STR: a Mathematica package for the method of uniqueness

based on arXiv:1811.04935

Michelangelo Preti

Nordita

ACAT2019, Saas-Fee 14/3/2019
Introduction

High-precision computation in QFT

Multi-loop Feynman diagrams

Many analytic methods were developed:

- Integration by parts (IBP)  [A.N.Vasiliev, Y.M.Pismak, Y.R.Khonkonen '81][K.G.Chetyrkin, F.V.Tkachov '81]
- Gegenbauer polynomials  [K.G.Chetyrkin, A.L.Kataev, F.V.Tkachov '80]
- Mellin Transform  [M.C.Bergere, Y.M.P.Lam '74]  [N. I. Usyukina '75]
- Differential equations  [E. Remiddi '90]
Introduction

High-precision computation in QFT  Multi-loop Feynman diagrams

Many analytic methods were developed:

- Integration by parts (IBP) [A.N. Vasiliev, Y.M. Pismak, Y.R. Khonkonen ’81] [K.G. Chetyrkin, F.V. Tkachov ’81]
- Gegenbauer polynomials [K.G. Chetyrkin, A.L. Kataev, F.V. Tkachov ’80]
- Mellin Transform [M.C. Bergere, Y.M.P. Lam ’74] [N. I. Usyukina ’75]
- Differential equations [E. Remiddi ’90]
- Method of uniqueness (or Star-Triangle Relations) [M.D’Eramo, G. Parisi and L. Peliti ’71] [N. I. Usyukina ’83] [D. Kazakov ’83]

Reduction tool for complicated Feynman diagrams encoded in a sequence of simple transformations without performing any explicit integration.
**Introduction**

High-precision computation in QFT \[\leftrightarrow\] Multi-loop Feynman diagrams

Many analytic methods were developed:

- Integration by parts (IBP) \([A.N.\text{Vasiliev, Y.M.\text{Pismak, Y.R.\text{Khonkonen \text{\textquoteleft81}}}]}\) \([K.G.\text{Chetyrkin, F.V.\text{Tkachov \text{\textquoteleft81}}}]\]
- Gegenbauer polynomials \([K.G.\text{Chetyrkin, A.L.\text{Kataev, F.V.\text{Tkachov \text{\textquoteleft80}}}]}\]
- Mellin Transform \([M.C.\text{Bergere, Y.M.P.\text{Lam \text{\textquoteleft74}}}][N. \text{I. \text{Usyukina \text{\textquoteleft75}}}]\]
- Differential equations \([E. \text{Remiddi \text{\textquoteleft90}}}\]
- **Method of uniqueness** (or Star-Triangle Relations) \([M.D'\text{Eramo, G.\text{Parisi and L.\text{Peliti \text{\textquoteleft71}}}][N. \text{I. \text{Usyukina \text{\textquoteleft83}}}][D.\text{Kazakov \text{\textquoteleft83}}}]\]

Reduction tool for complicated Feynman diagrams encoded in a sequence of simple transformations without performing any explicit integration.

Analytic solutions for integrals (Example $\beta$-function at 5-loop in the $\phi^4$-theory) \([D.\text{Kazakov \text{\textquoteleft84}}}\]

Manifestly conformal structure \([E.S.\text{Fradkin, M.Y.\text{Palchik \text{\textquoteleft78}}}]\]

Relation with Yang-Baxter equation (Example *Fishnet* theories) \([L.\text{Lipatov \text{\textquoteleft92}}][V.\text{Kazakov et al \text{\textquoteleft16}}][V.\text{Kazakov, E.\text{Olivucci, M.P. \text{\textquoteleft18}}}]\]
The uniqueness method
The method of uniqueness

- Each vertex represents a point in the D-dimensional Euclidean space. Black dot vertices corresponds to integrated points over $\mathbb{R}^D$. 

![Diagram of method of uniqueness]
The method of uniqueness

- Each vertex represents a point in the D-dimensional Euclidean space. Black dot vertices corresponds to integrated points over $\mathbb{R}^D$.

