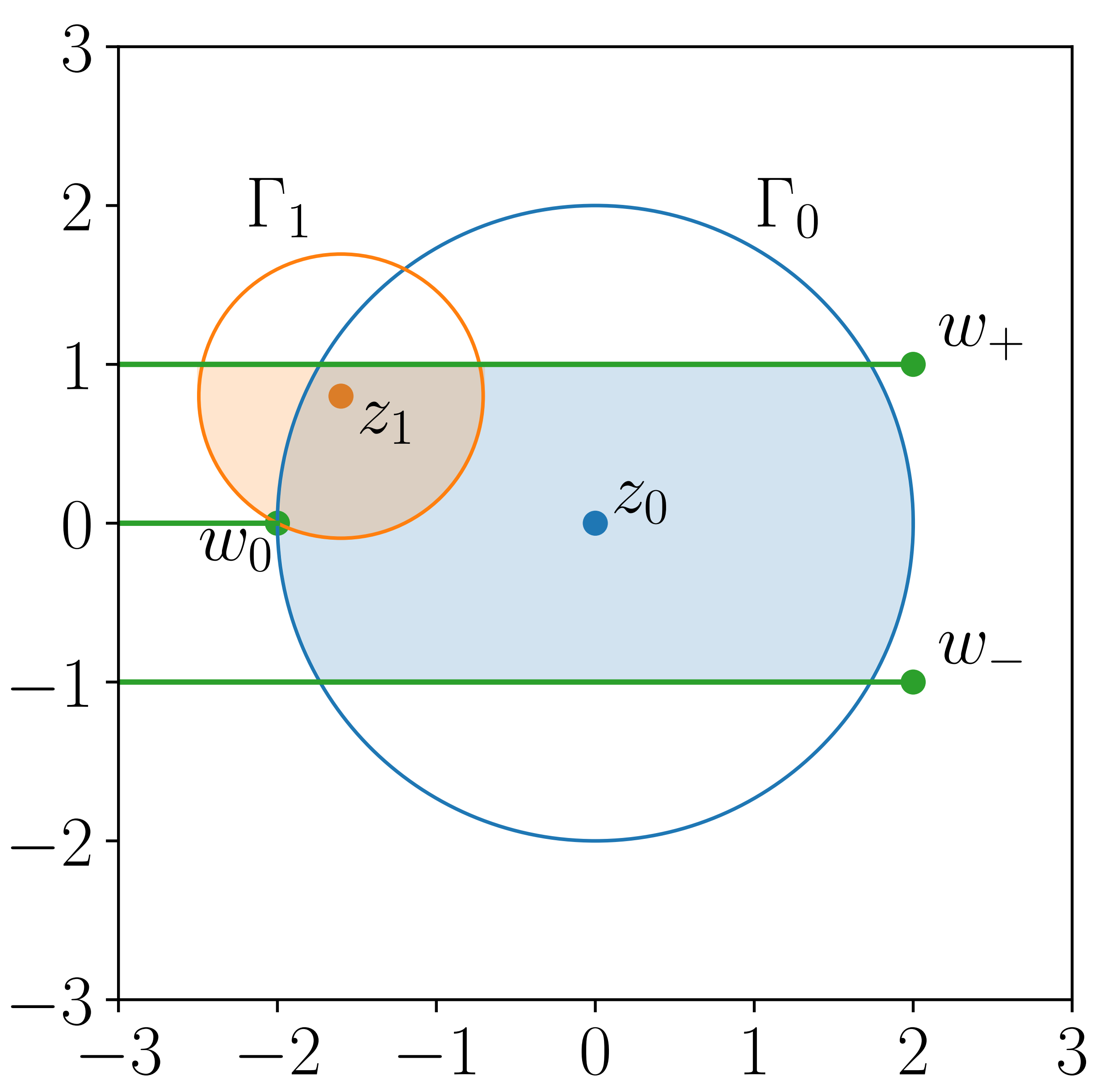
Analytic continuation

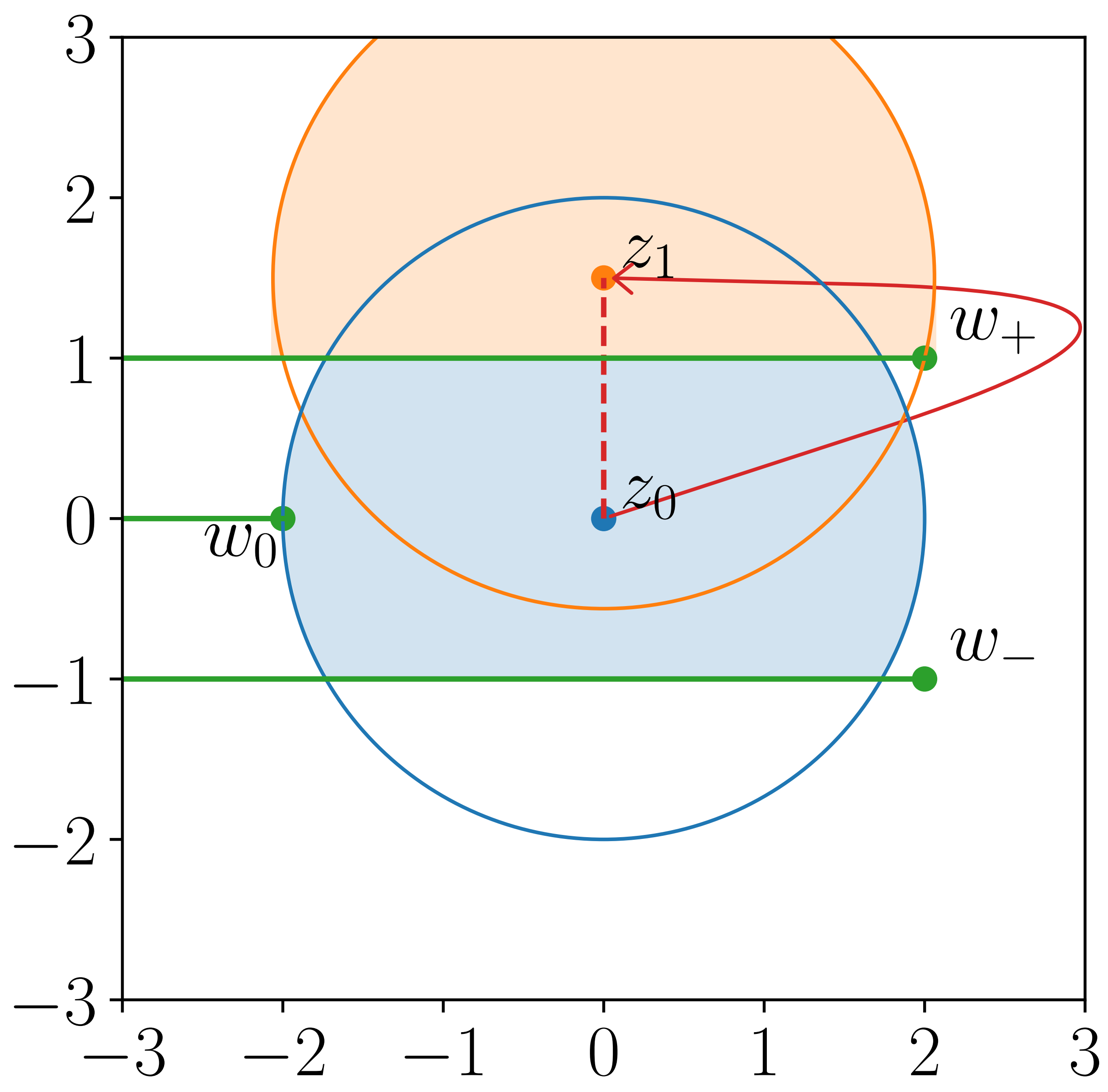
- As we saw, the analytic continuation must be discussed in the entire complex plane
- Power series have a limited **radius of convergence**.
- The radius is determined by the position of the **nearest singularity**.



Analytic continuation

- As we saw, the analytic continuation must be discussed in the entire complex plane
- Power series have a limited **radius of convergence**.
- The radius is determined by the position of the **nearest singularity**.





*For simplicity, we are not showing all the intermediate circles.

