TOPAZ0 4.4 - Summary of upgrading *

G. Montagna, O. Nicrosini, G. Passarino and F. Piccinini

August 26, 1999

Abstract

Version 4.4 of `TOPAZ0` has been created in the period October 1998 - February 1999 in order to achieve the following goals:

- Inclusion of NLO corrections into the calculation of Realistic Observables (RO),
- creation of a new branch where RO are computed directly in terms of Pseudo-Observables (PO).

The version marked 26/8/1999 will be the last of the 4.4 family (modulo eventual bugs). The new 4.5 version will contain the full upgrading for LEP2 energies.

$\bowtie$

1 Changes in version 4.4

Added 23/6/99 A popular request always was to get rid of `REAL*16` in `TOPAZ0`. A *new* version has been created which is pure `REAL*8`. At present, several tests have been performed, all positive. Please **report** eventual problems to

- (`giampiero@to.infn.it`),

should you decide to use it.

1.1 *Miscellanea*

16/3/1999 New, the code runs regardless of whether s_0 or $E_{\rm th}$ (effectively) impose an acol cut tighter than the value acol-cut the users specify. (M.Grunewald)

21/5/1999 New,FSR has been updated for the setup where both fermion and anti-fermion are within the same angular acceptance. Changes are in function TWOC. (M.Grunewald, J.Mnich)

*Written by G. Passarino

3/6/1999 A small change in subroutine `TEPAIRS`, the arrays `WRKSTRV,WRKSTRS` and `WRKSTRH`, only used in the calls to `D01EAF` have been reduced to one (`WRKSTR`). (M.Grunewald, J.Mnich)

21/6/1999 The code runs regardless of whether s_0 or $E_{\rm th}$ (effectively) impose an acol cut tighter than the value acol-cut the users specify. A bug on tau variables, introduced on 16/3/1999 is fixed.

23/6/1999 There was a complaint about getting the following message:

```
** MAXCLS too small to obtain required accuracy
** ABNORMAL EXIT from NAG Library routine D01EAF: IFAIL =      1
** NAG soft failure - control returned
 D01EAF CALLED BY 'ROUTINENAME'
```

This is only a soft exit from `NAGLIB` with control returned and no effect, it is useful for the author. A new flag has been introduced that prevents `NAGLIB` from printing messages (it corresponds to `NAGLIB` flag `IFAIL`= 1, instead of `IFAIL`= -1)

30/6/1999 Compilation bugs fixed (J.Mnich)

26/7/1999 `ONP = I`, hard pairs are improved with third order radiator, selected by `ORISPP = 'T'`.

24/8/1999 Little fixes/improvements in TWEAK and TCINFO (S.Wynhoff)

25/8/1999 Singlet pairs for extrapolated observables included by setting `OSING = SP` (default is `OSING = VV`, i.e. no singlet pairs). Since singlet pairs, as well as multi-peripheral contributions, seem to be subtracted from the experimental data one should *not* include them, unless explicitly requested. Furthermore, τ-pairs have been added in mode `ONP = 'I'`, showing a very small effect.

$\bowtie$

2 Main features

Moreover, there are few upgradings also in the old parts:

$\boxed{\textit{New}}$ The Initial/Final QED interference has been extended to properly take into account the exact photon phase space, also in presence of realistic cuts. Moreover, Initial/Final QED interference is also available for $\overline{q}q$ final states.

$\boxed{New}$ The structure of the code is now governed by a main decision branch

$$\texttt{OMODES} = \begin{cases} \texttt{CALC} & \text{computes RO in terms of} m_t, M_H, \alpha_S(M_Z^2)^\dagger \\ \texttt{FITE} & \text{computes RO in terms of PO}^1 \\ \texttt{FITC} & \text{computes RO in terms of PO}^2 \end{cases}$$

† This is the old branch of `TOPAZ0` where, however, the calculation of RO has been upgraded with the inclusion of NLO corrections. The upgrading took place for both extrapolated (s' or $M^2(\overline{f}f)$ cuts) RO and RO with kinematical and acceptance cuts.

3 Flags

Here we recall once more the meaning of the various flags.

All the internal flags of the program are hard coded and initialized in `SUBROUTINE TINIT`, which is called by the job-control. file. For more details concerning flags already present in earlier versions, the reader is referred to the documentation of the previous releases.

<u>`ACOLL`</u> : array dimensioned as `ACOLL(3)`; it enters the maximum acollinearity angle (deg) for electrons, μ and τ, respectively; `ACOLL(2)` and `ACOLL(3)` used only if `OEXT = C`;

<u>`DEL`</u> : the semi-aperture of the electromagnetic jet (deg); used only if `OCN = 'Y'`.

<u>`E0`</u> : array dimensioned as `E0(3)`; `E0(1)` is the minimum energy (GeV) of the final-state electrons, used if `OTHRE = E`; `E0(2)` and `E0(3)` are the same as `E0(1)` for μ and τ respectively, used if `OTHRMT = E`;

<u>`OAAS`</u> : the scale in α_s in mixed $\mathcal{O}(\alpha\alpha_s)$ corrections;

<u>`OAL`</u> : $\boxed{New}$ selects the value of $\alpha_{em}^{-1}(M_Z^2)$: There is an agreement in our community on using the following strategy. The default `OAL = 'D'` defines

$$\alpha(M_Z) = \frac{\alpha(0)}{1 - \Delta\alpha^{(5)}(M_Z) - \Delta_{\rm top}(M_Z) - \Delta_{\rm tb}^{\alpha\alpha_S}(M_Z)}, \tag{1}$$

where one has $\Delta\alpha^{(5)}(M_Z) = \Delta\alpha_{\rm lept} + \Delta\alpha^{(5)}_{\rm had}$. In all codes the input should be $\Delta\alpha^{(5)}_{\rm had}$, while the calculation of the top contribution to $\Delta\alpha$ is left for the code.

