

h1. FlySA

Compilation:

```
make veryclean (only first time)
make deps (only first time and after every 'make veryclean')
make
```

Usage:

printscore: printscore of a datafile (simple). Print the chi square and the root mean square (RMS)

```
"Usage: printscore [-a <accuracy>] [-D] [-f <float_prec>] [-g
<g(u)>] [-G]\n"
"                [-h] [-i <stepsize>] [-o] [-p] [-s
<solver>] [-v]\n"
"                [-x <sect_title>]\n"
"                <datafile>\n";
```

unfold: solve a model parameters and return solution for given time points

```
"Usage: unfold [-a <accuracy>] [-D] [-f <float_prec>] [-g
<g(u)>] [-G]\n"
"                [-h] [-i <stepsize>] [-j <timefile>] [-o] [-p
<pstep>]\n"
"                [-s <solver>] [-t <time>] [-v] [-x
<sect_title>]\n"
"                [-z <gast_time>]\n"
"                <datafile> [<genotype>]\n";
```

scramble: generate new random model parameters within constraint limits. Note:

```
'-x sect_title' is the section you want to regenerate
"Usage: scramble [-f <float_prec>] [-h] [-v]\n"
"                [-w <out_file>] [-x <sect_title>]\n"
"                <datafile>\n";
```

The programs have the same arguments, though some arguments may have different behaviour according to the program.

The list of arguments:

- a <arg> : solver accuracy
- b <arg> : backup frequency
- f <arg> : float precision
- g <s,t,h> : choose g(u) function [sqrt, tanh, exp, hvs]
- h : print the help message (help may show obsolete options...)
- i <arg> : stepsize
- m <o,w> : (o)LS or (w)LS
- r <0/1> : shows RMS or chi square (recommended value 1)
- s <arg> : solver [a=Adams, bd=BaDe, bs=BuSt (default), e=Euler, h=Heun, mi or m=Milne, me=Meuler, r4 or r=Rk4, r2=Rk2, rck=Rkck, rf=Rkf, kr=Krylov]
- w <arg> : output dir. If omitted, create a directory name <input_name_out>
- x <sect_title> : input parameters to be read (it should exist in the input file). Default is 'input'

That's it!

Run example:

```
./fly/fly_sa -s kr -l -i 4.0 -a 0.0001 -w output/fly_output  
input/default.config
```

%run a simple SA on file default.config using Krylov numerical solver with solver stepsize 4 and accuracy 0.0001, display log to terminal and generate output at output/fly_output

h1. FlySSm

This is a version of FlySA to use with different optimizers.

Compilation:

```
make veryclean (only first time)
make deps (only first time and after every 'make veryclean')
make
```

Usage:

After compilation files ggn.o and ggn are created. File ggn.o can be used together with ggn.h in a different C code, external to FlySA. In ggn.c there is a ggn() function that takes in input all variable parameters of the model, and returns a score.

There is also a main() function where user can see an example of usage of the ggn() function.

The example can be executed by running the ggn executable, which will make one iteration with hardcoded parameters and print out the score.