

# STR: a Mathematica package for the method of uniqueness

Michelangelo Preti <sup>a,b</sup>

<sup>a</sup>*Laboratoire de Physique Théorique, École Normale Supérieure,  
24 rue Lhomond 75005 Paris, France*<sup>1</sup>

<sup>b</sup>*PSL Universités, Sorbonne Universités, UPMC Université Paris 6, CNRS*

## Abstract

We present **STR** (**Star-Triangle Relations**), a *Mathematica*<sup>®</sup> package designed to solve Feynman diagrams by means of the method of uniqueness in any Euclidean spacetime dimensions  $D$ . The method of uniqueness is a powerful technique to solve multi-loop Feynman integrals in theories with conformal symmetry imposing some relations between the powers of propagators and the spacetime dimension. In our algorithm we include both identities for scalar and Yukawa type integrals. The package provides a graphical environment in which is possible to draw the desired diagram with the mouse input and a set of tools to modify and compute it. Throughout the use of a graphic interface, the package should be easily accessible to users with little or no previous experience on diagrams computation. This manual includes some pedagogical examples of computation of Feynman graphs as the scalar two-loop kite master integral and a fermionic diagram appearing in the computation of the spectrum of the  $\gamma$ -deformed  $\mathcal{N} = 4$  SYM in the double-scaling limit.

---

<sup>1</sup>E-mail: [michelangelo.preti@lpt.ens.fr](mailto:michelangelo.preti@lpt.ens.fr)

# Program Summary

**Author:** Michelangelo Preti

**Title:** STR

**Licence:** GNU LGPL

**Programming language / external routines:** Wolfram *Mathematica*<sup>®</sup> version 8.0 or higher

**Operating system:** cross-platform

**RAM:** > 512 MB RAM recommended

**Current version:** 1.0

**Nature of the problem:** The main goal of a perturbative quantum field theory (QFT) is the exact computation of multi-loop Feynman diagrams that are crucial in many applications in particle, statistical and condensed matter physics. Since the number of diagrams grows rapidly with the perturbation theory order and the precision of numerical calculations is not satisfactory, the development of analytic tools to compute multi-loop integrals plays a central role in QFT. One of those powerful techniques is the *method of uniqueness* that provides a reduction method for involved Feynman diagrams as a sequence of simple transformations without performing any explicit integration. Considering that such a sequence could be extremely long and not unique, the development of a simple and automatic method is needed.

**Solution method:** The aim of the package STR is to solve Feynman diagrams by means of the method of uniqueness for any Euclidean spacetime dimension  $D$ . Its main feature is to provide a simple graphical environment to draw Feynman integrals and interact with them only by the use of the mouse. All the operations of the method of uniqueness are implemented in the same window as well as for the drawing and output tools.

**Web page:** <https://github.com/miciosca/STR>

**Contact:** [michelangelo.preti@lpt.ens.fr](mailto:michelangelo.preti@lpt.ens.fr) — for bugs, possible improvements and questions.

## 1 Introduction

In recent years, remarkable progresses have been achieved in evaluating higher-order corrections to various quantities in terms of multi-loop Feynman diagrams in perturbative quantum field theories (QFT). This investigation is stimulated by the efforts to make more precise predictions as well as a deeper understanding of the structure of perturbation series in QFT by attempting to resum it.

With this purpose, different methods of multi-loop calculations have been developed. One of the most efficient technique is the *integration by parts* method (IBP) [1, 2, 3] that allows to reduce an involved Feynman integral in terms of a base of a finite number of *master integrals*. This method is implemented in *Mathematica*<sup>®</sup> and C++ with the package FIRE5 [4] by means of the Laporta algorithm [5]. Sometimes the IBP method is not sufficient to solve the diagrams and then it has to be combined with other techniques as the one of *Gegenbauer polynomial* [6], or *Mellin transform* [7, 8], or *HQET* (see [9, 10, 11] for recent applications in 4 and 3 dimensions), or *differential equations* [12]. For a review of the various methods see also [13].

Another very powerful technique in multi-loop computations is the *method of uniqueness*, also known as *star-triangle relation*. In the following we will focus on this method. This relation was used the first time to solve three-dimensional integrals in the computation of the critical exponents of Bose liquids [14]. In the context of conformal field theories, the star-triangle relation was firstly considered in [15] and successively in the framework of multi-loop calculations in [1, 16, 17]. This method provides a reduction tool for complicated Feynman diagrams encoded in a sequence of simple transformations without performing any explicit integration. Once the appropriate sequence is found, the diagram can be easily integrated. For example, the analytic expression of the  $\beta$ -function at 5-loop in the  $\varphi^4$ -theory was computed with the uniqueness method in [18, 19]. Moreover, in [20] was pointed out that the star-triangle relation is a kind of Yang-Baxter equation. Indeed, this intuition was crucial for the computation of the sufficiently large *fishnet* diagrams in the  $\varphi^6$ ,  $\varphi^4$  and  $\varphi^3$  theories in  $D = 3$ ,  $D = 4$  and  $D = 6$  respectively and also for the R-matrix formulation [21] of the integrable non-compact Heisenberg spin-chain proposed in [22, 23]. The method of uniqueness hold also in the case we are considering a Yukawa theory with  $\psi\varphi\bar{\psi}$  vertices. Its generalization, including  $\gamma$ -matrices and propagators of spin particles, were considered in [24, 15, 25, 26].

In this paper we present **STR (Star-Triangle Relations)**, a *Mathematica*<sup>®</sup> package designed to solve Feynman diagrams by means of the method of uniqueness. We are interested to solve integrals in position space in any Euclidean spacetime dimension  $D$  both in the cases of scalar and fermionic star-triangle relations. The main feature of the package is to provide an user-friendly graphical way to solve Feynman integrals interacting with the algorithm only by the use of the mouse. Indeed, the main function of the package will furnish a graphical environment in which it is possible to draw any kind of Feynman diagram left- or right-clicking and dragging the mouse. Once the graph is set, the user can interact with it with the help of many tools designed to modify it, identify unique star or triangles and compute it (or part of it) by means of the uniqueness relations. At any step of the process, the user can also print or export the main data (list of the uniqueness equations for the weights, the result of the computations and the integral representation of the diagram) and the graph itself.

The paper is organized as follows. In section 2 we review the method of uniqueness for the scalar and fermionic cases. In section 3 we present in detail the manual of the **STR** package, while in section 4 we show some simple applications. An appendix follows, which contain a list of notation and conventions.

## 2 The star-triangle relations (method of uniqueness)

In this paper we will consider the usual Feynman diagrams with vertices connected by lines labeled by an index (weight). Each vertex represents a point in the  $D$ -dimensional Euclidean space  $\mathbb{R}^D$  while the lines of the graph (with weights  $\alpha_i$ ) are associated with the following massless

propagators ( $x_{ij} \equiv x_i - x_j$ )

$$\overset{\alpha}{\text{---}}_{x_1} \text{---} x_2 = \frac{1}{(x_{12}^2)^\alpha} \quad \text{and} \quad \overset{\alpha}{\text{---}}_{x_1} \text{---} x_2 = \frac{\not{x}_{12}}{(x_{12}^2)^{\alpha+1/2}}, \quad (2.1)$$

respectively the scalar and spin-1/2 fermionic massless propagators. The symbol  $\not{x}$  stands for the contraction between the position  $x^\mu$  and the element of the Clifford algebra in  $D$  dimensions, namely  $\sigma_\mu x^\mu$  or  $\bar{\sigma}_\mu x^\mu$  depending if the fermion is chiral or anti-chiral in even dimensions and  $\gamma_\mu x^\mu$  in odd dimensions (see Appendix A). The black dot vertices denote that the corresponding points are integrated over  $\mathbb{R}^D$ . All the computation will be performed in coordinate space. However one can also choose to compute the Feynman integrals in momentum space using the following Fourier transforms

$$\frac{1}{(x_{12}^2)^\alpha} = \frac{\mathfrak{a}_0(\alpha)}{4^\alpha \pi^{D/2}} \int d^D k \frac{e^{ik \cdot x_{12}}}{(k^2)^{D/2-\alpha}} \quad \text{and} \quad \frac{\not{x}_{12}}{(x_{12}^2)^{\alpha+1/2}} = \frac{-i \mathfrak{a}_{1/2}(\alpha)}{4^\alpha \pi^{D/2}} \int d^D k \frac{e^{ik \cdot x_{12}} \not{k}}{(k^2)^{D/2-\alpha+1/2}}, \quad (2.2)$$

where

$$\mathfrak{a}_\ell(\alpha) = \frac{\Gamma(\frac{D}{2} - \alpha + \ell)}{\Gamma(\alpha + \ell)} \quad \text{with} \quad \mathfrak{a}_\ell(\alpha) \mathfrak{a}_\ell(D/2 - \alpha) = 1. \quad (2.3)$$

We call the *weight of the diagram* (or of a portion of it) the sum of all the weights of the constituent lines. For instance, the weight of a three-leg vertex (also called *star*) is defined as the sum of the weights of the three lines that converge at it, and the weight of a *triangle* is defined as the sum of the weights of the three lines that form it. We shall say that a line, star and triangle are *unique* if their weights are 0,  $D$  and  $D/2$  respectively. If a Feynman diagram contains unique stars or triangles, its computation is drastically simplified. Indeed the so-called *method of uniqueness*, based on a set of relations between unique stars and triangles, provides a simple evaluation of massless Feynman diagrams according to the rules listed below. In this section we are reviewing [17, 19, 16, 27, 26].

**1. Merging rules:** In the following, we list a set of identities to represent simple loop of propagators as a single line with different weight. It is straightforward to see that in the case in which more than two lines are passing through the same two points, the loop can be written as a combination of the following simple rules.

