Tools for the SM and the New Physics
Corfu September 10-13

Hands on dark matter codes
MicrOMEGAs

Alexander Pukhov
Skobeltsyn Institute of Nuclear Physics.
Moscow, Russia.

MicrOMEGAs is a package for calculation of DM properties for generic model of particle interaction.

Main authors: G.Belanger, F.Boudjema (LapTH), A.Pukhov(SINP), A.Semenov(JINR)

Main publictions and steps of development:

1.MicrOMEGAs: A Program for calculating the **relic density in the MSSM** *hep-ph/0112278*
2.MicrOMEGAs: Version 1.3 *hep-ph/0405253*
3.MicrOMEGAs 2.0: A Program to calculate the **relic density of dark matter in a generic model** *hep-ph/0607059*
4.Dark matter **direct detection rate** in a generic model with micrOMEGAs 2.2 *arXiv:0803.2360*
5.**Indirect search** for dark matter with micrOMEGAs2.4 *arXiv:1004.1092*
With P.Brun, S.Rosier-Lees, P.Salati
6. micrOMEGAs_3: A program for calculating dark matter observables arXiv:*1305.0237*
7. Limits on dark matter proton scattering from **neutrino telescopes** using micrOMEGAs *arXiv:1507.07987*
8.**Collider limits** on new physics within micrOMEGAs4.3 *arXiv:1606.03834*
With D.Barducci, J.Bernon, S.Kraml,U.Laa

General characteristics

Operation system Linux or Darwin. In principle it should work on any UNIX platform.

Language C (C99).
Own code size 14Mb
Included packages :

CalcHEP for matrix element generation
LanHEP for model generation
LoopTools for Dm,Dm → gamma, gamma(Z)
SuSpect, NMSSMTools, CpsuperH spectrum calculation for specific models
Lilith for Higgs physics

All together 76Mb
Downloaded in runtime:

HiggsBounds/HiggsSignals for Higgs physics
SMODELS - for collider analyses

Needed compilers: gcc, gfortran
Language for user main code: C/C++/Fortran

Installation of micrOMEGAs package

micrOMEGAS site

http://lapth.in2p3.fr/micromegas

Click *Download and Install* *(left -top part of the screen)*

And then DOWNLOAD (right-top part of the screen)

The name of received file should be

micromegas_4.3.5.tgz

Unpack it by **tar -xvzf micromegas_4.2.5.tgz**

It should create directory **micromegas_4.2.5/** which occupies about 80 Mb of disk space. You will need more disk space after compilation of specific models and generation of matrix elements .

In case of problems and questions

email: micro.omegas@lapp.in2p3.fr

File structure of micrOMEGAs package.

Makefile	
CalcHEP_src/	generator of matrix elements
sources/	micrOMEGAs own codes
man/	
manual_4.2.tex, manual_4.2.pdf	description of micrOMEGAs routines

Packages/

SuSpect_2.41 NMSSMTools_4.7.1 CpsuperH2.3,
LoopTools-2.1 LanHEP

model directories:

MSSM/	
NMSSM/	Next-to-Minimal SuSy Model
CPVMSSM/	MSSM with complex parameters
UMSSM/	MSSM + U(1) gauge field
IDM/	Inert dublet model
LHM/	Little Higgs Model
Z3IDM/	Z^3 model
Z4IDMS/	Z^4 model

To compile micrOMEGAS use 'gmake' or 'make'

To clean use [g]make clean

Structure of MODEL directory

Makefile	
main.c main.F	files with main program for given model
lib/	directory for specific model routines
Makefile	
*.c, **.F,**cpp	source codes
alib.a	compiled library
work/	CalcHEP working directory intended for matrix element generation
models/	model specification
vars1.mdl **func1.mdl** **prtcls1.mdl** **lgrng1.mdl** **extlib1.mdl**	
so_generated/	directory to store generated matrix elements
calchep/	for interactive CalcHEP sessions

Makefile supports compilation of C,Fortran and C++ user codes

[g]make main=XXX.c => executable XXX
[g]make main=YYY.F => executable YYY
[g]make main=ZZZ.cpp => executable ZZZ

[g]make is equivalent to **[g]make main=main.c**

Model Files: Inert Doublet Model

Inert Doublet model contains two SU(2)*U(1) doublets

$$H_1 = \begin{pmatrix} 0 \\ <v> + h/\sqrt{2} \end{pmatrix} \;, \quad H_2 = \begin{pmatrix} \tilde{H}+ \\ (\tilde{X} + i \cdot \tilde{H}3)/\sqrt{2} \end{pmatrix}$$

The Lagrangian contains only *even* powers of H_2 doublet

$$L = (SM\ terms) + D^\mu H_2^* D_\mu H_2$$

$$-\mu^2 H_2^2 - \lambda_2 H_2^4 - \lambda_3 H_1^2 H_2^2 - \lambda_4 |H_1^* H_2|^2 - \lambda_5 Re[(H_1^* H_2)^2]$$

Because of symmetry $H_2 \to -H_2$ the lightest of $\tilde{H}^+, \tilde{X}, \tilde{H}3$ is stable

Parameters $\mu, \lambda_3, \lambda_4$ can be expressed in terms of masses

New couplings are λ_2 , $\lambda_L = \lambda_3 + \lambda_4 + \lambda_5$

See details **arXiv:1106.1719**

Model Files: Free parameters of the model.

Free model parameters are presented in *MODEL/**work/models/vars1.mdl***
For example:

```
Inert Doublet Model
 Variables
 Name | Value     |>  Comment                       <|
EE    |0.31333    |Electromagnetic coupling constant
SW    |0.474      |sin of the Weinberg angle
MZ    |91.187     |Mass of Z
MHX   |111        |Mass of Inert Doublet Higgs
MH3   |222        |Mass of CP-odd Higgs
MHC   |333        |Mass of charged Higgs
LaL   |0.01       |Coupling in Inert Sector
................................
```

Model Files: Constrained parameter of the model.

