"Abusing" QGRAF

P. Nogueira
CFIF, IST-UTL, Lisbon, Portugal

ACAT05, Desy-Zeuthen, May 2005

NB: document revised on May 31, 2005

Plan of the talk:

1) preliminary considerations; the issue "optimal" versus "fast" versus "elegant"

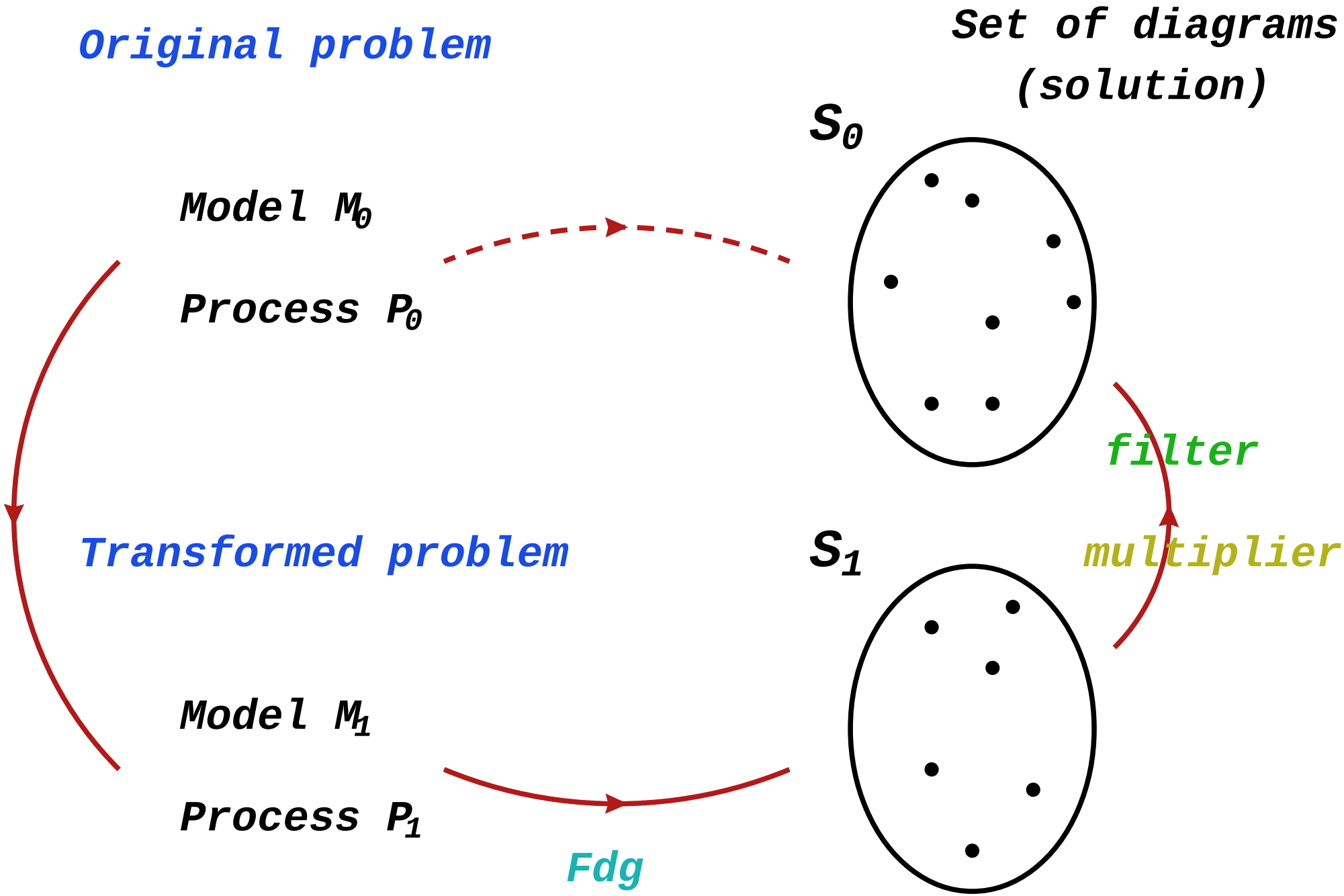
2) some examples of generating Feynman diagrams that require going beyond the basic options currently provided by QGRAF

