# Some Hints in Numerical Higher-Order Computation

冯 锋
中国矿业大学（北京）

2018.12 高能物理研究中心

# Collaborators:

- Y. JIA （贾宇）
  高能物理研究所（IHEP）
- W.-L. SANG （桑文龙）
  西南大学 （SWU）

# Outline:

- General Procedures in Automated Higher-Order Computation
- Automated Computation involving Wilson Lines
- Some Hints in Sector Decomposition
- Some Problems

# Automated
# Higher-Order Computation

# Basic Procedures:

1. Feynman Diagrams & Amplitudes
2. Trace & Contraction
3. Partial Fragmentation & IBP Reduction
4. Master Integrals – Numerical
5. Other Processing – Plots etc.

# Basic Procedures:

1. Feynman Diagrams & Amplitudes

   - FeynArts - Mathematica
     **http://www.feynarts.de**
   - Qgraf - Fortran
     **http://cfif.ist.utl.pt/~paulo/qgraf.html**
   - FeynRules - Mathematica
     **http://feynrules.irmp.ucl.ac.be**
   - ...

# Basic Procedures:

1. Feynman Diagrams & Amplitudes

## Load Package

```
<< Qgraf`

LoadFeynRules["sm"];
```

## QgRun

```
HAmp = QgRun["
output='out';
style='math.sty';
model='sm.mod';
in=C[p1],Cbar[p2];
out=g[k1],g[k2];
loops=2;
loop_momentum=q;
options=notadpole,onshell;
true=vsum[QCD,6,6];
true=vsum[QED,0,0];
"];
```

```
-----------------------------------------------------------
                qgraf-3.1.4
-----------------------------------------------------------

output='out';
style='math.sty';
model='sm.mod';
in=C[p1],Cbar[p2];
out=g[k1],g[k2];
loops=2;
loop_momentum=q;
options=notadpole,onshell;
true=vsum[QCD,6,6];
true=vsum[QED,0,0];

-----------------------------------------------------------

24P --- 7+ 17- --- 5N+ 2C+ 17C-

169V --- 3^142 4^27

-----------------------------------------------------------

 -  4^3 --- 0 diagrams
3^2 4^2 --- 23 diagrams
3^4 4^1 --- 244 diagrams
3^6  -  --- 1122 diagrams

total = 1389 diagrams
```

# Basic Procedures:

1. Feynman Diagrams & Amplitudes

## Load Package

```
<< Qgraf`

LoadFeynRules["sm"];
```

## QgRun

```
HAmp = QgRun["
output='out';
style='math.sty';
model='sm.mod';
in=C[p1],Cbar[p2];
out=g[k1],g[k2];
loops=2;
loop_momentum=q;
options=notadpole,onshell;
true=vsum[QCD,6,6];
true=vsum[QED,0,0];
"];
```

```
-----------------------------------------------------------
                qgraf-3.1.4
-----------------------------------------------------------

output='out';
style='math.sty';
model='sm.mod';
in=C[p1],Cbar[p2];
out=g[k1],g[k2];
loops=2;
loop_momentum=q;
options=notadpole,onshell;
true=vsum[QCD,6,6];
true=vsum[QED,0,0];

-----------------------------------------------------------

24P --- 7+ 17- --- 5N+ 2C+ 17C-

169V --- 3^142 4^27

-----------------------------------------------------------

 -  4^3 --- 0 diagrams
3^2 4^2 --- 23 diagrams
3^4 4^1 --- 244 diagrams
3^6  -  --- 1122 diagrams

total = 1389 diagrams
```

# Basic Procedures:

1. Feynman Diagrams & Amplitudes

math.sty

math.sty 〉 No Selection

```
<prologue>
(* QGraf Output in Mathematica Style By F.Feng *)

<diagram>

Amp[[<diagram_index>]]:= Hold[[(<sign><symmetry_factor>)*
<in_loop>InField[[<field>,<field_index>,<momentum>]]*
<end>
<back><out_loop>OutField[[<field>,<field_index>,<momentum>]]*
<end>
<back><propagator_loop>Propagator[[
<back>Field[[<field>,<field_index>]],
<back>Field[[<dual-field>,<dual-field_index>]],
<back><momentum>]]*
<end>
<back><vertex_loop>Vertex[[
<back><ray_loop>Field[[<field>,<field_index>,<momentum>]],
<back><end><back>]]*
<end><back><back>]];

<epilogue>

<exit>
```

# Basic Procedures:

1. Feynman Diagrams & Amplitudes

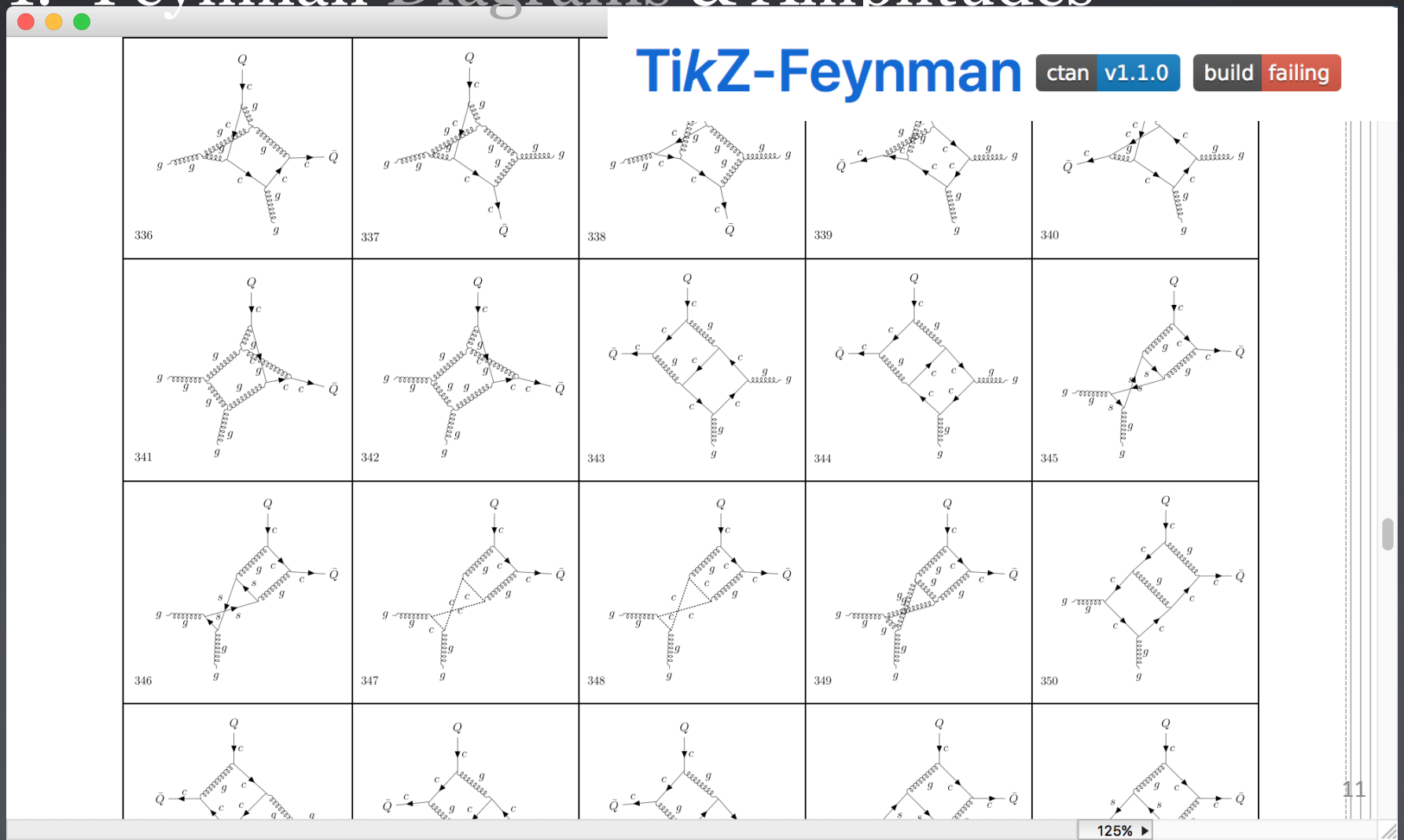
math.sty

math.sty 〉 No Selection

```
<prologue>
(* QGraf Output in Mathematica Style By F.Feng *)

<diagram>

Amp[[<diagram_index>]]:= Hold[[(<sign><symmetry_factor>)*
<in_loop>InField[[<field>,<field_index>,<momentum>]]*
<end>
<back><out_loop>OutField[[<field>,<field_index>,<momentum>]]*
<end>
<back><propagator_loop>Propagator[[
<back>Field[[<field>,<field_index>]],
<back>Field[[<dual-field>,<dual-field_index>]],
<back><momentum>]]*
<end>
<back><vertex_loop>Vertex[[
<back><ray_loop>Field[[<field>,<field_index>,<momentum>]],
<back><end><back>]]*
<end><back><back>]];

<epilogue>

<exit>
```

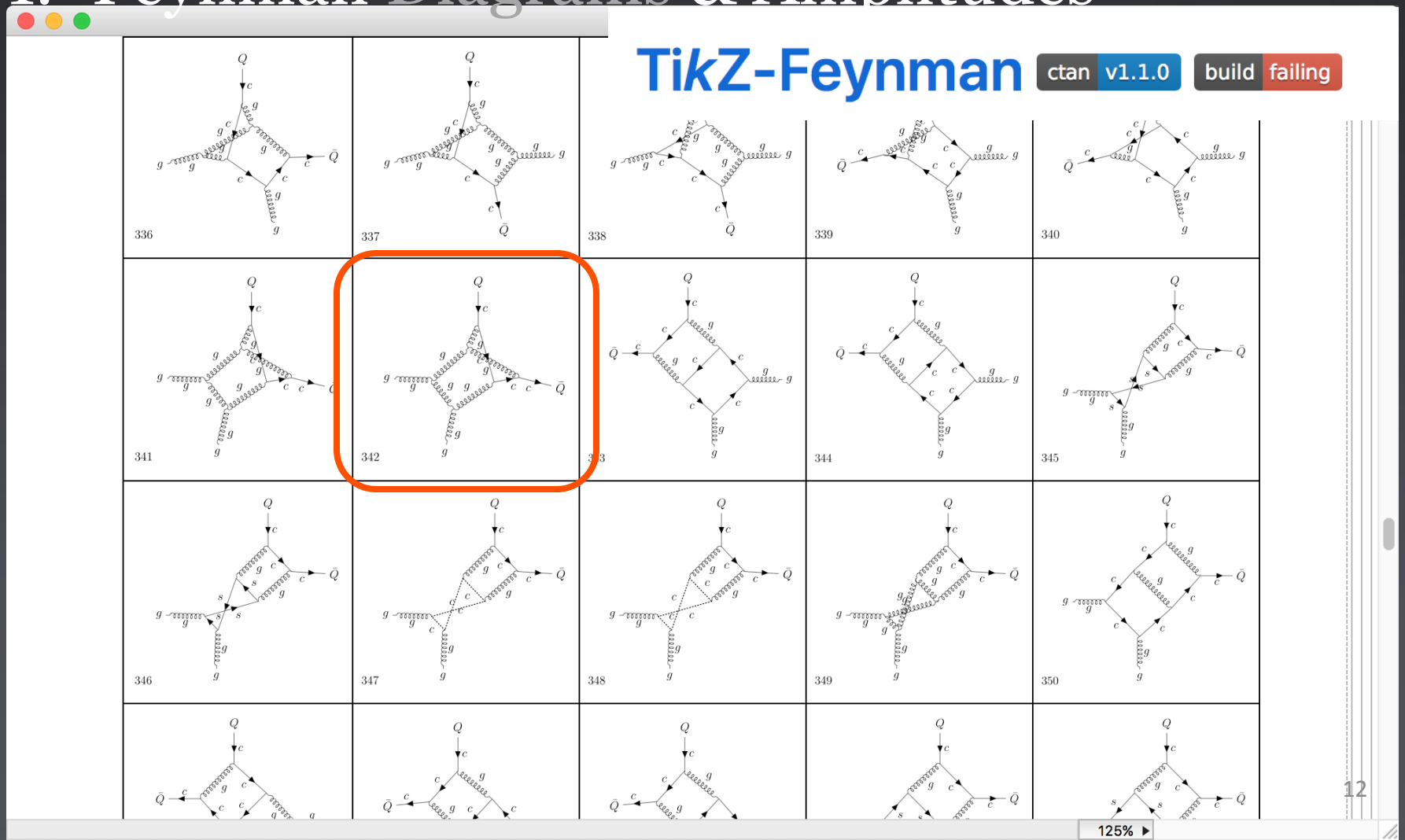
# Basic Procedures:

1. Feynman Diagrams & Amplitudes



# Basic Procedures:

1. Feynman Diagrams & Amplitudes



# Basic Procedures:

## 1. Feynman Diagrams & Amplitudes

```
HAmp[[342]]
```

```
Hold[+1 InField(C, -1, p1) InField(Cbar, -3, p2)
    OutField(g, -2, k1) OutField(g, -4, k2) Propagator(Field(g, 1), Field(g, 2), -q1)
    Propagator(Field(g, 7), Field(g, 8), -q2) Propagator(Field(g, 3), Field(g, 4), k1 + q1)
    Vertex(Field(g, -2, -k1), Field(g, 1, -q1), Field(g, 3, k1 + q1))
    Propagator(Field(g, 5), Field(g, 6), k2 - q1)
    Vertex(Field(g, -4, -k2), Field(g, 2, q1), Field(g, 5, k2 - q1))
    Propagator(Field(C, 9), Field(Cbar, 10), p1 - q2)
    Propagator(Field(g, 11), Field(g, 12), -k1 - q1 + q2)
    Vertex(Field(g, 4, -k1 - q1), Field(g, 8, q2), Field(g, 12, k1 + q1 - q2))
    Vertex(Field(Cbar, 10, q2 - p1), Field(C, -1, p1), Field(g, 7, -q2))
    Propagator(Field(C, 13), Field(Cbar, 14), k1 - p2 + q1 - q2)
    Vertex(Field(Cbar, -3, p2), Field(C, 13, k1 - p2 + q1 - q2), Field(g, 11, -k1 - q1 + q2))
    Vertex(Field(Cbar, 14, -k1 + p2 - q1 + q2), Field(C, 9, p1 - q2), Field(g, 6, q1 - k2))]
```