Constrained parameters are stored in file
MODEL/**work/models/func1.mdl**

```
Inert Doublet
 Constraints
 Name  |> Expression
CW     |sqrt(1-SW^2)
MW     |MZ*CW
Mb     |MbEff(Q)
Mc     |McEff(Q)
mu2    |MHX^2-laL*(2*MW/EE*SW)^2
la3    |2*(MHC^2-mu2)/(2*MW/EE*SW)^2
la5    |(MHX^2-MH3^2)/(2*MW/EE*SW)^2
```

Model Files: Particles of the model

List fo particles presented in file MODEL/work/models/prtcls1.mdl

```
Full   Name     | P | aP|  number  |spin2|mass|width|color|aux|> LaTeX(A)
photon          |A  |A  |22        |2    |0   |0    |1    |G  |A
Z boson         |Z  |Z  |23        |2    |MZ  |!wZ  |1    |G  |Z
gluon           |G  |G  |21        |2    |0   |0    |8    |G  |G
W boson         |W+ |W- |24        |2    |MW  |!wW  |1    |G  |W^+
neutrino        |n1 |N1 |12        |1    |0   |0    |1    |L  |\nu^e
electron        |e1 |E1 |11        |1    |0   |0    |1    |   |e
mu-neutrino     |n2 |N2 |14        |1    |0   |0    |1    |L  |\nu^\mu
muon            |e2 |E2 |13        |1    |Mm  |0    |1    |   |\mu
tau-neutrino    |n3 |N3 |16        |1    |0   |0    |1    |L  |\nu^\tau
tau-lepton      |e3 |E3 |15        |1    |Mt  |0    |1    |   |\tau
u-quark         |u  |U  |2         |1    |0   |0    |3    |   |u
d-quark         |d  |D  |1         |1    |0   |0    |3    |   |d
c-quark         |c  |C  |4         |1    |Mc  |0    |3    |   |c
s-quark         |s  |S  |3         |1    |Ms  |0    |3    |   |s
t-quark         |t  |T  |6         |1    |Mtop|wtop |3    |   |t
b-quark         |b  |B  |5         |1    |Mb  |0    |3    |   |b
Higgs           |h  |h  |25        |0    |Mh  |!wh  |1    |   |h
odd Higgs       |~H3|~H3|36        |0    |MH3 |!wH3 |1    |   |(H3)
Charged Higgs   |~H+|~H-|37        |0    |MHC |!wHC |1    |   |(H+)
second Higgs    |~X |~X |35        |0    |MHX |!wHX |1    |   |(X)
```

Names of particles of odd sector are started with tilde ~
or ~~ (second DM)

Model Files: Feynman rules

Stored in file MODEL/work/models/lgrng1.mdl

```
Inert Dublet
 Lagrangian
P1  |P2  |P3  |P4  |>   Factor         <|> dLagrangian/ dA(p1) dA(p2)dA(p3)
A   |W+  |W-  |    |-EE                 |m3.p2*m1.m2-m1.p2*m2.m3- …...
A   |~H+|~H-|    |EE                  |m1.p3-m1.p2
B   |b   |A   |    |EE/3                |G(m3)
B   |b   |G   |    |GG                  |G(m3)
B   |b   |Z   |    |-EE/(12*CW*SW)      |4*SW^2*G(m3)-3*G(m3)*(1-G5)
B   |b   |h   |    |-EE*Mb/(2*MW*SW)    |1
B   |t   |W-  |    |-EE*Sqrt2/(4*SW)    |G(m3)*(1-G5)
W+  |W-  |~X  |~X  |EE^2/(2*SW^2)       |m1.m2
h   |~X  |~X  |    |-2*MW*SW/EE         |la3+la4+la5
Z   |Z   |~X  |~X  |EE^2/(2*CW2*SW^2)   |m1.m2
     …...............
```

Dark Matter in micrOMEGAs models.

Discrete symmetry.

MicrOMEGAs assumes a presence of discrete symmetry which is responsible for stability of Dark Matter. For instance it could be a Z_2 symmetry which divides all particles on two classes, odd and even. Then the lightest odd particle is stable and can be treated as DM.

For micrOMEGAs odd particles are particles whose name is started for tilde “~”. For example, ~X,~H3,~H+ in IDM.

In case of Z_4 symmetry internal charge for DM particles can be +/- 1 or 2. DM1- the lightest particle with charge 1 is always stable. But the lightest particle with charge 2 in stable if its mass less then mass of 2 DM1 particles. One can also construct a model with complex symmetry like $Z_2 \times Z_3$ which always has 2 DM particles.

MicrOMEGAs can work with models with 2DM classes which are marked by “~” and “~~”

Free model parameters.

[g]make called from MODEL directory creates file
work/VandP.c
Which contains information about free and constrained variables and model particles. This file is linked to *main* routine provides it information about variables and particles of the model. Interface with model is based on names of parameters and particles.

assignValW(*name,value*) assigns new value to parameter.

In order to download set of parameters micrOMEGAs has a function

readVar(*fileName***)**

Structure of file records has to be

name value [# comment]

For instance, in case of IDM

```
laL       0.001    # coupling
MHX       600      # inert sector Higgs
 Mh       125      # SM Higgs mass
la2       0.01     # coupling
MHC       604      # mass of charged Higgs
MH3       601      # mass of CP odd Higgs
```

Constrained models parameters

After assignment of parameters one has to call
sortOddParticles(outText);
which calculates constrained parameters and fined DM particle[s].
In case of conflict in calculation of constrained parameter this routine returns error code and *outText* contains a name of parameter which can not be calculated.
In case of success sortOddParticles detects the lightest odd particle[s] *CDM1 [CDM2]* and their masses
Mcdm1 [Mcdm2, Mcdm=min(Mcdm1,Mcdm2)]

Values of constrained parameters can be obtained by
findValW(*name*)
routine. Masses of particles can be obtained by pMass(*name*)

Generation of matrix elements

numout *cc ; // numout – is a type for matrix element in micrOMEGAs.

cc = newProcess(char*Process);   // call CalcHEP to calculate symbolically and compile matrix element for given process. For instance

cc = newProcess(“e,E->m,M”);

Matrix element is presented as a shared library and stored in directory

MODEL**/work/so_generated**

Name of library is related to names of particles in the process.
If model library already was generated and the model was not changed, then library is not recompiled.

