CalcHEP - calculator for High Energy Physics
micrOMEAGs - calculation of DM properties

CalcHEP team : A. Belyaev, N. Christensen, A.Pukhov
micrOmegas: G.Belanger, F. Boudjma, A.Semenov, A.Pukhov

http://theory.sinp.msu.ru/~pukhov/calchep.html
http://lapth.cnrs.fr/micromegas

Presented by **Alexander Pukhov**,
Skobeltsyn Institute of Nuclear Physics, Moscow.

Beijing, July 2016

Main features

- **Language** **C**
- **Symbolic calculation of squared diagrams**. CalcHEP uses built-in symbolic calculator. Result of calculation can be presented in formats of Reduce, Mathematica, C.
- **Tree level** calculations in Unitary and Feynman gauges. The last one is free of diagram cancellation at high energies
- **Two modes of calculation**

  a) GUI with menus and help facilities. Good for beginners.

  b) batch mode, subprocess cycle

- **Model files**

  a) can be generated by LanHEP, feynRules

  b) effective loop induced vertices for H decay

  c) spin 3/2 and 2

  d) color sextets and 333 vertex.

- **Les Houches Interface**
  a) SLHA interface with spectrum generators (SuSpect, SoftSUSY)
  b) LHAPDF5 & LHAPDF6 link and export.
  c) events (LEF)

- **Run-time generation** and dynamic linking of new codes. It allows 'on fly' width calculation **i**ncluding 1->3, 1->4 channels and virtual W/Z contribution.

- **Paralleling** for symbolic and numerical calculations.
  Batch mode - PC farms
  GUI - fork and threads.

- **Generation of codes** of matrix element for other packages (micrOMEGAs)

How to start to work

Compilation: needs gcc and X11-devel (with h-files)

```
cd calchep_3.6.28
make
```