# Basic Procedures:

## 1. Feynman Diagrams & Amplitudes

### Propagator in SM

#### QCD

```
Propagator[Field[U,fi1_],Field[Ubar,fi2_],mom_]:=I MatDelta[TI[fi1],TI[fi2]] Mat[GSD[mom]+Mass[U],{DI[fi1],DI[fi2]}]/(SPD[mom]-Mass[U]^2);
Propagator[Field[D,fi1_],Field[Dbar,fi2_],mom_]:=I MatDelta[TI[fi1],TI[fi2]] Mat[GSD[mom]+Mass[D],{DI[fi1],DI[fi2]}]/(SPD[mom]-Mass[D]^2);
Propagator[Field[C,fi1_],Field[Cbar,fi2_],mom_]:=I MatDelta[TI[fi1],TI[fi2]] Mat[GSD[mom]+Mass[C],{DI[fi1],DI[fi2]}]/(SPD[mom]-Mass[C]^2);
Propagator[Field[S,fi1_],Field[Sbar,fi2_],mom_]:=I MatDelta[TI[fi1],TI[fi2]] Mat[GSD[mom]+Mass[S],{DI[fi1],DI[fi2]}]/(SPD[mom]-Mass[S]^2);
Propagator[Field[T,fi1_],Field[Tbar,fi2_],mom_]:=I MatDelta[TI[fi1],TI[fi2]] Mat[GSD[mom]+Mass[T],{DI[fi1],DI[fi2]}]/(SPD[mom]-Mass[T]^2);
Propagator[Field[B,fi1_],Field[Bbar,fi2_],mom_]:=I MatDelta[TI[fi1],TI[fi2]] Mat[GSD[mom]+Mass[B],{DI[fi1],DI[fi2]}]/(SPD[mom]-Mass[B]^2);
```

```
Propagator[Field[g,fi1_],Field[g,fi2_],mom_]:=-I SUNDelta[CI[fi1],CI[fi2]] (MTD[LI[fi1],LI[fi2]]/SPD[mom]-(1-ξ)FVD[mom,LI[fi1]] FVD[mom,LI[fi2]]/(SPD[mom]^2));
Propagator[Field[gh,fi1_],Field[ghbar,fi2_],mom_]:=I SUNDelta[CI[fi1],CI[fi2]]/(SPD[mom]);
```

#### EW

### Vertex in SM

#### QCD

```
Vertex[Field[Ubar,fi1_,mom1_],Field[U,fi2_,mom2_],Field[g,fi3_,mom3_]]:=I Gstrong Mat[GAD[LI[fi3]],{DI[fi1],DI[fi2]}] Mat[SUNT[CI[fi3]],{TI[fi1],TI[fi2]}];
Vertex[Field[Dbar,fi1_,mom1_],Field[D,fi2_,mom2_],Field[g,fi3_,mom3_]]:=I Gstrong Mat[GAD[LI[fi3]],{DI[fi1],DI[fi2]}] Mat[SUNT[CI[fi3]],{TI[fi1],TI[fi2]}];
Vertex[Field[Cbar,fi1_,mom1_],Field[C,fi2_,mom2_],Field[g,fi3_,mom3_]]:=I Gstrong Mat[GAD[LI[fi3]],{DI[fi1],DI[fi2]}] Mat[SUNT[CI[fi3]],{TI[fi1],TI[fi2]}];
Vertex[Field[Sbar,fi1_,mom1_],Field[S,fi2_,mom2_],Field[g,fi3_,mom3_]]:=I Gstrong Mat[GAD[LI[fi3]],{DI[fi1],DI[fi2]}] Mat[SUNT[CI[fi3]],{TI[fi1],TI[fi2]}];
Vertex[Field[Tbar,fi1_,mom1_],Field[T,fi2_,mom2_],Field[g,fi3_,mom3_]]:=I Gstrong Mat[GAD[LI[fi3]],{DI[fi1],DI[fi2]}] Mat[SUNT[CI[fi3]],{TI[fi1],TI[fi2]}];
Vertex[Field[Bbar,fi1_,mom1_],Field[B,fi2_,mom2_],Field[g,fi3_,mom3_]]:=I Gstrong Mat[GAD[LI[fi3]],{DI[fi1],DI[fi2]}] Mat[SUNT[CI[fi3]],{TI[fi1],TI[fi2]}];
Vertex[Field[g,fi1_,mom1_],Field[g,fi2_,mom2_],Field[g,fi3_,mom3_]]:=Gstrong SUNF[CI[fi1],CI[fi2],CI[fi3]](MTD[LI[fi1],LI[fi2]] FVD[mom1-mom2,LI[fi3]]+MTD[LI[fi2],LI
Vertex[Field[g,fi1_,mom1_],Field[g,fi2_,mom2_],Field[g,fi3_,mom3_],Field[g,fi4_,mom4_]]:=-I Gstrong^2 (SUNF4[CI[fi1],CI[fi2],CI[fi3],CI[fi4]](MTD[LI[fi1],LI[fi3]]MTD[
Vertex[Field[ghbar,fi1_,mom1_],Field[gh,fi2_,mom2_],Field[g,fi3_,mom3_]]:=-Gstrong SUNF[CI[fi1],CI[fi2],CI[fi3]] FVD[mom1,LI[fi3]];
```

# Basic Procedures:

1. Feynman Diagrams & Amplitudes

```
HAmp[[342]] // ReleaseHold
```

$$\Big(\delta_{\mathrm{CI}(1)\,\mathrm{CI}(2)}\,\delta_{\mathrm{CI}(3)\,\mathrm{CI}(4)}\,\delta_{\mathrm{CI}(5)\,\mathrm{CI}(6)}\,\delta_{\mathrm{CI}(7)\,\mathrm{CI}(8)}\,\delta_{\mathrm{CI}(11)\,\mathrm{CI}(12)}\,f_{\mathrm{CI}(-4)\,\mathrm{CI}(2)\,\mathrm{CI}(5)}\,f_{\mathrm{CI}(-2)\,\mathrm{CI}(1)\,\mathrm{CI}(3)}\,f_{\mathrm{CI}(4)\,\mathrm{CI}(8)\,\mathrm{CI}(12)}\,g^{\mathrm{LI}(1)\,\mathrm{LI}(2)}$$
$$g^{\mathrm{LI}(3)\,\mathrm{LI}(4)}\,g^{\mathrm{LI}(5)\,\mathrm{LI}(6)}\,g^{\mathrm{LI}(7)\,\mathrm{LI}(8)}\,g^{\mathrm{LI}(11)\,\mathrm{LI}(12)}\,g_s^6\,\mathrm{OutField}(g,-2,\mathrm{k1})\,\mathrm{OutField}(g,-4,\mathrm{k2})$$
$$\left(g^{\mathrm{LI}(-2)\,\mathrm{LI}(1)}\left(\mathrm{q1}-\mathrm{k1}^{\mathrm{LI}(3)}\right)+g^{\mathrm{LI}(1)\,\mathrm{LI}(3)}\left(-\mathrm{k1}-2\,\mathrm{q1}^{\mathrm{LI}(-2)}\right)+g^{\mathrm{LI}(3)\,\mathrm{LI}(-2)}\left(2\,\mathrm{k1}+\mathrm{q1}^{\mathrm{LI}(1)}\right)\right)$$
$$\left(g^{\mathrm{LI}(-4)\,\mathrm{LI}(2)}\left(-\mathrm{k2}-\mathrm{q1}^{\mathrm{LI}(5)}\right)+g^{\mathrm{LI}(2)\,\mathrm{LI}(5)}\left(2\,\mathrm{q1}-\mathrm{k2}^{\mathrm{LI}(-4)}\right)+g^{\mathrm{LI}(5)\,\mathrm{LI}(-4)}\left(2\,\mathrm{k2}-\mathrm{q1}^{\mathrm{LI}(2)}\right)\right)$$
$$\left(g^{\mathrm{LI}(4)\,\mathrm{LI}(8)}\left(-\mathrm{k1}-\mathrm{q1}-\mathrm{q2}^{\mathrm{LI}(12)}\right)+g^{\mathrm{LI}(8)\,\mathrm{LI}(12)}\left(-\mathrm{k1}-\mathrm{q1}+2\,\mathrm{q2}^{\mathrm{LI}(4)}\right)+\right.$$
$$\left.g^{\mathrm{LI}(12)\,\mathrm{LI}(4)}\left(2\,\mathrm{k1}+2\,\mathrm{q1}-\mathrm{q2}^{\mathrm{LI}(8)}\right)\right)\mathrm{ColorLine}\left(T_{\mathrm{CI}(11)}.T_{\mathrm{CI}(6)}.T_{\mathrm{CI}(7)},\{\mathrm{p2},\mathrm{p1}\}\right)\mathrm{SpinLine}\Big($$
$$\gamma^{\mathrm{LI}(11)}.(\mathrm{Mass}(C)+\gamma\cdot(\mathrm{k1}-\mathrm{p2}+\mathrm{q1}-\mathrm{q2})).\gamma^{\mathrm{LI}(6)}.(\mathrm{Mass}(C)+\gamma\cdot(\mathrm{p1}-\mathrm{q2})).\gamma^{\mathrm{LI}(7)},\{\mathrm{p2},\mathrm{p1}\}\Big)\Big)\Big/$$
$$\left(\mathrm{q1}^2\,\mathrm{q2}^2\,(\mathrm{k1}+\mathrm{q1})^2\,(\mathrm{k2}-\mathrm{q1})^2\,(-\mathrm{k1}-\mathrm{q1}+\mathrm{q2})^2\left((\mathrm{p1}-\mathrm{q2})^2-\mathrm{Mass}(C)^2\right)\right.$$
$$\left.\left((\mathrm{k1}-\mathrm{p2}+\mathrm{q1}-\mathrm{q2})^2-\mathrm{Mass}(C)^2\right)\right)$$

# Basic Procedures:

2. Trace & Contraction

- FeynCalc - Mathematica Package
  **https://github.com/FeynCalc**
- Form - C Program (tForm, parForm)
  **https://github.com/vermaseren/form**
- FeynCalc/FormLink - Combined
  **https://github.com/FormLink**
- ...

# Basic Procedures:

## 2. Trace & Contraction

Untitled-1

```
In[1]:= << FeynCalcFormLink`

In[8]:= exp = DiracTrace[GAD[μ].(GSD[p1 + p2] + m).GAD[ν].(GSD[k1 + k2] + m) GSD[p1, p2, k1, k2]]
          (FVD[p1 + k1, μ] FVD[p2 + k2, ν] + MTD[μ, ν])
```

Out[8]= $(g^{\mu\,\nu} + (\mathrm{k1} + \mathrm{p1}^{\mu})\,(\mathrm{k2} + \mathrm{p2}^{\nu}))\,\mathrm{tr}((\gamma\cdot\mathrm{p1}).(\gamma\cdot\mathrm{p2}).(\gamma\cdot\mathrm{k1}).(\gamma\cdot\mathrm{k2})\,\gamma^{\mu}.(m + \gamma\cdot(\mathrm{p1} + \mathrm{p2})).\gamma^{\nu}.(\gamma\cdot(\mathrm{k1} + \mathrm{k2}) + m))$

```
In[9]:= FeynCalcFormLink[exp] // Factor
```

```
Symbols D,m;
Dimension D;
Vectors k1,k2,p1,p2;
AutoDeclare Index lor;
Format Mathematica;
AutoDeclare Symbol str;
L ltr1 =
(g_(1,lor1)*(m*gi_(1)+g_(1,p1)+g_(1,p2))*g_(1,lor2)*(m*gi_(1)+g_(1,k1)+g_(1,k2))*g_(1,p1)*g_(1,p2)*g_(1,k1)*g
_(1,k2));
tracen,1;
.sort;
L resFL = (str1*(d_(lor1,lor2)+(k1(lor1)+p1(lor1))*(k2(lor2)+p2(lor2))));
id str1=ltr1;
.sort;
contract 0;
.sort;
#call put("%E", resFL)
.end
```



125%

# Basic Procedures:

## 2. Trace & Contraction

Untitled-1

Piping the script to FORM and running FORM

Time needed by FORM : 0.006 seconds. FORM finished. Got the result back to Mathematica as a string.

Start translation to Mathematica / FeynCalc syntax

Total wall clock time used: 0.15 seconds. Translation to Mathematica and FeynCalc finished.

