Automating the computation of (generalized) functional supertraces beyond one loop

Renato Fonseca
renatofonseca@ugr.es
High-Energy Physics Group, University of Granada


https://www.voyagetips.com/en/things-to-do-in-geneva/

HEFT 2025,
CERN, June 2025

Junta de Andalucía
MINISTERIO DE CIENCIA, INNOVACIÓN Y UNIVERSIDADES
AGENCIA ESTATAL DE INVESTIGACIÓN
MINISTERIO DE CIENCIA, INNOVACIÓN Y UNIVERSIDADES
Cofinanciado por la Unión Europea
AGENCIA ESTATAL DE INVESTIGACIÓN
AST22_6.5
PID2022-139466NB-C22
PID2022-139466NB-C21
P21-00199

1-loop effective action

Goldstone Salam Weinberg (1962)
Coleman Weinberg (1973)
Jackiw (1973)
Cornwall Jackiw Tomboulis (1973)
...

$$\Gamma^{(1)}_{\text{eff}}[\phi] = S_{\hbar^0}[\phi] + S_{\hbar^1}[\phi] + \frac{i}{2}\text{Tr}\log\left(\frac{\delta^2 S_{\hbar^0}[\phi]}{\delta\phi(x)\delta\phi(y)}\right)$$

Evaluated at

$$\phi_{\text{cl}} = \langle 0|\phi|0\rangle_J$$

For one scalar field

For fermions:
$\text{Tr} \rightarrow \text{STr}$

Several scalars:
Add internal index
$\phi \rightarrow \phi_a$

Gauge bosons:
Same idea; some work needed to maintain explicit gauge invariance

$$\text{Tr}\log\underbrace{\left(\frac{\delta^2 S_{\hbar^0}[\phi]}{\delta\phi_a(x)\delta\phi_b(y)}\right)}$$

Trace is performed over the internal indices and x, y
(the coincidence limit)

$$\equiv Q = \Delta^{-1} - X$$

where Δ comes from the kinetic and mass terms

$$\text{Tr}\log Q = \underbrace{\text{Tr}\log\Delta^{-1}} - \underbrace{\boxed{\sum_{n=1}^{\infty}\frac{1}{n}\text{Tr}\left[(\Delta X)^n\right]}}$$

Let us focus on this part

Universal part
Depends only on the free Lagrangian

Model specific
Depends on the interactions X

Programs that compute supertraces

Significant extra functionalities.
Under update to handle 1+ loops

STrEAM

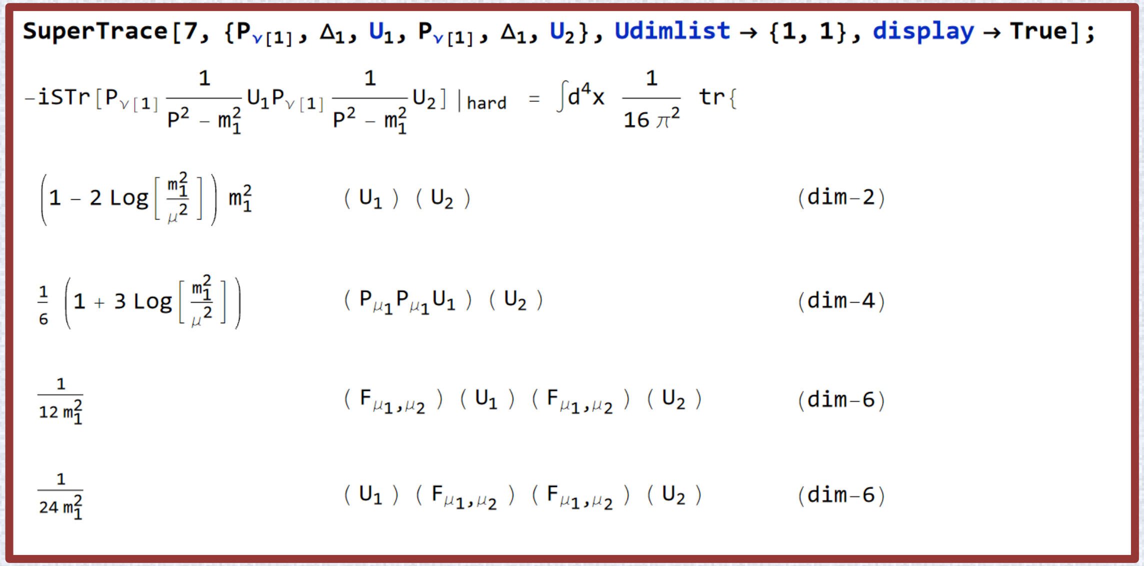
Cohen, Lu, Zhang
2011.02484, 2012.07851

SUPER TRACER

Fuentes-Martin, König, Pagès,
Thomsen, Wilsch 2012.08506

MATCHETE

Fuentes-Martin, König, Pagès,
Thomsen, Wilsch 2212.04510



```
SuperTrace[7, {Pν[1], Δ1, U1, Pν[1], Δ1, U2}, Udimlist → {1, 1}, display → True];
```

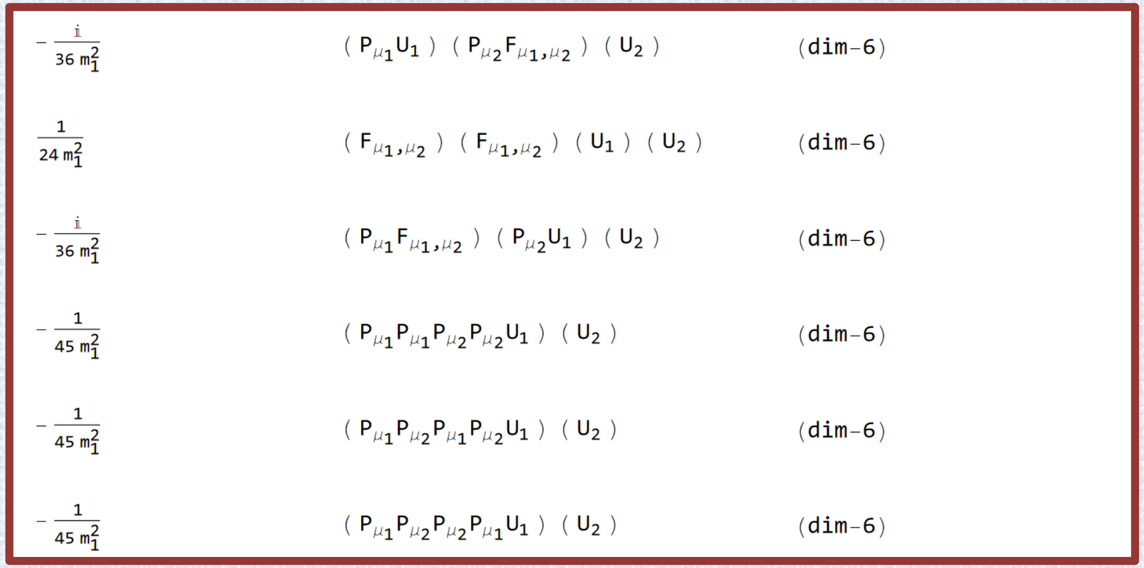
$$-i\mathrm{STr}\left[P_{\nu[1]}\frac{1}{P^2-m_1^2}U_1P_{\nu[1]}\frac{1}{P^2-m_1^2}U_2\right]\Big|_{\text{hard}} = \int d^4x\,\frac{1}{16\pi^2}\,\mathrm{tr}\{$$

$$\left(1-2\,\mathrm{Log}\left[\frac{m_1^2}{\mu^2}\right]\right)m_1^2 \quad (U_1)\,(U_2) \quad (\text{dim}-2)$$

$$\frac{1}{6}\left(1+3\,\mathrm{Log}\left[\frac{m_1^2}{\mu^2}\right]\right) \quad (P_{\mu_1}P_{\mu_1}U_1)\,(U_2) \quad (\text{dim}-4)$$

$$\frac{1}{12\,m_1^2} \quad (F_{\mu_1,\mu_2})\,(U_1)\,(F_{\mu_1,\mu_2})\,(U_2) \quad (\text{dim}-6)$$

$$\frac{1}{24\,m_1^2} \quad (U_1)\,(F_{\mu_1,\mu_2})\,(F_{\mu_1,\mu_2})\,(U_2) \quad (\text{dim}-6)$$



$$-\frac{i}{36\,m_1^2} \quad (P_{\mu_1}U_1)\,(P_{\mu_2}F_{\mu_1,\mu_2})\,(U_2) \quad (\text{dim}-6)$$

$$\frac{1}{24\,m_1^2} \quad (F_{\mu_1,\mu_2})\,(F_{\mu_1,\mu_2})\,(U_1)\,(U_2) \quad (\text{dim}-6)$$

$$-\frac{i}{36\,m_1^2} \quad (P_{\mu_1}F_{\mu_1,\mu_2})\,(P_{\mu_2}U_1)\,(U_2) \quad (\text{dim}-6)$$

$$-\frac{1}{45\,m_1^2} \quad (P_{\mu_1}P_{\mu_1}P_{\mu_2}P_{\mu_2}U_1)\,(U_2) \quad (\text{dim}-6)$$

$$-\frac{1}{45\,m_1^2} \quad (P_{\mu_1}P_{\mu_2}P_{\mu_1}P_{\mu_2}U_1)\,(U_2) \quad (\text{dim}-6)$$

$$-\frac{1}{45\,m_1^2} \quad (P_{\mu_1}P_{\mu_2}P_{\mu_2}P_{\mu_1}U_1)\,(U_2) \quad (\text{dim}-6)$$

In this talk I will discuss achieving similar results beyond 1 loop

The $\mathrm{Tr}\left[(\Delta X)^n\right]$ terms (the "power-like" terms)

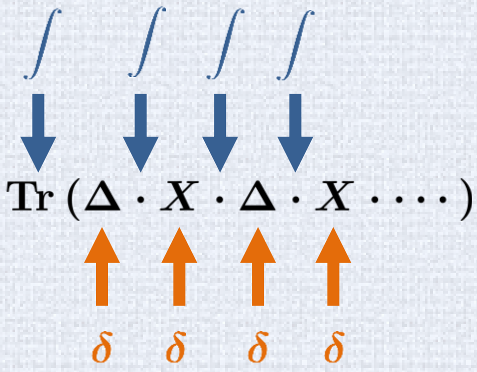
For local interactions, Q (and thus X and Δ) contains delta functions and it's derivatives:

$$Q_{ab}(x,y) = Q_{ab}^{(0)}(x)\,\underline{\delta(x-y)} + Q_{ab,\mu}^{(1)}(x)\,\underline{\left[\partial_x^\mu \delta(x-y)\right]} + Q_{ab,\mu\nu}^{(2)}(x)\,\underline{\left[\partial_x^\mu \partial_x^\nu \delta(x-y)\right]} + \cdots$$