Staring GUI session

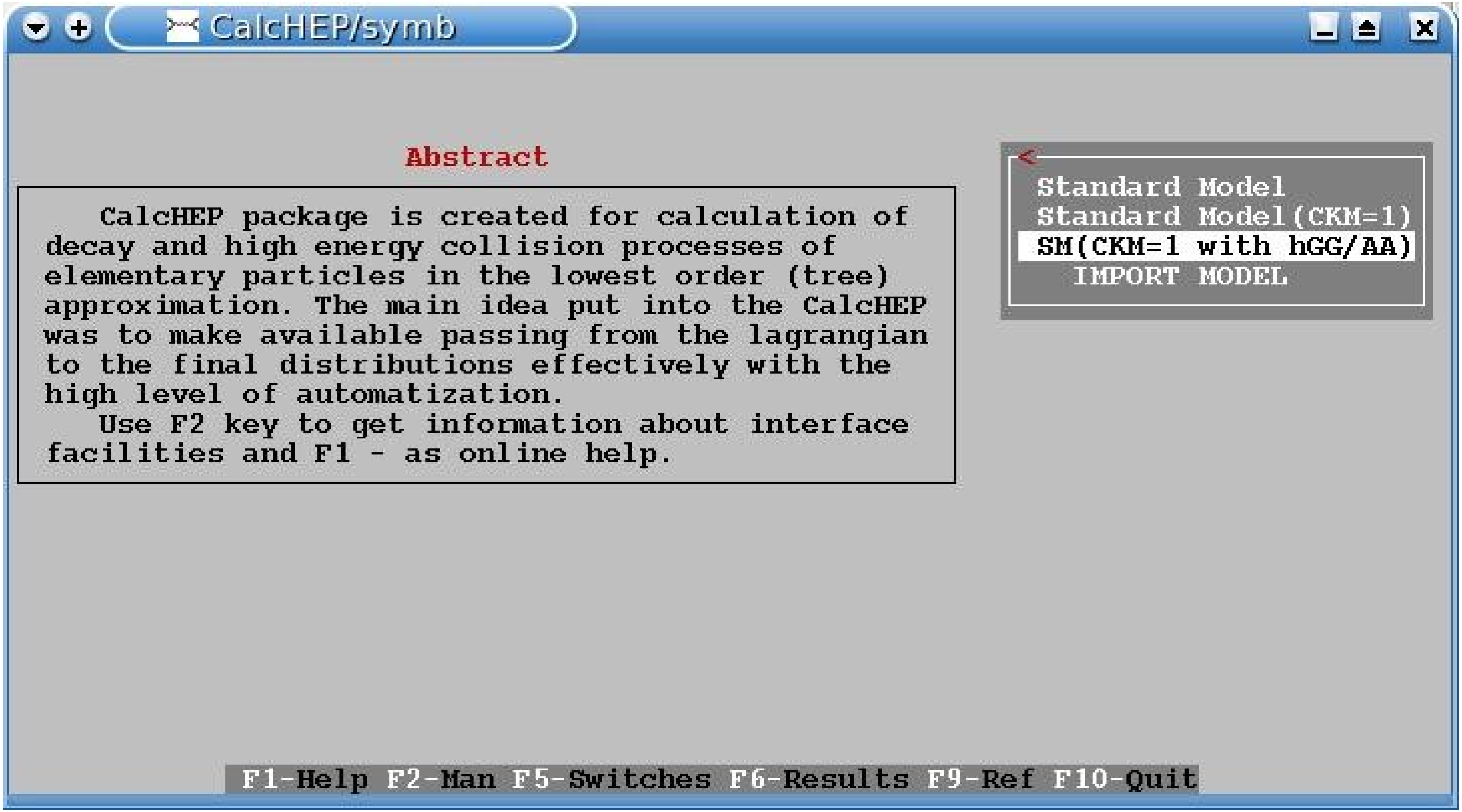
```
cd work
./calchep
```

Starting batch session

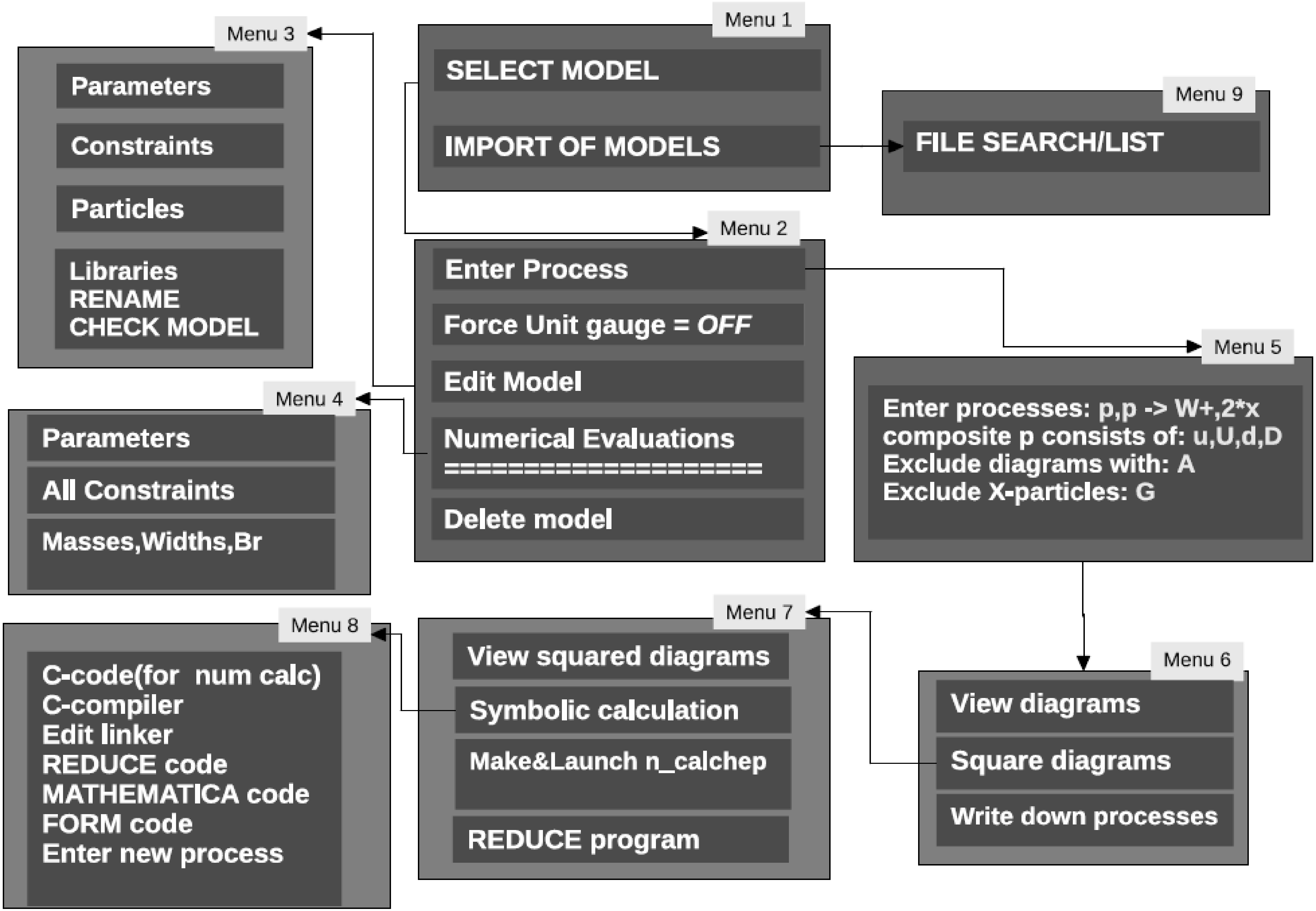
```
cd work

./calchep_batch  batch_file
```

180 pages manual is disposed on CalcHEP web site.