- The contribution of a simple loop of bosonic propagators is an ordinary product

$$\begin{array}{c} \alpha_1 \\ \text{---} \\ \text{---} \\ \alpha_2 \\ \text{---} \end{array} \text{---} x_1 \text{---} x_2 = \overset{\alpha_1 + \alpha_2}{\text{---}}_{x_1} \text{---} x_2. \quad (2.4)$$

- The contribution of a simple loop of a bosonic and a fermionic propagator is a fermionic

line with weight given by the sum of the original weights

$$\begin{array}{c} \alpha_1 \\ \curvearrowright \\ x_1 \qquad x_2 \\ \curvearrowleft \\ \alpha_2 \end{array} = x_1 \xrightarrow{\alpha_1 + \alpha_2} x_2 . \quad (2.5)$$

- The contribution of a simple loop of fermionic propagators, since we are considering the spin structure to be contracted, is equal to a bosonic line with weight given by the sum of the original weights

$$\begin{array}{c} \alpha_1 \\ \curvearrowright \\ x_1 \qquad x_2 \\ \curvearrowleft \\ \alpha_2 \end{array} = \mathbb{1} \ x_1 \xrightarrow{\alpha_1 + \alpha_2} x_2 . \quad (2.6)$$

To compute it we used the  $D$ -dimensional Clifford algebras (A.2) for odd  $D$  and (A.8) for even  $D$ . We are considering that adjacent fermionic lines are contracted in the spin structure alternating  $\sigma$ 's and  $\bar{\sigma}$ 's (even  $D$ ) or simply contracting  $\gamma$ 's (odd  $D$ )<sup>2</sup>, then the bosonic propagator in the right-hand side of the formula will carry the spin indices in the identity matrix  $\mathbb{1}$  that corresponds to  $\mathbb{1}_{2^{D/2-1}}$  for even  $D$  or  $\mathbb{1}_{2^{(D-1)/2}}$  for odd  $D$  (see Appendix A).

**2. Star-triangle relations:** Those are the main identities of the uniqueness method that allows to integrate a unique star into a unique triangle. In this paper we will consider only unique stars and triangles, even if there are also some *semi-unique*<sup>3</sup> identities allowing to integrate a star in a linear combination of other stars and triangles with different weights. In the following we will present the relations in the case of a full bosonic and a Yukawa type stars/triangles.

- The bosonic star-triangle relation can be written in the following graphical representation

$$\begin{array}{c} x_1 \\ | \\ \alpha_1 \\ \bullet \\ \alpha_3 \quad \alpha_2 \\ / \quad \backslash \\ x_3 \quad x_2 \end{array} \stackrel{\sum_k \alpha_k = D}{=} \pi^{D/2} \mathfrak{O}_0(\alpha_1, \alpha_2, \alpha_3) \begin{array}{c} x_1 \\ / \quad \backslash \\ D/2 - \alpha_2 \quad D/2 - \alpha_3 \\ \backslash \quad / \\ D/2 - \alpha_1 \\ x_3 \quad x_2 \end{array} , \quad (2.7)$$

where the function  $\mathfrak{O}_\ell$  of many arguments has the following property

$$\mathfrak{O}_\ell(\alpha_1, \alpha_2, \dots, \alpha_n) = \prod_{k=1}^n \mathfrak{O}_\ell(\alpha_k) . \quad (2.8)$$

<sup>2</sup>For instance in this case the numerator is: for even  $D$   $\not{x}_{12} \not{x}_{12} = (\sigma_\mu)_{\alpha\dot{\alpha}} (\bar{\sigma}_\nu)^{\dot{\alpha}\beta} x_{12}^\mu x_{12}^\nu = x_{12}^2 (\mathbb{1}_{2^{D/2-1}})_\alpha^\beta$  and for odd  $D$   $\not{x}_{12} \not{x}_{12} = (\gamma_\mu)_\alpha^\beta (\gamma_\nu)_\beta^\gamma x_{12}^\mu x_{12}^\nu = x_{12}^2 (\mathbb{1}_{2^{(D-1)/2}})_\alpha^\gamma$ .

<sup>3</sup>By *semi-uniqueness* we mean star and triangles one-step-deviating from uniqueness, in other words they have weight  $D - 1$  and  $D/2 + 1$  respectively [16].

The same relation can be written in its integral form

$$\int \frac{d^D x_0}{(x_{10}^2)^{\alpha_1} (x_{20}^2)^{\alpha_2} (x_{30}^2)^{\alpha_3}} \stackrel{\sum_k \alpha_k = D}{=} \frac{\pi^{D/2} \mathfrak{O}_0(\alpha_1, \alpha_2, \alpha_3)}{(x_{12}^2)^{D/2-\alpha_3} (x_{23}^2)^{D/2-\alpha_1} (x_{31}^2)^{D/2-\alpha_2}}. \quad (2.9)$$

- The Yukawa star-triangle relation can be written in the following graphical representation

$$\text{Star Graph} \stackrel{\sum_k \alpha_k = D}{=} \pi^{D/2} \mathfrak{O}_0(\alpha_1) \mathfrak{O}_{1/2}(\alpha_2, \alpha_3) \text{Triangle Graph}. \quad (2.10)$$

The same relation can be written in its integral form

$$\int \frac{d^D x_0 \not{x}_{02} \not{x}_{30}}{(x_{10}^2)^{\alpha_1} (x_{02}^2)^{\alpha_2+1/2} (x_{30}^2)^{\alpha_3+1/2}} \stackrel{\sum_k \alpha_k = D}{=} \frac{\pi^{D/2} \mathfrak{O}_0(\alpha_1) \mathfrak{O}_{1/2}(\alpha_2, \alpha_3) \not{x}_{12} \not{x}_{31}}{(x_{12}^2)^{D/2-\alpha_3+1/2} (x_{23}^2)^{D/2-\alpha_1} (x_{31}^2)^{D/2-\alpha_2+1/2}}. \quad (2.11)$$

Notice that the order of contraction of the  $\sigma$  matrices is not changing after the integration, indeed

$$\begin{aligned} \text{even } D \quad \not{x}_{02} \not{x}_{30} &= (\sigma_\mu \bar{\sigma}_\nu)_\alpha^\beta x_{02}^\mu x_{30}^\nu \implies (\sigma_\mu \bar{\sigma}_\nu)_\alpha^\beta x_{12}^\mu x_{31}^\nu = \not{x}_{12} \not{x}_{31}, \\ \text{odd } D \quad \not{x}_{02} \not{x}_{30} &= (\gamma_\mu \gamma_\nu)_\alpha^\beta x_{02}^\mu x_{30}^\nu \implies (\gamma_\mu \gamma_\nu)_\alpha^\beta x_{12}^\mu x_{31}^\nu = \not{x}_{12} \not{x}_{31}, \end{aligned} \quad (2.12)$$

and the same if we are choosing the case in which  $\sigma \leftrightarrow \bar{\sigma}$ .

**3. Chain rules:** This set of identities is needed to integrate two propagators meeting in one internal point (a simple loop in momentum space) in terms of a single propagator. Multiple chains are integrated successively by means of a combination of the following rules.

- A chain of bosonic propagators is integrated by means of the following identity

$$x_1 \xrightarrow{\alpha_1} \bullet_{x_0} \xrightarrow{\alpha_2} x_2 = \pi^{D/2} \mathfrak{O}_0(\alpha_1, \alpha_2, D - \alpha_1 - \alpha_2) x_1 \xrightarrow{\alpha_1 + \alpha_2 - D/2} x_2, \quad (2.13)$$

that in the integral form is

$$\int \frac{d^D x_0}{(x_{10}^2)^{\alpha_1} (x_{02}^2)^{\alpha_2}} = \frac{\pi^{D/2} \mathfrak{O}_0(\alpha_1, \alpha_2, D - \alpha_1 - \alpha_2)}{(x_{12}^2)^{\alpha_1 + \alpha_2 - D/2}}. \quad (2.14)$$

Notice that this identity can be interpreted as the star-triangle relation (2.7) in which the missing weight  $\alpha_3 = D - \alpha_1 - \alpha_2$  is determined by the uniqueness requirement.

- A chain of a fermionic and a bosonic propagator is integrated by means of the following

identity

$$\begin{array}{c} \alpha_1 \\ \xrightarrow{\quad} \\ x_1 \end{array} \xrightarrow{\quad} \begin{array}{c} \alpha_2 \\ \bullet \\ x_0 \end{array} \xrightarrow{\quad} x_2 = \pi^{D/2} \mathfrak{O}_0(\alpha_2) \mathfrak{O}_{1/2}(\alpha_1, D - \alpha_1 - \alpha_2) \begin{array}{c} \alpha_1 + \alpha_2 - D/2 \\ \xrightarrow{\quad} \\ x_1 \end{array} \xrightarrow{\quad} x_2, \quad (2.15)$$

that in the integral form is

$$\begin{aligned} \int \frac{d^D x_0 \not{x}_{10}}{(x_{10}^2)^{\alpha_1+1/2} (x_{02}^2)^{\alpha_2}} &= -\frac{i \mathfrak{O}_{1/2}(\alpha_1) \mathfrak{O}_0(\alpha_2)}{4^{\alpha_1+\alpha_2-D/2}} \int \frac{d^D k e^{ik \cdot x_{12}} \not{k}}{(k^2)^{D-\alpha_1-\alpha_2+1/2}} \\ &= \frac{\pi^{D/2} \mathfrak{O}_0(\alpha_2) \mathfrak{O}_{1/2}(\alpha_1, D - \alpha_1 - \alpha_2) \not{x}_{12}}{(x_{12}^2)^{\alpha_1+\alpha_2-D/2+1/2}}, \end{aligned} \quad (2.16)$$

where in the intermediate step we have used the Fourier representation of the propagators (2.2) and then, transforming back to coordinate space, the identity (2.3). Notice that, as the bosonic case, this identity can be interpreted as the star-triangle relation (2.10) in which, after relabeling the weights as  $\alpha_3 \rightarrow \alpha_1$ ,  $\alpha_1 \rightarrow \alpha_2$  and  $\alpha_2 \rightarrow \alpha_3$ , the missing weight  $\alpha_3 = D - \alpha_1 - \alpha_2$  is determined by the uniqueness requirement.