Out[9]=

$$
\begin{aligned}
&4\big(2\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}\cdot\mathrm{p1}^2\,m^2 + 2\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{p1}\cdot\mathrm{p2}^2\,m^2 + D\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}\cdot\mathrm{p1}\,m^2 + 2\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}\cdot\mathrm{p1}\,m^2 - D\,\mathrm{k1}\cdot\mathrm{p1}\,\mathrm{k2}\cdot\mathrm{p2}\,m^2 - 2\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{k1}\cdot\mathrm{p1}\,\mathrm{k2}\cdot\mathrm{p2}\,m^2 - \\
&2\,\mathrm{k1}\cdot\mathrm{p1}\,\mathrm{k2}\cdot\mathrm{p1}\,\mathrm{k2}\cdot\mathrm{p2}\,m^2 - \mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}^2\,\mathrm{p1}^2\,m^2 + 2\,\mathrm{k1}\cdot\mathrm{k2}^2\,\mathrm{p1}\cdot\mathrm{p2}\,m^2 + D\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{p1}\cdot\mathrm{p2}\,m^2 + 4\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{p1}\cdot\mathrm{p2}\,m^2 - \mathrm{k1}^2\,\mathrm{k2}^2\,\mathrm{p1}\cdot\mathrm{p2}\,m^2 + \\
&2\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{k2}\cdot\mathrm{p1}\,\mathrm{p1}\cdot\mathrm{p2}\,m^2 + 2\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}\cdot\mathrm{p1}\,\mathrm{p1}\cdot\mathrm{p2}\,m^2 - 2\,\mathrm{k1}^2\,\mathrm{k2}\cdot\mathrm{p2}\,\mathrm{p1}\cdot\mathrm{p2}\,m^2 - 2\,\mathrm{k1}\cdot\mathrm{p1}\,\mathrm{k2}\cdot\mathrm{p2}\,\mathrm{p1}\cdot\mathrm{p2}\,m^2 - 2\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{k1}\cdot\mathrm{p1}\,\mathrm{p2}^2\,m^2 + \\
&\mathrm{k1}^2\,\mathrm{k2}\cdot\mathrm{p1}\,\mathrm{p2}^2\,m^2 - \mathrm{k1}\cdot\mathrm{k2}\,\mathrm{p1}^2\,\mathrm{p2}^2\,m^2 - 4\,\mathrm{k1}\cdot\mathrm{p2}^2\,\mathrm{k2}\cdot\mathrm{p1}^2 - 2\,D\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}\cdot\mathrm{p1}^2 + 2\,\mathrm{k1}^2\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}\cdot\mathrm{p1}^2 + 4\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}\cdot\mathrm{p1}^2 - 4\,\mathrm{k1}\cdot\mathrm{p1}^2\,\mathrm{k2}\cdot\mathrm{p2}^2 + \\
&2\,D\,\mathrm{k1}\cdot\mathrm{p1}\,\mathrm{k2}\cdot\mathrm{p2}^2 - 2\,\mathrm{k1}^2\,\mathrm{k1}\cdot\mathrm{p1}\,\mathrm{k2}\cdot\mathrm{p2}^2 - 4\,\mathrm{k1}\cdot\mathrm{p1}\,\mathrm{k2}\cdot\mathrm{p2}^2 + 4\,\mathrm{k1}\cdot\mathrm{k2}^2\,\mathrm{p1}\cdot\mathrm{p2}^2 + 2\,\mathrm{k1}^2\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{p1}\cdot\mathrm{p2}^2 + 2\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{k2}^2\,\mathrm{p1}\cdot\mathrm{p2}^2 - \\
&2\,D\,\mathrm{k1}\cdot\mathrm{p2}^2\,\mathrm{k2}\cdot\mathrm{p1} - 4\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{k1}\cdot\mathrm{p2}^2\,\mathrm{k2}\cdot\mathrm{p1} + 4\,\mathrm{k1}\cdot\mathrm{p2}^2\,\mathrm{k2}\cdot\mathrm{p1} - 2\,D\,\mathrm{k1}\cdot\mathrm{p1}\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}\cdot\mathrm{p1} - 4\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{k1}\cdot\mathrm{p1}\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}\cdot\mathrm{p1} + 4\,\mathrm{k1}\cdot\mathrm{p1}\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}\cdot\mathrm{p1} + \\
&2\,D\,\mathrm{k1}\cdot\mathrm{p1}^2\,\mathrm{k2}\cdot\mathrm{p2} + 4\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{k1}\cdot\mathrm{p1}^2\,\mathrm{k2}\cdot\mathrm{p2} - 4\,\mathrm{k1}\cdot\mathrm{p1}^2\,\mathrm{k2}\cdot\mathrm{p2} + 2\,D\,\mathrm{k1}\cdot\mathrm{p1}\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}\cdot\mathrm{p2} + 4\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{k1}\cdot\mathrm{p1}\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}\cdot\mathrm{p2} - \\
&4\,\mathrm{k1}\cdot\mathrm{p1}\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}\cdot\mathrm{p2} + 2\,D\,\mathrm{k1}\cdot\mathrm{p1}\,\mathrm{k2}\cdot\mathrm{p1}\,\mathrm{k2}\cdot\mathrm{p2} - 2\,\mathrm{k1}^2\,\mathrm{k1}\cdot\mathrm{p1}\,\mathrm{k2}\cdot\mathrm{p1}\,\mathrm{k2}\cdot\mathrm{p2} - 4\,\mathrm{k1}\cdot\mathrm{p1}\,\mathrm{k2}\cdot\mathrm{p1}\,\mathrm{k2}\cdot\mathrm{p2} - 2\,D\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}\cdot\mathrm{p1}\,\mathrm{k2}\cdot\mathrm{p2} + \\
&2\,\mathrm{k1}^2\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}\cdot\mathrm{p1}\,\mathrm{k2}\cdot\mathrm{p2} + 8\,\mathrm{k1}\cdot\mathrm{p1}\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}\cdot\mathrm{p1}\,\mathrm{k2}\cdot\mathrm{p2} + 4\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}\cdot\mathrm{p1}\,\mathrm{k2}\cdot\mathrm{p2} + 4\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{k1}\cdot\mathrm{p2}^2\,\mathrm{p1}^2 - 2\,\mathrm{k1}\cdot\mathrm{p1}\,\mathrm{k2}\cdot\mathrm{p2}^2\,\mathrm{p1}^2 + \\
&4\,\mathrm{k1}\cdot\mathrm{k2}^2\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{p1}^2 + 2\,D\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{p1}^2 - 4\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{p1}^2 + 2\,\mathrm{k1}\cdot\mathrm{p2}^2\,\mathrm{k2}^2\,\mathrm{p1}^2 + D\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}^2\,\mathrm{p1}^2 - \mathrm{k1}^2\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}^2\,\mathrm{p1}^2 + \\
&2\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}^2\,\mathrm{p1}^2 - 2\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}^2\,\mathrm{p1}^2 + 2\,\mathrm{k1}\cdot\mathrm{p2}^2\,\mathrm{k2}\cdot\mathrm{p1}\,\mathrm{p1}^2 + \mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}^2\,\mathrm{k2}\cdot\mathrm{p1}\,\mathrm{p1}^2 - D\,\mathrm{k1}^2\,\mathrm{k2}\cdot\mathrm{p2}\,\mathrm{p1}^2 + 2\,\mathrm{k1}^2\,\mathrm{k2}\cdot\mathrm{p2}\,\mathrm{p1}^2 - \\
&2\,\mathrm{k1}^2\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{k2}\cdot\mathrm{p2}\,\mathrm{p1}^2 - 2\,\mathrm{k1}^2\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}\cdot\mathrm{p2}\,\mathrm{p1}^2 - 2\,\mathrm{k1}\cdot\mathrm{p1}\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}\cdot\mathrm{p2}\,\mathrm{p1}^2 - \mathrm{k1}^2\,\mathrm{k2}^2\,\mathrm{k2}\cdot\mathrm{p2}\,\mathrm{p1}^2 - \mathrm{k1}\cdot\mathrm{p1}\,\mathrm{k2}^2\,\mathrm{k2}\cdot\mathrm{p2}\,\mathrm{p1}^2 + \\
&2\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}\cdot\mathrm{p1}\,\mathrm{k2}\cdot\mathrm{p2}\,\mathrm{p1}^2 - 4\,\mathrm{k1}\cdot\mathrm{k2}^2\,\mathrm{k1}\cdot\mathrm{p1}\,\mathrm{p1}\cdot\mathrm{p2} - 2\,D\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{k1}\cdot\mathrm{p1}\,\mathrm{p1}\cdot\mathrm{p2} + 4\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{k1}\cdot\mathrm{p1}\,\mathrm{p1}\cdot\mathrm{p2} + 4\,\mathrm{k1}\cdot\mathrm{k2}^2\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{p1}\cdot\mathrm{p2} + \\
&2\,D\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{p1}\cdot\mathrm{p2} - 4\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{p1}\cdot\mathrm{p2} - 8\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{k1}\cdot\mathrm{p1}\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{p1}\cdot\mathrm{p2} + 2\,\mathrm{k1}^2\,\mathrm{k1}\cdot\mathrm{p1}\,\mathrm{k2}^2\,\mathrm{p1}\cdot\mathrm{p2} + 2\,D\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}^2\,\mathrm{p1}\cdot\mathrm{p2} + \\
&4\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}^2\,\mathrm{p1}\cdot\mathrm{p2} - 4\,\mathrm{k1}\cdot\mathrm{p1}\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}^2\,\mathrm{p1}\cdot\mathrm{p2} - 4\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}^2\,\mathrm{p1}\cdot\mathrm{p2} - 2\,D\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{k2}\cdot\mathrm{p1}\,\mathrm{p1}\cdot\mathrm{p2} + 2\,\mathrm{k1}^2\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{k2}\cdot\mathrm{p1}\,\mathrm{p1}\cdot\mathrm{p2} + \\
&4\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{k2}\cdot\mathrm{p1}\,\mathrm{p1}\cdot\mathrm{p2} + 2\,\mathrm{k1}^2\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}\cdot\mathrm{p1}\,\mathrm{p1}\cdot\mathrm{p2} + 2\,\mathrm{k1}\cdot\mathrm{p2}\,\mathrm{k2}^2\,\mathrm{k2}\cdot\mathrm{p1}\,\mathrm{p1}\cdot\mathrm{p2} - 2\,D\,\mathrm{k1}^2\,\mathrm{k2}\cdot\mathrm{p2}\,\mathrm{p1}\cdot\mathrm{p2} + 4\,\mathrm{k1}^2\,\mathrm{k2}\cdot\mathrm{p2}\,\mathrm{p1}\cdot\mathrm{p2} - \\
&2\,D\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{k2}\cdot\mathrm{p2}\,\mathrm{p1}\cdot\mathrm{p2} - 2\,\mathrm{k1}^2\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{k2}\cdot\mathrm{p2}\,\mathrm{p1}\cdot\mathrm{p2} + 4\,\mathrm{k1}\cdot\mathrm{k2}\,\mathrm{k2}\cdot\mathrm{p2}\,\mathrm{p1}\cdot\mathrm{p2} + 2\,\mathrm{k1}^2\,\mathrm{k1}\cdot\mathrm{p1}\,\mathrm{k2}\cdot\mathrm{p2}\,\mathrm{p1}\cdot\mathrm{p2} - 2\,\mathrm{k1}^2\,\mathrm{k2}^2\,\mathrm{k2}\cdot\mathrm{p2}\,\mathrm{p1}\cdot\mathrm{p2} -
\end{aligned}
$$

125%

# Basic Procedures:

3. Partial Fragmentation & IBP Reduction

- APART - MATHEMATICA
  **https://github.com/F-Feng**
  FIRE - MATHEMATICA Program & C++
  **http://science.sander.su**

- AIR - MAPLE Program
  **https://www.phys.ethz.ch/~pheno/air/**

- REDUZE - C (MPI supported)
  **https://reduze.hepforge.org**

- ...

# Basic Procedures:

## 3. Partial Fragmentation & IBP Reduction

JHEP **0810**, 107 (2008)

- Integrate By Part (IBP)

$$F(a_1, \cdots, a_n) = \int \cdots \int \frac{d^d k_1 \cdots d^d k_h}{E_1^{a_1} \cdots E_n^{a_n}}$$

where $k_i$, $i = 1, \cdots, h$, are loop momenta and the denominators $E_r$ are either quadratic or linear with respect to the loop momenta $k_i$ of the graph. Irreducible polynomials in the numerator can be represented as denominators raised to negative powers.