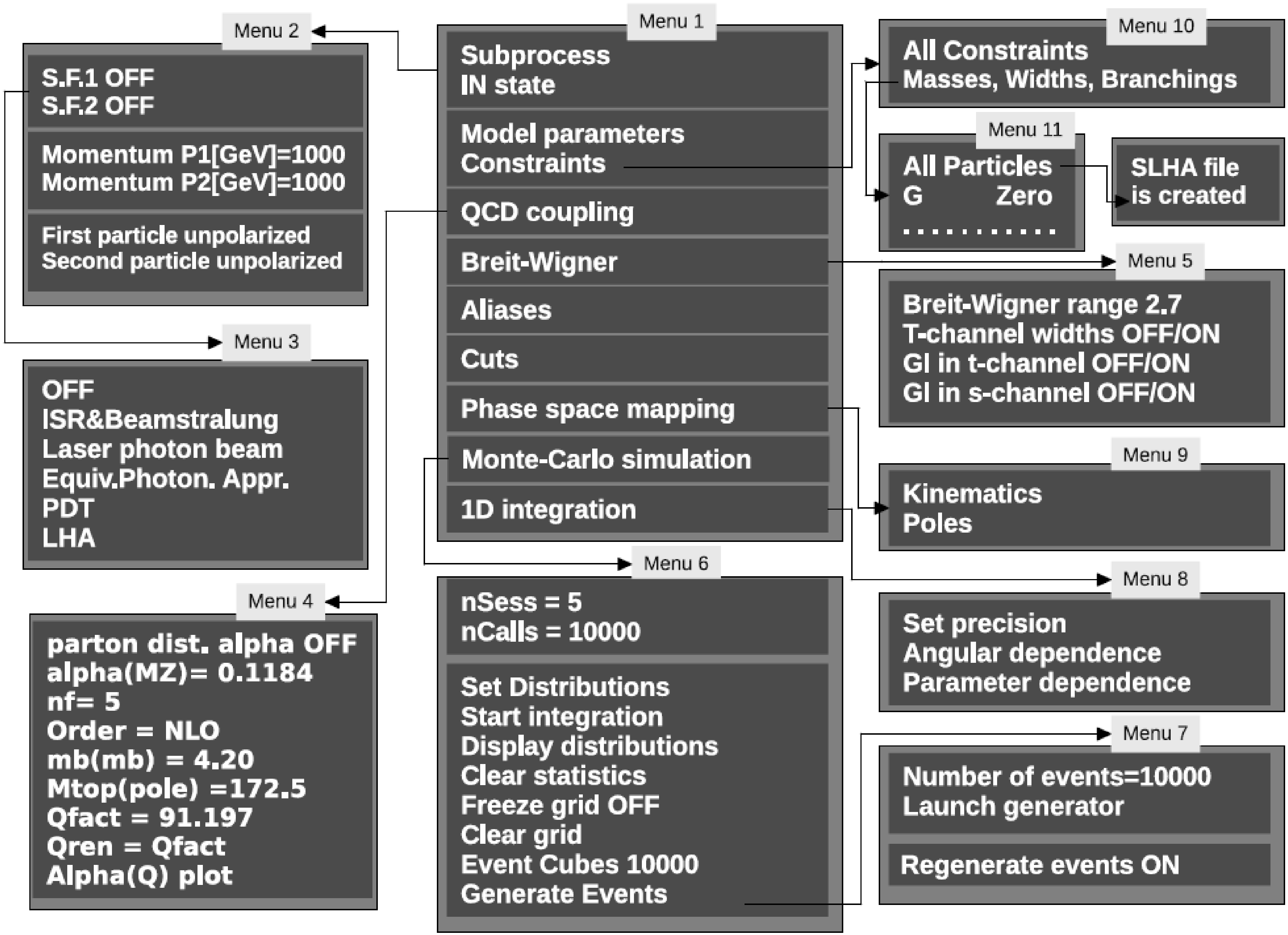
Graphic interface



CalcHEP menu structure: symbolic part



Menu structure of the numerical part



Automatic width calculation

- Code for particle widths are generated and compiled in run time and linked dynamically.
- 1->2, 1->3, 1->4 decays are tested subsequently until open channels are detected.
- To take into account radiation corrections of Higgs decay effective quark masses are used:

  MqEff(Q) ~ MqRun(Q/3)   where Q= Higgs Mass

- There is an option to calculate widths for processes with virtual W/Z. CalcHEP a) calculates width of process where W is replaced of e^+,\nu b) takes into account branching; c) takes into account relation between 1->3 and 1->4 widths.
- h->G,G and G->AA can be treated via effective vertexes.
- Quite realistic description of Higgs decay can be obtained by this way. SM and MSSM widths and branching where compared with HDECAY

H->GG && H->AA for SM, MSSM, ...

Feynman rules

```
G |G |h |  |-4*LGGh*Rqcdh     |(p1.p2*m1.m2-p1.m2*p2.m1)
G |G |G |h |-4*LGGh*GG*Rqcdh  | ….
A |A |h |  |-4*LAAh           |(p1.p2*m1.m2-p1.m2*p2.m1)
```

Constraints

```
aQCDh |alphaQCD(Mh)/pi
Rqcdh |sqrt(1+149/12*aQCDh+68.6482*aQCDh^2-212.447*aQCDh^3)
LGGh  |-cabs(hGGeven(Mh, aQCDh,3,% spin/2, color, mass, coupling
                                      1,      3,     Mtp,  1/VEV,
                                      1,      3,     Mbp,  1/VEV,
                                      1,      3,     Mcp,  1/VEV))
Quq   | 4/9
Qdq   | 1/9
LAAh  |-cabs(hAAeven(Mh,aQCDh,2, % spin/2, color, mass, coupling
                                     2,        1,     MW,   2/VEV,
                                     1,        1,     Ml,   1/VEV)
         +Quq*hAAeven(Mh,aQCDh,2,    1,       3,     Mtp, 1/VEV,
                                     1,       3,     Mcp, 1/VEV)
         +Qdq*hAAeven(Mh,aQCDh,1,    1,       3,     Mbp, 1/VEV))
```

QCD NLO corrections are included.

LanHEP for H->GG & H->AA

At level of model generation LanHEP has a function CoefVrt

```
_h=[h,H], _p=[c,b,t,l] in parameter a_hF__p
  =CoefVrt([anti(_p),_p, _h]) /(mass _p).
```