In the case of a scalar, the free part reads

$$\Delta_{ab}^{-1}(x,y) = \delta_{ab}\left(P^2 - m^2\right)\delta(x-y) \qquad P^\mu \equiv i\partial_x^\mu$$

However, at the same time there are many integrals (<u>in fact as many as deltas</u>)



$$\Longrightarrow \quad \mathrm{Tr}\log\left(\frac{\delta^2 S_{\hbar^0}[\phi]}{\delta\phi(x)\delta\phi(y)}\right) = \int (\cdots)\,\underbrace{\delta(x-x)}\,dx$$

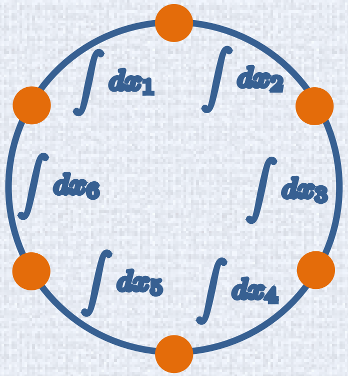
Need to regulate

The usual solution is to Fourier expand the delta function

$$\delta(x-y) = \frac{1}{(2\pi)^d}\int_k \exp\left[ik(x-y)\right]$$

The $\mathrm{Tr}\left[(\Delta X)^n\right]$ terms

(the “power-like” terms)



$X(x_1, P_{x_1})\,\boldsymbol{\delta(x_1 - x_2)}$

$\Delta(x_2, P_{x_2})\,\boldsymbol{\delta(x_2 - x_3)}$

$X(x_3, P_{x_3})\,\boldsymbol{\delta(x_3 - x_4)}$

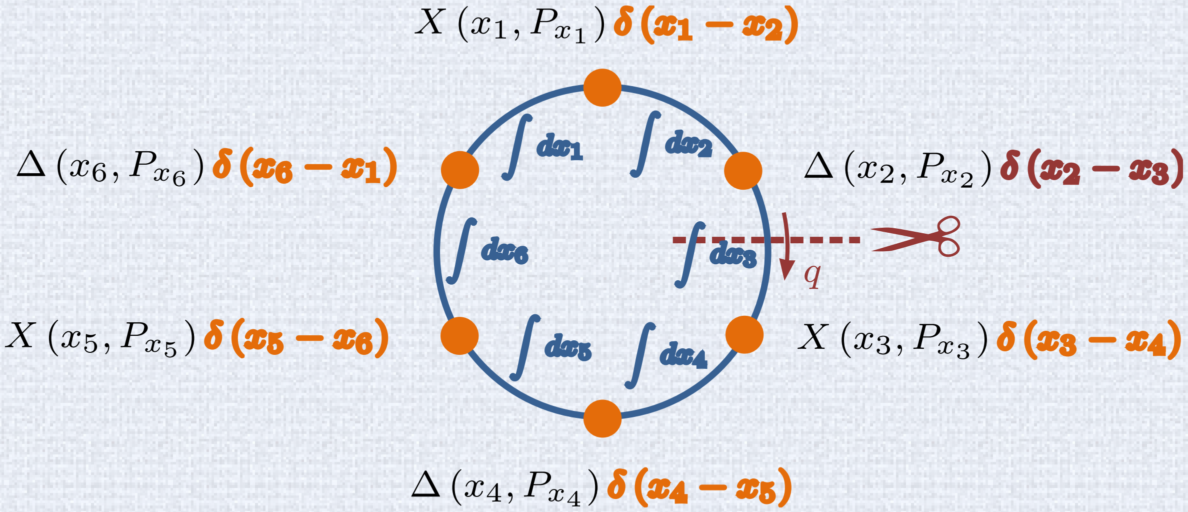
$\Delta(x_4, P_{x_4})\,\boldsymbol{\delta(x_4 - x_5)}$

$X(x_5, P_{x_5})\,\boldsymbol{\delta(x_5 - x_6)}$

$\Delta(x_6, P_{x_6})\,\boldsymbol{\delta(x_6 - x_1)}$

$\int dx_1$ $\int dx_2$ $\int dx_3$ $\int dx_4$ $\int dx_5$ $\int dx_6$

The Tr $\left[(\Delta X)^n\right]$ terms (the "power-like" terms)



$X(x_1, P_{x_1})\,\boldsymbol{\delta(x_1 - x_2)}$

$\Delta(x_6, P_{x_6})\,\boldsymbol{\delta(x_6 - x_1)}$

$\Delta(x_2, P_{x_2})\,\boldsymbol{\delta(x_2 - x_3)}$

$X(x_5, P_{x_5})\,\boldsymbol{\delta(x_5 - x_6)}$

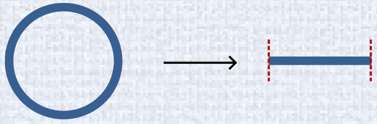
$X(x_3, P_{x_3})\,\boldsymbol{\delta(x_3 - x_4)}$

$\Delta(x_4, P_{x_4})\,\boldsymbol{\delta(x_4 - x_5)}$

Fourier expand **one of the delta functions:**

$$\delta(x_2 - x_3) = \frac{1}{(2\pi)^d}\int_q \exp\left[iq(x_2 - x_3)\right]$$

Note also that the loop diagram becomes a tree diagram



The result of such "cut":

$$\mathrm{Tr}\left[(\Delta X)^n\right] = \int \overbrace{\mathrm{tr}\left[\left(\Delta(P-q)\,X(x, P-q)\right)^n\right]}^{\supset \mathcal{L}_{\mathrm{eff}}} dx$$

The P's apply to everything in front of them ("open derivatives")

Trace only over internal indices and integral over q

q is the momenta variable from the Fourier expansion

No delta's and no integrals remaining in the effective Lagrangian

One should expand $\Delta(P-q)$ to the desired order in P

$$\Delta(P-q) = \frac{1}{q^2 - m^2}\sum_{m=0}^{\infty}\left(\frac{2q\cdot P - P^2}{q^2 - m^2}\right)^m$$

Gauge theories

$$P_\mu = \cancel{i\partial_\mu}\, iD_\mu \qquad D_\mu = \partial_\mu - igA_\mu^a T^a \qquad F_{\mu\nu} = -i\left[P_\mu, P_\nu\right]$$

... and use background field method

One needs to calculate traces of the form

The U_i are different blocks inside what we've been calling X

$$\mathrm{tr}\left[\left(U_1 P_{\mu_1} P_{\mu_2} \cdots\right) \Delta \left(U_2 P_{\nu_1} P_{\nu_2} \cdots\right) \Delta \cdots\right]$$

Derivatives stop at the end of the trace. This end is determined by the position of the cut

Open derivatives (the P's) act on everything to their right. At the end of the trace,

$$\left(P_\mu\right)_{\text{end}} = i\left(D_\mu\right)_{\text{end}} = gA_\mu^a T^a$$

Not gauge covariant

Gaillard (1986)
Chan (1986)
Cheyette (1988)
Henning, Lu, Murayama (2014, 2016)

To make the each individual term of the calculation explicitly gauge invariant, one can use the covariant derivative expansion (CDE)

$$P_\mu - q_\mu \rightarrow -q_\mu + G_{\mu\nu}^{\text{CDE}} \partial/\partial q_\nu \qquad U_k \rightarrow U_k^{\text{CDE}}$$

Open derivatives disappear

Gauge covariant

DeWitt (1965)
Avramidi (1990 - 2000)
Kuzenko McArthur (2003)
Fuentes-Martín Moreno-Sánchez
Palavric Thomsen (2024)

Alternative to CDE

For local interactions, Q (and thus X and Δ) contains delta functions and it's derivatives:

$$Q_{ab}\left(x,y\right)=Q_{ab}^{(0)}\left(x\right)\delta\left(x-y\right)+Q_{ab,\mu}^{(1)}\left(x\right)\left[\partial_{x}^{\mu}\delta\left(x-y\right)\right]+Q_{ab,\mu\nu}^{(2)}\left(x\right)\left[\partial_{x}^{\mu}\partial_{x}^{\nu}\delta\left(x-y\right)\right]+\cdots$$

Flashback to slide 4

In a gauge theory, it is much better to think differently:

$$\delta\phi_{a}\left(x\right)/\delta\phi_{b}\left(x\right)=\cancel{\delta_{ab}\delta\left(x-y\right)}\,I_{ab}\left(x,y\right)\delta\left(x-y\right)$$

The coincidence limit (x=y) of the I function is just a delta function: $I_{ab}\left(x,x\right)=\delta_{ab}$

But crucially D's/P's act on it covariantly:

$$D_{\mu}I\left(x,x'\right)=-iI(x,x')\sum_{n=1}^{\infty}\frac{n}{(n+1)!}\left(x-x'\right)^{\nu_{1}}\cdots\left(x-x'\right)^{\nu_{n}}D'_{\nu_{1}}\cdots D'_{\nu_{n-1}}F_{\nu_{n}\mu}\left(x'\right)$$

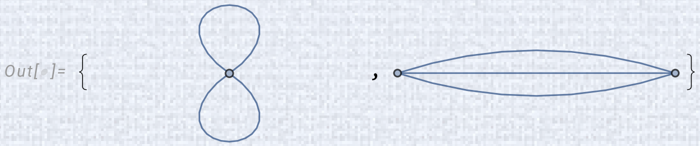
$$D'_{\mu}I\left(x,x'\right)=i\left[\sum_{n=1}^{\infty}\frac{n}{(n+1)!}\left(x'-x\right)^{\nu_{1}}\cdots\left(x'-x\right)^{\nu_{n}}D_{\nu_{1}}\cdots D_{\nu_{n-1}}F_{\nu_{n}\mu}\left(x\right)\right]I(x,x')$$

Kuzenko McArthur
(2003)