3) recent evolution of QGRAF, and projection for the near future

** What if QGRAF (or some other Feynman diagram generator) cannot provide an exact solution to a given problem?*

** Is it possible to transform the original problem into a new one that can be solved exactly?*

** Is it acceptable to have some duplication of diagrams? (with the symmetry factors adding up to the right value)*


Original problem
Set of diagrams
(solution)
S_0
Model M_0
Process P_0
filter
multiplier
Transformed problem
S_1
Model M_1
Process P_1
Fdg

"filter"

module (based on a set of operative conditions) that validates some diagrams and eliminates the remaining ones

"multiplier"

module that replaces an input diagram with a number of others (at least one), by assigning them additional (and different) labels

Simple example (just for completeness)

**) counterterms*

if for each interaction vertex V and propagator P

$$V_c = \alpha\, V_{c1} + \alpha^2\, V_{c2} + \ldots$$

$$P_c = \alpha\, P_{c1} + \alpha^2\, P_{c2} + \ldots$$

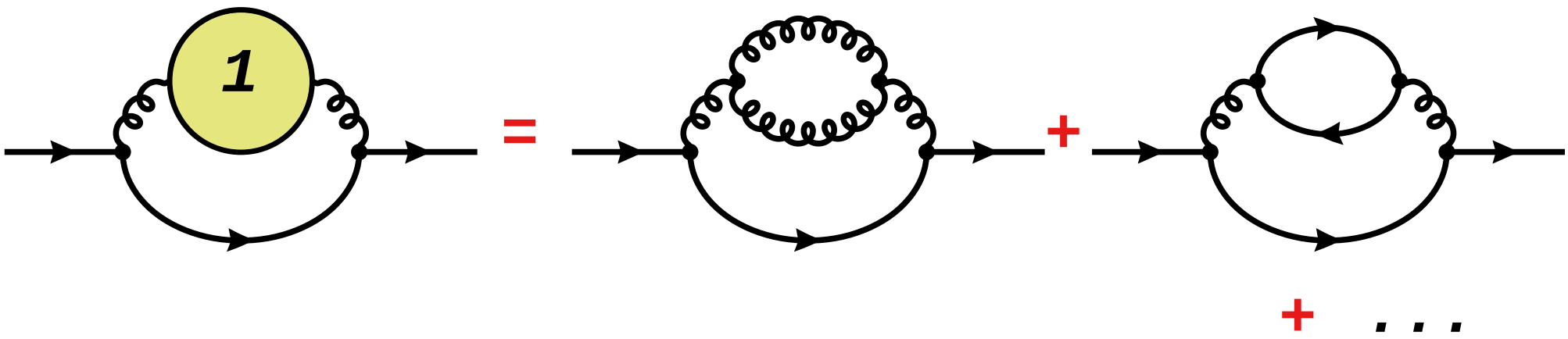
then the terms of order α^n obtained from the m-loop diagrams contribute to the counterterms of order m+n (ie, m+n loops)

** there is some redundancy left (ie, there are equivalent terms generated by this procedure)*

Problem 1

use "blobs" (of fixed loop order) to represent propagator type sub-diagrams, instead of the usual diagrammatic expansion, assuming that no mixed propagators exist

* *there should be no overlap between diagrams*
* *if QGRAF is generating diagrams for a propagator then the loop order of the blobs should be smaller than the number of loops provided as input*



Solution

1) define multiple propagators in the model file, one per loop order (from 0 to k, say)

2) require no self-energies (nosigma)*

3) perform various runs (nloop=0,1,...,k) and constrain sum of blob loop orders:

$$\Sigma_b \; lo(b) = k\text{-}nloop$$

** please note that each tree-level propagator of the modified model represents a connected 2-point function of the original model (with a certain number of loops)*

The new model (propagators)

```
[ quark0, QUARK0, - ; lo= 0]
[ quark1, QUARK1, - ; lo= 1]
[ quark2, QUARK2, - ; lo= 2]

[ gluon0, gluon0, + ; lo= 0]
[ gluon1, gluon1, + ; lo= 1]
[ gluon2, gluon2, + ; lo= 2]

[ ghost0, GHOST0, - ; lo= 0]
[ ghost1, GHOST1, - ; lo= 1]
[ ghost2, GHOST2, - ; lo= 2]
```

(propagator insertions up to 2 loops)

The new model (vertices)

[quark0, QUARK0, gluon0]

