CERN-TH/2003-287
December 2003

# The tauola-photos-F environment for the TAUOLA and PHOTOS packages†, *release II*

**P. Golonka[a,b,c], B. Kersevan[d,e], T. Pierzchała[f], E. Richter-Wąs[b,g] Z. Wąs[b,h] and M. Worek[b,i]**

[a] *CERN, EP/ATR, CH-1211 Geneva 23, Switzerland.*
[b] *Inst. of Nuclear Physics PAS, Radzikowskiego 152, 31-342 Cracow, Poland.*
[c] *Faculty of Nuclear Physics and Techniques, AGH, Reymonta 19, 30-059 Cracow, Poland.*
[d] *Faculty of Mathematics and Physics, University of Ljubljana, Jadranska 19, SI-1000 Ljubljana, Slovenia.*
[e] *Jozef Stefan Institute, Jamova 39, SI-1000 Ljubljana, Slovenia.*
[f] *Inst. of Physics, Univ. of Silesia, Uniwersytecka 4, 40-007 Katowice, Poland.*
[g] *Inst. of Physics, Jagellonian University, Reymonta 4, 30-059 Cracow, Poland.*
[h] *CERN, Theory Division, CH-1211 Geneva 23, Switzerland.*
[i] *Institute of Nuclear Physics, NCSR "Demokritos", 15310, Athens, Greece.*

**Abstract**

We present the system for versioning two packages: the `TAUOLA` of $\tau$ lepton decay and `PHOTOS` for radiative corrections in decays. The following features can be chosen in automatic or semi-automatic way: (1) format of the common block `HEPEVT`; (2) version of the physics input (for `TAUOLA`): as published, as initialized by CLEO collaboration, as initialized by ALEPH collaboration (it is suggested to use this version only with the help of the collaboration advice), new optional parametrizations of matrix elements in $4\pi$ decay channels; (3) type of application: stand-alone, universal interface based on the information stored in the `HEPEVT` common block including longitudinal spin effects in the elementary $Z/\gamma^* \to \tau^+\tau^-$ process, extended version of the standard universal interface including full spin effects in the $H/A \to \tau^+\tau^-$ decay, interface including the case of mixed scalar–pseudoscalar $h\tau\tau$ couplings, interface for `KKMC` Monte Carlo, (4) random number generators; (5) compiler options.

*To be submitted to Comput. Phys. Commun.*

† This work is partly supported by the Polish State Committee for Scientific Research (KBN) grants Nos 2P03B00122, 2P03B13025, 5P03B09320, the European Community's Human Potential Programme under contract HPRN-CT-2000-00149 Physics at Colliders )

**PROGRAM SUMMARY**

*Title of the environment:* `tauola-photos-F`, release II.
*Computer:* PC running GNU/Linux operating system
*Programming languages and tools used:* `CPP`: standard C-language preporcessor, `GNU Make` builder tool, also `FORTRAN-77` compiler.
*Size of the package:* 10 MB uncompressed,
2 MB packed for distrubution into `tar.gz` archive.
*Keywords:*
particle physics, Monte Carlo methods, tau decays, TAUOLA [1], PHOTOS [2]
*Nature of the physical problem:*
The code of Monte-Carlo generators often needs to be tuned to the needs of large HEP Collaborations and experiments. Usually, the changes do not introduce important changes in the algorithm, but rather modify initialization and form of hadronic current in $\tau$ decays. The format of the event record (`HEPEVT` common block) used to exchange information between building blocks of Monte-Carlo systems often need modification. Thus, there is a need to maintain various, slightly modified versions of the same code. The package presented here allows to produce ready-to-compile versions of `TAUOLA` [1] and `PHOTOS` [2] Monte Carlo generators in versions and with appropriate demonstration programs.

New algorithm, universal interface of `TAUOLA` to work with `HEPEVT` common block is also documented here. Finally minor technical improvements of `TAUOLA` and `PHOTOS` are also listed.
*Method of solution:*
The standard `UNIX` tool: C-language preprocessor is used to produce ready-to-distribute version of `TAUOLA` [1] and `PHOTOS` [2] code. The final version of `FORTAN77` code is produced from the library of "pre-code" that is included in the package.
*Typical running time:*
Depends on the speed of computer used and demonstraton program chosen. Typically a few seconds.
*Accessibility:*
web page: `http://cern.ch/wasm/goodies.html`

# Contents



# 1 Introduction

The `TAUOLA` [1–3] and `PHOTOS` [4, 5] are the computing projects of rather long history. Written and maintained by the well defined authors, nonetheless migrated into wide range of applications where became ingredients of the complicated simulations chains. As a consequence a large number of different versions are presently in use. From the algorithmic point of view, they often differ only in a few small details, but incorporate substantial amount of specific results from the distinct $\tau$-lepton measurements. Often program versions differ because of the requirements of interfaces to other packages used in the simulation chains (*e.g.* format of the event record has to be adjusted).

Present utility for constructing specific versions of `TAUOLA` and `PHOTOS`, is prepared for the software librarians and advanced users. The idea was to create a repository which allows to include and keep main options of `TAUOLA` developed for different purposes. At the same time repository can provide the standard `FORTRAN` files which can be handled later in the same way as the published versions of the packages. The other, relatively new project which we will call `TAUOLA universal interface` is also discussed here.

Our present document is not aimed to be the manual of the `PHOTOS` and `TAUOLA` packages. It is assumed that the user is familiar with the programs themselves and their documentation, see Refs. [1–3] and [4, 5]. The purpose of the paper is to document how all parts of the code including different versions and options are combined into one distribution package. Minor improvements in the codes, since the time of publication are also listed here. The paper is an extended and enlarged version of the earlier documentation [6] and superseeds it.

**Motivations for versioning**:

1. PHOTOS: Versions of Fortran code are necessary because of the different versions of the `HEPEVT` common block being in use in the HEP libraries (single/double precisions, dimension of matrices).

2. TAUOLA: Versions of Fortran code are motivated by: (A) different versions of initialization of physics parameters; (B) interfaces with different Monte Carlo generators for production of $\tau$-lepton(s); and (C) different versions of the `HEPEVT` common block:

   - (A) Different physics initializations:
     (1) As published in [3], with improvements from Ref. [7];
     (2) As initialized by ALEPH collaboration [8] (it is suggested to use this version only with the help of the collaboration advice);
     (3) As initialized by CLEO collaboration [9] (see printout of this version for details);
     (4) New parallel $4\pi$ channels of $\tau$ decay based on Novosibirsk data [10].
     (5) New parallel $4\pi$ channels of $\tau$ decay based on Refs. [7, 11].
   - (B) Different interfaces with MC generators:
     (1) Old demo program as in published version [3];
     (2) Interface to `KKMC` [12];
     (3) New universal interface using `HEPEVT` common block including spin effects in the elementary $Z/\gamma^* \to \tau^+\tau^-$ process [13].
     (4) The extended version of the standard universal interface including transverse spin effects in the $H/A \to \tau^+\tau^-$ decay [14–16].
     (5) The extended version of the standard universal interface including the case of mixed scalar–pseudoscalar $h\tau\tau$ couplings [17].
   - (C) Different versions of the `HEPEVT` common block.

3. TAUOLA and PHOTOS: different versions of random number generators.

4. TAUOLA and PHOTOS: makefiles with different compiler flags.

The aim is to provide full backward compatibility at the level of Fortran source with the various versions being at present in use. Standard Unix tools and infrastructure are used in the discussed setup: `cpp` the C-language preprocessor: its `#if`, `#elif` and `#include` directives, as well as symbolic links and `cat` command. It is expected that the user will use this setup to create her/his version of `TAUOLA` and `PHOTOS` libraries (subdirectories `tauola/` and `photos/`) and other subdirectories of the setup will be erased/stored separately.