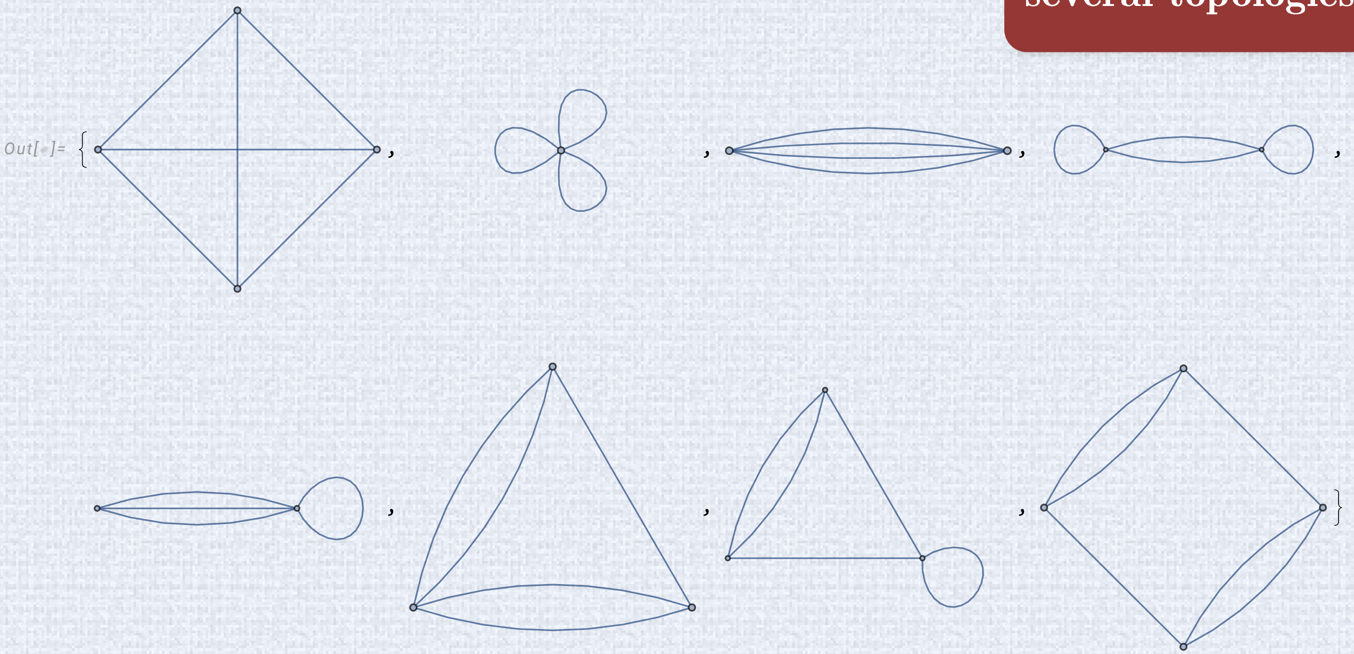
Multi-loops vacuum diagrams

Must consider several topologies

```
In[ ]:= AdjacencyGraph /@ vacuumDiagrams1PI[2]
```



```
In[ ]:= AdjacencyGraph /@ vacuumDiagrams1PI[3]
```



# loops	Vacuum diagrams	1PI vacuum diagrams
1	1	**1**
2	3	**2**
3	15	**8**
4	111	**43**
5	1076	**334**
6	13870	**3727**
7	220520	**55533**

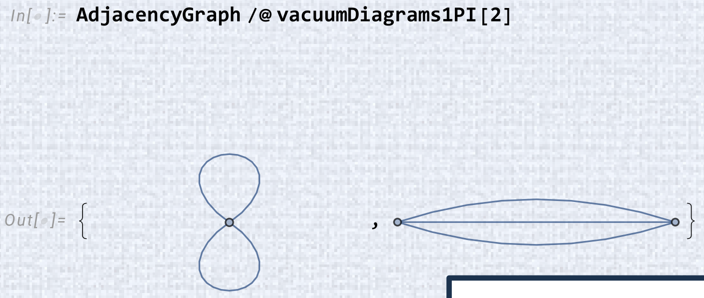
Number of vacuum diagrams scales rapidly with the number of loops

Analytical results for the momenta integrals are limited beyond 3 loops

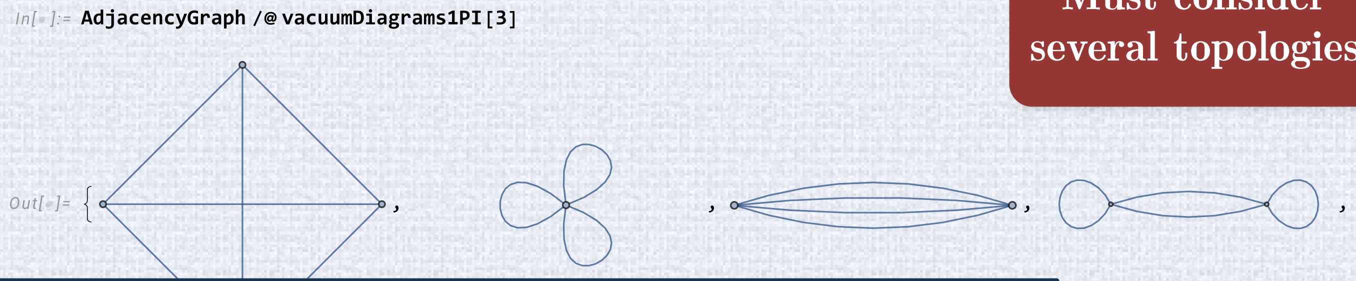
Still, for very simple models it might be possible to have results up to a large number of loops using functional methods

Multi-loops vacuum diagrams

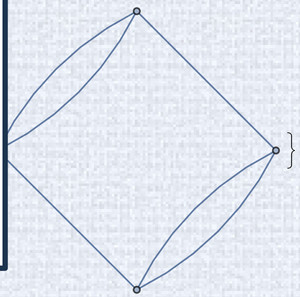
Must consider several topologies



```
In[ ]:= AdjacencyGraph /@ vacuumDiagrams1PI[2]
```



```
In[ ]:= AdjacencyGraph /@ vacuumDiagrams1PI[3]
```



# loops	Vacuum diagrams	
1	1	
2	3	
3	15	**8**
4	111	**43**
5	1076	**334**
6	13870	**3727**
7	220520	**55533**

Considerable literature on multi-loop functional calculations

Jack, Osborn (1982)
Kuzenko McArthur (2003)
Gersdorff Santos (2023)
Fuentes-Martín Palavric Thomsen (2023)
Banerjee, Chakrabortty, Ramkumar (2024)
Fuentes-Martín Moreno-Sánchez Palavric Thomsen (2024)
...

Number of vacuum diagrams scales rapidly with the number of loops

Analytical results for the momenta integrals are limited beyond 3 loops

Still, for very simple models it might be possible to have results up to a large number of loops using functional methods

Effective action beyond 1-loop

1 Take the expression and build the underlying graph

Example

`A[i,j,k]**Δ[i]**P[μ,i]**Δ[j]**Δ[k]**B[i,m,n]**P[μ,m]**Δ[m]**Δ[n]**C[n,j,o]**Δ[o]**E[m,k,o]`

4 Vertices

6 propagators (in front of one of the two vertices to which it connects and with a label of the corresponding edge)

2 open derivatives
(with a label of the edge; applies away from the preceding vertex)

Effective action beyond 1-loop

1

Take the expression and build the underlying graph

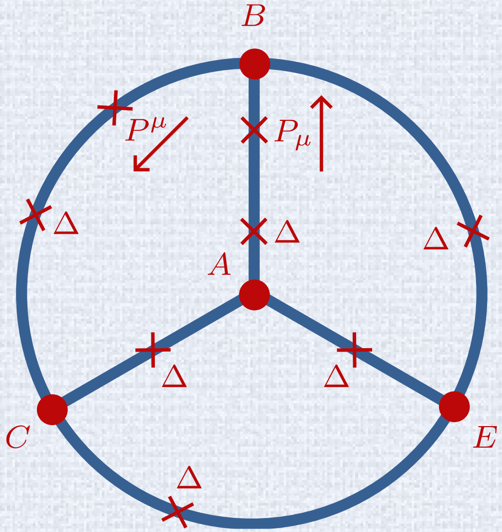
Example

`A[i,j,k]**Δ[i]**P[μ,i]**Δ[j]**Δ[k]**B[i,m,n]**P[μ,m]**Δ[m]**Δ[n]**C[n,j,o]**Δ[o]**E[m,k,o]`

4 Vertices

6 propagators (in front of one of the two vertices to which it connects and with a label of the corresponding edge)

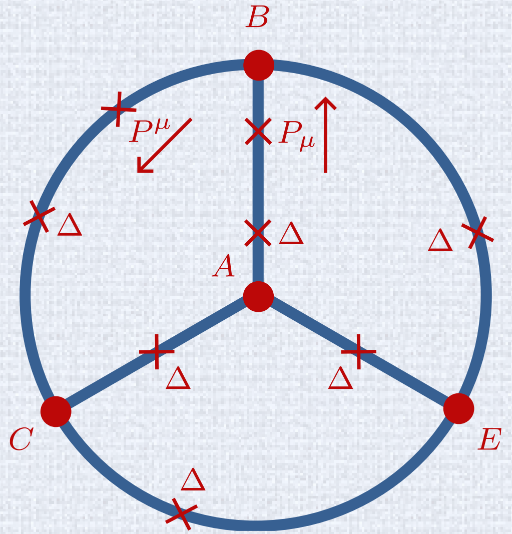
2 open derivatives
(with a label of the edge; applies away from the preceding vertex)



Effective action beyond 1-loop

2

For an ℓ -loop diagram, make ℓ cuts transforming it into a tree/connected diagram