- A chain of fermionic propagators is integrated by means of the following identity

$$\begin{array}{c} \alpha_1 \\ \xrightarrow{\quad} \\ x_1 \end{array} \xrightarrow{\quad} \begin{array}{c} \alpha_2 \\ \bullet \\ x_0 \end{array} \xrightarrow{\quad} x_2 = -\pi^{D/2} \mathfrak{O}_0(D - \alpha_1 - \alpha_2) \mathfrak{O}_{1/2}(\alpha_1, \alpha_2) \mathbb{1}_{x_1} \xrightarrow{\quad} \begin{array}{c} \alpha_1 + \alpha_2 - D/2 \\ \xrightarrow{\quad} \\ x_2 \end{array}. \quad (2.17)$$

Since we are considering that adjacent fermionic lines are contracted in the spin structure (see footnote 2), in the right-hand side of the formula, it appears the identity matrix  $\mathbb{1}$  that corresponds to  $\mathbb{1}_{2^{D/2-1}}$  for even  $D$  or  $\mathbb{1}_{2^{(D-1)/2}}$  for odd  $D$  (see Appendix A). In the integral form the previous identity is

$$\begin{aligned} \int \frac{d^D x_0 \not{x}_{10} \not{x}_{02}}{(x_{10}^2)^{\alpha_1+1/2} (x_{02}^2)^{\alpha_2+1/2}} &= -\frac{\mathfrak{O}_{1/2}(\alpha_1, \alpha_2)}{4^{\alpha_1+\alpha_2-D/2}} \mathbb{1} \int \frac{d^D k e^{ik \cdot x_{12}}}{(k^2)^{D-\alpha_1-\alpha_2}} \\ &= -\mathbb{1} \frac{\pi^{D/2} \mathfrak{O}_0(D - \alpha_1 - \alpha_2) \mathfrak{O}_{1/2}(\alpha_1, \alpha_2)}{(x_{12}^2)^{\alpha_1+\alpha_2-D/2}}, \end{aligned} \quad (2.18)$$

where in the intermediate step, we have used the Fourier representation of the propagators (2.2) and then, transforming back to coordinate space, the identity (2.3). Notice that this identity can be interpreted as the star-triangle relation (2.10) in which, after relabeling the weights with  $\alpha_3 \leftrightarrow \alpha_1$ , the missing weight  $\alpha_3 = D - \alpha_1 - \alpha_2$  is determined by the uniqueness requirement.

### 3 The STR *Mathematica*<sup>®</sup> package

In the following we introduce the *Mathematica*<sup>®</sup> package STR and its usage. The purpose of the package is to simplify the computation of Feynman integrals by means of the *uniqueness method* that we reviewed in the previous section, using a graphical interactive approach. In this section we present how to setup the package in a *Mathematica*<sup>®</sup> notebook and a detailed manual.

### 3.1 Download and installation

The set up of the package is very simple<sup>4</sup>. The package files can be downloaded from the arXiv servers in the source directory of this paper but, considering the possibility of updates of the code to correct bugs or to add new features, we suggest to download it from the GitHub repository at the following address

<https://github.com/miciosca/STR>.

In order to load the package in a *Mathematica*<sup>®</sup> notebook, one has to save the file `STR.m` in the same directory of the notebook and run the command

```
In[1]:= SetDirectory[NotebookDirectory[]];
      << STR.m
```

### 3.2 Manual

Running the following line

```
In[3]:= ? STR`*
```

it is possible to list the new functions available with the package. Those functions are the following

- `STR[dimension]`: 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 in the STR graphical environment;
- `STRintegral`: it shows the integral representation of the integral drawn in the STR graphical environment;
- `STRprefactor`: it shows the prefactor of the integral `STRintegral` that contains all the functions generated by acting with the uniqueness method in the STR graphical environment;
- `STRgraph`: it generates a modifiable version of the diagram drawn in the STR graphical environment;
- `STRSimplify[expr, dimension]`: specifying the dimension of the Euclidean spacetime *dimension*, it rewrites *expr* (the output of `STRprefactor`) in terms of Euler gammas by means of (2.3) and (2.8).

In the following paragraphs we will present in detail all those functions.

---

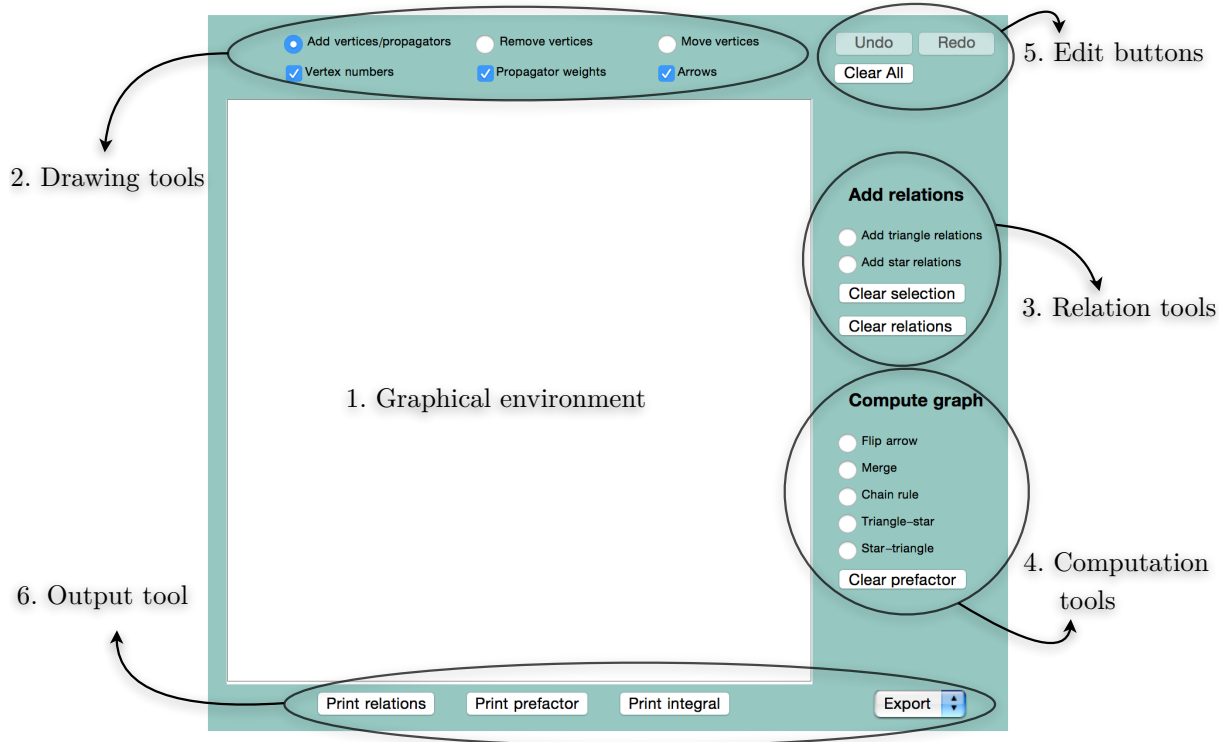
<sup>4</sup>For more detailed set up instructions see also [28]



**STR:** After the package is loaded, run the function `STR` specifying the dimension of the spacetime without semicolon at the end. For example

```
In[3]:= STR[4]
```

where we selected the dimension of the Euclidean spacetime to be  $D = 4$ . In general the user can choose any dimension in the argument of `STR`, even with some non-numerical character as needed, for instance, in dimensional regularization. The output will be the following interactive panel



The purpose of this function is to allow the user to draw a diagram only using the mouse and interact with it to compute it (or part of it) by means of the uniqueness method (see section 2). Any operation in this panel will affect the diagrams and the output in real time. This panel can be divided in six groups of functions (as highlighted above in the figure) presented in details in the following list.

1. **Graphical environment:** This white window is the portion of the interactive panel in which the user is allowed to draw Feynman diagrams interacting with the mouse. The algorithm tracks the position of the mouse cursor and draws vertices and propagators clicking or dragging (see "Drawing tools" for a detailed guide to drawing). This space is obviously not continuous but it is a lattice with spacing optimized considering the dimensions of the graphical elements that represents vertices and propagators. All the operations that the user can do on the graphs can be done in this space selecting the appropriate option in the tools around the window.
2. **Drawing tools:** These are the main tools to draw and modify diagrams in the graphical

environment. Any modification made when those option are selected will affect the content of the functions `STRintegral` and `STRgraph`.

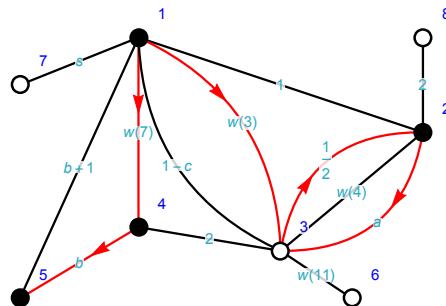
- **Add vertices/propagators:** When this option is selected the user can draw Feynman diagrams in the graphical environment only using the mouse. A single left- or right-click in a empty space of the window will place an isolated external (not integrated) point represented by a white dot and labeled by a number  $k$  identifying its position  $x_k$ . Clicking on an already placed external point will turn the white dot into a black one that represent and internal (integrated on  $\mathbb{R}^D$ ) point. Repeating this action again, it is possible to turn the color of the vertices any number of times.