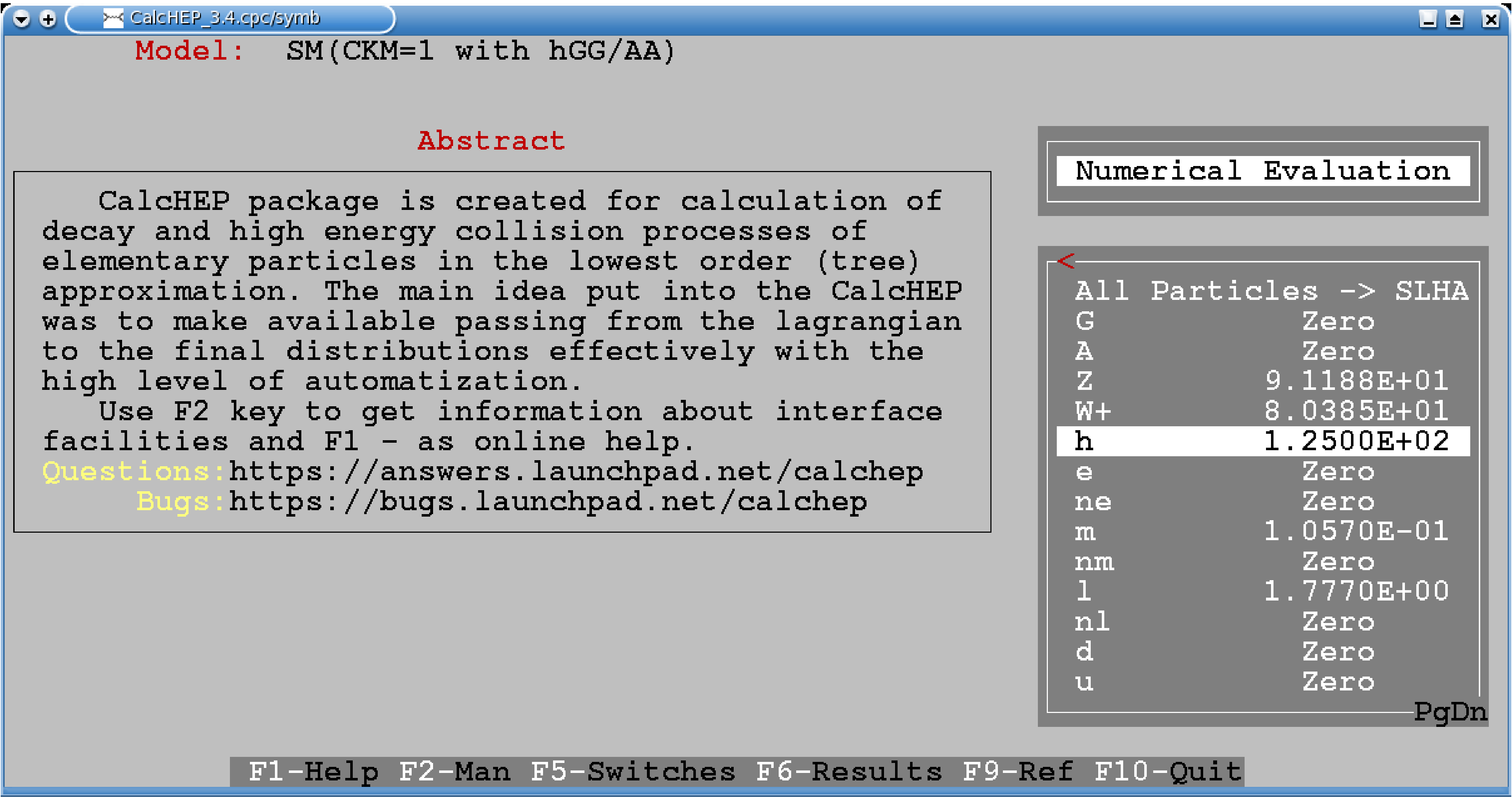
which selects a coupling at vertex. When

```
LGGh =-cabs( hGGeven(M_h,aQCD_h,3,
1,3,Mbp,a_hF_b, 1,3,Mcp,a_hF_c, 1,3,Mtp,a_hF_t)
           ).
```

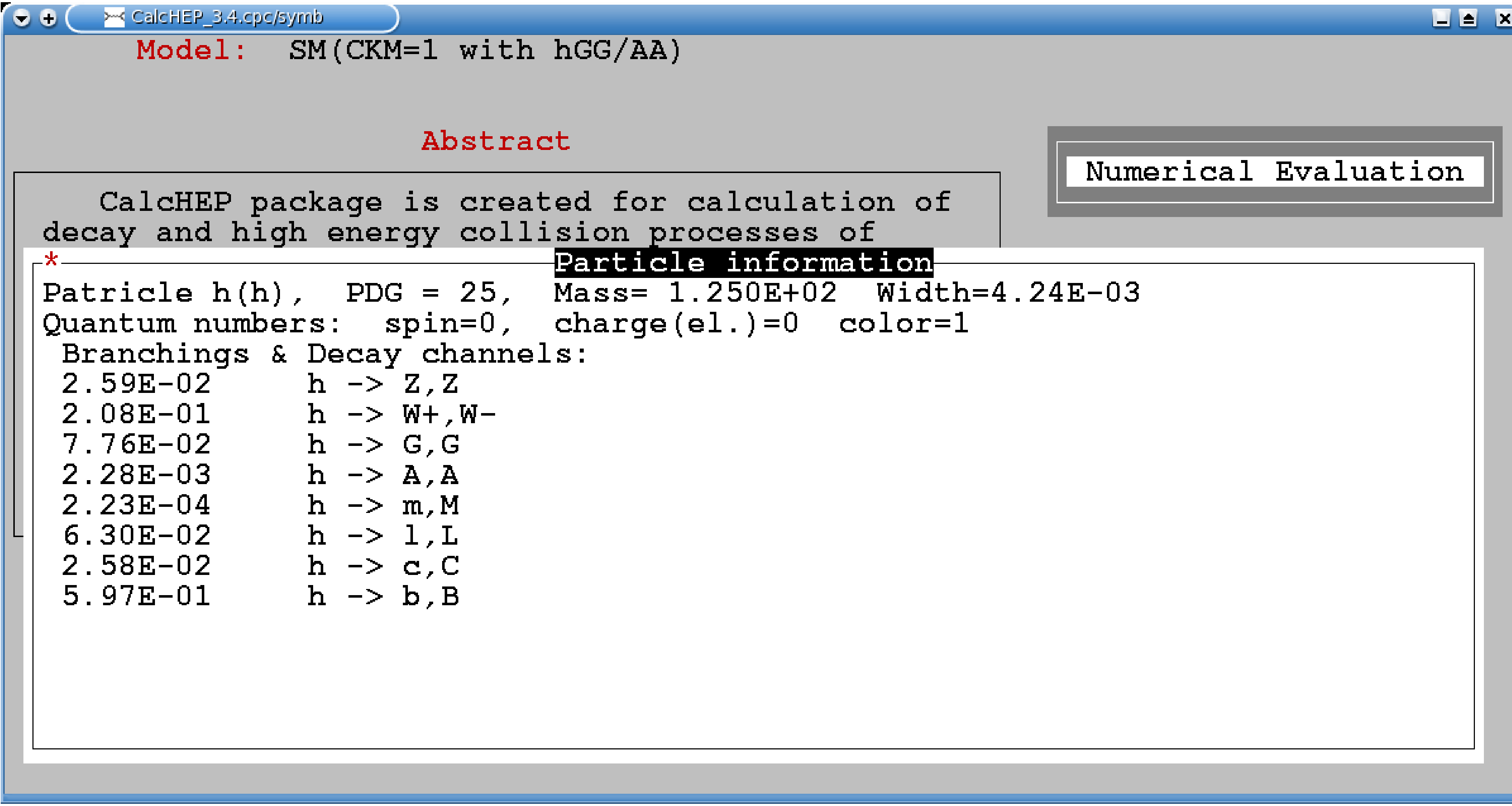
For odd Higgs one can use

```
CoefVrt([anti(_p),_p, _h],[gamma5,im])
```

One create *decaySLHA.txt* file which contains quantum numbers, masses, widths, and branching for all particles or to test properties of each particle separately.