TOPAZO includes in $\Delta\alpha_{\rm lept}$ the recently computed $\mathcal{O}\left(\alpha^3\right)$ terms. and have as default $\Delta\alpha^{(5)}_{\rm had} = 0.0280398$. As explained above this parameter can be reset by the user. Using the default one gets $1/\alpha^{(5)}(M_z) = 128.877$, to which one must add the top contribution and the $\mathcal{O}\left(\alpha\alpha_S\right)$ corrections induced by the $t-b$ doublet. $\boxed{\textit{New}}$ If `OAL = 'N'` then a new value for $\Delta\alpha^{(5)}_{\rm had}$ is selected through the calling procedure `CALL TCAQED(ALPHANEW)`.

<u>`OCHAN`</u> : selects the full Bhabha scattering matrix element `(F)` (*default*) or only the s-channel part `(S)`;

<u>`OCN`</u> : includes `(Y)` or not `(N)` the contribution of electromagnetic jets for final-state calorimetric electrons;

<u>`OCREE`</u> : if `OEXT = E` and `OCHAN = S`, i.e. for s-channel electrons, it allows for an s' cut;

<u>`OCUT`</u> : array dimensioned as `OCUT(NRTS)`; selects a cut on s': no cut `(NC)`, cut on hadronic channels only `(HC)`, cut on all channels `(FC)`. Here s' is the invariant mass of the e^+e^- system after initial-state radiation;

<u>`OCUTES`</u> : array dimensioned as `OCUTES(NRTS)`; if `OCREE = Y`, it allows `(C)` or not `(N)` for an s' cut for each energy point;

<u>`OCUTF`</u> : as `OCUT`, but for final-state fermions invariant mass;

<u>`OCUTS`</u> : array dimensioned as `OCUTST(NRTS)`; if `OFB = Y` selects `(Y)` or not `(N)` a cut on the invariant mass after initial-state radiation for μ and τ cross sections;

<u>`OEXT`</u> : selects the extrapolated `(E)` or cut `(C)` branch for μ and τ. With `OEXT = 'P'` only pseudo-observables are computed and control is returned;

<u>`OFB`</u> : selects `(Y)` or not `(N)` different treatment of cuts for cross sections and forward-backward asymmetries for μ and τ;

<u>`OFS`</u> : selects the treatment of higher order final-state QED corrections; `(D)`, giving completely factorized FS QED corrections or `(Z)`, giving factorized leading terms and non-leading contributions summed up;

`OHC` : next-to-leading and higher order hard photon contribution included `(Y)` or not `(N)`;

`OIFAIL` : $\boxed{\textit{New}}$ A new flag has been introduced that prevents `NAGLIB` from printing messages, `OIFAIL = 'N'`, defualt (it corresponds to `NAGLIB` flag `IFAIL`= 1, instead of `IFAIL`= -1);

`OIMAG = 'N'` : (Default `= 'Y'`) $\boxed{\textit{New}}$ does not include the imaginary parts of $\alpha(s)$ and of the form factors.

`OLHO = 'Y'` : $\boxed{\textit{New}}$ includes in $\Delta\alpha_{\rm lept}$ the recently computed $\mathcal{O}\left(\alpha^3\right)$ terms;

`ONIF` : array dimensioned as `ONIF(NRTS)` selecting `(Y)` or not `(N)` the effect of initial-finale state QED interference for each energy; for leptons only. Default is `'N'`;

`ONIFH` : $\boxed{\textit{New}}$ array dimensioned as `ONIFH(NRTS)` selecting `(Y)` or not `(N)` the effect of initial-finale state QED interference for each energy and for $\overline{q}q$ final states. Default is `'N'`;

`ONP` : array dimensioned as `ONPT(NRTS)` selecting `(Y)` or not `(N)` the pair production correction for each energy. The *improved* formulation (see section 8) is the new default, i.e. default is `'I'`. Note that the *improved* formulation of pair production is active only for the extrapolated observables;

`OPREC` : $\boxed{\textit{New}}$ Default is `OPREC= 'L'` and implies that call to `NAGLIB` subroutine D01EAF is performed with `MNCLS= 0`, i.e. zero for the minimum number of calls. Instead, `OPREC= 'H'` sets `MNCLS= 1000` in some of `TOPAZ0` subroutines. In order to gain CPU-time the combination `OPREC= 'L'` with a relatively high `SE`. e.g. `SE` $\geq$ `1.D0`, is recommended;

`ORAD` : selects the type of next-to-leading and higher order hard photon contributions to be included `(A,D,E,F,Y)`. Only the last two should be used: `F` is the *recommended* choice; $\boxed{\textit{New}}$ `Y` has been implemented after version 4.3 and uses the order α^3 YFS radiator;

`ORISPP` : $\boxed{\textit{New}}$ selects the second(third) order radiator in the QED convolution of the hard initial state pair production, `ORISPP = 'S(T)'` (`'T'` is default);

<u>`OPO = 'ORT'`</u>]: $\boxed{\textit{New}}$ The PO $A_{\rm FB}^{0,e,\mu,\tau}$ are computed in all branches *WITHOUT* imaginary parts (while the PO $R_{e,\mu,\tau}$ always include mass effects). With the choice `OPO = 'ICO'` imaginary parts are properly accounted for, everywhere. The default is `OPO = 'ORT'`;

<u>`OSING`</u> : $\boxed{\textit{New}}$ Default is `'VV'`, i.e. exclusion of singlet pairs. They can be included, for extrapolated observables, with `OSING = 'SP'` (which requires `ONP = 'I'`;

<u>`OSP`</u> : a remnant of the past, used to compute extrapolated lineshapes and asymmetries with cuts, default is `'N'` and should not be changed;

<u>`OTHRE`</u> : selects invariant mass `(M)` or energy `(E)` threshold for electrons;

<u>`OTHRMT`</u> : if `OEXT = C`, selects invariant mass `(M)` or energy `(E)` threshold for μ and τ;

<u>`OWBOX`</u> : weak boxes included `(Y)` or not `(N)`. Already off the Z-resonance the *correct* choice is `Y`;

<u>`OWEAK`</u> : residual weak corrections: running `(R)` or fixed `(F)`; the **`recommended`** choice is `R`;