For example, cross sections of 2->2 processes can be   calculated by

cs= cs22(cc,L,Pcm,cos_min,cos_max,&err);

Pcm – momentum in Center of Mass reference frame
cos_min, cos_max - cuts for cosine of scattering angle in the same frame
L=1 in case you have generated codes only for one process. For general case L numerates subprocesses.

So, micrOMEGAs works with a matrix element which is compiled by CalcHEP for given model and passed to micrOMEGAs

Loop induced vertices

micrOMEGAs is able to get numerical coefficients at vertex implemented in Lagrangian and use them to construct loop induced vertexes.
It is implemented for construction of Higgs-gamma-gamma and Higgs-glue-glue vertices which are needed for interface with HIGGSBOUNDS and LILITH for applying LHC constrains on Higgs particle
Also it need for correct calculation of Higgs width.

MicrOMEGAs fuctions

double complex lAAhiggs(Q, HiggsName);
double complex lGGhiggs(Q, HiggsName);

$$\lambda F_{\mu\nu} F^{\mu\nu}$$

double complex lAA5higgs(Q,HiggsName);
double complex lGG5higgs(Q,HiggsName);

$$\lambda F_{\mu\nu} \tilde{F}^{\mu\nu}$$

Q is reserved for the case of off-shell vertex.

For example in IDM Lagrangian

func1.mdl	lgrgn1.mdl
LAAH |-cabs(lAAhiggs(Mh,"h"))	A |A |h | |-4*LAAH |p1.p2*m1.m2-m2.p1*m1.p2

Structure of main programs

main.c,   main.F main.cpp files presented in micrOMEGAs model directories consist from several blocks enclosed into

```
#ifdef XXXXX
   ....................
#endif
```

User can switch on/off any of this block via corresponding *#define* instruction in the top of file

```
#define MASSES_INFO              // Display information about mass spectrum
#define CONSTRAINTS              //   Display B->s,gamma, Bs->mu,mu,
#define LILITH                   //  Test of  Higgs properties
#define HIGGSBOUNDS
#define OMEGA                    // Calculate relic density
#define INDIRECT_DETECTION       // Signals of DM annihilation in galaxy hallo
//#define RESET_FORMFACTORS      // Redefinition of  Form Factors and other parameters
#define CDM_NUCLEON              // Calculate amplitudes and cross-sections for  CDM-
                                    nucleon  collisions
//#define CDM_NUCLEUS            // Calculate number of events for 1kg*day and recoil
                                   energy distribution  for various nuclei
#define NEUTRINO                 // neutrino telescope
```

Example of micrOMEGAs session for IDM

./main data1.par

```
VERTEX: W- W+ h
VERTEX: L l h
VERTEX: C c h
VERTEX: T t h
VERTEX: B b h
VERTEX: ~H- ~H+ h

Dark matter candidate is '~X' with spin=0/2

=== MASSES OF HIGGS AND ODD PARTICLES: ===
Higgs masses and widths
PROCESS: h->2*x
PROCESS: W+->2*x
PROCESS: Z->2*x
PROCESS: h->W-,E,ne
Delete diagrams with W+<1
PROCESS: h->Z,ne,Ne
Delete diagrams with Z<1
    h   125.00 3.97E-03

Masses of odd sector Particles:
~X   : MHX   =   600.0 || ~H3  : MH3   =   601.0 || ~H+  : MHC   =   604.0

LILITH(DB15.09):  -2*log(L): 25.96; -2*log(L_reference): 0.00; ndf: 38; p-value: 9.31E-01
```

Continue

```
==== Calculation of relic density =====
PROCESS:  ~X,~X   ->AllEven,1*x{A,Z,G,W+,W-,ne,Ne,e,E,nm,Nm,m,M,nl,Nl,l,L,u,U,.....
PROCESS: ~H3,~X  ->AllEven,1*x{A,Z,G,W+,W-,ne,Ne,e,E,nm,Nm,m,M,nl,Nl,l,L,u,U,...
PROCESS: ~H3,~H3->AllEven,1*x{A,Z,G,W+,W-,ne,Ne,e,E,nm,Nm,m,M,nl,Nl,l,L,...
                    ……….

Xf=2.62e+01 Omega=1.13e-01
# Channels which contribute to 1/(omega) more than 1%.
# Relative contributions in % are displayed
   21% ~X ~X ->W+ W-
   14% ~X ~X ->Z Z
   11% ~H3 ~H3 ->W+ W-
    9% ~H+ ~H- ->W+ W-
    7% ~H3 ~H3 ->Z Z
    6% ~H+ ~X ->A W+
    5% ~H3 ~H+ ->A W+
    4% ~H+ ~H- ->A A
    4% ~H3 ~H+ ->Z W+
    3% ~H+ ~X ->Z W+
    3% ~H+ ~H- ->A Z
    2% ~H+ ~H- ->Z Z
    2% ~H+ ~X ->W+ h
    1% ~H+ ~H- ->h h
```

```
==== Indirect detection =======
 annihilation cross section 6.18E-26 cm^3/s
 contribution of processes
  ~X,~X -> W+ W-        6.01E-01
  ~X,~X -> Z Z          3.99E-01
sigmav=6.18E-26[cm^3/s]
Photon flux  for angle of sight f=0.10[rad]
and spherical region described by cone with angle 0.10[rad]
Photon flux = 9.37E-16[cm^2 s GeV]^{-1} for E=300.0[GeV]
Positron flux  =  1.04E-13[cm^2 sr s GeV]^{-1} for E=300.0[GeV]
Antiproton flux  =  5.91E-13[cm^2 sr s GeV]^{-1} for E=300.0[GeV]


==== Calculation of CDM-nucleons amplitudes  =====
PROCESS: QUARKS,~X->QUARKS,~X{u,U,d,D,c,C,s,S,t,T,b,B
Delete diagrams with _S0_!=1,_V5_,A
CDM[antiCDM]-nucleon micrOMEGAs amplitudes:
proton:  SI  1.497E-11 [1.497E-11]  SD  0.000E+00 [0.000E+00]
neutron: SI  1.512E-11 [1.512E-11]  SD  0.000E+00 [0.000E+00]
CDM[antiCDM]-nucleon cross sections[pb]:
 proton  SI 9.767E-14 [9.767E-14] SD 0.000E+00 [0.000E+00]
 neutron SI 9.962E-14 [9.962E-14] SD 0.000E+00 [0.000E+00]
===============Neutrino Telescope=======  for  Sun
 E>1.0E+00 GeV neutrino/anti-neutrino fluxes   1.81E+01/2.05E+01 [1/Year/km^2]
IceCube22 exclusion confidence level = 1.29E-07%
 E>1.0E+00 GeV Upward muon flux    2.337E-07 [1/Year/km^2]
 E>1.0E+00 GeV Contained muon flux 6.999E-07 [1/Year/km^3]
```

