Setting up WimpSim and running WimpAnn

July 19, 2018

First, make sure that you have `wget` installed by running:

```
brew install wget
```

Next navigate to the directory where you would like to install `WimpSim`. Keep in mind that this directory will likely end up containing a lot of data, so it might be advisable to put it somewhere in your `/data/user/` directory. Personally, I have mine in the path `/data/user/jlazar/wimpsim`. Once there, download `WimpSim` and all its dependencies by running:

```
wget http://wimpsim.astroparticle.se/code/pythia-6.4.26.f
wget http://staff.fysik.su.se/~edsjo/darksusy/tars/darksusy-6.1.0.tar.gz
wget http://wimpsim.astroparticle.se/code/nusigma-1.19-pyr.tar.gz
wget http://wimpsim.astroparticle.se/code/wimpsim-4.1.1.tar.gz
```

Note: These are the most up to date versions of these programs as of July 19, 2018. Make sure that these are still up to date on the WimpSim website. Next compile Pythia by running:

```
gfortran -O -c pythia-6.4.26.f
```

Next all the `.tar` files by running:

```
tar -xvf darksusy-6.1.0.tar.gz
tar -xvf nusigma-1.19-pyr
tar -xvf wimpsim-4.1.1.tar.gz
```

These will all take a decent amount of time, and will produce quite a bit of command line output. Next change to the unpacked `DarkSUSY` directory with:

```
cd darksusy-6.1.0
```

And finally run:

```
./configure
make
```

Once again, both of these will produce a lot of output text on the command line. Next change directories to the `nusigma` directory and compile it by running:

```
cd ../nusigma-1.19-pyr
./configure
make libnusigma
```

All the dependencies should now be configured. Now navigate to the `WimpSim` directory by running:

```
cd ../wimpsim-4.1.1
```

Before configuring, we must change the paths in `configure` to reflect the paths to `DarkSUSY`, `nusigma`, and `Pythia`. These are under the variable names `DARKSUSY_DIR`, `NUSIGMA_DIR`, and `PYTHIA` respectively. For me these paths are `/data/user/jlazar/wimpsim/darksusy-6.1.0`, `/data/user/jlazar/wimpsim/nusigma-1.19-pyr`, and `/data/user/jlazar/wimpsim/pythia-6.4.26.o`. Now compile `WimpSim` by running:

```
./conf.gfortran
make all
```

Congratulations, `WimpSim` is now installed! There are a few more steps to go before being able to run `WimpAnn`. First you must make a file with the neutrino oscillation parameters. There is one on my website which is up to date as of July 19, 2018. You can get this file with:

```
wget http://www.jefflazar.zone/research/files/nuosc_params.txt
```

I moved this to my `/wimpsim-4.1.1/data/` directory for neatness. This can done with:

```
mv nuosc_params.txt data/
```

If you (reasonably) do not trust my parameters and want to make your own, the layout of the file is:

```
<theta12>
<theta13>
<theta23>
<cp-violating phase>
<Delta 21>
<Delta 31>
```

Then create a directory for your `WimpAnn` goodies. I chose to make mine at `/data/user/jlazar/wimpsim/wimpann`. Set this and some other variables with:

```
WA_DIR=/data/user/jlazar/wimpsim/wimpann
NUOSC_FILE=/data/user/jlazar/wimpsim/wimpsim-4.1.1/data/nuosc_params.txt
NUM_RUNS=1000000
```

Now turn our attention to the setup code for `WimpAnn`, `wasetup.pl`. You can go into this file with:

```
vi scr/wasetup.pl
```

and edit which m_χ (in GeV) you want to simulate, and which decay channels as well. The decay channels are numbered according to fig. 1. Now that you have set those parameters, you can run:

```
./scr/wasetup.pl $WA_DIR $NUOSC_FILE $NUM_RUNS
```

I got some errors at this point about `run-wa-rev.one` and `run-wa.one` not being moved successfully. Not to worry though, you can do this easily with:

```
cp scr/run-wa.one scr/run-wa-rev.one $WA_DIR/scr
```

Then go to the `WimpAnn` directory and start the simulation with:

```
cd $WA_DIR
./scr/run-wa.one
```

`run-wa.one` is a script which runs the simulations one at a time and then moves the simulations that are done from the `runs` directory to `runs-done`. If you need to do further simulations, return to the `wimpsim-4.1.1` directory, and edit the `wasetup.pl` file to suit your needs.

	Pythia codes		
Channel number	Particle 1	Particle 2	Annihilation products
1	1	-1	$d\bar{d}$
2	2	-2	$u\bar{u}$
3	3	-3	$s\bar{s}$
4	4	-4	$c\bar{c}$
5	5	-5	$b\bar{b}$
6	6	-6	$t\bar{t}$
7	21	21	gg
8	24	-24	W^+W^-
9	23	-23	Z^0Z^0
10	13	-13	$\mu^-\mu^+$
11	15	-15	$\tau^-\tau^+$
12	12	-12	$\nu_e\bar{\nu}_e$
13	14	-14	$\nu_\mu\bar{\nu}_\mu$
14	16	-16	$\nu_\tau\bar{\nu}_\tau$
100	-	-	KK DM (UED B(1))

Figure 1: The annihilation channels implemented into WimpAnn, together with the particle codes for these channels in Pythia. Note that channel 10 is just included to use the same channel numbering as in DarkSUSY, it gives no neutrinos for annihilations in the Sun/Earth. Channel 100 is using a standard set of branching fractions for UED DM B(1).