- Lines labelled by the weight $\alpha$ are associated with the massless propagators

\[
\frac{1}{(x_{12}^2)^\alpha} = \frac{a_0(\alpha)}{4^\alpha \pi^{D/2}} \int d^D k \frac{e^{i k \cdot x_{12}}}{(k^2)^{D/2-\alpha}} \quad \text{and} \quad \frac{\Phi_{12}}{(x_{12}^2)^{\alpha+1/2}} = \frac{-i a_{1/2}(\alpha)}{4^\alpha \pi^{D/2}} \int d^D k \frac{e^{i k \cdot x_{12}} k}{(k^2)^{D/2-\alpha+1/2}},
\]

where

\[
a_\ell(\alpha) = \frac{\Gamma \left( \frac{D}{2} - \alpha + \ell \right)}{\Gamma(\alpha + \ell)} \quad \text{with} \quad a_\ell(\alpha) a_\ell(D/2 - \alpha) = 1
\]
The method of uniqueness

We call the weight of the diagram (or of a portion of it) the sum of all the weights of the constituent lines.
The method of uniqueness

We call the weight of the diagram (or of a portion of it) the sum of all the weights of the constituent lines.

Star of weight 4
The method of uniqueness

We call the weight of the diagram (or of a portion of it) the sum of all the weights of the constituent lines.

Star of weight 4

Triangle of weight 1+p
The method of uniqueness

We call the weight of the diagram (or of a portion of it) the sum of all the weights of the constituent lines.

Star of weight 4

Triangle of weight 1+p

A line, star and triangle are unique if their weights are 0, D and D/2 respectively
The method of uniqueness

We call the weight of the diagram (or of a portion of it) the sum of all the weights of the constituent lines.

A line, star and triangle are **unique** if their weights are 0, D and D/2 respectively.

Method of uniqueness is a set of relations for unique portions of diagram.

- Merging rules
- Star-Triangle relations
- Chain rules
Merging rules

Identities to represent a simple loop of propagators as a single line with different weight

Adjacent fermionic lines are contracted in their spin structure. Then merging a fermionic loop, the spin indices are carried by the D-dimensional identity matrix.
**Star-Triangle relations**

Identities between unique star and unique triangle

\[
\sum_k \alpha_k = D \quad \Rightarrow \quad \pi D/2 \, a_0(\alpha_1, \alpha_2, \alpha_3)
\]

where the function \( a_\ell \) of many arguments has the following property

\[
a_\ell(\alpha_1, \alpha_2, \ldots, \alpha_n) = \prod_{k=1}^{n} a_\ell(\alpha_k)
\]
Star-Triangle relations

Identities between unique star and unique triangle

\[
\int \frac{d^D x_0}{(x_{10}^2)^{\alpha_1} (x_{20}^2)^{\alpha_2} (x_{30}^2)^{\alpha_3}} \sum_k \alpha_k = D \quad \frac{\pi^{D/2}}{(x_{12}^2)^{D/2-\alpha_3} (x_{23}^2)^{D/2-\alpha_1} (x_{31}^2)^{D/2-\alpha_2}}
\]

where the function \( a_l \) of many arguments has the following property

\[
a_l(\alpha_1, \alpha_2, \ldots, \alpha_n) = \prod_{k=1}^{n} a_l(\alpha_k)
\]
Chain rules

Identities to integrate two propagators meeting in one internal point (a simple loop in momentum space) in terms of a single propagator

\[ = \pi^{D/2} a_0(\alpha_1, \alpha_2, D - \alpha_1 - \alpha_2) \]

This relations can be interpreted as Star-Triangle relations in which one of the propagators is sent to infinity.
Chain rules

