New functionality of the CompHEP 4.6 package

E.E. Boos,[1,*] V.E. Bunichev,[1,†] M.N. Dubinin,[1,‡] L.V. Dudko,[1,§]
V.F. Edneral,[1,¶] A.P. Kryukov,[1,**] and V.I. Savrin[1,††]

[1]*Skobeltsyn Institute of Nuclear Physics,*
Lomonosov Moscow State University 119991 Moscow, Russia

(Received xx.xx.2025; Revised xx.xx.2025; Accepted xx.xx.2025)

Abstract

A new version of the high energy physics computational package CompHEP has been released. This version adds the ability to perform calculations with complex form-factors, enabling the calculation of decay widths and scattering cross sections in next-to-leading order according to perturbation theory. A new CompHEP module has also been developed for the automatic, sequential calculation of arrays of decay widths and scattering cross sections as functions of multidimensional arrays of model parameter values. CompHEP now includes a module for statistical processing of results and the construction of confidence regions in the model parameter space. A new module for graphical presentation of results has been added for constructing one-dimensional, two-dimensional and three-dimensional graphs in the ROOT package format. In addition, release 4.6.1 introduces support for the LHAPDF 6 library, parallel computation and unified event generation over all subprocesses, an update of the physical constants to PDG 2024, and cross-validation of the results against MadGraph5 and WHIZARD.

PACS numbers: 11.10.-z, 11.10.Ef, 12.60.-i, 14.65.Ha, 95.35.+d

Keywords: High energy physics, Feynman diagrams, Automation of calculations

1. INTRODUCTION

The increased accuracy of modern collider experiments has led to a demand for calculations of characteristics and modeling of processes with a higher level of theoretical certainty in next-to-leading order according to perturbation theory with complex form-factors at the interaction vertices. Modern models of new physics have many new parameters whose values are undefined. Typically, it is necessary to iterate through the entire multidimensional set of such values, perform calculations with each of the defined value sets, and compare the calculated results with experimental data. This chain of actions is extremely resource-intensive and carries the risk of errors. It is necessary to automate this process. To objectively and theoretically justify the comparison of modeling results with experimental data, it is necessary to apply modern statistical analysis methods to construct confidence regions in the model parameter space and identify the most reliable values of these parameters. To improve the capabilities of theoretical

* E-mail: boos@theory.sinp.msu.ru
† E-mail: bunichev@theory.sinp.msu.ru
‡ E-mail: dubinin@theory.sinp.msu.ru
§ E-mail: dudko@sinp.msu.ru
¶ E-mail: edneral@theory.sinp.msu.ru
** E-mail: kryukov@theory.sinp.msu.ru
†† E-mail: savrin@theory.sinp.msu.ru

analysis and identify the most sensitive characteristics, it is necessary to present the modeling results in a convenient graphical format in the form of one-dimensional, two-dimensional, and three-dimensional graphs that can be used for further analysis.

CompHEP[2] is a software package designed for modeling processes in high energy physics. To implement the requirements listed above, we expanded the functionality of the CompHEP package by adding capabilities for calculations with complex form-factors, enabling the calculation of decay widths and scattering cross sections in next-to-leading order according to perturbation theory. We also added capabilities for automatic sequential calculations of arrays of decay widths and scattering cross sections depending on multidimensional arrays of model parameter values. Next we added a module for statistical processing of calculation results and construction of confidence regions in the parameter space using experimental data. Finally, we added a module for extended graphical presentation of results: construction of one-dimensional, two-dimensional, and three-dimensional graphs in the ROOT[3] package format. These functions have been implemented within the CompHEP package graphical interface, as well as in batch mode. The programs implementing the new functionality are written in the "C" programming language. They were successively developed and debugged during a number of scientific studies [4–11] and, as a result, were generalized, modified, and combined within the new version of the CompHEP 4.6 package [1].

2. ADVANCED COMPHEP CAPABILITIES FOR COMPUTING WITH COMPLEX FORM-FACTORS

When calculating in the next-to-leading order according to perturbation theory, form-factors with a complex dependence on the momenta of the incoming particles arise at the vertices of the model. CompHEP now supports calculations with such form-factors. In previous versions of CompHEP, the list of available functions was limited to a small set of elementary functions. In the new version of CompHEP, the user can enter functions of any complexity, including those using the momenta of the incoming particles as arguments. Two scenarios for using user-defined form-factors have been implemented.

2.1. Using form-factors in model vertex functions

In the first scenario, form-factors can be specified in the model's vertex functions while a CompHEP symbolic session is running. To do this, the user must define the body of their

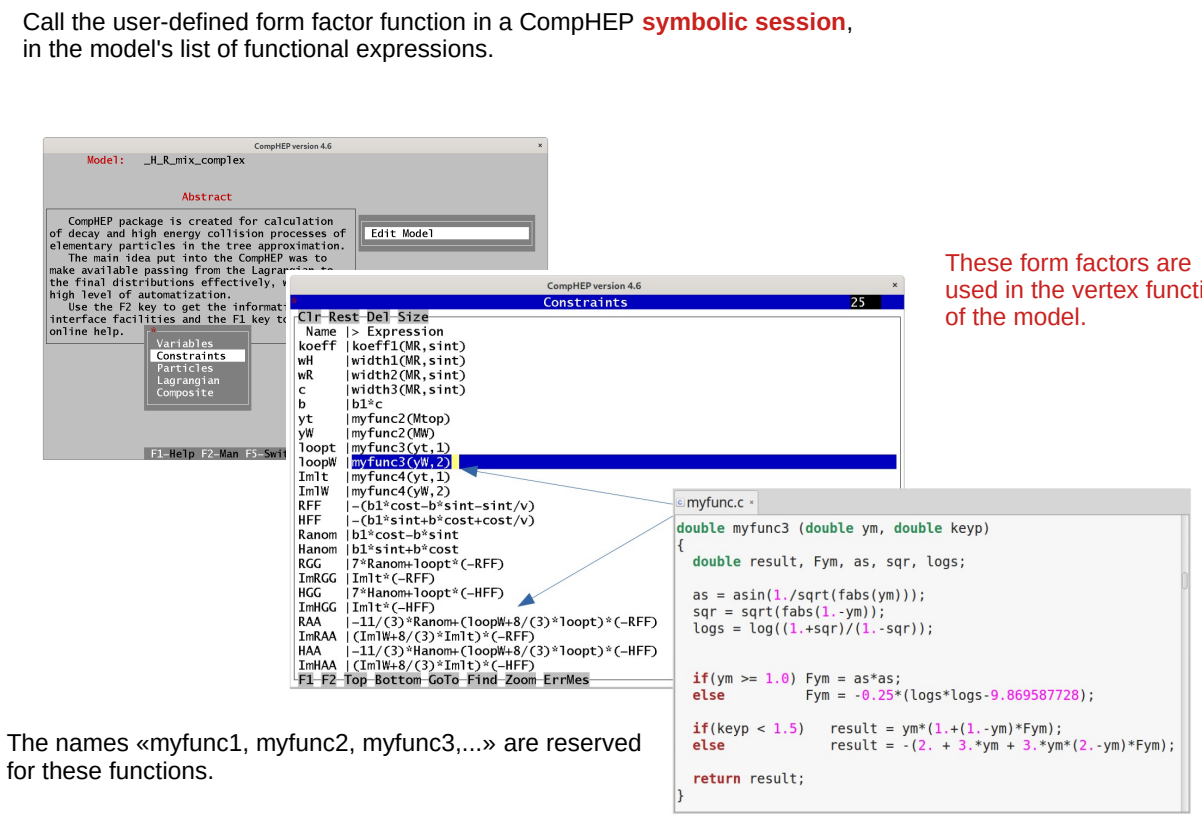


Figure 1. Calling a user-defined function defined in the file "myfuncN.c" in the list of functional expressions of the model in CompHEP.

function in one of the files with the reserved names "myfunc1, myfunc2, myfunc3,...". After defining them in the "myfuncN.c" file and recompiling CompHEP, these functions can be called and used in various expressions in the "Constraints" menu, which lists the model's functional expressions used in the current CompHEP model, as shown in Fig. 1.