To add a propagator, the user has to left- or right-click, drag and release the mouse on the desired points of the interactive window. Doing this procedure with the left button of the mouse one will place a scalar propagator represented by a black line as in (2.1), with the right button one will place a fermionic propagator represented by a red line with an arrow as in (2.1). The starting/ending point can be any empty spot of the window or even an already placed white or black vertex. When the mouse is dragged it will appear a dashed temporary line (black or red depending if the propagator is bosonic of fermionic) in order to track the creation of the line. When the mouse is released. if the spot is empty, a vertex will be drawn in the releasing position. Moreover, the dashed line will disappear and it will be substituted with the final propagator. In the case of a fermionic propagator, the arrow points always to the releasing point. In the case in which between two vertices there are more then one line, those will be represented as curved lines giving the possibility to the user to distinguish all of them and see their related weights.

When a propagator is placed, it will appear a pop-up window with an input field in it. The user has to specify there the name of the weight of the propagator (namely  $\alpha$  in (2.1)) that can contains any number, letter and arithmetic operation. Pressing **Enter** or clicking on the "OK" button, the name of the edge weight will be updated on the graphical environment. If the weight of the propagator is not specified, it will be named as  $w(k)$  with  $k$  an increasing integer.

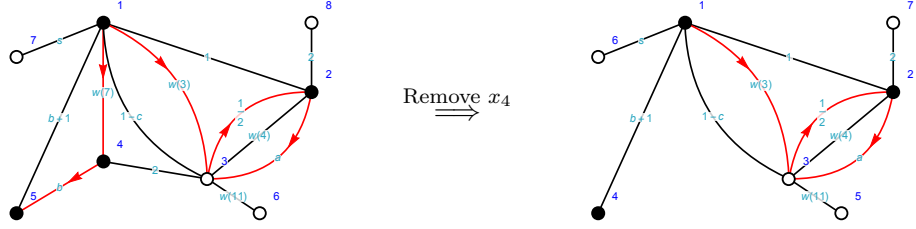
With the standard appearance options, all the vertices will appear with an unique number  $k = 1, \dots, n$  with  $n$  the total number of vertices, all the fermionic propagators with an arrow pointing in the chosen direction and all the propagator with a label representing their weights.

As an example see the following diagram



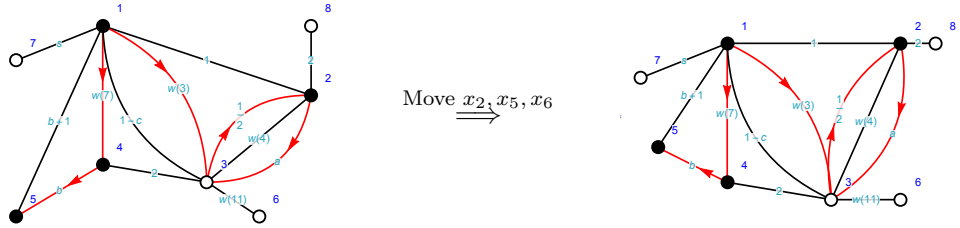
which contains 4 internal integrated points (black), 4 external non-integrated points (white), 8 scalar propagators (black lines) and 5 fermionic propagators (red lines with arrows). Notice that adjacent fermionic lines are considered contracted in their spin indices carried by  $\sigma$  or  $\gamma$  matrices. If more than two fermionic lines are merging in the same vertex (see the external point labeled by 3), the algorithm doesn't know which propagators are contracted between themselves or if, for example, the ones in the loop are traced. In this case, since all the possible operations with multiple fermions are proportional to  $\mathbb{1}$  (see section 2), the identity will be explicitly written in the final result and then the user can choose if contract it with the other fermionic structures or trace it.

- **Remove vertices:** Selecting this tool, the user can erase from the diagram an internal or external vertex and all the propagators attached to it by left-clicking on it. The remaining vertices will be renamed such that their labels will run from 1 to the total number of vertices. For example



where we removed the vertex in the position  $x_4$ .

- **Move vertices:** Selecting this tool, the user can modify the shape of an existing diagram moving the vertices in the graphical environment. Left-clicking on a vertex and dragging it, it will be placed in a new position together with the propagators connected to it. When the node is in the final desired position, the user has to release the mouse. The positions of propagator weights, vertex numbers and arrows will be updated in real-time tracking the mouse cursor. For example



where we moved the vertices in the position  $x_2$ ,  $x_5$  and  $x_6$ .

- **Vertex numbers:** Checking/unchecking this option, the numerical labels written near the vertices will appear/disappear from the graphical environment.
- **Propagator weights:** Checking/unchecking this option, the labels representing the propagator weights will appear/disappear from the graphical environment.
- **Arrows:** Checking/unchecking this option, the arrows of the fermionic propagators will appear/disappear from the graphical environment.

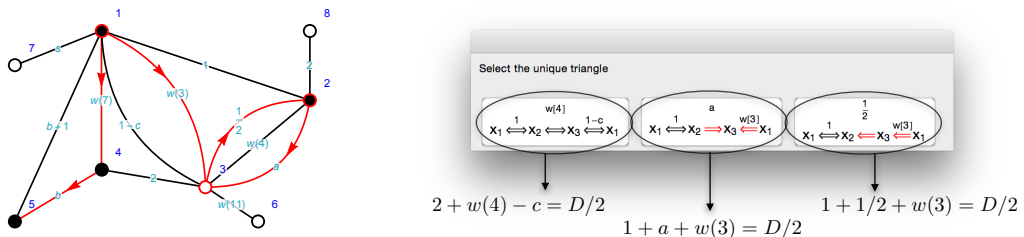
3. **Relation tools:** These tools allow the user to identify unique stars or triangles in the graph. Those options are helpful only when the weights of stars or triangles are not automatically true or false: for instance when the weights of the propagators are non-numeric symbols and we want to impose a relation between those labels and the spacetime dimension. When a star or a triangle is identified as unique by those options, the function `STRrelations` is updated. If the user will add a new uniqueness relation that is automatically false or incompatible with the other in `STRrelations`, he will be notified by a pop-up window that the selected star or triangle cannot be unique.

- **Add triangle relations:** Selecting this tool the user can identify a triangle in the diagram imposing the uniqueness relation on its propagator weights  $\alpha_k$  such that  $\sum_k \alpha_k = D/2$  with  $k = 1, 2, 3$  as presented in section 2.

In order to select the desired triangle directly on the graph, after selecting `Add triangle relations`, the user has to left- or right-click on the three black or white dots at the vertices of it. Notice that in this package we are providing solutions for scalar and Yukawa triangles as reviewed in section 2. When a vertex is selected, it will be highlighted by changing the color of its border from black to red. In the case in which the vertices are not connected by propagators or if they don't identify a bosonic or Yukawa triangle, a pop-up window will notify the user about the issue, and the selection will be reset. If the selected vertices identify a proper triangle, the uniqueness relation for its weights will be registered updating the function `STRrelations`. Finally the user will be notified by a pop-up window that the relation is added successfully. Clicking on the "OK" button or pressing `Enter`, the window will be closed and all the vertices will be deselected turning back their borders from red to black.

If between two selected points there are more than one propagator, it could happen that the same set of vertices identify more than one triangle. In this case, it will appear a pop-up window containing a list of buttons that represent all the possible scalar and Yukawa *sub-triangles* passing for the selected set of vertices. Any button appears with a schematic representation of the related sub-triangle: the position of the vertices  $x_k$  in the order selected by the user connected by black double arrows or red standard arrows representing respectively scalar and fermionic propagators with their directions. Any arrows is labelled by the related propagator weight.

For example



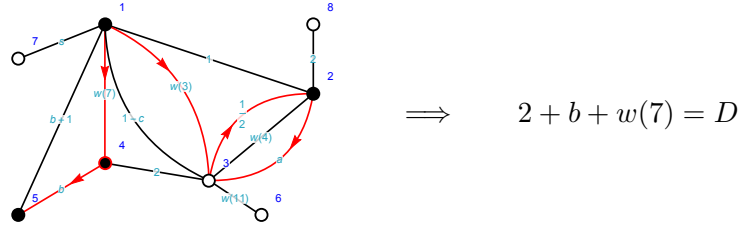
we want to identify as an unique triangle the Yukawa one with weights 1,  $a$  and  $w(3)$  in the diagram considered above. Selecting the three vertices at the positions  $x_1$ ,

$x_2$  and  $x_3$ , the color of the vertices border turns to red as in figure and the pop-up window in the right will appear. The possible triangles for those vertices are three and our choice corresponds to the one in the middle. Then the `STRrelations` function will be updated with the uniqueness relation  $1 + a + w(3) = D/2$  with a notification of the successful addition.

- **Add star relations:** Selecting this tool the user can identify a star in the diagram imposing the uniqueness relation on its propagator weights  $\alpha_k$  such that  $\sum_k \alpha_k = D$  with  $k = 1, 2, 3$  as presented in section 2.

In order to select the desired star directly on the graph, after selecting **Add star relations**, the user has to left- or right-click on a black (integrated) vertex in which three propagators merge. Notice that in this package we are providing solutions for scalar and Yukawa stars as reviewed in section 2. Once the vertex is selected, it will be highlighted by changing the color of its border from black to red. In the case in which the vertex is not integrated (white dot) or more than three propagators are meeting in it or it doesn't identify a scalar or Yukawa star, a pop-up window will notify the user about the issue, and the selection will be reset. If the selected vertex identifies a proper star, the uniqueness relation for its weights will be registered updating the function `STRrelations`. Finally the user will be notified by a pop-up window that the relation is added successfully. Clicking on the "OK" button or pressing **Enter**, the window will be closed and all the vertices will be deselected turning back their borders from red to black.

For example



we want to identify as an unique star, the one with weights 2,  $b$  and  $w(7)$  in the diagram considered above. Selecting the vertex at the positions  $x_4$ , the color of its border turns to red as in figure and the `STRrelations` function is updated with the uniqueness relation  $2 + b + w(7) = D$  with a notification of the successful addition.