Identities to integrate two propagators meeting in one internal point (a simple loop in momentum space) in terms of a single propagator

\[
\int \frac{d^D x_0}{(x_{10}^2)^{\alpha_1}(x_{02}^2)^{\alpha_2}} = \frac{\pi^{D/2} \mathfrak{a}_0(\alpha_1, \alpha_2, D - \alpha_1 - \alpha_2)}{(x_{12}^2)^{\alpha_1+\alpha_2-D/2}}
\]

\[
\int \frac{d^D x_0 \ \mathcal{F}_{10}}{(x_{10}^2)^{\alpha_1+1/2}(x_{02}^2)^{\alpha_2}} = \frac{\pi^{D/2} \mathfrak{a}_0(\alpha_2) \mathfrak{a}_{1/2}(\alpha_1, D - \alpha_1 - \alpha_2) \mathcal{F}_{12}}{(x_{12}^2)^{\alpha_1+\alpha_2-D/2+1/2}}
\]

\[
\int \frac{d^D x_0 \ \mathcal{F}_{10} \mathcal{F}_{02}}{(x_{10}^2)^{\alpha_1+1/2}(x_{02}^2)^{\alpha_2+1/2}} = -\frac{\pi^{D/2} \mathfrak{a}_0(D - \alpha_1 - \alpha_2) \mathfrak{a}_{1/2}(\alpha_1, \alpha_2)}{(x_{12}^2)^{\alpha_1+\alpha_2-D/2}}
\]

This relations can be interpreted as Star-Triangle relations in which one of the propagators is sent to infinity.
The **STR** package
The new functions

The package solves Feynman integrals by means of the method of uniqueness using a **graphical interactive** approach.
The new functions

The package solves Feynman integrals by means of the method of uniqueness using a **graphical interactive** approach

*Draw Phase*: It is possible to draw a diagram only clicking and dragging the mouse

*Computation Phase*: The uniqueness relations are applied selecting the right tool and clicking on the diagram
The new functions

The package solves Feynman integrals by means of the method of uniqueness using a **graphical interactive** approach

**Draw Phase**: It is possible to draw a diagram only clicking and dragging the mouse

**Computation Phase**: The uniqueness relations are applied selecting the right tool and clicking on the diagram

The functions in the package are:

- **STR[D]**: specifying the dimension of the Euclidean spacetime dimension, the function opens a graphical panel in which is possible to draw and modify the desired Feynman diagrams;

- **STRrelation**: it is a list of relations that identify the unique stars and triangles;

- **STRintegral**: it shows the integral representation of the diagram in STR;

- **STRprefactor**: it shows the prefactor of the integral STRintegral that contains all the functions generated by acting with the uniqueness method;

- **STRgraph**: it generates a modifiable version of the diagram drawn in STR;

- **STRSimplify[expr,D]**: specifying the dimension of the Euclidean spacetime dimension, it rewrites expr (the output of STRprefactor) in terms of Euler gammas.
The structure of the package
The structure of the package

Dimension D

STR

EventHandler

Graphics
The structure of the package

- Dimension D
- STR
  - EventHandler
  - Graphics
- Mouse
The structure of the package

- Dimension D
- STR
- EventHandler
- Graphics
- Mouse
- Computational tools

Diagram shows the relationships between these components.
The structure of the package

Dimension D

STR

EventHandler

Graphics

Mouse

Computational tools

Drawing tools
The structure of the package

Dimension D

EventHandler

Graphics

Mouse

Computational tools

Real-time graph

Drawing tools
The structure of the package

Dimension D

STR

EventHandler

Graphics

Computational tools

Drawing tools

Mouse

Real-time graph

STRrelations, STRprefactor, STRintegral and STRgraph
The function STR[D]

1. Graphical environment

2. Drawing tools

3. Relation tools

4. Computation tools

5. Edit buttons

6. Output tool
Let’s play with diagrams
Let's play with diagrams
Let’s play with diagrams
Let’s play with diagrams
Let’s play with diagrams
Let's play with diagrams
Let’s play with diagrams
Let’s play with diagrams

```
In[5]:= Str[D]

1
1

1

weight

Out[5]=

Add relations
- Add triangle relations
- Add star relations
- Clear selection
- Clear relations

Compute graph
- Flip arrow
- Merge
- Chain rule
- Triangle-star
- Star-triangle
- Clear prefactor

Print relations
Print prefactor
Print integral
Export
```
Let’s play with diagrams
Let’s play with diagrams
Let’s play with diagrams
Let’s play with diagrams
Let’s play with diagrams
Let’s play with diagrams
Let's play with diagrams
Let’s play with diagrams
Let's play with diagrams
Let’s play with diagrams