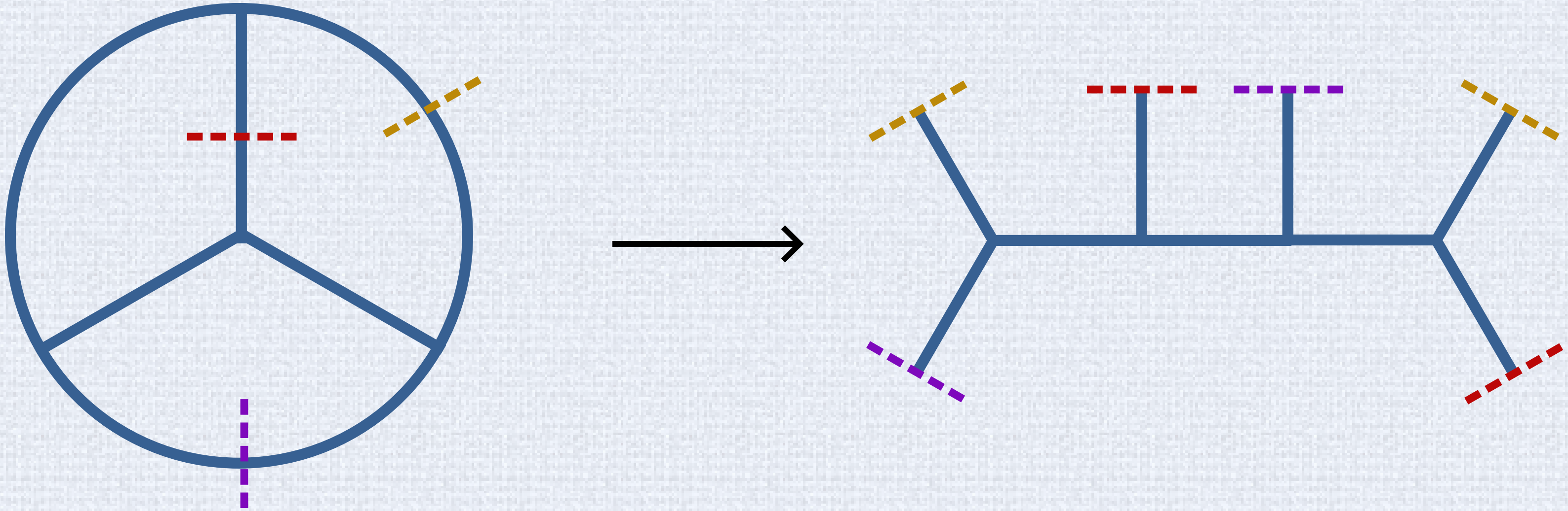
Effective action beyond 1-loop

2 **For an ℓ -loop diagram, make ℓ cuts transforming it into a tree/connected diagram**

Effective action beyond 1-loop

2

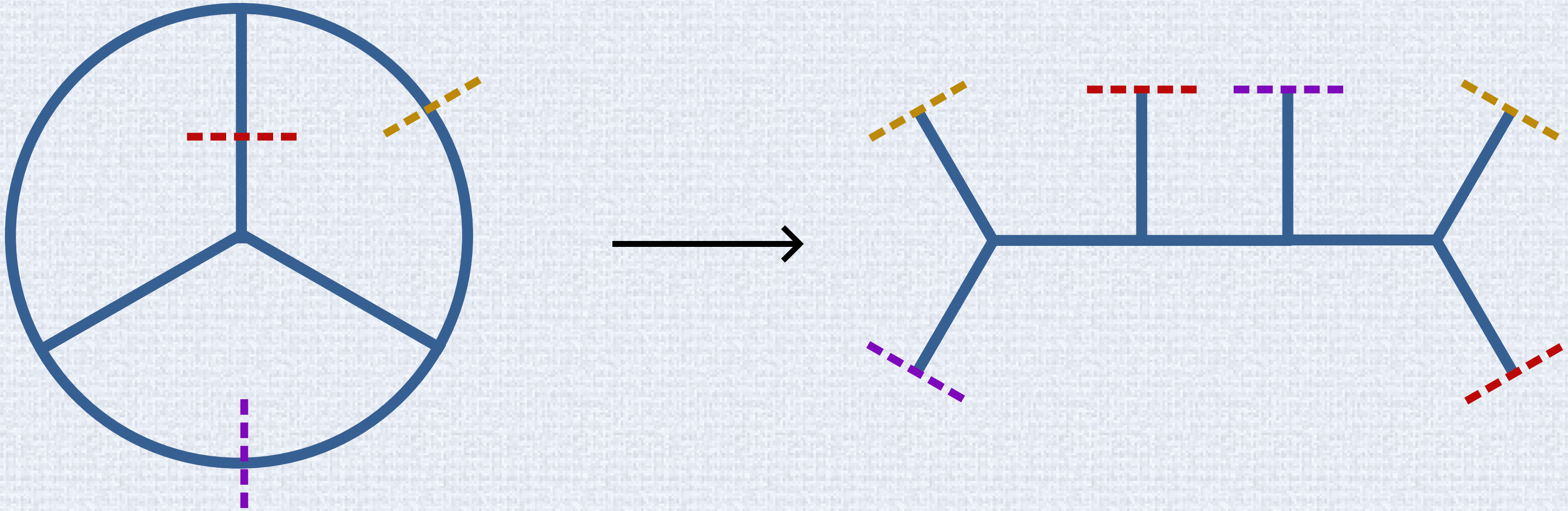
For an ℓ-loop diagram, make ℓ cuts transforming it into a tree/connected diagram



Effective action beyond 1-loop

2

For an ℓ -loop diagram, make ℓ cuts transforming it into a tree/connected diagram

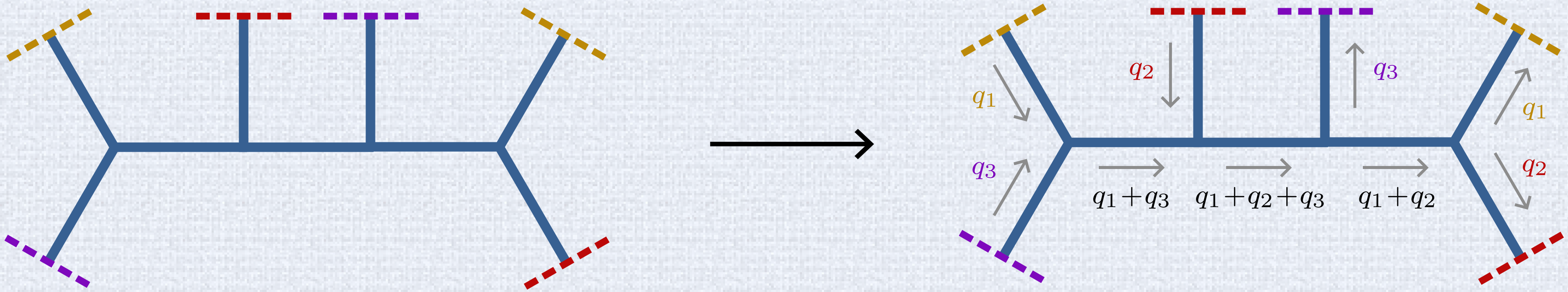


Location of the cuts is not uniquely determined. It is only necessary to turn the diagram into a tree diagram
In each edge/line one can also pick the exact position of the cuts (e.g. before or after the propagators)

Effective action beyond 1-loop

3

Label the momenta through these ℓ cuts, which fully determine the momenta in all lines. One must integrate over these momenta.



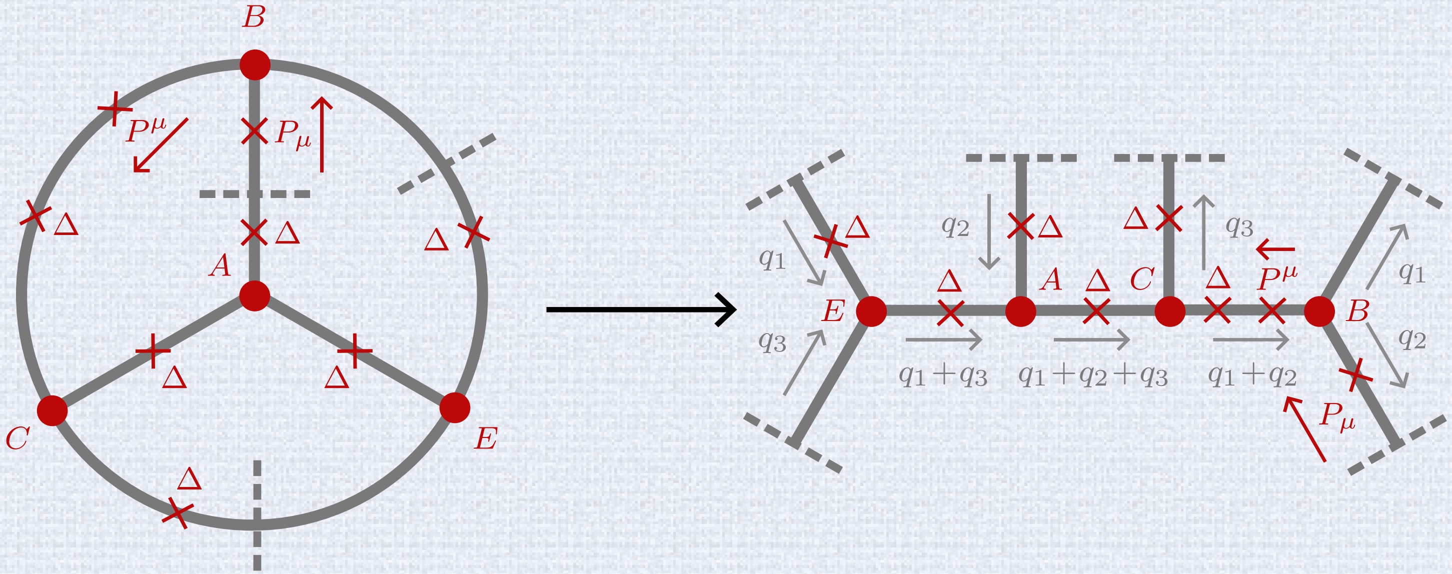
$$\left[\frac{1}{(2\pi)^d}\right]^3 \int\int\int (\cdots)\, d^dq_1 d^dq_2 d^dq_3$$

Effective action beyond 1-loop

3

Shift each open P by the momenta flowing in the line.
Each P acts on every vertex/propagator it find in front, until it meets the cuts.