<u>`OUO`</u> : includes `(S)` the NLO two-loop $\mathcal{O}\left(\alpha^2 m_t^2\right)$ corrections. It is the *recommended* choice;

<u>`OVNAL`</u> : $\boxed{\textit{New}}$ if `OAL = N`, enters the user-defined value $\Delta\alpha^{(5)}_{\rm had}$ = `OVNAL`;

<u>`OXCUT`</u> : if `OCUT = Y`, it enters the value of the s' cut, s_0/s;

<u>`OXCUTES`</u> : if `OCUTES(I) = C`, it enters the s_0/s threshold;

<u>`OXCUTF`</u> : if `OCUTF = Y`, it enters the cut on the value of invariant mass squared of the final-state fermions;

OXCUTS : if `OCUTS = Y`, it enters the value of the s' cut, s_0/s;

SC : if `OAAS = Y`, defines the scale in α_s in mixed $\alpha\alpha_s$ corrections;

SOCUT : array dimensioned as `SOCUT(3)`; `SOCUT(1)` is the minimum invariant mass (GeV) of the final-state electrons, used if `OTHRE = M`; `SOCUT(2)` and `SOCUT(3)` are the same as `SOCUT(1)` for μ and τ respectively, used if `OTHRMT = M`;

SEPA : $\boxed{\textit{New}}$ A new flag has been inserted corresponding to the value of Δ, the soft/hard separator in pairs. It helps in testing stability (see section 8). Tests have been performed so you are not supposed to change it;

THMIN : array dimensioned as `THMIN(3)`; it enters the minimum scattering angle (deg) for electrons, μ and τ, respectively (*symmetrical angular acceptance* is understood); `THMIN(2)` and `THMIN(3)` used only if `OEXT = C`;

THMINP : the same as `THMIN`, for the antiparticles;

XMED : fictitious separator for 1-dimensional integrations to improve numerical convergence;

ZPCUT : the minimum fraction of squared invariant mass of the final state after radiation of the additional initial-state pair;

4 I/O

`SUBROUTINE TCINFO(MODE)` $\boxed{\textit{New}}$ (courtesy of M. Grünewald, further modification suggested by S.Wynhoff) For users and within fit programs it is very useful to be able to dump all flags and cut settings to check whether they are what the users want. `MODE = 0` dump flags, while `MODE = 1` dump cuts. The output will be something like this:

```
TOPAZ0 flag values:

General flags:
OEXT    :     E  OTHRMT :     E  OTHRE   :     E  OCHAN   :     S
OAL     :     D  OWEAK  :     R  OFB     :     N  OHC     :     Y
```

```
ORAD   :    F  OWBOX  :    Y  OPWM   :    F  OFS    :    Z
OCN    :    Y  OBHABHA:    N  OREST  :    E  OINDX  :   NO
OLHO   :    Y  OMA    :    N  OIMAG  :    Y  OMODES : CALC
OFM    :    U  OPO    :  ORT  ONPAR  :   FP  OAAS   :    N
OSP    :    N  OCREE  :    N  OPREC  :    L  OIFAIL :    N
ORISPP :    T  OSING  :   VV

General flag arrays:
ONP    :  N  N  N  N  N
OCUTS  :  N  N  N  N  N
OCUTES :  N  N  N  N  N
ONIF   :  Y  Y  Y  Y  Y
ONIFH  :  Y  Y  Y  Y  Y
OCUT   : FC FC FC FC FC
OCUTF  : NC NC NC NC NC

Special flags:
OU0 : S  OU1 : Y  OU2 : N  OU3 : Y  OU4 : N
OU5 : Y  OU6 : Y  OU7 : N  OU8 : C

Miscellaneous:
SC  :     1.000000  SE  :     0.100000  XMED:     0.980000
SEPA:    16.000000


TOPAZ0 cut values:

General cuts:
OXCUT  :     0.01000000  OXCUTF :     0.01000000
OXCUTES:     0.00000000  OXCUTS :     0.00000000
ZPCUT  :     0.01000000  DEL    :     0.50000000

Lepton:       Electron           Muon             Tau
SOCUT       0.00000000      0.00000000      0.00000000
THMIN      44.00000000     40.00000000     40.00000000
THMINP      0.00000000      0.00000000      0.00000000
ACOLL      10.00000000     10.00000000     10.00000000
E0          1.00000000      1.00000000      1.00000000
```

5 Defaults

To summarize the internal default we recall that

```
OMODES= 'CALC'
ONPAR= 'FP'
OFM= 'U'
OWEAK= 'R'
OPWM= 'F'
OWBOX= 'Y'
OHC= 'Y'
ORAD= 'F'
OAL= 'D'
OAAS= 'N'
OCN= 'Y'
OFS= 'Z'
OFB= 'N'
OBHABHA= 'N'
OREST= 'E'
OLHO= 'Y'
OMA= 'N'
OIMAG= 'Y'
OPO= 'ORT'
OINDX= 'NO'
OPREC= 'L'
ONP(1)= 'N'          ! NO initial state Pair Production
ONIF(1)= 'N'         ! NO Initial/Final QED interference for leptons
ONIFH(1)= 'N'        !           "                        for quarks
OCHAN= 'S'           ! electron only in s-channel
OXCUT= 0.01D0        ! s' cut of 0.01 s
ZPCUT= 0.01D0        ! pair production cut
OCUT(1)= 'FC'
OCUTF(1)= 'NC'
OIFAIL= 'N'
ORISPP= 'T'
OSING= 'VV'
OU0= 'S'
OU1= 'Y'
OU2= 'N'
OU3= 'Y'
OU4= 'N'
OU5= 'Y'
OU6= 'Y'
OU7= 'N'
OU8= 'C'
```

Moreover, for `OEXT = 'E'` the default is `OCUT(1) = 'FC'`, i.e. s'-cut for all channels. For `OEXT = 'C'` the default is `OCUT(1) = 'HC'`, i.e. hadrons are with a s'-cut. Finally, the default is `OCUTF(1) = 'NC'`, i.e. no cut on $M^2(\overline{f}f)$.