Higgs decay


CalcHEP_3.4.cpc/symb
Model: SM(CKM=1 with hGG/AA)
Abstract
CalcHEP package is created for calculation of
decay and high energy collision processes of
Numerical Evaluation
Particle information
Patricle h(h), PDG = 25, Mass= 1.250E+02 Width=4.24E-03
Quantum numbers: spin=0, charge(el.)=0 color=1
Branchings & Decay channels:
2.59E-02 h -> Z,Z
2.08E-01 h -> W+,W-
7.76E-02 h -> G,G
2.28E-03 h -> A,A
2.23E-04 h -> m,M
6.30E-02 h -> l,L
2.58E-02 h -> c,C
5.97E-01 h -> b,B

Model implementation: SLHAplus

SLHAplus is a library of auxiliary functions used for for model realization. Now it includes

a) **functions for SLHA interface (file exchange):**

slhaRead(fileName, mode)

slhaVal(BlockName, Scale, N_key_symbols, keys ...)

Example:

```
Block MASS
 25  125
```

Mh=slhaVal("MASS", 0.,1,25)

```
BLOCK STAUMIX  # Stau Mixing Matrix
  1  1     4.86991070E-02   # cos(theta_tau)
  1  2     9.98813495E-01   # sin(theta_tau)
  2  1    -9.98813495E-01   # -sin(theta_tau)
```

Zl12=slhaVal("STAUMIX", QSUSY, 2, 1, 2)

If mode&4==0 SLHA width is used instead of automatic width calculation

SLHAplus: other functions

b) **functions for diagonalizing of** mass matrices : i) real symmetric ii) hermitian; iii) complex; iv) complex symmetric. based of Jacobi algorithm (arXiv:1008.0181). Special

```
NeDiag |rDiagonal(4, MG1, zero, -MZ*SW*cb, MZ*SW*sb, MG2, MZ*CW*cb,
                                     -MZ*CW*sb, zero, -mu, zero)
MNE1   |MassArray(NeDiag, 1)
MNE2   |MassArray(NeDiag, 2)
Zn11   |MixMatrix(NeDiag, 1, 1)
Zn12   |MixMatrix(NeDiag, 1, 2)
```

c) Functions to construct effective hGG and hAA vertices at NLO level HGGEven/Odd and hAAEven/Odd presented above.

d) **QCD functions** for running alpha QCD, masses, effective Yukawa couplings.

Effective vertices

CalcHEP allows only 3 and 4 legs vertices with limited types of color structures

$$3\bar{3},\quad 3\bar{3}8,\quad 88,\quad 888,\quad 333,\quad \bar{3}\bar{3}\bar{3},\quad 6\bar{6},\quad 6\bar{6}8,\quad 33\bar{6},\quad \bar{3}\bar{3}6$$

Auxillary field with point-like propagator can be used to construct multi-leg vertices and vertices with other color structures. Usually LanHEP solves this problem. Suppose we need $L = k\bar{t}\sigma_{\mu\nu}\hat{\lambda}^a t \cdot F_a^{\mu\nu}$ Then in terms of LanHEP

```
lterm  k*i/2*T*gamma^mu*gamma^nu*lambda^a*t*
(deriv^mu*G^nu^a-deriv^nu*G^mu^a+i*GG*f_SU3^a^b^c*G^mu^b*G^nu^c).
```

```
             Interaction
P1|p2|p3    |p4|totFcat| Lorentz part
T |t |G     |  |k/2    |G(p3)*G(m3)-G(m3)*G(p3)
G |G |~01.t|  |1/2    |m1.m3*m2.M3-m1.M3*m2.m3
T |t |~00.t|  |k*GG   |G(M3)*G(m3)
```

```
               Particles
  P | aP| PDG |spin2|mass|width|color|aux|  * - point-like prop
 ~00|~01|     |2    |Maux|0    |8    |!* |  ! - self conjugation
```

Available [B]SM models

SM, IDM – included in CalCHEP distribution.
MSSM,NMSSM,CPVMSSM – calchep web page
and micrOMEGAs
LHM,UMSSM, Z3M, Z4M - micrOMEGAs