Since the cut diagram has no loops, all P's are guaranteed to eventually hit a cut

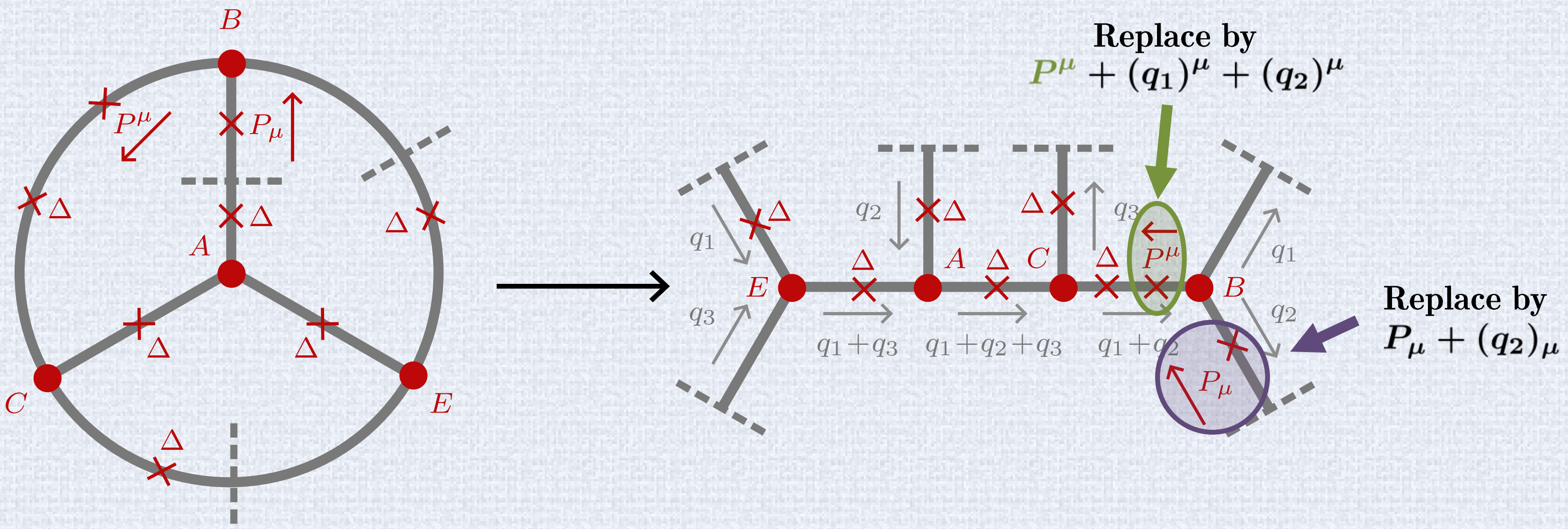


Effective action beyond 1-loop

3

Shift each open P by the momenta flowing in the line.
Each P acts on every vertex/propagator it find in front, until it meets the cuts.

Since the cut diagram has no loops, all P's are guaranteed to eventually hit a cut



Replace by
$P^\mu + (q_1)^\mu + (q_2)^\mu$

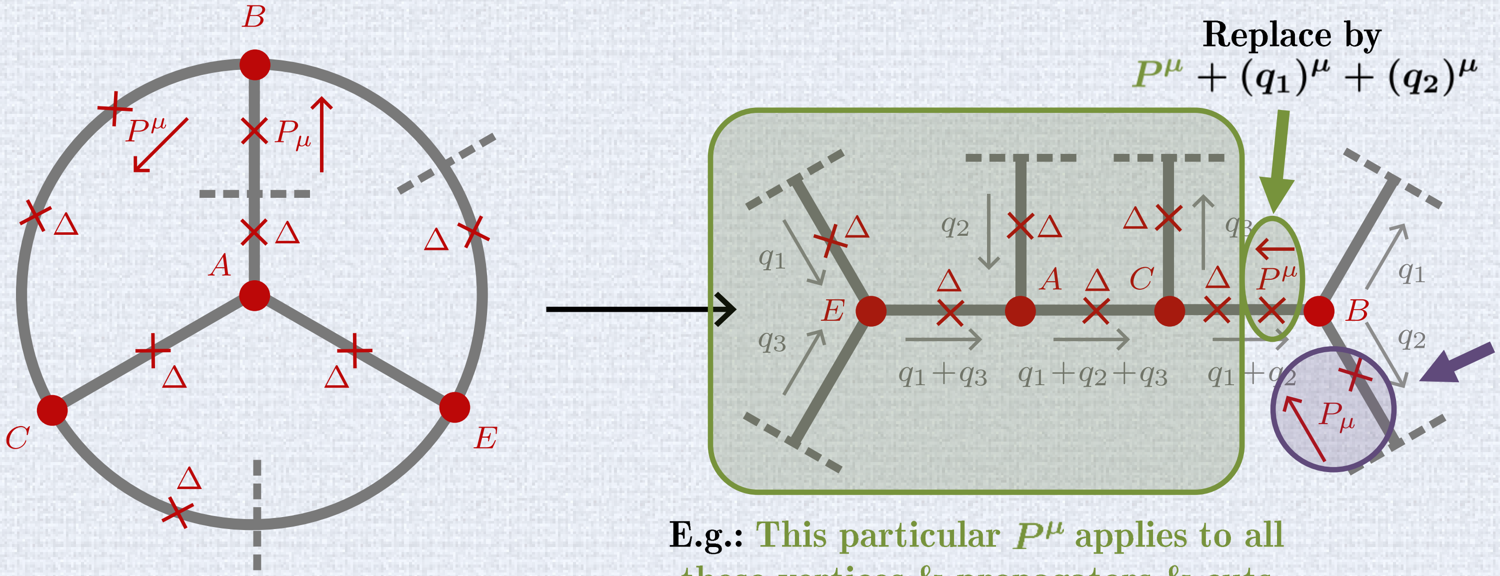
Replace by
$P_\mu + (q_2)_\mu$

Effective action beyond 1-loop

3

Shift each open P by the momenta flowing in the line.
Each P acts on every vertex/propagator it find in front, until it meets the cuts.

Since the cut diagram has no loops, all P's are guaranteed to eventually hit a cut



Replace by
$\boldsymbol{P^\mu + (q_1)^\mu + (q_2)^\mu}$

Replace by
$\boldsymbol{P_\mu + (q_2)_\mu}$

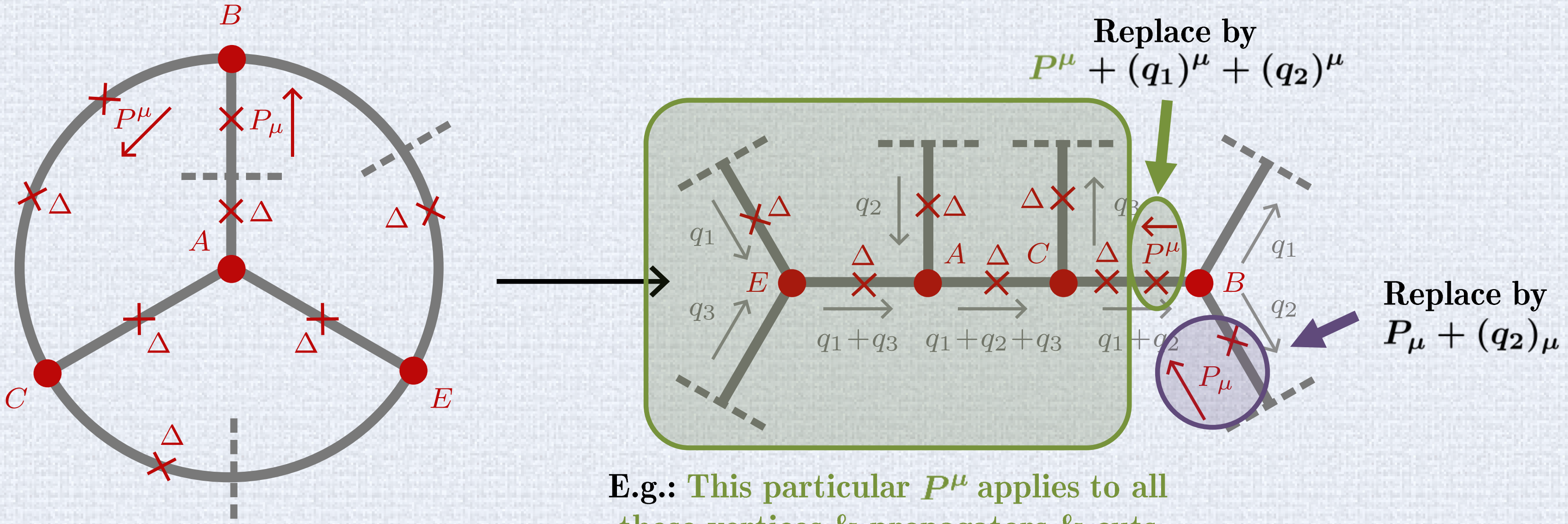
E.g.: This particular P^μ applies to all these vertices & propagators & cuts

Effective action beyond 1-loop

3

Shift each open P by the momenta flowing in the line.
Each P acts on every vertex/propagator it find in front, until it meets the cuts.

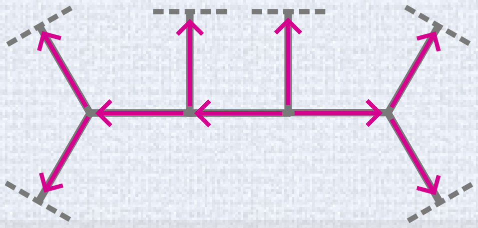
Since the cut diagram has no loops, all P's are guaranteed to eventually hit a cut



Replace by
$P^\mu + (q_1)^\mu + (q_2)^\mu$

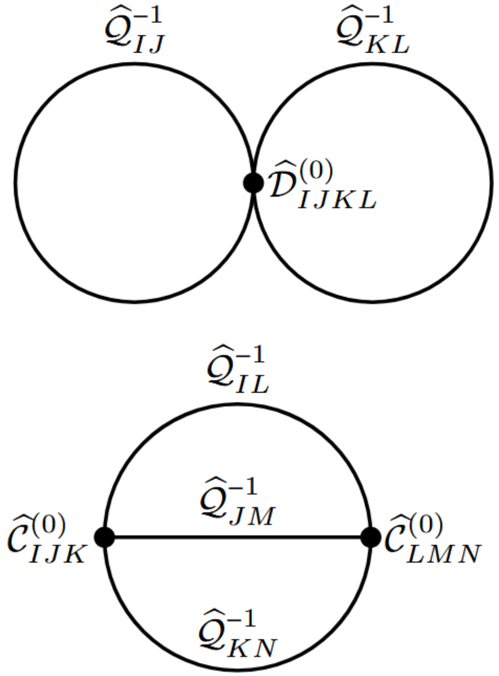
Replace by
$P_\mu + (q_2)_\mu$

E.g.: This particular P^μ applies to all these vertices & propagators & cuts



Note 1: This also applies to the open derivatives inside the propagators (the Δ's)
Note 2: It is possible to avoid commuting P's by applying first those closest to the cuts and progressively moving on to the P's in the center of the diagram

Not the only way to do things



$$G_{\text{f8}} = \sum_{m,n,r} (\text{-}1)^{m+n+r} \int_{x,k,\ell} D^{\underline{m},\underline{n},\underline{r}}_{abcd}(x)\, k^{\underline{m}}\, \ell^{\underline{r}} \\ \times \left[(P_x - k)^{\underline{n}}\, Q^{-1}_{ba}(x,\, P_x - k)\right]\left[Q^{-1}_{dc}(x,\, P_x - \ell)\right],$$

$$G_{\text{ss}} = \sum_{m^{(\prime)},\, n^{(\prime)} \boxed{,\, s}} (-1)^{m+n+m'+n'} \frac{i^s}{s!} \int_{x,k,l} C^{\underline{m},\underline{n}}_{abc}(x) \boxed{\partial^{\underline{s}}_{x}} C^{\underline{m}',\underline{n}'}_{def}(x) \\ \times \left[\boxed{\partial^{\underline{s}}_{k}} Q^{-1}_{cf}(x,\, P_x + k + \ell)\right] k^{\underline{m}'}\, \ell^{\underline{n}'} \\ \times \left[(P_x - k)^{\underline{m}}\, Q^{-1}_{ad}(x,\, P_x - k)\right] \\ \times \left[(P_x - \ell)^{\underline{n}}\, Q^{-1}_{be}(x,\, P_x - \ell)\right], \qquad (23)$$

Fuentes-Martín Palavric Thomsen (2023); see also Kuzenko McArthur (2003)

Algorithm just described would reproduce this expression ...