[quark0, QUARK0, gluon1]
[quark0, QUARK1, gluon0]
[quark1, QUARK0, gluon0]

[quark0, QUARK1, gluon1]
[quark1, QUARK0, gluon1]
[quark1, QUARK1, gluon0]
[quark0, QUARK0, gluon2]
[quark0, QUARK2, gluon0]
[quark2, QUARK0, gluon0]

etc

*) to avoid a multiplicity of field names in the output file:

```
<field><back>
```

*) effective loop order constraint is a propagator weight constraint**

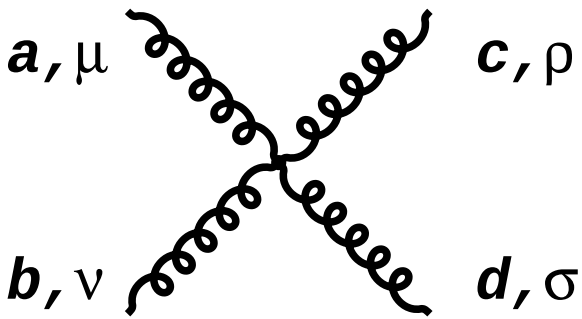
```
true = psum[ lo, 2, 2] ;
```

** version 3.1

Problem 2

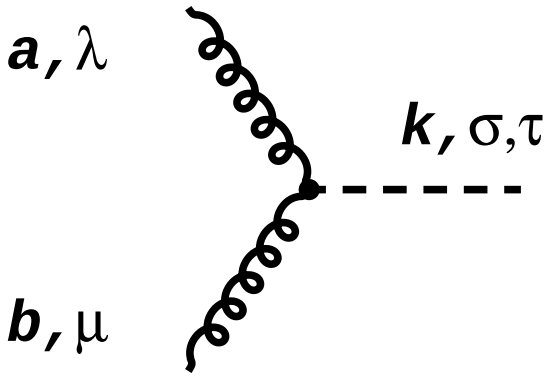
obtain an expansion for QCD such that there exists a global colour factor for any diagram

* *the obstacle is the 4-gluon vertex, which contains three different pieces:*

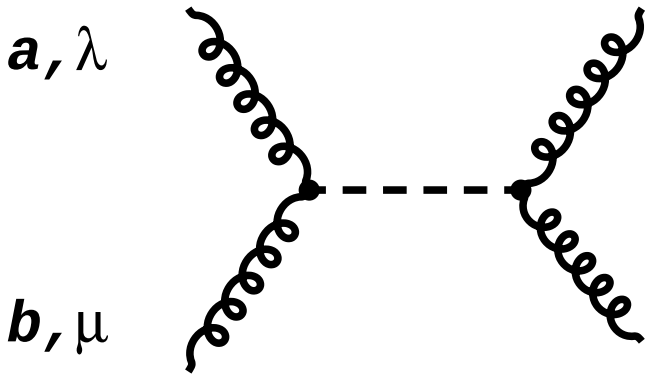


$$\sim g^2 \left(f_{kab} f_{kcd} (g_{\mu\rho} g_{\nu\sigma} - g_{\mu\sigma} g_{\nu\rho}) + f_{kac} f_{kdb} (g_{\mu\sigma} g_{\rho\nu} - g_{\mu\nu} g_{\rho\sigma}) + f_{kad} f_{kbc} (g_{\mu\nu} g_{\sigma\rho} - g_{\mu\rho} g_{\sigma\nu}) \right)$$

