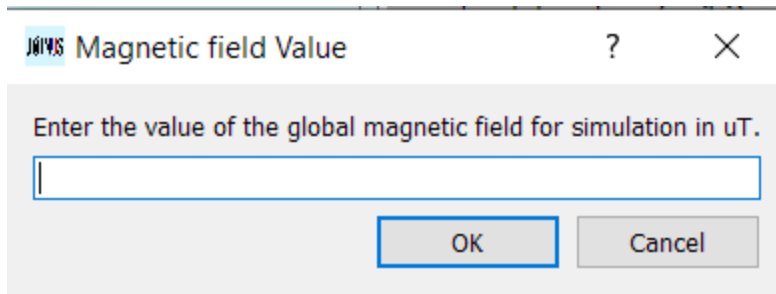


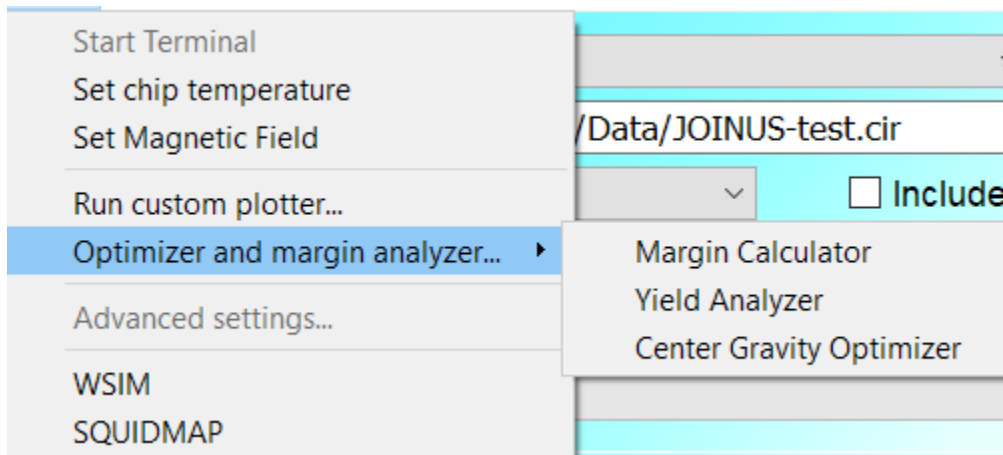
JOINUS New features:

Added magnetic effect to the menu and now magnetic field will change critical current. Critical magnetic field is affected by the temperature and can be set from "myglobalvars.cpp" under "BC2" parameter. Now the default is set to 1mT.



Margin and optimizer

The optimizer is now in three different parts:



Margin calculator includes files {"margincalc.cpp", "margincalc.h" and "margincalc.ui"}.

Dialog

Netlist file : OK

Reference file for netlist outputs : Cancel

File of netlist variables : Reset

Temporary folder :

Output folder : ☐ Magnetic noise ☐ Thermal noise

step between two consecutive values

Nominal value

L1 J1

Margin calculation method:
Nominal margins are set to [-90% - +200%]
One parameter is swept while all other parameters remain at nominal value.
The margin analyzer stops when the step to the next parameter value is below the specified maximum margin uncertainty.

Simulate netlist with nominal values

Settings

Maximum uncertainty of margins [%] :

Temperature [K]:

Rn*Ic (uV) :

Yield analyzer includes {"yieldcalc.cpp", " yieldcalc.h" and " yieldcalc.ui"}.

Dialog

Netlist file : OK

Reference file for netlist outputs : Cancel

File of netlist variables : Reset

Temporary folder :

Output folder : ☐ Magnetic noise ☐ Thermal noise

Output yield

1

0.5

Nominal value

Parameter value

2σ

Yield analyzing method:
1- A standard deviation of a Gaussian distribution is selected for parameters under study.
2- Values of all chosen parameters are picked randomly within this distribution.
3- They are inserted in a new netlist.
4- Simulation is performed and the correct operation of the circuit is checked (pass/fail).
5- Steps 2 to 4 are run multiple times to estimate statistically the yield of the circuit, which is the proportion of sets of parameters that correspond to circuits operating correctly.
6- A higher standard deviation is chosen and the full procedure from step 1 is started again.
7- The yield with standard deviation curve is plotted and can be saved.