6 Subroutine TOPAZO

The main subroutine in the code is

- `SUBROUTINE TOPAZ0(SE,NRTS,RTS,ZMT,TQMT,HMT,ALST,OTPPO,OTPRO,OTPDRO).`

New The new entry is `OTPDRO(5,9,NTRS)`. Probably it is of no help for the user but in here are recorded precious informations about de-convoluted observables. It is of invaluable help in cross-checking with other codes.

ARRAY `OTPPO(34)` contains an extended list of PO:

```
*     PO ARE
*
*     W MASS      *     G_NU        *     G_ELECTRON       *     G_MUON
*     G_TAU       *     G_UP        *     G_DOWN(STRANGE)  *     G_CHARM
*     G_BOTTOM    *     SIN^2(E)    *     SIN^2(B)         *     A_FB(L)
*     A_LR        *     TOTAL WIDTH *     G_H/G_E          *     SIGMA0_H
*     R_B         *     A_FB(B)     *     HADRONIC WIDTH   *     INVISIBLE
*     A_FB(C)     *     R_C         *     A_LR(B)          *     A_LR(C)
*     SIN^2(C)    *     RHO(E)      *     RHO(C)           *     RHO(B)
*     1/ALPHA^5   *     1/ALPHA     *     SIGMA0_L         *     ALPHA_S(M_T)
*     G_H/G_MU    *     G_H/G_TAU
```

The scaling factor for numerical integration error is (after version 4.1) argument of the subroutine. This is important, since the needed CPU time is largely due to the choice of `SE`. Recommended is `SE = 1.D0`, especially when one computes all quantities in one stroke, but on a single-channel calculation it is feasible (and somewhat recommended) to reach even better precision, as with `SE = 0.1D0`. Remember that, far away from the Z-resonance all calculations are longer, much longer. For fitting procedures it might be advisable a lower precision, something like `SE = 10.D0` or even higher. One should remember that for extrapolated observables there is a one-dimensional integral so we start with a requested relative error `REQ = 1.D-5*SE`. For observables with cuts we go up to three-dimensional integrals so we request `REQ = 1.D-4*SE` or even `REQ = 1.D-3*SE` (for 3-dim).

- The main branch in `TOPAZ0` is governed by the flag `OEXT`: selects the extrapolated `(E)` or cut `(C)` branch for μ and τ. It is understood that `OEXT`

`= 'P'` will be used only for pseudo observables. After version 4.1 we have 3 additional flags. They are: `OBHABHA, OREST, OINDX`. The philosophy is as follows.

$$\boxed{\texttt{OEXT = 'E'}} \quad \to \quad \boxed{\texttt{OBHABHA = 'N'.or.'B'.or.'R'}} \tag{2}$$

1. `OBHABHA = 'N'` (default) everything is as before, i.e. Bhabha with cuts and the rest extrapolated with a, possible, s'-cut.

2. `OBHABHA = 'B'` Only Bhabha scattering (cross-section and forward-backward asymmetry) is computed.

3. `OBHABHA = 'R'` In this case `OREST = 'E'` and Bhabha scattering is not computed. One has the following options:

   a) `OINDX = 'MU'` Only the cross-section and the forward-backward asymmetry for muons (s'-cut) are computed.

   b) `OINDX = 'TA'` Only the cross-section and the forward-backward asymmetry for taus (s'-cut) are computed.

   c) `OINDX = 'BQ'` Only the cross-section and the forward-backward asymmetry for b-quarks (s'-cut) are computed.

   d) `OINDX = 'CQ'` Only the cross-section and the forward-backward asymmetry for c-quarks (s'-cut) are computed.

   e) `OINDX = 'HA'` Only the total cross-section for hadrons (s'-cut) is computed.

$$\boxed{\texttt{OEXT = 'C'}} \quad \to \quad \boxed{\texttt{OBHABHA = 'N'}} \tag{3}$$

Again, everything as before. But

$$\boxed{\texttt{OEXT = 'C'}} \quad \to \quad \boxed{\texttt{OBHABHA = 'R'}} \quad \to \quad \boxed{\texttt{OREST = 'C'}} \tag{4}$$

In this case `OINDX` can only assume two values `'MU'.or.'TA'` and one computes the cross-section and the forward-backward asymmetry for muons or taus with kinematical cuts. Now few examples:

```
OEXT= 'E'
CALL TINIT(NRS,OEXT)
OVAL= 'B'
OF7= 'OBHABHA'
CALL TCFLAG(OF7,OVAL)
```

will give you only Bhabha.

```
      OEXT= 'E'
      CALL TINIT(NRS,OEXT)
      OVAL= 'R'
      OF7= 'OBHABHA'
      CALL TCFLAG(OF7,OVAL)
      OV2= 'MU'
      OF5= 'OINDX'
      CALL TCFLAG(OF5,OV2)
```

Will give you only extrapolated muons.

```
      OEXT= 'C'
      CALL TINIT(NRS,OEXT)
      OVAL= 'R'
      OF7= 'OBHABHA'
      CALL TCFLAG(OF7,OVAL)
      OV2= 'MU'
      OF5= 'OINDX'
      CALL TCFLAG(OF5,OV2)
      OVAL= 'C'
      OF5= 'OREST'
      CALL TCFLAG(OF5,OVAL)
```

Will give you only muons with kinematical cuts. Here I want to remind you how to change cuts. For instance, the following calling sequence

```
*     CUTS FOR BHABHA ARE CHANGED  (FOR MUONS 2, FOR TAUS 3)
      IND= 1
*     THIS WILL SET THE ANGLE FOR E^-
      THMN= 50.D0
*     THIS WILL SET THE ANGLE FOR E^+
      THMNP= 50.D0
*     THIS WILL SET ACOLLINEARITY
      ACLL= 30.D0
*     THIS WILL SET ENERGY THRESHOLD
      EOC= 1.D0
*     THE REST IS UNCHANGED
      DMY= -1.D0
*     CHANGE IS PERFORMED
      CALL TCUTSET(IND,DMY,THMN,THMNP,ACLL,EOC,DMY,DMY,DMY,DMY)
```

is used to change angles, acollinearity and energy thresholds. Alternatively

```
      IND= 1
      DMY= -1.D0
      XCUT= 0.8D0
      CALL TCUTSET(IND,DMY,DMY,DMY,DMY,DMY,XCUT,DMY,DMY,DMY)
```

will set the s'/s cut to 0.8 for all extrapolated quantities. More,

```
IND= 1
DMY= -1.D0
XPCUT= 0.8D0
CALL TCUTSET(IND,DMY,DMY,DMY,DMY,DMY,DMY,DMY,XPCUT,DMY)
```

will set the $z_{\rm min}$ cut for initial pair production to $z_{\rm min} = 0.8$. Etc, etc, etc.