This figure shows an example of the use of form-factors in vertex functions for a model with mixing of the radion field and the Higgs field [7]. The vertex functions of this model are given in Table 1, where the constants and the functions of the momenta are given by the following expressions: $b_{QCD} = 7$, $b_2 = \frac{19}{6}$, $b_Y = -\frac{41}{6}$, $C_h = \sin\theta + c\cdot\cos\theta$, $C_r = \cos\theta - c\cdot\sin\theta$, $F_W = -(2 + 3y_W + 3y_W(2 - y_W)f(y_W))$, $F_t = y_t(1 + (1 - y_t)f(y_t))$, $y_i = 4m_i^2/(2p_1\cdot p_2)$,

$$f(z) = \begin{cases} \left[\sin^{-1}\left(\frac{1}{\sqrt{z}}\right)\right]^2, & z \geq 1 \\ -\frac{1}{4}\left[\log\frac{1+\sqrt{1-z}}{1-\sqrt{1-z}} - i\pi\right]^2, & z < 1. \end{cases}$$

Triple vertices			Feynman rules
G_μ	G_ν	h	$\frac{g_s^2}{8\pi^2}\cdot\left[b_{QCD}\cdot\frac{C_h}{\Lambda_r}+F_t\cdot\left(\frac{C_h}{\Lambda_r}+\frac{\cos\theta}{v}\right)\right]\cdot\left(g^{\mu\nu}p_1p_2-p_1^\nu p_2^\mu\right)$
G_μ	G_ν	r	$\frac{g_s^2}{8\pi^2}\cdot\left[b_{QCD}\cdot\frac{C_r}{\Lambda_r}+F_t\cdot\left(\frac{C_r}{\Lambda_r}-\frac{\sin\theta}{v}\right)\right]\cdot\left(g^{\mu\nu}p_1p_2-p_1^\nu p_2^\mu\right)$
A_μ	A_ν	h	$\frac{e^2}{8\pi^2}\cdot\left[(b_2+b_Y)\cdot\frac{C_h}{\Lambda_r}+\left(F_W+\frac{8}{3}F_t\right)\cdot\left(\frac{C_h}{\Lambda_r}+\frac{\cos\theta}{v}\right)\right]\cdot\left(g^{\mu\nu}p_1p_2-p_1^\nu p_2^\mu\right)$
A_μ	A_ν	r	$\frac{e^2}{8\pi^2}\cdot\left[(b_2+b_Y)\cdot\frac{C_r}{\Lambda_r}+\left(F_W+\frac{8}{3}F_t\right)\cdot\left(\frac{C_r}{\Lambda_r}-\frac{\sin\theta}{v}\right)\right]\cdot\left(g^{\mu\nu}p_1p_2-p_1^\nu p_2^\mu\right)$

Table 1. Vertices of the interaction of the Higgs boson and radion with gluons and photons.

2.2. Using the global form-factor in a CompHEP numerical session

In the second scenario, it is possible to call a user-defined form-factor function during a CompHEP numerical session. To do this, the user must define the body of their function in the "userFactor.c" file. After recompiling CompHEP, this function can be called and used as a common multiplier of the full square of the process amplitude modulus. To activate the use of this form-factor, the user must enable the "Use Form Factor" option in the list of numerical procedures by switching it from "NO" to a non-zero key, as illustrated in Fig. 2.

3. USING TABLE-DEFINED FUNCTIONS IN FUNCTIONAL EXPRESSIONS OF THE MODEL

3.1. Using functions of multiple arguments, specified as a table, in the CompHEP model

An important innovation is the ability to use functions of several arguments specified in the form of a table in the CompHEP model. The file names "width1.txt, width2.txt, width3.txt, ..." are reserved for this purpose. The first two columns of the table file contain the argument values and the third contains the function value. The resulting table is copied to the CompHEP root directory and can be accessed in the list of functional expressions of the model, as demonstrated in Fig. 3.

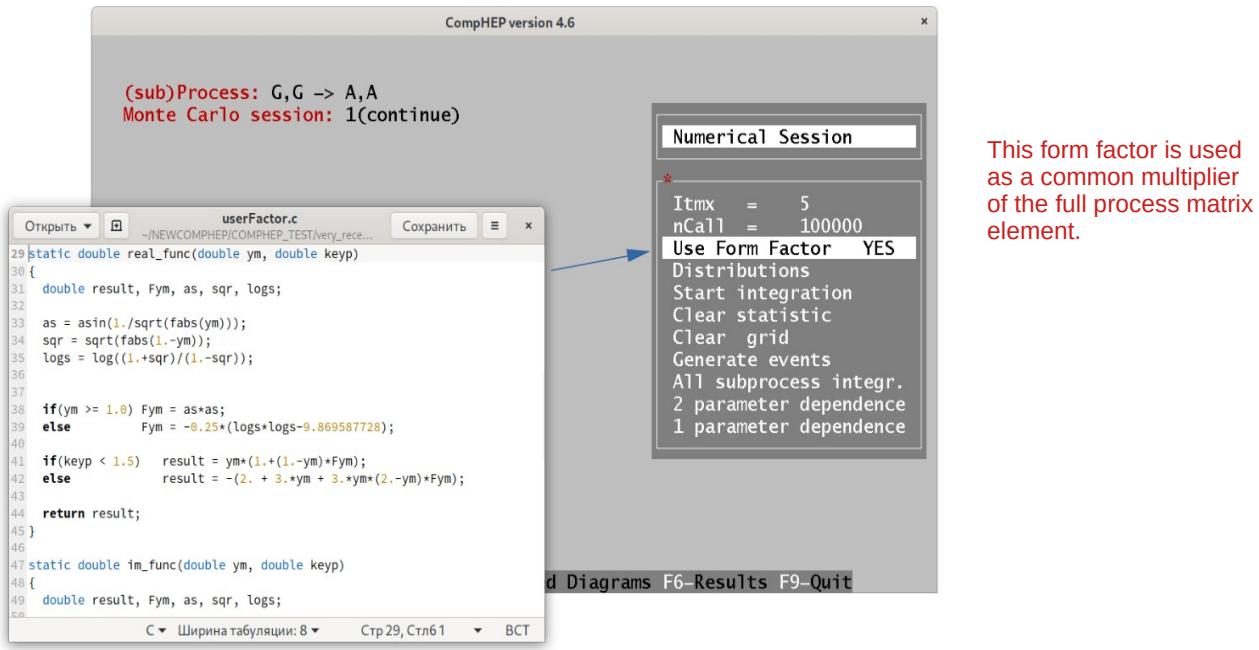


Figure 2. Calling a user-defined form-factor function during a CompHEP numerical session.

3.2. Multiplication any part of the squared amplitude modulus by a function specified in a table

Another new feature is the ability to multiply any part of the squared amplitude modulus by a function specified in a table. The file names “koeff1.txt, koeff2.txt, koeff3.txt, ...” are reserved for this purpose, as is the variable name “koeff” in the model constants file. A table named “koeffN.txt” must be copied to the CompHEP root directory and defined in the model file (Fig. 4). When viewing the squared amplitude modulus diagrams, you can then mark any of them using the “K” symbol on the keyboard. During integration, such diagrams will be multiplied by the function values from the “koeffN.txt” table, depending on the values of its argument. This option can be useful when calculating with higher-dimensional operators within SMEFT.

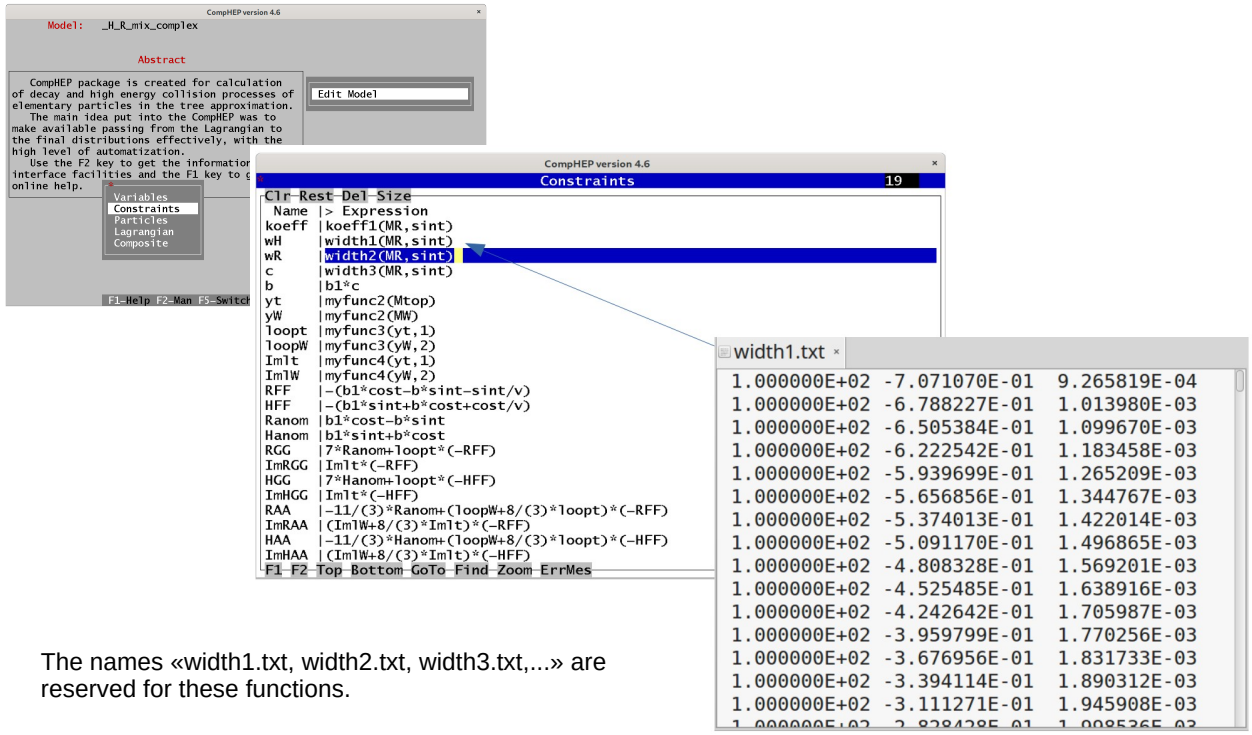
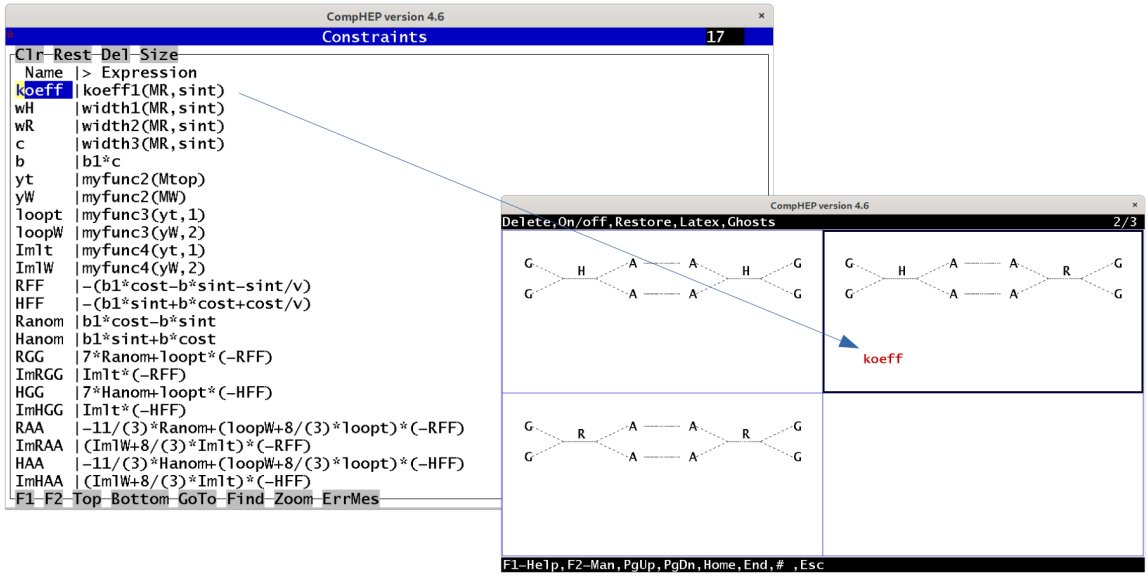


