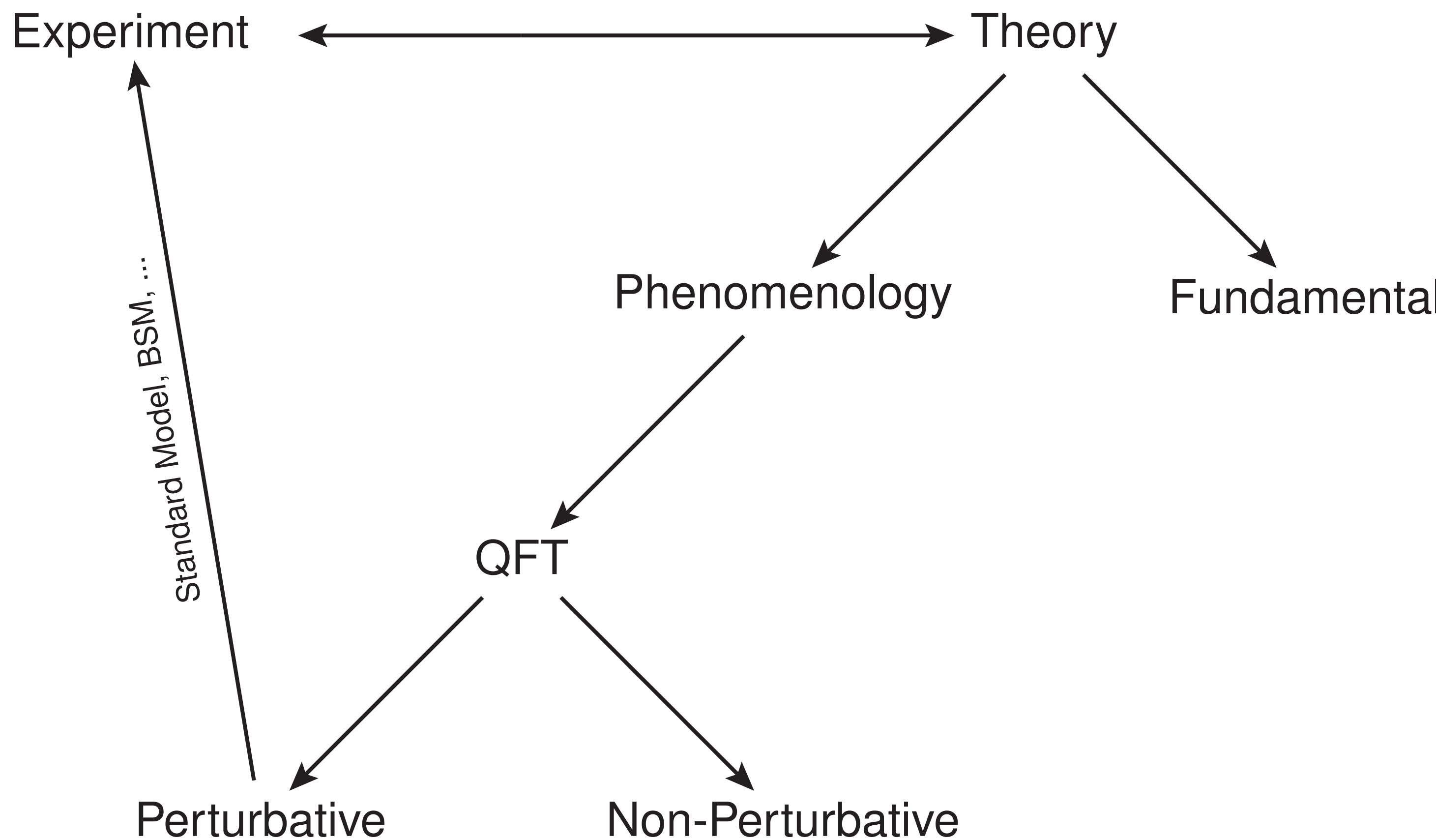
Automated Calculations

J.A.M. Vermaseren

Nikhef

With code by Ben Ruyl, Takahiro Ueda, J.V. and Andreas Vogt.


Experiment
Theory
Phenomenology
Fundamental
QFT
Perturbative
Non-Perturbative
Standard Model, BSM, ...

Calculations: signals, backgrounds.

General rule: the calculation of backgrounds is more than an order of magnitude more work than the computation of the signals.

PROBLEM: Notoriety only lies with the signals.

sion was on bread-and-butter physics from the collider experiments. It was the slow day of the week. Many of the physicists took the day off to go shopping in Milan or skiing at Cervino, on the slopes of the Matterhorn. They returned that night complaining that it had been wet, foggy, and even worse, flat.

Glashow drove down to Milan to do some shopping, and returned commenting that every time he passed a magazine stand, he saw Rubbia's face staring out from a prominently displayed copy of *L'Uomo Vogue*. He wondered why he hadn't made it to the cover of any fashion magazines when he won his Nobel.

Most of the UA1 gang returned to Geneva and to work.

THURSDAY

At three-thirty on Thursday afternoon Rubbia left the conference, on his way to Rome. An hour later, James Stirling and Steve Ellis took over the job that Altarelli had started.

Stirling was a dapper Englishman, in his mid-thirties, who had worked on QCD theory for years. Ellis was a bearded American of forty-one, who had come to CERN for one year "with the intention to get my hands dirty doing some phenomenology again . . . to work with the people who have the machine, because it's fun." When Ellis came to CERN, he and Stirling, together with a young Dutch theorist, started working on possible standard backgrounds to the UA1 missing-energy events. "I was just unhappy," Ellis said, "that I was hearing so much about supersymmetry and so little about what they were actually doing at UA1."