Calculation of DM relic density

For 1 DM case

darkOmega(&Xf,fast,Beps)
darkOmegaFO(&Xf,fast,Beps)

fast =1 for for fast calculation
Beps removes co-annihilation if $exp(\frac{2M_{cdm}-M_1-M_2}{T}) < Beps$

Return Ω h^2 and X_f =Mcdm/T_f , where $Y(T_f) = 2.5\, Y_{eq}(T_f)$

Solution of equation $$\frac{dY}{ds} = \frac{<v\sigma>}{3H}(Y^2 - Y_{eq}^2)$$

where s – entropy density, H – Hubble rate, Y-abundance

$$\Delta Y = Y - Y_{eq} \quad => \Delta Y \approx \frac{d\log Y_{eq}}{ds}\frac{3H}{2<v\sigma>}$$

Starting point for Runge-Kutta for darkOmega and solution for darkOmegaFO

$$\frac{1}{Y(T_0)} = \frac{1}{Y(T_f)} + \int_{s_0}^{s_f}\frac{<v\sigma>}{3H}ds$$

With 20% precision

$$Y(T_0)^{-1} = \int\limits_{s_0}^{s_f} \frac{< v\sigma >}{3H} ds$$

It gives a possibility to estimate contribution of different channels to DM formation

printChannels(Xf, cut, Beps, prnc, file)

```
# Relative contributions in % are displayed
   21% ~X ~X ->W+ W-
   14% ~X ~X ->Z Z
   11% ~H3 ~H3 ->W+ W-
    9% ~H+ ~H- ->W+ W-
    7% ~H3 ~H3 ->Z Z
    6% ~H+ ~X ->A W+
    5% ~H3 ~H+ ->A W+
```

Dark Matter asymmetry.

If DM particles is not self-conjugated one can assume Dm- antiDm asymmetry similar to barion asymmetry. In micrOMEGAs gobal parameter deltaY presents difference between DM/anti-DM abundances. darkOmega[FO] takes it into account.

Dark Matter asymmetry.

If DM particles is not self-conjugated one can assume Dm- antiDm asymmetry similar to barion asymmetry. In micrOMEGAs gobal parameter

deltaY

presents difference between DM/anti-DM abundances.
darkOmega[FO] takes it into account.

dmAsym parameter is calculated

Ω(+/-) = Ω(1 +/-dmAsym)/2

dmAsym contributes to all function of direct/indirect detection and neutrino telescope.

For 2 and 1 DM

darkOmega2(fast,Beps) return Ω h^2

If Mcdm1 and Mcdm2 are different we first have freeze out for heavy DM. The lightest one is in thermal equilibrium with SM particles and returns fast to equilibrium state in case of deviation.

$$\frac{d\Delta Y}{ds} = \frac{2Y_{eq} < v\sigma >}{3H}\Delta Y - \frac{dY_{eq}}{ds}$$

Thus Runge-Kutta needs very small step for solution

Special stiff solution (Numerical Recipes in C) is applied.

Not sensityve to deltaY, does not calculate X_f, no printChannels

Direct Detection

To predict results of direct detection experiment in the given model of elementary particles interaction we have to calculate cross sections of DM – nuclei elastic scattering.

So, we have in the model

DM - quarks interaction

Then we have to calculate

DM - nucleon scattering cross section

And at next step

DM -nuclei scattering cross section

Velocities of DM particles in halo of Milky Way are about orbital velocities of stars

$$v \approx 220km/s \approx 10^{-3}c$$

We can treat such scattering as scattering at $v \to 0$ limit, taking into account that elastic cross section can be finite in this limit.

DM – fermion interaction in the v → 0 limit

	DM Spin	$\hat{\mathcal{O}}_e$ Even operators	$\hat{\mathcal{O}}_o$ Odd operators
SI	0 1/2 1	$2M_\chi\phi_\chi\phi^*_\chi\overline{\psi}_f\psi_f$ $\overline{\psi}_\chi\psi_\chi\overline{\psi}_f\psi_f$ $2M_\chi A^*_{\chi\mu}A^\mu_\chi\overline{\psi}_f\psi_f$	$i(\partial_\mu\phi_\chi\phi^*_\chi - \phi_\chi\partial_\mu\phi^*_\chi)\overline{\psi}_f\gamma^\mu\psi_f$ $\overline{\psi}_\chi\gamma_\mu\psi_\chi\overline{\psi}_f\gamma^\mu\psi_f$ $+i\lambda_{q,o}(A^{*\alpha}_\chi\partial_\mu A_{\chi,\alpha} - A_\chi{}^\alpha\partial_\mu - A^*_{\chi\alpha})\overline{\psi}_f\gamma_\mu\psi_f$
SD	1/2 1	$\overline{\psi}_\chi\gamma_\mu\gamma_5\psi_\chi\overline{\psi}_f\gamma_\mu\gamma_5\psi_f$ $\sqrt{6}(\partial_\alpha A^*_{\chi\beta}A_{\chi\nu} - A^*_{\chi\beta}\partial_\alpha A_{\chi\nu})\epsilon^{\alpha\beta\nu\mu}\overline{\psi}_f\gamma_5\gamma_\mu\psi_f$	$-\frac{1}{2}\overline{\psi}_\chi\sigma_{\mu\nu}\psi_\chi\overline{\psi}_f\sigma^{\mu\nu}\psi_f$ $i\frac{\sqrt{3}}{2}(A_{\chi\mu}A^*_{\chi\nu} - A^*_{\chi\mu}A_{\chi\nu})\overline{\psi}_f\sigma^{\mu\nu}\psi_f$

SI – Spin independent (scalar) – interactions without spin flip.
SD – Spin dependent – interactions with spin flip.
Even - DM and DM* have the same amplitude.
Odd - DM and DM* amplitudes have different signs.

