TPJU 10/2000

The tauola-photos-F environment for versioning the TAUOLA and PHOTOS packages †

P. Golonka[a,b], **E. Richter-Wąs**[a,c,d] *and* **Z. Wąs**[a,e]

[a]*Institute of Nuclear Physics, ul. Kawiory 26a, 30-055 Cracow, Poland.*
[b]*Faculty of Nuclear Physics and Techniques, Cracow, Poland.*
[c]*Department of Computer Science, Jagellonian University*
[d]*CERN, PPE, CH-1211 Geneva 23, Switzerland.*
[e]*CERN, Theory Division, CH-1211 Geneva 23, Switzerland.*

Abstract

We present the system for versioning two packages: the `TAUOLA` of τ 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); (3) type of application: stand-alone, universal interface through `HEPEVT`, interface for `KKMC` Monte Carlo; (4) random number generators; (5) compiler options.

---

† Work supported in part by Polish Government grants KBN 2P03B11819, the Maria Skłodowska-Curie Joint Fund II PAA/DOE-97-316, and the Polish–French Collaboration within IN2P3 through LAPP Annecy.

Home page at http://wasm.home.cern.ch/wasm/

TPJU 10/2000
August 2000

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 τ-lepton measurements. Often program versions differ because of the requirements of interfaces to other packages used in the simulation chains (eg. format of the event record has to be adjusted).

Present utility setup for constructing specific versions of `TAUOLA` and `PHOTOS` is prepared for the software librarians and advanced users interested in updating both packages in the multipurpose environment. 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 exactly the same way as the published versions of the packages.

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, refs. [1–3] and [4, 5].

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, maximal number of entries).

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 τ-lepton(s); and (C) different versions of the `HEPEVT` common block:

   - (A) Different physics initializations:
     (1) As published in [3];
     (2) As initialized by ALEPH collaboration [6] (it is suggested to use this version only with the help of the collaboration advice);
     (3) As initialized by CLEO collaboration [7] (see printout of this version for details);
     (4) Further coding of some individual decay modes.

   - (B) Different interfaces with MC generators:
     (1) Old demo program as in published version [3];
     (2) Interface to `KKMC` [8];
     (3) New universal interface using `HEPEVT` common block.

   - (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 tools are used in the discussed setup: cpp the C-language pre-compiler: its `if`, `elif` and `include` commands, as well as Unix logical 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. `demo-factory`: Directory for updating input in demo files. For specialized use only, see section 2.3.

4. `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`.

5. `include/`: Directory containing the `HEPEVT-xxx.h` files for different versions of the `HEPEVT` common block. The logical link `HEPEVT.h` to the one actually used will be placed in this directory later.

subdirectories necessary to run demo and,
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. `platform/`: Directory containing system-dependent versions of `make.inc` files for supported platforms.

4. `make.inc`: Logical 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 to physics event generators based on the `HEPEVT` common block. In this demo `HEPEVT` is filled from JETSET74 [9] Monte Carlo generator.
   (c) `tauola/demo-KK-face`: Interface to KK Monte Carlo [8].

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
2. `2kD-all` – dimension 2000 double precision
3. `4kD-all` – dimension 4000 double precision
4. `2kR-all` – dimension 2000 single precision
5. `10kD-all` – dimension 10000 double precision

The action of creating required version of the library is performed with the help of cpp pre-compiler. It creates file `photos.f` from file `photos.F`. Once it is done, the logical link to the required version of the `HEPEVT` common block is created. 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; (aleph) as adopted by ALEPH collaboration, (cleo) as adopted by CLEO collaboration.

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, how they match the actual option of `TAUOLA` library, are mandatory. At present, code used in refs. [10] and [11] 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 τ-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, eg. compatible with the ones used by the collaboration, should replace files:

  – `./photos-random.h`,

  – `./tauola-random.h`

  with the files including the appropriate wrappers of his 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

Universal interface to different Monte Carlo generators is provided through event record `HEPEVT`. As a demonstration example it is interfaced with `JETSET` generator, however it should work in the same manner with `PYTHIA, HERWIG` or `ISAJET` generators.

- τ-lepton should be forced to be stable in the event generator.

- Content of the HEPEVT common block is searched for all τ leptons and neutrinos.
- It is checked if there are τ-flavour pairs (two τ-leptons or τ-lepton and τ-neutrino) originating from the same mother.
- Decay of the τ-flavour pairs are performed with TAUOLA. In some cases spin correlations are included explicitly. This is the case of the decay of W- and Z-bosons: $W \to \tau\nu$ and $Z \to \tau\tau$ and decays of scalar and pseudoscalar Higgses: $H \to \tau\tau$, $A \to \tau\tau$, $H^{\pm} \to \tau\nu$. At present it is treated in the incomplete manner only. Parallel or antiparallel spin configurations are generated, but for $\tau\tau$ pairs originating from Z/γ each possibility is taken with 50 % chances.
- Photon radiation in decay is performed with PHOTOS.

Finally let us note that the calculation of the τ polarization created from the Z and/or virtual γ (as function of the direction), represents rather non trivial extension. Dedicated study of the production matrix elements of the host generator is necessary. Separate paper [12] will be devoted to this point.

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 you need to clone/write it.
   - 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`

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/jetset-demo` 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 [8]. 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, either directories `./jetset` and `./glibk` need to be kept or replaced by the appropriate links. The `make.inc` logical link pointing to `make-xxxx.inc` file in directory `./platform`, defining appropriate compiler flags, need to be kept also.

5 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.

This 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 [13]. 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.

Necessary strategy may thus require fluency at certain moment 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 also.

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, S. Jadach, J. H. Kühn and A. Weinstein are also acknowledged. One of the authors (ZW) acknowledges warm hospitality of the Zurich ETH L3 group at the final stage of the project completion.

References

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

[2] M. Jezabek, Z. Was, S. Jadach, and J. H. Kuhn, *Comput. Phys. Commun.* **70** (1992) 69.

[3] S. Jadach, Z. Was, R. Decker, and J. H. Kuhn, *Comput. Phys. Commun.* **76** (1993) 361–380.

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

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

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

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

[8] S. Jadach, Z. Wąs, and B. F. L. Ward, *Comput. Phys. Commun.* **130** (2000) 260; ;p to date source is available from http://home.cern.ch/jadach/.

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

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

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

[12] Z. Was *et al.*, unpublished.

[13] P. Golonka, MSc thesis written under supervision of Z. Wąs; P. Golonka's home page is at http://lhotse.ifj.edu.pl/~piters.