introduce a new fictitious field that couples to gluons only

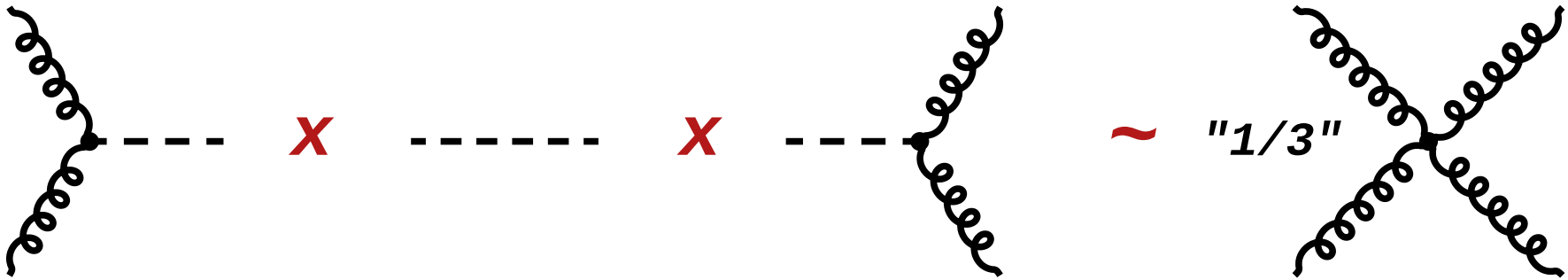


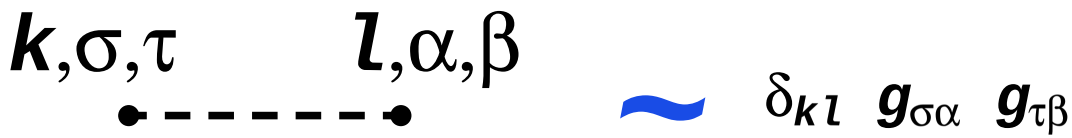
$\sim$ $(g/\sqrt{2})\, f_{kab}\, (g_{\lambda\sigma}g_{\mu\tau} - g_{\lambda\tau}g_{\mu\sigma})$



$\sim$ $g^2\, f_{kab} f_{kcd} (g_{\lambda\nu}g_{\mu\rho} - g_{\lambda\rho}g_{\mu\nu})$

* *new field is non-propagating (ie, the propagator is momentum-independent)*



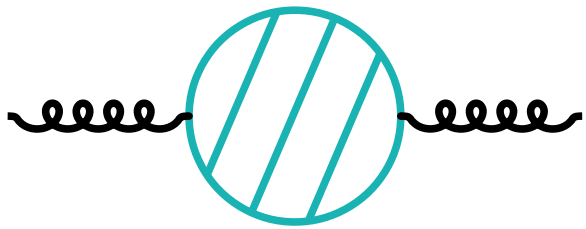


NB1: algebra may be simplified

NB2: factors like +i or -i are not included

L	A	B	C
1	3	3	4
2	18	32	27
3	254	534	494
4	4970	14038	12562
5	120074	450566	395865

L - loops



1PI*

A: normal counting (QCD, onepi, nosnail)

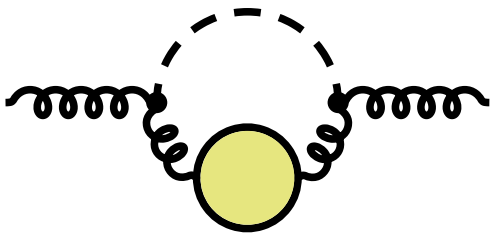
B: modify normal counting by splitting each quartic vertex into 3 terms

C: adjusted counting for the cubic lagrangean (notadpole, no bridges for real fields)

Number of diagrams increases but many of them are simpler. The effective impact can be estimated in a real calculation.

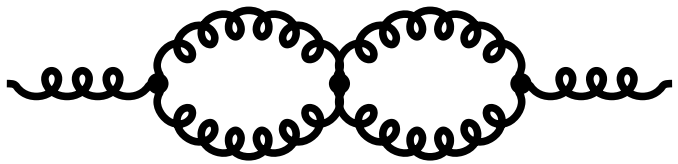
Numbers in column C are smaller than those in column B because sometimes there is a symmetry that is automatically accounted for in the diagrams for the cubic model.*

** there remain extra diagrams in column C, so those numbers can be lowered:*



these are snails!