SeaSyde

[**TA**, R. Bonciani, S. Devoto, N.Rana, A.Vicini, arXiv:2205.03345]

https: //github.com/TommasoArmadillo/SeaSyde

SEASYDE

- **SeaSyde** (**S**eries **E**xpansion **A**pproach for **SY**stems of **D**ifferential **E**quations) is a general package for solving a system of differential equations using the series expansion approach;
- Seasyde can handle **complex kinematic variables** by introducing an original algorithm for the analytic continuation in the complex plane, thus being able to handle **complex internal masses**;
- **SeaSyde** can deal with arbitrary system of differential equations, covering also the case of **elliptic integrals**;
- Other public packages implementing the series expansion technique are **AMFlow**, **DiffExp**, **Line**.

Creating a numerical grid

Feynman Integrals

IBPs (Kira+FireFly+Ratracer)

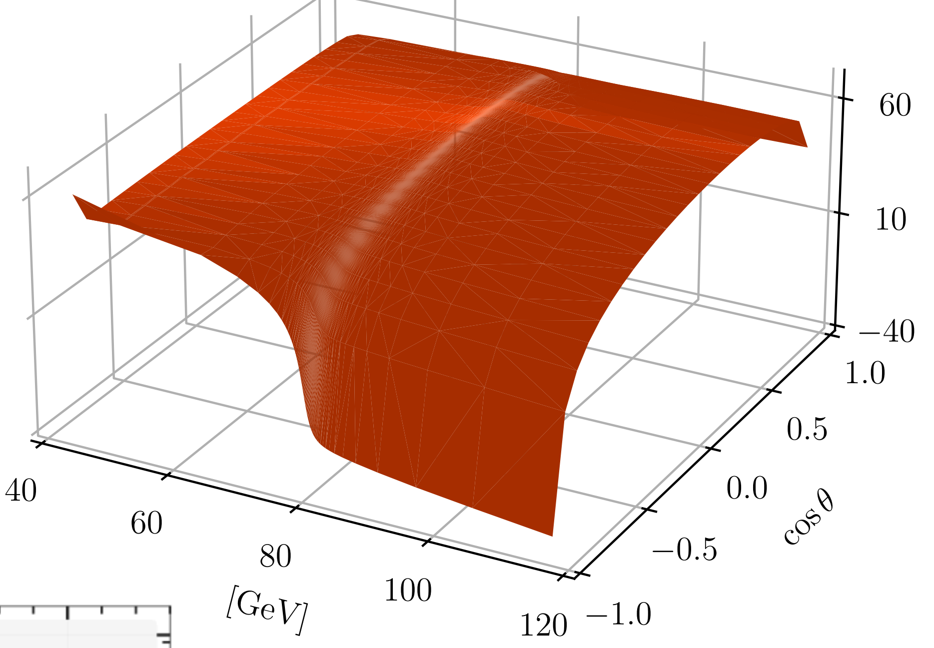
Master Integrals

Differential Equations (In-house Mathematica package based of LiteRed and Kira)

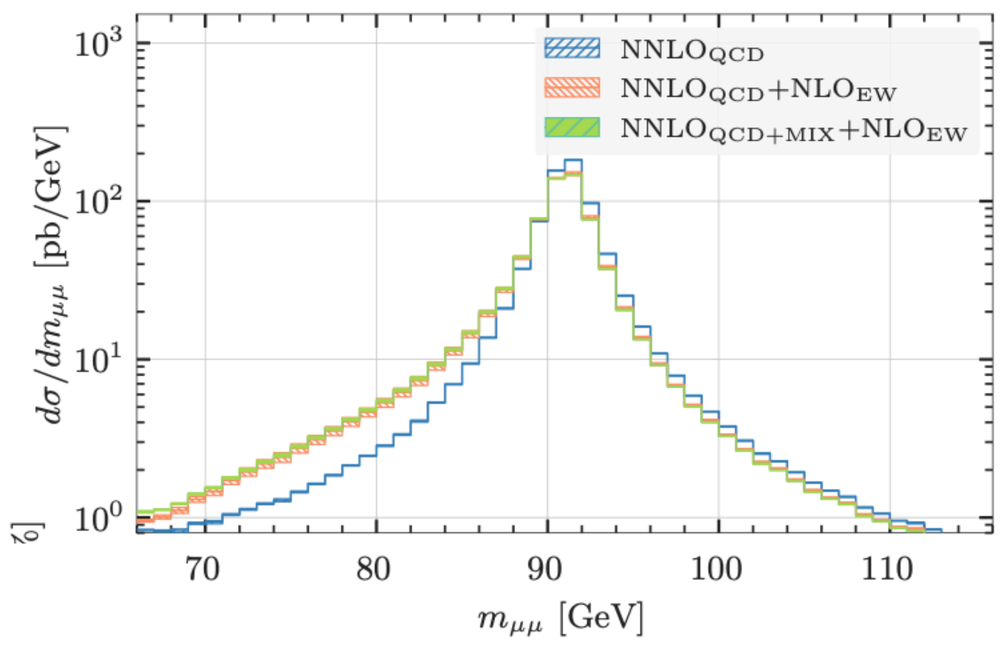
Boundary Conditions (AMFlow)