- Basic idea:

$$\int \cdots \int d^d k_1 d^d k_2 \cdots \frac{\partial}{\partial k_i} \left[ \frac{p_j}{E_1^{a_1} \cdots E_n^{a_n}} \right] = 0$$

- List of equations:

$$\sum \alpha_i F(a_1 + b_{i,1}, \cdots, a_n + b_{i,n}) = 0$$

# Basic Procedures:

## 3. Partial Fragmentation & IBP Reduction

Apart function in MATHEMAITCA

In[1]:= `Apart[1/((x - a) (x - b) (x - c))]`

Out[1]= $$-\frac{1}{(a-b)(b-c)(x-b)}-\frac{1}{(a-c)(c-b)(x-c)}+\frac{1}{(a-b)(a-c)(x-a)}$$

In[2]:= `Apart[1/((x + a) (y + 2 b) (3 x + 4 y))]`

Out[2]= $$\frac{1}{(a+x)(3x-8b)(2b+y)}-\frac{4}{(a+x)(3x-8b)(3x+4y)}$$

# Basic Procedures:

## 3. Partial Fragmentation & IBP Reduction

APART Package:

```
In[1]:= << CalcExt`Apart`
```

```
In[2]:= $Apart[1/((x - a) (x - b) (x - c)), {x}]
```

Out[2]= $$-\frac{\left\|\frac{1}{a-x}\right\|}{(a-b)(a-c)}+\frac{\left\|\frac{1}{b-x}\right\|}{(a-b)(b-c)}+\frac{\left\|\frac{1}{c-x}\right\|}{(a-c)(c-b)}$$

```
In[3]:= $Apart[1/((x + a) (y + 2 b) (3 x + 4 y)), {x, y}]
```

Out[3]= $$-\frac{\left\|\frac{1}{(a+x)(2b+y)}\right\|}{3a+8b}+\frac{4\left\|\frac{1}{(a+x)(3x+4y)}\right\|}{3a+8b}+\frac{3\left\|\frac{1}{(2b+y)(3x+4y)}\right\|}{3a+8b}$$

# Basic Procedures:

## 3. Partial Fragmentation & IBP Reduction

$$F[l,m,n] = \int \frac{d^4k}{(2\pi)^4} \frac{(k\cdot p_2)^{-l}}{(m^2 - k^2 - 2k\cdot p_1 - p_1^2)^m(-m^2 + k^2 + 2k\cdot p_2 + p_2^2)^n}$$

```
In[1]:= << HighEnergyPhysics`fc`

In[2]:= << FIRE`

In[3]:= Replacement = {p1^2 → m^2, p2^2 → m^2, p1 p2 → SP[p1, p2]};
        Internal = {k};
        External = {p1, p2};
        Propagators = {k p2, -2 k p1 - k^2 + m^2 - p1^2, 2 k p2 + k^2 - m^2 + p2^2};
        PrepareIBP[];
        startinglist = {IBP[k, k], IBP[k, p1], IBP[k, p2]} /. Replacement;
        Prepare[];

In[10]:= Burn[];

In[11]:= F[{-1, 1, 2}]
```

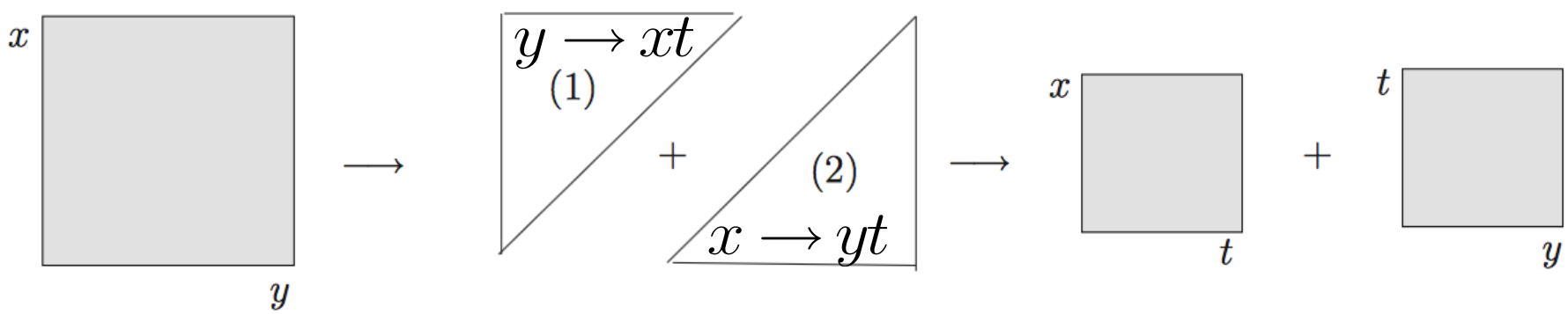
Out[11]= $\frac{(d-2)\,G(\{0,0,1\})}{8\,(m^2 - \mathrm{p1}\cdot\mathrm{p2})} + \frac{(d-2)\,G(\{0,1,0\})}{8\,(m^2 - \mathrm{p1}\cdot\mathrm{p2})} + \frac{1}{4}\,(4-d)\,G(\{0,1,1\})$

# Basic Procedures:

## 4. Master Integrals - Numerical

Sector Decomposition

$$
\begin{aligned}
I &= \int_0^1 dx \int_0^1 dy\, x^{-1-a\varepsilon} y^{-b\varepsilon} (x + (1-x)y)^{-1} \\
&= \int_0^1 dx \int_0^1 dy\, x^{-1-a\varepsilon} y^{-b\varepsilon} (x + (1-x)y)^{-1} \Big[\theta(x-y) + \theta(y-x)\Big]
\end{aligned}
$$



# Basic Procedures:

4. Master Integrals - Numerical

Sector Decomposition

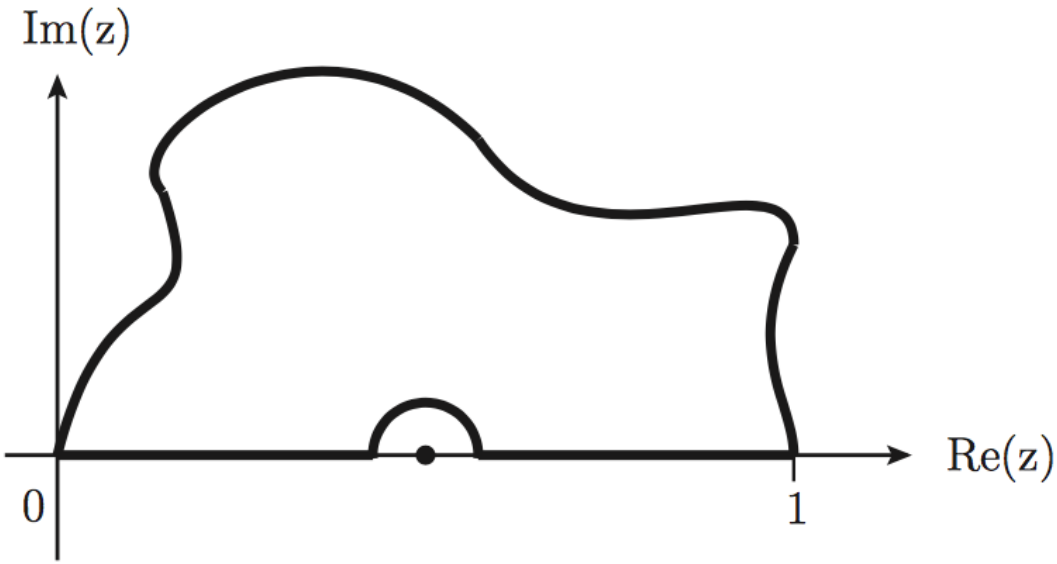
$$
\begin{aligned}
I &= \int_0^1 dx \int_0^1 dy\, x^{-1-a\varepsilon} y^{-b\varepsilon} (x + (1-x)y)^{-1} \\
&= \int_0^1 dx \int_0^1 dy\, x^{-1-a\varepsilon} y^{-b\varepsilon} (x + (1-x)y)^{-1} \Big[\theta(x-y) + \theta(y-x)\Big] \\
&= \int_0^1 dx x^{-1-(a+b)\varepsilon} \int_0^1 t^{-b\varepsilon} \Big(1 + (1-x)t\Big)^{-1} \\
&\quad + \int_0^1 dy y^{-1-(a+b)\varepsilon} \int_0^1 t^{-1-a\varepsilon} \Big(1 + (1-y)t\Big)^{-1}
\end{aligned}
$$

# Basic Procedures:

4. Master Integrals - Numerical

Sector Decomposition

*Deformation of the integration contour*



# Basic Procedures:

4. Master Integrals - Numerical

Sector Decomposition

$$\vec{z}(\vec{x}) = \vec{x} - i\,\vec{\tau}(\vec{x})$$

$$\tau_k = \lambda\, x_k(1 - x_k)\,\frac{\partial \mathcal{F}(\vec{x})}{\partial x_k}\,.$$

$$\mathcal{F}(\vec{z}(\vec{x})) = \mathcal{F}(\vec{x}) - i\,\lambda \sum_j x_j(1 - x_j)\left(\frac{\partial \mathcal{F}}{\partial x_j}\right)^2 + \mathcal{O}(\lambda^2)$$

# Basic Procedures:

4. Master Integrals - Numerical
   - FIESTA - MATHEMATICA
     **https://bitbucket.org/feynmanIntegrals/**
   - SECDEC - MATHEMATICA/PYTHON
     **http://secdec.hepforge.org**
   - SECTOR_DECOMPOSITION - C++
     **http://wwwthep.physik.uni-mainz.de/~stefanw/sector_decomposition/**
   - CSECTORS - MATHEMATCA/C++
     **http://prac.us.edu.pl/~gluza/csectors/**
   - ...

# Basic Procedures:

4. Master Integrals - Numerical

- Prepare Database
  Sector decomposion done with FIESTA

- Extract Sectors
  Expression in each sector exported to FORTRAN format

- Numerical Integration
  Run FORTRAN code using different numerical integration routines

- Combine Sector Results

# Basic Procedures:

4. Master Integrals - Numerical

- CubPack - Fortran

  **http://nines.cs.kuleuven.be/software/cubpack/**

- Cuba - C & Fortran

  **http://www.feynarts.de/cuba/**

- ParInt - C & Fortran

  **https://cs.wmich.edu/parint/**

- Hcubature - C & Fortran

  **https://github.com/stevengj/cubature**

- ...

# Basic Procedures:

4. Master Integrals - Numerical

Float Precision

- Long-Double Precision (18 $\sim$ 19 digits)
  **Fortran native:** `REAL(KIND=10)`

- Quadruple Precision (33 $\sim$ 36 digits)
  **Fortran native:** `REAL(KIND=16)`

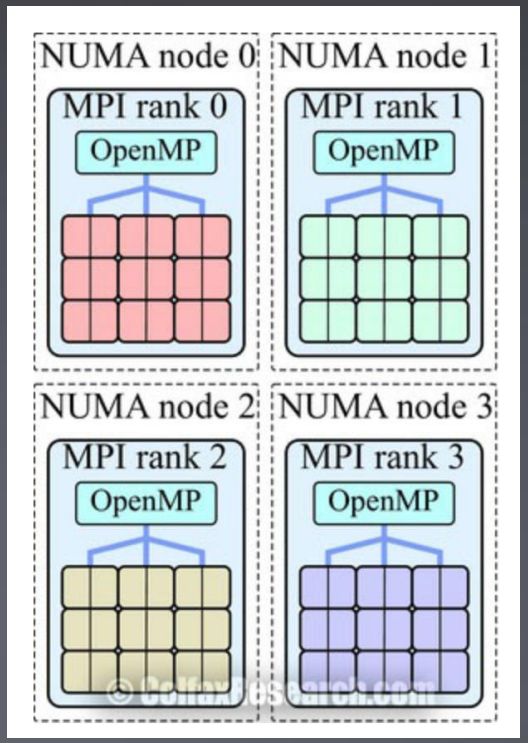
- Double-Quadruple Precision ($\sim$ 66 digits)
  **http://www.davidhbailey.com/dhbsoftware/**

- Arbitrarily High Numeric Precision
  **http://www.davidhbailey.com/dhbsoftware/**

# Basic Procedures:

4. Master Integrals - Numerical Parallelization

- Shared-Memory: OpenMP
- Distributed: Message Passing Interface(MPI)
- Hybrid OpenMP + MPI

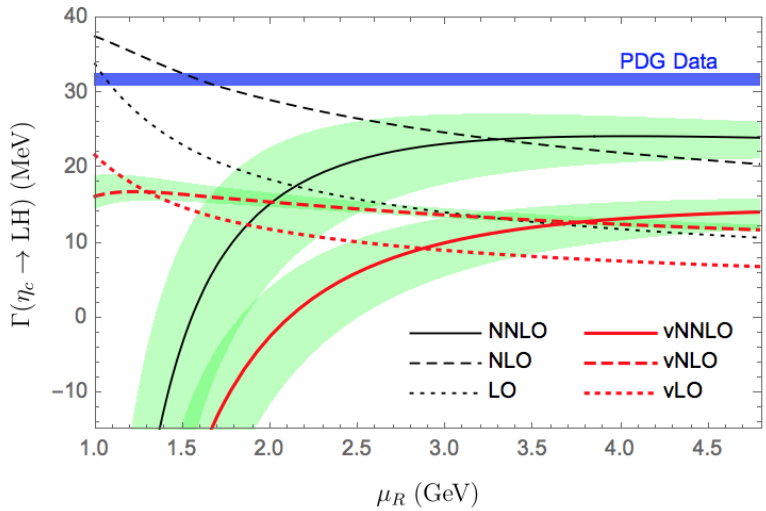


# Basic Procedures:

## 5. Other Processing – Plots etc.

Plots-v2.nb

```
  }]]
Export[FileNameJoin[{NotebookDirectory[], "TeX", "draft", "Gmc.eps"}], %];
```




### Bottom-Gamma

```
Show[bGammaPlot, Graphics[{
   Text[Style["PDG Data", Blue], {10.4, 17}, {-1, 0}],
   , {Thickness[0.003], Line[{{6.3, -5}, {7.3, -5}}]}, Text[Style["NNLO"], {7.5, -5}, {-1, 0}]
   , {Dashed, Thickness[0.003], Line[{{6.3, -10}, {7.3, -10}}]},
```



100%

# Basic Procedures:

1. Feynman Diagrams & Amplitudes
2. Trace & Contraction
3. Partial Fragmentation & IBP Reduction
4. Master Integrals – Numerical
5. Other Processing – Plots etc.