Operator expansion

SI and SD operators have the following normalization conditions for scattering at rest:

$\text{SI} : |A^{SI}|^2 = 64 M_{DM}^2 M_f^2$
$\text{SD}: |A^{SD}|^2 = 192 M_{DM}^2 M_f^2$

Assuming effective Lagrangian

$$\hat{\mathcal{L}}_{eff}(x) = \sum_{q,s=(even,odd)} \lambda_{q,s}\hat{\mathcal{O}}_{q,s}(x) + \xi_{q,s}\hat{\mathcal{O}}'_{q,s}(x)$$

micrOMEGAs creates new model with effective operators and finds coefficients λ and ξ calculating amplitudes for collision at rest

$$\langle q(p_1), \chi(p_2)|\hat{S}\hat{\mathcal{O}}_{e,o}|q(p_1), \chi(p_2)\rangle$$

$$\langle \bar{q}(p_1), \chi(p_2)|\hat{S}\hat{\mathcal{O}}_{e,o}|\bar{q}(p_1), \chi(p_2)\rangle$$

Twist-2 operators

One exeption: to select contribution of twist-2 operators

$$\mathcal{O}_{q,t} = \frac{1}{2}(\bar{\chi}\gamma_\mu\partial_\nu\chi)\bar{q}(\gamma^\mu\overrightarrow{\partial}^\nu - \gamma^\mu\overleftarrow{\partial}^\nu + \gamma^\nu\overrightarrow{\partial}^\mu - \gamma^\nu\overleftarrow{\partial}^\mu + im_q g^{\mu\nu})q$$

micrOMEGAs tests forward scattering amplitudes for finite momentum

$$< q(p_1), \chi(p_2)|\mathcal{O}_{q,t}\mathcal{O}_{q,e}|q(p_1), \chi(p_2) >= -32m_q M_\chi(4(p_1.p_2)^2 - m_q^2 M_\chi^2)$$

Nucleon form factors for light quarks

Each operator at quark level leads to the same type operator a nucleon level with form factor

Even scalar form factors

The operator $\langle N|m_q\overline{\psi}_q\psi_q|N\rangle$ is interpreted as the contribution of quark q to the nucleon mass, M_N,

$$\langle N|m_q\overline{\psi}_q\psi_q|N\rangle = f_q^N M_N$$

$f_{u,d,s}^N$ are known from hardon spectroscopy, data of πN scattering and lattice calculations (s-quark)

Odd scalar form factors

$$\langle N|\overline{\psi}_q\gamma_\mu\psi_q|N\rangle = \mathrm{f}_{V_q}^N\langle N|\overline{\psi}_N\gamma_\mu\psi_N|N\rangle$$

Just give us (quark) - (anti-quark) number counting because of vector current conservation.

$$f_{V_u}^P = 2 \quad f_{V_d}^P = 1$$

Even vector form factor $\gamma 5 \gamma_\mu$

Describe contribution of quarks and and anti-quarks to nucleon spin

Odd vector form factor $\sigma_{\mu\nu}$

Describe difference of contribution of quarks and antiquarks to nucleon spin.

Form factors of light quarks are presented by global parameters

Proton		Neutron		
Name	value	Name	value	comments
ScalarFFPd ScalarFFPu ScalarFFPs	0.0191 0.0153 0.0447	ScalarFFNd ScalarFFNu ScalarFFNs	0.0273 0.011 0.0447	Scalar form factor
pVectorFFPd pVectorFFPu pVectorFFPs	-0.427 0.842 -0.085	pVectorFFNd pVectorFFNu pVectorFFNs	0.842 -0.427 -0.085	Axial-vector form factor
SigmaFFPd SigmaFFPu SigmaFFPs	-0.23 0.84 -0.046	SigmaFFNd SigmaFFNu SigmaFFNs	0.84 -0.23 -0.046	Tensor form factor

Twist-2 form factors

are obtained via integration of structure functions.

$$< N(p)|\mathcal{O}^{\mu\nu}_{q,t}|N(p) >= (p^\mu p^\nu/M_N - g^{\mu\nu}M_N/4)\int_0^1 (q(x)+\bar{q}(x))x dx$$

Heavy quark form factors

are obtained by QCD calculations. No contribution to SD part and odd SI.

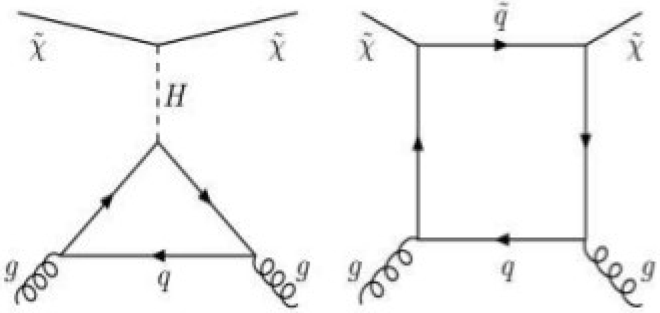
The anomaly of the trace of energy-momentum tensor in QCD implies (Wanstein,Zakharov, Shifman)

$$M_N\langle N|N\rangle = \langle N|\sum_{q\leq n_f} m_q\overline{\psi}_q\psi_q(1+\gamma) + (\frac{\beta^{n_f}}{2\alpha_s^2})\alpha_s G_{\mu\nu}G^{\mu\nu}|N\rangle$$

where $\beta^{n_f} = -\alpha_s^2/4\pi(11 - 2n_f/3 + \alpha_s/4\pi(102 - 38n_f/3))$.

$$\langle N|m_Q\bar{\psi}_Q\psi_Q|N\rangle = -\frac{\Delta\beta}{2\alpha_s^2(1+\gamma)}\langle N|\alpha_s G_{\mu\nu}G^{\mu\nu}|N\rangle \approx \frac{2}{27}\langle N|M_N\bar{\psi}_N\psi_N|N\rangle$$

Heavy quark loops



Diagrams that contribute to DM-gluon interaction via heavy quark loops

Heavy quarks interact with nucleon via gluon condensate. For triangle (Higgs) heavy quark condensate is a good approximation. For box diagrams one needs loop calculation.