7 MI branch

If `OMODES.NE.CALC` then each quantity can be given as the sum of a Standard Model Independent (SMI) component $\oplus$ its complement, Standard Model Dependent (SMD), $\overline{\rm SM}$. 5 or 9 parameters are input, according to `ONPAR = 'FP','NP'`.

- For `ONPAR = 'FP'` 5 parameters are Input: $\{{\rm PO}\} = M_{_Z}, \Gamma_{_Z}, \sigma^0_{\rm h}$, the ratio R_l and the asymmetries $A^{0,l}_{\rm FB}$.
- For `ONPAR = 'NP'` 9 parameters are Input: $\{{\rm PO}\} = M_{_Z}, \Gamma_{_Z}, \sigma^0_{\rm h}$, the ratios $R_{e,\mu,\tau}$ and the asymmetries $A^{0,e,\mu,\tau}_{\rm FB}$.

If one gives nine parameters, i.e. without assuming lepton universality, then the R are defined as the ratios of the physical partial width absrbing everything, also kinematic mass effects ($\Gamma_\tau < \Gamma_\mu < \Gamma_e$); thus the SM calculation of R_τ comes out 0.23% larger than R_e and $R_\mu.The A^{0,l}_{\rm FB}$, on the other hand, are all equal, as they are defined via the effective couplings where kinematic mass effects are no longer relevant.

1) Here `TOPAZ0` will compute $\sigma_{\rm had}, \sigma_{e,\mu,\tau}$ and $A^{e,\mu,\tau}_{\rm FB}$ in terms of $\{{\rm PO}\}$ for an extrapolated setup where only an s'-cut is allowed. The e^+e^- channel is understood to be the s-channel alone, i.e. after t-channel subtraction.
2) Here `TOPAZ0` will compute (in terms of $\{{\rm PO}\}$) $\sigma_{\rm had}$ for an extrapolated setup where only an s'-cut is allowed. Instead, for $\sigma_{e,\mu,\tau}$ and $A^{e,\mu,\tau}_{\rm FB}$ cuts are allowed. The e^+e^-channel can be s-channle alone or the complete s+t-channel. For the latter the s-channel component is computed starting from PO and $t-t$ and $s-t$ are part of the SM remnant.

In both cases the SMD part is included at fixed $m_t, M_{_H}$ and $\alpha_{_S}(M^2_{_Z})$. For $M_{_Z}$ two alternatives are foreseen: either $M_{_Z}$ is kept fixed in the $\overline{\rm SM}$ component while it is varied in the SMI component (`OFM = 'F'`) or it is constantly updated also in $\overline{\rm SM}$ (`OFM = 'U'`). CPU time is not depending sizably on this choice so that we recommend `OFM = 'U'` (default). $\boxed{\textit{Warning}}$ If you call again subroutine `TOPAZ0` with a new set of energy points you must *NOT* initialize `OFM = 'F'`, otherwise vertices, boxes etc. will not be computed at the right energies.

In order to use the new branch `OMODES = 'FITE','FITC'` one has to be sure of the following setting

$$\begin{array}{r} \mathrm{OEXT.EQ.'E'} \\ \mathrm{OWEAK.EQ.'R'} \\ \mathrm{OCUTF(1).EQ.'NC'} \end{array} \tag{5}$$

In particular, for this branch, `TOPAZO` does not allow to cut on $M^2(\overline{f}f)$ but only on s'.

The following example helps in understanding the calling sequence:

```
      IMPLICIT REAL*8(A-H,O-Z)
      PARAMETER(NX=5,NY=6,NPO=34,NRO=26*NX,NFIT=7*NY)
      CHARACTER*1 OEXT,OVAL
      CHARACTER*3 OF3
      CHARACTER*4 OF4
      CHARACTER*6 OF6
*
      COMMON/TK/IFK
      COMMON/TFITO/APO(9)
      COMMON/TFIT/FIT(NFIT)
*
      DIMENSION RS(NX)
      DIMENSION OTPPO(NPO),OTPRO(NRO),OTPDROD(5,9,NX)
*
      IFK= 0
      NRS= 5                      ! NUMBER OF ENERGIES
      NRST= NRS+1                 !
      ZM= 91.1867D0               !
      TQM= 173.8D0                ! SM INPUT PARAMETERS
      ALS= 0.119D0                !
      HM= 100.D0                  !
*
      RS(1)= ZM-3.D0              !
      RS(2)= ZM-1.8D0             !
      RS(3)= ZM                   ! ENERGIES
      RS(4)= ZM+1.8D0             !
      RS(5)= ZM+3.D0              !
*
      APO(1)= ...                 !
      APO(2)= ...                 !
      APO(3)= ...                 !
      APO(4)= ...                 !
      APO(5)= ...                 ! INPUT PO
```

```
      APO(6)= ...                  !
      APO(7)= ...                  !
      APO(8)= ...                  !
      APO(9)= ...                  !
*
      OEXT= 'E'                    ! EXTRAPOLATED SETUP, S'-CUT
      CALL TINIT(NRS,OEXT)         !
      OF4= 'FITE'                  !
      OF6= 'OMODES'                ! OMODES IS SET TO MODE 'FITE'
      CALL TCFLAG(OF6,OF4)         !
      OVAL= 'F'                    !
      OF3= 'OFM'                   ! OFM IS SET TO MODE 'F' (FIXED)
      CALL TCFLAG(OF3,OVAL)        !
*
      SE= 1.D0
      CALL TOPAZO(SE,NRS,RS,ZM,TQM,HM,ALS,OTPPO,OTPRO,OTPDRO)
```

