

Introduction to YamboPy

Michael Atambo - Technical University of Kenya RASESMA 2023

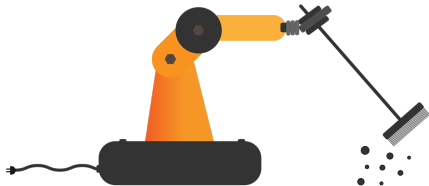


CNR
Istituito di Struttura
della Materia



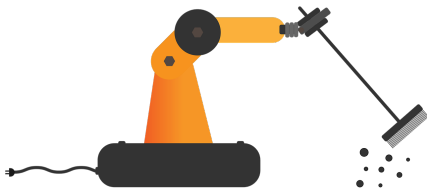
Automation in Condensed Matter: Motivation

- ▶ Modern computational science is fast paced
- ▶ **Significant** computing resources are available
- ▶ The landscape of possible materials is **huge**
 - ▶ you want magnetism? >40000 materials on materials project deposited
 - ▶ XYZ? probably something exists,



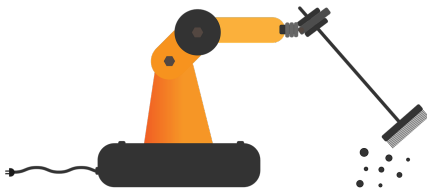
What is Automation In Our Context?

- ▶ Hey siri, *where is my phone?*
- ▶ We want to be able to compute something related to condensed matter
- ▶ Preferably with physics codes we already use,



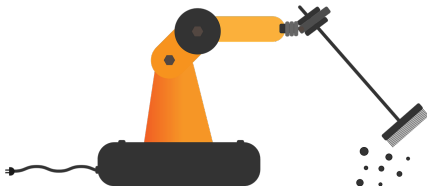
Automation in Condensed Matter

- ▶ Tools exist to provide some solutions
 - ▶ <https://workflows.community/systems>
 - ▶ [Awesome List](#)
- ▶ **Varied** in what they can do
- ▶ but,



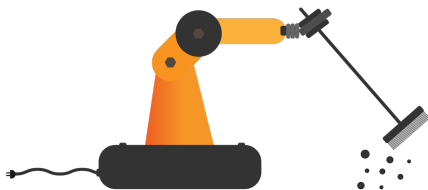
Automation in Condensed Matter

- ▶ What would be the user **needs**?
- ▶ What has to be **stored**?
- ▶ What can be **automated**?



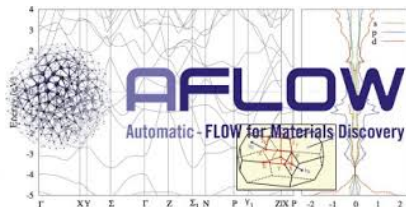
Automation in Condensed Matter

- ▶ Some Select (focused) existing tools,
- ▶ **AiiDA** (aiida.net)
- ▶ **AFLOW** (<https://www.aflowlib.org/>)
- ▶ **Atomate**, *Comput. Mater. Sci.* 139, 140–152 (2017).
- ▶ **Express** (<https://github.com/MineralsCloud/Express.jl>)



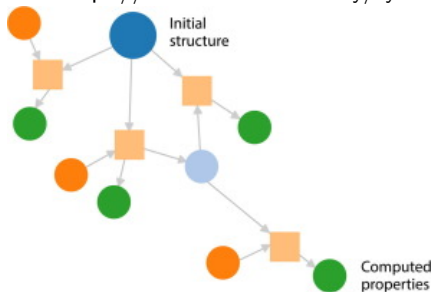
Aiida and Aflow:

- ▶ Large and complex,
- ▶ Aims
 - ▶ track data as its generated
 - ▶ track provenance
 - ▶ provide querying interface
 - ▶ scripting (automation),
 - ▶ **workflows**
 - ▶ reproducibility. . .



Workflows

- ▶ These provide some means to encode logic,
- ▶ This is a big idea:
 - ▶ We can encode the users scientific knowledge!
- ▶ Engines:
 - ▶ <https://workflows.community/systems>



Automation and Scripting

- ▶ Reducing our focus:
 - ▶ first step is scripting
 - ▶ we wont get to workflows today, or AiiDA
- ▶ Yambo is a focused tool that can address what we want to do:
 - ▶ automate/script yambo calculation
 - ▶ analyze the results of a calculation



Yambopy

- ▶ This project provides python scripts to manage and postprocess quantum espresso and yambo calculation.
- ▶ Can automate your calculations,
- ▶ provide post-processing capabilities
- ▶ Visualization
- ▶ Input file manipulation, etc.
- ▶ Its FOSS! <https://github.com/yambo-code/yambopy>