Solving the system (SeaSyde)

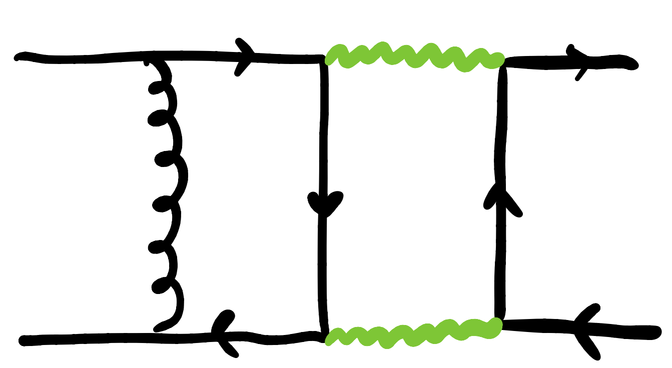
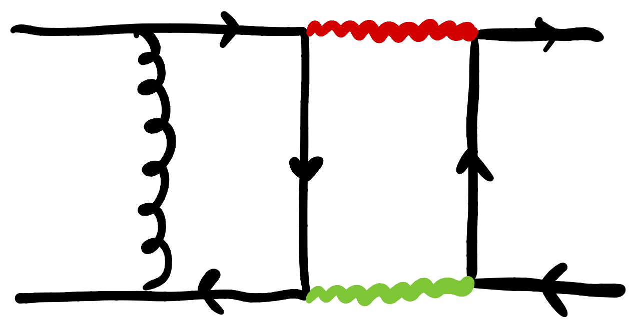
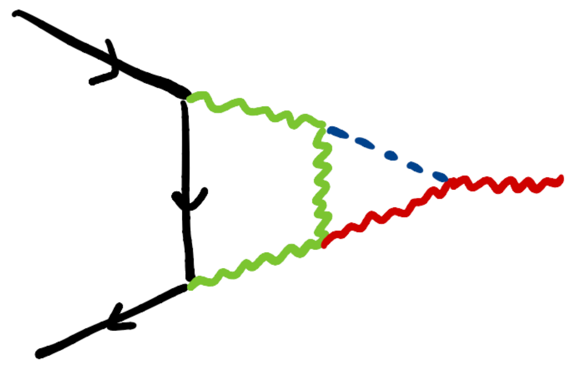
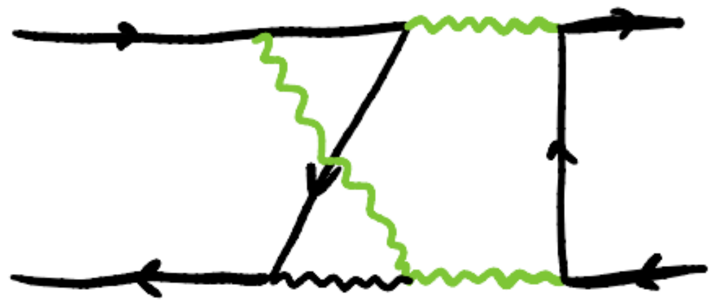
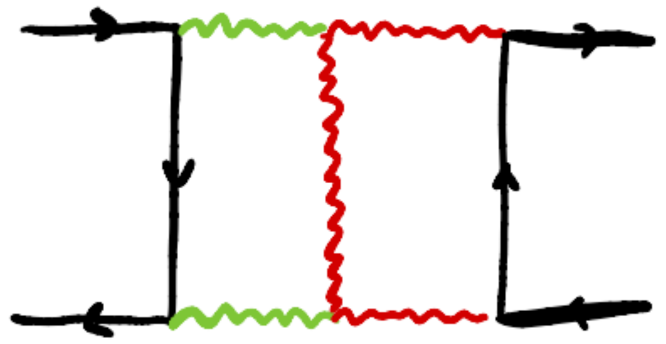
Numerical grid



