

Improving the performance of Python code for NMR data processing

Typ práce / Project type	SFG / student project
Jazyk práce / Language	English
Vedoucí/ Supervisor	Dr. Petrov, Oleg
Kontakt / Contact	oleg.petrov@matfyz.cuni.cz
Pracoviště / Laboratory	Laboratoř (MFF Troja)
Klíčová slova / Keywords	NMR; signal; processing; with; Python
Časová náročnost / Workload	80 hodin

NMR signal processing includes a number of tasks which can be divided into three main categories. The first comprises repetitive tasks – signal apodization, baseline correction, phasing, peak quantitation – performed routinely after each experiment. Next are auxiliary tasks needed to be performed only occasionally: signal denoising, linear prediction, etc. The last are tasks specific for given NMR applications: special transforms, data imputation, filtering, etc. These tasks are usually implemented ‘on-board’, by software embedded in an NMR spectrometer. Yet, there is a constant demand for extended functionality and for improvement of existing solutions in terms of performance and ease of use.

Cíl projektu / Project Aim

There is a collection of Python scripts for NMR data processing [1] designed as plugins for Topspin NMR software [2]. The package includes 18 tools (still growing). It has been used in MFF Troja for over two years already. But to make it attractive for outside NMR users, we need to optimize its code. Namely, dynamic Python code has to be converted into static C code by means of a Cython compiler [3] to make it run at speeds comparable to native C. We are looking for a SFG student experienced in Python/Cython programming for this task.

Co se naučíte / What you will learn

- * **Experimental Physics:** Typical 1D and 2D NMR applications; routine data processing; new trends in processing.
- * **Coding:** Adaptation of general-purpose scientific Python packages – NumPy and SciPy – for solving NMR-related problems; two ways to “cythonize” Python.

Postup řešení projektu / Key milestones

- * To begin with, you are to optimize two stable scripts – `baseline` and `denoise`, – which suffer from time-consuming “for” loops in their code.
- * Once succeeded, you are welcome to proceed with the rest of the package, or contribute with your own data processing algorithms, within a bachelor project next year.

[1] <https://github.com/olegpetrov/xpy3-tools>

[2] <https://www.bruker.com/en/products-and-solutions/mr/nmr-software/topspin/>

[3] <https://cython.readthedocs.io/en/latest/src/tutorial/pure.html>