Optcom

pseudo software requirement document

Sacha Medaer

June 2019

1 Introduction

Optcom is a *Python* library which aims to simulate optical communication systems. This work is based mainly but not only on the two books of Agrawal. It has been inspired by *pyofss* of David Bolt, *SPIP* of Stéphane Ballac and *OptiSystem*. Here are a few objectives that *Optcom* aims to reach and which are not offered by the previously cited softwares/libraries.

- Create a non sequential and non directional optical system by assembling individual components.
- Choose the set of equations used for the simulation as well as the solver type.
- Be a Free and Open Source Software (FOSS) with a community (hopefully) and an exhaustive documentation in order for users to build their own components.

2 Proposition of Otpcom requirements for Master thesis

The **bold** bullet point are not implemented yet.

- Components:
  - Pulses
    - Soliton
    - Gaussian
    - Hyperbolic secant
    - Super Gaussian
    - **Laser** (which ones?)
  - Single Mobe Fiber with:
    - Attenuation
    - Dispersion
    - SPM
    - XPM
    - **FWM**
  - Ideal Phase Modulation
  - Ideal Divider
  - Ideal Combiner
  - Ideal Coupler

  - Coupler with (and thus Divider and Combiner):
    * Fiber para.
    * Taylor series expansion of the coupling coefficient
    * 2 by 2 fibers assymetries

    (as they use the same coupled mode equations, it could be easily extended to multi cores fiber)
  - Ideal Mach-Zehnder Modulator
  - **Mach-Zehnder Modulator** with:
    * **bias voltage**
    * **modulation voltage**
    * Coupler para.
  - Ideal Amplifier
  - **Fiber Amplifier** with:
    * **rate equations**
    * **gain saturation**
    * Fiber para.

- Schrodinger equations:
  - GNLSE
  - NLSE
  - approx NLSE (3 main different types)

- Solver:
  - Standard DE/PDE solvers
  - SSFM
  - Symmetric SSFM
  - Reduced SSFM
  - Truncated SSFM
  - **Embedded SSFM**
  - RK4IP
  - RK4IP optimized for GNLSE
  - **Embedded RK4IP**

- Sofware utilities
  - plotting tools
    * plotting in temporal domain, spectral domain, phase, ...
    * waterfall plot, 3D plot, multi plots, animated plot, ...

- System (called layout in Optcom):
  - layout as modified graph structured (be able to use graph algorithms)
    * loop management
    * co-propagation management
    * multi sources management

    - multi signal types (electrical / optical)
  - multi fields and multi channels (per field) propagation (N.B.: all the components, equations, solvers, ... have been implemented to work with multi fields/channels signal, however the theoretical background is still to be proven for more than 2 of them)
  - **save / load fields**
- **Software Documentation** (will probably be the biggest single work at that point)
- Software code related:
  - conventional style guide (PEP 8)
  - conventional docstring (PEP 257)
  - **github/gitlab repositery with python package installer, wiki, ...**
  - **tutorials**
  - **multi-threading** (maybe GPU acceleration?)