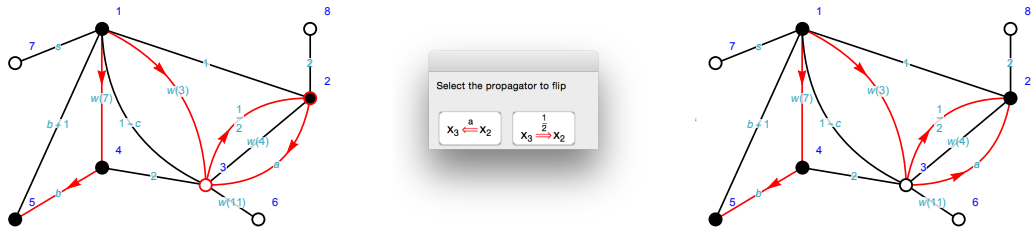
- **Clear selection:** Clicking on this button, the user can deselect the vertices previously selected. In other words if some vertex has red border, the border will be turned back to black.
- **Clear relations:** Clicking on this button, the user can reset the `STRrelations` function erasing all the uniqueness relations saved in it.

4. **Computation tools:** All the main operations that the user can do on the graph are included in this set of tools. Indeed, it allows to use the uniqueness method reviewed in section 2 on the diagram drawn with the `Drawing tools`. All those operations update in real-time the functions `STRprefactor`, `STRintegral` and `STRgraph`.

- **Flip arrow:** Selecting this tool, the user can flip the direction of the arrow of a chosen fermionic propagator. In order to select the propagator, one has to left- or right-click on the two vertices at the endpoints of it. The border of the selected vertices will turn from black to red to enlightening it and the arrow will be flipped updating the `STRprefactor` function with a minus sign.

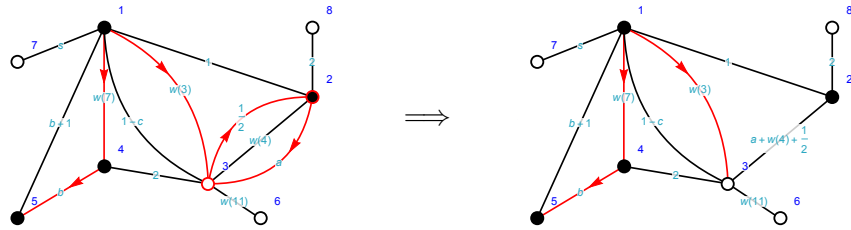
If between the selected vertices there are more than one fermionic propagators, the user has to choose which arrow has to be flipped between them. This choice can be done through a pop-up window containing a number of buttons equal to the number of fermionic propagators between the two points. Any button contains a schematic representation of the related propagator, in particular the direction of the arrow and its weight. Once the user selects the desired button, the arrow is flipped and the function `STRprefactor` is updated in the same way of the previous case.

For example



where we flipped the fermionic line passing through the vertices in the positions  $x_2$  and  $x_3$  with weight  $a$ .

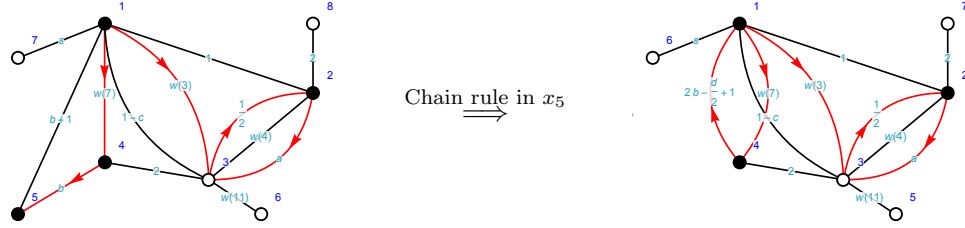
- **Merge:** With this tool, the user can combine together multiple propagators lying between two vertices. Selecting two vertices connected by more than one propagator, the color of their border will turn from black to red to highlight them and all the propagators between them will collapse in a single one following the rules give in (2.4), (2.5) and (2.6). In the case in which the fermionic propagators have different arrow orientations, the algorithm will take into account the phase generated by the flipping. All the functions `STRprefactor`, `STRintegral` and `STRgraph` will be updated. For example



where we merged the propagators between the vertices in the positions  $x_2$  and  $x_3$  according with the rules (2.5) and (2.6). Notice that this operation generates a scalar propagator from a fermionic loop, then in the updated version of `STRprefactor` it will appear the identity  $\mathbb{1}$ .

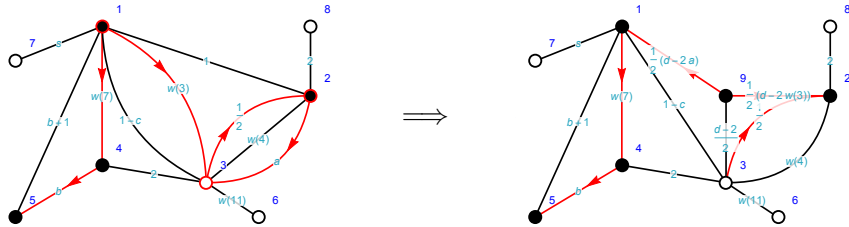
- **Chain rule:** Selecting this tool, the user can solve the integral related to one internal (black) vertex connected with two propagators using the chain rules (2.13), (2.15) and

(2.17). In order to solve the integral, one has only to left- or right-click on the desired internal vertex on the diagram. Then the graph will be modified and the functions `STRprefactor`, `STRintegral` and `STRgraph` will be updated accordingly. For example



where we solved the integral in the position  $x_5$  with the rule (2.15) in dimension  $D = d$ . Notice that in the resulting diagram some lines are overlapping but obviously they are not crossing.

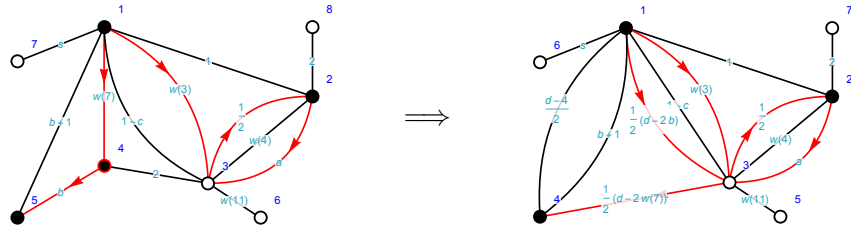
- **Triangle-star:** Choosing this tool, the user can select a triangle in the diagram turning it into a star adding an integration point (black dots) according to the star-triangle relations (2.7) and (2.10). The procedure to select the triangle is exactly the same presented for the tool `Add triangle relations`. If the selected triangle is not unique, a pop-up window will notify it, otherwise it will turn in a star. In the case in which the selected vertices identify more than one triangle, as in the case examined in `Add triangle relations` section, the user has to choose between all the possible sub-triangles in a pop-up window. Then the graph will be modified and the functions `STRprefactor`, `STRintegral` and `STRgraph` will be updated accordingly. For example



where we selected the three vertices at the positions  $x_1$ ,  $x_2$  and  $x_3$  and then the triangle with weights 1,  $a$  and  $w(3)$  that we set to be unique with `Add triangle relations`. To compute the new diagram, the package is using the (inverse) star-triangle relation (2.10) in dimension  $D = d$  adding a new integrated vertices in position  $x_9$ .

- **Star-triangle:** With this tool, the user can select a star in the diagram turning it into a triangle solving the integration in the internal vertex (black dot) according to the star-triangle relations (2.7) and (2.10). The procedure to select the star is exactly the same presented for the tool `Add star relations`. If the selected star is not unique, a pop-up window will notify it, otherwise it will turn in a triangle. Then the graph will be modified and the functions `STRprefactor`, `STRintegral` and `STRgraph` will

be updated accordingly. For example



where we selected the vertex at the position  $x_4$  identifying the star with weights 2,  $b$  and  $w(7)$  that we set to be unique with **Add star relations**. To compute the new diagram, the package is using the star-triangle relation (2.10) in dimension  $D = d$  solving the integral in position  $x_4$ .

- **Clear prefactor:** Clicking on this button the user can reset the function **STRprefactor** at its initial value 1.

5. **Edit buttons:** Those buttons are self-explanatory, in particular

- **Undo/Redo:** The **Undo** button erases the most recent action the user did in the interactive panel. The **Redo** button is needed to re-apply or undo what the user just undid one time. Both of the buttons can be used repeatedly to apply their effects multiple times.
- **Clear All:** This button is needed to reset the working-space to the initial conditions: the diagram in the graphical environment is erased and the **STRprefactor**, **STRintegral**, **STRrelations** and **STRgraph** function restored to their original value.

6. **Output tools:** These buttons allow the user to print or export the data computed using all the previous tools.

- **Print relations:** This button prints below the **STR** interactive panel the list of uniqueness relations encoded in the function **STRrelations**.
- **Print prefactor:** This button prints below the **STR** interactive panel the result of the operations made on the diagram via the **Computation tools** encoded in the function **STRprefactor**.
- **Print integral:** This button prints below the **STR** interactive panel the integral representation of the diagram currently drawn in the graphical environment also encoded in the function **STRintegral**.
- **Export:** This button will open a menu in which is possible to choose to export data or the diagram. Choosing **Data**, all the functions **STRprefactor**, **STRintegral** and **STRrelations** will be exported and they will be ready to be modified in the *Mathematica*<sup>®</sup> notebook. Choosing **Graph**, the function **STRgraph** will be updated containing exactly the diagram drawn in the graphical environment.