# Automated Computation involving Wilson Lines

$$\Phi(0, x^-, \mathbf{0}_\perp)_{ba} = \mathrm{P}\exp\left[ig_s \int_{x^-}^{\infty} dy^- n \cdot A(0^+, y^-, \mathbf{0}_\perp)\right]_{ba}$$

# Wilson Lines

- Feynman Rules for Gauge Links

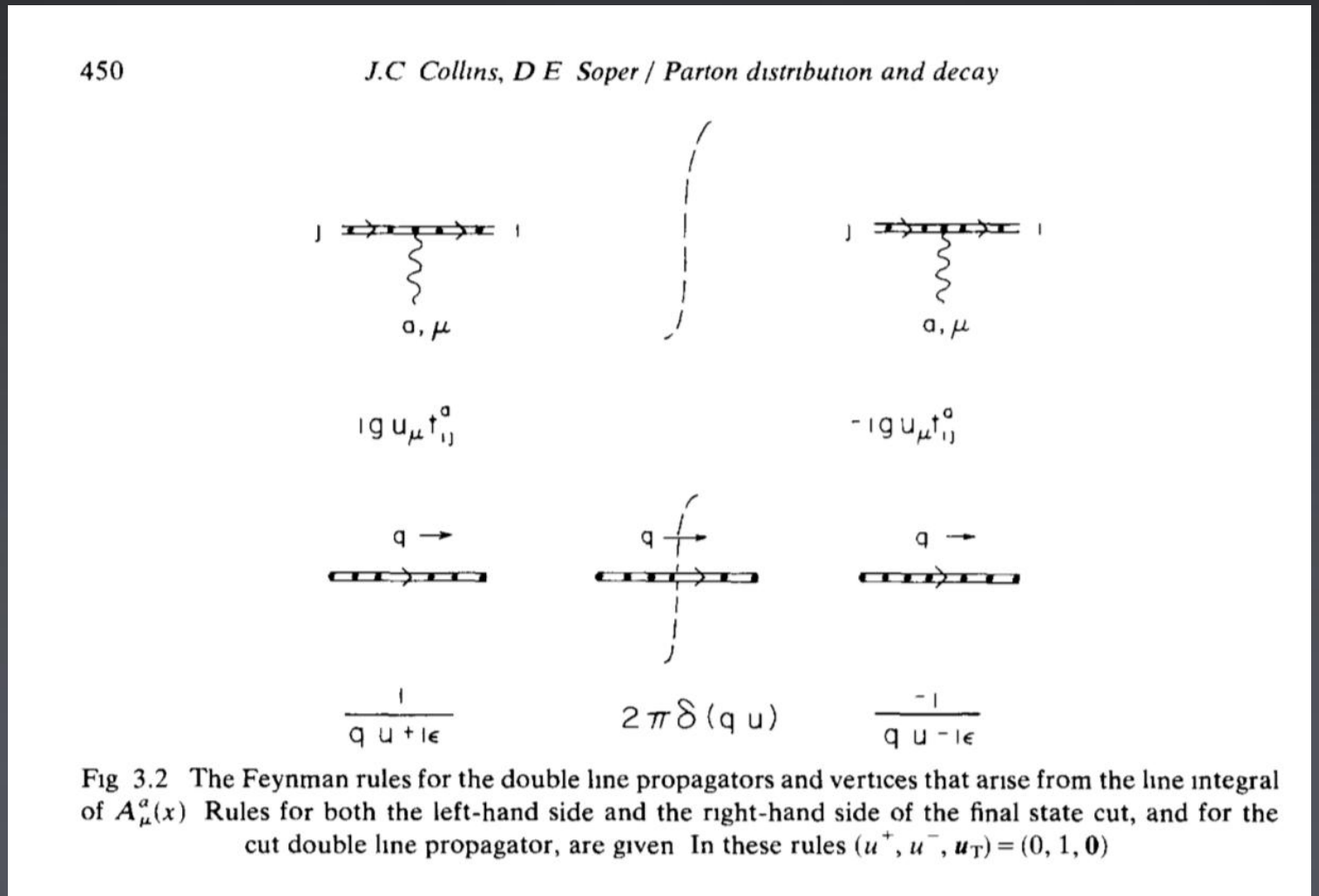



Fig 3.2 The Feynman rules for the double line propagators and vertices that arise from the line integral of $A^a_\mu(x)$ Rules for both the left-hand side and the right-hand side of the final state cut, and for the cut double line propagator, are given In these rules $(u^+, u^-, \boldsymbol{u}_T) = (0, 1, \mathbf{0})$

# Wilson Lines

- Feynm

given in fig. 3.2, here with the **8** representation of SU(3): $t^a_{bc} = -if_{abc}$. Except for a changed normalization and a reversal of the natural direction of momentum flow, these rules are the same as the rules for the $f(x)$'s.



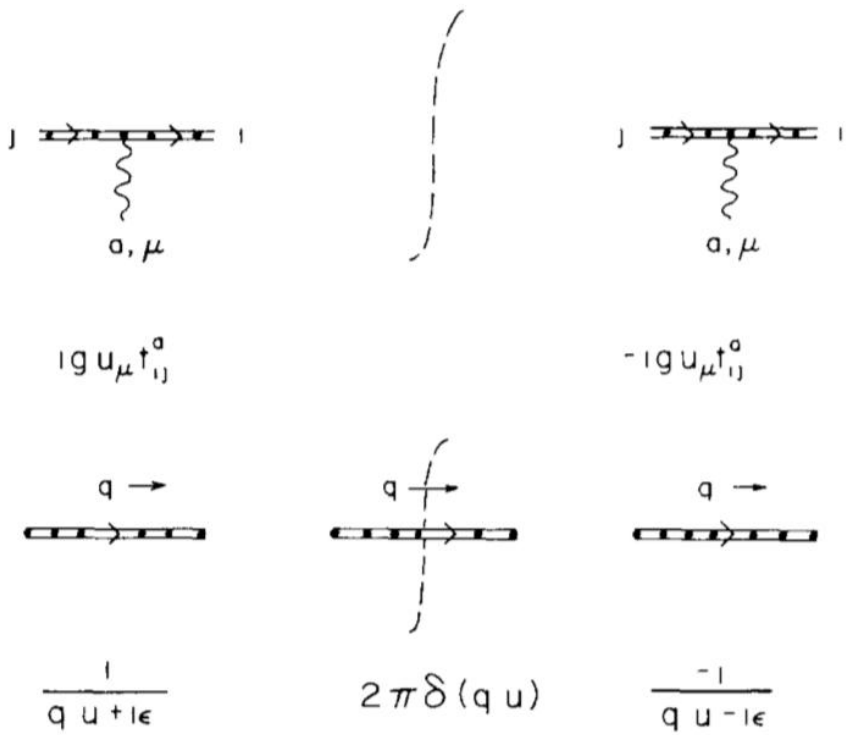
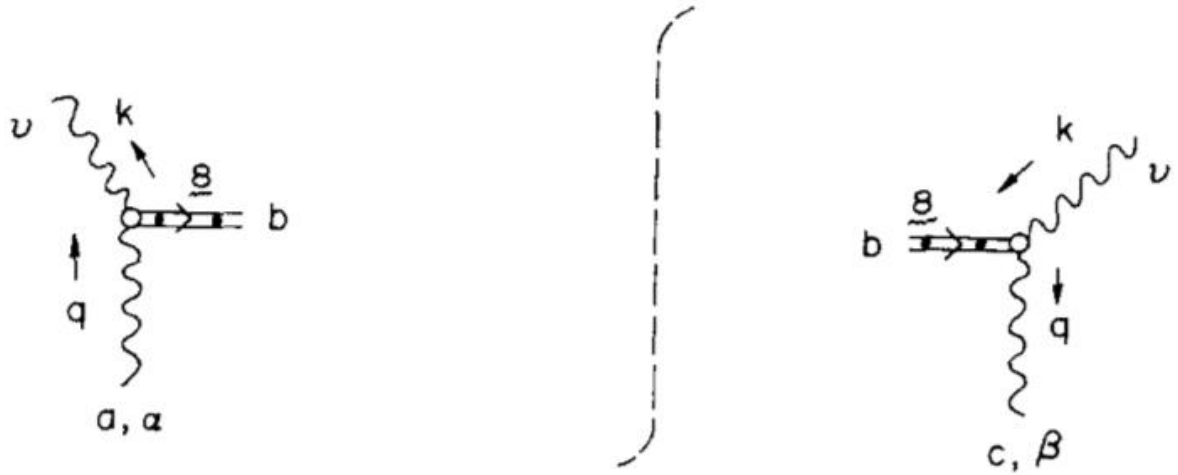



Fig 3.2 The Feynman rules for the double line propagators and vertices that arise from the line integral of $A^a_\mu(x)$ Rules for both the left-hand side and the right-hand side of the final state cut, and for the cut double line propagator, are given In these rules $(u^+, u^-, \boldsymbol{u}_T) = (0, 1, \mathbf{0})$

# Wilson Lines

- Feynman Rules for Composite Operator






$$i\left[k\,u\,g^{\nu}_{\alpha} - q^{\nu}\,u_{\alpha}\right]\delta_{ba} \qquad\qquad -i\left[k\,u\,g_{\beta\nu} - q_{\nu}\,u_{\beta}\right]\delta_{cb}$$

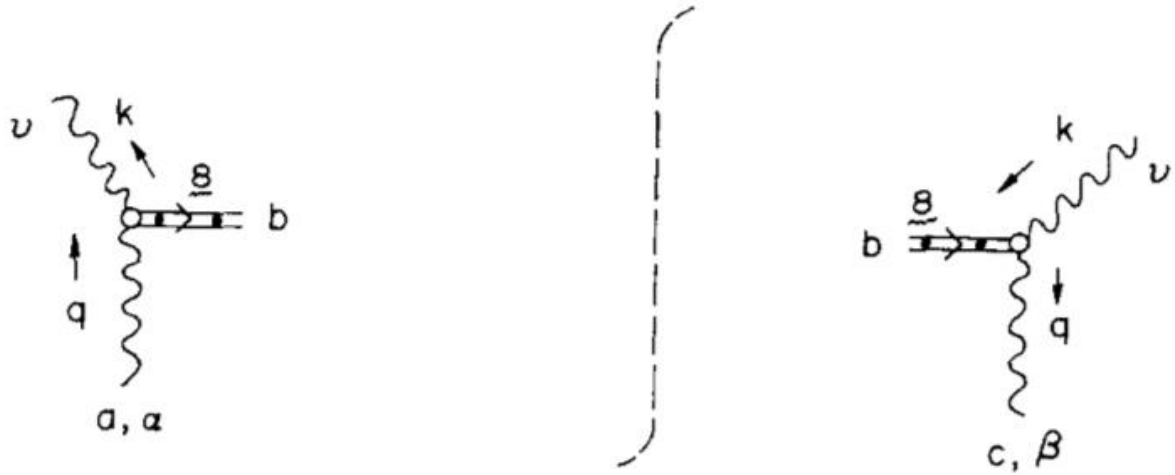
Fig 3 4 The vertex coupling the "observed" gluon to the double line propagator in fig 3 3

# Wilson Lines

- Feynm

given in fig. 3.2, here with the **8** representation of SU(3): $t^a_{bc} = -if_{abc}$. Except for a changed normalization and a reversal of the natural direction of momentum flow, these rules are the same as the rules for the $f(x)$'s.

*J C Collins, D E Soper / Parton distribution and decay* 451




$$i\left[k_\alpha g^\nu_\alpha - q^\nu k_\alpha\right]\delta_{ba} \qquad -i\left[k_\beta g_{\beta\nu} - q_\nu k_\beta\right]\delta_{cb}$$

Fig 3 4 The vertex coupling the "observed" gluon to the double line propagator in fig 3 3

# Model File for gluon FF

```
[ model = 'FF Model' ]
%-----------------------------------
% Propagators
%-----------------------------------
[q, qbar, -]
[Q, Qbar, -]

[gh, ghbar, -]
[g, g, +, notadpole]

[e, e, +, external]
[n, nbar, +]
%-----------------------------------
% Vertices
%-----------------------------------
[qbar, q, g; QCD='+1']
[Qbar, Q, g; QCD='+1']

[g, g, g, g; QCD='+2']
[g, g, g; QCD='+1']
[ghbar, gh, g; QCD='+1']

[nbar, n, g; QCD='+1']

[nbar, e, g; QCD='+0']
```