They had started working in January with the 1983 data from UA1 and UA2; both experiments had published events in which they had seen W's or Z's created accompanied by gluon jets. With these events, Ellis and Stirling had deduced what possible circumstances could occur to hide the W's and Z's when they were created, leaving only the jets visible, which would appear in the detectors as monojets. It was much like the work Denegri had been doing sporadically at UA1. They had worked through the laborious calculations of QCD that specified the frequency with which those circumstances would be expected to occur. They were doing nothing particularly original. They were simply doing more conclusively what other physicists had done in bits and pieces. By early February, they had results for the possible background to the monojets that were significantly larger than those UA1 had published. It was then that they decided they would discuss their work at St. Vincent.

Stirling spoke first, abo
gluon and quark jets—in V
tion with the monojets. He
calmly, letting the calculati
case step by step. For eve
would check back with tl
deduction had been valid.
would say, or, "This gives ı
ing that their theoretical ca
accurately matched the ava
UA2.

"QCD," he concluded, "ı
serious, or not serious at a
is just a background to mo

Then Ellis took the stag
and half television sports a
castic: "I want to send a w
boring physics might accou
cal and maybe unpatriotic
anyway." And he proceede

The subtitle of his talk w
first transparency read: "Re
sum of many small things is
what he called his roads, or
ingredients of the Altarelli C
lations. He first calculated h
from Z-zero's invisibly decay
a gluon jet thrown in. This
the missing-energy events. T
to expect from this in their
is no appreciable contributi
Ellis, on the other hand, cla
two of the UA1 events. ("Thi
could be as great as four.")

He then proceeded with
"monojets" that could be ex
tarelli Cocktail. They were a
an event there. But, as Ellis sa
them up—all the possible ba
had reported as inexplicable
that could look like new phy

The run ended on December 7. Until then, the UA1 physicists worked their insane hours and nursed the machine and the central detector and tried to figure out how to prove that what they had were W's. By the time the collider shut down for the year, UA1 had maybe five events that might be W's, although they could not prove them. To make sure their physicists left the offices for the Christmas vacation and maybe even relaxed, the management not only turned off the heat, they turned off the computers. They knew that many physicists would gladly freeze to death if they thought they could get prime computer time.

By the end of December, the UA1 physicists had concocted a better way to prove whether an event was really a W or just looked like one. In Paris over the vacation the Saclay physicists, particularly Michel Spiro, the spokesman, and Denegri, had finally figured out the details of how to prove the existence of the neutrino in the W decay. They created a program that would add up the energy deposited throughout the nearly hermetically sealed detector and calculate if there was an imbalance. Simple Newtonian physics dictates that every action must have an equal and opposite reaction; hence, the energy deposited from the collision on one side of the detector had to balance with that on the other. If it didn't, it meant something had escaped detection, and the only thing that could escape would be a neutrino. If the missing energy, as it was called—the energy of the neutrino—when added to the energy of the electron in the collision, equaled the expected mass of the W, then the probability that the event was a W became overwhelmingly great.

The detector had been designed to completely surround the collision point for just this reason, to prove the existence of the neutrino by proving that energy was missing. Now they really could prove that they had W's. Real W's that even Richter would believe in. They had only five of them. But they sure as hell appeared to be real.

Rubbia had spent Christmas with his family, "looking at the Pyramids and sailing the Nile." It was his longest vacation in a decade, and the first time he had spent so much time with his family in probably twice that long. He called it decompressing. Before he left, Denegri had called him with the news that they could prove the existence of the W's.

Rubbia's only concern was whether they could make a convincing argument to the physics community. Whether he could overcome his track record. People had not forgotten the Alternating

Currents and the high-
mind him, it was the
Christmas play. This ti
Carlo, bursts onto the
team of experimentors
begin to claim that the
them.

"Shutta up," he scr
de premature rumore!'

The experimentors
on to announce that
announcement, he pic
a W on it, and then a
He puts the two card
discovered the W. He
when placed against
"Wiza more sophi
Carlo continues, "we

Then he flips the
holds it over his head
"What is thisa Y d
an anomaly!"

Rubbia saw the pla
Kate Morgan, of UA1.
I'm finished. He slump
after you could not m
front of Rubbia. It hu

On January 12, 1983, R
on the results of the
which had been orga
collider. It was a sho

At UA1, the week
The analysis went on
the ever-present fear t
that they had no way
out and say that the
wrong in assuming th
impact that Rubbia wa
right?

Even Sadoulet wa

Solution: Let the experimentalists calculate the backgrounds themselves. Hence automated programs like GRACE, CompHEP, Madgraph, FeynCalc/FormCalc, Pythia, GoSam, ...

To create such a system, FORM was started in 1984 as the beginning of such a system (ESP project). It is actually used in many of the automated systems.

FORM is however more useful and personally, after it came out in 1989, I got 'sidetracked' into 3-loop QCD. Hence today we are not going to concentrate on 1-loop automation but more on what came out of that.

Computation of many loops is not always for direct confrontation with experiments, but often just for setting the stage for other calculations. Examples:

- Beta functions.
- Anomalous dimensions/Splitting functions. Leads to PDF's.

Of course there are also more generic quantities that are much closer to the experiments like

- Sum rules.
- Total crosssections.
- Structure functions.
- g-2.

What are the tools one has for such multi-loop calculations?

The main workhorse these days is IBP: integration by parts (Chetyrkin and Tkachov).

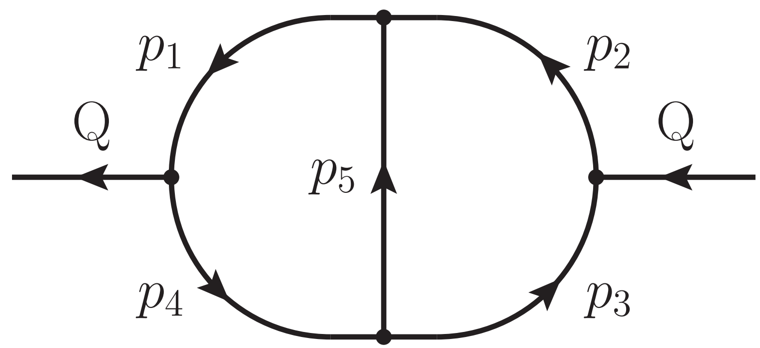
What does that mean in practise?

