Numerical Evaluation of Multi-loop Integrals

Sophia Borowka

MPI for Physics, Munich

Max-Planck-Institut für Physik (Werner-Heisenberg-Institut)

IMPRS EPP

In collaboration with: J. Carter and G. Heinrich

Based on arXiv:1204.4152 [hep-ph]

LoopFest XI, Pittsburgh
May 11th, 2012

http://projects.hepforge.org/secdec

Making Predictions in the LHC Era

- Precise predictions are desirable for the upcoming wealth of LHC data
- Higher orders in perturbation theory are needed
- Multi-dimensional parameter integrals need to be evaluated which can contain UV, soft and collinear singularities

Making Predictions in the LHC Era II

- A lot of progress has been achieved towards the goal of describing hadron collider processes consistently at NLO
- Calculations beyond NLO are also progressing well, but automation is difficult, and analytic methods to calculate e.g. two-loop integrals involving massive particles reach their limit
- Numerical methods are in general easier to automate, problems mainly are
  - Extraction of IR and UV singularities
  - Numerical convergence in the presence of integrable singularities (e.g. thresholds)
  - Speed/accuracy

Many people are/have been working on purely numerical methods, e.g. Soper/Nagy et al., Binoth/Heinrich et al., Kurihara et al., Passarino et al., Lazopoulos et al., Anastasiou et al., Freitas et al., Becker et al., Czakon/Mitov et al., ...

Making Predictions in the LHC Era II

- A lot of progress has been achieved towards the goal of describing hadron collider processes consistently at NLO
- Calculations beyond NLO are also progressing well, but automation is difficult, and analytic methods to calculate e.g. two-loop integrals involving massive particles reach their limit
- Numerical methods are in general easier to automate, problems mainly are
  - Extraction of IR and UV singularities (solved with SecDec 1.0)
  - Numerical convergence in the presence of integrable singularities (e.g. thresholds)
  - Speed/accuracy

Many people are/have been working on purely numerical methods, e.g. Soper/Nagy et al., Binoth/Heinrich et al., Kurihara et al., Passarino et al., Lazopoulos et al., Anastasiou et al., Freitas et al., Becker et al., Czakon/Mitov et al., ...

Making Predictions in the LHC Era II

- A lot of progress has been achieved towards the goal of describing hadron collider processes consistently at NLO
- Calculations beyond NLO are also progressing well, but automation is difficult, and analytic methods to calculate e.g. two-loop integrals involving massive particles reach their limit
- Numerical methods are in general easier to automate, problems mainly are
  - Extraction of IR and UV singularities (solved with SecDec 1.0)
  - Numerical convergence in the presence of integrable singularities (e.g. thresholds) (solved with SecDec 2.0)
  - Speed/accuracy

Many people are/have been working on purely numerical methods, e.g. Soper/Nagy et al., Binoth/Heinrich et al., Kurihara et al., Passarino et al., Lazopoulos et al., Anastasiou et al., Freitas et al., Becker et al., Czakon/Mitov et al., ...

Public Implementations of the Sector Decomposition Method on the Market

- sector_decomposition (uses GiNaC) C. Bogner & S. Weinzierl '07
- FIESTA (uses Mathematica, C) A. Smirnov, V. Smirnov & M. Tentyukov '08 '09
- SecDec (uses Mathematica, Perl, Fortran/C++) J. Carter & G. Heinrich '10

Limitation until recently:
Multi-scale integrals were limited to the Euclidean region (i.e., no thresholds)

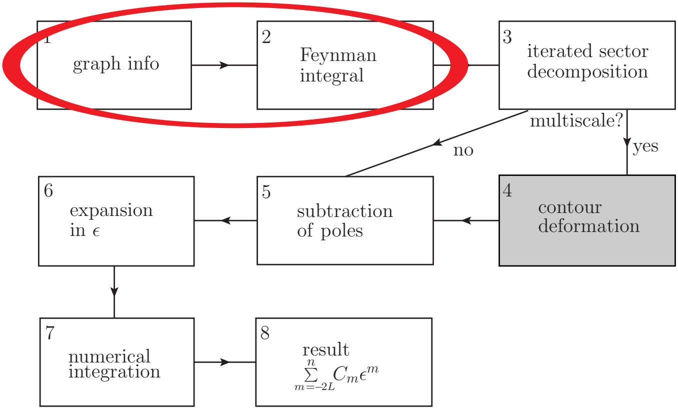
NOW:
Extension of SecDec to general kinematics! SB, J. Carter & G. Heinrich '12

SecDec 2.0 Computes ...