## 2 Organization of the directory tree

Once unpacked, the main directory `TAUOLA` is created.
It contains `README` file and:

**subdirectories including Fortran pre-code**

1. `photos-F/`: Main directory containing PHOTOS pre-code with options.
2. `tauola-F/`: Main directory containing TAUOLA pre-code with options.
3. `tauola-factory/`: Directory containing the `glue` programme which prepare non-standard directory `tauola-F/`.
4. `demo-factory/`: Directory for updating input in demo files. For specialized use only, see Section 2.3.
5. `randg/`: Directory containing random number generators which are kept separately from the rest of `TAUOLA` and `PHOTOS` source code. This should facilitate replacement with the versions of random number generators favoured by the user. Random number generators are kept in Fortran subroutines placed in files `photos-random.h` and `tauola-random.h`.
6. `include/`: Directory containing the `HEPEVT-xxx.h` files for different versions of the `HEPEVT` common block. The symbolic link `HEPEVT.h` to the one actually used will be placed in this directory at the time of initialization. The `phyfix.xxx` files with versions of fortran subroutine `PHYFIX` used only by `TAUOLA universal interface` are also stored there. The version to be used need to be copied into `phyfix.h`. for more details see `README-phyfix`.

**subdirectories necessary to run demo and<br>to install packages on different platforms**

1. `glibk/`: Directory containing histograming package, used by demos only.
2. `jetset/`: Directory containing `JETSET` MC package, used by demos only.
3. `jetset2/`: Directory containing `PYTHIA` MC package, used by demos only.

4. `eli/`: Directory containing files necessary to construct demo to be run with `PYTHIA` MC package.

5. `platform/`: Directory containing system-dependent versions of `make.inc` files for supported platforms.

6. `make.inc`: Symbolic link to the chosen `make-xxxx.inc` located in subdirectory `platform/`. The `make-xxxx.inc` files define machine-dependent flags for compilers *etc.* to be used by all `makefiles`.

**The following directories are created once the actions of the setup are completed:**

1. `photos/`: Standard directory with Fortran code of `PHOTOS` library and its demo.

2. `tauola/`: Standard directory with Fortran code of `TAUOLA` library, its demos and example outputs.

   (a) `tauola/demo-standalone`: Demo program for `TAUOLA` executed in a standalone mode.

   (b) `tauola/demo-jetset`: Demo program for `TAUOLA` executed with `universal interface` and physics event generators using the `HEPEVT` common block. In this demo `HEPEVT` is filled from JETSET74 [18] Monte Carlo generator.

   (c) `tauola/demo-pythia`: Optionally demo program for `TAUOLA` executed with `universal interface` and `HEPEVT` common block filled from PYTHIA [18] Monte Carlo generator can be created as well.

   (d) `tauola/demo-KK-face`: Interface to KK Monte Carlo [12].

## 2.1 Options for PHOTOS Monte Carlo

Different options of PHOTOS which can be created correspond solely to the different versions of the `HEPEVT` common block. The possible options are:

1. `KK-all` – for KK Monte Carlo, older versions

2. `2kD-all` – arrays dimension 2000, double precision

3. `4kD-all` – arrays dimension 4000, double precision

4. `2kR-all` – arrays dimension 2000, single precision

5. `10kD-all` – arrays dimension 10000, double precision

The action of making required version of the library is performed with the help of `cpp` preprocessor. It creates file `photos.f` from file `photos.F`. Once it is done, the symbolic link to the required version of the `HEPEVT` common block is defined. This link is used in construction of `tauola` library, see next section.

## 2.2 Options for TAUOLA Monte Carlo

Basic options for physics initializations are: `cpc`; `cleo`; `aleph`. As results of the action performed by the package:

1. `tauola/` subdirectory is erased;
2. Directory structure of `tauola/` is rebuilt;
   - `tauola/` directory is filled with the Fortran code, libraries and makefiles;
   - `tauola/demo-xx` are filled with the Fortran code of demos;

The three possible versions of created `tauola.f` correspond to form-factors and branching ratios defined respectively as in: (cpc) published version of TAUOLA [3, 7]; (aleph) as adopted by ALEPH collaboration [8], (cleo) as adopted by CLEO collaboration [9].

**Remarks:**

• The `makefile` files are prepared to run `TAUOLA` within environment of the distribution `TAUOLA` directory, however the templates for makefiles are compatible with these of the `KK` Monte Carlo. Thus if `tauola` directory is copied into respective place of the `KKMC` distribution tree, and `make makflag` of `KK/ffbench/` is executed, it overwrites `makefile` file in `tauola/`. The new ones are produced from `makefile.templ` and match the `KKMC` structure.

• Additional parametrizations for form-factors, which can be useful in some applications, are stored in the directory `TAUOLA/tauola-F/suppl`. They are not ready to use and some cross checks are mandatory. At present, code used in Refs. [19] and [20] is stored there.

## 2.3 How to change setting of `TAUOLA` input parameters

It is often necessary to change some of the `TAUOLA` input parameters like branching ratios, mass of the $\tau$-lepton, *etc.* It is convenient to have it done once for all applications *i.e.* `demo-KK-face`, `demo-jetset` and `demo-standalone`. The purpose of the `demo-factory` directory is exactly that. Here one can create the `.F` files for the interfaces, by the set of paste commands embodied in the script `klej`, out of the blocks of the Fortran code. More precisely the following files can be recreated:

- For demo-KK-face: `./prod/Tauface.F`
- For demo-jetset: `./prod/tauola_photos_ini.F`
- For demo-standalone: `./prod/taumain.F`

For details of the intialization routines, which are semi-identical in the three cases, see Refs. [1–3]. This requires special care from the physics point of view. In many cases input parameters are inter-related with the actual choice of form factors. The changes should be thus performed consistently.

**How to proceed:**

1. Some of the routines in directory `./source` have to be updated by hand first. They are stored in individual files. The ones which usually should not be modified are write protected.

2. Later execution of the script `klej` will create the following files from the pieces stored in directory `./source` simply by pasting them together:

   - `./prod/Tauface.F`,
   - `./prod/tauola_photos_ini.F`,
   - `./prod/taumain.F`.

   Automatic check (`diff`) with the archive versions stored in directory `./back` will also be executed.

3. Finally the following commands copy the files into appropriate places:

   (a) `cp prod/Tauface.F ../tauola-F/tauface-KK-F/Tauface.F`

   (b) `cp prod/tauola_photos_ini.F ../tauola-F/jetset-F/tauola_photos_ini.F`

   (c) `cp prod/taumain.F ../tauola-F/standalone-F/taumain.F`

## 2.4 Random number generators

- `PHOTOS` and `TAUOLA` have their own copies of the random number generators. They are contained in the `include` files placed in the directory `randg`.

- The user who wants to implement her/his own generators, *e.g.* compatible with the ones used by the collaboration, should replace files:

  – `./photos-random.h`,

  – `./tauola-random.h`

  by the files including the appropriate wrappers of his/her own random generators (or empty files if the generators of the same name reside elsewhere).

## 2.5 Compiler flags etc

Platform dependent parts of the `makefiles` are stored in directory `platform/`. At present options for LINUX and AIX platforms are available only. But it is rather straightforward to extend them to the new ones.

# 3 Universal interface with HEPEVT common block

In the present section, let us document for the first time, how to install and use the `TAUOLA universal interface` with “any” production generator, to include spin effects in the $Z/\gamma^* \to \tau^+\tau^-$ process [13, 14]. The approximate spin correlation are calculated from the information stored in the `HEPEVT` common block [21] filled by “any” $\tau$ production program. As a demonstration example it is interfaced with `JETSET` generator, however it works in the same manner with `PYTHIA, HERWIG` and should work with `ISAJET` generator as well. In fact, such an interface can be considered as a separate software project, to some degree independent both from the specific problem of $\tau$ production and its decay. The aim of this interface is not to replace the matrix element calculations, but rather to provide a method of calculating/estimating spin effects in cases when spin effects would not be taken into account at all.