Figure 3. Calling a function specified as a table and defined in the file “width1.txt” in the list of functional expressions of the new model in CompHEP.

4. CALCULATION OF AN ARRAY OF DECAY WIDTH VALUES DEPENDING ON MULTIDIMENSIONAL ARRAYS OF MODEL PARAMETER VALUES

The previous section demonstrated the ability to use functions defined in a table in CompHEP. Using such a table, one can specify the dependence of the particle decay width on the model parameters. The new version of CompHEP implements functionality for creating a table of particle decay width values and then using it to calculate scattering cross sections. New capabilities are built into the standard CompHEP interface. To calculate an array of decay width values depending on the values of model parameters in the numerical session of decay width calculations, a new item has been added: “2 parameter depend.”, which opens the possibility of specifying parameters for sequential automatic calculation of decay width values depending on arrays of values of two model parameters. As shown in Fig. 5, after selecting this menu item, dialog submenus appear for selecting the first parameter, setting the range of its values and the

Multiplication of any part of the square of the amplitude modulus by a function, specified as a table of values and defined in the file "koeff1.txt" in the CompHEP **symbolic session** in the list of functional expressions of the model.



The names «koeff1.txt, koeff2.txt, koeff3.txt,...» are reserved for these functions.

Figure 4. Multiplication of a part of the square of the amplitude modulus by a function given in the form of a table.

number of partition points in this interval. After setting the range of values for the first parameter, menu items for setting the second parameter sequentially open (Fig. 5). After setting the second parameter, calculations of the particle decay width values are launched for all specified values of the parameter grid. In this case, the corresponding calculations are performed for all subprocesses. After completing the calculations, the parameter values and their corresponding widths are written to a file. The width values for different subprocesses are written to different files named “table_N.txt”, where N is the subprocess number.

The corresponding width values for different subprocesses can be summed and written to a separate file. A new menu item, “Sum 2d tables” (Fig. 6), is provided for this purpose. Selecting this menu item opens a dialog box for selecting table files from the list. After selecting the required files, the corresponding values are summed and written to the file “sum2d_1.txt”. Fig. 6 demonstrates the calculation of a table with the width values of two-body particle decays. A table of three-body particle decays can also be obtained, and then the partial widths

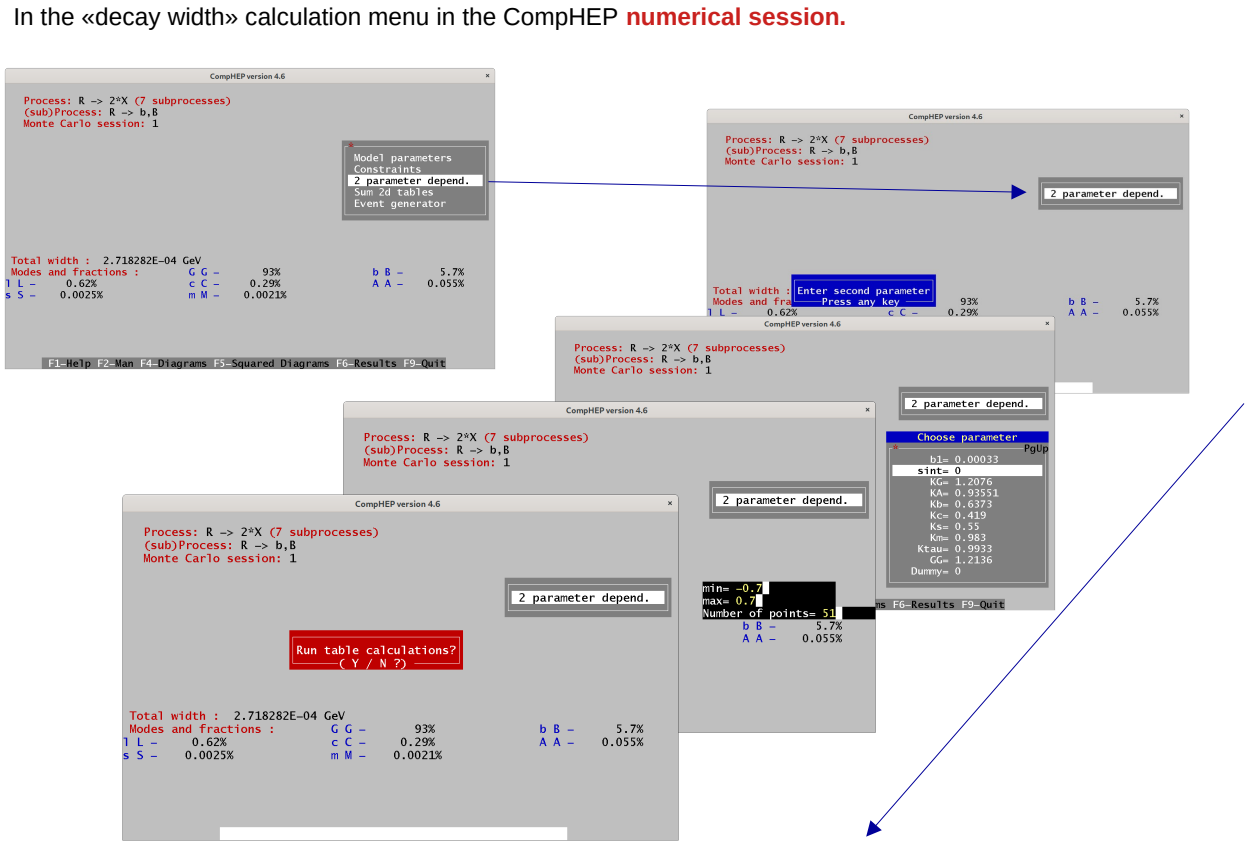


Figure 5. The sequence of steps within the CompHEP interface for calculation an array of particle decay width values as a function of multidimensional arrays of model parameter values. The direction of the sequence of steps is shown by blue arrows.

of two-body and three-body particle decays are summed in the manner described above. The resulting file can be renamed “widthN.txt”, copied to the root directory of CompHEP, and used when calling the function that determines the decay width of a new particle in the new model file, as already demonstrated in Fig. 3.

Thus, we have defined the new particle width, based on the values of two parameters. The decay width determined in this way will be used to calculate the scattering cross sections for the complete processes involving particle production and decay, depending on the model parameters.

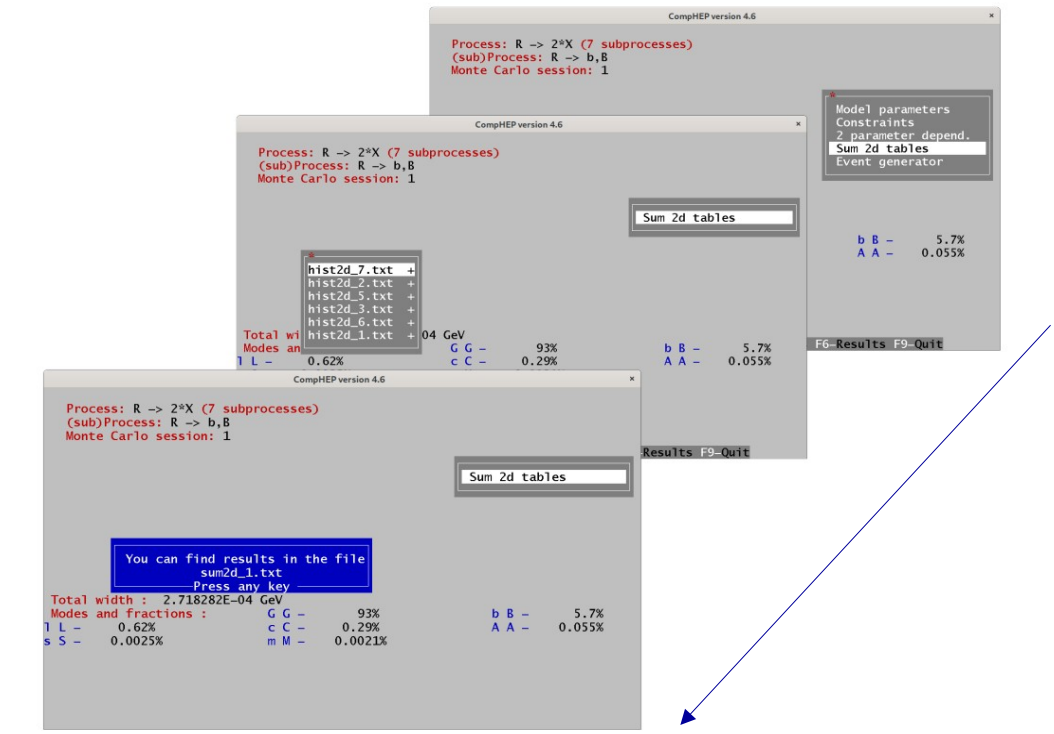


Figure 6. Menu items for summing the corresponding decay width values for the selected tables. The direction of the sequence of actions is shown by blue arrow.