Simulate netlist with nominal values

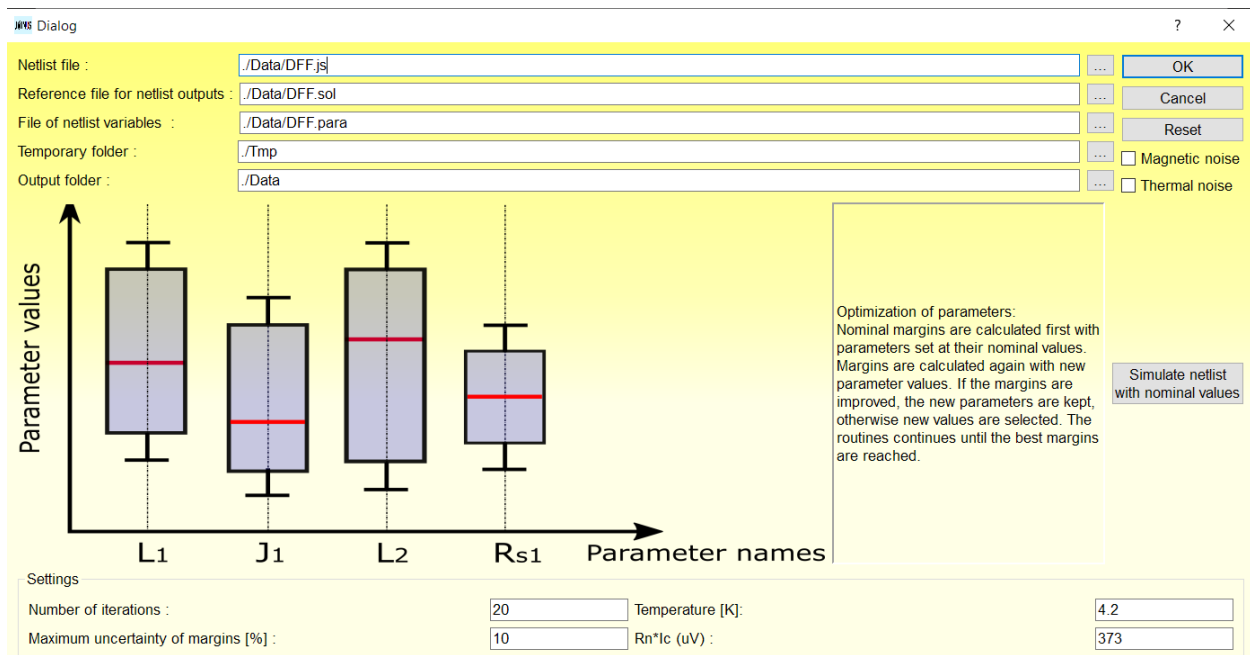
Settings

Standard deviation of parameter [%] : Number of iterations :

Standard deviation of junctions [%]: Temperature [K]:

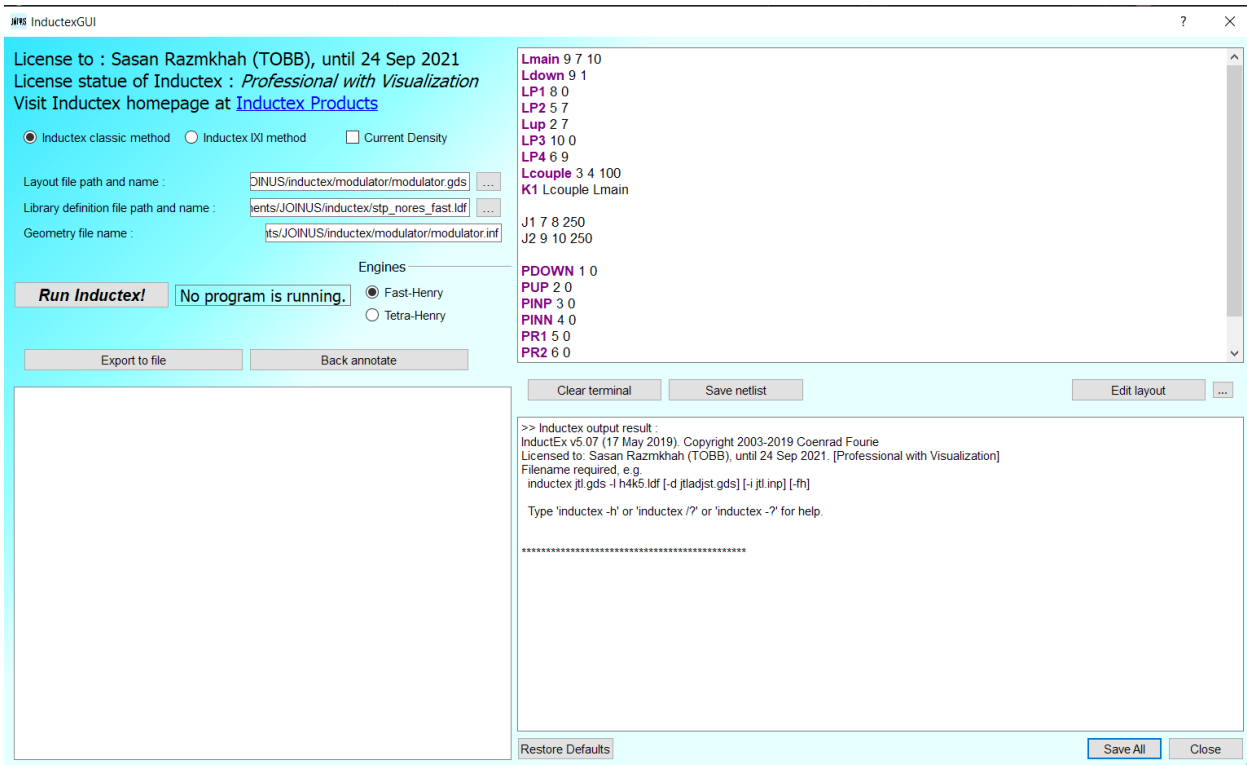
Maximum uncertainty of margins [%] : Rn*Ic (uV) :

Center of gravity optimizer includes {"cgoptimizer.cpp", " cgoptimizer.h" and " cgoptimizer.ui"}.