- The numerical grid is then implemented in a **Monte-Carlo**;
- This approach is **completely general and easy to automate** and can be applied, in principle, to **any integral family**.



Examples

	NCDY - 2L Mixed	CCDY - 2L Mixed	Z on shell - 2L EW	NCDY - 2L EW	NCDY - 2L EW
Example Topology					
Number of masters	36	56	51	104	126
Reduction Kira 2.3 + Firefly	12 hours (32 core)	16 hours (32 core)	1 day (32 core)	30 m (120 core + Ratracer)	8 h (120 core + Ratracer)
AMFlow 1 point	50 min (32 core)	75 min (32 core)	6 h 45 m (32 core)	1 hour (120 core)	4 hours (120 core)
Dimension equations	700 Kb	2.1 Mb	/	45 Mb	350 Mb
SeaSyde 3250 points	5 days	10 days	/	??	??

Conclusion

- The method of differential equations, and in particular its semi-analytical approach, is a powerful technique for evaluating **Feynman integrals with multiple scales**;
- The main bottleneck is obtaining the **IBP relations** which are necessary to write down the differential equations;
- We implemented the method in the publicly available Mathematica package **SeaSyde**, which can handle arbitrary internal complex masses;
- The method has been already applied in the calculation of the mixed QCD-EW corrections to the Neutral and charged current Drell-Yan;
- The techniques that we developed for NNLO Mixed corrections can be generalised to **full 2-loop EW** calculations.

THANK YOU

SeaSyde

[**TA**, R. Bonciani, S. Devoto, N.Rana, A.Vicini, arXiv:2205.03345]

SEASYDE

A SIMPLE EXAMPLE

$$\begin{cases} f'(x) + \dfrac{1}{x^2-4x+5} f(x) = \dfrac{1}{x+2} \\ f(0) = 1 \end{cases}$$

$$f_{hom}(x) = x^r \sum_{k=0}^{\infty} c_k x^k$$

$$f_{hom}(x) = 5 - x - \frac{3}{10}x^2 + \frac{11}{150}x^3 + \ldots$$

$$\begin{cases} r c_0 = 0 \\ \frac{1}{5}c_0 + c_1(r+1) = 0 \\ \frac{4}{25}c_0 + \frac{1}{5}c_1 + c_2(2+r) = 0 \\ \frac{11}{125}c_0 + \frac{4}{25}c_1 + \frac{1}{5}c_2 + c_3(3+r) = 0 \\ \ldots \end{cases}$$

SeaSyde

[**TA**, R. Bonciani, S. Devoto, N.Rana, A.Vicini, arXiv:2205.03345]

SEASYDE

A SIMPLE EXAMPLE

$$\begin{cases} f'(x) + \dfrac{1}{x^2 - 4x + 5} f(x) = \dfrac{1}{x+2} \\ f(0) = 1 \end{cases}$$

$$f_{part}(x) = f_{hom}(x) \int_0^x dx' \frac{1}{(x'+2)} f_{hom}^{-1}(x') \\ = \frac{1}{2}x - \frac{7}{40}x^2 + \frac{2}{75}x^3 + \dots$$

$$f(x) = c\, f_{hom}(x) + f_{part}(x) \\ = 1 + \frac{3}{10}x - \frac{47}{200}x^2 + \frac{3}{250}x^3 + \dots$$

SeaSyde

[**TA**, R. Bonciani, S. Devoto, N.Rana, A.Vicini, arXiv:2205.03345]