Structure functions

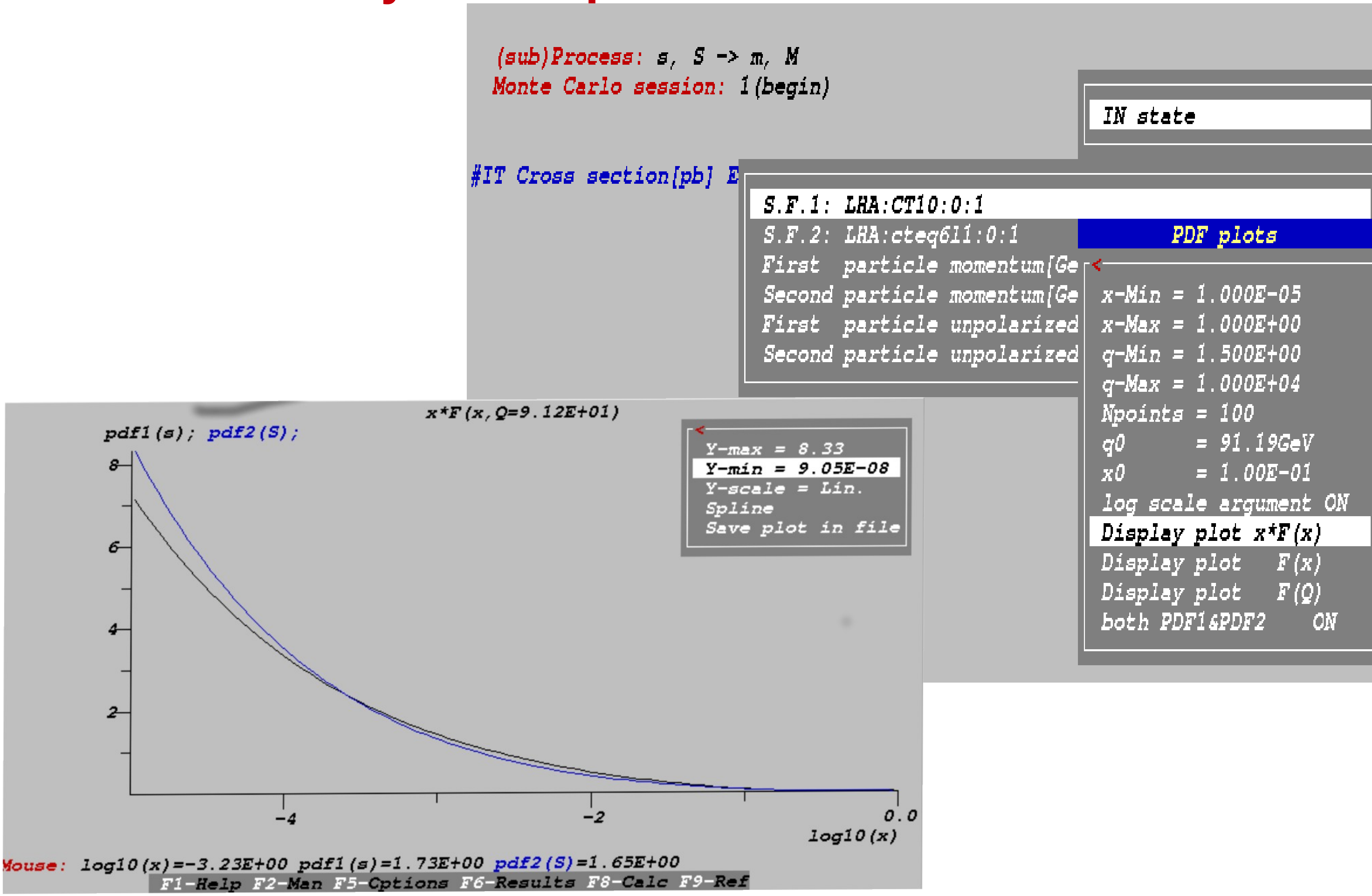
- **CalcHEP contains several build-in structure functions for** e^+e^-
  ISR+Beamstrahlung
  Equivalent photo approximation
  Laser photons

- **Link with LHAPDF5 and LHAPDF6**
  export LHAPDFPATH=*path_to_LHAPDF_library*
  One should see
  ls ***$LHAPDFPATH/libLHAPDF.so***
  If LHAPDFPATH is not defined then dummy library is linked.

- **One can export LHAPDF6 tables into CalcHEP**
  CalcHEP/bin/lhapdf2odt *LHAPDF6***/share/LHAPDF/CT10 0**
  creates file CN10:0.pdt for CalcHEP/pdTables

- **By default CalcHEP/pdTables contains**
  CT10.pdt NNPDF23_lo_as_0130.LHgrid.pdt
  cteq6l1.pdt MRST2004qed_proton.pdt
  NNPDF23_lo_as_0130_qed.pdt

There is a way to compare different structure functions.

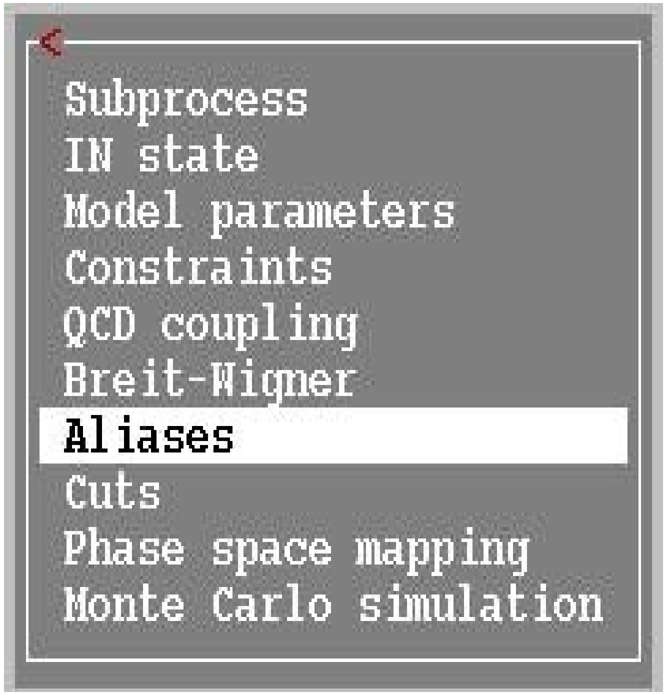


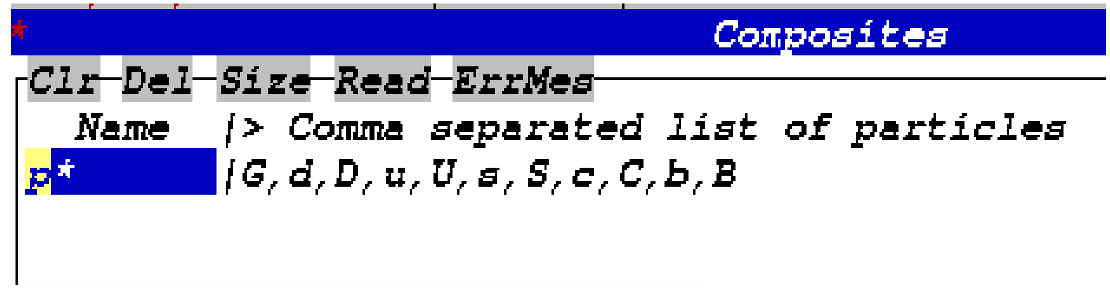
List of built-in functions for QCD scale cuts and distributions.