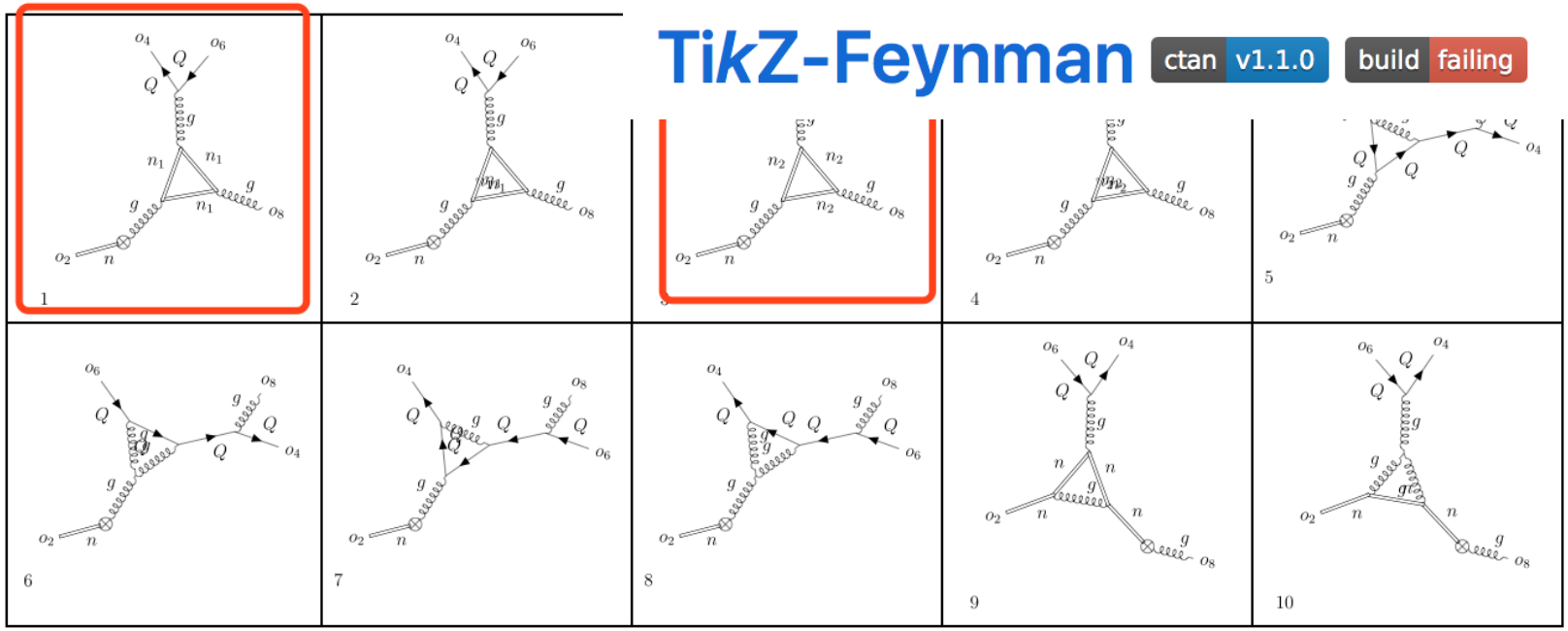
e is external only

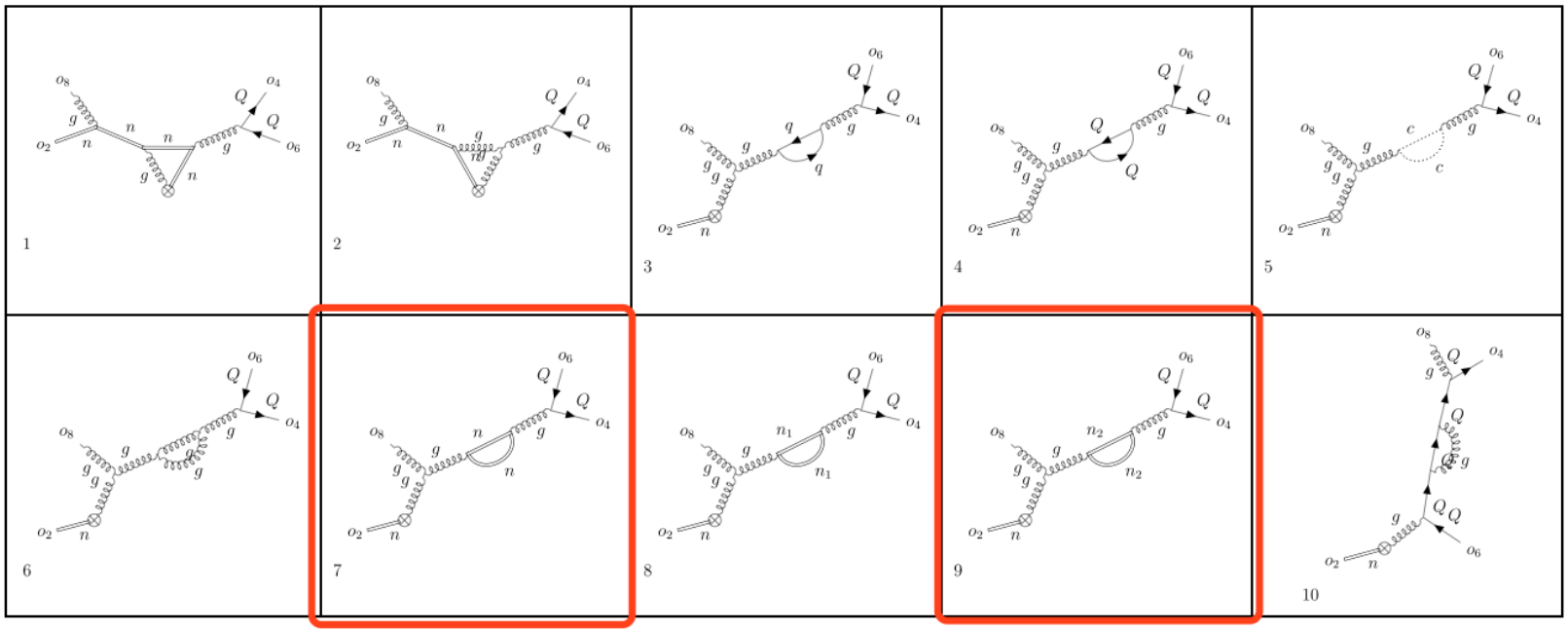
Composite Operator

# TikZ-Feynman

ctan v1.1.0 | build failing

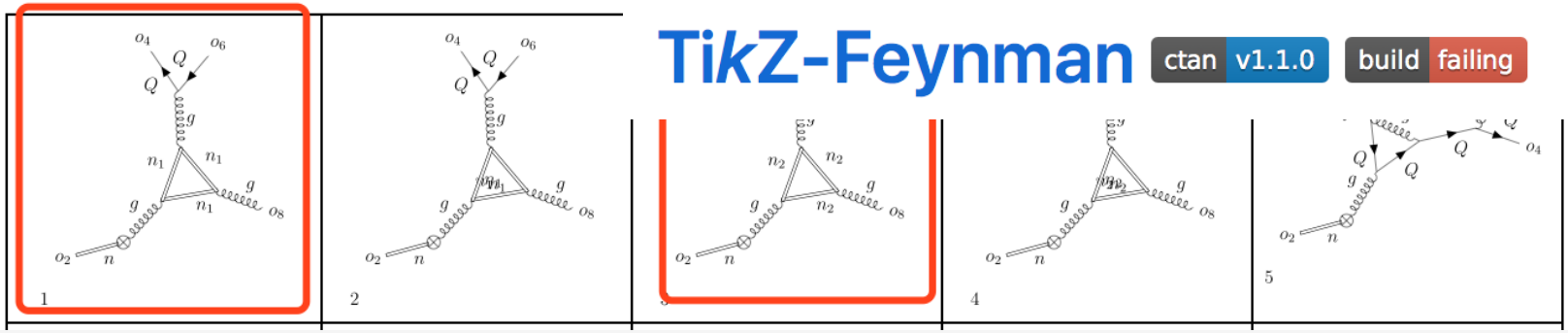
```
[ mo
%---
% P
%---
[q,
[Q,

[gh
[g,

[e,
[n,
%---
% Ve
%---
[qba
[Qba

[g,
[g,
[ghb

[nba

[nbar, e, g; QCD='+0']
```



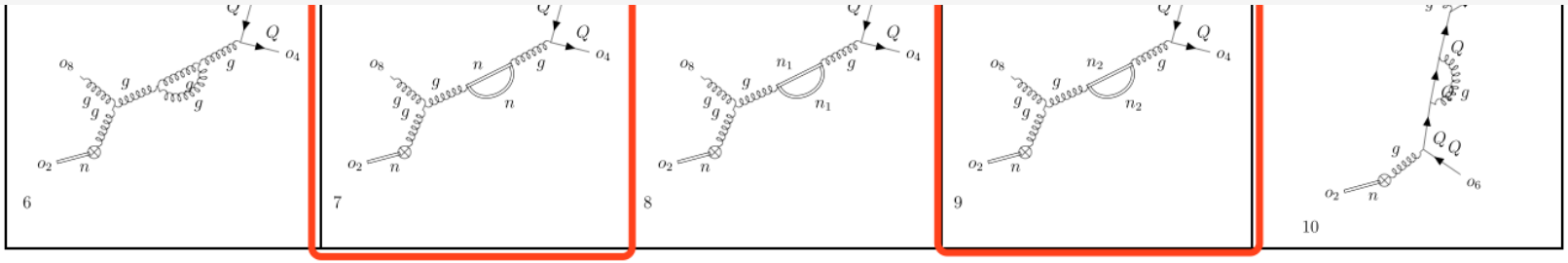


# TikZ-Feynman

ctan v1.1.0 | build failing



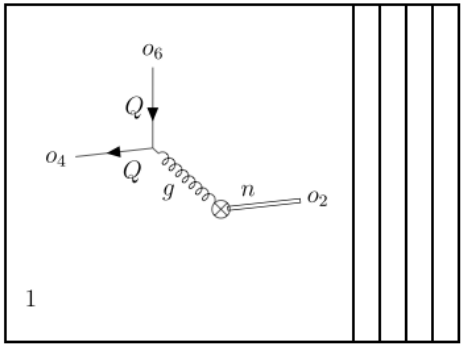
```
FreeEikonalLoopQ[Hamp_]:=Module[{fs,vs,ps,tmp,tmp1},
tmp=Hamp/.{Propagator->GPropagator,Vertex->GVertex};
vs=Cases[tmp,_GVertex,{0,Infinity}]//Union;
fs=Map[(tmp1=Cases[#,Field[ni:n|nbar|n1|n1bar|n2|n2bar,l1_,p_]:>Field[ni,l1]];Thread[tmp1-
fs=Flatten[fs];
ps=Cases[tmp,GPropagator[f1_,f2_,p_]:>GPropagator[f1,f2],{0,Infinity}]//Union;
ps=ps/.Field[ni:n|nbar|n1|n1bar|n2|n2bar,l1_,p_]:>Field[ni,l1]/.fs;
ps=Plus@@ps;
ps=ps//.GPropagator[x__,v_GVertex]+ GPropagator[v_GVertex,y__]:>GPropagator[x,v,y] ;
Return[FreeQ[ps,GPropagator[x_GVertex,___,x_GVertex]]];
];
RemoveEikonalLoop[amp_List]:=Select[amp,FreeEikonalLoopQ];
```



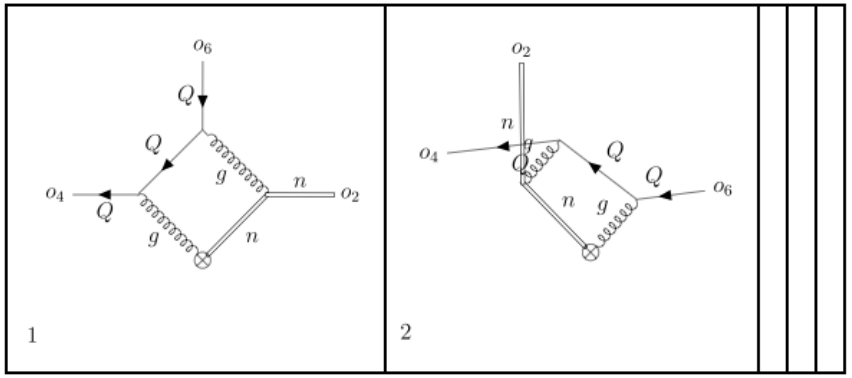
```
[nbar, e, g; QCD='+0']
```

```
[ model = 'FF Model' ]
%-----------------------------------
% Propagators
%-----------------------------------
[q, qbar, -]
[Q, Qbar, -]

[gh, ghbar, -]
[g, g, +, notadpole]

[e, e, +, external]
[n, nbar, +]
%-----------------------------------
% Vertices
%-----------------------------------
[gbar, g, g; QCD='+1']
```