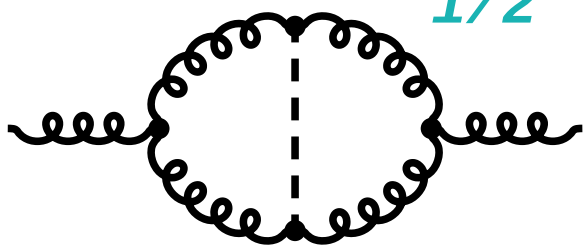
Example: the diagram



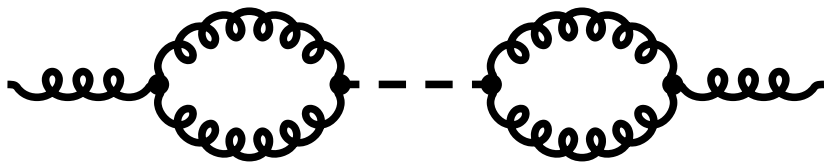
$f_s = 1/4$

gives 3 terms if expanded but in the cubic model there are just 2 diagrams

1/2



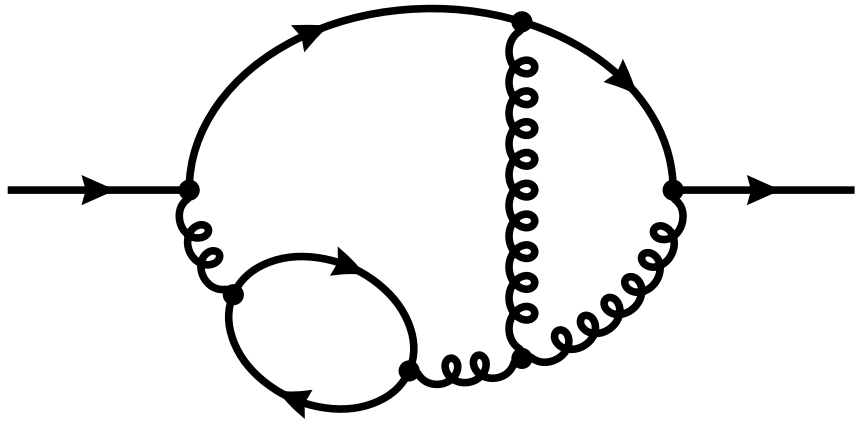
1/4



Of course, there is a (partial) symmetry in the symbolic amplitude for the original diagram.

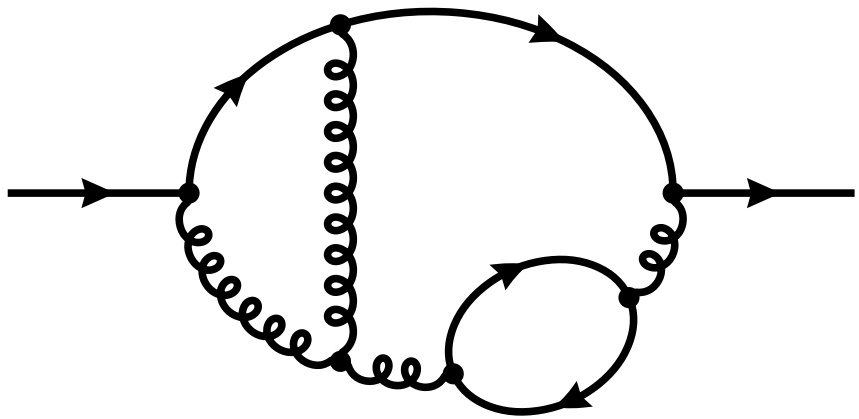
Problem 3

assume that in a certain model some quantities are nearly independent of the fermion flow direction (differences occuring on simple factors like the diagram sign or the colour factor); how can one reduce the number of diagrams to be computed, and thus reducing the cpu time required by the computation?



D_a

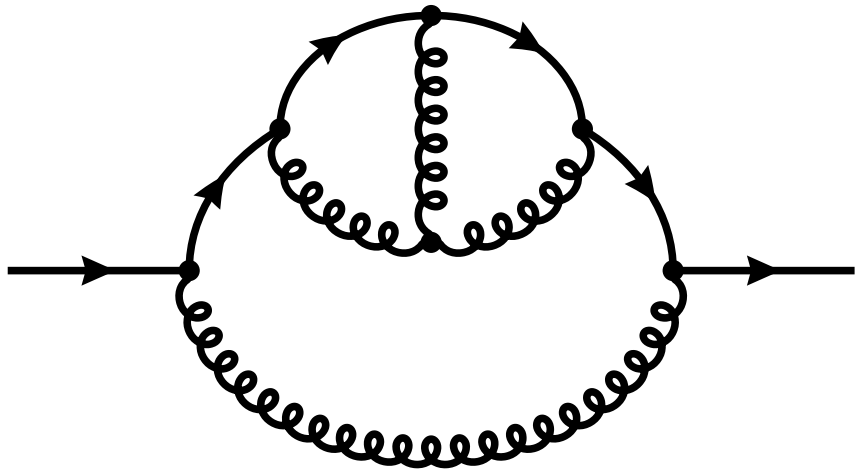
model: QCD



D_b

1) D_a and D_b are two different diagrams

2) D_a and D_b are related by fermion number flow inversion (ignore "time")