For renormalizable interactions corresponding boxes where presented in

DM spin 1/2 M.Drees & M.Nojiri hep-ph/9307208
DM spin 0 and 1 Hisano,Junji,Nagai,Ryo,Nagata,Natsumi arXiv:1502.02244

MicrOmegas replaces propagators on corresponding loop functions without testing type of interaction arXiv 0803.2360

Nucleon amplitudes and cross sections in micrOMEGAs

nucleonAmplitudes(name_of_DM ,pA0,pA5,nA0,nA5);
Output: *pA0,pA5,nA0,nA5* – 2 dimension arrays
Proton
pA0[even SI, odd SI] pA5[even SD, odd SD]
Neutron
nA0[even SI, odd SI] nA5[even SD, odd SD]

Then DM-nucleon cross section in [pb] units are

$\sigma_{si}= C*A^2 \quad \sigma_{SD}= 3*C*A^2$ where $C=4/\pi*3.89E8*(M_N*M_{dm}/(M_N+ M_{dm)})^2$

Nuclei interactions

Nuclei form factors

For zero DM velocity DM-nucleus SI cross section reads

$$\sigma_0^{SI} = \frac{4\mu^2}{\pi}\left(\lambda_p Z + \lambda_n (A-Z)\right)^2 \ , \quad \mu = \frac{M_{cdm} M_A}{M_{cdm} + M_A}$$

where λ_p, λ_n are amplitudes for DM scattering on nucleons; M_A, Z, A are the nucleus mass, charge, and atomic number respectively. For a small DM velocity, $v \approx 10^{-3}c$, we neglect the dependence on the small momentum transfer in the cross section but include this dependence in the nucleus form factor

$$\frac{d\sigma^{SI}}{dE} = \frac{\sigma_0^{SI}}{E_{max}} F_A^2(q) \ , \ \ 0 < E < E_{max} = 2\left(\frac{v^2\mu^2}{M_A}\right)$$

For SI interactions, $F(q)$ is a Fourier transform of the nucleus distribution function,

$$F_A(q) = \int e^{-iqx} \rho_A(x) d^3x$$

micrOMEGAs use the Fermi distribution function

$$\rho_A(r) = \frac{c_{norm}}{1 + exp((r - R_A)/a)}$$

where $a = 0.52fm$ nuclei surface thickness, and
$R_A = 1.23A^{\frac{1}{3}} - 0.6\mathrm{fm}$ nuclei radius

There are similar but more complicated formulas for SI nucleus cross section which depends on 3 form factors, proportion to nucleus momentum J and does not lead to A enhancement.

micrOMEGAs function for nuclei

nucleusRecoil(

f, - velocity distribution f(v[km/s]) normalized by $\int\limits_0^\infty v f(v) dv = 1$

A, - atomic number

Z, - nucleus charge

J, - number of spin states

Sxx, - SD formfactors

dNdE - recoil energy distribution stored in array

)

dNdERecoil(E[keV],dNdE) interpolates dNdE table and gives spectrum in 1/keV/kg/day units

For example:

nEvents=nucleusRecoil(Maxwell,73,Z_Ge,J_Ge73,SxxGe73,dNdE);

```
         Result depends on global parameter
 rhoDM     0.3[GeV/cm^3] Dark Matter density at Rsun
```

How to get plot for dNdE ontained by nucleusRecoil?

displayPlot(title,xName,xMin,xMax,lScale, N, ...)
displays several curves/histograms on one plot. Here title presents title of plot,
xName Is a name of variable,
xMin,xMax are the lower and upper limits for x
lScale is a logarithmic scale flag for x-axis,
N is a number of curves/histograms to display.

After the parameter N displayPlot expects N*4 parameters, where each tetrad can contain

textual	Dim	array of data	array of error
label	Dim	array of data	NULL
	0	(double* f)(double x)	NULL
	0	(double* f)(double x, void*arg)}	arg

For linear scale lScale=0 , the arrays of data and errors should correspond to a grid

x[i]=xMin+(i+0.5)(xMax-xMin)/Dim

For Log scale lScale=1

x[i]=xMin (xMax/xMin)^((i+0.5)/Dim)