Each loop in a diagram corresponds to an integral $\int d^D p$ where p is the loop momentum and we integrate over $D = 4 - 2\epsilon$ dimensions. For such integrals holds that

$$\int d^D p \ \frac{d}{dp^\mu} \left(q^\mu I\right) \ = \ 0$$

in which I is a typical integrant.

Let us see what that does to a typical Feynman diagram.



This diagram represents the integral

$$I(n_1, n_2, n_3, n_4, n_5) \;=\; \int d^D p_1 \; d^D p_2 \frac{1}{(p_1^2)^{n_1} (p_2^2)^{n_2} (p_3^2)^{n_3} (p_4^2)^{n_4} (p_5^2)^{n_5}}$$

Let us assume that instead of over p_1 we integrate over $P = p_5$. IBP gives:

$$0 \;=\; \int d^D P \frac{d}{dP^\mu}(P^\mu J(n_1,n_2,n_3,n_4,n_5))$$

in which J is the integrant in the above formula.

Working out the derivative and using momentum conservation as in $2P.p_1 = P^2 + p_1^2 - p_2^2$ we obtain:

$$\begin{aligned} 0 \;=\; & (D - 2n_5 - n_1 - n_4) I(n_1,n_2,n_3,n_4,n_5) \\ - \; & n_1(I(n_1+1,n_2,n_3,n_4,n_5-1) - I(n_1+1,n_2-1,n_3,n_4,n_5)) \\ - \; & n_4(I(n_1,n_2,n_3,n_4+1,n_5-1) - I(n_1,n_2,n_3-1,n_4+1,n_5)) \end{aligned}$$

What we see is that our original integral is equal to 4 other integrals that have different powers of the denominators. AND it are always n_1 and n_4 that are raised while one of the others is lowered.

HENCE: if n_2, n_3 and n_5 are integers, repeated application will make one of them zero. This would be a simpler diagram which we know how to integrate.

This is the core of the IBP method.

Before we continue, let us introduce a new notation:

$$\begin{aligned}
\mathbf{1} &= I(n_1, n_2, n_3, n_4, n_5) \\
\mathbf{1}^+ &= I(n_1 + 1, n_2, n_3, n_4, n_5) \\
\mathbf{4}^+ &= I(n_1, n_2, n_3, n_4 + 1, n_5) \\
\mathbf{2}^- &= I(n_1, n_2 - 1, n_3, n_4, n_5)
\end{aligned}$$

etc. which gives

$$0 = (D - 2n_5 - n_1 - n_4)\mathbf{1} - n_1\mathbf{1}^+\mathbf{5}^- + n_1\mathbf{1}^+\mathbf{2}^- - n_4\mathbf{4}^+\mathbf{5}^- + n_4\mathbf{4}^+\mathbf{3}^-$$

Now we are ready for the next step: let us assume that n_5 is not an integer. In that case repeated application does not terminate.

The solution is to write all possible IBP identities. For the derivative we have two choices: p_1 and p_2 and for the extra momentum inside the derivative we have three choices: p_1, p_2 and Q. This means that we can construct 6 equations.

We have done this by computer and obtain:

$$\begin{aligned}
0 &= +n_5\mathbf{4}^-\mathbf{5}^+ - n_5\mathbf{3}^-\mathbf{5}^+ + n_5\mathbf{2}^-\mathbf{5}^+ - n_5\mathbf{1}^-\mathbf{5}^+ \\
&\quad +n_4\mathbf{4}^+ - n_4\mathbf{1}^-\mathbf{4}^+ - n_1\mathbf{1}^+ + n_1\mathbf{1}^+\mathbf{4}^- + (-n_1+n_4)\mathbf{1} \qquad (1)\\
0 &= +n_5\mathbf{2}^-\mathbf{5}^+ - n_5\mathbf{1}^-\mathbf{5}^+ + n_4\mathbf{4}^+ - n_4\mathbf{1}^-\mathbf{4}^+ + (-2\epsilon - 2n_1 - n_4 - n_5 + 4)\mathbf{1} \qquad (2)\\
0 &= +n_5\mathbf{2}^-\mathbf{5}^+ - n_5\mathbf{1}^-\mathbf{5}^+ + n_4\mathbf{4}^+ + n_4\mathbf{4}^+\mathbf{5}^- \\
&\quad -n_4\mathbf{3}^-\mathbf{4}^+ - n_4\mathbf{1}^-\mathbf{4}^+ + n_1\mathbf{1}^+\mathbf{5}^- - n_1\mathbf{1}^+\mathbf{2}^- + (-n_1+n_5)\mathbf{1} \qquad (3)\\
0 &= -n_5\mathbf{4}^-\mathbf{5}^+ + n_5\mathbf{3}^-\mathbf{5}^+ - n_5\mathbf{2}^-\mathbf{5}^+ + n_5\mathbf{1}^-\mathbf{5}^+ \\
&\quad +n_3\mathbf{3}^+ - n_3\mathbf{2}^-\mathbf{3}^+ - n_2\mathbf{2}^+ + n_2\mathbf{2}^+\mathbf{3}^- + (-n_2+n_3)\mathbf{1} \qquad (4)\\
0 &= -n_5\mathbf{2}^-\mathbf{5}^+ + n_5\mathbf{1}^-\mathbf{5}^+ + n_3\mathbf{3}^+ + n_3\mathbf{3}^+\mathbf{5}^- \\
&\quad -n_3\mathbf{3}^+\mathbf{4}^- - n_3\mathbf{2}^-\mathbf{3}^+ + n_2\mathbf{2}^+\mathbf{5}^- - n_2\mathbf{1}^-\mathbf{2}^+ + (-n_2+n_5)\mathbf{1} \qquad (5)\\
0 &= -n_5\mathbf{2}^-\mathbf{5}^+ + n_5\mathbf{1}^-\mathbf{5}^+ + n_3\mathbf{3}^+ - n_3\mathbf{2}^-\mathbf{3}^+ + (-2\epsilon - 2n_2 - n_3 - n_5 + 4)\mathbf{1} \qquad (6)
\end{aligned}$$