5. CALCULATION OF AN ARRAY OF SCATTERING CROSS-SECTION VALUES DEPENDING ON MULTIDIMENSIONAL ARRAYS OF MODEL PARAMETER VALUES

5.1. Calculation within the CompHEP graphical interface

The previous section demonstrated the creation of tables with particle width values that can be used in further calculations. Now we will demonstrate the new CompHEP functionality for calculating cross-section tables for multiparticle processes depending on two parameters. This takes into account the dependence of the particle decay widths on these parameters. To perform such calculations, in the CompHEP cross-section calculation numerical session, select the new item: “2 parameter dependence” and repeat the sequence of actions (previously shown for calculating the width) for selecting two model parameters, specifying the ranges of their

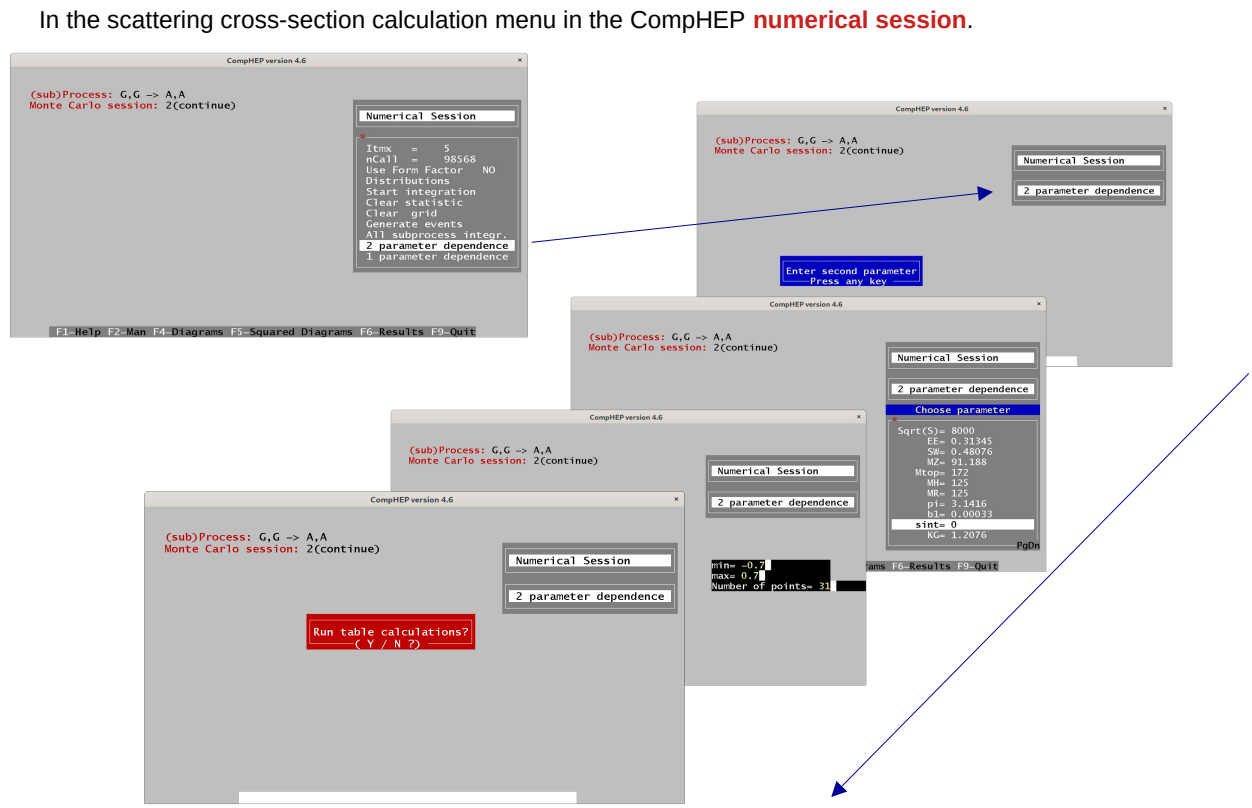


Figure 7. Menu items for creating a table of process cross-section values as a function of multidimensional arrays of model parameter values, setting their value ranges, and the number of split points for both intervals. The direction of the sequence of actions is shown by blue arrows.

values, and the number of partition points for both intervals (Fig. 7). It should be kept in mind that the choice of parameters for calculating cross-sections depending on their values, the range of these values, and the partition steps for the parameter intervals must be the same as when calculating the corresponding tables for particle decay widths. Next, the cross-section values are calculated on a given parameter grid. When calculating the values of the process matrix element, the particle width values are called up from previously obtained tables. Tables with cross-section values for different subprocesses are written into different files with the names: “table_N.txt”, where N is the subprocess number.

Similar to the width calculation, new “Distributions → Sum 2d tables” items have been added to the cross-section menu for summing the corresponding cross-section values from tables for different subprocesses, as shown in Fig. 8. In addition to the summation function, additional functions for algebraic operations with tables have been added to the menu. A detailed overview

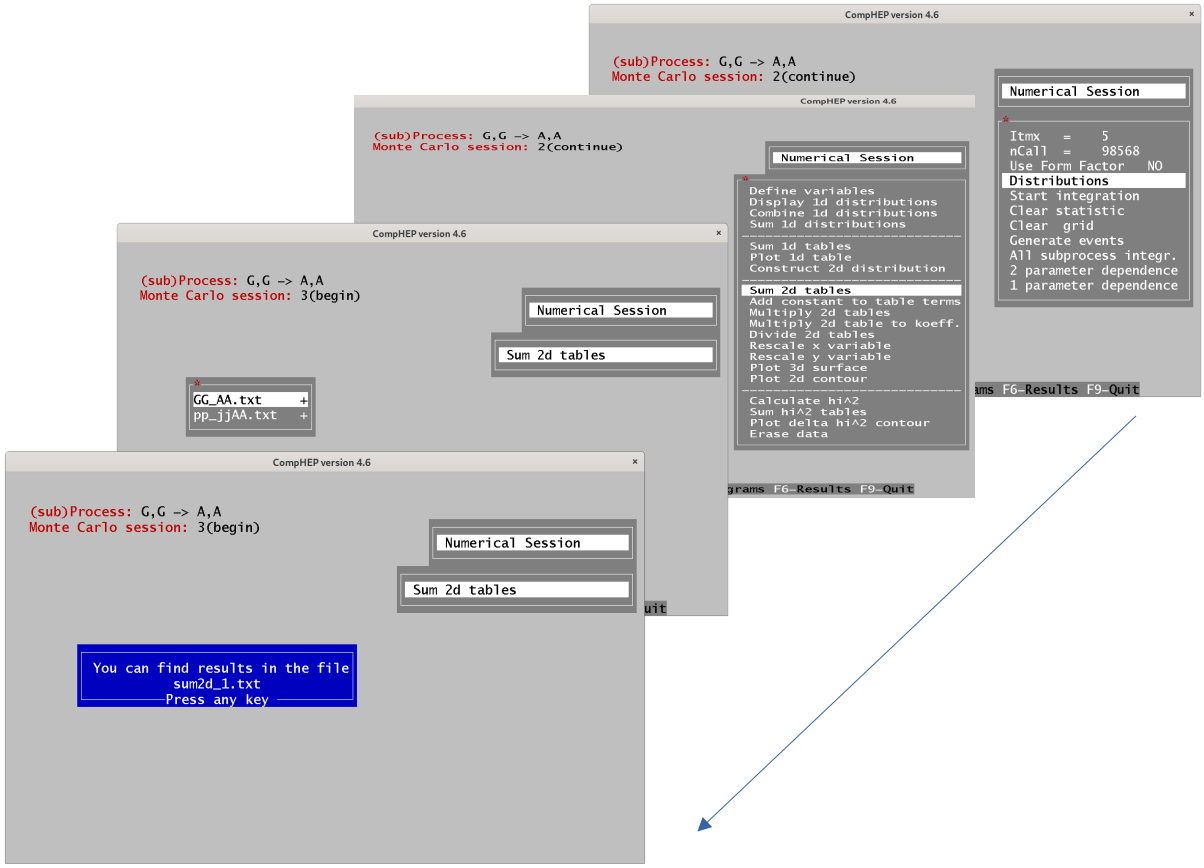


Figure 8. Menu items for summing the corresponding scattering cross-section values for selected tables. The direction of the sequence of actions is shown by blue arrow.

of these functions is presented in Section 7.