## 3.1 HEPEVT common block and HEPEVT event record

The question of adopting universal interface[1] to different versions of `HEPEVT` event record goes beyond the technicalities discussed in Section 2.1. In fact it is quite involved, as depending on specific needs due to the way how the `HEPEVT event record` is actually coded into `HEPEVT common block`. It vary from application to application because of physics requirement. During MC4LHC [22] workshop these issues were discussed and thanks to interaction with the users `TAUOLA universal interface` was found to work with all 3 basic options of `HEPEVT` fills by `PYTHIA` 6.2 (`MSTP(128)=0,1,2`), as well as with `HERWIG` (6.150) [23] option. In our demonstration example we keep only working interface with `PYTHIA` version 5.7, the working example with other installations can be found *e.g.* in [24].

The discussion of the related problems go beyond this document, we refer the reader to Section 8 for more details. We are also convinced that related problems, because they originate from physics, will appear in event records of other languages such as C++.

## 3.2 Longitudinal spin effects in $Z/\gamma^* \to \tau^+\tau^-$ process

Let us start with the simpler case when only longitudinal spin degrees are included. Then the interface acts in the following way:

- The user has to verify if the $\tau$ lepton is forced to be stable in the package performing generation of the $\tau$ production.
- The content of the `HEPEVT` common block is searched for all $\tau$ leptons and $\tau$ neutrinos.
- It is checked if there are $\tau$ flavour pairs (two $\tau$ leptons or $\tau$ lepton and $\tau$ neutrino) originating from the same mother(s).

[1] The question is even more serious in case of `PHOTOS`.

- The decays of the $\tau$ flavour pairs are performed with the subroutine TAUOLA. Longitudinal spin correlations are generated in the case of the $\tau$ produced from decay of: $W \to \tau\nu$, $Z/\gamma^* \to \tau\tau$, and the charged Higgs boson $H^\pm \to \tau\nu$ (for the neutral Higgs boson the full spin correlations are used, see Section 3.3). Parallel or anti-parallel spin configurations are generated, before calling on the $\tau$ decay, and then the decays of 100 % polarised $\tau$'s are executed.

- In the case of the Higgs boson (for the spin correlations to be generated) the identifier of the $\tau$ mother must be that of the Higgs boson. The particle code convention as that used by the PYTHIA 5.7 [25] Monte Carlo is adopted, but it can be changed by the user, as explained later in the Section.

- In case of the $W$ and $Z/\gamma^*$ it is not necessary, becaue the interface assume these production mechanisms as defaults, even if the $W$ or $Z/\gamma^*$ bosons are not explicitelly present in event record. For example if from the same mother as that of the $\tau$, a $\nu_\tau$ is also produced, the $W$ (of the momentum equal to sum of $\tau$, and $\nu_\tau$) will be taken as the true-mother of the $\tau$. Similarly, if from the same mother another $\tau$ with opposite charge is produced, the $Z/\gamma^*$ is assumed to be the mother of the $\tau$ pair.

- The spin effects in the states of $\tau^+$, $\tau^-$ produced from explicit or implicit $Z/\gamma^*$ states are upgraded up to nearly KORALZ standards [26, 27]. Bremsstrahlung photons are merged either with beams or with $\tau$'s [13].

- Photon radiation in the decay is performed with PHOTOS package [4, 5].

- Let us note that the calculation of the $\tau$ polarisation created from the $Z$ and/or virtual $\gamma^*$ (as function of the direction) represents a rather non-trivial extension if high precision is required. Generally, the dedicated study of the production matrix elements of the host generator is necessary in every individual case. The method applied here is based on approximations [13, 14].

The TAUOLA interface is organised in a modular form to be used conveniently in "any environments".

### Initialisation

Initialisation is performed with the `CALL TAUOLA(MODE,KEYSPIN)`, `MODE=-1`. All necessary input is directly coded in subroutine TAUOLA placed in a file `tauface-jetset.f`.

The following input parameters are set at this call. We omit those, which are standard input for TAUOLA as defined in its documentation. They are hard-coded in the subroutine.

| Parameter | Meaning |
|---|---|
| POL | Internal switch for spin effects in $\tau$ decays and only for longitudinal case. Normally the user should set POL=1.0, and when POL=0.0 spin polarisation effects are absent |
| KFHIGGS(3) | (KF=25, 35, 36) Flavour code for $h$, $H$ and $A$ |
| KFHIGCH | (KF=37) Flavour code for $H^+$ |
| KFZ0 | (KF=23) Flavour code for $Z^0$ |
| KFGAM | (KF=22) Flavour code for $\gamma$ |
| KFTAU | (KF=15) Flavour code for $\tau^-$ |
| KFNUE | (KF=16) Flavour code for $\nu_\tau$ |
| xmtau | $\tau$-lepton mass, used only for KFHIGGS(3), see Section 3.3 |
| xmh | Higgs boson mass, used only for KFHIGGS(3) |
| psi | mixing scalar-pseudoscalar angle, this angle is used only for Higgs boson with flavour code equal KFHIGGS(3) |

**Event generation**

For every event generated by the production generator, all $\tau$ leptons will be decayed with the single CALL TAUOLA(0,KEYSPIN) (KEYSPIN=1/0 denotes spin effects switched on/off). In its execution, all $\tau$ leptons, will be first localised, their positions stored in internal common block TAUPOS, and the information necessary for calculation of $\tau$ spin state will be read, from HEPEVT common block. Later spin state for the given $\tau$ (or $\tau$ pair) will be generated, and finally decay of polarised $\tau$ will be performed with the standard TAUOLA action. In particular, the decay products of $\tau$ will be boosted to the laboratory frame and added to the complete event configuration stored in HEPEVT common block.

**Calculation of the $\tau$ spin state**

Once CALL TAUOLA(0,KEYSPIN) is executed and $\tau$ leptons are found, the spin states need to be calculated.

First, we look in HEPEVT for the position of $\tau$'s mothers and store them in matrix IMOTHER(20). Each mother giving $\tau$ lepton(s) is stored only once, independently of the number of produced $\tau$'s. Later, for every `IMOTHER(i)` we execute the following steps:

1. The daughters which are either $\tau$ leptons or $\nu_\tau$ are searched for.
2. Daughters are combined in pairs, case of more than 1 pair is not expected to be important and ad hoc pairing is then performed.
3. The two main cases are thus ($\tau$ $\tau$) or ($\tau$ $\nu_\tau$).
4. The default choices in these two main cases are, respectively, $Z/\gamma^*$ or $W$, unless the identifier of `IMOTHER(i)` is different and explicitely defined, e.g. as neutral (or charged) Higgs boson.
5. Calculation of the spin parameters is independent from kinematic variables and straightforward in all cases except $Z/\gamma^*$ (see [13]).
6. For $Z/\gamma^*$, the polarization function $P_Z$ [13] is calculated with the help of the function `PLZAPX(HOPE,IMO,NP1,NP2)`. The `HOPE` is the logical parameter defined in subroutine `TAUOLA` placed in file `tauface-jetset.f`. It tells whether spin effects can be calculated or not. If available information is incomplete `HOPE` is set to `.false.`. Then `PLZAPX(HOPE,IMO,NP1,NP2)` returns 0.5. The `IMO` denotes position of the $\tau$ mother in `HEPEVT` common block, the `NP1` denotes position of $\tau^+$ and the `NP2` denotes position of $\tau^-$.
7. To calculate reduced $2 \rightarrow 2$ body kinematical variables $s$ and $\cos\theta$ subroutine `ANGULU(PD1,PD2,Q1,Q2,COSTHE)` is used. 4-momenta of the incoming effective beams and outgoing $\tau^+$ and $\tau^-$ are denoted by `PD1`, `PD2`, `Q1`, `Q2`, respectively. The variables $s$ and $\cos\theta$ are then used in function $P_Z$.