- Feynman graphs for arbitrary kinematics, and more general parametric functions with no poles within the integration region

Feynman graph | or | parametric function

Operational Sequence of the SecDec Program



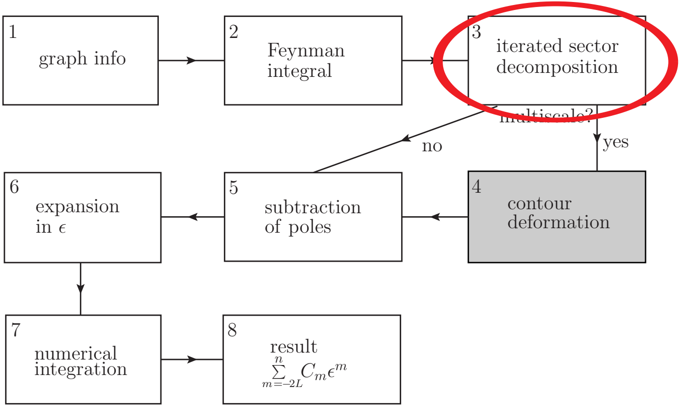
General Feynman Integral

- Graph infos are converted into tensorial **Feynman integral** $G^{\mu_1 \cdots \mu_R}$ in D dimensions at L loops with N propagators to power ν_j of rank R
- After loop momentum integration, a generic scalar **Feynman integral**

$$G = \frac{(-1)^{N_\nu}}{\prod_{j=1}^{N} \Gamma(\nu_j)} \Gamma(N_\nu - LD/2) \int_0^\infty \prod_{j=1}^{N} \mathrm{d}x_j \; x_j^{\nu_j - 1} \, \delta(1 - \sum_{l=1}^{N} x_l) \frac{\mathcal{U}^{N_\nu - (L+1)D/2}(\vec{x})}{\mathcal{F}^{N_\nu - LD/2}(\vec{x})}$$

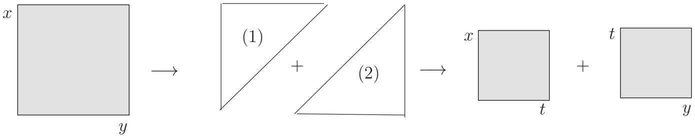
where $N_\nu = \sum_{j=1}^{N} \nu_j$ and where $\mathcal{F}$ and $\mathcal{U}$ can be constructed via **topological cuts**

Operational Sequence of the SecDec Program



Sector Decomposition

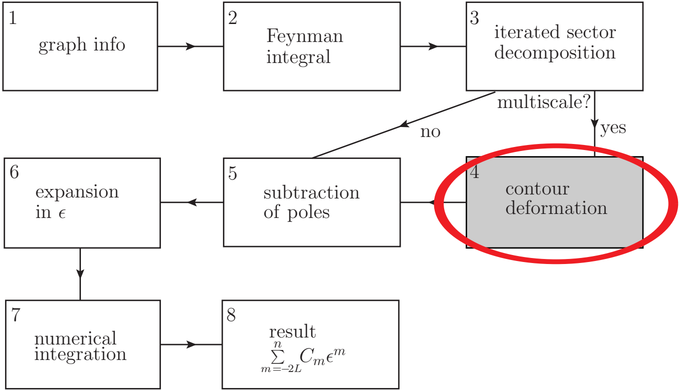
- Overlapping divergences are factorized



$$\int_0^1 dx \int_0^1 dy \, \frac{1}{(x+y)^{2+\epsilon}} = \int_0^1 dx \int_0^1 dt \, \frac{1}{x^{1+\epsilon}(1+t)^{2+\epsilon}} + \int_0^1 dt \int_0^1 dy \, \frac{1}{y^{1+\epsilon}(1+t)^{2+\epsilon}}$$

- Iterated **sector decomposition** is done, where dimensionally regulated soft, collinear and UV singularities are factored out
  Hepp '66, Binoth & Heinrich '00