For Recoil energy

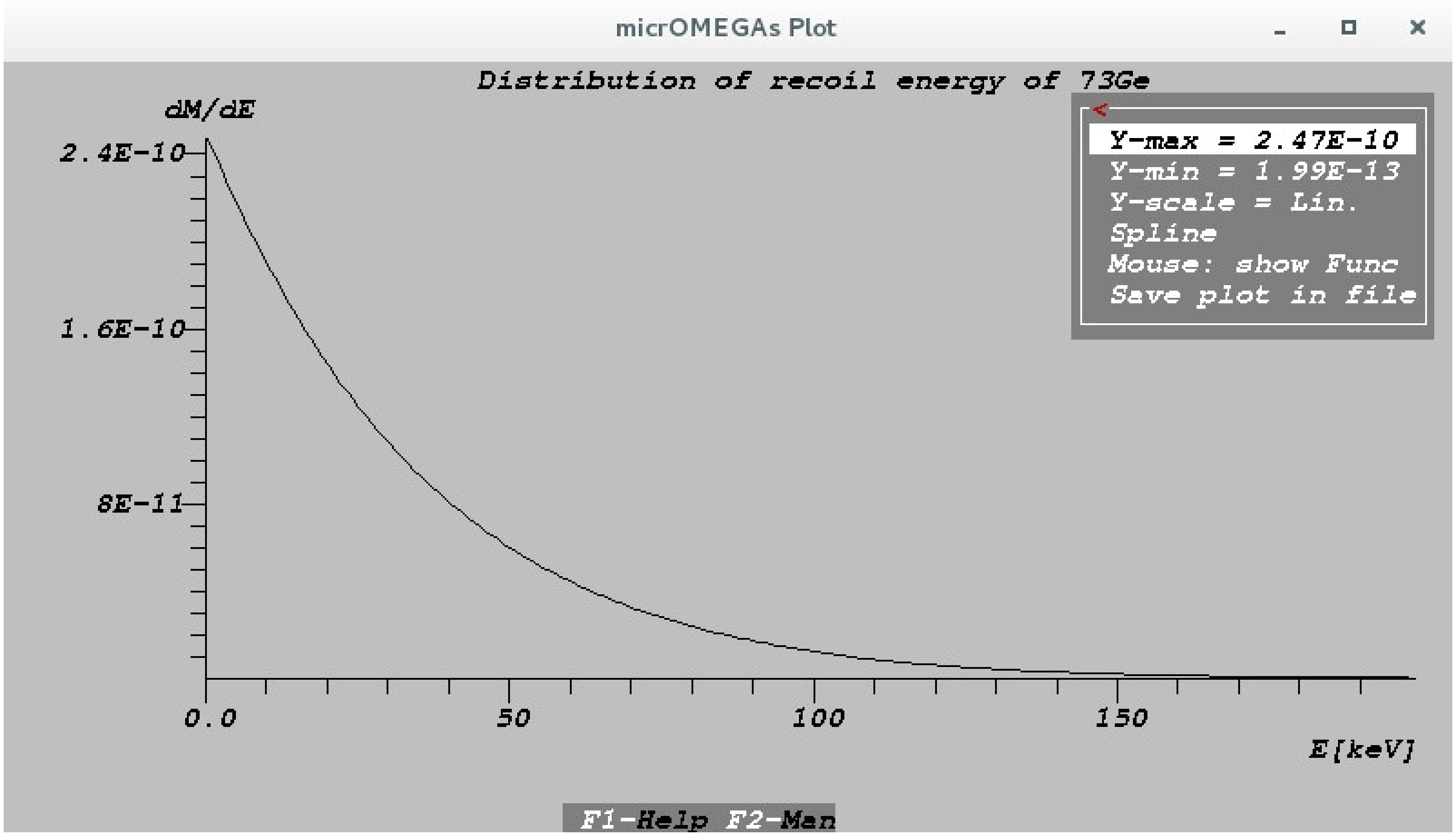
```
displayPlot("Distribution of recoil energy of 131Xe","E[KeV]",0,200,0,1, "
                  “dN/dE",0,dNdERecoil,dNdE);
```

One has to uncomment

```
#define SHOWPLOTS
```

to see different plots for distribution produced by micrOMEAGs.

There is an menu driving option to save plot in Root, PAW, and GnuPlot formats.



Y-max = 2.47E-10
Y-min = 1.99E-13
Y-scale = Lin.
Spline
Mouse: show Func
Save plot in file

Neutrino telescope

micrOMEGAs uses direct detection module to calculate number of DM captured by Sun/Earth.

Captured DM is concentrated in the center of Sun/Earth and neutrino produced in result of DM annihilation can be detected by neutrino telescope experiment (IceCube, Super-Kamiokande, Baksan).

DM annihilation inside of Sun/Earth is different from annihilation in vacuum. Also there are effects of propagation and oscilation.

For flux of resulting muon neutrinos micrOMEGAs uses tables obtained by
WimpSim package: J. Edso et.al arXiv 0709.3898
Or
PPPC4DMnu: M. Cirelli, et.al. arXiv 1312.6408

Agreement between two sets is not very good.

MicrOMEGAs routine basicNuSpectra reads these tables depending on
WIMPSIM flag
WIMPSIM=1 for WimpSIm
WIMPSIN=0 for PPPC4DMnu

basicNuSpectra(*forSun,Mcdm,pdg, pol, nu, nu_bar*)
where
forSun is 1 or 0,
pdg - is PDG number of annihilation channel.
pol=-1(1) corresponds to longitudinal (trans-verse) polarisation of vector bosons or to left-handed (right-handed) fermions, pol=0 is used for unpolarized spectra.
Arrays **nu, nu_bar** contains spectra.

SpectdNdE(E,spect) interpolates arrays.

Combining DM capture rate and annihilation spectra micrOMEGAs calculates muon neutrino fluxes at Earth surface

neutrinoFlux(Maxwell,forSun, nu,nu_bar);

After that one can apply iceCude22 limits for neutrino spectra: iceCube22 arXiv 0902.2460

exLevIC22(nu,nu_bar,NULL) exclusion level.

MicrOMEGAs is able to calculate muon spectra produced to neutrinos, but we have not now angular resulution for muon flax. It should be improved to apply micrOMEGAs to other neutrino telescope experiments

Indirect detection in micrOMEGAs

Indirect detection -detection of photons, positrons and antiprotons signal obtained in result of DM annihilation in Galactic Hallo.

For various spectra we use NZ=250 dimention arrays and interpolation function for them is SpectdNdE(E,spectArr)
One can use displayPlot to see and compare difference spectra.

vsigma=calcSpectrum(key,Sg,Se,Sp,Sne,Snm,Snl,&err)
Calculates vσ cross section in cm^3/sec units of DM annihilation and photon Sg, positron Se, antiprotons Sp, and 3 neutrino spectra at one collision of DM particles.
Here the avarage over Dm,Dm/antiDm is done. dmAssym is taken into account, In case of 2 DM particles we have an average over all types of collisions. PITHIA 6.4 was used for hadronisation of primary annihilation channels.

