```
filename: Vertex-detailed.cxx
commented printout of gfitter/GSM/Vertex.cxx, J. Haller, 2011-03-03
"Several functions calculated here are not correctly quoted, but copied from zfitter/dizet, "
"others are quoted here "
==========================================================================================
===
Correct references:

D. Bardin, M. Bilenky, S. Riemann, T. Riemann et al.,hep-ph/9709229
"Electroweak Working Group Report" section 4.4: ZFITTER basics, p. 86-91

Dmitri Yu. Bardin (Dubna, JINR), P. Christova (Shoumen U.), M. Jack (DESY, Zeuthen), L. Kalinovskaya
(Dubna, JINR), A. Olchevski (Dubna, JINR & CERN), S. Riemann, T. Riemann (DESY, Zeuthen). DESY-99-070.
Aug 1999. 192 pp.
"ZFITTER v.6.21: A Semianalytical program for fermion pair production in e+ e- annihilation"
Published in Comput.Phys.Commun. 133 (2001) 229-395
e-Print: hep-ph/9908433v3

=========================================================================================
original file begins here, with insertions from zfitter and coments from zfitter group
=========================================================================================

/************************* ************************************************************
 * Project: GSM - Electroweak fitting package                                         *
 * Package: GSM                                                                       *
 * Class  : Vertex                                                                    *
 *                                                                                    *
 * Description:                                                                       *
 *      Auxiliary Theory for Vertex corrections                                       *
 *                                                                                    *
 * Sources                                                                            *
 *      hep-ph/9709229v1, hep-ph/9908433                                              *
 *      dizet6_42.f (ZFitter) line 3376-3480                                          *
 *                                                                                    *
 * Authors (alphabetical):                                                            *
 *      Martin Goebel   <martin.goebel@desy.de>  - DESY, Germany                      *
 *                                                                                    *
 * Copyright (c) 2006:                                                                *
 *      CERN, Switzerland,                                                            *
 *                                                                                    *
 * Redistribution and use in source and binary forms, with or without                 *
 * modification, are permitted according to the terms listed in LICENSE               *
 * (http://mva.sourceforge.net/license.txt)                                           *
 *                                                                                    *
 * File and Version Information:                                                      *
 * $Id: Vertex.cxx,v 1.6 2007/08/27 22:13:07 hoecker Exp $
 ***************************************************************************************/
#include "TMath.h"

#include "Gfitter/GMath.h"
#include "Gfitter/GConstants.h"
#include "Gfitter/GTheory.h"
#include "Gfitter/GTheoryRef.h"
#include "Gfitter/GParameterRef.h"
#include "Gfitter/GReference.h"

#include "GSM/GSMMath.h"
#include "GSM/Vertex.h"

using namespace Gfitter;
using std::complex;

ClassImp(GSM::Vertex)

GSM::Vertex::Vertex()
   : Gfitter::GAuxTheory(),
     m_isUpToDate_Update( kFALSE )
{
   SetTheoryName( GetName() );
   SetExistDerivative( kFALSE );

   BookParameter( "MZ"      , & p_MZ );
```

```
   BookParameter( "mt"      , & p_mt );
   BookTheory   ( "GSM::MW" , & t_MW );
}

void GSM::Vertex::UpdateLocalFlags( GReference& /* ref */ )
{
   m_isUpToDate_Update = kFALSE;
}

void GSM::Vertex::Update()
{
   if (m_isUpToDate_Update) return;

   // now, it is uptodate (I mean... it will be)
   m_isUpToDate_Update = kTRUE;

   Double_t MW      = GetMW();
   Double_t MZ      = p_MZ;
   Double_t mt      = p_mt;

   Double_t mt2     = mt*mt;
   Double_t MW2     = MW*MW;
   Double_t MZ2     = MZ*MZ;
   Double_t R       = MW2/MZ2;

   Double_t Sqr     = TMath::Sqrt(4.0*R - 1.0);
   Double_t Atan    = TMath::ATan(Sqr/(2.0*R - 1.0));
   complex<Double_t>  I (0,1.0);

   // Vertex functions in the limit of vanishing fermion masses
   // for Z decay
   // eq. hep-ph/9709229v1 (265)
   // Spence( 1 ) = Li2( 1 ) = Zeta2
   m_V1ZZ           = - 5.5 - 8.0*( GMath::Zeta2() - TMath::DiLog(2.0) ) + I*imag(V1b(-MZ2, MZ2));

    "-----------------------------------------------------------------------------------------------------
    -"
"   comment from dizet l. 3139 ff.:"
The m_V1ZZ is not at all " eq. hep-ph/9709229v1 (265)", see p. 88 there.