Operational Sequence of the SecDec Program

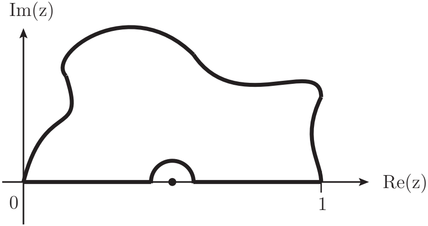


Contour Deformation I

- For kinematics in the physical region, $\mathcal{F}$ can still vanish

$$\mathcal{F}_{example} = m^2(1+t_1)^2 - s\,t_1 - i\delta$$

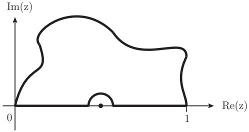
but a deformation of the integration contour



and Cauchy's theorem can help

$$\oint_c f(t)\mathrm{d}t = \int_0^1 f(t)\mathrm{d}t + \int_1^0 \frac{\partial z(t)}{\partial t} f(z(t))\mathrm{d}t = 0$$

Contour Deformation II



- The integration contour is deformed by

$$\vec{t} \to \vec{z} = \vec{t} + \mathrm{i}\vec{y} \ ,$$

$$y_j(\vec{t}) = -\lambda t_j (1 - t_j) \frac{\partial \mathcal{F}(\vec{t})}{\partial t_j}$$

Soper '99

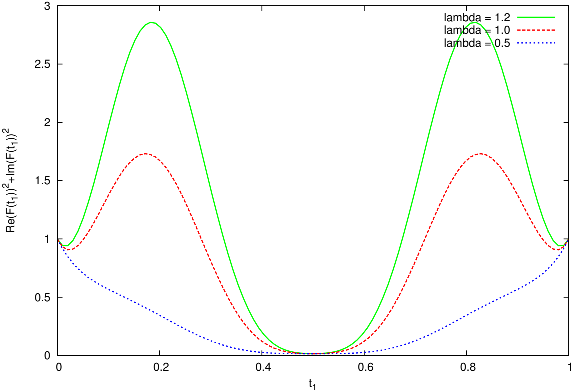
- Integrand is analytically continued into the complex plane

$$\mathcal{F}(\vec{t}) \to \mathcal{F}(\vec{t} + \mathrm{i}\vec{y}(\vec{t})) = \mathcal{F}(\vec{t}) + \mathrm{i} \sum_j y_j(\vec{t}) \frac{\partial \mathcal{F}(\vec{t})}{\partial t_j} + \mathcal{O}(y(\vec{t})^2)$$

Soper, Nagy, Binoth; Kurihara et al., Anastasiou et al., Freitas et al., Becker et al.

Find the Optimal Deformation Parameter λ I

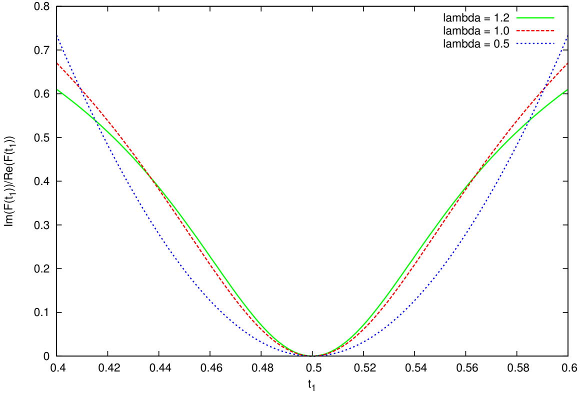
- Robust method: check the maximally allowed λ for $\mathcal{F}$ and maximize the modulus at critical points



example is for 1-loop bubble, $m^2 = 1.0$, $s = 4.5$ with Feynman parameter t_1

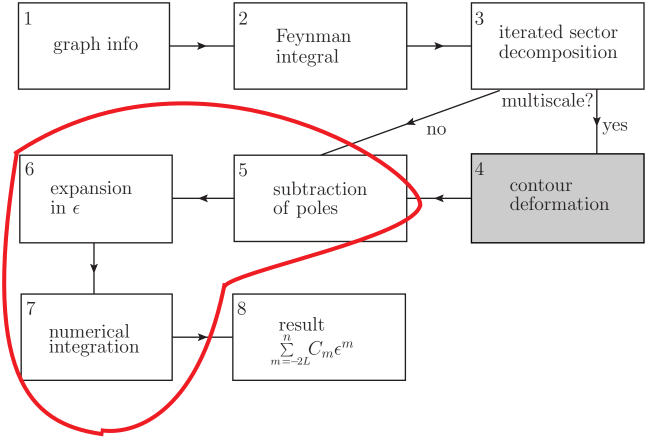
Find the Optimal Deformation Parameter λ II