Therefore 9 parameters have to be provided, according to the following sequence

$$
\begin{aligned}
\texttt{APO(1)} &= M_z, \\
\texttt{APO(2)} &= \Gamma_z, \\
\texttt{APO(3)} &= \sigma^0_{\mathrm{h}}, \\
\texttt{APO(4,5,6)} &= R_{e,\mu,\tau}, \\
\texttt{APO(7,8,9)} &= A^{0,e,\mu,\tau}_{\mathrm{FB}}.
\end{aligned}
\tag{6}
$$

The output can be downloaded with the following sequence

```
      DO IT=1,7
       DO IF=1,NRST
        K= NRST*(IT-1)+IF
        IF(IT.EQ.1) THEN
         SET(IF)= FIT(K)
        ELSE IF(IT.EQ.2) THEN
         SMUT(IF)= FIT(K)
        ELSE IF(IT.EQ.3) THEN
         STAUT(IF)= FIT(K)
        ELSE IF(IT.EQ.4) THEN
         SHADT(IF)= FIT(K)
        ELSE IF(IT.EQ.5) THEN
         AET(IF)= FIT(K)
        ELSE IF(IT.EQ.6) THEN
         AMUT(IF)= FIT(K)
        ELSE IF(IT.EQ.7) THEN
```

```
    ATAUT(IF)= FIT(K)
   ENDIF
  ENDDO
 ENDDO
```

where one has

$$
\begin{aligned}
\mathrm{SET(IF)} &= \sigma^s_e, \\
\mathrm{SMUT(IF)} &= \sigma_\mu, \\
\mathrm{STAU(IF)} &= \sigma_\tau, \\
\mathrm{SHADT(IF)} &= \sigma_{\mathrm{had}}, \\
\mathrm{AET(IF)} &= A^{s,e}_{\mathrm{FB}}, \\
\mathrm{AMUT(IF)} &= A^{\mu}_{\mathrm{FB}}, \\
\mathrm{ATAUT(IF)} &= A^{\tau}_{\mathrm{FB}},
\end{aligned}
\tag{7}
$$

where the index IF is running over the energies. `IFK` is a counter needed when `OFM.EQ.'F'` is selected.

There is an additional flag: with `OPO = 'ORT'(odox)` the PO $A^{0,e,\mu,\tau}_{\mathrm{FB}}$ are computed in all branches *WITHOUT* imaginary parts (while the PO $R_{e,\mu,\tau}$ always include mass effects). With the choice `OPO = 'ICO'(noclast)` imaginary parts are properly accounted for, everywhere. The default is `OPO = 'ORT'`.

- Also in `OMODES = 'FITE', 'FITC'` mode (with `OEXT` rigorously fixed to `'E'`) one can compute single channel obervables. The following choices are available:

```
OEXT= 'E'
CALL TINIT(NRS,OEXT)
OF4= 'FITE'
OF6= 'OMODES'
CALL TCFLAG(OF6,OF4)
OVAL= 'R'
OF7= 'OBHABHA'
CALL TCFLAG(OF7,OVAL)
OV2= 'EL'                ! 'MU' , 'TA', 'HA'
OF5= 'OINDX'
CALL TCFLAG(OF5,OV2)

OEXT= 'E'
CALL TINIT(NRS,OEXT)
OF4= 'FITC'
OF6= 'OMODES'
CALL TCFLAG(OF6,OF4)
OVAL= 'R'
```

```
      OF7= 'OBHABHA'
      CALL TCFLAG(OF7,OVAL)
      OV2= 'EL'                  ! 'MU' , 'TA'
      OF5= 'OINDX'
      CALL TCFLAG(OF5,OV2)
```

In this case $\sigma^{(s)}(e), A_{\rm FB}^{(s),e}$ or $\sigma(\mu), A_{\rm FB}^{\mu}$ or $\sigma(\tau), A_{\rm FB}^{\tau}$ or $\sigma(\mathrm{had})$ are computed with s'-cut.

```
      OEXT= 'E'
      CALL TINIT(NRS,OEXT)
      OF4= 'FITC'
      OF6= 'OMODES'
      CALL TCFLAG(OF6,OF4)
      OVAL= 'R'
      OF7= 'OBHABHA'
      CALL TCFLAG(OF7,OVAL)
      OV2= 'EL'
      OF5= 'OINDX'
      CALL TCFLAG(OF5,OV2)
```

In this case $\sigma^{(s)}(e), A_{\rm FB}^{(s),e}$ are computed with kinematical and acceptance cuts. $\boxed{\textit{Warning}}$ There is no appreciable gain in CPU time in going to single channel mode.