D_c

model: QCD

1) inversion of fermion number flow does not generate a new diagram (ignore "time")

Solution (for the quark propagator, say):

1) replace the fermion field by a real scalar field H

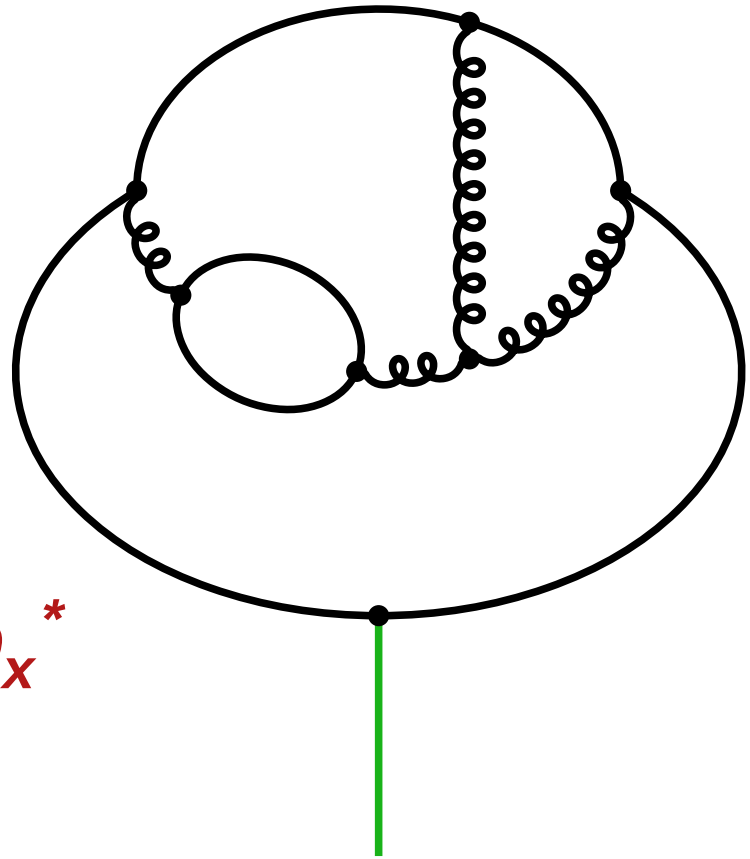
2) add a new scalar field D (external)

3) add a new vertex HHD

4) increment the number of loops by 1 and run QGRAF for the tadpole of D

5) reintroduce the fermion in the diagrams, ie replace H by the original fermion (pick a flow) and compute the diagram sign