- Faster convergence: minimize the complex argument of $\mathcal{F}$



example is for 1-loop bubble, $m^2 = 1.0$, $s = 4.5$ with Feynman parameter t_1

Operational Sequence of the SecDec Program



Subtraction, Expansion, Numerical Integration

Subtraction

- The factorized poles in a subsector integrand $\mathcal{I} \propto \mathcal{U}, \mathcal{F}$ are extracted by subtraction (e.g. logarithmic divergence)

$$\int_0^1 \mathrm{d}t_j \, t_j^{-1-b_j\epsilon} \mathcal{I}(t_j, \epsilon) = -\frac{\mathcal{I}(0,\epsilon)}{b_j\epsilon} + \int_0^1 \mathrm{d}t_j \, t_j^{-1-b_j\epsilon} (\mathcal{I}(t_j,\epsilon) - \mathcal{I}(0,\epsilon))$$

Expansion

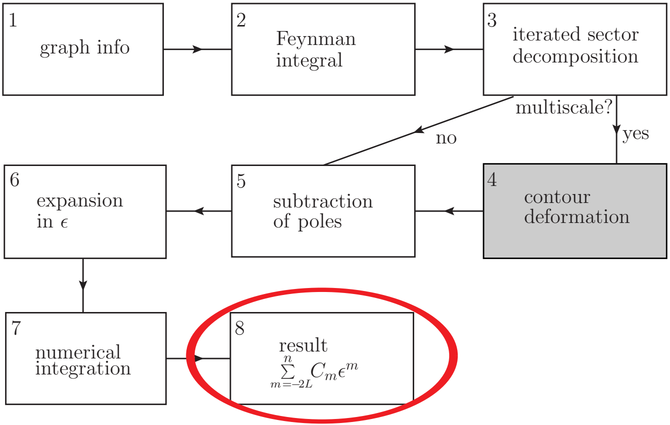
- After the extraction of poles, an expansion in the regulator ϵ is done

Numerical Integration

- Monte Carlo integrator programs containted in CUBA library or BASES can be used for numerical integration

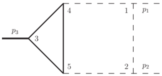
Hahn et al. '04 '11, Kawabata '95

Operational Sequence of the SecDec Program



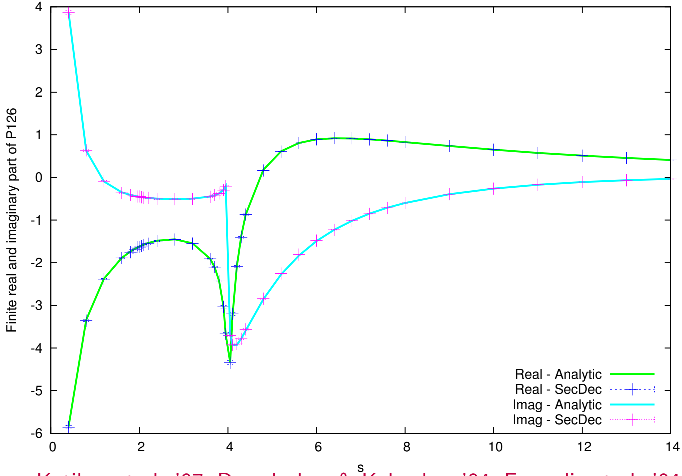
Results

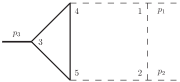
- Successful application of the public **SecDec 1.0** program to massless multi-loop diagrams up to 5-loop 2-point functions and 4-loop 3-point functions for Euclidean kinematics
- Successful application of **SecDec 2.0** to various multi-scale examples, e.g., the massive 2-loop vertex graph, planar and non-planar 6- and 7-propagator massive 2-loop box diagrams
- Timings for the 2-loop vertex diagram and a relative accuracy of 1% using the CUBA 3.0 library on an Intel(R) Core i7 CPU at 2.67GHz



s/m^2	timing (finite part)
3.9	13.6 secs
14.0	12.1 secs

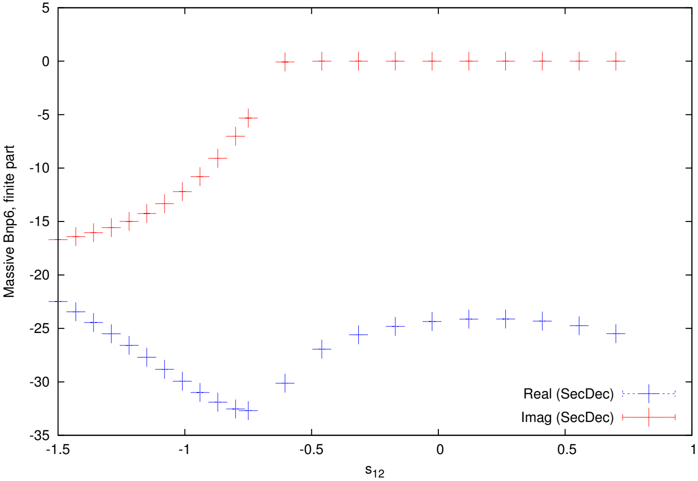
Results II: Massive Two-loop Vertex Graph G