5.2. Calculation within the CompHEP batch mode

The previous section demonstrated the ability to calculate the scattering cross-section as a function of multidimensional arrays of model parameter values within the CompHEP graphical interface. This is very convenient for processes with a small number of final states, a small number of subprocesses, and a small number of parameter grid points. Such tasks can be quickly run on a personal computer. However, for complex processes and a large number of parameter grid points, the ability to run calculations remotely on powerful computing clusters is required. For this purpose, CompHEP has implemented the functionality for running cross-section calculations on the parameter grid in batch mode. Before running calculations in batch mode, you must go through the entire chain of parameter settings, their ranges, and grid partition

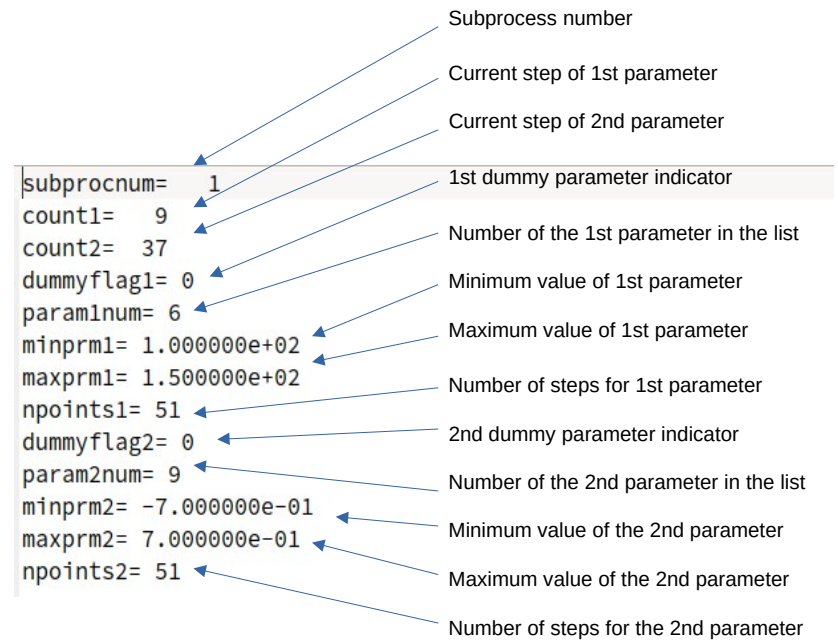


Figure 9. Settings parameters of the “tablecalc.dat” file.

intervals in graphical mode, as shown in the previous section in Fig. 8. In the dialog box, when asked whether to continue the calculation “Run table calculations Y/N,” select “No” by pressing the “N” key. This will create a text file with the calculation run settings “tablecalc.dat” in the current working directory. This file can be left unchanged or edited, depending on the task at hand. An example of the settings for the “tablecalc.dat” file is shown in Fig. 9. Now you need to save the results and close the CompHEP session. After that, go to the directory with the saved results and run the “./table_batch” command in the terminal in command mode. After running the command, a sequential calculation of the cross-section values on the specified parameter grid will begin in the background. In this case, the calculations will be automatically performed for all subprocesses and the corresponding result tables will be written to different files with the names “table_N.txt”, where N is the subprocess number.

6. CALCULATION OF THE χ^2 FUNCTION AND CONSTRUCTION OF CONFIDENCE REGIONS IN THE MODEL PARAMETER SPACE

In the previous sections, we demonstrated how to obtain tables of cross-section values as functions of two parameters. These tables can now be used to estimate model parameter exclusion regions (or parameter confidence regions). For this purpose, the CompHEP interface has been enhanced with the ability to calculate tables of partial χ^2 functions for each subprocess, sum the values of the partial χ^2 tables, and construct exclusion contours for given model parameters. CompHEP utilizes the statistical analysis methods described in [12–16]. Parameter confidence regions are determined by finding the minimum of the χ^2 function. The complete function χ^2 is defined as:

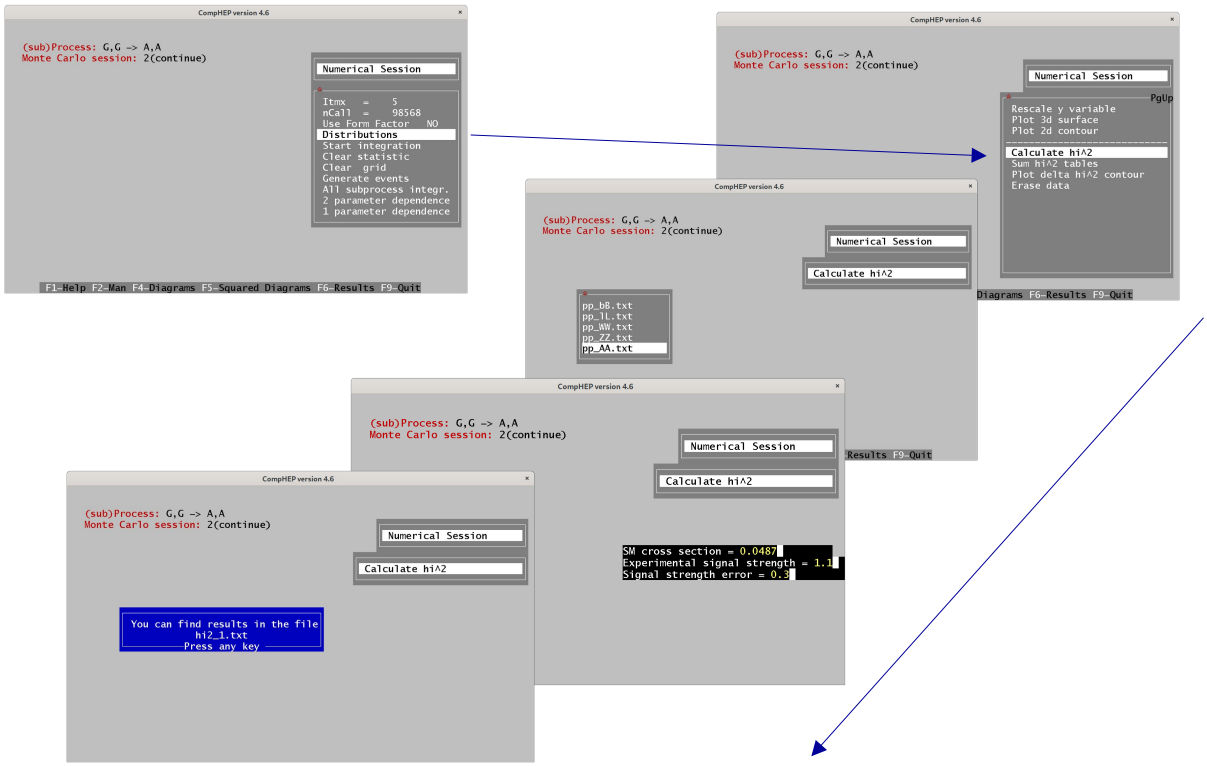
$$\chi^2(\mu_i) = \sum_i^{N_{ch}} \frac{(\mu_i - \hat{\mu}_i)^2}{\sigma_i^2}, \tag{1}$$

where N_{ch} is the number of channels for the production of a new particle, μ_i is the theoretical (fitted) value of the signal strength for a given channel, $\hat{\mu}_i$ is the experimental value of the signal strength for a given channel, σ_i is the measurement error of the experimental signal strength for a given channel. The fitted value of the signal strength is defined as the cross section of the process predicted by the new model divided by the cross section predicted by the Standard Model: $\mu_i = \sigma_{BSM,i}/\sigma_{SM,i}$. The experimental value of the signal strength is calculated as:

$$\hat{\mu}_i = \frac{N_{obs,i} - N_{backgr,i}}{N^{SM}_{signal,i}}, \tag{2}$$

where $N_{obs,i}$ is the total observed number of events, $N_{backgr,i}$ is the number of background events, $N^{SM}_{signal,i}$ is the number of events predicted by the Standard Model. Finding the minimum of the function χ^2 allows us to determine the 68%, 90%, and 99% confidence regions of the parameters using the formula $\chi^2 = \chi^2_{min} + \Delta\chi^2$, where $\Delta\chi^2$ is the quantile of χ^2 and is determined by the cumulative distribution function. If different channels of new particle production are weakly correlated and the signal strengths corresponding to them have a Gaussian distribution, then the overall probability density function can be obtained by multiplying the partial probability density functions of the individual channels, and the combined χ^2 can be obtained by summing the partial χ^2 of the individual channels. To construct tables of partial χ^2 functions, select the new item "Distributions → Calculate hi2" in the CompHEP menu for calculating cross sections. This will open a menu for selecting tables with cross section values for various processes, as shown in Fig. 10. After selecting a process table, enter sequentially: the Standard Model cross section value for the given process → the signal strength value for the given process obtained from the experiment → the signal strength measurement error value for the given process. This

In the distribution construction menu in the CompHEP **numerical session**.



The table of χ^2 values is written to the file “hi2_1.txt”.

Figure 10. Menu items for selecting a scattering cross-section table for a given process to calculate the corresponding partial function χ^2 table. The direction of the sequence of actions is shown by blue arrows.

will create a table of the dependence of the partial χ^2 function for the given process on two parameters (Fig. 10).