**STRrelations:** This function doesn't need arguments. Writing in the notebook **STRrelations** and pressing Shift-Enter the output will be a list of equations involving the propagator weights



and the spacetime dimension registered from the STR interactive window using `Add triangle relations` and `Add star relations`. This function is updated in real-time but to use it in the *Mathematica*<sup>®</sup> notebook one needs to press `Export` and then `Data` in the STR window. For example the two relations added in the previous examples are registered in `STRrelations` as follows

**STRrelations**

$$\{2(1+a+w[3]) == d, 2+b+w[7] == d\}$$

**STRprefactor and STRintegral:** Those functions don't need arguments. They represent together the integral representation of the diagram in the graphical environment. Indeed, for a given graph, its integral representation respect to the initial diagram can be written as `STRprefactor`  $\int$  `STRintegral`. The `STRprefactor` function contains the numerical prefactor of the integral initially set to 1 and then updated with all the contributions given by the computations on the graph by means of the `Computation tools`. The `STRintegral` function contains the integrand related to the diagram drawn in the interactive panel in terms of product of propagators. The points that are integrated are represented by their measure  $d^D x_i$ . If in `STRintegral` no integration measures are present, means that no integrals are left and all the points are external (white dots). Those functions are updated in real-time but to use them in the *Mathematica*<sup>®</sup> notebook one needs to press `Export` and then `Data` in the STR window. For example, let's start with the same diagram of before

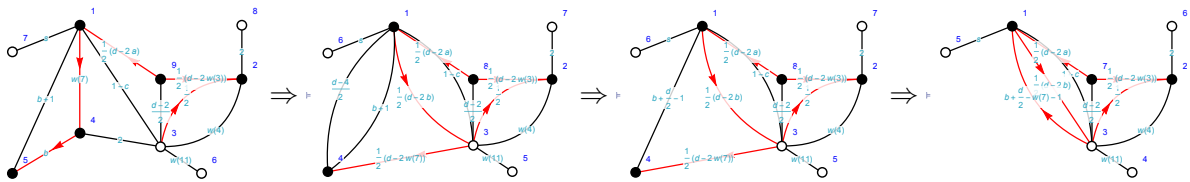
**STRprefactor**  
: 1

**STRintegral**  
:  $\int d\mathbf{x}_1 d\mathbf{x}_2 d\mathbf{x}_4 d\mathbf{x}_5 \Delta_0[\mathbf{x}_1, \mathbf{x}_2] \Delta_0[\mathbf{x}_1, \mathbf{x}_3]^{1-c} \Delta_0[\mathbf{x}_1, \mathbf{x}_5]^{1+b} \Delta_0[\mathbf{x}_1, \mathbf{x}_7]^a \Delta_0[\mathbf{x}_2, \mathbf{x}_3]^{w[4]} \Delta_0[\mathbf{x}_2, \mathbf{x}_8]^2 \Delta_0[\mathbf{x}_3, \mathbf{x}_4]^2 \Delta_0[\mathbf{x}_3, \mathbf{x}_6]^{w[11]} \Delta_{\frac{1}{2}}[\mathbf{x}_1, \mathbf{x}_3]^{w[3]} \Delta_{\frac{1}{2}}[\mathbf{x}_1, \mathbf{x}_4]^{w[7]} \Delta_{\frac{1}{2}}[\mathbf{x}_2, \mathbf{x}_3]^a \sqrt{\Delta_{\frac{1}{2}}[\mathbf{x}_3, \mathbf{x}_2]} \Delta_{\frac{1}{2}}[\mathbf{x}_4, \mathbf{x}_5]^b$

where

$$d\mathbf{x}_i \Rightarrow d^D x_i \quad \text{and} \quad \Delta_k[\mathbf{x}_i, \mathbf{x}_j]^\alpha \Rightarrow \frac{\not{x}_{ij}^{2k}}{(x_{ij}^2)^{\alpha+k}} \quad (3.1)$$

Setting the unique relations between weights as in the previous example in `STRrelations` with  $D = d$ , we can use the `Computation tools` to compute part of the integral as follows



and then we can read the result exporting again the updated functions `STRprefactor` and `STRintegral` that now are

**STRprefactor**

$$\pi^{d/2} a_0[1] a_0[2] a_0\left[-1 + b + \frac{d}{2}\right] a_{\frac{1}{2}}[a, w[3]] a_{\frac{1}{2}}[b, w[7]] a_{\frac{1}{2}}\left[\frac{d}{2} - w[7], 1 - b + w[7]\right]$$

**STRintegral**

$$\int dx_1 dx_2 dx_7 \Delta_0[x_1, x_3]^{1-c} \Delta_0[x_1, x_5]^s \Delta_0[x_2, x_3]^{w[4]} \Delta_0[x_2, x_6]^2 \Delta_0[x_3, x_4]^{w[11]} \Delta_0[x_3, x_7]^{\frac{1}{2}(-2+d)} \\ \Delta_{\frac{1}{2}}[x_1, x_3]^{\frac{1}{2}(-2b+d)} \Delta_{\frac{1}{2}}[x_3, x_1]^{-1+b+\frac{d}{2}-w[7]} \sqrt{\Delta_{\frac{1}{2}}[x_3, x_2]} \Delta_{\frac{1}{2}}[x_7, x_1]^{\frac{1}{2}(-2a+d)} \Delta_{\frac{1}{2}}[x_7, x_2]^{\frac{1}{2}(d-2w[3])}$$

where

$$a_k[\alpha_1, \alpha_2, \dots, \alpha_n] \Rightarrow \mathfrak{O}_k(\alpha_1, \alpha_2, \dots, \alpha_n), \tag{3.2}$$

defined in (2.3) and (2.8).

In the output, it's possible that the following symbols appear

$$\text{tr}[\mathbb{1}] = \text{tr}(\mathbb{1}) \quad \text{and} \quad \mathbb{1}[i, j] = \mathbb{1}, \tag{3.3}$$

where the first is the trace of the identity computed following appendix A and the second is the identity inserted in the diagram between the points  $i$  and  $j$ . Notice that in the computation of the diagram the name of the positions  $i$  and  $j$  can change, than the user has to track the original position of the identity to the new one.

**STRgraph:** This function doesn't need arguments. Writing in the notebook `STRgraph` and pressing Shift-Enter the output will be a `Graphics[] Mathematica`® object reproducing the current diagram in the graphical environment of the STR interactive window. This function is updated in real-time but to use it in the *Mathematica*® notebook one needs to press `Export` and then `Graph` in the STR window.

**STRSimplify:** This function, given the spacetime dimension, will rewrite the output of the function `STRprefactor` in terms of known functions following the definitions (2.3) and (2.8).

## 4 Some explicit examples

In this section we will present two examples of computation of Feynman diagrams with the method of uniqueness by means of STR package.

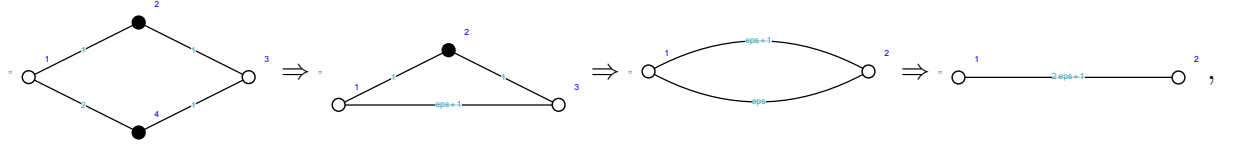
### 4.1 Example 1: The kite self-energy diagram

Let's start with a very basic example of computation of a Feynman diagram using the STR package. We want to compute the two loop scalar *kite* self-energy diagram in dimensional regularization  $D = 4 - 2\epsilon$  in the case in which all the propagators weights are  $\alpha_i = 1$  with

$i = 1, \dots, 5$ . This diagram was computed in [18] by means of the uniqueness method. It is well known that this diagram can be written as follows

$$\begin{array}{c} \diagup \\ | \\ \diagdown \end{array} = -\frac{1}{\epsilon} \left[ \begin{array}{c} \diagup \\ | \\ \diagdown \\ \text{---} \\ \diagup \\ | \\ \diagdown \end{array} - \begin{array}{c} \diagup \\ | \\ \diagdown \end{array} \right], \quad (4.1)$$

using the semi-uniqueness method (see footnote 3). Both of the diagrams in the right side can be computed with STR running the command STR[4-2eps] where eps is the dimensional regulator. Indeed the first one, using Chain rule and Merge can be transformed as follows



where in the last step we have

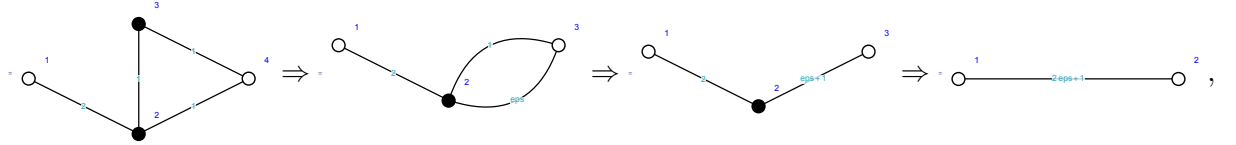
**prefactor1 = STRprefactor**

$$\pi^{4-2\text{eps}} \mathfrak{a}_0[1, 1, 2-2\text{eps}] \mathfrak{a}_0[1, 2, 1-2\text{eps}]$$

**int1 = STRintegral**

$$\Delta_0[\mathbf{x}_1, \mathbf{x}_2]^{1+2\text{eps}}$$

Similarly we can compute the second diagram with the following steps



where in the last step we have

**prefactor2 = STRprefactor**

$$\pi^{4-2\text{eps}} \mathfrak{a}_0[1, 1, 2-2\text{eps}] \mathfrak{a}_0[2, 1-3\text{eps}, 1+\text{eps}]$$