![Diagram with vertices and edges labeled with mathematical expressions]
Let's play with diagrams
Let’s play with diagrams
Let's play with diagrams

\[ \frac{\pi^{D/2}}{2} \alpha_0 [a] \alpha_1 [b, c] \]

\[ \Delta_0 [x_2, x_3] \Delta_1 [x_1, x_2] \Delta_1 [x_1, x_1] \]

\[ \frac{1}{2} (D-2a) \quad \frac{1}{2} (D-2b) \quad \frac{1}{2} (D-2c) \]

\[ \Delta [x_1, x_1, x_1, x_1] \]
Some physical examples
Example: The kite self-energy

\[ \begin{array}{c}
\begin{array}{c}
1 \\
1 \\
1
\end{array}
\end{array} \quad \begin{array}{c}
\begin{array}{c}
1 \\
1 \\
1
\end{array}
\end{array} \quad = \quad - \frac{1}{\epsilon} \left[ \begin{array}{c}
\begin{array}{c}
1 \\
2 \\
1
\end{array}
\end{array} \quad \begin{array}{c}
\begin{array}{c}
1 \\
1 \\
1
\end{array}
\end{array} \quad - \quad \begin{array}{c}
\begin{array}{c}
1 \\
2 \\
1
\end{array}
\end{array} \quad \begin{array}{c}
\begin{array}{c}
1 \\
1 \\
1
\end{array}
\end{array} \right] \]

They can be computed by the package in dimensional reg. running STR[4-2\epsilon].
Example: The kite self-energy

$$\begin{array}{c}
\begin{array}{c}
\begin{array}{c}
\begin{array}{c}
1 \quad 1
\end{array}
\begin{array}{c}
1
\end{array}
\begin{array}{c}
1
\end{array}
\end{array}
\begin{array}{c}
1
\end{array}
\begin{array}{c}
1
\end{array}
\end{array}
\end{array}
\end{array}
\begin{array}{c}
\begin{array}{c}
\begin{array}{c}
1
\end{array}
\begin{array}{c}
1
\end{array}
\begin{array}{c}
1
\end{array}
\end{array}
\begin{array}{c}
2
\end{array}
\begin{array}{c}
1
\end{array}
\end{array}
\end{array}
\end{array}
\begin{array}{c}
\begin{array}{c}
\begin{array}{c}
1
\end{array}
\begin{array}{c}
2
\end{array}
\begin{array}{c}
1
\end{array}
\end{array}
\begin{array}{c}
1
\end{array}
\begin{array}{c}
1
\end{array}
\end{array}
\end{array}
\end{array}
\begin{array}{c}
\begin{array}{c}
\begin{array}{c}
1
\end{array}
\begin{array}{c}
1
\end{array}
\begin{array}{c}
1
\end{array}
\end{array}
\begin{array}{c}
2
\end{array}
\begin{array}{c}
1
\end{array}
\end{array}
\end{array}
\end{array}
\begin{array}{c}
\begin{array}{c}
\begin{array}{c}
1
\end{array}
\begin{array}{c}
2
\end{array}
\begin{array}{c}
4
\end{array}
\end{array}
\begin{array}{c}
1
\end{array}
\begin{array}{c}
3
\end{array}
\end{array}
\end{array}
\end{array}
\end{array}
\end{array}
= - \frac{1}{\epsilon} \left[ \begin{array}{c}
\begin{array}{c}
\begin{array}{c}
1
\end{array}
\begin{array}{c}
1
\end{array}
\begin{array}{c}
1
\end{array}
\end{array}
\begin{array}{c}
2
\end{array}
\begin{array}{c}
1
\end{array}
\end{array}
\end{array}
\begin{array}{c}
\begin{array}{c}
\begin{array}{c}
1
\end{array}
\begin{array}{c}
1
\end{array}
\begin{array}{c}
1
\end{array}
\end{array}
\begin{array}{c}
2
\end{array}
\begin{array}{c}
1
\end{array}
\end{array}
\end{array}
\end{array}
\end{array}
\right]
$$