### Run summary

After the series of events is generated the optional `CALL TAUOLA(1,KEYSPIN)` can be executed. The information on the whole sample, such as number of the generated $\tau$ decays, branching ratios calculated from matrix elements *etc.*, will be printed.

### Demonstration program

Our main program `demo.f` is stored in subdirectory `demo-jetset`. It reads in the file `init.dat` which includes some input parameters for the particular run, such as number of events to be generated (by `JETSET/PYTHIA`), or the type of the interaction it should use to produce $\tau$'s, *etc.* We address the reader directly to the code for more details. It is self-explanatory. The printouts at the initialization has the form presented at figure 1, and the printout at the end of the execution has the form presented in figure 2.

```
***************************************************************************
*                         *****TAUOLA UNIVERSAL INTERFACE: ******         *
*                         *****VERSION 1.10, November 2003 ******         *
*                         **AUTHORS: P. Golonka, B. Kersevan, ***         *
*                         **T. Pierzchala, E. Richter-Was, ******         *
*                         ****** Z. Was, M. Worek ***************         *
*                         **USEFUL DISCUSSIONS, IN PARTICULAR ***         *
*                         *WITH C. Biscarat and S. Slabospitzky**         *
*                         ****** are warmly acknowledged ********         *
*                                                                         *
*                         ********** INITIALIZATION  ************         *
*             1.00000     tau polarization switch must be 1 or 0          *
*             1.57080     Higs scalar/pseudo mix CERN-TH/2003-166         *
***************************************************************************
```

Figure 1: Printout of TAUOLA initialization

```
***************************************************************************
*                         *****TAUOLA UNIVERSAL INTERFACE: ******         *
*                         *****VERSION 1.10, November 2003 ******         *
*                         **AUTHORS: P. Golonka, B. Kersevan, ***         *
*                         **T. Pierzchala, E. Richter-Was, ******         *
*                         ****** Z. Was, M. Worek ***************         *
*                         **USEFUL DISCUSSIONS, IN PARTICULAR ***         *
*                         *WITH C. Biscarat and S. Slabospitzky**         *
*                         ****** are warmly acknowledged ********         *
*                         ****** END OF MODULE OPERATION ********         *
***************************************************************************
```

Figure 2: Printout of `TAUOLA` end-of-run messages

### 3.3 Transverse spin effects in $H/A \to \tau^+\tau^-$ decay

The extended version of the standard `universal interface`, of the `TAUOLA` $\tau$-lepton decay library, includes the complete spin effects for $\tau$ leptons originating from the spin zero particle *i.e.* in $H/A \to \tau^+\tau^-$ decay [14–16]. As usual, the interface is expected to work with any Monte Carlo generator providing production (and subsequent decay into pair of $\tau$ leptons) of the Higgs boson. Since the Higgs boson spin is zero, the spin correlations of its decay products do not depend at all on the mechanism of the Higgs boson production. Technical difficulties related to the choice of $\tau^+$ and $\tau^-$ spin quantization frames, present in the case of $e^+e^- \to Z/\gamma^* \to \tau^+\tau^-$ [28, 29] (initial state bremsstrahlung effects included or not), are not present. The analytical form of the density matrix is simple. To calculate the density matrix for the pair of $\tau$-leptons it is thus enough to: know their four momenta, know that they indeed originate from the Higgs boson and, assume the type of the Yukawa interaction. Such information is stored in `HEPEVT` common block [21] used by practically all Monte Carlo generators for Higgs boson production.

The interface acts in the following way and accordingly to principles described in [1]:

- The algorithm is organized in two steps. In the first step, the $\tau$ lepton pair is generated and the $\tau$ leptons are decayed in their respective rest frames as if there were no spin effects at all. In the second step, the spin weight is calculated and rejection is performed. If the event is rejected, only the generation of the $\tau$ lepton decays is repeated.
- The spin weight for Higgs boson decay into $\tau^+\tau^-$ is given by the following formula
$$wt = \frac{1}{4}\Big(1 + \sum_{i=1}^{3}\sum_{j=1}^{3} R_{ij} h_1^i h_2^j\Big) \tag{1}$$
- The components of the density matrix $R_{ij}$ are equal $R_{33} = -1$, $R_{11} = \pm 1$, $R_{22} = \pm 1$, respectively for pure scalar and pseudoscalar Higgs boson, and all other components are zero. The $h^i$ and $h^j$ are the polarimetric vectors of the $\tau^\pm$ leptons (they are calculated by `TAUOLA`).
- In general, the Yukawa coupling can be written as $\bar{\tau}(a + ib\gamma_5)\tau$. With our choice of quantization frames, the following non-zero components of $R_{ij}$ are obtained (see Ref. [17]):
$$R_{33} = -1, \qquad R_{11} = R_{22} = \frac{a^2\beta^2 - b^2}{a^2\beta^2 + b^2}, \qquad R_{12} = -R_{21} = \frac{2ab\beta}{a^2\beta^2 + b^2}, \tag{2}$$
where $\beta = \sqrt{1 - \frac{4m_\tau^2}{m_H^2}}$. With Yukawa coupling written in a form $\bar{\tau}N(\cos\phi + i\sin\phi\gamma_5)\tau$ and with the limit $\beta \to 1$, the expressions reduce to the components of the rotation matrix by an angle $-2\phi$:
$$R_{11} = R_{22} = \cos 2\phi, \qquad R_{12} = -R_{21} = \sin 2\phi. \tag{3}$$

The following changes had to be introduced to the algorithm explained in Ref. [13]:

- The quantization frames for the spin states of $\tau^+$ and $\tau^-$ need to be properly oriented with respect to each other. In our solution they are simply connected by the boost along $\tau$ lepton momenta as defined in the Higgs boson rest-frame. At the technical level this is enforced by the `TRALOR` routine [1] defining the relation of the $\tau^\pm$ spin quantization frames and the laboratory frame. As an intermediate step this routine uses the Higgs boson rest frame.

- The density matrix for pure scalar/pseudoscalar case was taken from Ref. [30], in the most general case it is present *e.g.* in Ref. [31]. We have adapted it to the quantization frames as specified in previous points. Two cases are implemented: pure scalar Higgs boson and Higgs boson with mixed scalar-pseudoscalar coupling.

- Generation of $\tau^+$ and $\tau^-$ decays is implemented in subroutine `SPINHIGGS` placed in the file `tauface-jetset.f` and performed following the method explained and used in Refs. [1, 29].

- The function `wthiggs(IFPSEUDO,HH1,HH2)` is used to calculate the spin weight. The `HH1` and `HH2` are the $\tau^+$ and $\tau^-$ polarimetric vectors calculated inside the subroutine `DEKAY` placed in the file `tauola.f`. The `IFPSEUDO` is the logical parameter defined in subroutine `TAUOLA` placed in file `tauface-jetset.f`. It tells whether spin effects are calculated for the scalar or for the mixed scalar-pseudoscalar Higgs boson. If we have pure scalar Higgs boson it is set to `.false.`. If we have mixed scalar-pseudoscalar case it is set to `.true.`. Pure pseudoscalar case is for $\phi = \frac{\pi}{2}$.

- We have assumed that production generator provides two-body Higgs boson decays to $\tau$ leptons. In particular, that it does not provide any bremsstrahlung corrections. Instead, `PHOTOS` [4, 5] (see also [32]) can be used for that purpose, once generation of $\tau^\pm$ decays is completed.

- More complete inclusion of bremsstrahlung corrections would require a substantial re-write and extension of the program to the solution presented in Ref. [12] or a similar one.