and in case you are wondering where the original equation went, it is obtained by subtracting the third equation from the second. Similarly we can subtract the fifth equation from the sixth:

$$0 = -n_4\mathbf{4}^+\mathbf{5}^- + n_4\mathbf{3}^-\mathbf{4}^+ - n_1\mathbf{1}^+\mathbf{5}^- + n_1\mathbf{1}^+\mathbf{2}^- + (-2\epsilon - n_1 - n_4 - 2n_5 + 4)\mathbf{1} \tag{7}$$

$$0 = -n_3\mathbf{3}^+\mathbf{5}^- + n_3\mathbf{3}^+\mathbf{4}^- - n_2\mathbf{2}^+\mathbf{5}^- + n_2\mathbf{1}^-\mathbf{2}^+ + (-2\epsilon - n_2 - n_3 - 2n_5 + 4)\mathbf{1} \tag{8}$$

Let us now take the second equation and rewrite it a bit:

$$n_4 \mathbf{4}^+ \;=\; -n_5 \mathbf{2}^- \mathbf{5}^+ + n_5 \mathbf{1}^- \mathbf{5}^+ + n_4 \mathbf{1}^- \mathbf{4}^+ + (2\epsilon + 2n_1 + n_4 + n_5 - 4)\mathbf{1}$$

or

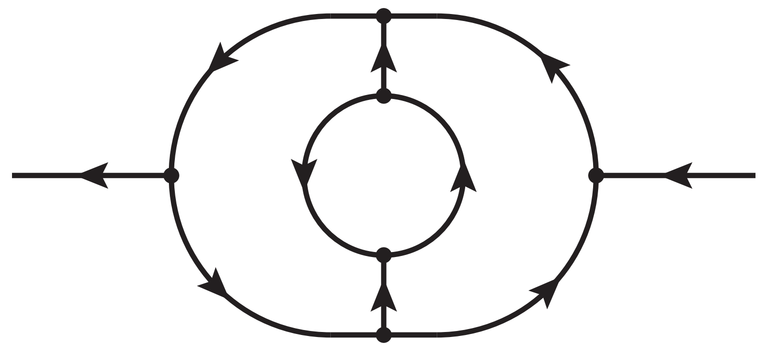
$$\mathbf{1} \;=\; \frac{1}{n_4 - 1}(-n_5 \mathbf{2}^- \mathbf{5}^+ \mathbf{4}^- + n_5 \mathbf{1}^- \mathbf{5}^+ \mathbf{4}^- + (n_4 - 1)\mathbf{1}^- + (2\epsilon + 2n_1 + n_4 + n_5 - 5)\mathbf{4}^-)$$

This equation can be used to either lower n_4 or n_1. The only problem is that once n_4 is equal to one the denominator in the RHS becomes zero and the equation does not work. However, by symmetry this equation can be used to bring n_1, n_2, n_3, n_4 down to one successively. After that, combining equations and shifting n_5 a few times we obtain the final equation:

$$\begin{aligned} I(1,1,1,1,1+n_5)(2\epsilon + 2n_5) \;=\; & I(1,1,1,1,n_5)(4\epsilon + 2n_5 - 2) \\ & + I(0,1,1,1,1+n_5)n_5 + I(0,1,1,2,n_5) \\ & - I(1,0,1,1,1+n_5)n_5 - I(1,1,0,1,1+n_5)n_5 \\ & - I(1,1,0,2,1+n_5) + I(1,1,1,0,1+n_5)n_5 \\ & - I(2,0,1,1,1+n_5) + I(2,1,1,0,n_5) \end{aligned}$$

Note that all terms in the RHS, except the first, are missing one line. This equation can be used to shift the value of n_5 by one. Hence we can express all our integrals in terms of integrals that miss a line and the single integral $I(1,1,1,1,1+x)$ in which $0 < x < 1$.

How do we obtain such non-integer powers?. It could have been that we had a three loop diagram like



Integrating the internal loop over D dimensions gives a power with ϵ in it. Hence if one does a 4-loop computation, eventually there will be 3-loop diagrams in which one line is non-integer, and 2-loop diagrams with up to two lines with a non-integer power.

Integrals that can not be reduced further and have to be computed by different means are called master integrals. To compute those is a science by itself. Here we will assume that somebody (Chetyrkin, Lee+Smirnov+Smirnov) has already computed the ones we need and concentrate on the reduction to the master integrals.