They can be computed by the package in dimensional reg. running STR[4-2eps]
Example: The kite self-energy

\[
\begin{align*}
\begin{array}{ccc}
1 & 1 & 1 \\
1 & 1 & 1 \\
\end{array} &= - \frac{1}{\epsilon} \left[
\begin{array}{ccc}
1 & 1 & 1 \\
2 & 1 & 1 \\
\end{array} - \begin{array}{ccc}
1 & 1 & 1 \\
2 & 1 & 1 \\
\end{array}\right]
\end{align*}
\]

They can be computed by the package in dimensional reg. running STR[4-2\epsilon]
Example: The kite self-energy

\[
\begin{align*}
\begin{array}{c}
\begin{array}{cc}
1 & 1 \\
1 & 1 \\
\end{array}
\end{array}
\end{align*}
\begin{array}{c}
\begin{array}{cc}
1 & 1 \\
2 & 1 \\
\end{array}
\end{array}
\begin{array}{cc}
1 & 1 \\
2 & 1 \\
\end{array}
\end{align*}
\begin{align*}
= - \frac{1}{\epsilon} \left[
\begin{array}{cc}
1 & 1 \\
2 & 1 \\
\end{array}
\end{align*}
\begin{align*}
\begin{array}{cc}
1 & 1 \\
2 & 1 \\
\end{array}
\end{align*}
\begin{array}{cc}
1 & 1 \\
2 & 1 \\
\end{array}
\right]
\end{align*}
\]

They can be computed by the package in dimensional reg. running STR[4-2\epsilon]
Example: The kite self-energy

They can be computed by the package in dimensional reg. running STR[4-2eps]
Example: The kite self-energy

\[
\begin{array}{ccc}
1 & 1 & 1 \\
1 & 1 & 1 \\
\end{array}
\quad = \quad -\frac{1}{\epsilon} \left[
\begin{array}{ccc}
1 & 1 & 1 \\
2 & 1 & 1 \\
\end{array}
\right] - 
\begin{array}{ccc}
1 & 1 & 1 \\
2 & 1 & 1 \\
\end{array}
\right]
\]

They can be computed by the package in dimensional reg. running STR[4-2eps]

\[
prefactor1 = STRprefactor
\]

\[
\pi^{4-2\,\text{eps}} \, a_0 \, [1, 1, 2 - 2\,\text{eps}] \, a_0 \, [1, 2, 1 - 2\,\text{eps}]
\]

\[
int1 = STRintegral
\]

\[
\Delta_0 \, [x_1, x_2]^{1+2\,\text{eps}}
\]
Example: The kite self-energy

\[ \frac{1}{\epsilon} \left[ \begin{array}{ccc} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{array} - \begin{array}{ccc} 1 & 1 & 1 \\ 2 & 1 & 1 \\ 1 & 1 & 1 \end{array} \right] \]

They can be computed by the package in dimensional reg. running STR[4-2eps]

prefactor1 = STRprefactor

\[ \pi^{4-2\epsilon} \alpha_0[1, 1, 2 - 2\epsilon] \alpha_0[1, 2, 1 - 2\epsilon] \]

int1 = STRintegral

\[ \Delta_0[x_1, x_2]^{1+2\epsilon} \]
Example: The kite self-energy

\[
\begin{array}{c}
\begin{array}{ccc}
1 & 1 & 1 \\
1 & 1 & 1 \\
\end{array}
\end{array}
\right] = -\frac{1}{\epsilon} \left[ \begin{array}{c}
\begin{array}{ccc}
1 & 1 & 1 \\
2 & 1 & 1 \\
\end{array}
\end{array} - \begin{array}{c}
\begin{array}{ccc}
1 & 1 & 1 \\
2 & 1 & 1 \\
\end{array}
\end{array} \right]
\]