# 4 How to use the package

1. Start with `make Clean` from main directory to secure against mismatches.

2. Check platform dependent makefiles.

   - Go to subdirectory `platform/`
   - Determine if `make-xxx.inc` file specific for your computer is present there: for LINUX it is `make-linux.inc`; for AIX it is `make-aix.inc`; for other it need to be adopted.
   - Erase symbolic link `make.inc` existing in this directory and create a new one which points to the chosen `make-xxx.inc`:

    - `rm make.inc`
    - for LINUX: `ln -s make-linux.inc make.inc`
    - for AIX: `ln -s make-aix.inc make.inc`

   Afterwards check whether link to the `make.inc` is present in the main directory.

3. Settings of `TAUOLA` input parameters can be changed for all implemented applications, see chapter 2.3 for details.

4. `PHOTOS` and `TAUOLA` have their own private random generators. If you wish to replace them, you should do it at this point, see chapter 2.4.

5. Create required versions of `photos/` and `tauola/` directories. It is mandatory to create `photos/` directory first, *i.e.* before creating `tauola/`.

   - Go to directory `photos-F`
   - Type one of the following commands to choose the required version of `HEPEVT`:
     - `make KK-all`
     - `make 2kD-all`
     - `make 4kD-all`
     - `make 2kR-all`
     - `make 10kD-all`
   - Go to directory tauola-F:
   - Type one of the following commands to choose the required version of `TAUOLA` initialization:
     - `make cpc`
     - `make cleo`
     - `make aleph`

     Then user can type

     - `make pythia`

     to construct additional demo executable with `PYTHIA`. In this demonstration programme, old `PYTHIA` version 5.720 [18] is used. The user is anyway expected to attach the package to a new software, up to the date at the time of installation. In case of this example matching of the common block dimensions is not automatically assured. The option `make 4kD-all` has to be used.

6. The required version of `PHOTOS` and `TAUOLA` will reside in newly (re)created directories `./photos` and `./tauola`.

7. Following `demos` can be invoked from that directories:

   - Demo for `PHOTOS` resides in `./photos/demo`, and can be invoked by command `make` followed by `make run`.

- Demo for `TAUOLA` stand-alone resides in `./tauola/demo-standalone`, and can be invoked by the command `make` followed by `make run`.
- Demo for `TAUOLA` with `JETSET` being a host Monte Carlo resides in `./tauola/demo-jetset`, and can be invoked by the command `make` followed by `make run`.
- Demo for `TAUOLA` with `PYTHIA` being a host Monte Carlo resides in `./tauola/demo-pythia`, and can be invoked by the command `make` followed by `make run`.
- Interface to `KKMC` resides in `./tauola/KK-face/Tauface.f`. It has to be moved to `./KK2f/Tauface.f` of distribution directory of KKMC [12]. The rest of the `./tauola` directory should replace the original one of the `KK` Monte Carlo distribution.

Finally, let us remark that most of the `TAUOLA` tree is not necessary and can be erased at this point. Code and makefiles of directories `./tauola` and `./photos` are sufficient. To execute demo programs, directories `./jetset` and `./glibk` need to be kept or replaced by the appropriate links. The `make.inc` symbolic link pointing to `make-xxxx.inc` file in directory `./platform` need to be kept also. The file `make-xxxx.inc` defines appropriate compiler flags.

# 5 New hadronic currents for $\tau \to 4\pi$ channels of $\tau$ decay

The parameterization of form-factors developed for $4\pi$ channels of $\tau$ decay and based on Novosibirsk data has been coded in a form suitable for the `TAUOLA` Monte Carlo package [10]. There exist two $4\pi$ final states in $\tau^-$ decay ( *i.e.* $\tau^- \to \nu_\tau \pi^- \pi^0 \pi^0 \pi^0$, $\tau^- \to \nu_\tau \pi^- \pi^- \pi^+ \pi^0$). Both are available as a parallel decay modes in the `cpc`, `cleo` or `aleph` versions. To get the new $4\pi$ channels of $\tau$ decay (`BINP` version) one should act in the following way:

1. Go to directory `tauola-factory`.
    - There is a `glue` programme which prepare a fortran code of `TAUOLA` library and place it in the directory `../tauola-F`.
    - Type the following command to see the parameters of the `glue` programme:
        - `./glue`
    - Type the following command to choose the `BINP` version:
        - `./glue binp`
    - Type the following command to come back to the standard version:
        - `./glue standard`

2. Go to the main directory.

3. Act according to the instructions described in the previous chapter.

4. The required `TAUOLA` version (`cpc`, `cleo` or `aleph`) with new $4\pi$ channels of $\tau$ decay (`BINP` version) will reside in newly (re)created directorie `./tauola`.

Another parameterization of form-factors for $4\pi$ channels based on references [7, 11] is available as well. It can be installed, similar as in the case described above. The only difference is that flag `binp` need to be replaced with `Karlsruhe`.

# 6 Universal tool for HEP Monte Carlo generator comparison: `MC-TESTER`

Within the scope of the `TAUOLA-PHOTOS` project work, a pilot project of `MC-TESTER` tool for semi-automatic comparisons of Monte-Carlo generators has been deployed. Initially, the project was intended to be a part of this package: it was intended to be used as debugging/analysis tool for comparing the results produced by various versions of TAUOLA and PHOTOS code.

We decided however to publish `MC-TESTER` as an independent project [33]. The scope of its possible use spans far beyond its original use case. We have however preserved the relationship between the two projects: we include two versions of `TAUOLA` code (that may be produced using `TAUOLA-PHOTOS` ) in `MC-TESTER` distribution (versions 1.0 and 1.1) as a test that verifies the correctness of `MC-TESTER` installation.

More information and the code of `MC-TESTER` is available from its web page [34].

# 7 Minor upgrades in `TAUOLA` and `PHOTOS`

There is no need to publish the new versions of the programs, since the last publication the upgrades are minor. Nonetheless let us list here the main improvements collected over the years. The list of fixes for numerically rather insignificant bugs, as well as necessary changes due to evolution of compilers will not be given here. For that purpose we refer the reader to the comments in the code, to the `README` files included in the distribution and to the package web page [35].

## 7.1 `TAUOLA`

Except modifications listed in Section 1, there were no changes introduced in `cpc` and `aleph` initializations of `TAUOLA` since the time they became available. In case of `cleo` initialization, change in $\tau \to \nu\pi^+\pi^+\pi^-\pi^0$ channel was introduced; normalization of $\omega$ current was adjusted to experimental data [36]. To this end $\omega$ contribution was diminished from 68% down to 40% of the total rate for the $\tau \to \nu\pi^+\pi^+\pi^-\pi^0$ channel. The new version of TAUOLA 2.7, was nonetheless marked in all initializations to document progress in bug fixing *etc.* The creation dates printed in the outputs are retained. They mark the time when the bulk of the work was done.

Since publication [10] careful analysis of hadronic current for $4\pi$ final states was performed, related extensive studies and comparisons are documented in Ref. [37]. New parametrization based on Refs. [7, 11] was also installed in TAUOLA. Different options of models discussed in Ref. [10] were probed. Let us list changes with respect to [10] which are installed now in the `TAUOLA` code if used with the `binp` option. We will use the opportunity to list misprints found in Ref. [10] as well:

1. In Table 4 of Ref. [10] in all columns of function arguments $\sqrt{Q^2}$ were listed rather than $Q^2$, as written in the table header.

2. Formula (20) [10] should read $g_\sigma(s) = (1 - 4m^2/s)^{1/2}$, rather than $g_\sigma(s) = (s - 4m^2/s)^{1/2}$.

3. All propagators derived from formula (18) in Ref. [10], should be normalized to $-1$ at $q^2 = 0$, also imaginary part should be set to zero below opening of the appropriate decay channel (*i.e.* typically for $q^2 < 4m^2$).