A simple strategy where one needs a 9 parameter fit and where it is possible to cluster cross-sections and forward-backward asymmetries into to separate sets of energies is available through the following routine `TTHEORY`.

```
      SUBROUTINE TTHEORY(IC,ICN,NRS,RS,ZMT,TQMT,ALST,HMT,PO,RO)
      IMPLICIT REAL*8(A-H,O-Z)
*
      PARAMETER(NX=27,NY=28,NPO=34,NRO=26*NX,NFIT=7*NY)
*
      CHARACTER*1 OEXT
      CHARACTER*4 OF4
      CHARACTER*6 OF6
*
      COMMON/TFITO/APO(9)
      COMMON/TFIT/FIT(NFIT)
*
      DIMENSION RS(NRS),PO(9),RO(ICN,NRS)
      DIMENSION OTPPO(NPO),OTPRO(NRO),OTPDRO(5,9,NX)
```

```
      DIMENSION SET(NY),AET(NY),SMUT(NY),AMUT(NY),STAUT(NY),
     #          ATAUT(NY),SHADT(NY)
*
      NRST= NRS+1
      ZM= ZMT
      TQM= TQMT
      ALS= ALST
      HM= HMT
*
      DO I=1,9
          APO(I)= PO(I)
      ENDDO
*
      OEXT= 'E'
      CALL TINIT(NRS,OEXT)
      OF4= 'FITE'
      OF6= 'OMODES'
      CALL TCFLAG(OF6,OF4)
*
      SE= 1.D2
      CALL TOPAZO(SE,NRS,RS,ZM,TQM,HM,ALS,OTPPO,OTPRO,OTPDRO)
*
      DO IT=1,7
       DO IF=1,NRST
        K= NRST*(IT-1)+IF
        IF(IT.EQ.1) THEN
         SET(IF)= FIT(K)
        ELSE IF(IT.EQ.2) THEN
         SMUT(IF)= FIT(K)
        ELSE IF(IT.EQ.3) THEN
         STAUT(IF)= FIT(K)
        ELSE IF(IT.EQ.4) THEN
         SHADT(IF)= FIT(K)
        ELSE IF(IT.EQ.5) THEN
         AET(IF)= FIT(K)
        ELSE IF(IT.EQ.6) THEN
         AMUT(IF)= FIT(K)
        ELSE IF(IT.EQ.7) THEN
         ATAUT(IF)= FIT(K)
        ENDIF
       ENDDO
      ENDDO
*
      IF(IC.EQ.1) THEN
       DO I=1,NRS
        RO(1,I)= SET(I)
```

```
      RO(2,I)= SMUT(I)
      RO(3,I)= STAUT(I)
      RO(4,I)= SHADT(I)
     ENDDO
    ELSE IF(IC.EQ.2) THEN
     DO I=1,NRS
      RO(1,I)= AET(I)
      RO(2,I)= AMUT(I)
      RO(3,I)= ATAUT(I)
     ENDDO
    ENDIF
*
    RETURN
    END
```

An example of how to call it is the following:

```
    IMPLICIT REAL*8(A-H,O-Z)
    DIMENSION RS(27),PO(9),RO(4,27)
*
    NRS= 27
    IC= 1
    ICN= 4
    ZM= 91.1867D0
    TQM= 173.8D0
    ALS= 0.119D0
    HM= 100.D0
    DO I=1,27
     RS(I)= ZM-1.3D0+(I-1)*0.1D0
    ENDDO
    PO(1)= ZM
    PO(2)= 0.249549E+01
    PO(3)= 0.414761E+02
    PO(4)= 0.207435E+02
    PO(5)= 0.207437E+02
    PO(6)= 0.207906E+02
    PO(7)= 0.160835E-01
    PO(8)= 0.160835E-01
    PO(9)= 0.160835E-01
*
    CALL TTHEORY(IC,ICN,NRS,RS,ZM,TQM,ALS,HM,PO,RO)
*
    DO I=1,NRS
     PRINT 1,RS(I),RO(1,I),RO(2,I),RO(3,I),RO(4,I)
```

```
      ENDDO
    1 FORMAT('  E_CM (GEV)            = ',E20.5//
     #       '  SIGMA(E)   (NB)       = ',E20.7/
     #       '  SIGMA(MU)  (NB)       = ',E20.7/
     #       '  SIGMA(TAU) (NB)       = ',E20.7/
     #       '  SIGMA(HAD) (NB)       = ',E20.7///)
*
      STOP
      END
```

A more complex procedure can be constructed.

```
      SUBROUTINE TTHEORY(NE,NEM,RS,ZMT,TQMT,ALST,HMT,PO,RO)
      IMPLICIT REAL*8(A-H,O-Z)
      PARAMETER(NX=210,NY=211,NPO=34,NRO=26*NX,NFIT=7*NY)
*
      CHARACTER*1 OEXT
      CHARACTER*4 OF4
      CHARACTER*6 OF6
*
      COMMON/TFITO/APO(9)
      COMMON/TFIT/FIT(NFIT)
*
      DIMENSION RS(7,6,NEM),PO(9),NE(6),RO(7,6,NEM),ARS(NX)
      DIMENSION OTPPO(NPO),OTPRO(NRO),OTPDRO(5,9,NX)
      DIMENSION SET(NY),AET(NY),SMUT(NY),AMUT(NY),STAUT(NY),
     #          ATAUT(NY),SHADT(NY)
*
      EXTERNAL M01CAF
*
      ZM= ZMT
      TQM= TQMT
      ALS= ALST
      HM= HMT
*
      DO I=1,9
          APO(I)= PO(I)
      ENDDO
*
*-----IC LOOP OVER CHANNELS | 1 = XS(E), 2 = XS(MU), 3 = XS(TAU), 4 = XS(HAD)
*                             5 = AFB(E), 6 = A_FB(MU), 7 = A_FB(TAU)
*
*-----IY LOOP OVER YEARS OF DATA TAKING
*
```

```
*-----IE LOOP OVER ENERGY POINTS IN YEAR IY
*
*-----THE TRULY INDEPENDENT SET OF ENERGIES IS SELECTED:
*     RS --> ARS(1,...,NC) WITH ALL ARS DIFFERENT
*
      ARS(1)= RS(1,1,1)
      NC= 1
      DO IC=1,7
       DO IY=1,6
        IF(IC.EQ.1.AND.IY.EQ.1) THEN
         IEMN= 2
        ELSE
         IEMN= 1
        ENDIF
        DO IE= IEMN,NE(IY)
          IK= 0
          DO N=1,NC
           IF(RS(IC,IY,IE).EQ.ARS(N)) THEN
            IK= IK+1
           ELSE
            IK= IK
           ENDIF
          ENDDO
          IF(IK.EQ.0) THEN
           NC= NC+1
           ARS(NC)= RS(IC,IY,IE)
          ENDIF
        ENDDO
       ENDDO
      ENDDO
*
*-----ENERFIES ARE SORTED IN ASCENDING ORDER
*
      IFAIL= 0
      CALL M01CAF(ARS,1,NC,'ASCENDING',IFAIL)
*
      PRINT*,'----------------------------------------------'
      PRINT*,NC
      DO N=1,NC
       PRINT*,ARS(N)
      ENDDO
      PRINT*,'----------------------------------------------'
*
      OEXT= 'E'
      CALL TINIT(NRS,OEXT)
      OF4= 'FITE'
```

```
      OF6= 'OMODES'
      CALL TCFLAG(OF6,OF4)
*
      SE= 1.D2
      CALL TOPAZ0(SE,NC,ARS,ZM,TQM,HM,ALS,OTPPO,OTPRO,OTPDRO)
*
      NCT= NC+1
      DO IT=1,7
       DO IF=1,NCT
        K= NCT*(IT-1)+IF
        IF(IT.EQ.1) THEN
         SET(IF)= FIT(K)
        ELSE IF(IT.EQ.2) THEN
         SMUT(IF)= FIT(K)
        ELSE IF(IT.EQ.3) THEN
         STAUT(IF)= FIT(K)
        ELSE IF(IT.EQ.4) THEN
         SHADT(IF)= FIT(K)
        ELSE IF(IT.EQ.5) THEN
         AET(IF)= FIT(K)
        ELSE IF(IT.EQ.6) THEN
         AMUT(IF)= FIT(K)
        ELSE IF(IT.EQ.7) THEN
         ATAUT(IF)= FIT(K)
        ENDIF
       ENDDO
      ENDDO
*
*-----THE REQUESTED RO IS ASSIGNED TO EACH ENERGY
*
      DO I=1,NC
       DO IC=1,7
        DO IY=1,6
         DO IE= 1,NE(IY)
          IF(RS(IC,IY,IE).EQ.ARS(I)) THEN
           IF(IC.EQ.1) THEN
             RO(IC,IY,IE)= SET(I)
           ELSE IF(IC.EQ.2) THEN
             RO(IC,IY,IE)= SMUT(I)
           ELSE IF(IC.EQ.3) THEN
             RO(IC,IY,IE)= STAUT(I)
           ELSE IF(IC.EQ.4) THEN
             RO(IC,IY,IE)= SHADT(I)
           ELSE IF(IC.EQ.5) THEN
             RO(IC,IY,IE)= AET(I)
           ELSE IF(IC.EQ.6) THEN
```

```
          RO(IC,IY,IE)= AMUT(I)
        ELSE IF(IC.EQ.7) THEN
          RO(IC,IY,IE)= ATAUT(I)
        ENDIF
       ENDIF
      ENDDO
     ENDDO
    ENDDO
   ENDDO
*
   RETURN
   END
```