The combined χ^2 function can be obtained by summing the values of the partial χ^2 for different processes, as well as the corresponding partial χ^2 for different experiments. To do this, select the “Distributions → Sum hi2” item (Fig. 11). Now this combined table can be used to plot the exclusion contours of the model parameters. To plot the exclusion contours, select the “Distributions → Plot delta hi2” menu item (Fig. 12). In this case, the program searches for the minimum in the combined χ^2 table and plots the values of the two-dimensional quantiles corresponding to the 68%, 90% and 99% confidence levels relative to this minimum. At the χ^2 heights, corresponding to these levels, the program makes slices by parallel planes. The intersection of these cutting planes with the graph of the two-dimensional χ^2 function forms two-dimensional parameter exclusion contours at the 68%, 90%, and 99% confidence levels. The

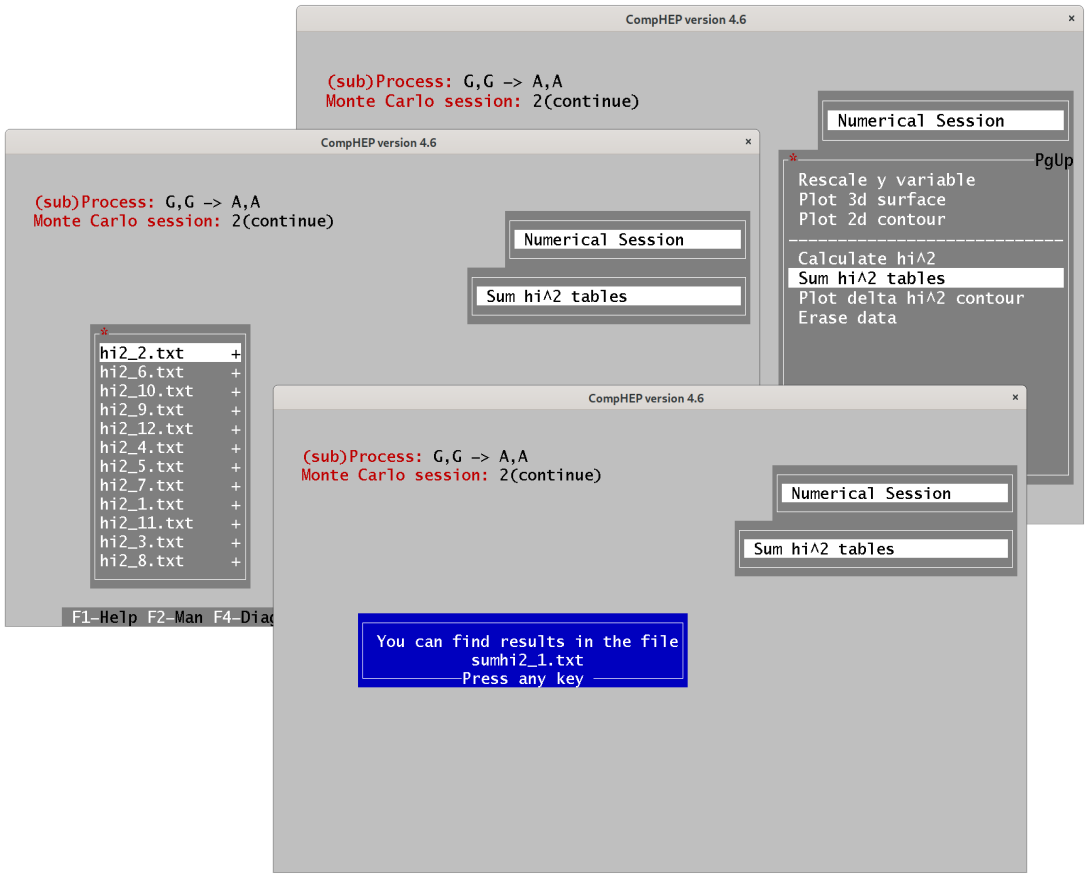


Figure 11. Menu items for summing the corresponding partial χ^2 values for the selected tables. Direction of action: from the top image to the bottom.

result of this analysis is written as a "C++" program in the ROOT package format. It should be noted that the new version of CompHEP adds new functions for presenting tables with results in the ROOT package format, allowing for further calculations and presentation in popular graphical formats. An overview of this new functionality is presented in Section 8.

7. ALGEBRAIC MANIPULATIONS WITH TABLES OF DECAY WIDTHS AND SCATTERING CROSS SECTIONS

In addition to creating tables from width and cross-section values, the new version of CompHEP includes functions for algebraic manipulation with these tables. The summation function was already presented in previous sections. The user can also add a constant to all table values, multiply corresponding values from two tables, multiply all table values by a constant, and divide values from one table by corresponding values from another. Functions for rescaling the

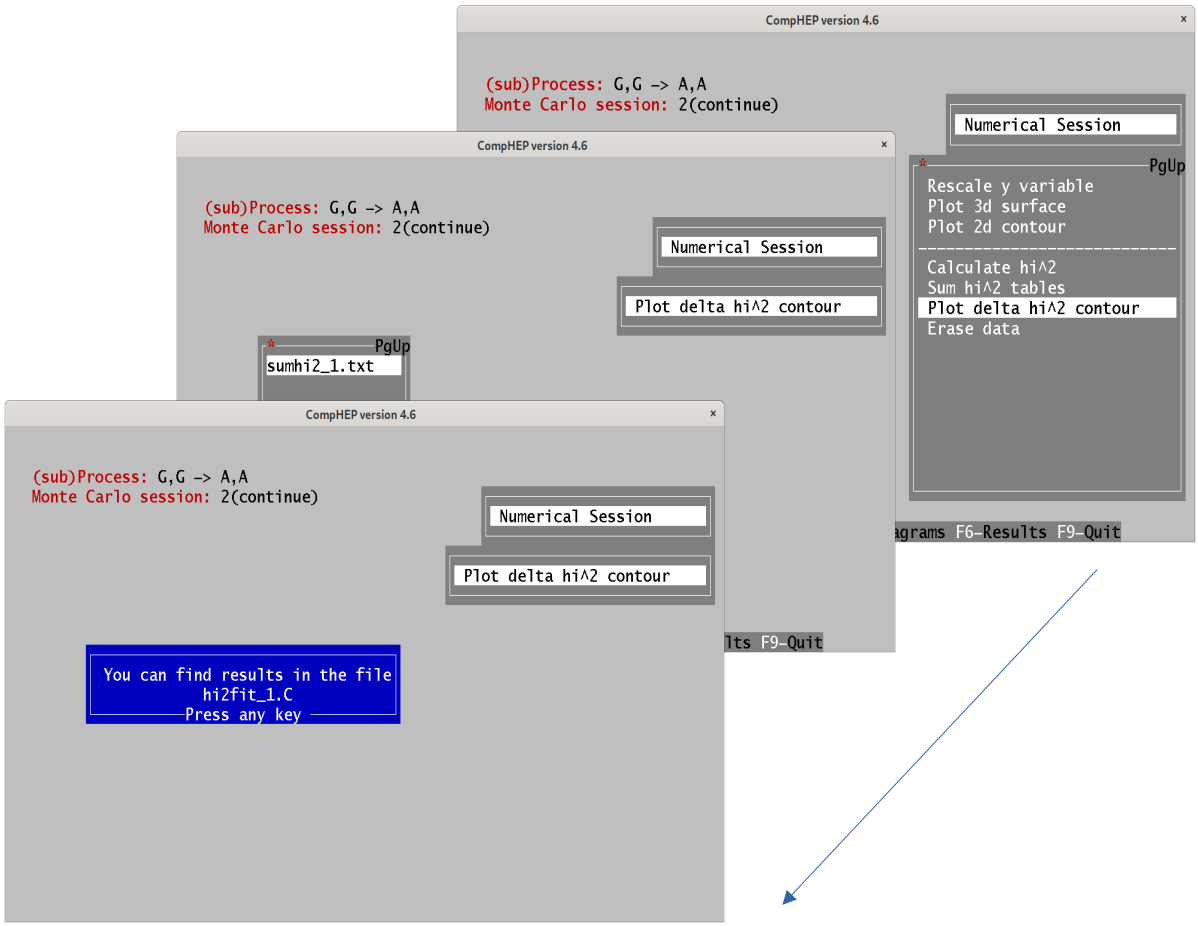


Figure 12. Menu items for constructing exclusion contours for given model parameters. The direction of the sequence of actions is shown by blue arrow.

values of both parameters are also provided. Fig. 13 shows the CompHEP dialog menu with the listed functions. It should be noted that the function for dividing values from one table by values from another can be used to construct tables of relative particle decay widths, and the function for multiplying values from two tables can be used to calculate the cross section for the complete process of particle production followed by decay using the formula: $\sigma_{full} = \sigma_{production} \times Br_{decay}$.