They can be computed by the package in dimensional reg. running STR[4-2\epsilon]

\textbf{prefactor1 = STRprefactor}

\[\pi^{4-2\epsilon} a_0 [1, 1, 2 - 2\epsilon] a_0 [1, 2, 1 - 2\epsilon]\]

\textbf{int1 = STRintegral}

\[\Delta_0 [x_1, x_2]^{1+2\epsilon}\]
Example: The kite self-energy

\[
\begin{align*}
\begin{array}{c}
\frac{1}{\epsilon} \left[ \begin{array}{cc}
1 & 1 \\
2 & 1 \\
\end{array} \right] - \\
\begin{array}{c}
1 \\
2 \\
1 \\
\end{array}
\end{array}
\end{align*}
\]

They can be computed by the package in dimensional reg. running STR[4-2\text{eps}]

\begin{verbatim}
prefactor1 = STRprefactor
\pi^{4-2\text{eps}} \alpha_0[1, 1, 2-2\text{eps}] \alpha_0[1, 2, 1-2\text{eps}] 

int1 = STRintegral
\Delta_0[x_1, x_2]^{1+2\text{eps}}
\end{verbatim}
Example: The kite self-energy

\[
\begin{array}{c}
\includegraphics{example_diagram.png}
\end{array}
\]

They can be computed by the package in dimensional reg. running STR[4-2eps]

\textbf{prefactor1} = STR\textbf{prefactor}

\[\pi^{4-2\epsilon} \, a_0[1, 1, 2 - 2\epsilon] \, a_0[1, 2, 1 - 2\epsilon]\]

\textbf{int1} = STR\textbf{integral}

\[\Delta_0 [x_1, x_2]^{1+2\epsilon}\]
Example: The kite self-energy

\[
\begin{array}{ccc}
1 & 1 & 1 \\
1 & \Delta & 1 \\
1 & 1 & 1
\end{array} = -\frac{1}{\epsilon} \left[
\begin{array}{ccc}
1 & 1 & 1 \\
2 & 1 & 1 \\
2 & 1 & 1
\end{array} - \begin{array}{ccc}
1 & 1 & 1 \\
1 & \Delta & 1 \\
1 & 1 & 1
\end{array}\right]
\]

They can be computed by the package in dimensional reg. running STR[4-2eps]

\[
\begin{align*}
prefactor1 &= \text{STRprefactor} \\
prefactor2 &= \text{STRprefactor} \\
\int1 &= \text{STRintegral} \\
\int2 &= \text{STRintegral}
\end{align*}
\]

\[
\begin{align*}
prefactor1 &= \pi^{4-2\text{eps}} a_0[1, 1, 2 - 2\text{eps}] a_0[1, 2, 1 - 2\text{eps}] \\
prefactor2 &= \pi^{4-2\text{eps}} a_0[1, 1, 2 - 2\text{eps}] a_0[2, 1 - 3\text{eps}, 1 + \text{eps}] \\
\int1 &= \Delta_0[x_1, x_2]^{1+2\text{eps}} \\
\int2 &= \Delta_0[x_1, x_2]^{1+2\text{eps}}
\end{align*}
\]
Example: The kite self-energy

\[
\begin{pmatrix}
1 & 1 \\
1 & 1
\end{pmatrix} = -\frac{1}{\epsilon}
\begin{pmatrix}
1 & 1 \\
2 & 1
\end{pmatrix} -
\begin{pmatrix}
1 & 1 \\
2 & 1
\end{pmatrix}
\]

They can be computed by the package in dimensional reg. running STR[4-2eps]

\[
prefactor1 = \text{STRprefactor} \quad \pi^{4-2\epsilon} \quad \alpha[1,1,2-2\epsilon] \quad \alpha[1,2,1-2\epsilon]
\]

\[
int1 = \text{STRintegral} \\
\Delta_0[x_1, x_2]^{1+2\epsilon}
\]

\[
prefactor2 = \text{STRprefactor} \quad \pi^{4-2\epsilon} \quad \alpha[1,1,2-2\epsilon] \quad \alpha[2,1-3\epsilon,1+\epsilon]
\]

\[
int2 = \text{STRintegral} \\
\Delta_0[x_1, x_2]^{1+2\epsilon}
\]