Here the meaning of the various variables is the following:
NE(I), I=1,6 The number of energy points in year I.
NEM The $\max_I$ NE(I).
RS(IC,IY,IE) All energies for channel, year of data taking, energy loop in each year. Channels are specified according to:

$$\begin{aligned}
\mathrm{IC} &= 1 \to \sigma^{(s)}(e),\\
\mathrm{IC} &= 2 \to \sigma(\mu),\\
\mathrm{IC} &= 3 \to \sigma(\tau),\\
\mathrm{IC} &= 4 \to \sigma(\mathrm{had}),\\
\mathrm{IC} &= 5 \to A^{(s),e}_{\mathrm{FB}},\\
\mathrm{IC} &= 6 \to A^{\mu}_{\mathrm{FB}},\\
\mathrm{IC} &= 7 \to A^{\tau}_{\mathrm{FB}}.
\end{aligned} \tag{8}$$

8 More on ISPP

TOPAZ0 version < 4.4, use an effective, naive, implementation of ISPP (**ONP = Y**) based on KKKS interfaced with soft photon re-summation. As shown by Gunter Quast the agreement with MIZA is perfect around the resonance but quickly deteriorates on the high energy side.

- **First comment**. One should remember that JMS is a generalization of KF ansatz for re-summation of both soft photons and soft lepton pairs. If one goes back to KF paper it is clearly stated the region of applicability of the result, energy of pairs not exceeding Δ, with $E \gg \Delta \gg 2m$. In other words, one should not simulate too hard pairs with the soft result (for instance $z_{\min} = 0.25\,s$ means that all primary pairs with, at least 50% of the nominal energy are accepted. Away from resonance hard pairs become important.

In order to have some partial improvement upon the present situation the evaluation of **extrapolated observables** has been upgraded. It is not the global, up to LEP 2 energies, full 2f-4f interface but it amounts to the following changes in version 4.4 (**changes implemented already in April!!!**):

1. **ONP** has the additional value of **ONP = I**, where

2. KF result for virtual pairs and for soft and exponentiated pairs is used; it is also applicable to hadron pairs, because one uses KKKS results for $\mathcal{O}\left(\alpha^2\right)$ and write it in terms of moments. Then one matches it to KF and generalize KF to soft and exponentiated hadrons pairs.

3. Next, one uses generalized KF for V+S and cut to the same s' cut of IS QED radiation. This choice, however, is not strictly needed since the code allowes to have different cuts for pairs and for photons.

4. Finally, one uses soft only up to some cut that is compatible with $E \gg \Delta \gg 2\,m$. Above it one uses KKKS but not added linearly, KKKS in convolution with IS QED radiation. The radiator used here is a LL radiator, with options `ORISPP = S` (second order) and `ORISPP = T` (third order). The last, posted, version of `TOPAZ0` includes this option.

None of these approaches, including JMS, is the complete result. All of them are some sort of ansatz, more or less justified.

The improved ISPP treatment is not fully correct, as explained above, since hard pairs are taken from KKKS result which is only an approximation to the true answer. A new flag has been inserted corresponding to the value of Δ, the soft/hard separator in pairs. It helps in testing stability. Therefore `TCUTSET` has one more argument

```
      SUBROUTINE TCUTSET(IND,SOC,THMN,THMNP,ACLL,EOC,XCUT,XCUTF,XPCUT,
     #                   XSEPA)
```

and you set s' and $z_{\min}$ with

```
      VAL= 0.01D0
      CALL TCUTSET(IND,DMY,DMY,DMY,DMY,DMY,VAL,DMY,VAL,DMY)
```

By selecting `ONP = 'I'` one gets an answer for **ISPP** (for extrapolated observables) that is in good agreement with alternative treatments (Arbuzov, JMS). For those observables where cuts are included the code returns (by internal rearrangement) the old treatment `ONP = 'Y'`.