8. OVERVIEW OF NEW COMPHEP FEATURES FOR GRAPHICAL PRESENTATION OF RESULTS

To facilitate the presentation of results, CompHEP includes built-in routines for writing tables of one-dimensional and multidimensional functions as "C++" programs, in a format suitable for running and further processing in the ROOT analysis package. These routines are

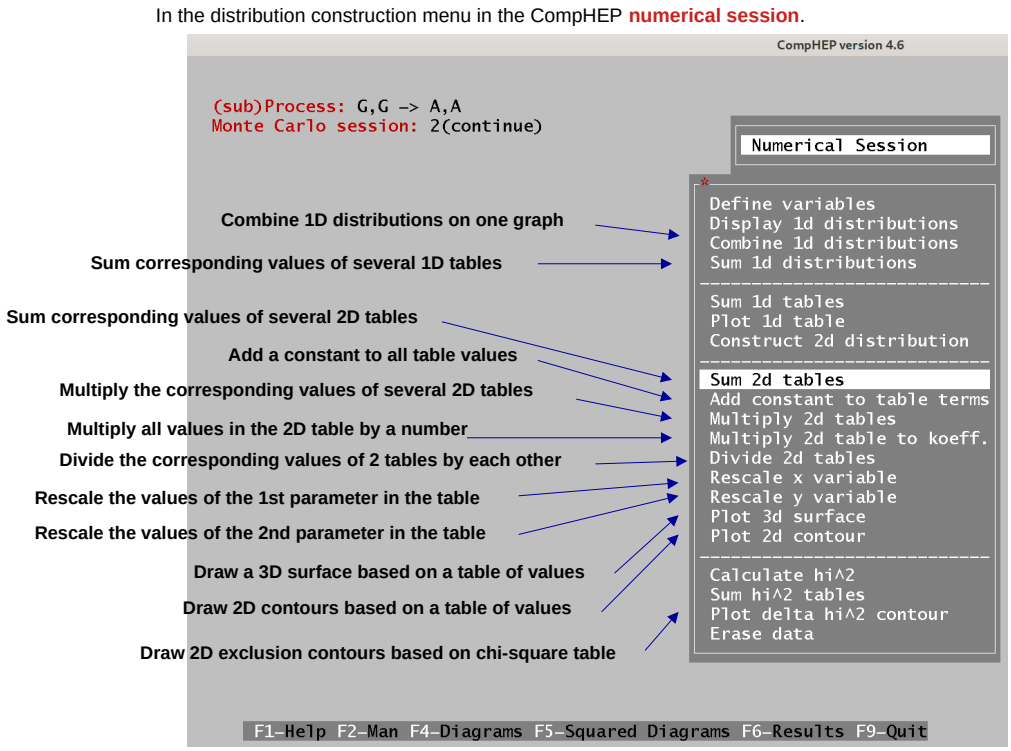


Figure 13. Menu items for algebraic manipulation of tables of values and items for graphical presentation of results.

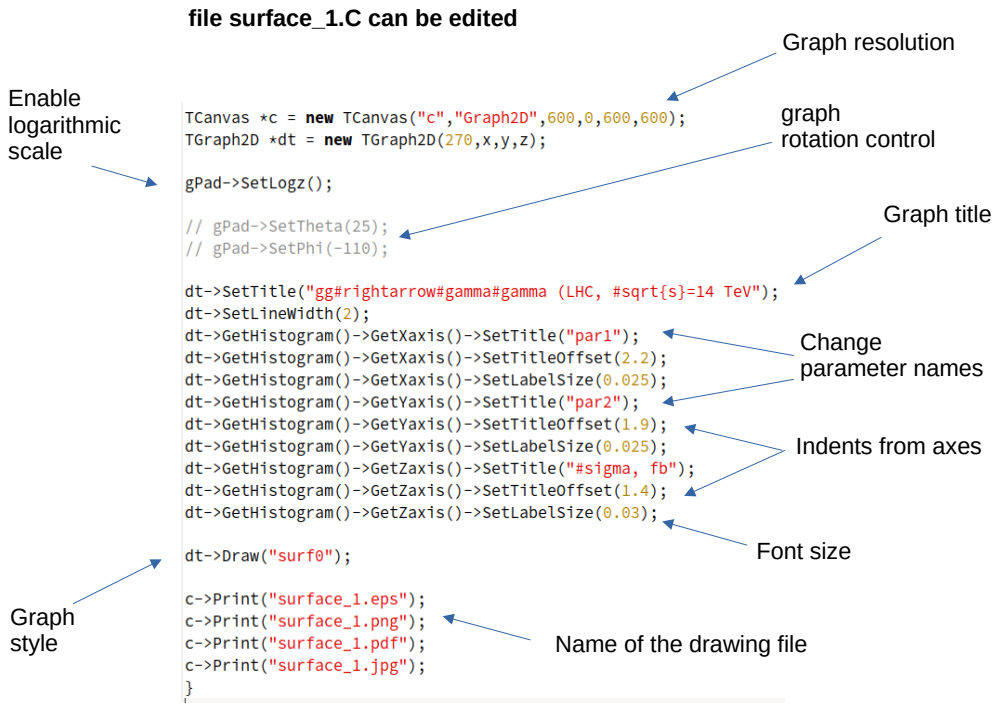


Figure 14. Graphic settings parameters for the 2D image file "surface_1.C"

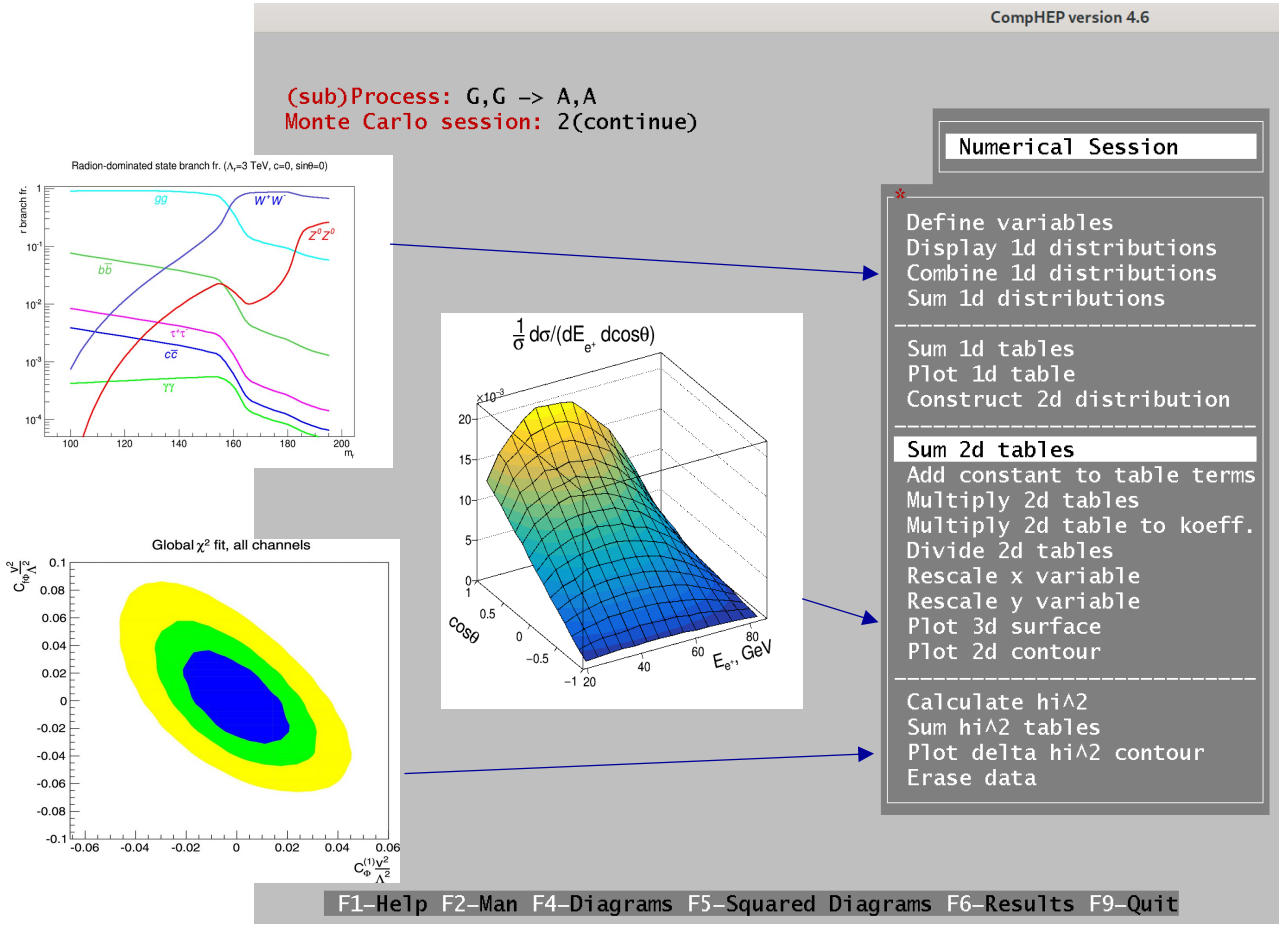


Figure 15. Examples of the new CompHEP functions for graphical presentation of results.

grouped together in the CompHEP numerical session interface for calculating cross-sections in the "Distributions" section. Fig. 13 shows the menu items with the locations of these routines.

The list of routines is divided into several blocks. The first block is designed for working with one-dimensional distributions of kinematic variables, the second block is for working with tables of cross-section values depending on a single model parameter, and also for constructing 3D graphs of the dependence of kinematic distributions on one model parameter. The third block includes functions for processing tables depending on two model parameters. The fourth block is designed for working with two-dimensional tables of the χ^2 function. For example, the "Plot 3d surface" command creates the file "surface_1.C", which can be run in the ROOT environment with the command ".x surface_1.C". A 3D surface image will be created in PDF, EPS, PNG, and JPG formats. The graphic settings for the created "C++" program are shown in Fig. 14. Examples of using CompHEP's new functions for graphical representation of results are demonstrated in Fig. 15.

9. NEW FUNCTIONALITY IN COMPHEP 4.6.1

The functionality described in the previous sections is available in the CompHEP 4.6 line. Below we summarize the main capabilities added in the 4.6.1 release on top of version 4.6.0.