Meaning of key parameter: 1-takes into account W/Z polarization
2-include gammas from 2->2+gamma
4-print cross sections

Halo profile

DM distribuion is defined by DM density at Sun, parameter rhoDM and hallo profile. By default micrOMEGAs uses Zhao profile

$$F_{halo}(r) = \left(\frac{R_\odot}{r}\right)^\gamma \left(\frac{r_c^\alpha + R_\odot^\alpha}{r_c^\alpha + r^\alpha}\right)^{\frac{\beta-\gamma}{\alpha}}$$

with alpha=1, beta=3 rc=20kpc.

setProfileZhao(α,β,γ,rc) change these parameters.

As Well as

setHaloProfile(F) allows to substitute any profile presented by function F(r)

The command **setHaloProfile(hProfileZhao)** sets back the Zhao profile

Photon flux

gammaFluxTab(fi,dfi,sigmav,Sg,Sobs)
fi is the angle between the line of sight and the center of the galaxy,
dfi is half the cone angle which characterizes the detector resolution (the solid angle is 2π(1 − cos(df i)) ,
sigmav is the annihilation cross section,
Sg - photo spectrum at point of annihilation
Sobs is resulting photon flux in [1/(GeV cm^2 s)] units.

For all implemented models we have
Dm,Dm -> photon, photon and Dm,Dm -> photon, Z
loop induced signals. This signals are not compiled automatically in run-time but generated in advance by means of FormCalc.

One has to uncomment
//#define LoopGAMMA
to force micrOMEGAs to work with point like gamma signal.
Function   loopGamma(&vcs_gz,&vcs_gg) calculates annihilation calculates annihilation rates vcs_gz and vcs_gg [cm^3/s]

Then gammaFlux(fi,dfi,vcs_gz[gz])) returns corresponding fluxes

Antiproton and positron fluxes

- **posiFluxTab**(Emin,sigmav, Se, Sobs)

- **pbarFluxTab**(Emin,sigmav, Sp, Sobs)

The same style as for photons. But depends on propagation parameters

```
K_dif     | 0.0112| kpc^2/Myr |The normalized diffusion coefficient
L_dif     |  4    | kpc       | Vertical size of the Halo diffu
Delta_dif| 0.7    |           |Slope of the diffusion coefficient
Tau_dif   | 10^16 |   s       |Electron energy loss time
Vc_dif    | 0     |  km/s     |Convective Galactic wind
```

And finally
  solarModulation(Phi, mass, stellarTab, earthTab)
allows to take into account solar modulation effect.
  Here Phi potential [MeV], mass is mass of particle,
  stellarTab flux before modulation
  earthTab flux after modulation.

Implementation of new models in micrOMEGAs

- The command

  ./newProject *MODEL*

launched from the root micrOMEGAs directory creates the directory *MODEL*, which containa all files needed to run micrOMEGAs with the exception of the new model files.

- The new model files in the CalcHEP format should then be included in the sub-directory MODEL/work/models. The files needed are

  vars1.mdl, func1.mdl,   prtcls1.mdl, lgrng1.mdl extlib1.mdl

- Model files can be created by mean of

  LanHEP,   FeynRules, Sarah

LanHEP is included in micrOMEGAs package. Each model directory contains lanhep subdirectory with source files   with Makefile which calls LanHEP.

See LanHEP manual

micromega_X.Y/Packages/LanHEP/manual/man31.pdf

Follow to examples presented in micrOMEGAs.

The simplest one is IDM/lahhep

SLHAplus[arXiv 1008.0181]: Tools for model implementation

Routines **slhaRead, slhaVal openAppend, aPrintF**

File with particle spectrum | CalcHEP model file

```
                                                 |
BLOCK MASS   # Mass spectrum                     |  slhaRead(file_name, mode)
#  PDG Code      mass             particle       |
 25    1.15137179E+02    # neutral Higgs         | Mh    | slhaVal("MASS",0,1,25)
 37    1.48428409E+03    # charged Higgs         | MHC   | slhaVal("MASS",0,1,37)
                                                 |
BLOCK NMIX  # Neutralino Mixing Matrix           |
 1  1      9.98499129E-01   # Zn11               | Zn12 | slhaVal("NMIX",0,2,1,2)
 1  2     -1.54392008E-02   # Zn12               |
```

Example: SUGRA with SuSpect

```
open   |openAppend("suspect2_lha.in")
input1|aPrintF("Block MODSEL  # Select model\n  1  1   # SUGRA\n")
input2|aPrintF("Block SMINPUTS\n 5 %E#mb(mb)\n 6 %E#mt(pole)\n",MbMb,Mtp)
input3|aPrintF("BLOCK MINPAR\n 1 %E #m0\n 2 %E #m1/2\n ",Mzero,Mhalf)
input4|aPrintF("3 %E #tb\n 4 %E #sign(mu)\n 5 %E #A0\n",tb,sgn,A0)
sys    |System("$CALCHEP/../Packages/SuSpect_2.41/suspect2.exe")
rd     |slhaRead("suspect2_lha.out",0) % mode=4 do not read decays
Mh     |slhaVal("MASS", 0 , 1 , 25)
```

SLHAplus updated

Now people use SLHA format more widely than it was proposed by Peter Skands. micrOMEGAs SLHA package was updated correspondingly. For example

```
Block HiggsBoundsResults
#CHANNELTYPE 1: channel with the highest statistical sensitivity
 1  1       328                                                   # channel id
 1  2         1                                                   # HBresult
 1  3  0.72692779334500290                                        # obsratio
 1  4         1                                                   # ncombined
 1  5  ||(p p)->h+..., h=1 where h is SM-like (CMS-PAS-HIG-12-008)|| # channel
```

```
slhaSTRFormat("HiggsBoundsResults","1 5 ||%[^|]||", channal);
```

```
Block FOBS # Flavour observables
# ParentPDG type value       q   NDA ID1 ID2 ID3 ... comment
   5     1   2.95061156e-04  0    2   3  22       # BR(b->s gamma)
  521    4   8.35442304e-02  0    2 313  22       # Delta0(B->K* gamma)
  531    1   3.24270419e-09  0    2  13 -13       # BR(B_s->mu+ mu-)
```

```
Bsg= slhaValFormat("FOBS", 0., "5 1 %E 0 2 3 22")
```