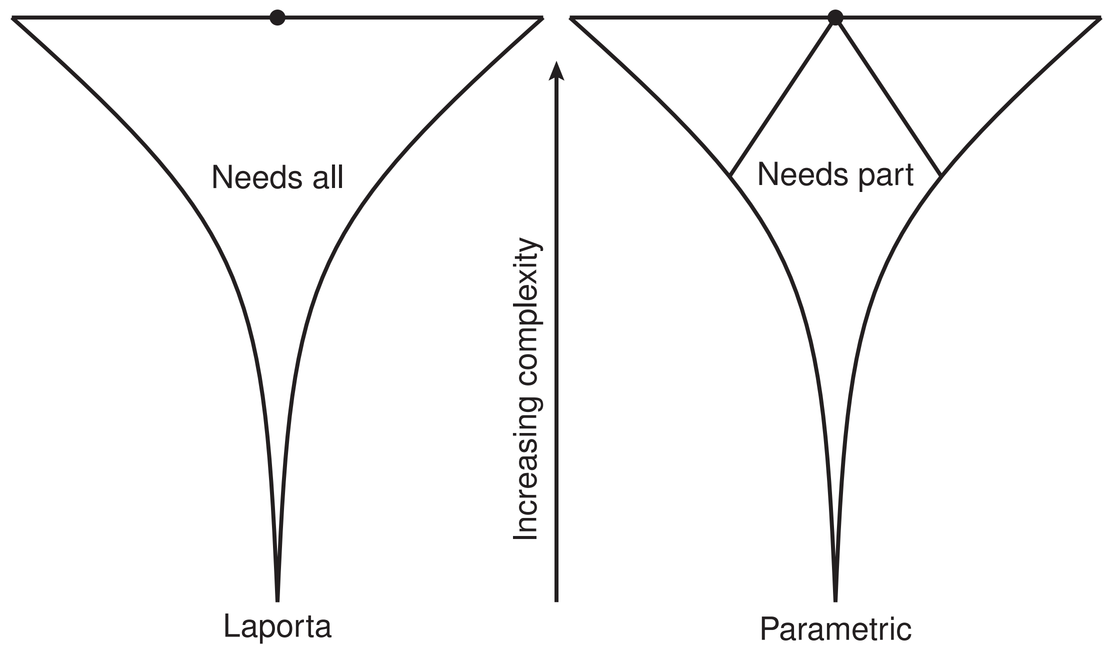
Currently there are two approaches.

- The Laporta method.
- The parametric method.

The equations we saw above follow the parametric method. This means that we have symbolic equations to successively reduce parameters. Its advantage is that one goes more or less directly towards simpler integrals. The disadvantage is that the derivation of the useful equations can take much time. Each topology has to be treated separately and may take much effort.

In the Laporta method we start with the basic equations and substitute a one for all variables. In the above example that would give 6 equations. If we define the complexity of a parameter the absolute value of the amount that it deviates from this start value, the complexity of the integral is the sum of all those. Hence we start with equations of complexity zero.

Next we use Gaussian elimination to eliminate the terms with the highest complexity. And we remember the identities by which we eliminated them. After this we generate more equations by taking the original equations and raising the complexity of one of the variables. We can do this for all 5 of the variables and this leads to 30 new equations, but there are also integrals of higher complexity. Again we eliminate the integrals with the highest complexity first and then there are some equations left with only integrals of a lower complexity that complement our earlier integrals. If we keep doing this (creating more and more equations with higher and higher complexity) eventually we will have solutions of all integrals in our problem, expressing them in terms of a limited set of master integrals.
There are several programs that can do this automatically. The best known are Reduze and Fire. The advantage is that they are very general once the basic set of equations is known. The enormous disadvantage is that they only work well when all intermediate results can be stored. The consequences are that they cannot go to very large complexities.



It should be clear that if a reduction can be used for many computations or need very large complexities, it is worth making such a reduction program, while for computations that do not recur often and have only limited complexities, the Laporta method is superior.

At Nikhef we have been involved mostly developing the parametric method. This was needed for the computation of higher Mellin moments in Deep Inelastic Scattering (DIS). This led at one time to integrals of complexity 59. (3-loop integrals, 9 variables). With the parametric method they could be solved by brute force. For this we have the Mincer program for 3-loop massless propagators. It is a FORM program that is very much optimized.

```
             Thread 23 reporting
Time =   74215.89 sec    Generated terms =   2904708739
           d372c         Terms in thread =    768150841
             Harmonics,0 Bytes used      = 333465128756

Time =   19414.59 sec    Generated terms =  70440435004
           d372c         Terms in output =  16284532632
             Harmonics,0 Bytes used      =4749134840748
                    .
                    .
Time =  122344.47 sec    Generated terms =            8
           d372c         Terms in output =            8
                         Bytes used      =          508

   d372c =
```

```
    + 9612381231427083802230061617904462541014700948430710402030790208768834\
     053372/28294434062104394976090603024843543192464092159681426876464843\
      75 - 592616977488389683359511458824192/171150573951955826353125*ep^-3
       - 40797021443232155915564864542619346937692148766672/1389457319508335\
      28152778706734358359375*ep^-2 + 3977935148744488205016978145931735245\
      852266243198715120262848/209002680761480801012688382080528332004845l0\
      3515625*ep^-1 - 4644414393484969555603045635325952/
      43792813525484570171625*ep^-1*z3 - 346708546516013809664/31121455365*
      z5 - 23222071967424847778015228176662976/14597604508494856723875*z4
    - 9520732613532305198061305875037953452048269312/
      39523986651792953877180696983525625*z3;

122344.47 sec + 10251515.20 sec: 10373859.67 sec out of 924983.33 sec
```

The Forcer project.

After three comes four. As part of the ERC project HEPGAME we developed a four-loop program for the evaluation of four-loop massless propagator diagrams. This had to be seen as an automation project: as much as possible should be automated, so that in the end, changing 4 to 5 should hopefully create most of the parts needed for a 5-loop version. As such it contains a number of elements of AI. In the future we hope to take this even further.

What goes in it?