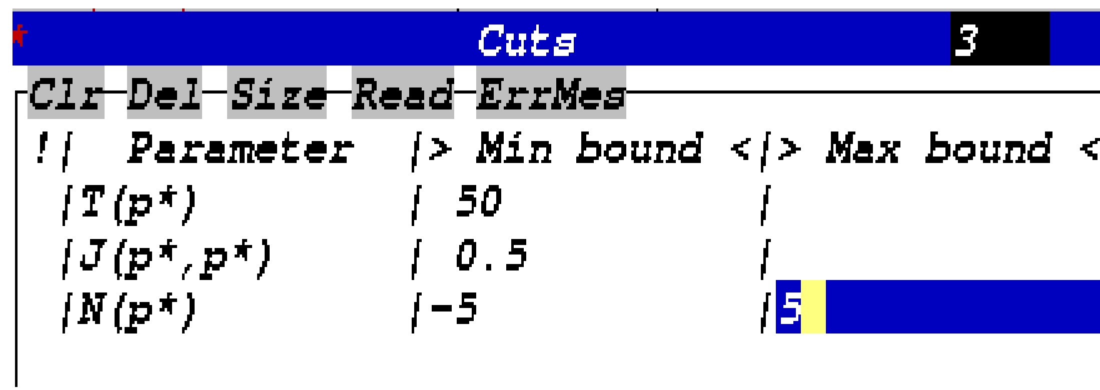
A - Angle in degree units
C - Cosine of angle
D - Jet separation min(pT1^2,pT2^2)*(cosh(d_Rapidity)-cos(d_AzimuthAngle))
J - Jet cone angle
E - Energy of the particle set
M - Mass of the particle set
P - Cosine of the angle between the first particle in the list and the direction of boosting of the particle set into the rest frame of the particles set
T - Transverse momentum (P_t) of the particle set
Y - Rapidity of the particle set
N - Pseudo-rapidity of the particle set
Z - Transverse energy

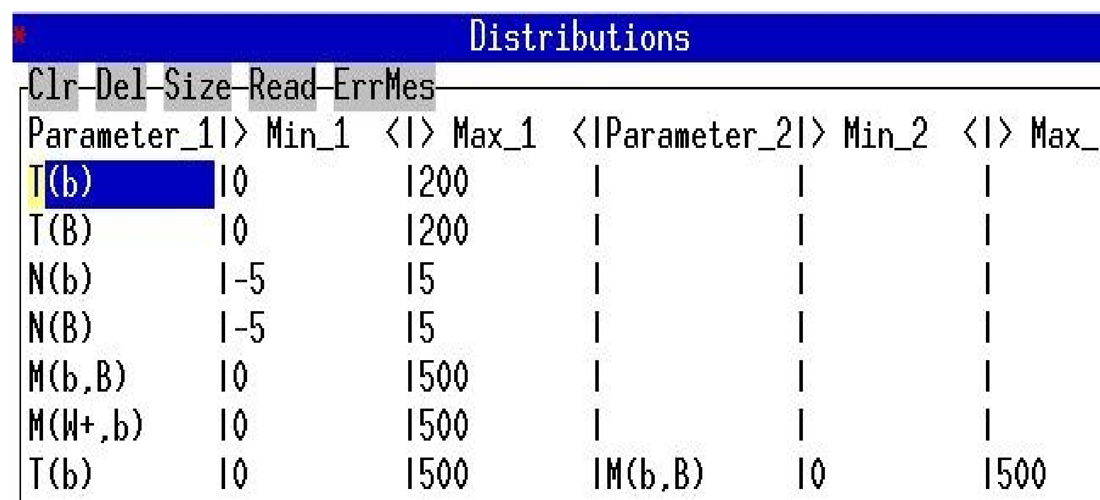
U*user* *user defined function via **usrfun(**“user”**)***

Aliases, Cuts, Distributions









In case of identical particles constructing distributions **F** CalcHEP normally sums all possibilities. But **F^(F`)** and **F_** evaluate **max** and **min** values.

If cut/distribution can not be realized, then it is ignored.

Definition of QCD scale

Suppose one would like to calculate cross section

p,p-> Jet, dm,dm

for scale

Q= 0.5*(PT(JeT) + ME).

It can be done by one definition for all subprocesses

Q= 0.5*(T(“p*”)+ Z(“dm”,”dm”))

Integration and generation of events

Slightly improved Vegas with thread paralleling is used for integration and events generation

- First Vegas cycle is used for grid adaptation.
- Second Vegas cycle is used to get profile of integrand – find maxima of for subcubes and estimate efficiency of events generation. Filling of histograms.
- Events generation. Events have weights ~ cross section of process. If Vegas finds a point where integrand exceeds maximum detected on step 2 then weight increased or more events are generated in the given sub-cube.

Files generated in Monte Carlo session

Monte Carlo session generates distr_N file which contains distributions. N is session number which increases automatically when user changes parameters.

- $CALCHEP/bin/show_distr distr_N

allows to display distribution. CalcHEP creates plot which can be saved in PAW, GNUPLOT, and ROOT formats.

- $CALCHEP/bin/sum_distr distr_N1 distr_N2 … > distr_sum

crates sum of distributions obtained in different sessions.

Monte Carlo session generates event_N files. To make a summary LHE file (subprocesses summation and decays implementation) one can use

- $CALCHEP/bin/event_mixer Luminosity nEvents directories .....

Resulting file is events_mixer.lhe . File *decaySLHA.txt* (if it exists) is added to define q-numbers, widths, and branching of BMS particles.

Simple batch scripts.

CalcHEP/bin directory contains several scripts which performs cycle calculations in batch mode

name_cycle pcm_cycle subproc_cycle par_scan par_scan_sum

They have to be launched from *work/results* directory where

n_calchep compiled in symbolic session is disposed. For example,

../bin/subrocess_sycle Nevents

performs cycle over all subrocesses and generates LHE file with information about decays of BSM particles. These scripts really See explanation in CALCHEP/bin/README or in manual.

The **calchep_batch** program

- All steps of calculation can be done in batch mode by one command

  ./calchep_batch <input file>

- Control of calculation can be done via WEB browser. Before calculation the program writes on the screen name of html file. For example:

  0pen the following link in your browser:
  file:///home/pukhov/CALCHEP/calchep_3.6.28/work/html/index.html

- One can use parallel calculation of batch task on PS farm.