4. In definition of the $\rho$ propagator in formula (18) in Ref. [10], the term $dm(q^2)$ was neglected. The full $\rho$ propagator should read

$$\begin{aligned} D_\rho(q^2) &= q^2 - M_\rho^2 - M_\rho\Gamma_\rho\, dm(q^2) + iM_\rho\Gamma_\rho \frac{g_\rho(q^2)}{g_\rho(M_\rho^2)} \\ dm(q^2) &= \Big(h_\rho(q^2) - h_\rho(M_\rho^2) - (q^2 - M_\rho^2)\frac{dh_\rho(q^2)}{dq^2}\Big|_{q^2=M_\rho^2}\Big)/g_\rho(M_\rho^2), \end{aligned} \tag{4}$$

where

$$h_\rho(q^2) = \begin{cases} \dfrac{\sqrt{1-\frac{4m^2}{q^2}}\ln\left(\frac{1+\sqrt{1-\frac{4m^2}{q^2}}}{1-\sqrt{1-\frac{4m^2}{q^2}}}\right)(q^2-4m^2)}{\pi} & \text{for } q^2 > 4m^2, \\ -8\frac{m^2}{\pi} & \text{for } q^2 = 0 \text{ GeV}^2, \\ 0 & \text{otherwise .} \end{cases} \tag{5}$$

5. The full $\omega$ propagator should look (term $\frac{g_\omega(q^2)}{g_\omega(M_\omega^2)}$ was approximated by 1)

$$D_\omega(q^2) = q^2 - M_\omega^2 + iM_\omega\Gamma_\omega \frac{g_\omega(q^2)}{g_\omega(M_\omega^2)} \tag{6}$$

$$\frac{g_\omega(q^2)}{g_\omega(M_\omega^2)} = \begin{cases} \max\left(0\ ,\ \begin{matrix} 1 + 17.560(\sqrt{q^2} - M_\omega) + 141.110(\sqrt{q^2} - M_\omega)^2 + \\ +894.884(\sqrt{q^2} - M_\omega)^3 + 4977.35(\sqrt{q^2} - M_\omega)^4 + \\ +7610.66(\sqrt{q^2} - M_\omega)^5 - 42524.4(\sqrt{q^2} - M_\omega)^6 \end{matrix}\right) & \text{for } \sqrt{q^2} < 1.0 \text{ GeV}, \\ -1333.26 + 4860.19\sqrt{q^2} - 6000.81(\sqrt{q^2})^2 + 2504.97(\sqrt{q^2})^3 & \text{otherwise .} \end{cases} \tag{7}$$

Function (7) is involved, because of effects due to three body phase space, threshold effects of pion masses as well as opening of the resonant $\rho$ channel in $\omega$ decay.

6. In formula (16) in Ref. [10] overall sign was missing.

7. In formula (21) in Ref. [10] coefficient in front of second line should include $s$ instead of $a$.

8. We have recalculated using two methods, the outcome from formula (21) [10]. We have found some numerical difference with respect to the one used before. The size of the difference is small, except the region above 1.3 GeV which is not important anyway (see Ref. [37]).

9. In the comment for the formula (21) in Ref. [10] it should be stated that three-vectors and energies are defined in the $a_1$ $(3\pi)$ system rather than $\tilde{\rho}$ $(4\pi)$ system.

10. In formula (25) in Ref. [10] overall coefficient $z_{mix}$ was missing, but it was set in the code to $z_{mix} = 1$ anyway.

11. At the beginning of Section 5 in Ref. [10] the numerical constant $\Lambda^2 = 1.2GeV$ should be defined, rather than $\Lambda$ alone.

All points from the above list do not change the numerical content of the code, except point 8 which brings modification, and consequently lead to numerically small compensating change of the content of the tables of functions $G$ presented in Ref. [10].

We have found that while playing with different versions of the `TAUOLA` code describing the Novosibirsk hadronic current, we had for some time an overall sign error in the space-like component of the hadronic current. For all distributions involving pions only, it was fully compensated by the choice of the fit functions $G(Q^2)$ (formulae (14,15) and (22-24) in Ref. [10]). Also all figures in Ref. [10] are correct.

However, as a consequence of the bug, the distributions involving $\nu_\tau$ were affected in `TAUOLA BINP`. The effect was not big enough for comparisons with `TAUOLA CLEO` to hint for the problem. The differences were within expected systematic errors of the two parameterizations. We have also found an additional misprint in the numerical value for our fit functions: $G_{\pi^+\pi^-\pi^+\pi^0}$, $G^{\omega}_{\pi^+\pi^-\pi^+\pi^0}$ and $G_{\pi^+\pi^0\pi^0\pi^0}$ (see Table 4, Ref. [10]). An overall constant or function was missing. Once corrections are taken together, the values in Table 4 should be multiplied respectively by:

- $$\frac{76.565\sqrt{0.71709 * \sqrt{Q^2} - 0.27505}}{M_\rho^4\sqrt{Q^2}} \quad \text{for the} \quad G_{\pi^+\pi^-\pi^+\pi^0}(Q^2), \tag{8}$$

- $$\frac{886.84\sqrt{0.70983 * \sqrt{Q^2} - 0.26689}}{M_\rho^4\sqrt{Q^2}} \quad \text{for the} \quad G^{\omega}_{\pi^+\pi^-\pi^+\pi^0}(Q^2), \tag{9}$$

- $$\frac{96.867\,|\texttt{z\_forma1(Q}^2\texttt{)}|\sqrt{0.70907*\sqrt{Q^2}-0.26413}}{M_\rho^4\sqrt{Q^2}} \quad \text{for the} \quad G_{\pi^+\pi^0\pi^0\pi^0}(Q^2) \qquad (10)$$

before being used in formulae (14,15) and (22-24) of Ref. [10]. Numerical values of the |`z_forma1(Q`$^2$`)`| function are tabulated in the new `TAUOLA` function `ZFA1TAB(Q2)`, as was the case with the functions $G$.

Finally, during the work, as a byproduct, we have switched on the interference between $\omega$ and non-$\omega$ part of the current. This changed the overall rate, mainly due to interference of small, but spread over all phase space, tails of the $\omega$ Breit-Wigner propagator. Numerical effect is not visible in differential distribution, but affects the total rate by 1.017(1). We changed normalization of the $G_{\pi^+\pi^-\pi^+\pi^0}(Q^2)$ and $G^\omega_{\pi^+\pi^-\pi^+\pi^0}(Q^2)$ by the same factor, to reproduce the total $4\pi$-rate as in PDG 2000. Note that interference of $\omega$ current with the remaining part is providing 1.7 % effect on the total rate.

## 7.2 PHOTOS

The introduced changes are collected in the following subsections.

### 7.2.1 Subroutine PHCORK

During the test phase of the `Photos+` [38], a C++ implementation of the `PHOTOS` algorithm, it turned out that generation becomes unstable for events produced at $TeV$ energies. The use of `PHOTOS` at this energy ranges was sudied then for the first time. The kinematics of events produced by `PYTHIA` 5.7 was often not precise enough, because of rounding errors for (`REAL*4`) floating-point variables.

The particles described by tree-structured event record (*i.e.* `HEPEVT` common block used by PHOTOS), form decay "branches": composed of a "mother" - the decaying particle, and "daughters" - its decay products. When the kinematics of a "branch" is not correct (the sum of 4-momentum of decay products doesn't match the 4-momentum of the "mother" particle or the particles momenta are off shell), numerical unstabilities are encountered. It is not possible to perform boosts of the particles correctly if boost parameter are large (in certain events where errors occurred we found *e.g.* a pion with momentum of order of $TeV/c$).

A special routine `PHCORK` which corrects these kinds of inconsistencies have been developed and introduced[2].

The routune works in one of four modes, selected at initialization of `PHOTOS`:

- mode 1: no correction performed,
- mode 2: energy is corrected from mass,
- mode 3: mass is corrected from energy,

[2]This routine became available since 1999 and can be found in `CERNLIB` version of `PHOTOS`.

- mode 4: energy is corrected from mass for particles up to a mass of 0.4 GeV, for heavier ones: mass is corrected.