... but not this one:
1- No need for the s sum
(a Taylor expansion)
2- Instead, the open derivatives in $Q^{-1}_{cf}(x, P_x + k + \ell)$ must be allowed to act on one of the C vertices

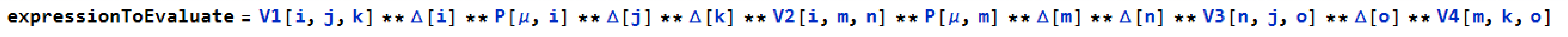
The recipe described in the previous slides can be used automatically for any topology; no need to think case by case.

I've been trying to code it into a Mathematica package. It is a work in progress but I will report on its current status

The code

Work in progress

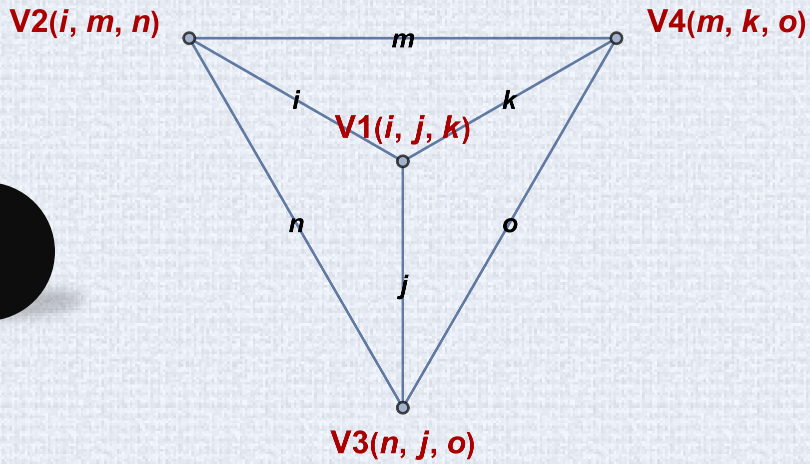
Results shown here are just to illustrate how it works. Not cross checked!!!



```
expressionToEvaluate = V1[i, j, k] ** Δ[i] ** P[μ, i] ** Δ[j] ** Δ[k] ** V2[i, m, n] ** P[μ, m] ** Δ[m] ** Δ[n] ** V3[n, j, o] ** Δ[o] ** V4[m, k, o]
```

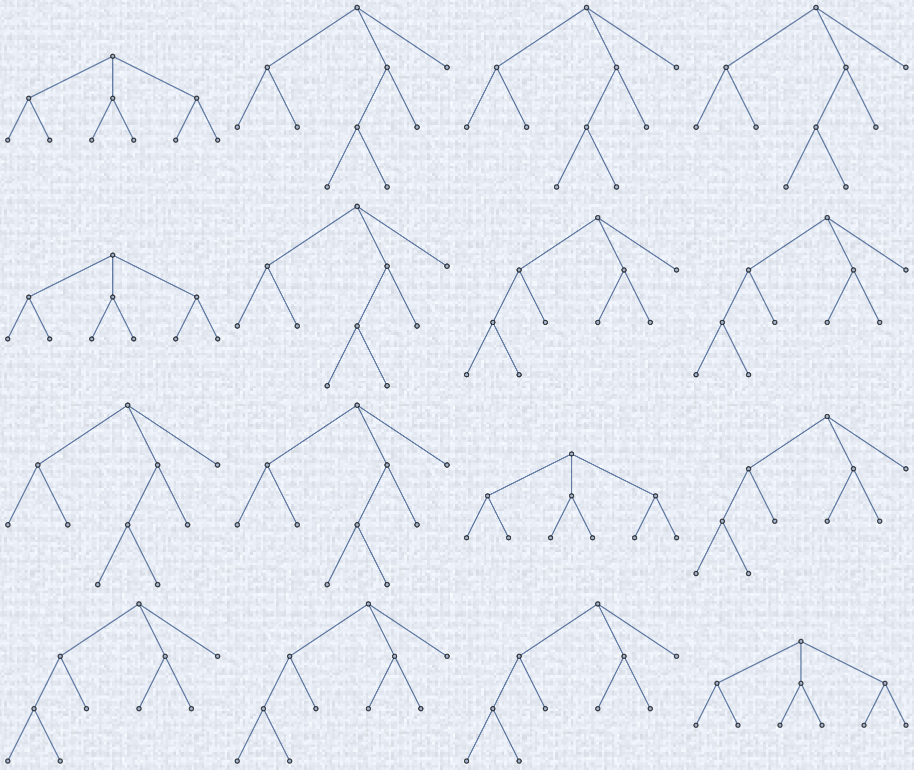
All results in this and the following slides are obtained in an automated way. Code would have easily handled other examples.

1



User does not need to be aware of this, but internally the code finds the graph associated to the given expression.

2



Find all ways to make 3 cuts which leaves the graph connected (must be a tree graph). Picks out one with short branches

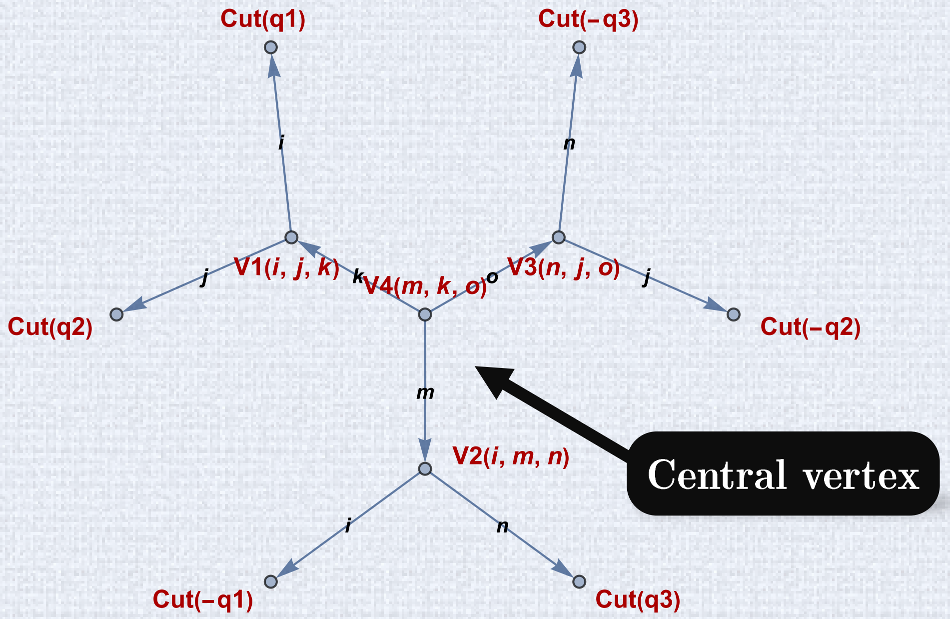
The code

Work in progress

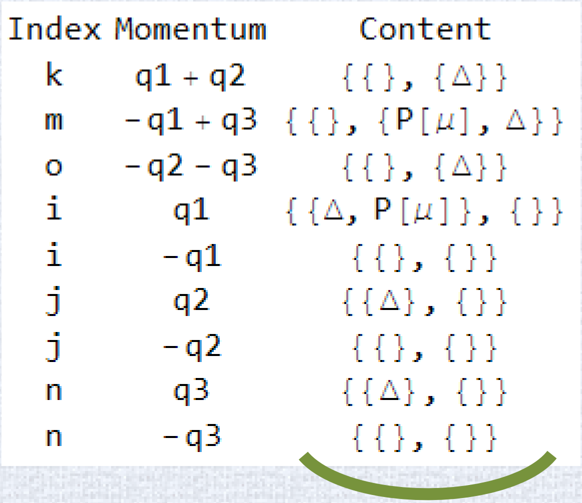
Results shown here are just to illustrate how it works. Not cross checked!!!

```
expressionToEvaluate = V1[i, j, k] ** Δ[i] ** P[μ, i] ** Δ[j] ** Δ[k] ** V2[i, m, n] ** P[μ, m] ** Δ[m] ** Δ[n] ** V3[n, j, o] ** Δ[o] ** V4[m, k, o]
```

2



Central vertex



Index	Momentum	Content
k	q1 + q2	{{}, {Δ}}
m	- q1 + q3	{{}, {P[μ], Δ}}
o	- q2 - q3	{{}, {Δ}}
i	q1	{{Δ, P[μ]}, {}}
i	- q1	{{}, {}}
j	q2	{{Δ}, {}}
j	- q2	{{}, {}}
n	q3	{{Δ}, {}}
n	- q3	{{}, {}}

Picks this cut. Assigns a preferred direction to each edge: open derivatives *P* will be applied in this direction.
Direction is chosen so as to minimize the size of the branches which need to be differentiated. Momenta has also been assigned to all edges.