9.1. Support for LHAPDF 6 and modern parton distributions

CompHEP has been migrated from the legacy Fortran interface of LHAPDF 5 to the C++ application programming interface of LHAPDF 6 [17], which gives direct access to the modern sets of parton distribution functions. Both the built-in CompHEP structure functions (selected as "PDF:") and the external LHAPDF sets (selected as "LHA:") are available simultaneously within the same build, and for a given beam the structure function is chosen from a single list in which the desired PDF set and its member number are specified. The LHAPDF sets installed on the system are discovered automatically at startup, and this list can be refreshed manually with the "F10" key in the numerical session. For each proton and antiproton beam an LHAPDF variant of the beam is generated automatically, so that an LHAPDF structure function can be selected without additional configuration; a step-by-step "Create new beam" wizard has been added to the "Edit Beams Table" menu of the symbolic session: it guides the user through choosing the beam name, the beam type (proton or antiproton), the parton content (selected among the partons available in the current model) and the structure function — a built-in PDF set, an LHAPDF set, or none ("OFF") — after which the new beam becomes available for calculations like any predefined one.

9.2. Parallel computation over all subprocesses and combined results

When a process contains many subprocesses, the integration over all of them can now be performed in parallel. In the graphical interface, if CompHEP is configured with the "-mp N" option, the "All subprocess integr." procedure forks up to N concurrent worker processes, each integrating one subprocess in its own subdirectory. The per-subprocess cross sections are collected into a summary table that reports, for every subprocess and for the total, the cross section, its absolute and relative errors, the χ^2 of the fit and the number of matrix-element calls. If kinematic distributions are defined before the integration, the corresponding histograms are filled for each subprocess and automatically merged bin by bin into a combined distribution, which can be displayed or exported to the ROOT format. The same functionality is retained in

batch mode: the "num_batch.pl -mp N" script performs the parallel all-subprocess integration outside the graphical interface, for example on a computing cluster.

9.3. Unified event generation for all subprocesses

A new "Gen.LHEF all subproc." item has been added to the event generation menu of the graphical interface. It automatically computes the subprocess cross sections if required, distributes the requested number of events among the subprocesses proportionally to their cross sections, generates the events for each subprocess, and mixes them into a single output file using cross-section-weighted random interleaving. The resulting file is written in the Les Houches Event Format [20] with a properly merged HepML [21] header describing all subprocesses. The same event generation is available in batch mode through the "num_batch.pl" script. Since the Les Houches Event Format has become the community standard, it is now the only event format produced by CompHEP, and the older event formats have been removed.

9.4. Updated physical constants

The physical constants in all twelve model files have been updated consistently to the PDG 2024 values [22]. In particular, the top-quark mass was set to 172.5 GeV, the Higgs-boson mass to 125 GeV, and the CKM matrix elements were brought in line with the current world averages.

9.5. Validation and cross-validation of the release

The release is accompanied by a comprehensive test suite covering numerical, symbolic, gauge-invariance and unit tests. In addition, the numerical results have been cross-validated in a reproducible Docker environment against the general-purpose generators MadGraph5 [18] and WHIZARD [19], with sub-percent agreement across all validated channels, and the built-in structure functions have been verified to be consistent with the corresponding LHAPDF sets.

Finally, a number of further improvements have been made: an auto-convergence mode in batch calculations, which iterates the integration with an increasing number of calls until the statistical error drops below a requested value; a modernized export of Feynman diagrams to LaTeX based on the axodraw2 package [23], which compiles directly with pdflatex; reproducible build and distribution facilities (parallel compilation, a Docker image, a Spack recipe and signed releases); and numerous stability and portability fixes.

10. SUMMARY

The CompHEP package for modeling processes in high-energy physics has been expanded. CompHEP now supports calculations with complex form factors, enabling the calculation of decay widths and scattering cross sections in next-to-leading order according to perturbation theory. A CompHEP module has also been developed for automatic sequential calculations of arrays of decay widths and scattering cross sections as functions of multidimensional arrays of model parameter values. A module for statistical processing of results and the construction of confidence regions in the model parameter space has been added to CompHEP. A module for advanced graphical representation of results has been added for constructing of one-dimensional, two-dimensional, and three-dimensional graphs in the ROOT package format.

ACKNOWLEDGMENTS

This study was conducted within the scientific program of the National Center for Physics and Mathematics, section No. 5 “Particle Physics and Cosmology”.

[1] https://gitverse.ru/comphep, http://comphep.sinp.msu.ru

[2] E. Boos *et al.* [CompHEP], Nucl. Instrum. Meth. A **534**, 250-259 (2004) doi:10.1016/j.nima.2004.07.096 [arXiv:hep-ph/0403113 [hep-ph]].

[3] R. Brun and F.Rademakers, Nucl. Instrum. Meth. A **389**, 81-86 (1997) doi:10.1016/S0168-9002(97)00048-X

[4] E. Boos, V. Bunichev, M. Dubinin and Y. Kurihara, Phys. Rev. D **89**, 035001 (2014) doi:10.1103/PhysRevD.89.035001 [arXiv:1309.5410 [hep-ph]].

[5] E. Boos, V. Bunichev, M. Dubinin and Y. Kurihara, Phys. Lett. B **739**, 410-415 (2014) doi:10.1016/j.physletb.2014.11.015 [arXiv:1402.4143 [hep-ph]].

[6] E. Boos, V. Bunichev and M. Dubinin, PoS **QFTHEP2013**, 015 (2013) doi:10.22323/1.183.0015

[7] E. E. Boos, V. E. Bunichev, M. A. Perfilov, M. N. Smolyakov and I. P. Volobuev, Phys. Rev. D **92**, no.9, 095010 (2015) doi:10.1103/PhysRevD.92.095010 [arXiv:1505.05892 [hep-ph]].

[8] E. E. Boos, V. E. Bunichev and I. P. Volobuev, J. Exp. Theor. Phys. **124**, no.5, 722-730 (2017) doi:10.1134/S106377611704001X

[9] E. Boos and V. Bunichev, Phys. Rev. D **101**, no.5, 055012 (2020) doi:10.1103/PhysRevD.101.055012 [arXiv:1910.00710 [hep-ph]].

[10] E. E. Boos, V. E. Bunichev and S. S. Trykov, Phys. Rev. D **107**, no.7, 075021 (2023) doi:10.1103/PhysRevD.107.075021 [arXiv:2205.07364 [hep-ph]].

[11] E. E. Boos, V. E. Bunichev, S. I. Keizerov and S. S. Trykov, Phys. Part. Nucl. **56**, no.3, 750-755 (2025) doi:10.1134/S106377962470223X

[12] J. R. Espinosa, C. Grojean, M. Muhlleitner and M. Trott, JHEP **05**, 097 (2012) doi:10.1007/JHEP05(2012)097 [arXiv:1202.3697 [hep-ph]].

[13] J. R. Espinosa, C. Grojean, M. Muhlleitner and M. Trott, JHEP **12**, 045 (2012) doi:10.1007/JHEP12(2012)045 [arXiv:1207.1717 [hep-ph]].

[14] CMS Physics Analysis Summaries, Combination of standard model Higgs boson searches and measurements of the properties of the new boson with a mass near 125 GeV, CMS-PAS-HIG-13-005, see https://cds.cern.ch/record/1542387.

[15] A. Djouadi, Phys. Rept. **457**, 1-216 (2008) doi:10.1016/j.physrep.2007.10.004 [arXiv:hep-ph/0503172 [hep-ph]].

[16] J. R. Espinosa, M. Muhlleitner, C. Grojean and M. Trott, JHEP **09**, 126 (2012) doi:10.1007/JHEP09(2012)126 [arXiv:1205.6790 [hep-ph]].

[17] A. Buckley, J. Ferrando, S. Lloyd, K. Nordström, B. Page, M. Rüfenacht, M. Schönherr and G. Watt, Eur. Phys. J. C **75**, 132 (2015) doi:10.1140/epjc/s10052-015-3318-8 [arXiv:1412.7420 [hep-ph]].

[18] J. Alwall, R. Frederix, S. Frixione, V. Hirschi, F. Maltoni, O. Mattelaer, H. S. Shao, T. Stelzer, P. Torrielli and M. Zaro, JHEP **07**, 079 (2014) doi:10.1007/JHEP07(2014)079 [arXiv:1405.0301 [hep-ph]].

[19] W. Kilian, T. Ohl and J. Reuter, Eur. Phys. J. C **71**, 1742 (2011) doi:10.1140/epjc/s10052-011-1742-y [arXiv:0708.4233 [hep-ph]].

[20] J. Alwall, A. Ballestrero, P. Bartalini, S. Belov, E. Boos, A. Buckley, J. M. Butterworth, L. Dudko, S. Frixione and L. Garren, *et al.* Comput. Phys. Commun. **176**, 300-304 (2007) doi:10.1016/j.cpc.2006.11.010 [arXiv:hep-ph/0609017 [hep-ph]].

[21] S. Belov, L. Dudko, D. Kekelidze and A. Sherstnev, Comput. Phys. Commun. **181**, 1758-1768 (2010) doi:10.1016/j.cpc.2010.06.026 [arXiv:1001.2576 [hep-ph]].

[22] S. Navas *et al.* [Particle Data Group], Phys. Rev. D **110**, no.3, 030001 (2024) doi:10.1103/PhysRevD.110.030001

[23] J. C. Collins and J. A. M. Vermaseren, [arXiv:1606.01177 [cs.OH]].