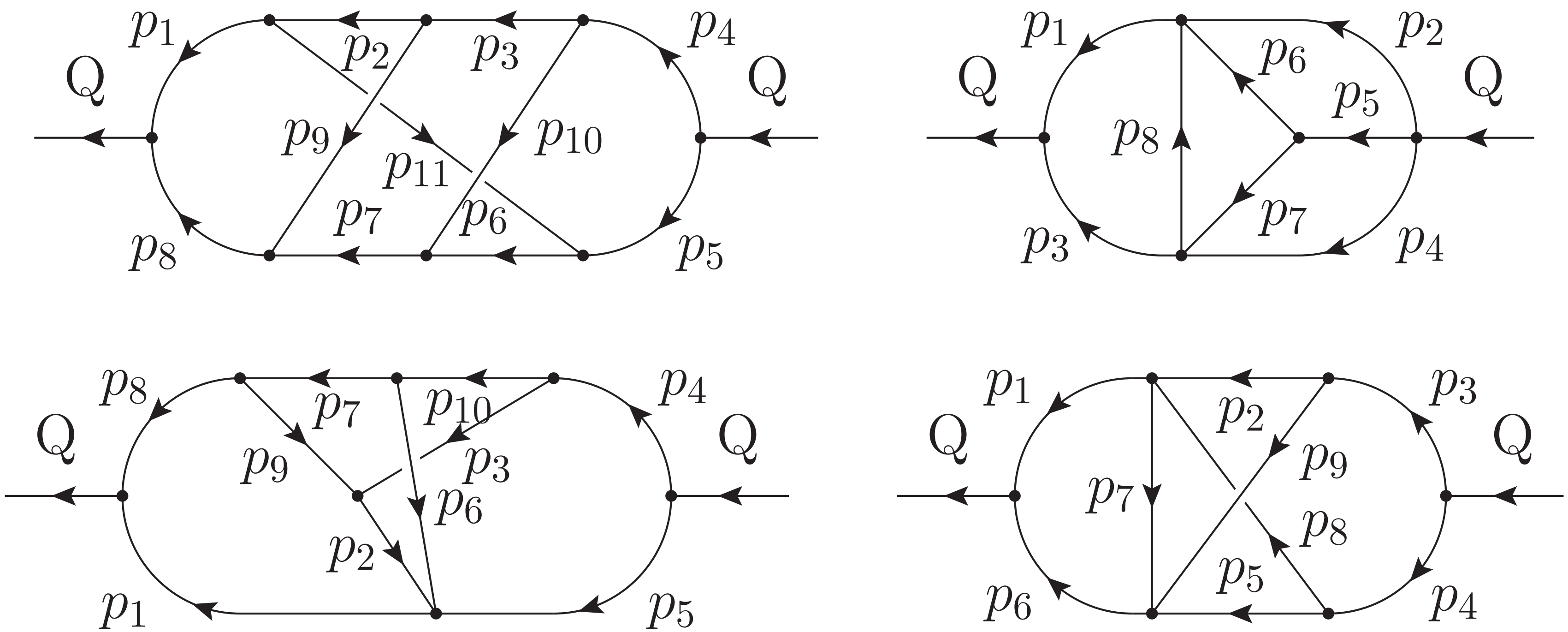
We need a diagram generator. Fortunately this problem has been solved many years ago. We use QGRAF, although others exist. It can give FORM-friendly output. It is kind of a wild horse though, because the author has obfuscated the code and has particular ideas about how the files should be dealt with. But eventually all worked well.

Currently an effort is under way to switch to the GRACE diagram generator written by Kaneko-san. It is much faster and this project would give good access to detailed information about the diagrams.

Next each diagram needs to get a notation. For this first its topology should be recognized and it should get the notation for that topology. In the past that was pretty messy code (for the Mincer programs), but FORM has gotten many new and original capabilities and in the end it was relatively simple to do this. For 5 loops it might be slow though because the time it needs is at least proportional to the number of diagrams times the number of topologies. On the other hand, this program needs to run only once for each process under consideration. The whole convdia.frm program has currently fewer than 1000 lines of which 250 lines of commentary.

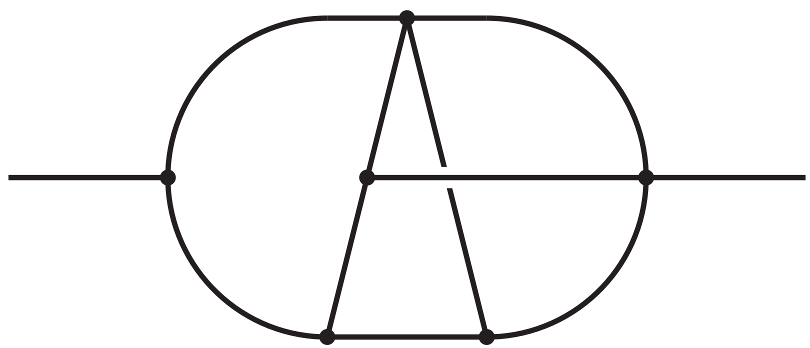
We need a reduction scheme for each topology. There may be hundreds or thousands of topologies. In the case of Mincer each topology could be done ‘by hand’. Here this is worse than impractical. This needs to be automated.

Some of the ’fun’ topologies:



Next, each time a line is removed, we obtain a simpler topology, which needs to be treated as well. But each topology has its own ‘perfect’ notation. Hence we need a rewriting of the variables, called rewiring. This is where traditionally most errors are made. It definitely needs to be automated. And is has been.

For Forcer there are 17 topologies that still need special attention. 16 of those concern master integrals. All other topologies can be treated either by doing one of the integrals because we know how to deal with it, or by repeated use of a single identity. The master topologies need a scheme in which the variables are reduced one by one. For 4-loop diagrams there are 14 variables to be treated this way. Currently we had to derive these schemes by hand guided computer programs and the process took typically several days to weeks for each. Some schemes take more than 1000 lines of FORM code. The worst however was the one topology that did not have a master integral. We call it bubu and it looks like



After quite a few weeks we did get a scheme that works and is not outrageously slow. This did take the invention of extra capabilities in FORM.

We hope to automate the derivation of these reductions as well. Takahiro Ueda is currently working very hard on it. In principle it is possible to give an algorithm, but there are some side conditions. One is efficiency. One would like to live to see the answers. Another side condition is that the whole derivation and the later computations should fit inside the computer and the FORM capabilities.

Next one needs a program that manages the computation of all those diagrams. For this we have the minos program. It is a kind of database that was designed in the early 90's for just this type of work. It works a bit like 'make' in running the diagrams one by one, just like the compilation of source codes with many source files. At the same time it maintains databases of the diagrams and the results of the calculations. More recently it has been reprogrammed into the new language "Rust". The hope is that also a new version of Form can be created in this language.