6) restore original contributions due to other diagrams

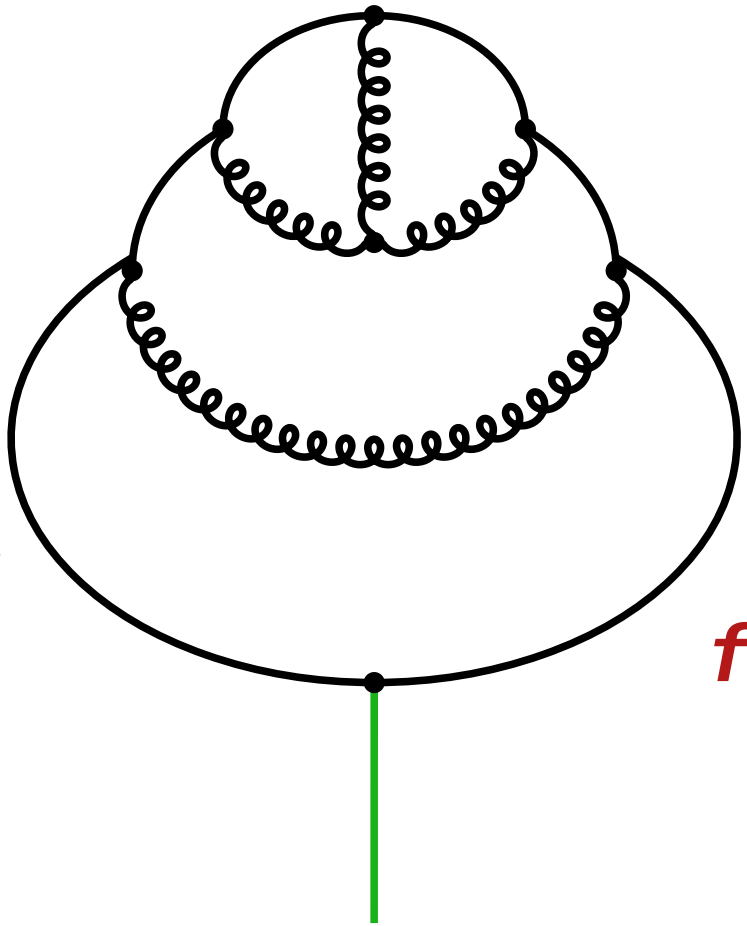


D_X^*

$f_s(D_X^*) = f_s(D_a^*)$

1) D_X "contains" diagrams D_a^* and D_b^*

2) the symmetry factors are identical



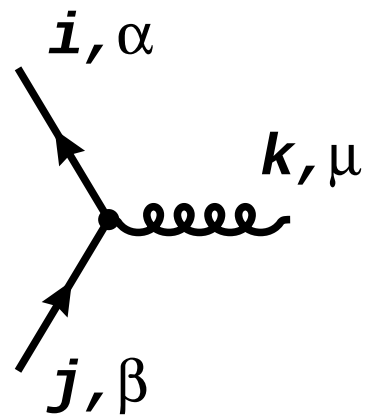
D_y^*

$$f_s(D_y^*) = (1/2)\ f_s(D_c^*)$$

1) D_y "contains" diagram D_c^*

2) the symmetry factors differ by a factor of 1/2

original contributions due to other diagrams can be restored as follows



has colour factor $(T^k)_{ij}$

For each quark line do the one-time replacement

$$T^a\ T^b \ldots T^z \longrightarrow T^a\ T^b \ldots T^z \pm T^z \ldots T^b\ T^a$$

Some references:

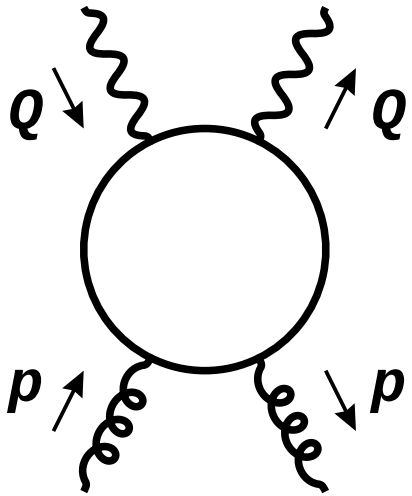
"The next-next-to-leading QCD approximation for non-singlet moments of deep-inelastic structure functions", S.A. Larin, T. van Ritbergen and J. Vermaseren, Nucl. Phys. B 427, 41-52 (1994)

"The 3-loop calculation of the moments of deep inelastic structure functions", S.A. Larin, P. Nogueira, T. van Ritbergen and J. Vermaseren, Nucl. Phys. B492, 338-378 (1997)

"Some higher moments of deep inelastic structure functions at next-to-next-to-leading order of perturbative QCD", A. Retey and J. Vermaseren, Nucl. Phys. B 604, 281-311 (2001)

"The 16^{th} Moment of the Non-Singlet Structure Functions $F_2(x,Q^2)$ and $F_L(x,Q^2)$ to $O(\alpha_s^3)$", J. Bluemlein and J. Vermaseren, Phys. Lett. B 606, 130-138 (2005)

"The Longitudinal Structure Function at Third Order", S. Moch, J.A.M. Vermaseren and A. Vogt, Phys. Lett. B 606, 123-129 (2005)



extra symmetries (paired momenta)

diagram generation used 5-loop propagator

Recent evolution of QGRAF

(since the previous ACAT)

1) improved memory management (v3.0)

2) implementation of parameter functions (v3.0)

3) implementation of constraints defined in terms of vertex and propagator weights (v3.1)

**** point 3) is related with problem 1***

The presence of variables/parameters leads to

****) better output format***

****) model can be defined in a better way***

****) new filters (eg, weights)***

****) generalization of that part of the algorithm related with symmetries (elimination, symmetry factor); this will eliminate some weak points***