Summing the two contribution together

\[
\text{Kite} = -\frac{1}{\epsilon} \quad (\text{prefactor1 int1 - prefactor2 int2}) \quad // \quad \text{FullSimplify}
\]

\[
\pi^{4-2\epsilon} \quad \alpha[1,1,2-2\epsilon] \quad (\alpha[1,2,1-2\epsilon] \quad - \quad \alpha[2,1-3\epsilon,1+\epsilon]) \quad \Delta_0[x_1, x_2]^{1+2\epsilon} \quad \frac{1}{\epsilon}
\]
Example: The kite self-energy

\[ \begin{align*}
1 & & 1 \\
1 & & 1 \\
1 & & 1 \\
\end{align*} = - \frac{1}{\epsilon} \left[ \begin{align*}
1 & & 1 \\
2 & & 1 \\
2 & & 1 \\
\end{align*} - \begin{align*}
1 & & 1 \\
1 & & 1 \\
\end{align*} \right] \\
\end{align*} \\
\]

They can be computed by the package in dimensional reg. running STR[4-2\epsilon]

\[ \text{prefactor1} = \text{STRprefactor} \]
\[ \pi^{4-2\epsilon} a_0 [1, 1, 2 - 2 \epsilon] a_0 [1, 2, 1 - 2 \epsilon] \]
\[ \text{int1} = \text{STRintegral} \]
\[ \Delta_0 [x_1, x_2]^{1+2 \epsilon} \]

\[ \text{prefactor2} = \text{STRprefactor} \]
\[ \pi^{4-2\epsilon} a_0 [1, 1, 2 - 2 \epsilon] a_0 [2, 1 - 3 \epsilon, 1 + \epsilon] \]
\[ \text{int2} = \text{STRintegral} \]
\[ \Delta_0 [x_1, x_2]^{1+2 \epsilon} \]

Summing the two contribution together

\[ \text{Kite} = - \frac{1}{\epsilon} \left( \text{prefactor1 int1} - \text{prefactor2 int2} \right) \text{ // FullSimplify} \]
\[ \pi^{4-2\epsilon} a_0 [1, 1, 2 - 2 \epsilon] (a_0 [1, 2, 1 - 2 \epsilon] - a_0 [2, 1 - 3 \epsilon, 1 + \epsilon]) \Delta_0 [x_1, x_2]^{1+2 \epsilon} \]

That correspond to the result in the literature [D.Kazakov ’84]. In particular for D=4

\[ \text{Series[STRSimplify[Kite, 4 - 2 \epsilon], \{\epsilon, 0, 0\}] // FullSimplify} \]
\[ 6 \pi^4 \text{Zeta}[3] \Delta_0 [x_1, x_2] + O[\epsilon]^1 \]
Example: Diagram with a fermionic loop

Let’s consider that three-point function of the beginning in D=4
Example: Diagram with a fermionic loop

Let’s consider that three-point function of the beginning in D=4
Example: Diagram with a fermionic loop

Let’s consider that three-point function of the beginning in D=4
Example: Diagram with a fermionic loop

Let's consider that three-point function of the beginning in D=4
Example: Diagram with a fermionic loop

Let’s consider that three-point function of the beginning in $D=4$
Let's consider that three-point function of the beginning in $D=4$. 