SEASYDE

A SIMPLE EXAMPLE

$$\begin{cases} f'(x) + \dfrac{1}{x^2 - 4x + 5} f(x) = \dfrac{1}{x+2} \\ f(0) = 1 \end{cases}$$

$$\begin{aligned} f(x) &= c\, f_{hom}(x) + f_{part}(x) \\ &= 1 + \frac{3}{10}x - \frac{47}{200}x^2 + \frac{3}{250}x^3 + \ldots \end{aligned}$$

- This procedure can be generalised to **systems of differential equations**;
- The method has been firstly implemented in the Mathematica package **DiffExp** for a **real kinematic variable** [F.Moriello, arXiv:1907.13234], [M.Hidding, arXiv:2006.05510]
- The great advantage of this approach is that we can reach **arbitrary precision** just by adding more terms in the serie

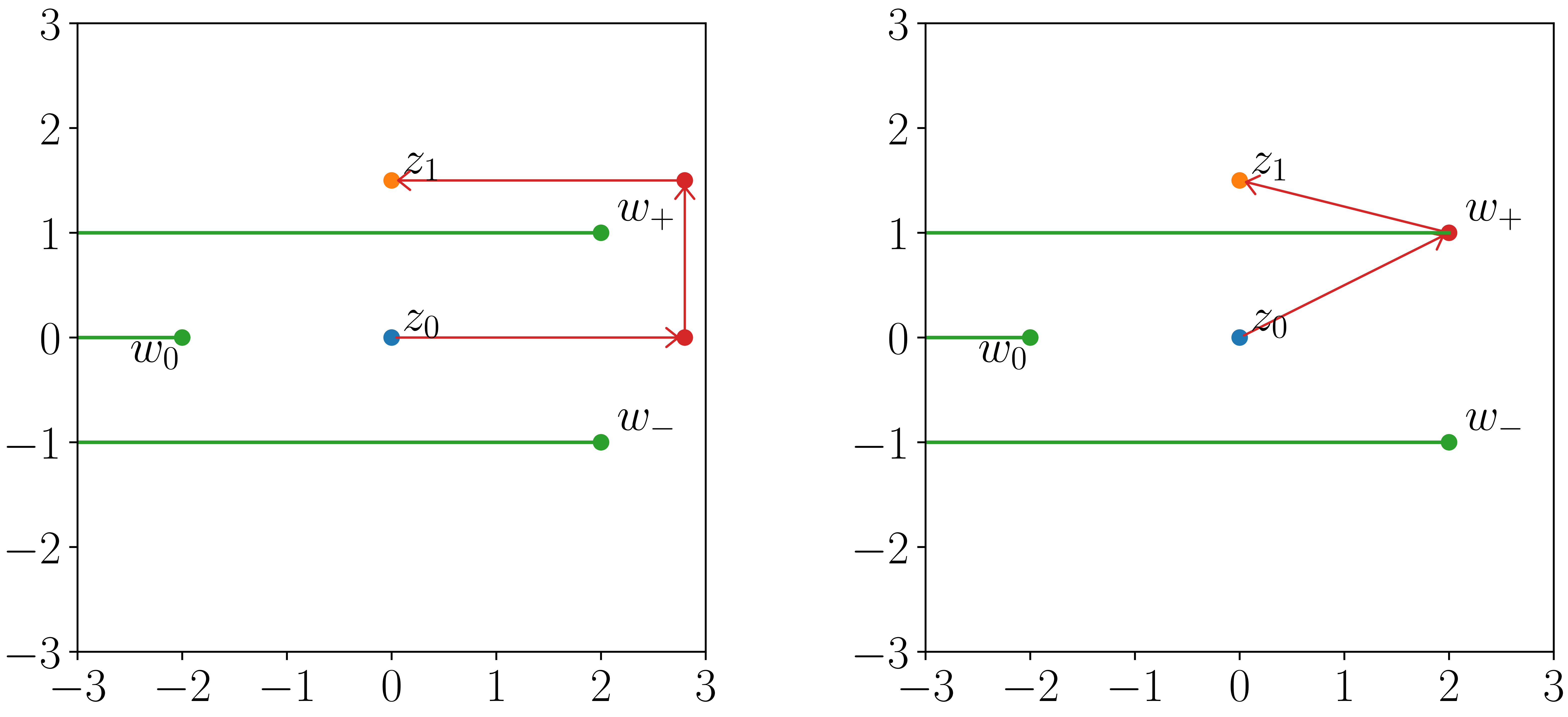
Precision

Number of terms	Precision	Execution time
50 terms	10^{-14}	$\sim$ 14 min
75 terms	10^{-19}	$\sim$ 26 min
100 terms	10^{-25}	$\sim$ 50 min
125 terms	10^{-33}	$\sim$ 75 min
150 terms	10^{-40}	$\sim$ 90 min

- The precision is controlled by the number of terms we decide to keep in the series expansion;
- The number of terms impact on the execution time.

Taylor vs Logarithmic

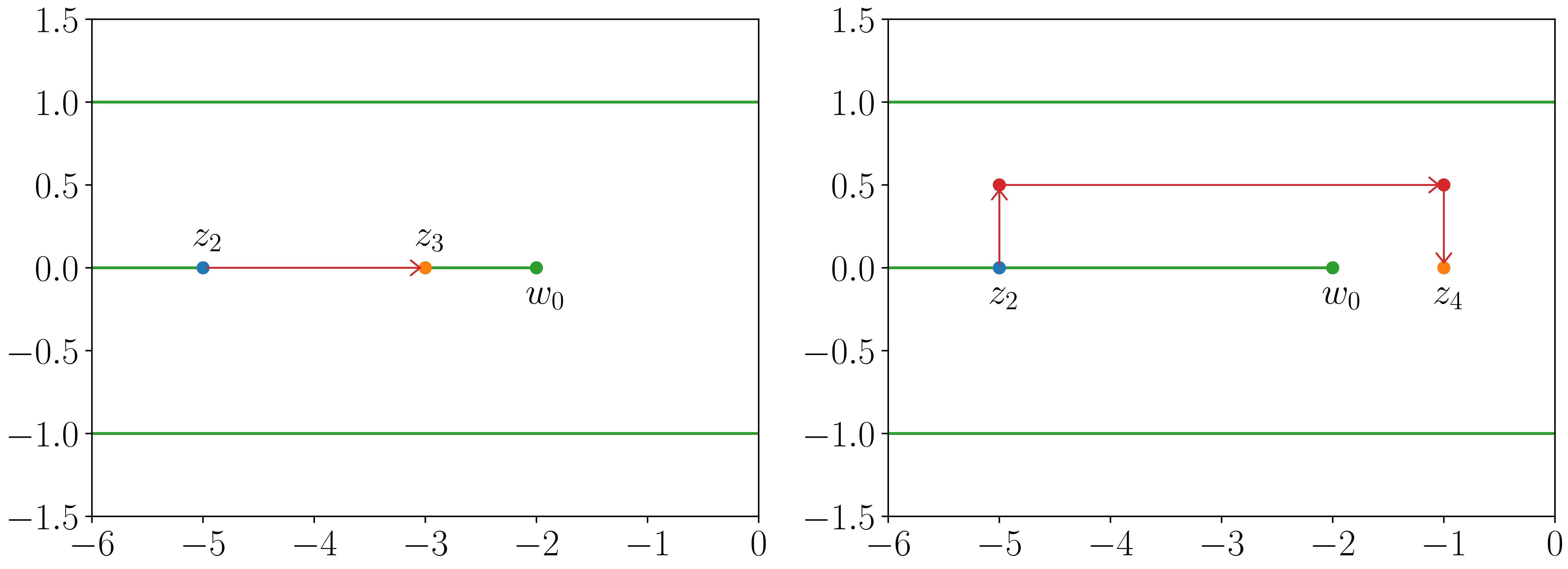
- When moving along an horizontal line, the Feynman prescription plays an important role