File:
But it is taken from dizet:

   V1ZZ=-5.5D0-8.D0*(F1-SPENCE(2.D0))

      V1ZIM=DIMAG(XV1B(-AMZ2,AMZ2))
      V1WIM=DIMAG(XV1B(-AMZ2,AMW2))
      XV1ZZ=DCMPLX(V1ZZ,V1ZIM)
      XV1ZW=DCMPLX(V1ZW,V1WIM)

   "=================================================================="


   // eq. hep-ph/9709229v1 (265)
   // see also in hep-ph/9908433 (A.4.33)
   m_V1ZW           = ( -3.5 - 2.0*R - (3.0 + 2.0*R)*TMath::Log(R) - 2.0*(1.0 + R)*(1.0 + R)
                        *(GMath::Zeta2() - TMath::DiLog(1.0 + 1.0/R)) + I*imag(V1b(-MZ2,MW2)) );
    "-----------------------------------------------------------------------------------------------------
    -"
"   comment from dizet l. 3139 ff.:"
The m_V1ZW is close to hep-ph/9709229v1 (265), see p. 88 there, but not taken from there.
"Especially nice: in (265) is a typo: Li2(1 + ~RV) instead of: SPENCE(1.D0+1.D0/R), which is also
correct in  m_V1ZW,"
"see: DiLog(1.0 + 1.0/R)"
The v1b is also taken exactly from dizet, see below.
The  hep-ph/9908433 (A.4.33), see p.176 there, is completely different, in terms of C_0 function.

   SPERR=SPENCE(1.D0+1.D0/R)
   V1ZW=-3.5D0-2.D0*R-(3.D0+2.D0*R)*ALR-2.D0*(1.D0+R)**2*(F1-SPERR)

  "=================================================================="
  // eq. hep-ph/9709229v1 (266)
   // see also in hep-ph/9908433 (A.4.34)
   m_V2ZWW          = ( 2.0/(9.0*R*R) + 43.0/(18.0*R) - 1.0/6.0 - 2.0*R
```

```
                        + (-1.0/(12.0*R*R) - 1.5/R + 7.0/3.0 + 2.0*R)*Sqr*Atan - 2.0*R*(2.0 +
                        R)*Atan*Atan );

   Double_t lam = GMath::IPow(MZ2,2) - 4.0*MW2*MZ2;

    "-------------------------------------------------------------------------------------------------
    -"
  "   comment from  dizet l. 3139 ff.:"
  hep-ph/9908433 (A.4.34) is just zero, so wrong reference; the next formula is just different, in
  terms od C_0
  hep-ph/9709229v1 (266) is just different too
  The v1b is also taken exactly from dizet, see below.

   V2ZWW=2.D0/9.D0/R2+43.D0/18.D0/R-1.D0/6.D0-2.D0*R
     *     +(-1.D0/12.D0/R2-1.5D0/R+7.D0/3.D0+2.D0*R)*SR*AT
     *     -2.D0*R*(2.D0+R)*AT**2

   "=================================================================="
   // for W decay
   // see dizet6_42.f (ZFitter) line 3376-3480

   m_V1WZ  = ( -5.0 - 2.0/R + (3.0 + 2.0/R)*TMath::Log(R)
               -2.0*GMath::IPow((1.0+R)/R,2)*(GMath::Zeta2() - TMath::DiLog(1.0 + R)) );
   m_V2WWZ = ( -9/(4.0*R) -1/(12.0*R*R) + 23/18.0
               + (0.5/R - 0.75/(R*R) - 1/(24.0*GMath::IPow(R,3)) + 1.0)*TMath::Log(R)
               - real(GSMMath::L(-MW2,MW2,MZ2))*(5/(6.0*R) + 1/(24.0*R*R) + 0.5)/MW2
               + (0.5 + 1.0/R)*lam*GMath::IPow( real(GSMMath::J(-MW2,MW2,MZ2)), 2 )
               - (0.5 + 1.0/R)*GMath::IPow( TMath::Log(R), 2 ) );
  "----------------------------"
"  from zfitter copied:"

    ALAM=AMZ2*AMZ2-4.D0*AMW2*AMZ2
      V1WZ=-5.D0-2.D0/R+(3.D0+2.D0/R)*ALR
     *    -2.D0*R1W2/R2*(SPENCE(1.D0)-SPENCE(R1W))
      V2WWZ=-9.D0/4.D0/R-1.D0/12.D0/R2+23.D0/18.D0
     *     +(1.D0/2.D0/R-3.D0/4.D0/R2
     *     -1.D0/24.D0/R/R2+1.D0)*ALR
     *     -DREAL(XL(-AMW2,AMW2,AMZ2))
     *     *(5.D0/6.D0/R+1.D0/24.D0/R2+1.D0/2.D0)/AMW2
     *     +(1.D0/2.D0+1.D0/R)*ALAM*DREAL(XJ(-AMW2,AMW2,AMZ2))
     *     *DREAL(XJ(-AMW2,AMW2,AMZ2))-(1.D0/2.D0+1.D0/R)*ALR*ALR