Throughout these graph manipulation, the code keeps track of what expressions are in each line (propagators and P's)

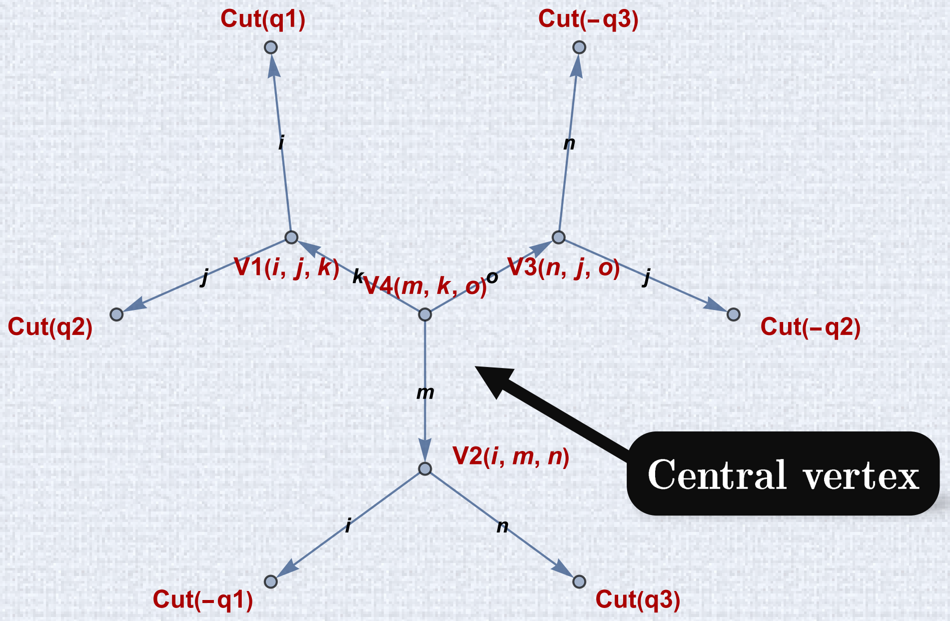
The code

Work in progress

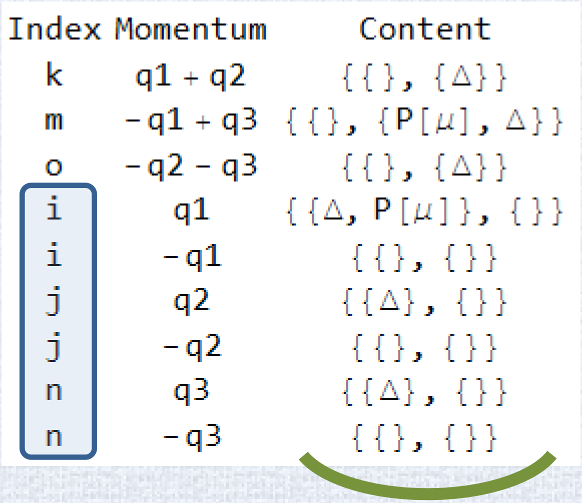
Results shown here are just to illustrate how it works. Not cross checked!!!

```
expressionToEvaluate = V1[i, j, k] ** Δ[i] ** P[μ, i] ** Δ[j] ** Δ[k] ** V2[i, m, n] ** P[μ, m] ** Δ[m] ** Δ[n] ** V3[n, j, o] ** Δ[o] ** V4[m, k, o]
```

2



Central vertex



Index	Momentum	Content
k	q1 + q2	{{}, {Δ}}
m	- q1 + q3	{{}, {P[μ], Δ}}
o	- q2 - q3	{{}, {Δ}}
i	q1	{{Δ, P[μ]}, {}}
i	- q1	{{}, {}}
j	q2	{{Δ}, {}}
j	- q2	{{}, {}}
n	q3	{{Δ}, {}}
n	- q3	{{}, {}}

Cut lines have the same label at this stage. But later on they are changed to (i,i’), (j,j’), (n,n’)

Picks this cut. Assigns a preferred direction to each edge: open derivatives *P* will be applied in this direction. Direction is chosen so as to minimize the size of the branches which need to be differentiated. Momenta has also been assigned to all edges.

Throughout these graph manipulation, the code keeps track of what expressions are in each line (propagators and P’s)

The code

Work in progress

Results shown here are just to illustrate how it works. Not cross checked!!!

```
expressionToEvaluate = V1[i, j, k] ** Δ[i] ** P[μ, i] ** Δ[j] ** Δ[k] ** V2[i, m, n] ** P[μ, m] ** Δ[m] ** Δ[n] ** V3[n, j, o] ** Δ[o] ** V4[m, k, o]
```

3

Shift each open P by the momenta flowing in the line

Next the code expands the set of all propagators (the Δ's in the lines) up to some desired order

Finally, the open P's are applied to the relevant vertices and cuts

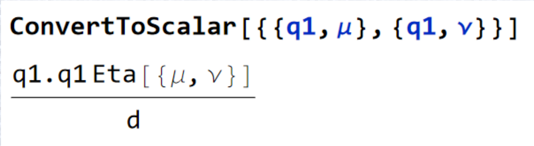
(For this example I expanded up P^2)

A very large number of terms is usually produced

Many are null or proportional to each other

To see this one needs to, further process the expressions, including reducing tensor integral to scalar integrals

E.g.: $$q_1^\mu q_1^\nu \rightarrow \frac{\eta^{\mu\nu} q_1^2}{d}$$



```
ConvertToScalar[{{q1, μ}, {q1, ν}}]
q1.q1 Eta[{μ, ν}]
─────────────────
        d
```

... but also multi-loop cases

The code

Work in progress

Results shown here are just to illustrate how it works. Not cross checked!!!

```
expressionToEvaluate = V1[i, j, k] ** Δ[i] ** P[μ, i] ** Δ[j] ** Δ[k] ** V2[i, m, n] ** P[μ, m] ** Δ[m] ** Δ[n] ** V3[n, j, o] ** Δ[o] ** V4[m, k, o]
```

3

Shift each open P by the momenta flowing in the line

Next the code expands the set of all propagators (the Δ's in the lines) up to some desired order

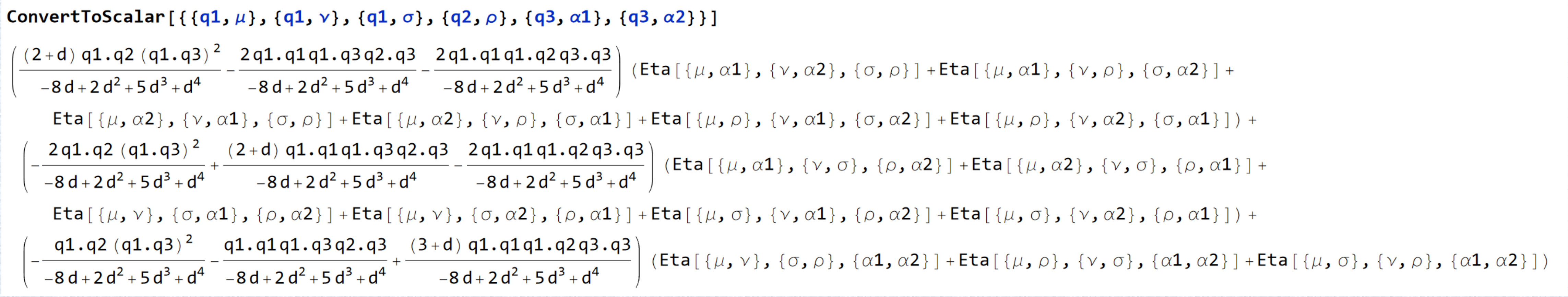
Finally, the open P's are applied to the relevant vertices and cuts

(For this example I expanded up P^2)

A very large number of terms is usually produced

Many are null or proportional to each other

To see this one needs to, further process the expressions, including reducing tensor integral to scalar integrals



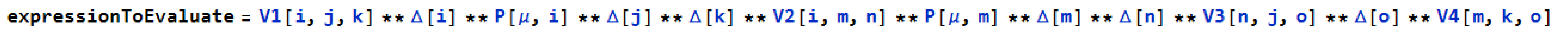
```
ConvertToScalar[{{q1, μ}, {q1, ν}, {q1, σ}, {q2, ρ}, {q3, α1}, {q3, α2}}]
```

$$\left(\frac{(2+d)\ q1.q2\ (q1.q3)^2}{-8d+2d^2+5d^3+d^4}-\frac{2\ q1.q1\ q1.q3\ q2.q3}{-8d+2d^2+5d^3+d^4}-\frac{2\ q1.q1\ q1.q2\ q3.q3}{-8d+2d^2+5d^3+d^4}\right)(\mathrm{Eta}[\{\mu,\alpha1\},\{\nu,\alpha2\},\{\sigma,\rho\}]+\mathrm{Eta}[\{\mu,\alpha1\},\{\nu,\rho\},\{\sigma,\alpha2\}]+$$
$$\mathrm{Eta}[\{\mu,\alpha2\},\{\nu,\alpha1\},\{\sigma,\rho\}]+\mathrm{Eta}[\{\mu,\alpha2\},\{\nu,\rho\},\{\sigma,\alpha1\}]+\mathrm{Eta}[\{\mu,\rho\},\{\nu,\alpha1\},\{\sigma,\alpha2\}]+\mathrm{Eta}[\{\mu,\rho\},\{\nu,\alpha2\},\{\sigma,\alpha1\}])+$$
$$\left(-\frac{2\ q1.q2\ (q1.q3)^2}{-8d+2d^2+5d^3+d^4}+\frac{(2+d)\ q1.q1\ q1.q3\ q2.q3}{-8d+2d^2+5d^3+d^4}-\frac{2\ q1.q1\ q1.q2\ q3.q3}{-8d+2d^2+5d^3+d^4}\right)(\mathrm{Eta}[\{\mu,\alpha1\},\{\nu,\sigma\},\{\rho,\alpha2\}]+\mathrm{Eta}[\{\mu,\alpha2\},\{\nu,\sigma\},\{\rho,\alpha1\}]+$$
$$\mathrm{Eta}[\{\mu,\nu\},\{\sigma,\alpha1\},\{\rho,\alpha2\}]+\mathrm{Eta}[\{\mu,\nu\},\{\sigma,\alpha2\},\{\rho,\alpha1\}]+\mathrm{Eta}[\{\mu,\sigma\},\{\nu,\alpha1\},\{\rho,\alpha2\}]+\mathrm{Eta}[\{\mu,\sigma\},\{\nu,\alpha2\},\{\rho,\alpha1\}])+$$
$$\left(-\frac{q1.q2\ (q1.q3)^2}{-8d+2d^2+5d^3+d^4}-\frac{q1.q1\ q1.q3\ q2.q3}{-8d+2d^2+5d^3+d^4}+\frac{(3+d)\ q1.q1\ q1.q2\ q3.q3}{-8d+2d^2+5d^3+d^4}\right)(\mathrm{Eta}[\{\mu,\nu\},\{\sigma,\rho\},\{\alpha1,\alpha2\}]+\mathrm{Eta}[\{\mu,\rho\},\{\nu,\sigma\},\{\alpha1,\alpha2\}]+\mathrm{Eta}[\{\mu,\sigma\},\{\nu,\rho\},\{\alpha1,\alpha2\}])$$