**int2 = STRintegral**

$$\Delta_0[\mathbf{x}_1, \mathbf{x}_2]^{1+2\text{eps}}$$

Summing up the two contributions according to (4.1), we obtain

$$\text{Kite} = -\frac{1}{\text{eps}} (\text{prefactor1 int1} - \text{prefactor2 int2}) // \text{FullSimplify}$$

$$\frac{\pi^{4-2\text{eps}} \mathfrak{a}_0[1, 1, 2-2\text{eps}] (\mathfrak{a}_0[1, 2, 1-2\text{eps}] - \mathfrak{a}_0[2, 1-3\text{eps}, 1+\text{eps}]) \Delta_0[\mathbf{x}_1, \mathbf{x}_2]^{1+2\text{eps}}}{\text{eps}}$$

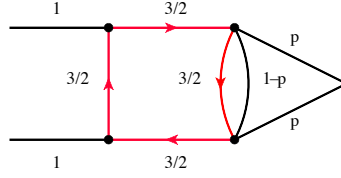
matching the result of [18]. Now, using the function STRSimplify and expanding around eps=0, we can compute the value of the Feynman diagram in  $D = 4$

$$\begin{array}{c} \diagup \\ | \\ \diagdown \end{array} = \text{Series}[\text{STRSimplify}[\text{Kite}, 4-2\text{eps}], \{\text{eps}, 0, 0\}] // \text{FullSimplify}$$

$$6 \pi^4 \text{Zeta}[3] \Delta_0[\mathbf{x}_1, \mathbf{x}_2] + \mathcal{O}[\text{eps}]^1$$

## 4.2 Example 2: Diagram with a fermionic loop

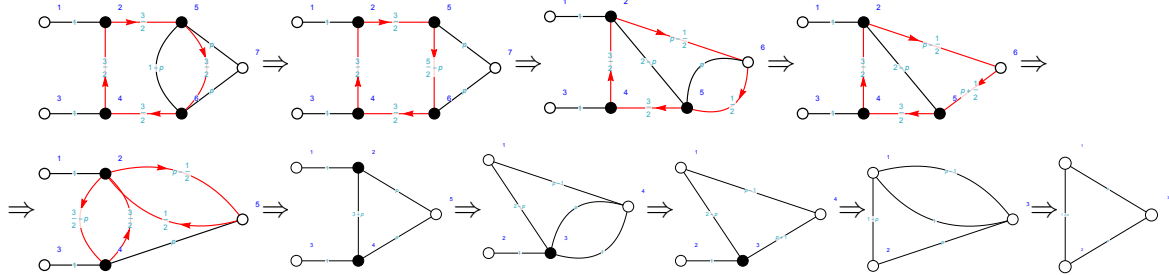
Here we present an example of computation of a Feynman diagram involving Yukawa star-triangle relations in  $D = 4$ . The diagram is the following



where  $p$  is real number and the chiral fermionic propagators are contracted in the spin indices forming a trace of  $\sigma$ -matrices.

This graph arises in the computation of the spectrum of length-2 operators in the  $\gamma$ -deformed  $\mathcal{N} = 4$  SYM theory in the double scaling limit [29]. This theory, proposed in [30, 31], is a 3-parameters deformation of the usual  $\mathcal{N} = 4$  SYM theory which breaks the  $R$ -symmetry down to  $U(1)^3$  subgroup. In this framework, it is possible to consider a double scaling limit<sup>5</sup>, combining large imaginary twists and the weak coupling limit [37]. Then the gauge fields decouple and we obtain a theory of three scalars and fermions interacting with quartic and Yukawa couplings. In order to compute the spectrum of the unprotected operators of length-2, we have to solve the diagram above that it can be computed straightforwardly with the uniqueness method with the help of the package STR.

Running the command `STR[4]` and drawing the diagram above, we can solve it with the following steps



where we used the tools `Star-triangle` and `Merge multiedge` several times. In the last step we don't have any residual integration and we obtain

**STRprefactor**

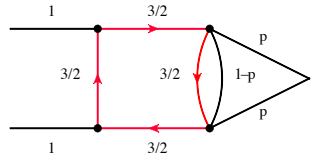
$$\pi^8 \text{tr}[1] a_0[2-p] a_0[p] a_0[1, 2-p, 1+p] a_0[1, 3-p, p] a_{\frac{1}{2}}\left[\frac{3}{2}, \frac{5}{2}-p\right] a_{\frac{1}{2}}\left[\frac{3}{2}, \frac{1}{2}+p\right]$$

**STRintegral**

$$\Delta_0[\mathbf{x}_1, \mathbf{x}_2]^{1-p} \Delta_0[\mathbf{x}_1, \mathbf{x}_3]^p \Delta_0[\mathbf{x}_2, \mathbf{x}_3]^p$$

<sup>5</sup>Similar double scaling limits were also studied in [32, 33, 34, 35, 36] in different contexts.

Now using the function `STRSimplify`, we can conclude that



$$= \frac{\text{STRSimplify}[\text{STRprefactor STRintegral}, 4] /. \text{tr}[\mathbb{1}] \rightarrow 2 // \text{FullSimplify} // \text{Factor}}{(-2 + p)^2 (-1 + p)^2 p^2} = \frac{2 \pi^8 \Delta_0[\mathbf{x}_1, \mathbf{x}_2]^{1-p} \Delta_0[\mathbf{x}_1, \mathbf{x}_3]^p \Delta_0[\mathbf{x}_2, \mathbf{x}_3]^p}{(-2 + p)^2 (-1 + p)^2 p^2}$$

where we used that  $\text{tr}(\mathbb{1}) = 2$  in  $D = 4$  (see appendix A).

## Acknowledgments

We thank Sergei Derkachov, Vladimir Kazakov and Enrico Olivucci for discussions and the critical reading of the draft. The work of MP is supported by the European Research Council (Programme "Ideas" ERC-2012-AdG 320769 AdS-CFT-solvable).

## A Notation and conventions

In this paper the metric tensor of the  $D$  dimensional Euclidean space is taken to be

$$g_{\mu\nu} = \delta_{\mu\nu} = \text{diag}(\underbrace{1, 1, \dots, 1, \dots, 1}_{D \text{ times}}) \quad (\text{A.1})$$

where  $\mu, \nu = 0, 1, \dots, D - 1$  are spacetime vector indices. We can introduce the  $D$  dimensional gamma matrices  $(\gamma^\mu)_{\alpha\beta}$  satisfying the Clifford algebra

$$\gamma^\mu \gamma^\nu + \gamma^\nu \gamma^\mu = 2\delta^{\mu\nu} \mathbb{1} \quad (\text{A.2})$$

The gamma matrices are hermitian and their size depends on the spacetime dimension: if  $D$  is even they are  $2^{D/2} \times 2^{D/2}$  matrices and if  $D$  is odd they are  $2^{(D-1)/2} \times 2^{(D-1)/2}$ . When the spin indices of the gamma matrices are not explicit we are considering the matrices contracted as  $\gamma^\mu \gamma^\nu = (\gamma^\mu)_{\alpha\beta} (\gamma^\nu)_{\beta\gamma}$ . The trace over the product of gamma matrices is dimension-dependent if  $D$  is odd, for instance the trace over an odd number of matrices is not-zero in general

$$\begin{aligned} \text{tr}(\gamma^\mu \gamma^\nu \gamma^\rho) &\neq 0 & \text{if } D = 1, 3 \\ \text{tr}(\gamma^\mu \gamma^\nu \gamma^\rho \gamma^\sigma \gamma^\lambda) &\neq 0 & \text{if } D = 1, 3, 5 \quad \text{etc.} \end{aligned} \quad (\text{A.3})$$

However when the spacetime dimension  $D$  is even, the traces are dimensional independent, for instance

$$\begin{aligned} \text{tr}(\text{odd number of } \gamma \text{'s}) &= 0 \\ \text{tr}(\gamma^\mu \gamma^\nu) &= \delta^{\mu\nu} \text{tr}(\mathbb{1}) \\ \text{tr}(\gamma^\mu \gamma^\nu \gamma^\rho \gamma^\sigma) &= (\delta^{\mu\sigma} \delta^{\nu\rho} - \delta^{\mu\rho} \delta^{\nu\sigma} + \delta^{\mu\nu} \delta^{\rho\sigma}) \text{tr}(\mathbb{1}) \quad \text{etc.} \end{aligned} \quad (\text{A.4})$$

and the  $D$ -dependence resides entirely in  $\text{tr}(\mathbb{1}) = 2^{D/2}$ .

The gamma matrices in even dimensions can be chosen to have the "off-block diagonal" form

$$\gamma^\mu = \begin{pmatrix} 0 & (\sigma^\mu)_{\alpha\dot{\beta}} \\ (\bar{\sigma}^\mu)^{\dot{\alpha}\beta} & 0 \end{pmatrix} \quad (\text{A.5})$$

by introducing the  $2^{D/2-1} \times 2^{D/2-1}$  Euclidean  $\sigma$  matrices

$$\sigma^\mu = (-i\vec{\sigma}, \mathbb{1}_{2^{D/2-1}}) \quad \text{and} \quad \bar{\sigma}^\mu = (i\vec{\sigma}, \mathbb{1}_{2^{D/2-1}}) \quad (\text{A.6})$$

where  $\vec{\sigma}$  is a  $D - 1$  dimensional vector of *Pauli matrices* and  $\mathbb{1}_{2^{D/2-1}}$  is the  $2^{D/2-1} \times 2^{D/2-1}$  identity matrix. Writing the identity matrix appearing in (A.2) as

$$\mathbb{1} = \begin{pmatrix} \delta_{\alpha}^{\beta} \equiv \mathbb{1}_{2^{D/2-1}} & 0 \\ 0 & \delta^{\dot{\alpha}}_{\dot{\beta}} \equiv \mathbb{1}_{2^{D/2-1}} \end{pmatrix} \quad (\text{A.7})$$

the  $\sigma$  matrices satisfy

$$\bar{\sigma}_\mu \sigma_\nu + \bar{\sigma}_\nu \sigma_\mu = 2\delta_{\mu\nu} \mathbb{1}_{2^{D/2-1}} \quad \text{and} \quad \sigma_\mu \bar{\sigma}_\nu + \sigma_\nu \bar{\sigma}_\mu = 2\delta_{\mu\nu} \mathbb{1}_{2^{D/2-1}} \quad (\text{A.8})$$

The traces identities are

$$\begin{aligned} \text{tr}(\text{odd number of } \sigma \text{'s}) &= 0 \\ \text{tr}(\sigma^\mu \bar{\sigma}^\nu) &= \text{tr}(\bar{\sigma}^\mu \sigma^\nu) = \delta^{\mu\nu} \text{tr}(\mathbb{1}_{2^{D/2-1}}) \\ \text{tr}(\sigma^\mu \bar{\sigma}^\nu \sigma^\rho \bar{\sigma}^\sigma) + \text{tr}(\bar{\sigma}^\mu \sigma^\nu \bar{\sigma}^\rho \sigma^\sigma) &= 2(\delta^{\mu\sigma} \delta^{\nu\rho} - \delta^{\mu\rho} \delta^{\nu\sigma} + \delta^{\mu\nu} \delta^{\rho\sigma}) \text{tr}(\mathbb{1}_{2^{D/2-1}}) \quad \text{etc.} \end{aligned} \quad (\text{A.9})$$

where  $\text{tr}(\mathbb{1}_{2^{D/2-1}}) = 2^{D/2-1}$ .