Example: Diagram with a fermionic loop
Example: Diagram with a fermionic loop

Let's consider that three-point function of the beginning in D=4
Example: Diagram with a fermionic loop

Let's consider that three-point function of the beginning in D=4
Example: Diagram with a fermionic loop

Let's consider that three-point function of the beginning in D=4
Example: Diagram with a fermionic loop

Let’s consider that three-point function of the beginning in $D=4$
Example: Diagram with a fermionic loop

Let's consider that three-point function of the beginning in D=4

\[ \text{STRprefactor} \]
\[ \pi^8 \text{tr}[\Pi] a_0[2-p] a_0[p] a_0[1, 2-p, 1+p] a_0[1, 3-p, p] a_1^{\frac{3}{2}} \left[ \frac{5}{2-p} a_1^{\frac{3}{2}} \left( \frac{1}{2} + p \right) \right] \]

\[ \text{STRintegral} \]
\[ \Delta_0[x_1, x_2]^{1-p} \Delta_0[x_1, x_3]^p \Delta_0[x_2, x_3]^p \]
Example: Diagram with a fermionic loop

Let's consider that three-point function of the beginning in D=4

**STRprefactor**

\[
\pi^8 \text{tr}[1] a_0[2-p] a_0[p] a_0[1, 2-p, 1+p] a_0[1, 3-p, p] a_1 \left[ \frac{3}{2}, \frac{5}{2} - p \right] a_1 \left[ \frac{3}{2}, \frac{1}{2} + p \right]
\]

**STRintegral**

\[
\Delta_0[x_1, x_2]^{1-p} \Delta_0[x_1, x_3]^p \Delta_0[x_2, x_3]^p
\]

Then using STRSimplify we obtain

\[
\]

\[
-2 \pi^8 \Delta_0[x_1, x_2]^{1-p} \Delta_0[x_1, x_3]^p \Delta_0[x_2, x_3]^p \frac{(-2 + p)^2}{(-1 + p)^2 p^2}
\]

Where we used tr[1]=2 in D=4.
Conclusions

Precision computations in perturbative QFT involve the evaluation of multi-loop diagrams. Number of diagrams grows rapidly with the perturbative order, then analytic tools are needed.
Conclusions

Precision computations in perturbative QFT involve the evaluation of multi-loop diagrams. Number of diagrams grows rapidly with the perturbative order, then analytic tools are needed.

Uniqueness method: reduction method for involved Feynman diagrams as a sequence of simple transformations.
Conclusions

Precision computations in perturbative QFT involve the evaluation of multi-loop diagrams.

Number of diagrams grows rapidly with the perturbative order, then analytic tools are needed.

Uniqueness method: reduction method for involved Feynman diagrams as a sequence of simple transformations.

Such a sequence could be extremely long and not unique.
Conclusions

Precision computations in perturbative QFT involve the evaluation of multi-loop diagrams. The number of diagrams grows rapidly with the perturbative order, which requires the use of analytic tools.

- **Uniqueness method:** reduction method for involved Feynman diagrams as a sequence of simple transformations.
  
  Such a sequence could be extremely long and not unique.

- **STR** Mathematica package:
  - Implementation in any Mathematica notebook, the uniqueness method in any dimension $D$.
  - User-friendly interface in which the user can interact with simple mouse moves.
  - Fully modifiable outputs.
Conclusions

Precision computations in perturbative QFT involve the evaluation of multi-loop diagrams. Number of diagrams grows rapidly with the perturbative order, then analytic tools are needed.

Uniqueness method: reduction method for involved Feynman diagrams as a sequence of simple transformations.

Such a sequence could be extremely long and not unique.

**STR** Mathematica package
- Implementation in any Mathematica notebook the uniqueness method in any dimension D
- User-friendly interface in which the user can interact with simple mouse moves
- Fully modifiable outputs

**Possible new implementations:**
- Higher spin particles
- Semi-uniqueness relations
Conclusions

Precision computations in perturbative QFT involve the evaluation of multi-loop diagrams. Number of diagrams grows rapidly with the perturbative order, then analytic tools are needed.

- Uniqueness method: reduction method for involved Feynman diagrams as a sequence of simple transformations.
- Such a sequence could be extremely long and not unique.

**STR Mathematica package**
- Implementation in any Mathematica notebook the uniqueness method in any dimension $D$
- User-friendly interface in which the user can interact with simple mouse moves
- Fully modifiable outputs

**Possible new implementations:**
- Higher spin particles
- Semi-uniqueness relations

THANK YOU!