taurex-emcee Manual v0.4.0

Andrea Bocchieri, Quentin Changeat, Lorenzo V. Mugnai, Enzo Pascale

Created on: January, 2024

Last updated: 28th January, 2024

Table of contents

List of figures

List of tables

This guide will walk you through the `taurex-emcee` code.

> **Warning:** This documentation is not completed yet. If you find any issue or difficulty, please contact the developers for help.

Important: A dedicated paper has been submitted and the related information will be published soon.

> **Caution:** In case of inconsistency between the documentation and the paper, always assume that the paper is correct.

Hint: Please note that `taurex-emcee` does not implement an automatic updating system. Be always sure that you are using the most updated version by monitoring GitHub.

Want to install it? Head here: *Installation*

Want to jump into the `taurex-emcee` program? Head here: *User guide*

Want to know more about the code? Head here: *API guide*

Want to collaborate? Head here: *Developer guide*

Curious about the license? Head here: *License*

Curious about the project history? Head here: Changelog

Chapter 1

User guide

This guide covers the general installation and use of `taurex-emcee`.

1.1 Introduction

taurex-emcee is a plugin for the TauREx3.1 atmospheric retrieval framework that extends the choice of sampling methods available to the user. The plugin provides an interface to the emcee sampler, a popular affine-invariant ensemble sampler widely used in the astronomy community. Running the sampler to convergence is automated through the autoemcee package, which also supports MPI parallelization. Thus, the *taurex-emcee* plugin allows users to easily launch parallelized retrievals of atmospheric spectra with *emcee*. This enables reliable, efficient, and fast retrievals.

1.2 Installation

1.2.1 Install with pip

The `taurex-emcee` package is hosted on PyPI repository. You can install it by

```
pip install taurex_emcee
```

1.2.2 Install from git

You can clone `taurex-emcee` from our main git repository

```
git clone https://github.com/ExObsSim/taurex-emcee.git
```

Move into the `taurex-emcee` folder

```
cd /your_path/taurex-emcee
```

Then, just do

```
pip install .
```

To test for correct setup you can do

```
python -c "import taurex_emcee"
```

You can verify if the plugin is functioning by seeing if TauREx successfully detects `taurex-emcee`.

```
taurex --plugins
```

If there are no errors then the installation was successful!

1.2.3 Uninstall `taurex-emcee`

`taurex-emcee` is installed in your system as a standard python package: you can uninstall it from your Environment as

```
pip uninstall taurex_emcee
```

1.2.4 Update `taurex-emcee`

If you have installed `taurex-emcee` using Pip, now you can update the package simply as

```
pip install taurex_emcee --upgrade
```

If you have installed `taurex-emcee` from GitHub, you can download or pull a newer version of `taurex-emcee` over the old one, replacing all modified data.

Then you have to place yourself inside the installation directory with the console

```
cd /your_path/taurex-emcee
```

Now you can update `taurex_-emcee` simply as

```
pip install . --upgrade
```

or simply

```
pip install .
```

1.2.5 Modify `taurex-emcee`

You can modify `taurex-emcee` main code, editing as you prefer, but in order to make the changes effective

```
pip install . --upgrade
```

or simply

```
pip install .
```

To produce new `taurex-emcee` functionalities and contribute to the code, please see *Developer guide*.

1.3 Overview

The `taurex-emcee` plugin enables the use of the `emcee` sampler in TauREx3.1 retrievals.

To make use of the plugin, you can simply replace your existing optimizer with the `emcee` sampler.

For example if we have a multinest optimizer defined as:

```
[Optimizer]
optimizer = multinest
multi_nest_path = ./multinest
```

The `taurex-emcee` enabled version is simply:

```
[Optimizer]
optimizer = emcee
```

The `emcee` sampler has a number of parameters that can be set. These are described in the table below.

Table 1.1 – Parameters for the `emcee` sampler.

Name	Description	Default
num_global_samples	Number of samples to initially draw from the prior	10000
num_chains	Number of independent ensembles to run	4
num_walkers	Ensemble size	max(100, 4 * ndim)
max_ncalls	Maximum number of likelihood function evaluations	1000000
growth_factor	Factor by which to increase the number of steps	10
max_improvement_loops	Number of times MCMC should be re-attempted	4
num_initial_steps	Number of sampler steps to take in first iteration	100
min_autocorr_times	If > 0, sets autocorelation criterion to converge	0
rhat_max	Sets Gelman-Rubin diagnostic to converge	1.01
geweke_max	Sets Geweke diagnostic to converge	2.0
progress	If True, show progress bars	True

Tip: Find detailed information on convergence criteria at Introduction to Bayesian Analysis Procedures.

1.4 Using with MPI

`taurex-emcee` can be used with MPI to perform retrievals using multiple CPU or GPU cores. The latter assumes you have a CUDA enabled GPU and have installed the taurex_cuda package.

To use `taurex-emcee` with MPI, you need to install the `mpi4py` package. This can be done using `pip`:

```
pip install mpi4py
```

Once installed, you can run `taurex-emcee` using MPI by using the `mpirun` command:

```
mpirun -n 4 taurex -i parfile.par -R -o retrieval.h5
```

where the `-n` flag specifies the number of cores to use.

If interested, you can also use multiple GPUs. Please refer to taurex_cuda for more information.

1.5 Quick start

This section of the documentation deals with setting up a basic notebook that uses TauREx3.1 and the *taurex-emcee* plugin to perform a retrieval.