   "=================================================================="
   // top quark additions
   Double_t RtW      = mt2/MW2;
   Double_t RtW1     = RtW - 1.0;

   Double_t J0W = 0, S3W = 0, S30W = 0;
   Double_t J0t = 0, S3t = 0, S30t = 0;

   // s = -MZ2
   GSMMath::S3Wana( mt2, MW2, -MZ2, J0W, S3W, S30W);
   GSMMath::S3Wana( MW2, mt2, -MZ2, J0t, S3t, S30t);

   // Information to this point in hep-ph/9709229 page 88-89,
   // hep-ph/9908433 page 175-177
   // and dizet6_42.f (ZFitter) line 3376-3480
   Double_t WWv2  = ( - 2.0*R*(2.00 + R)*MZ2*(S3W - S30W) + RtW
                      *((3.0*R*R + 2.5*R - 2.0 - (2.0*R - 0.5)*RtW +R*(0.5 - R)*RtW*RtW)*MZ2*S3W
                        -(R + 1.00 - (0.5 - R)*RtW)*(J0W - 2.0) +(2.0*R + 3/(2.0*RtW1*RtW1)
                                                                 - 2.0/RtW1 + 0.5 - (0.5 -
                                                                 R)*RtW)*TMath::Log(RtW)
                        -(R + 3/(2.0*RtW1) + 0.75 - (0.5 - R)*RtW) + 0.25/R*(J0W - 3.0)*1.0) );
   Double_t WWv11 = ( + 2.0*(1.0 + R)*(1.0 + R)*MZ2*(S3t-S30t) + (2.0*R + 3.0)*(J0t + TMath::Log(RtW) +
   TMath::Log(R))
                      - RtW*( R*(3.0*R + 2.0 - RtW - R*RtW*RtW)*MZ2*S3t + (R + 0.5 + R*RtW)*(J0t +
                      TMath::Log(RtW) - 2.0)
                              - (2.00*R + 3/(2.0*RtW1*RtW1) - 2.0/RtW1 + 0.5 + R*RtW)*TMath::Log(RtW)
                              + R + 3/(2.0*RtW1) + 5/4.0 + R*RtW ) );
   Double_t WWv12 = ( -RtW*(R*(2.0 + R - 2.0*R*RtW + R*RtW*RtW)*MZ2*S3t
                            -(0.5 - R + R*RtW)*(J0t + TMath::Log(RtW) - 1.0) + R*RtW*TMath::Log(RtW) )
                            );
```

```
   Double_t v_tb  = 1;

   m_Vtb          = v_tb*v_tb*( R*WWv2 - 0.5*(1.0 - 2.0*(1.0 - R)*(2/3.0))*WWv11 - 0.5*WWv12 );

   // now parameters are up-to-date
   SetUpToDate();
}
    "-----------------------------------------------------------------------------------------------
    -"
"   comment:"
hep-ph/9908433 page 175-177,  A.4.39 - A.4.41 are different, in terms of C_0.
hep-ph/9709229 page 88-89, eq. 168-170 are different realizations, close to the old paper NPB.
 In gfitter is just the code of zfitter.



   "================================================================="
// hep-ph/9709229v1 (265) - (267)
// deal with the imaginary part
complex<Double_t> GSM::Vertex::V1b( Double_t Q2, Double_t MV2 )
{
   Double_t r      = Q2/MV2;
   Double_t Logr   = TMath::Log(TMath::Abs(r));
   Double_t Rev1b  = ( -3.5 + 2.0/r + (3.0-2.0/r)*Logr + 2.0 * GMath::IPow(1.0-1.0/r,2) *
                       (TMath::DiLog(1.0-r) - GMath::Zeta2()) );
   Double_t Aiv1b = 0;

   if (Q2 < 0) Aiv1b = TMath::Pi()*(-3.0+2.0/r+2.0*GMath::IPow((1.0-1.0/r),2)*TMath::Log(1.0-r));
   complex<Double_t> v1b(Rev1b,Aiv1b);

   return v1b;
}
"-------------------------------------------------------------
"from dizet l. 926 ff.:"
The coding is just from zfitter.
hep-ph/9709229v1 (265) - (267), see p.88,  gives the complex functions, but the working out of imag.
part is from the zfitter code.

 REV1B=-3.5D0+2.D0/AL+(3.D0-2.D0/AL)*ALA
     *     +2.D0*(1.D0-1.D0/AL)**2*(SPENCE(1.D0-AL)-F1)
      AIV1B=0.D0
      IF(Q2.LT.0.D0)
     &AIV1B=PI*(-3.D0+2.D0/AL+2.D0*(1.D0-1.D0/AL)**2*LOG(1.D0-AL))
      XV1B=DCMPLX(REV1B,AIV1B)


"============  end of file  ======="
```