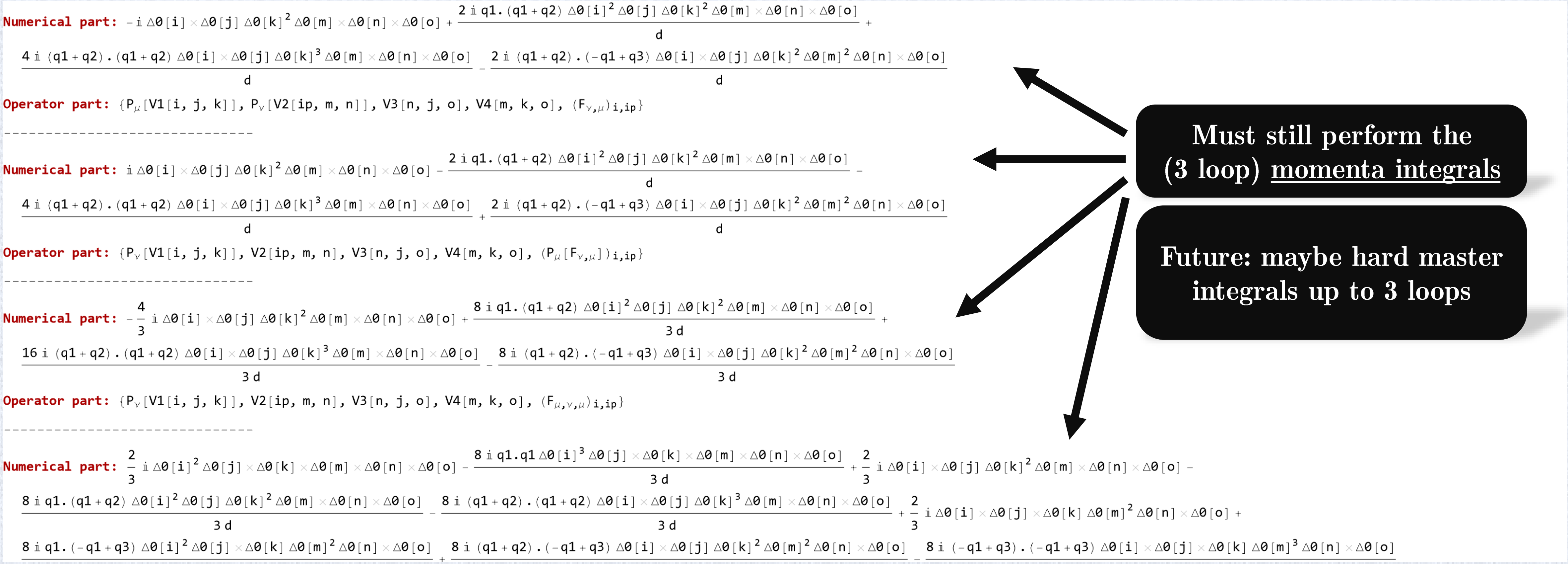
The code

Work in progress

Results shown here are just to illustrate how it works. Not cross checked!!!



```
expressionToEvaluate = V1[i, j, k] ** Δ[i] ** P[μ, i] ** Δ[j] ** Δ[k] ** V2[i, m, n] ** P[μ, m] ** Δ[m] ** Δ[n] ** V3[n, j, o] ** Δ[o] ** V4[m, k, o]
```



Numerical part:
$$-i\,\Delta0[i]\times\Delta0[j]\,\Delta0[k]^2\,\Delta0[m]\times\Delta0[n]\times\Delta0[o]+\frac{2\,i\,q1.(q1+q2)\,\Delta0[i]^2\,\Delta0[j]\,\Delta0[k]^2\,\Delta0[m]\times\Delta0[n]\times\Delta0[o]}{d}+\frac{4\,i\,(q1+q2).(q1+q2)\,\Delta0[i]\times\Delta0[j]\,\Delta0[k]^3\,\Delta0[m]\times\Delta0[n]\times\Delta0[o]}{d}-\frac{2\,i\,(q1+q2).(-q1+q3)\,\Delta0[i]\times\Delta0[j]\,\Delta0[k]^2\,\Delta0[m]^2\,\Delta0[n]\times\Delta0[o]}{d}$$

Operator part: $\{P_\mu[V1[i,j,k]],\ P_\nu[V2[ip,m,n]],\ V3[n,j,o],\ V4[m,k,o],\ (F_{\nu,\mu})_{i,ip}\}$

Numerical part:
$$i\,\Delta0[i]\times\Delta0[j]\,\Delta0[k]^2\,\Delta0[m]\times\Delta0[n]\times\Delta0[o]-\frac{2\,i\,q1.(q1+q2)\,\Delta0[i]^2\,\Delta0[j]\,\Delta0[k]^2\,\Delta0[m]\times\Delta0[n]\times\Delta0[o]}{d}-\frac{4\,i\,(q1+q2).(q1+q2)\,\Delta0[i]\times\Delta0[j]\,\Delta0[k]^3\,\Delta0[m]\times\Delta0[n]\times\Delta0[o]}{d}+\frac{2\,i\,(q1+q2).(-q1+q3)\,\Delta0[i]\times\Delta0[j]\,\Delta0[k]^2\,\Delta0[m]^2\,\Delta0[n]\times\Delta0[o]}{d}$$

Operator part: $\{P_\nu[V1[i,j,k]],\ V2[ip,m,n],\ V3[n,j,o],\ V4[m,k,o],\ (P_\mu[F_{\nu,\mu}])_{i,ip}\}$

Numerical part:
$$-\frac{4}{3}\,i\,\Delta0[i]\times\Delta0[j]\,\Delta0[k]^2\,\Delta0[m]\times\Delta0[n]\times\Delta0[o]+\frac{8\,i\,q1.(q1+q2)\,\Delta0[i]^2\,\Delta0[j]\,\Delta0[k]^2\,\Delta0[m]\times\Delta0[n]\times\Delta0[o]}{3\,d}+\frac{16\,i\,(q1+q2).(q1+q2)\,\Delta0[i]\times\Delta0[j]\,\Delta0[k]^3\,\Delta0[m]\times\Delta0[n]\times\Delta0[o]}{3\,d}-\frac{8\,i\,(q1+q2).(-q1+q3)\,\Delta0[i]\times\Delta0[j]\,\Delta0[k]^2\,\Delta0[m]^2\,\Delta0[n]\times\Delta0[o]}{3\,d}$$

Operator part: $\{P_\nu[V1[i,j,k]],\ V2[ip,m,n],\ V3[n,j,o],\ V4[m,k,o],\ (F_{\mu,\nu,\mu})_{i,ip}\}$

Numerical part:
$$\frac{2}{3}\,i\,\Delta0[i]^2\,\Delta0[j]\times\Delta0[k]\times\Delta0[m]\times\Delta0[n]\times\Delta0[o]-\frac{8\,i\,q1.q1\,\Delta0[i]^3\,\Delta0[j]\times\Delta0[k]\times\Delta0[m]\times\Delta0[n]\times\Delta0[o]}{3\,d}+\frac{2}{3}\,i\,\Delta0[i]\times\Delta0[j]\,\Delta0[k]^2\,\Delta0[m]\times\Delta0[n]\times\Delta0[o]-\frac{8\,i\,q1.(q1+q2)\,\Delta0[i]^2\,\Delta0[j]\,\Delta0[k]^2\,\Delta0[m]\times\Delta0[n]\times\Delta0[o]}{3\,d}-\frac{8\,i\,(q1+q2).(q1+q2)\,\Delta0[i]\times\Delta0[j]\,\Delta0[k]^3\,\Delta0[m]\times\Delta0[n]\times\Delta0[o]}{3\,d}+\frac{2}{3}\,i\,\Delta0[i]\times\Delta0[j]\times\Delta0[k]\,\Delta0[m]^2\,\Delta0[n]\times\Delta0[o]+$$
$$8\,i\,q1.(-q1+q3)\,\Delta0[i]^2\,\Delta0[j]\times\Delta0[k]\,\Delta0[m]^2\,\Delta0[n]\times\Delta0[o]\;+\;8\,i\,(q1+q2).(-q1+q3)\,\Delta0[i]\times\Delta0[j]\,\Delta0[k]^2\,\Delta0[m]^2\,\Delta0[n]\times\Delta0[o]\;-\;8\,i\,(-q1+q3).(-q1+q3)\,\Delta0[i]\times\Delta0[j]\times\Delta0[k]\,\Delta0[m]^3\,\Delta0[n]\times\Delta0[o]$$

Must still perform the (3 loop) momenta integrals

Future: maybe hard master integrals up to 3 loops

Result continues ... (14 terms in total)

Ongoing work: building a code to automatically compute super-“traces” beyond 1-loop

Future work: Obtain RGEs of the general EFT at higher loops

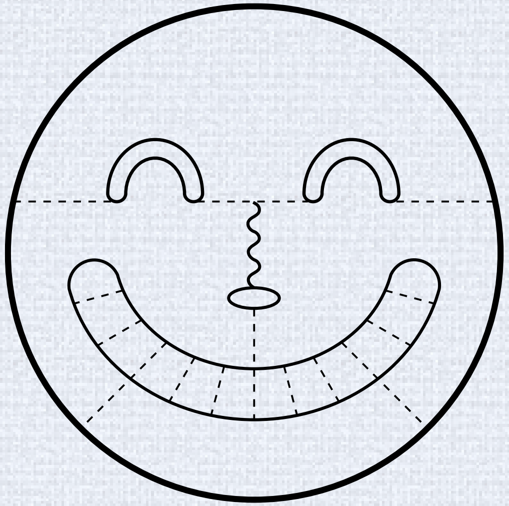
(see **José Santiago’s** talk)

Hypertracer?

Mathematicians might object: Beyond 1-loop there are no longer traces

SuperTensorContracter?

Not very catchy



FunGraphs

Thank you