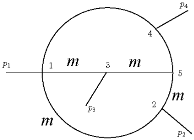




Kotikov et al. '97, Davydychev & Kalmykov '04, Ferroglia et al. '04, Bonciani et al. '04

Results III: Massive Non-planar 6-propagator Graph

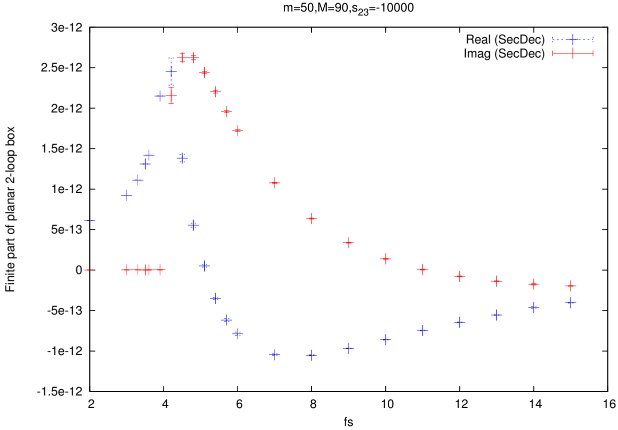


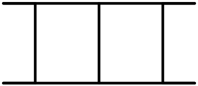


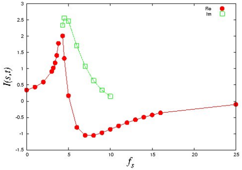
$m \neq 0$

massless case: Tausk '99

Results IV: Planar Massive Two-loop Box

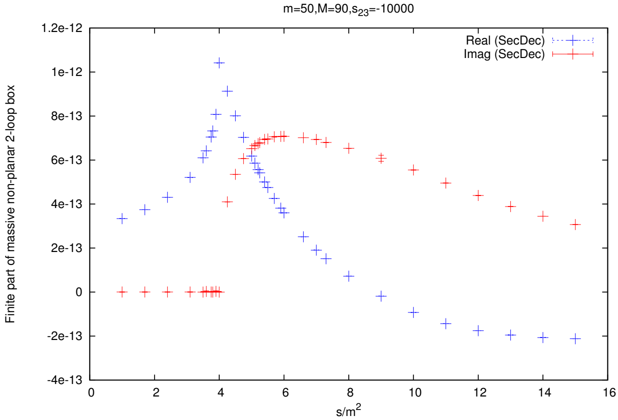


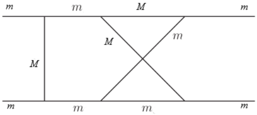


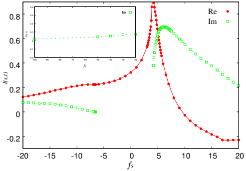


Fujimoto et al. '11

Results V: Non-planar Massive Two-loop Box







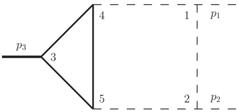
Fujimoto et al. '11

User Input I

- param.input: parameters for integrand specification and numerical integration

```
################ input parameters for sector decomposition ################
#-----------------------------------------------------------------------------------------------------------
# subdirectory for the mathematica output files (will be created if non-existent) :
# if not specified, a directory with the name of the graph given below will be created by default
subdir=2loop
#-----------------------------------------------------------------------------------------------------------
# if outputdir is not specified: default directory for
# the output will have integral name (given below) appended to directory above,
# otherwise specify full path for Mathematica output files here
outputdir=
#-----------------------------------------------------------------------------------------------------------
# graphname (can contain underscores, numbers, but should not contain commas)
graph=P126
#-----------------------------------------------------------------------------------------------------------
# number of propagators:
propagators=6
#-----------------------------------------------------------------------------------------------------------
# number of external legs:
legs=3
#-----------------------------------------------------------------------------------------------------------
# number of loops:
loops=2
#-----------------------------------------------------------------------------------------------------------
# construct integrand (F and U) via topological cuts (only possible for scalar integrals)
# default is 0 (no cut construction used)
cutconstruct=1
#########################################################
# parameters for subtractions and epsilon expansion
#########################################################
```