Search or jump to... Pull requests Issues Codespaces Marketplace Explore

yambo-code / yambopy Public
forked from alexmorata/yambopy

Watch 2 Fork 15 Star 8

<> Code Pull requests 1 Actions Projects Security Insights

master 11 branches 3 tags Go to file Add file Code + About

This branch is 80 commits ahead of alexmorata:master. 1 #1

Fulvio Paleari Updated YamboWFB. SOC supported in reading wfcs e9c7e29 on Nov 26, 2022 1,252 commits

command_line	MergeQP now works also in spin-polarised case	8 months ago
docs	first draft of docs	4 years ago
oeov	Fixed missing time reversal in symmetry expansion of δw_{xx}	8 months ago

Readme BSD-3-Clause license 8 stars 2 watching 15 forks

- ▶ From: Alejandro Molina Sanchez Fulvio Paleari. . .

Yambopy Features

- ▶ Create Yambo **Inputs**
- ▶ Run **Yambo**
- ▶ Post processing,
 - ▶ Yambo output
 - ▶ Netcdf examination
 - ▶ plotting
- ▶ QE **Inputs**
 - ▶ Phonons (ph.x)
 - ▶ Dynmat (dynmat.x)
 - ▶ PW (pw.x)
- ▶ Interacting with batch queues (PBS)

Yambopy Installation

- ▶ The package is available on pypi.

```
1 pip install yambopy
```

- ▶ can be installed from source also:

```
1 git clone https://github.com/yambo-code/yambopy.git
2 cd yambopy
3 python setup.py install
```

- ▶ requirements:

- ▶ Yambo
- ▶ numpy
- ▶ scipy
- ▶ matplotlib
- ▶ **matplotlib***
- ▶ QE...

How Yambopy works:

- ▶ Yambopy is written in python,
- ▶ The important stuff is all encapsulated in one class each
 - ▶ A class for QE input creation
 - ▶ ...
 - ▶ in general a class for every step

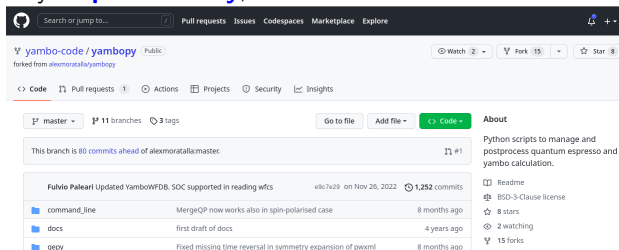
```
1 from qepy import PwIn
2 qe = PwIn()
3 qe.atoms = [['Si', [0.125, 0.125, 0.125]],
4             ['Si', [-.125, -.125, -.125]]]
5 qe.atypes = {'Si': [28.086, "Si.pbe-mt_fhi.UPF"]}
6 ...
7 qe.system['ntyp'] = 1
8 qe.system['ibrav'] = 2
9 qe.write('si.scf')
```

How Yambopy works:

- ▶ Other Classes that encode some functionality
 - ▶ **schedulerypy**
 - ▶ job execution (interaction with PBS)
 - ▶ **yambopy**
 - ▶ read/write input files and read outputs from yambo
 - ▶ **PWIn**
 - ▶ read/write QE input files, pw.x
 - ▶ **PhIn**
 - ▶ read/write QE input files, ph.x
 - ▶ **DynmatIn**
 - ▶ read/write QE input files, dynmat.x
 - ▶ **YamboIn**
 - ▶ read/write Yambo input files
- ▶ Commandline executables
 - ▶ **yambopy** scripts that can provide features of yambopy from the command line

What will you get out of the exercise?

- ▶ Automation reduces **mistakes**.
- ▶ Increase the **pace** of your work.
- ▶ **High Throughtput** studies.
- ▶ Key: **reproducibility**, can we achieve this?



The screenshot shows the GitHub interface for the repository `yambo-code / yambopy`. At the top, there is a search bar and navigation links for Pull requests, Issues, Codespaces, Marketplace, and Explore. The repository name is followed by 'Public' and statistics: Watch (2), Fork (15), and Star (8). Below this, there are tabs for Code, Pull requests (1), Actions, Projects, Security, and Insights. The main content area shows the current branch as 'master' with 11 branches and 3 tags. A message indicates 'This branch is 80 commits ahead of alexmorata:master'. A commit by Fulvio Paleari is highlighted, titled 'Updated YamboWFB: SOC supported in reading wfcs', dated Nov 26, 2022, with 1,252 commits. Below the commit, a table lists files: `command_line` (MergeQP now works also in spin-polarised case, 8 months ago), `docs` (first draft of docs, 4 years ago), and `oev` (Fixed missing time reversal in symmetry expansion of `oewml`, 8 months ago). On the right, the 'About' section describes the repository as 'Python scripts to manage and postprocess quantum espresso and yambo calculation.' and lists metadata: Readme, BSD-3-Clause license, 8 stars, 2 watching, and 15 forks.