To set up the topologies, the notations, the rewirings, the sequencing of the procedure callings, the reductions, etc. Ben and Takahiro developed a Python program pathforcer.py which sets up the complete infrastructure and derives all the necessary procedures (currently with exception of the special topologies mentioned above). This is, at the moment of this talk, about 2700 lines of Python code. The forcer.h library file contains the reductions of the special topologies (21000 lines), a number of service routines and all necessary declarations(about 1000 lines). This file is almost 2 Megabytes, The generated pathforcer.h file is almost 3 Megabytes. In addition there are three more files with libraries of which one for treating color factors. Together they are about 8000 lines. This whole system took three people about a year to contruct and is by now very reliable.

Together with optimizations for Higher Mellin moments of splitting functions, the whole code is now more than 200000 lines. Mostly machine generated of course.

Where does this leave us?

With a few little shell scripts we can do the full 3-loop beta function for semi-simple Lie groups (QCD is just a special case in which the group is SU(3)) here in the talk on my laptop. And this with a check using a gauge parameter, which has to drop out in the final results.

Programs.

The final program to compute the beta function was made by Andreas Vogt.

Of course, in this setup one can also run the 4-loop beta function, but a demo would be either a bit too slow or have to be run on some big remote computers, with, according to Murphy will not work during a talk. The time on a decent computer with 24 cores: 7 hours for all powers of the gauge parameter, 6^m12^s with one power of the gauge parameter and 1^m07^s without gauge parameter.

The 5-loop beta function, without gauge parameter, needed more sophistication. First of all: we do not have a 5-loop program like Forcer. On the other hand: for the beta function we need only the UV divergent parts of the diagrams. With the use of some theorems, one can rework the diagrams such that one integral can be done trivially and the remaining part can be done by Forcer. This goes however at the cost of introducing IR divergences. Hence one needs a program that can take out the various divergences in a controled way. In our case two of us constructed a program that could do this on an integral-by-integral basis (the Rstar program, made by Franz Herzog and Ben Ruijl). As a result a few days running (without gauge parameter) on a whole battery of computers gave the result.

It should be said that different groups used completely different methods to separate out the various divergences.

History:

- Baikov, Chetyrkin and Kühn: April 2016 (only for SU(3))
- Herzog, Ruijl, Ueda, JV and Vogt: Jan 2017 (Semi simple lie groups)
- Chetyrkin, Falcioni, Herzog and JV: Sept 2017 (S-S groups,Incl gauge check)
- Luthe, Maier, Marquard and Schröder: Sept 2017

All papers use a different method.

The Chetyrkin method (paper 3) uses a global IR subtraction method. Is fastest in CPU time (with forcer) but most complicated conceptually.

The last method is simplest conceptually, but the master integrals are much harder.

The first paper uses the Baikov method to do 4-loop integrals and is by far the most time consuming (quite a few years of CPU time), even though it uses a simplified version of the Chetyrkin method. The third paper uses Forcer and is fastest.

The second paper uses Forcer and the Rstar program. This shows most potential for more calculations (some have been done already). Used of the order of one week on a computer with 32 cores, and worked for general gauge group.

Independently, all methods gave the same answer!

$$
\begin{aligned}
\beta_0 &= \frac{11}{3}\, C_A \;-\; \frac{4}{3}\, T_F\, n_f \;\;, \\
\beta_1 &= \frac{34}{3}\, C_A^2 \;-\; \frac{20}{3}\, C_A\, T_F\, n_f - 4\, C_F\, T_F\, n_f \;\;, \\
\beta_2 &= \frac{2857}{54}\, C_A^3 \;-\; \frac{1415}{27}\, C_A^2\, T_F\, n_f \;-\; \frac{205}{9}\, C_F\, C_A\, T_F\, n_f \;+\; 2\, C_F^2\, T_F\, n_f \\
&\quad + \; \frac{44}{9}\, C_F\, T_F^2\, n_f^2 \;+\; \frac{158}{27}\, C_A\, T_F^2\, n_f^2 \;\;,
\end{aligned}
$$

$$
\begin{aligned}
\beta_3 = {} & C_A^4 \left(\frac{150653}{486} - \frac{44}{9}\zeta_3\right) + \frac{d_A^{abcd} d_A^{abcd}}{N_A} \left(-\frac{80}{9} + \frac{704}{3}\zeta_3\right) \\
& + C_A^3 T_F n_f \left(-\frac{39143}{81} + \frac{136}{3}\zeta_3\right) + C_A^2 C_F T_F n_f \left(\frac{7073}{243} - \frac{656}{9}\zeta_3\right) \\
& + C_A C_F^2 T_F n_f \left(-\frac{4204}{27} + \frac{352}{9}\zeta_3\right) + \frac{d_F^{abcd} d_A^{abcd}}{N_A} n_f \left(\frac{512}{9} - \frac{1664}{3}\zeta_3\right) \\
& + 46\, C_F^3 T_F n_f + C_A^2 T_F^2 n_f^2 \left(\frac{7930}{81} + \frac{224}{9}\zeta_3\right) + C_F^2 T_F^2 n_f^2 \left(\frac{1352}{27} - \frac{704}{9}\zeta_3\right) \\
& + C_A C_F T_F^2 n_f^2 \left(\frac{17152}{243} + \frac{448}{9}\zeta_3\right) + \frac{d_F^{abcd} d_F^{abcd}}{N_A} n_f^2 \left(-\frac{704}{9} + \frac{512}{3}\zeta_3\right) \\
& + \frac{424}{243} C_A T_F^3 n_f^3 + \frac{1232}{243} C_F T_F^3 n_f^3 \quad ,
\end{aligned}
$$