In cases, where a branch (sub-cascade starting from certain particle) have two mothers, first mother's momentum is corrected according to its children momentum sum and momentum of second mother.

Default mode is 1, which means no correction is applied, and full compatibility with older implementations.

Correction is performed in two steps. First, all daughters' energies (or masses, depending on requested mode of correction) are corrected to fulfil $E^2 - p^2 = M^2$ constraint.

In second step, a sum of all daughters' 4-momenta is calculated. The mother's momentum is corrected to this value. In cases when two mothers are present, only first mother's 4-momentum is corrected to a value:

$P_{mother_1} = (\sum_{daughters} P_{daughter}) - P_{mother_2}$.

#### 7.2.2 Interference correction

In the published version of `PHOTOS` [5] the interference between photon emission from two sources was available only for two-body decays into particles of opposite charge and equal mass. Since then the interference correction was extended to work also in case of decays into particles of distinct masses. On the technical side the interference correction was recoded to work more efficiently. For example this should prevent program to stop in case of double bremsstrahlung and interference corrections in decay of very heavy $Z/\gamma^* \to e^+e^-$ state.

#### 7.2.3 enabling flags

For some users, it is convenient to block/enable `PHOTOS` generation in some cases, for tests or because of unphysical coding of events into `HEPEVT` common block. Flags to block/enable `PHOTOS` generation in $\pi^0 \to e^+e^-\gamma$ and in $W$ decay into quarks were introduced. The flags are initialized in subroutine `PHOINI`.

#### 7.2.4 Correction weight in decay $W \to l\nu_l$

For the leptonic decays of the $W$ boson, `PHOTOS` predictions were carefully compared with the results from the first order matrix element generator [32]. It was found that even though the leading corrections were properly modelled by `PHOTOS` the non-leading missing effects were relatively large. The origin of these discrpancies was understood, and the appropriate correcting weight was introduced into `PHOTOS`. With these changes the agreement with the matrix element calculation improved sizably [39], especially in regions of phase-space populated with hard non-collinear photons. The new option, to initialize this correcting weight was introduced. It can be switched on/off with the new flag `IFW`, initialized in subroutine `PHOCIN`. The method can be easily extended to other processes, where matrix element is available from QED calculations or from modelling based on experimental data.

#### 7.2.5 PHOTOS 2.07

Once all these modification installed, new version 2.07 of PHOTOS was created.

# 8 Issues related to distinct event record schemes

Let us note that the packages presented here work with the standard HEPEVT event record, where the tree structure of the event evolution is properly coded into a unique tree structure. Generally it is not the case. However the programs were shown to work properly with the extension of the standard of the type as originating from PYTHIA [40] with MSTP(128)=0,1,2. Necessary adjustments turned out to be non-trivial and in fact correctness of their action can not be guarranteed in general case [41]. The case of HERWIG [23] turned out to be even more complex. Even though in this case HEPEVT common block is used as standard event record, nonetheless the mother daugter pointers are used in a different way. As a consequence events generated by HERWIG are far from the design format of the HEPEVT event record. We were able to tune the working of the TAUOLA universal interface to the requirements of operating with HERWIG, however the interfacing of PHOTOS turned out to be rather more difficult.

The principal issue from which difficulties with the HEPEVT decoding arise is that PYTHIA as well as HERWIG produce a new copy of the participating particles each time an physical effect (e.g. final state radiation) is added; consequently several copies of particle entries are produced, which cannot be consistently put in the HEPEVT structure. In order to circumvent this problem, the JMOHEP and/or JDAHEP entries are *overloaded*, *i.e.* the information stored in them is used for other purposes than originally designed for. For example, PYTHIA with MSTP(128)=0 setting uses the JMOHEP entry to point to the copies of a particle (c.f. Figure 3) and HERWIG uses the JMOHEP(2,I) entries to fill the colour information of the particle flow. A generic interface thus has to be aware of these possibilities and correct for them in order for the TAUOLA and PHOTOS procedures to work properly.

At present a working version of modified PHOTOS routines, which appear to work with HERWIG can be found in the distribution of AcerMC [24]. However rigorous testing is necessary before these modifications can be included in the official PHOTOS distribution. Original PHOTOS code had to be modified due to the 'overloaded' HEPEVT record, since its requirements for matching mother-daughter pointers were too strict for either PYTHIA or HERWIG. The modification was limited to PHOTOS_MAKE and PHOBOS routines. In addition the tracking of IDHW array was added to PHOTOS_MAKE to accommodate the HERWIG event record. A further modification was however necessary in the PHOIN routine since in HERWIG the entry JMOHEP(2,I) is not empty but filled with colour flow information, which in turn inhibited PHOTOS radiation off participating particles[3]

It has to be stressed that implementing the HERWIG interface to PHOTOS is desirable since HERWIG up to version (6.5) did not have any QED radiation implemented in the decays. In the later releases PHOTOS is installed in decays (and hadronic production)

[3]PHOTOS expects the non-zero second 'mother' JMOHEP(2,I) entry only for $gg(qq) \to t\bar{t}$ process, which is treated by a set of dedicated routines; this in turn clashes with HERWIG 'overloaded' JMOHEP(2,I) entry.

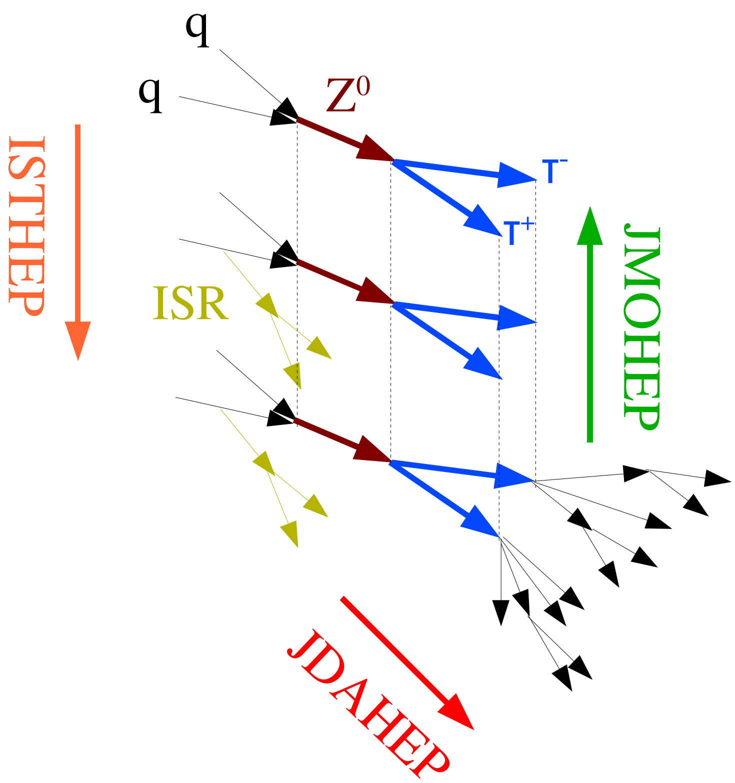


Figure 3: The usage of `JMOHEP` and `JDAHEP` pointers in `PYTHIA` with `MSTP(128)=0` setting. While `PYTHIA` keeps the `JDAHEP` entries to point in the direction of event evolution the `JMOHEP` is used to point to the copies of a particle produced when perturbative effects, such as initial state radiation(ISR) or $\tau$ decays are added; the particle copies at different stages of event simulation are consistently distinguished by the `ISTHEP` tags .

of $t$-quarks, as well as in $W$ decays. This is done however through specificaly tailored interfaces embodied within HERWIG code.