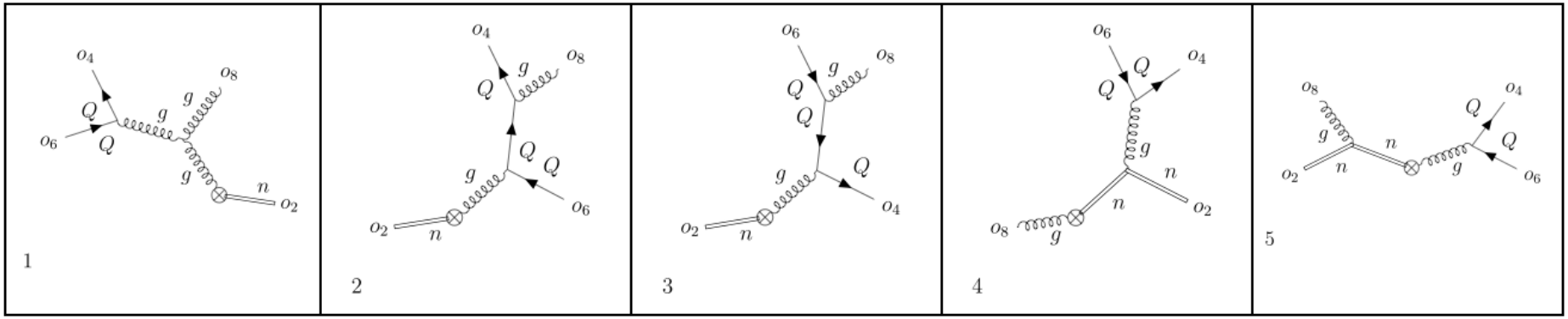
Tree




Virtual



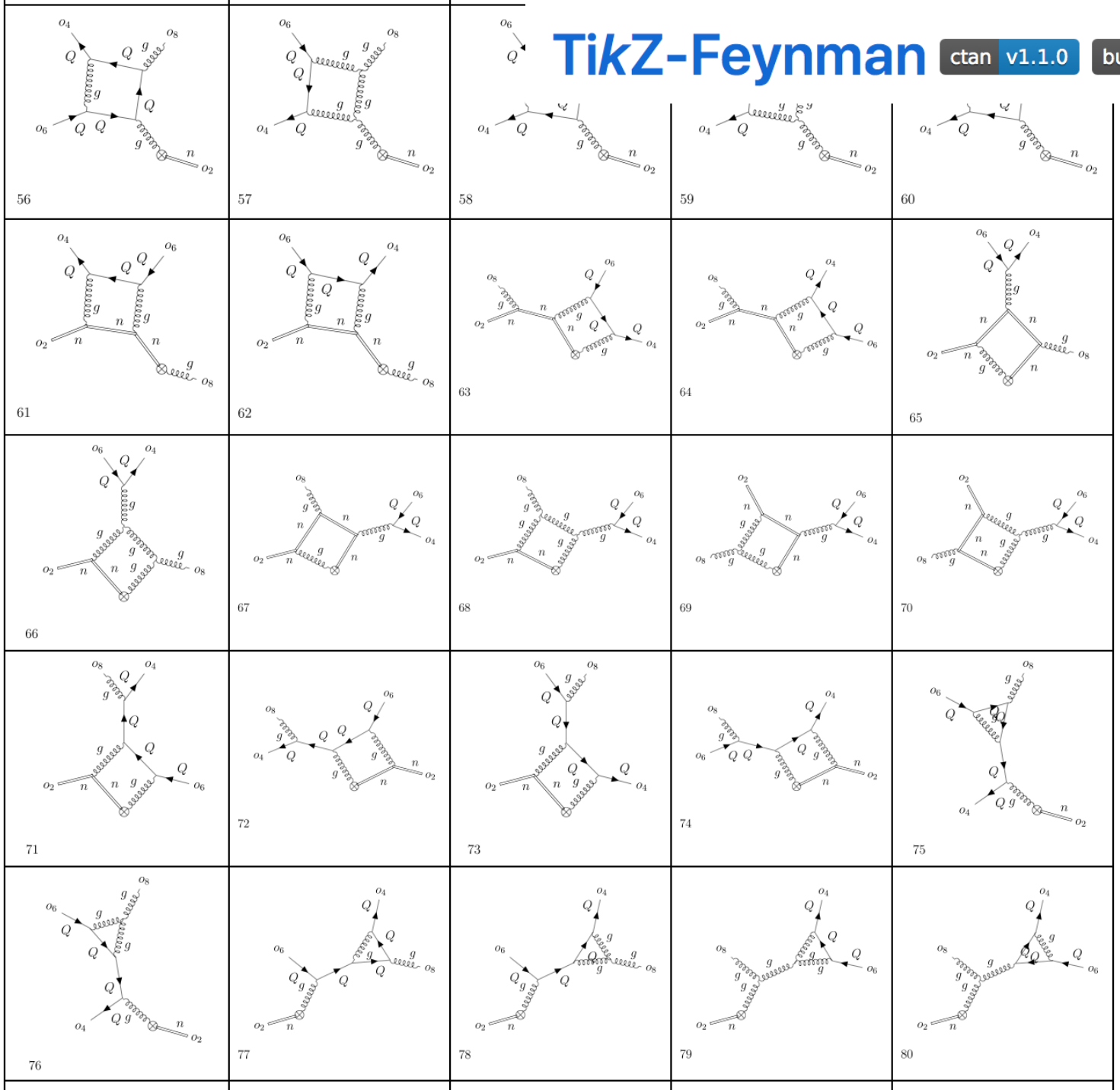

Real




```
[nbar, e, g; QCD='+0']
```

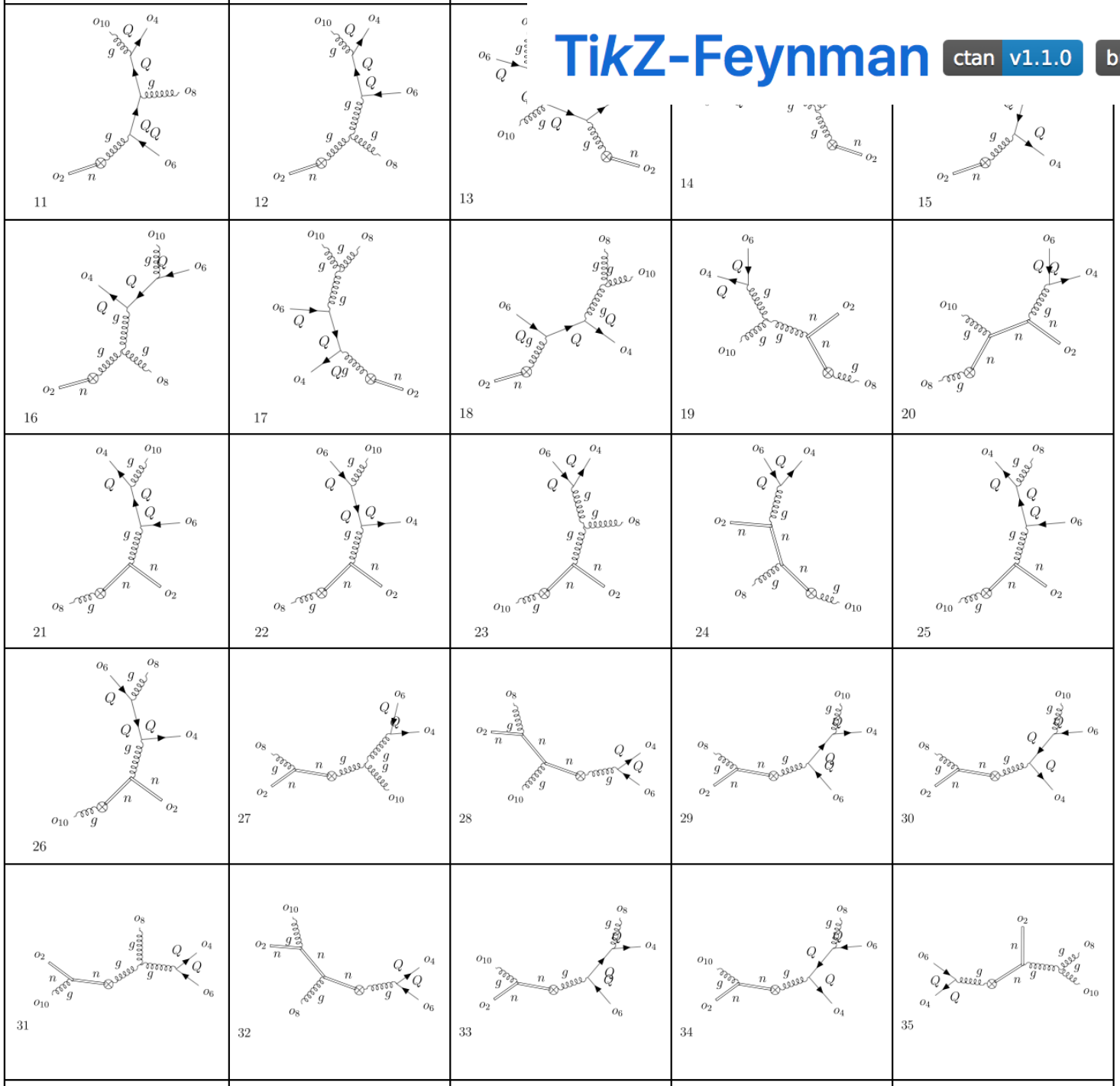
# TikZ-Feynman

ctan v1.1.0 build failing



# TikZ-Feynman

ctan v1.1.0 build failing



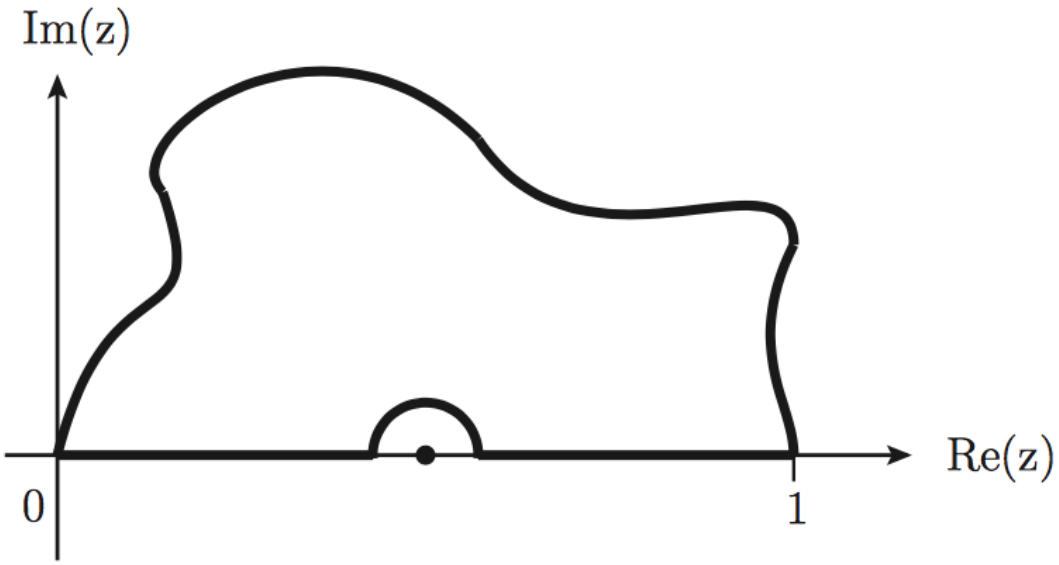
# Some Hints in Sector Decomposition

①

# Contour Deformation

- Before or After Sector Decomposition

*Deformation of the integration contour*



①

# Contour Deformation

- Before or After Sector Decomposition

$$z_k = x_k - i\lambda_k x_k(1 - x_k)\frac{\partial \mathcal{F}}{\partial x_k}$$

Since it is commutable between contour deformation and sector decomposition, we can perform the contour deformation after sector decomposition, especially when we write it to the `Fortran` source code, the file size or compilation problem can be easily solved, and each file size is as small as the case for the Euclidean cases. Now the contour deformation is performed dynamically at run time instead of in terms of huge static explicit expression, *e.g.*, we only need $n^2$ matrix elements in our `Fortran` source code, instead of $n!$ terms, each of which is a product of $n$ matrix elements, for the Jacobi determinant, with $n$ the dimension of the matrix.

②

# Optimization $\lambda_k = \lambda \hat{\lambda}_k$

$$z_k = x_k - i\lambda_k x_k (1 - x_k) \frac{\partial \mathcal{F}}{\partial x_k}$$

We want to find a maximal $\lambda_{\max}$ such that $\forall \lambda < \lambda_{\max}$, the contour deformation guarantees a negative imaginary part for the $\mathcal{F}$ term.

②

# Optimization $\lambda_k = \lambda\hat{\lambda}_k$

$$z_k = x_k - i\lambda_k x_k(1 - x_k)\frac{\partial \mathcal{F}}{\partial x_k}$$

We want to find a maximal $\lambda_{\max}$ such that $\forall\lambda < \lambda_{\max}$, the contour deformation guarantees a negative imaginary part for the $\mathcal{F}$ term.

for each specific $x_k = \tilde{x}_k$, the imaginary part $\mathcal{I}_m\mathcal{F}(\tilde{x}_k, \lambda)$ is a polynomial in $\lambda$

first positive root $\lambda_1^+(\tilde{x}_k)$

$$\mathcal{I}_m\mathcal{F}(\tilde{x}_k, \lambda_1^+(\tilde{x}_k)) = 0 \quad \texttt{MPSolve}$$

②

# Optimization $\lambda_k = \lambda\hat{\lambda}_k$

$$z_k = x_k - i\lambda_k x_k(1 - x_k)\frac{\partial \mathcal{F}}{\partial x_k}$$

We want to find a maximal $\lambda_{\max}$ such that $\forall\lambda < \lambda_{\max}$, the contour deformation guarantees a negative imaginary part for the $\mathcal{F}$ term.

for each specific $x_k = \tilde{x}_k$, the imaginary part $\mathcal{I}_m\mathcal{F}(\tilde{x}_k, \lambda)$ is a polynomial in $\lambda$

first positive root $\lambda_1^+(\tilde{x}_k)$

$$\mathcal{I}_m\mathcal{F}(\tilde{x}_k, \lambda_1^+(\tilde{x}_k)) = 0 \qquad \texttt{MPSolve}$$

$$\lambda_{\max} = \min\{\lambda_1^+(\tilde{x}_k)\} \qquad \texttt{Hooke-Jeeves} \text{ algorithm}$$

②

# Optimization $\lambda_k = \lambda\hat{\lambda}_k$

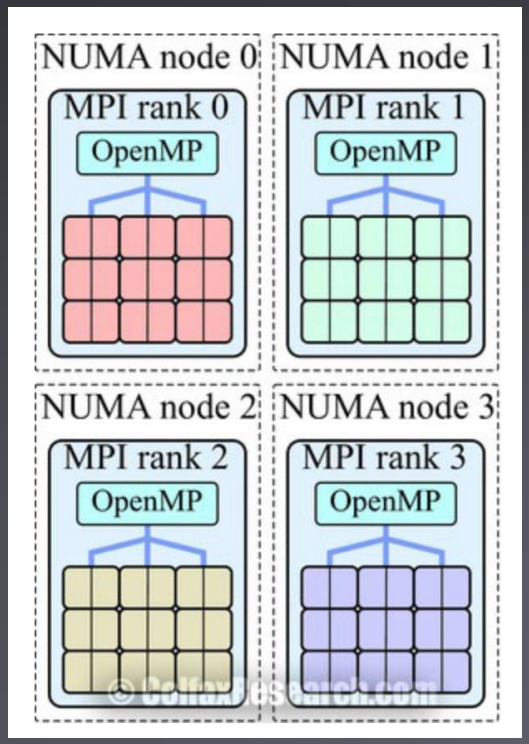
To find an optimal set $\lambda_k$, we first sample some random directions, for a specific direction $\hat{\lambda}_k$, we have a maximal $\lambda_{\max}$ by the procedure above, then we can split the range $(0, \lambda_{\max})$ into $N$ parts, and pick up an $\lambda_i$ in each part to perform the numerical integration with a fixed number of sampled evaluation points, and the $\lambda_m$ corresponding to lower error estimation is preferred. Finally we get an optimized direction $\hat{\lambda}_k$ and $\lambda_m$, then $\lambda_k = \lambda_m\hat{\lambda}_k$ is the optimal set which we want.