User Input II



- template.m: definition of the integrand (Mathematica syntax)

```
(******* USER INPUT for construction of integrand ***************************)

(******* Use with cutconstruct=1 *********************************************)

proplist={{ms[1],{3,4}},{ms[1],{4,5}},{ms[1],{5,3}},
          {0,{1,2}},{0,{1,4}},{0,{2,5}}};

(******* Use with cutconstruct=0 *********************************************)
(*
momlist={k1,k2};
proplist={k1^2-ms[1],(k1+p3)^2-ms[1],(k1-k2)^2-ms[1],
         (k2+p3)^2,(k2+p1+p3)^2,k2^2};
numerator={1};
*)

(******* Propagator powers (optional) ****************************************)

powerlist=Table[1,{i,Length[proplist]}];

(******* On-shell conditions (optional) **************************************)
onshell={ssp[1]->0,ssp[2]->0,ssp[3]->sp[1,2],sp[1,3]->0,sp[2,3]->0};

(******* Set Dimension *******************************************************)

Dim=4-2*eps;
(***************************************************************************)
```

Program Test Run

- ./launch -p param.input -t template.m

```
********** This is SecDec version 2.0 **********
Authors: Sophia Borowka, Jonathon Carter, Gudrun Heinrich
************************************************
graph = P126
primary sectors 1,2,3,4,5,6, will be calculated
calculating F and U . . .
done
written to /home/pcl335a/sborowka/Work/SecDecBeta/loop/2loop/P126/FUN.m

results of the decomposition will be written to
/home/pcl335a/sborowka/Work/SecDecBeta/loop/2loop/P126
doing sector decomposition . . .
done

working on pole structure: 2 logarithmic poles, 0 linear poles, 0 higher poles
C++ functions created for pole structure 2l0h0
compiling 2l0h0/epstothe0 ...
doing numerical integrations in P126/2l0h0/epstothe0
compiling 2l0h0/epstothe-1 ...
doing numerical integrations in P126/2l0h0/epstothe-1
compiling 2l0h0/epstothe-2 ...
doing numerical integrations in P126/2l0h0/epstothe-2
working on pole structure: 1 logarithmic pole, 0 linear poles, 0 higher poles
C++ functions created for pole structure 1l0h0
compiling 1l0h0/epstothe0 ...
doing numerical integrations in P126/1l0h0/epstothe0
compiling 1l0h0/epstothe-1 ...
doing numerical integrations in P126/1l0h0/epstothe-1
working on pole structure: 0 logarithmic poles, 0 linear poles, 0 higher poles
C++ functions created for pole structure 0l0h0
compiling 1l0h0/epstothe0 ...
doing numerical integrations in P126/0l0h0/epstothe0
Output written to /home/pcl335a/sborowka/Work/SecDecBeta/loop/2loop/P126/P126_pfull.res

To remove intermediate files, execute the command /home/pcl335a/sborowka/Work/SecDecBeta/loop/launchcleanP126
```

Get the Result

- resultfile P126_full.res

```
************************************************
***OUTPUT: P126 p5 ***************
 point: 7.0
 ext. legs: 0.0 0.0 7.0
 prop. mass: 1.0 0. 0. 0. 0. 0.
Prefactor=-Exp[-2EulerGamma*eps]
************************************************
****** eps^-2 coeff ******

result        =0.07563683
              +0.1003924148 I
error         =0.000493522517701388
              + 0.00139691015080074 I
CPUtime (all eps^-2 subfunctions)  =0.04
CPUtime (longest eps^-2 subfunction)  =0.01
.
.
.

****** eps^0 coeff ******

result        =0.906978296750816
              -0.908781551612644 I
error         =0.00754504726896407
              + 0.0442867373250588 I
CPUtime (all eps^0 subfunctions)  =2.44
CPUtime (longest eps^0 subfunction)  =0.51

************************************************

Time taken for decomposition = 2.005725

Total time for subtraction and eps expansion = 41.5057 secs
Time taken for longest subtraction and eps expansion = 17.8613 secs
```

Summary & Outlook

Summary

- Download the public SecDec program: http://projects.hepforge.org/secdec
- With SecDec the numerical evaluation of multi-loop integrals is possible for arbitrary kinematics

Outlook

- Implement contour deformation for more general parametric functions
- Include option to use Mathematica's numerical integrator
- Implement further variable transformation to tackle singularities very close to pinch singularities