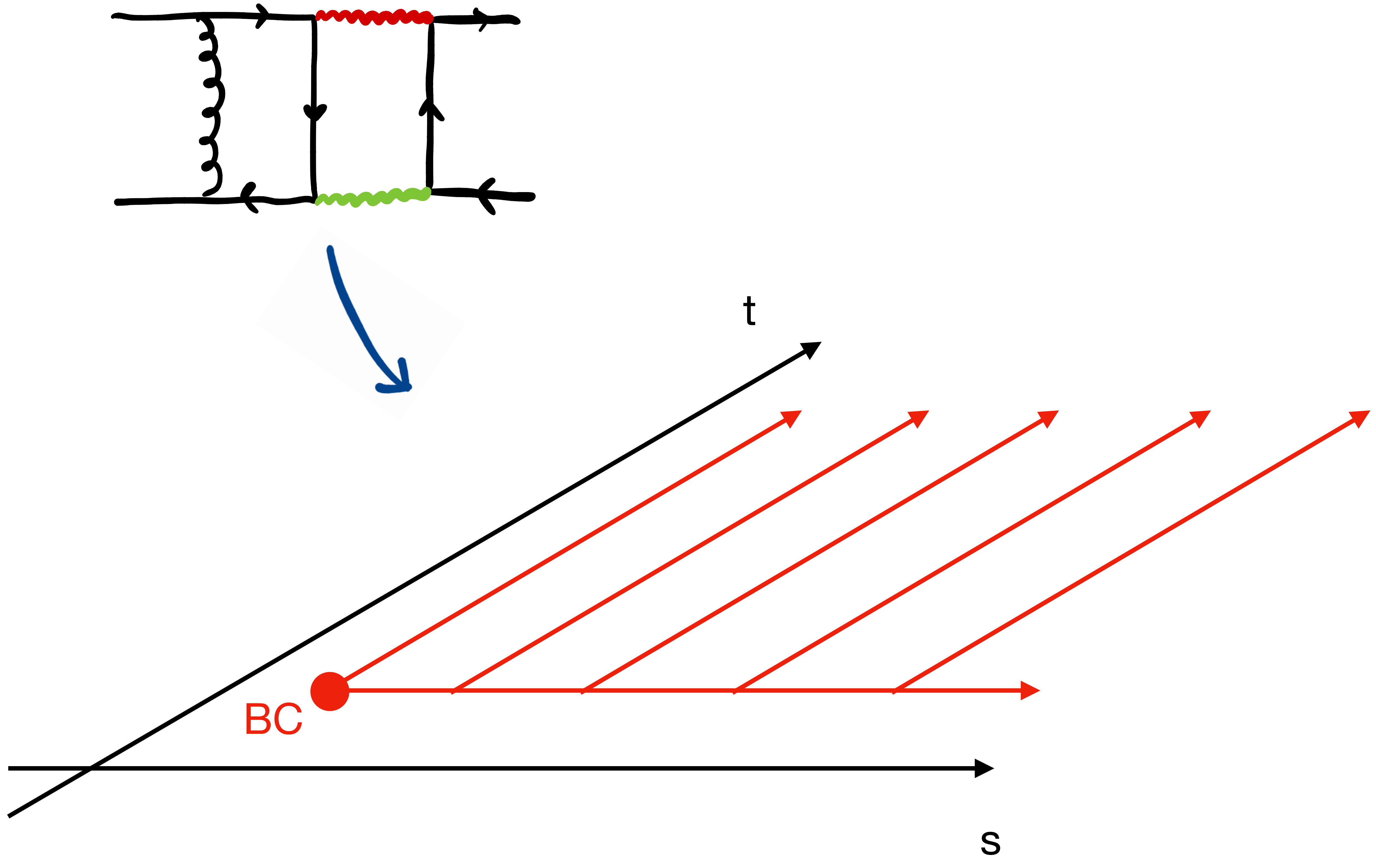
Analytic continuation

- When moving along an horizontal line, the **Feynman prescription** plays an important role

$$\frac{1}{s - m_V^2 + i\delta}$$



Creating a grid



- This approach is completely general and **easy to automate**;
- We have to solve a 56x56 system of differential equations w.r.t. to the Mandelstam variables s and t;
- Since we are not putting the system in canonical form, these are usually quite complicated and the solution might require some time;
- The computation of a grid with 3250 points required ∼3 weeks on 26 cores.