## References

- [1] A. N. Vasiliev, Yu. M. Pismak and Yu. R. Khonkonen, "Simple Method of Calculating the Critical Indices in the  $1/N$  Expansion", *Theor. Math. Phys.* **46**, 104 (1981), [Teor. Mat. Fiz.46,157(1981)].
- [2] F. V. Tkachov, "A Theorem on Analytical Calculability of Four Loop Renormalization Group Functions", *Phys. Lett.* **100B**, 65 (1981).
- [3] K. G. Chetyrkin and F. V. Tkachov, "Integration by Parts: The Algorithm to Calculate beta Functions in 4 Loops", *Nucl. Phys.* **B192**, 159 (1981).
- [4] A. V. Smirnov, "FIRE5: a C++ implementation of Feynman Integral REduction", *Comput. Phys. Commun.* **189**, 182 (2015), [arXiv:1408.2372](https://arxiv.org/abs/1408.2372).
- [5] S. Laporta, "High precision calculation of multiloop Feynman integrals by difference equations", *Int. J. Mod. Phys.* **A15**, 5087 (2000), [hep-ph/0102033](https://arxiv.org/abs/hep-ph/0102033).
- [6] K. G. Chetyrkin, A. L. Kataev and F. V. Tkachov, "New Approach to Evaluation of Multiloop Feynman Integrals: The Gegenbauer Polynomial  $x$  Space Technique", *Nucl. Phys.* **B174**, 345 (1980).
- [7] M. C. Bergere and Y.-M. P. Lam, "Asymptotic expansion of Feynman amplitudes. Part 1: The convergent case", *Commun. Math. Phys.* **39**, 1 (1974).
- [8] N. I. Usyukina, "On a Representation for Three Point Function", *Teor. Mat. Fiz.* **22**, 300 (1975).

- [9] A. Grozin, J. M. Henn, G. P. Korchemsky and P. Marquard, “*Three Loop Cusp Anomalous Dimension in QCD*”, *Phys. Rev. Lett.* **114**, 062006 (2015), [arXiv:1409.0023](#).
- [10] A. Grozin, J. M. Henn, G. P. Korchemsky and P. Marquard, “*The three-loop cusp anomalous dimension in QCD and its supersymmetric extensions*”, *JHEP* **1601**, 140 (2016), [arXiv:1510.07803](#).
- [11] M. S. Bianchi, L. Griguolo, A. Mauri, S. Penati, M. Preti and D. Seminara, “*Towards the exact Bremsstrahlung function of ABJM theory*”, *JHEP* **1708**, 022 (2017), [arXiv:1705.10780](#).
- [12] E. Remiddi, “*Differential equations for Feynman graph amplitudes*”, *Nuovo Cim. A110*, 1435 (1997), [hep-th/9711188](#).
- [13] V. A. Smirnov, “*Analytic tools for Feynman integrals*”, *Springer Tracts Mod. Phys.* **250**, 1 (2012).
- [14] M. D’Eramo, G. Parisi and L. Peliti, “*Theoretical predictions for critical exponents at the  $\lambda$  point of Bose liquids*”, *Lett. Nuovo Cim.* **2**, 878 (1971).
- [15] E. S. Fradkin and M. Ya. Palchik, “*Recent Developments in Conformal Invariant Quantum Field Theory*”, *Phys. Rept.* **44**, 249 (1978).
- [16] N. I. Usyukina, “*Calculation of many loop diagrams of perturbation theory*”, *Theor. Math. Phys.* **54**, 78 (1983), [*Teor. Mat. Fiz.*54,124(1983)].
- [17] D. I. Kazakov, “*The method of uniqueness, a new powerful technique for multiloop calculations*”, *Phys. Lett.* **133B**, 406 (1983).
- [18] D. I. Kazakov, “*Calculation of Feynman integrals by the method of uniqueness*”, *Theor. Math. Phys.* **58**, 223 (1984), [*Teor. Mat. Fiz.*58,343(1984)].
- [19] D. I. Kazakov, “*Multiloop Calculations: Method of Uniqueness and Functional Equations*”, *Theor. Math. Phys.* **62**, 84 (1985), [*Teor. Mat. Fiz.*62,127(1984)].
- [20] A. B. Zamolodchikov, “*‘Fishnet’ diagrams as a completely integrable system*”, *Phys. Lett.* **97B**, 63 (1980).
- [21] S. E. Derkachov, G. P. Korchemsky and A. N. Manashov, “*Noncompact Heisenberg spin magnets from high-energy QCD: 1. Baxter Q operator and separation of variables*”, *Nucl. Phys.* **B617**, 375 (2001), [hep-th/0107193](#).
- [22] L. N. Lipatov, “*High-energy asymptotics of multicolor QCD and two-dimensional conformal field theories*”, *Phys. Lett.* **B309**, 394 (1993), in: “*QCD and high-energy hadronic interactions. Proceedings, Hadronic Session of the 28th Rencontres de Moriond, Moriond Particle Physics Meeting, Les Arcs, France, March 20-27, 1993*”, 394-396p, [,147(1993)].
- [23] L. N. Lipatov, “*Asymptotic behavior of multicolor QCD at high energies in connection with exactly solvable spin models*”, *JETP Lett.* **59**, 596 (1994), [hep-th/9311037](#), [*Pisma Zh. Eksp. Teor. Fiz.*59,571(1994)].
- [24] K. Symanzik, “*On Calculations in conformal invariant field theories*”, *Lett. Nuovo Cim.* **3**, 734 (1972).
- [25] A. P. Isaev, “*Multiloop Feynman integrals and conformal quantum mechanics*”, *Nucl. Phys.* **B662**, 461 (2003), [hep-th/0303056](#).
- [26] D. Chicherin, S. Derkachov and A. P. Isaev, “*Conformal algebra: R-matrix and star-triangle relation*”, *JHEP* **1304**, 020 (2013), [arXiv:1206.4150](#).
- [27] V. V. Belokurov and N. I. Usyukina, “*Calculation of ladder diagrams in arbitrary order*”, *J. Phys.* **A16**, 2811 (1983).

- [28] M. Preti, “*WiLE: a Mathematica package for weak coupling expansion of Wilson loops in ABJ(M) theory*”, *Comput. Phys. Commun.* **227**, 126 (2018), [arXiv:1707.08108](#).
- [29] V. Kazakov, E. Olivucci and M. Preti, “*Four-point correlation functions of double-scaled  $N=4$  SYM*”, (to appear) , (2018).
- [30] S. Frolov, “*Lax pair for strings in Lunin-Maldacena background*”, *JHEP* **0505**, 069 (2005), [hep-th/0503201](#).
- [31] N. Beisert and R. Roiban, “*Beauty and the twist: The Bethe ansatz for twisted  $N=4$  SYM*”, *JHEP* **0508**, 039 (2005), [hep-th/0505187](#).
- [32] D. Correa, J. Henn, J. Maldacena and A. Sever, “*The cusp anomalous dimension at three loops and beyond*”, *JHEP* **1205**, 098 (2012), [arXiv:1203.1019](#).
- [33] M. Bonini, L. Griguolo, M. Preti and D. Seminara, “*Surprises from the resummation of ladders in the ABJ(M) cusp anomalous dimension*”, *JHEP* **1605**, 180 (2016), [arXiv:1603.00541](#).
- [34] M. de Leeuw, A. C. Ipsen, C. Kristjansen and M. Wilhelm, “*One-loop Wilson loops and the particle-interface potential in AdS/dCFT*”, *Phys. Lett.* **B768**, 192 (2017), [arXiv:1608.04754](#).
- [35] J. Aguilera-Damia, D. H. Correa and V. I. Giraldo-Rivera, “*Circular Wilson loops in defect Conformal Field Theory*”, *JHEP* **1703**, 023 (2017), [arXiv:1612.07991](#).
- [36] M. Preti, D. Trancanelli and E. Vescovi, “*Quark-antiquark potential in defect conformal field theory*”, *JHEP* **1710**, 079 (2017), [arXiv:1708.04884](#).
- [37] . Grdoan and V. Kazakov, “*New Integrable 4D Quantum Field Theories from Strongly Deformed Planar  $\mathcal{N} = 4$  Supersymmetric Yang-Mills Theory*”, *Phys. Rev. Lett.* **117**, 201602 (2016), [arXiv:1512.06704](#), [Addendum: *Phys. Rev. Lett.* **117**, no.25, 259903(2016)].