Search or jump to... Pull requests Issues Codespaces Marketplace Explore

yambo-code / yambopy Public Watch 2 Fork 15 Star 8

Code Pull requests 1 Actions Projects Security Insights

master 11 branches 3 tags Go to file Add file Code About

This branch is 80 commits ahead of alexmorata:master #1

Fulvio Paleari Updated YamboWFB: SOC supported in reading wfcs e9c7a29 on Nov 26, 2022 1,252 commits

command_line	MergeQP now works also in spin-polarised case	8 months ago
docs	first draft of docs	4 years ago
oev	Fixed missing time reversal in symmetry expansion of <code>oewml</code>	8 months ago

Readme BSD-3-Clause license 8 stars 2 watching 15 forks

Requirements

- ▶ These python scripts require **python3**
- ▶ We will need:
 1. **yambopy**, Already Installed
 2. **yambo**, Installed
 3. **Quantum Espresso**, installed



Aims of this Exercise:

- ▶ We will work on the Convergences and Approximations tutorial: [click here](#)
- ▶ And the database tutorial: [click here](#)
- ▶ Finally the BSE exercise: [Click here](#)



Retreiving the files:

- ▶ Copy the files to your home directory and work from there:

```
1 ~$ cp -r /media/ictpuser/smr3694/ictptutor/yambopy/tutorial ~/
  tutorial
2 ~$ cd ~/tutorial
```



Getting started

- ▶ You have to use spack to get the environment that provides yambopy:

```
1 ~$ spack load quantum-espresso  
2 ~$ spack load yambo  
3 ~$ spack load anaconda3
```



Running yambopy:

- ▶ First exercise,

```
1 ~$ cd ~tutorial/run_calculations
2 ~$ python gs_bn.py -sn
3 ...
```

- ▶ Follow the steps here: [click this](#)
- ▶ You will run QE followed by GW convergence



Analysis:

- ▶ Yambopy provides the capability to read and analyze the yambo NetCDF databases, for this exercise, start from this directory:

```
1 ~$ cd ~tutorial/databases_yambopy
2 ~$ yambopy save -nscf BSE_saves/QE_saves/hBN.save
3 ...
```

- ▶ Continue with the tutorial: [click here](#)



BSE with Yambo

- ▶ This requires the first exercise to have been done
- ▶ We continue from the first exercise's directory:

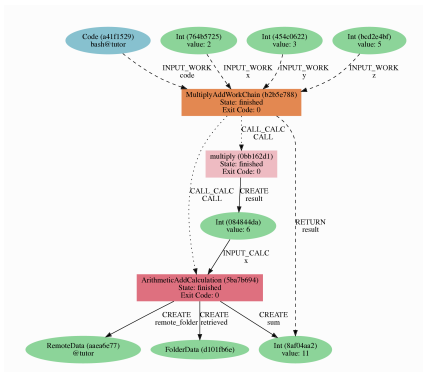
```
1 ~$ cd ~tutorial/run_calculations
2 ~$
3 ...
```

- ▶ Continue with the instructions here: [click this](#)



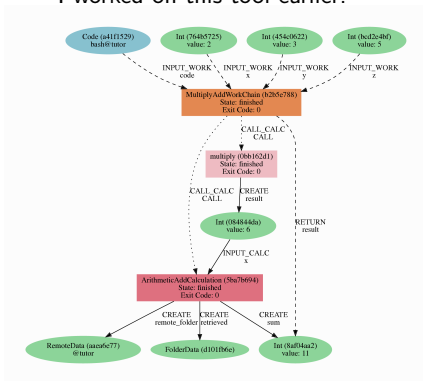
Going beyond Scripting:

- ▶ Richer Workflows
- ▶ AiiDA
- ▶ provenance
- ▶ databases
- ▶ querying
- ▶ HTC



AiiDA-Yambo

- ▶ AiiDA works through plugins,
- ▶ the core (AiiDA) provides the management engine, querying, provenance
- ▶ plugins interact with the physics codes (QE , VASP, Yambo, Siesta)
- ▶ For G0W0 you have **aiida-yambo**
(<https://github.com/yambo-code/aiida-yambo>)
 - ▶ I worked on this tool earlier.



Thanks

- ▶ Thank you for your attention.
- ▶ questions?