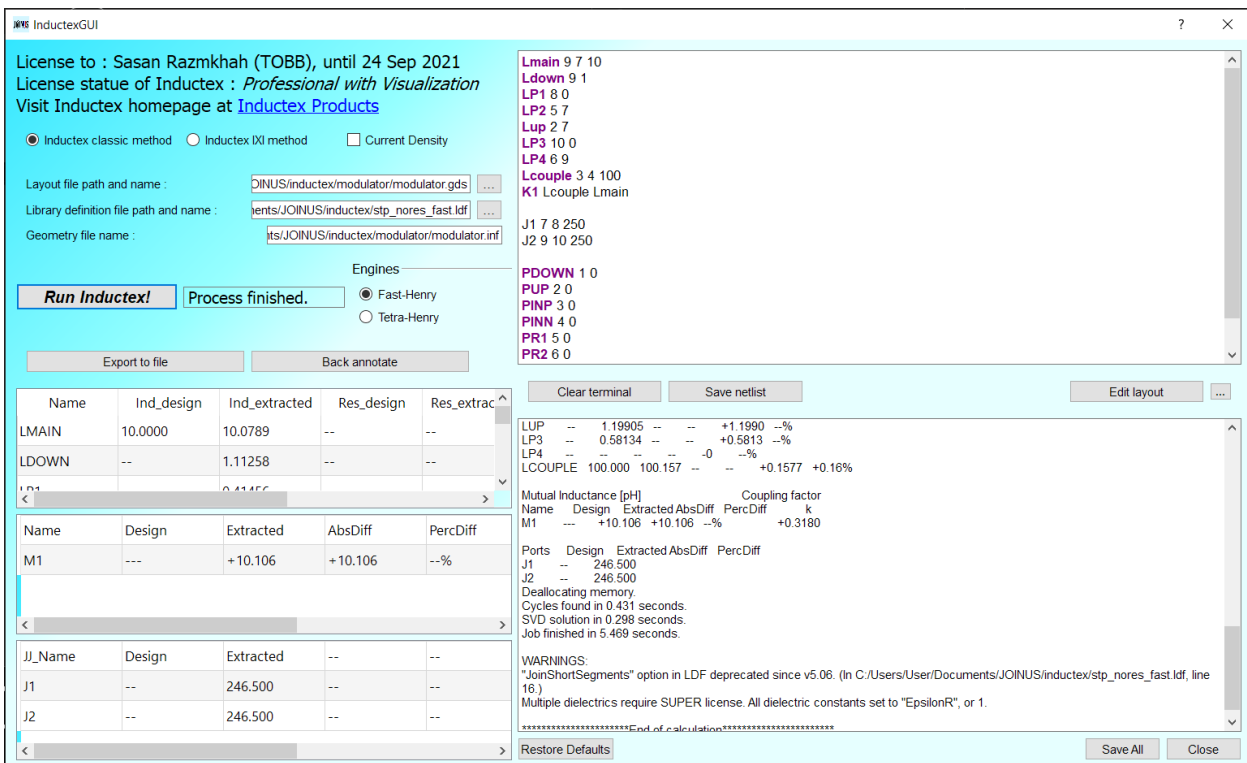


I will write examples and more explanation for them. The only issue is that Center of gravity optimizer needs a working input netlist to start the process and cannot find an operation point on its own. I will include that in future codes. The functions that are in common by all three of these classes are in “autooptim.cpp”.

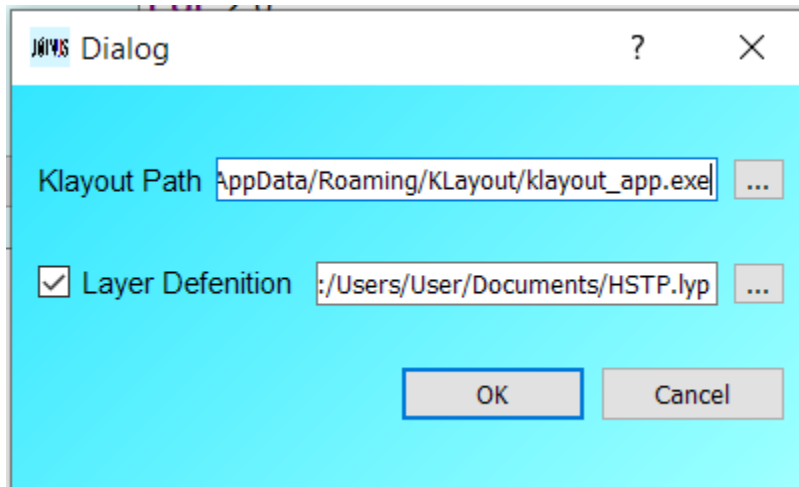
Inductex:



Inductex interface will run your current version of Inductex and will solve the netlist based on old multi-file or single IXI inductex input format.



It shows three different fields including inductances, mutual inductances, and junctions. The layout edit is now set with Klayout and program path and library can be included to show the correct layers in Klayout editor.



The Inductex is not yet function on MAC OS X.

Threading issue for MAC OS:

There was a thread issue with MAC OS X that the program created different threads for each run, and it could consume a lot of memory. The threading is now solved for MAC and optimization and simulation should run with no issue.