Since it belongs to the so-called SIMD (Single Instruction, Multiple Data) pattern, it can be easily parallelized on the HPC cluster.

③

# Numerical Integration

HCubature: provides Vectorized interface to evaluate the integrand in batches of several points at a time



Hybrid MPI+OpenMP

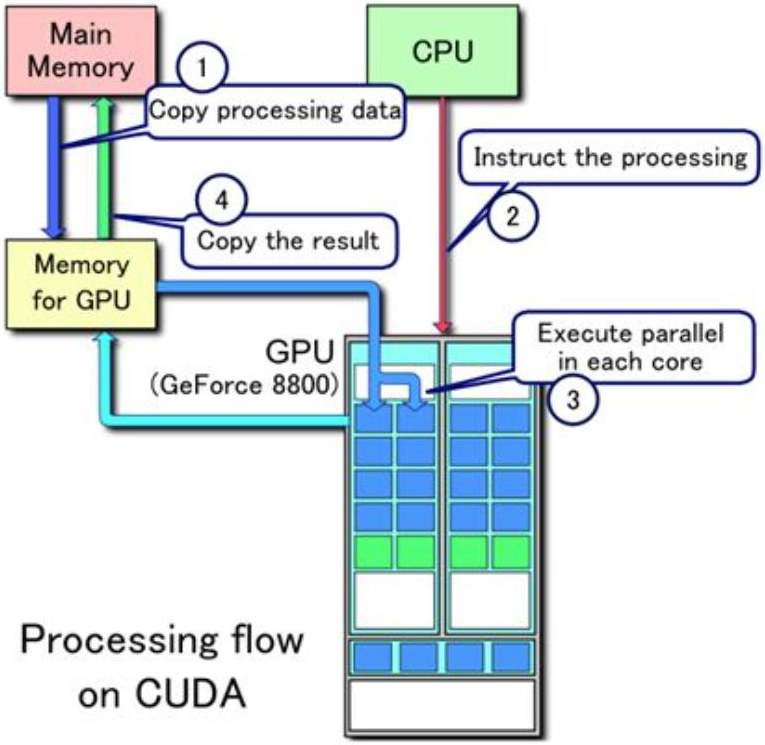
Trying GPU

③

# Numerical Integration

## Processing Flow

- Processing Flow of CUDA:
  - Copy data from main mem to GPU mem.
  - CPU instructs the process to GPU.
  - GPU execute parallel in each core.
  - Copy the result from GPU mem to main mem.




Processing flow on CUDA

④

# Extrapolation

$$\mathcal{I}(\varepsilon) = c_0 + c_1\varepsilon + c_2\varepsilon^2 + \cdots$$

A linear extrapolation method by solving linear systems of order $(\nu+1)\times(\nu+1)$:

$$\mathcal{I}(\varepsilon_l) = c_0 + c_1\varepsilon_l + c_2\varepsilon_l^2 + \cdots + c_\nu\varepsilon_l^\nu$$

non-linear extrapolation

$\varepsilon$-algorithm: $\varepsilon^{\alpha_k}\ln^{\gamma_k}\varepsilon$

④

# Extrapolation

Numer. Math. 35, 175 – 187 (1980)

Numerische Mathematik
© by Springer-Verlag 1980

$S$ is computed by solving the linear system

$$S_{n+i} = S + a_1 g_1(n+i) + \ldots + a_k g_k(n+i) \qquad i = 0, \ldots, k.$$

| | |
|---|---|
| – $g_i(n) = x_n^i$ | Richardson extrapolation process ($\{x_n\}$ is a given auxiliary sequence of parameters). |
| – $g_i(n) = x_{n+i-1}$ | $G$-transformation [9]. |
| – $g_i(n) = x_i^n$ | Summation processes. |
| – $g_i(n) = \Delta S_{n+i-1}$ or $g_i(n) = \Delta^i S_n$ | Shanks transformation [16]. |
| – $g_i(n) = (\Delta S_n)^i$ | Germain-Bonne transformation [8]. |
| – $g_i(n) = x_n^{i-1} \Delta S_n / g(n)$ | Levin generalized transformation [13] ($g$ is a given sequence). |
| – $g_i(n) = R^i(S_n \Delta x_n)/\Delta x_n$ | First generalization of the $\varepsilon$-algorithm [3] where $R$ is an operator which generalizes $\Delta$. |
| – $g_1(n) = x_n$, $g_i(n) = \Delta S_{n+i-2}$ | Process $p$ [3]. |

④

# Extrapolation

Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 108C (2017) 1773–1782

Procedia Computer Science

International Conference on Computational Science, ICCS 2017, 12-14 June 2017, Zurich, Switzerland

## Feynman loop numerical integral expansions for 3-loop vertex diagrams

E de Doncker[1] and F Yuasa[2]

[1] Western Michigan University, Kalamazoo MI 49008, U.S.A.
elise.dedoncker@wmich.edu

[2] High Energy Accelerator Research Organization (KEK), Tsukuba, Ibaraki, 305-0801, Japan
fukuko.yuasa@kek.jp

④

# Extrapolation

$$-\Gamma^3(1-\varepsilon)\, I_a^{V3}(\varepsilon) \sim -\left(\frac{1}{6}\,\varepsilon^{-2} + \frac{5}{3}\,\varepsilon^{-1} + \ldots\right) \tag{6}$$
$$\approx -0.16666667\,\varepsilon^{-2} - 1.6666667\,\varepsilon^{-1} - 11.4891337 - 59.817128\,\varepsilon + \mathcal{O}(\varepsilon^2)$$

Table 1: Results 3-loop vertex integral $-\Gamma^3(1-\varepsilon)\, I_a^{V3}$ by ParInt (on 4 nodes with 16 procs. per node of the *thor* cluster, in *long double* precision), $t_r = 10^{-13}$, max. # evals = 10 B, $\varepsilon = \varepsilon_\ell = 1.2^{-\ell}$. Successive columns list $\ell$, rel. error estimate $E_r$, time $T$ in seconds, and approximate expansion coefficients $C_{\kappa+j}$, $\kappa = -2$, $j \geq 0$.

| | INTEGRAL Fig 1(a) | | | | |
|---|---|---|---|---|---|
| $\ell$ | $E_r$ | $T[s]$ | RES. $C_{-2}$ | RES. $C_{-1}$ | RES. $C_0$ |
| 22 | 1.1e-07 | 55.3 | | | |
| 23 | 1.9e-07 | 55.3 | -0.16291495 | -2.103254 | |
| 24 | 3.2e-07 | 55.5 | -0.16692074 | -1.616737 | -14.650 |
| 25 | 4.7e-07 | 55.6 | -0.16665448 | -1.670242 | -11.106 |
| 26 | 7.2e-07 | 55.8 | -0.16666720 | -1.666470 | -11.519 |
| 27 | 1.3e-06 | 56.1 | -0.16666683 | -1.666626 | -11.494 |
| | | (6): | -0.16666667 | -1.666667 | -11.489 |

④

# Extrapolation

$$-\Gamma^3(1-\varepsilon)\, I_b^{V3}(\varepsilon) \sim -\left(\frac{\pi^2\zeta_3}{6} + 10\zeta_5\right)\varepsilon^{-1} - \left(\frac{119\pi^6}{2160} + \frac{31}{2}\zeta_3^2\right) + \mathcal{O}(\varepsilon) \tag{7}$$

$$\approx -12.34658\,\varepsilon^{-1} - 75.3620 + \mathcal{O}(\varepsilon)$$

Table 3: Results 3-loop vertex integral $-\Gamma^3(1-\varepsilon)\, I_b^{V3}$ by ParInt (on 4 nodes with 16 procs. per node of the *thor* cluster, in *long double* precision), $t_r = 10^{-13}$, max. # evals = 125 B, $\varepsilon = \varepsilon_\ell = 1.11^{-\ell}$. Successive columns list $\ell$, rel. error estimate $E_r$, time $T$ in seconds, and approximate expansion coefficients $C_{\kappa+j}$, $\kappa = -1$, $j \geq 0$.

| | INTEGRAL Fig 1(b) | | | |
|---|---|---|---|---|
| $\ell$ | $E_r$ | $T[s]$ | RES. $C_{-1}$ | RES. $C_0$ |
| 25 | 7.4e-06 | 972 | | |
| 26 | 1.2e-05 | 972 | -10.98454 | -31.1676 |
| 27 | 1.7e-05 | 972 | -12.11262 | -63.5043 |
| 28 | 2.4e-05 | 972 | -12.22893 | -71.5243 |
| 29 | 3.4e-05 | 972 | -12.34703 | -75.2208 |
| | | (7): | -12.34658 | -75.3620 |

# Some Problems

①

# IBP Reduction

- IBP identities in a rigorous way

$$\mathcal{I}(P) = \int dK\, f(K,P)$$

$$(K', P') = \begin{pmatrix} \mathbb{A}_{l\times l} & \mathbb{B}_{l\times n} \\ \mathbb{O}_{n\times l} & \mathbb{I}_n \end{pmatrix} \begin{pmatrix} K \\ P \end{pmatrix} \qquad \mathrm{Det}\mathbb{A} \neq 0$$

$$\begin{pmatrix} \mathbb{A}_{l\times l} & \mathbb{B}_{l\times n} \\ \mathbb{O}_{n\times l} & \mathbb{I}_n \end{pmatrix} = \mathbb{I}_{l+n} + \begin{pmatrix} \alpha_{l\times l} & \alpha_{l\times n} \\ \mathbb{O}_{n\times l} & \mathbb{O}_{n\times n} \end{pmatrix}$$

$$f(K', P') = f(K,P) + \frac{\partial f}{\partial v_i^\mu}(K,P)\alpha_{ij} v_j^\mu$$

①

# IBP Reduction

- IBP identities in a rigorous way

$$dK' = \mathrm{Det}(\mathbb{I}_l + \alpha_{l\times l})dK = (1 + \mathrm{Tr}\alpha_{l\times l})dK$$

$$\mathcal{I}(P') = \int \left[ f(K,P) + \frac{\partial f}{\partial v_i^\mu}(K,P)\alpha_{ij}v_j^\mu \right] (1 + \mathrm{Tr}\alpha_{l\times l})dK$$

$$\int \left[ \alpha_{ij}v_j^\mu \frac{\partial f}{\partial v_i^\mu}(K,P) + f(K,P)\mathrm{Tr}\alpha_{l\times l} \right] dK = 0$$

$$\int \left[ \alpha_{ij}v_j^\mu \frac{\partial f}{\partial v_i^\mu}(K,P) + f(K,P)\alpha_{ij}\delta_{ij} \right] dK = 0$$

①

# IBP Reduction

- IBP identities in a rigorous way

$$\int \frac{\partial}{\partial k_i^\mu}\left[v_j^\mu f(K,P)\right] = 0$$

Lie Algebra:

$$O_{ij} = \delta_{ij} + v_j^\mu \frac{\partial}{\partial v_i^\mu} = \frac{\partial}{\partial v_i^\mu} v_j^\mu$$

$$[O_{ij}, O_{kl}] = \delta_{il} O_{kj} - \delta_{kj} O_{il}$$

①

# IBP Reduction

- IBP identities with Dirac Delta functions

$$\delta(x) \longrightarrow \frac{1}{2\pi\, i}\left[\frac{1}{x-i\varepsilon}-\frac{1}{x+i\varepsilon}\right]$$

$$F(m,n,\cdots) \equiv \prod_i \int d^n k_i \, \frac{1}{x^m}\frac{1}{D_1^n}\cdots$$

$$F'(m+1,n,\cdots) \equiv \begin{cases} \displaystyle\prod_i \int d^n k_i \, (-1)^m \frac{\delta^{(m)}(x)}{m!}\frac{1}{D_1^n}\cdots & (m \geq 0) \\ \displaystyle\prod_i \int d^n k_i \, x^{-m}\delta(x)\frac{1}{D_1^n}\cdots & (m < 0) \end{cases}$$

①

# IBP Reduction

- IBP identities with Dirac Delta functions

$$F(a,b,c,d) =$$

$$\int d^D k_1 \int d^D k_2 \, \delta_+(k_1^2)\, \delta_+(k_2)^2\, \delta(k_1^+ + k_2^+ - (1-z)k^+)$$

$$\left[\frac{1}{p\cdot k_2 + m^2}\right]^a \left[\frac{1}{n\cdot k_2}\right]^b \left[\frac{1}{(k_1+k_2)^2}\right]^c \left[\frac{1}{(k_1+k_2+2p)^2}\right]^d$$

Using `LiteRed`

$$F(1,0,1,1) = \frac{(1-2\epsilon)^2 z F(0,0,0,1)}{2\epsilon^2(z-1)} + \frac{1}{2}F(1,0,0,1) + \frac{1}{2}F(1,0,1,0)$$

②

# Fuchsification

- $n \times n$-Matrix with $n > 150$ rusults in large expression (100MB) in fuchsification at a given point.
- Failed to reduce the Moser-rank to 0 in fuchsification at a given point.
- Matrix Exponentail, Jordan Decomposition, etc., seems really slow when Matrix is complicated.
- $\cdots$

Thanks for your attention!