$$
\begin{aligned}
\beta_4 = C_A^5 & \left(\frac{8296235}{3888} - \frac{1630}{81}\,\zeta_3 + \frac{121}{6}\,\zeta_4 - \frac{1045}{9}\,\zeta_5\right) \\
& + \frac{d_A^{abcd} d_A^{abcd}}{N_A}\, C_A \left(-\frac{514}{3} + \frac{18716}{3}\,\zeta_3 - 968\,\zeta_4 - \frac{15400}{3}\,\zeta_5\right) \\
& + C_A^4\, T_F\, n_f \left(-\frac{5048959}{972} + \frac{10505}{81}\,\zeta_3 - \frac{583}{3}\,\zeta_4 + 1230\,\zeta_5\right) \\
& + C_A^3\, C_F\, T_F\, n_f \left(\frac{8141995}{1944} + 146\,\zeta_3 + \frac{902}{3}\,\zeta_4 - \frac{8720}{3}\,\zeta_5\right) \\
& + C_A^2\, C_F^2\, T_F\, n_f \left(-\frac{548732}{81} - \frac{50581}{27}\,\zeta_3 - \frac{484}{3}\,\zeta_4 + \frac{12820}{3}\,\zeta_5\right) \\
& + C_A\, C_F^3\, T_F\, n_f \left(3717 + \frac{5696}{3}\,\zeta_3 - \frac{7480}{3}\,\zeta_5\right) - C_F^4\, T_F\, n_f \left(\frac{4157}{6} + 128\,\zeta_3\right) \\
& + \frac{d_A^{abcd} d_A^{abcd}}{N_A}\, T_F\, n_f \left(\frac{904}{9} - \frac{20752}{9}\,\zeta_3 + 352\,\zeta_4 + \frac{4000}{9}\,\zeta_5\right) \\
& + \frac{d_F^{abcd} d_A^{abcd}}{N_A}\, C_A\, n_f \left(\frac{11312}{9} - \frac{127736}{9}\,\zeta_3 + 2288\,\zeta_4 + \frac{67520}{9}\,\zeta_5\right)
\end{aligned}
$$

$$
\begin{aligned}
&+\ \frac{d_F^{abcd}d_A^{abcd}}{N_A}\,C_F\,n_f\ \left(-320+\frac{1280}{3}\,\zeta_3+\frac{6400}{3}\,\zeta_5\right)\\
&+\ C_A^3\,T_F^2\,n_f^2\ \left(\frac{843067}{486}+\frac{18446}{27}\,\zeta_3-\frac{104}{3}\,\zeta_4-\frac{2200}{3}\,\zeta_5\right)\\
&+\ C_A^2\,C_A\,T_F^2\,n_f^2\ \left(\frac{5701}{162}+\frac{26452}{27}\,\zeta_3-\frac{944}{3}\,\zeta_4+\frac{1600}{3}\,\zeta_5\right)\\
&+\ C_F^2\,C_A\,T_F^2\,n_f^2\ \left(\frac{31583}{18}-\frac{28628}{27}\,\zeta_3+\frac{1144}{3}\,\zeta_4-\frac{4400}{3}\,\zeta_5\right)\\
&+\ C_F^3\,T_F^2\,n_f^2\ \left(-\frac{5018}{9}-\frac{2144}{3}\,\zeta_3+\frac{4640}{3}\,\zeta_5\right)\\
&+\ \frac{d_F^{abcd}d_A^{abcd}}{N_A}\,T_F\,n_f^2\ \left(-\frac{3680}{9}+\frac{40160}{9}\,\zeta_3-832\,\zeta_4-\frac{1280}{9}\,\zeta_5\right)\\
&+\ \frac{d_F^{abcd}d_F^{abcd}}{N_A}\,C_A\,n_f^2\ \left(-\frac{7184}{3}+\frac{40336}{9}\,\zeta_3-704\,\zeta_4+\frac{2240}{9}\,\zeta_5\right)\\
&+\ \frac{d_F^{abcd}d_F^{abcd}}{N_A}\,C_F\,n_f^2\ \left(\frac{4160}{3}+\frac{5120}{3}\,\zeta_3-\frac{12800}{3}\,\zeta_5\right)
\end{aligned}
$$

$$
\begin{aligned}
&+\; C_A^2\, T_F^3\, n_f^3\; \left(-\frac{2077}{27} - \frac{9736}{81}\,\zeta_3 + \frac{112}{3}\,\zeta_4 + \frac{320}{9}\,\zeta_5\right) \\
&+\; C_A\, C_F\, T_F^3\, n_f^3\; \left(-\frac{736}{81} - \frac{5680}{27}\,\zeta_3 + \frac{224}{3}\,\zeta_4\right) \\
&+\; C_F^2\, T_F^3\, n_f^3\; \left(-\frac{9922}{81} + \frac{7616}{27}\,\zeta_3 - \frac{352}{3}\,\zeta_4\right) \\
&+\; \frac{d_F^{abcd} d_F^{abcd}}{N_A}\, T_F\, n_f^3\; \left(\frac{3520}{9} - \frac{2624}{3}\,\zeta_3 + 256\,\zeta_4 + \frac{1280}{3}\,\zeta_5\right) \\
&+\; C_A\, T_F^4\, n_f^4\; \left(\frac{916}{243} - \frac{640}{81}\,\zeta_3\right) \;-\; C_F\, T_F^4\, n_f^4\; \left(\frac{856}{243} + \frac{128}{27}\,\zeta_3\right) \quad .
\end{aligned}
$$

Where does this leave us?

We have now a number of new tools: Forcer and the Rstar program. They can be used for the computation of 4-loop (finite and infinite) quantities and 5-loop infinite objects.

In addition we have the (much more difficult) Chetyrkin method that might be usable as well for the computation of more 5-loop divergent quantities. And it could also be useful for high moments of 4-loop splitting functions (?).

We have produced a number of papers on splitting functions, Higgs decay at the 4 and 5 loop level. This has advanced QCD perturbation theory almost by a full order.

It is very clear that more progress can only come from even higher levels of automation.