Another two issues are related to the double counting. If there is no information passed from the host program and the common block PHOQED of PHOTOS is not appropriatelly filled, then there is no way for PHOTOS to know that in some decays it should not generate bremstrahlung (again) and thus 'double counting' will appear. While the question is not relevant for HERWIG, since no photon bremsstrahlung is implemented, it can become an issue in PYTHIA for some specific processes. Indeed, one can generally switch off the photon radiation off leptons (by setting a high limit on PARJ(90), c.f. [40]), however one cannot do it on the basis of specific processes and/or decays.

The issue is not shared in the TAUOLA universal interface since the decayed $\tau$ leptons are not decayed again and the original decay will remain.

# 9 Summary and future possibilities

We have presented the system for creating required version of PHOTOS and TAUOLA packages from their master versions. The master version are structured in relatively compact form without code duplications *etc.* The minor modifications for the programs, introduced since their publications, were also explained.

Creation of the system was the first step toward future attempts to develop packages without loss of their present physics content. Some experience, collected already in that direction, is summarized in [38]. We find the question of the language translation for the fixed program version relatively easy. Contrary, the question of project continuity into further upgrades motivated by the physics, needs to be think carefully over. Matching the programming styles of the *e.g.* OO C++ experts with the strategies of testing numerical correctness of consecutive versions is a rather crucial issue which has to be addressed. Tools and methods embodied in Fortran survive such translation with difficulty.

At certain moment necessary strategy may thus require fluency in the Fortran/OO languages and physics content of the project by the same person. Platform independent tools for mixing code in Fortran and OO languages might be of great help.

In this paper we have also documented for the first time technical side of the universal interface for the TAUOLA package. From now its version number 1.10 is printed in the output.

# Acknowledgements

Authors are grateful to ALEPH, CLEO, DELPHI and OPAL collaborations for providing their appropriate versions of the whole or parts of TAUOLA initialization. Useful discussions and suggestions from B. Bloch, C. Biscarat, S. Jadach, J. H. Kühn, M. Peskin, S. Slabospitzky and A. Weinstein are also acknowledged.

## References


[1] S. Jadach, J. H. Kühn, and Z. Wąs, *Comput. Phys. Commun.* **64** (1990) 275.

[2] M. Jeżabek, Z. Wąs, S. Jadach, and J. H. Kühn, *Comput. Phys. Commun.* **70** (1992) 69.

[3] S. Jadach, Z. Wąs, R. Decker, and J. H. Kühn, *Comput. Phys. Commun.* **76** (1993) 361.

[4] E. Barberio, B. van Eijk, and Z. Wąs, *Comput. Phys. Commun.* **66** (1991) 115.

[5] E. Barberio and Z. Wąs, *Comput. Phys. Commun.* **79** (1994) 291–308.

[6] P. Golonka, E. Richter-Wąs, and Z. Wąs, `hep-ph/0009302`.

[7] R. Decker, M. Finkemeier, P. Heiliger, and H. H. Jonsson, *Z. Phys.* **C70** (1996) 247–254, `hep-ph/9410260`.

[8] ALEPH Collaboration, B. Bloch, to contact send a mail to `Brigitte.Bloch-Devaux@cern.ch`.

[9] CLEO Collaboration, A. Weinstein, see `http://www.cithep.caltech.edu/~ajw/korb_doc.html#files`.

[10] A. E. Bondar *et al.*, *Comput. Phys. Commun.* **146** (2002) 139–153, `hep-ph/0201149`.

[11] H. Czyż and J. H. Kuhn, "*Four pion final states with tagged photons at electron positron colliders*", *Eur. Phys. J.* **C 18** (2001) 497–509.

[12] S. Jadach, B. F. L. Ward, and Z. Was, *Comput. Phys. Commun.* **130** (2000) 260–325, `hep-ph/9912214`.

[13] T. Pierzchała, E. Richter-Wąs, Z. Wąs, and M. Worek, *Acta Phys. Polon.* **B32** (2001) 1277–1296, `hep-ph/0101311`.

[14] Z. Wąs and M. Worek, *Acta Phys. Polon.* **B33** (2002) 1875–1884, `hep-ph/0202007`.

[15] G. R. Bower, T. Pierzchała, Z. Wąs, and M. Worek, *Phys. Lett.* **B543** (2002) 227–234, `hep-ph/0204292`.

[16] K. Desch, Z. Wąs, and M. Worek, *Eur. Phys. J.* **C29** (2003) 491–496, `hep-ph/0302046`.

[17] K. Desch, A. Imhof, Z. Was, and M. Worek, `hep-ph/0307331`.

[18] T. Sjostrand and M. Bengtsson, *Comput. Phys. Commun.* **43** (1987) 367.

[19] OPAL Collaboration, G. Abbiendi *et al.*, *Eur. Phys. J.* **C13** (2000) 197, `hep-ex/9908013`.

[20] DELPHI Collaboration, P. Abreu *et al.*, *Phys. Lett.* **B426** (1998) 411–427.

[21] Particle Data Group Collaboration, C. Caso *et al.*, *Eur. Phys. J.* **C3** (1998) 1–794.

[22] Chair W. Pokorski, N. Brook, session on decay packages of *" CERN Workshop on Monte Carlo tools for the LHC "* , July 7 - Aug 1 2003, CERN Geneva, `http://agenda.cern.ch/fullAgenda.php?ida=a031540`.

[23] G. Corcella, I. G. Knowles, G. Marchesini, S. Moretti, K. Odagiri, P. Richardson, M. H. Seymour, and B. R. Webber, *JHEP* **0101** (2001) 010, `hep-ph/0011363`.

[24] B. Kersevan and E. Richter-Wąs, *Comp. Phys. Commun.* **149** (2003) 142, `Available at http://cern.ch/Borut.Kersevan/AcerMC.Welcome.html`.

[25] T. Sjostrand, *Comput. Phys. Commun.* **82** (1994) 74–90.

[26] S. Jadach, B. F. L. Ward, and Z. Wąs, *Comput. Phys. Commun.* **79** (1994) 503.

[27] S. Jadach, B. F. L. Ward, and Z. Wąs, `hep-ph/9905205`.

[28] S. Jadach, B. F. L. Ward, and Z. Was, *Eur. Phys. J.* **C22** (2001) 423–430, `hep-ph/9905452`.

[29] S. Jadach and Z. Was, *Acta Phys. Polon.* **B15** (1984) 1151.

[30] M. Kramer, J. H. Kühn, M. L. Stong, and P. M. Zerwas, *Z. Phys.* **C64** (1994) 21, `hep-ph/9404280`.

[31] B. Grządkowski and J. F. Gunion, *Phys. Lett.* **B350** (1995) 218, `hep-ph/9501339`.

[32] A. Andonov, S. Jadach, G. Nanava, and Z. Wąs, *Acta Phys. Polon.* **B34** (2003) 2665–2672, `hep-ph/0212209`.

[33] P. Golonka, T. Pierzchała, and Z. Wąs, `hep-ph/0210252`.

[34] `http://cern.ch/Piotr.Golonka/MC/MC-TESTER`.

[35] `http://www.cern.ch/wasm/goodies.html`.

[36] Particle Data Group Collaboration, K. Hagiwara *et al.*, *Phys. Rev.* **D66** (2002) 010001.

[37] T. Pierzchala, "The extension of `TAUOLA` Monte Carlo library, simulating $\tau$ lepton decays", PhD Thesis, Silesian University 2004, see home page at `http://hpjmiady.ifj.edu.pl/~pierzchala`.

[38] P. Golonka, MSc Thesis written under supervision of Z. Wąs, see home page at `http://cern.ch/Piotr.Golonka/MC/photos`.

[39] G. Nanava and Z. Wąs, `hep-ph/0303260`.

[40] T. Sjostrand *et al.*, *Comput. Phys. Commun.* **135** (2001) 238–259, `hep-ph/0010017`.

[41] Talks given by B. Kersevan and M. Worek, presented at the *"Workshop on Monte Carlo tools for the LHC"*, CERN, July 2003.