Example of input file for calchep_batch command

```
Model:         Standard Model(CKM=1)
Model changed: False
Gauge:         Feynman

Process:   p,p->W,b,B
Decay:     W->le,n
Composite: p=u,U,d,D,s,S,c,C,b,B,G
Composite: W=W+,W-
Composite: le=e,E,m,M
Composite: n=ne,Ne,nm,Nm
Composite: jet=u,U,d,D,s,S,c,C,b,B,G

pdf1:      cteq6l 1(proton)
pdf2:      cteq6l 1(proton)
p1:        4000
p2:        4000

Run parameter: Mh
Run begin:     120
Run step size: 5
Run n steps:   3

alpha Q :          M45

Cut parameter:    M(b,B)
Cut invert:       False
Cut min:          100

Dist parameter:    M(b,B)
Dist min:          100
Dist max:          200
Dist n bins:       100
Dist title:        p,p->W,b,B
Dist x-title:      M(b,B) (GeV)

Dist parameter:    M(W,jet)
Dist min:          100
Dist max:          200
Dist n bins:       100
Dist title:        p,p->W,b,B
Dist x-title:      M(W,jet) (GeV)

Number of events (per run step): 10000
Filename:                        pp_Wbb_enbb

nSess_1:   5
nCalls_1:  100000
nSess_2:   5
nCalls_2:  100000

#Parallelization method:     local
Parallelization method:     pbs

Max number of nodes:   2
Max number of processes per node: 2
```

http://theory.npi.msu.su/~pukhov/calchep.html

CalcHEP - a package for calculation of Feynman diagrams and integration over multi-particle phase space.

Authors - Alexander Pukhov, Alexander Belyaev, Neil Christensen

The main idea of CalcHEP is to enable one to go directly from the Lagrangian to the cross sections and distributions effectively, with a high level of automation. The package can be compiled on any Unix platform.

General information
Main features , Acknowledgments News&Bugs Publications&Lectures Contributions

---

Manual
calchep_man_3.3.6.pdf (manual for version 3.3.6, July 19, 2012)
HEP computer tools (Lecture by Alexander Belyaev)
See also: Dan Green, High Pt physics at hadron colliders (Cambrige University Press)

Code download.
Licence Installation Current version 3.6.27(20.07.2016) Archive

---

Models:
MSSM_10.14(15.10.2014) NMSSM_8.15(25.08.2015) CPVMSSM_10.14(16.10.2014) SUSY models By A.Semenov LeptoQuarks 5DSM 6DSM
Model database HEPMDB

Related packages on Web:
Packages for model generation: LanHEP FeynRules SARAH
RGE and spectrum calculation: SuSpect Isajet SoftSUSY SPheno CPsuperH NMSSMTools
Particle widths in MSSM: SUSY-HIT HDECAY
Parton showers: PYTHIA

Contacts

Email: calchep@googlegroups.com *Launchpad service:* *Ask a question* *File a bug*

micrOMEGAs

CalcHEP has an option to generate code of matrix elements for other programs.

micrOMEGAs program is created for calculation of Dark Matter observables. It needs matrix elements to calculate cross sections of DM annihilation and its scattering on atomic nuclei. It uses CalcHEP routine

```
numout*cc=newProcess(char*name);
```

For example:

```
newProcess("e,E->m,M");//
```
$e^-, e^+ \to \mu^-, \mu^+$

All matrix elements are generated only one time and stored on disk as shared libraries.

- To calculate matrix element **micrOMEGAs** uses
  cc->interface->sqme(nSubproc,GG,momenta,&errode)

- To get information about particles included in reaction
  cc->interface->pinf(nSubproc,nParticle,&Mass,&PDG)

- To get information about quantum number of particles
  cc->interface->pinfAux(nSubproc,nParticle,
  &spin2,&color,&neutral);
- To change numerical numerical value of variables
  assignVal(name,newValue)

- To calculate constraints
  calcMainFunc()

- To check mass and widths:
  pMass(pName), **pWidth**(pName, decayList)

micrOMEGAs assumes existence of some discrete symmetry which leads to MD stability, for instance Z_2**.**
Names of particles with non-zero charge have to be started with **"~"** to help micrOMEGAs to recognize DM sector.

- **sortOddParicles**() calculates constrains, mass spectrum and detects the lightest odd particle, **CDM1**, cold Dark Matter.

- **micrOMEGAs** calculates **DM** r**elic density**.

  $\Omega h^2 =$ darkOmega(&Xf,Besp,fast)

  $X_f = M_{cdm}/T_{freeze-out}$ Beps controls co-annihilation.
- **Direct Detection signal**

  nucleonAmplitudes(CMD, AsiP,AsdP,AsiN,AsdN)
- Signals of **indirect detection**: Spectra of DM annihilation in galactic halo

  calcSpectrum(mode,SpA,SpE,SpP,SpNe,SpNm,SpNl,&err)

  propagation routines also included in micrOMEGAs.
- **Neutrino telescope** signal: DM annihilation in the center of Sun/Earth.

  neutrinoFlux(DMvelocityDist, forSun[Earth],nuFlux,nuBarFlux)

Astro&Colliders- 1606.03834

Astrophysical and collider BSM signal search.
Mainly based on interface with external packages via SLHA

- *HiggsBounds/HiggsSignals arXiv 1305.1933, 1311.0055,*
- *Lilith arXiv 1502.04138*
  micrOMEGAs calculates widths, branching for Higgses and their couplings with SM, passes them to Lilith/Higgs... , gets exclusion level
- *masslimits() checks that at least one BSM particle has a mass larger than LEP limits.*
- *Zinvisible() checks that width(Z->invizible) <0.5GeV*
- *LspNlsp_LEP() the same in case of small lsp – Nlsp mass difference.*
- *SmodelS (arXiv 1412.1745) test LHC constraints on Dark Sector particles for models with Z_2 internal symmetry (squarks, sleptons, charginos). micrOMEGAs calculates all p,p → Dark Sector (SUSY) reactions, passes them to SMODELS with SLAH decay tbales and gets exclusion level.*