1.5.1 Setup

First, let's import the packages we need.

[1]:
```
import numpy as np
import matplotlib.pyplot as plt
import corner
from taurex.log.logger import root_logger
%matplotlib inline
```

1.5.2 Forward model

Then, we'll load the input file and generate a suitable forward model.

For the input file, we'll use the "parfile-gpu.par" file that is included in the examples directory of this package.

[2]:
```
from taurex.parameter import ParameterParser

pp = ParameterParser()

# Parse the input file
input_file = "parfile-gpu.par"
pp.read(input_file)

# Setup global parameters
pp.setup_globals()

# Get the spectrum
observation = pp.generate_observation()

binning = pp.generate_binning()

# Generate a model from the input
model = pp.generate_appropriate_model(obs=observation)

# build the model
model.build()
```

```
taurex.ParamParser - INFO - Interpolation mode set to linear
taurex.ParamParser - WARNING - Xsecs will be loaded in memory
taurex.ParamParser - WARNING - Radis is disabled
taurex.ParamParser - WARNING - Radis default grid will be used
taurex.ClassFactory - INFO - Reloading all modules and plugins
taurex.ClassFactory - INFO - ----------Plugin loading---------
```

(continues on next page)

(continued from previous page)

```
taurex.ClassFactory - INFO - Loading ace
taurex.ClassFactory - INFO - Loading cuda
taurex.ClassFactory - INFO - Loading curvefit
taurex.ClassFactory - INFO - Loading emcee
taurex.ClassFactory - INFO - Loading petitrad
taurex.ClassFactory - INFO - Loading ultranest
taurex.TransmissionCudaModel - INFO - No pressure profile defined, using simple␣
↪pressure profile with
taurex.TransmissionCudaModel - INFO - parameters nlayers: 100, atm_pressure_range=(0.
↪0001,1000000.0)
taurex.TransmissionCudaModel - INFO - Building model...
taurex.TransmissionCudaModel - INFO - Collecting paramters
taurex.TransmissionCudaModel - INFO - Setting up profiles
taurex.TransmissionCudaModel - INFO - Computing pressure profile
taurex.ChemistryModel - INFO - Initializing chemistry model
taurex.ChemistryModel - INFO - Initializing chemistry model
taurex.TransmissionCudaModel - INFO - Setting up contributions
taurex.OpacityCache - INFO - Reading opacity CO2
taurex.OpacityCache - INFO - Loading opacity CO2 into model
taurex.OpacityCache - INFO - Reading opacity CH4
taurex.OpacityCache - INFO - Loading opacity CH4 into model
taurex.CudaOpacityCache - INFO - Re-homogenizing native grids!
taurex.CudaCiaCache - INFO - Re-homogenizing native grids!
taurex.TransmissionCudaModel - INFO - DONE
```

Now, set up the binning and the instrument.

[3]:
```
wngrid = None

if binning == "observed" and observation is None:
    root_logger.critical(
        "Binning selected from Observation yet None provided"
    )
    quit()

if binning is None:
    if observation is None or observation == "self":
        binning = model.defaultBinner()
        wngrid = model.nativeWavenumberGrid
    else:
        binning = observation.create_binner()
        wngrid = observation.wavenumberGrid
else:
    if binning == "native":
        binning = model.defaultBinner()
        wngrid = model.nativeWavenumberGrid
    elif binning == "observed":
        binning = observation.create_binner()
        wngrid = observation.wavenumberGrid
```

(continues on next page)

(continued from previous page)

```
    else:
        binning, wngrid = binning
```

[4]:
```
instrument = pp.generate_instrument(binner=binning)

num_obs = 1
if instrument is not None:
    instrument, num_obs = instrument

if observation == "self" and instrument is None:
    root_logger.critical("Instrument nust be specified when using self option")
    raise ValueError("No instruemnt specified for self option")

inst_result = None
if instrument is not None:
    inst_result = instrument.model_noise(
        model, model_res=model.model(), num_observations=num_obs
    )
```

```
taurex.TransmissionCudaModel - INFO - Computing pressure profile
taurex.ChemistryModel - INFO - Initializing chemistry model
taurex.Absorption - INFO - Recomputing active gas CO2 opacity
taurex.CudaOpacity - INFO - Transfering xsec grid to GPU
taurex.CudaOpacity - INFO - Moving to GPU once
taurex.Absorption - INFO - Recomputing active gas CH4 opacity
taurex.CudaOpacity - INFO - Transfering xsec grid to GPU
taurex.CudaOpacity - INFO - Moving to GPU once
taurex.Absorption - INFO - Done
taurex.CIACache - INFO - Loading cia H2-He into model
taurex.CIACache - INFO - Loading cia H2-H2 into model
taurex.CIA - INFO - Done
taurex.Rayleigh - INFO - Done
```

If the observation is on “self”, we will generate an observation given the forward model and the instrument function, to be used for self-retrievals.

[5]:
```
# Observation on self
if observation == "self":
    from taurex.data.spectrum import ArraySpectrum
    from taurex.util.util import wnwidth_to_wlwidth

    inst_wngrid, inst_spectrum, inst_noise, inst_width = inst_result

    inst_wlgrid = 10000 / inst_wngrid

    inst_wlwidth = wnwidth_to_wlwidth(inst_wngrid, inst_width)
    observation = ArraySpectrum(
        np.vstack([inst_wlgrid, inst_spectrum, inst_noise, inst_wlwidth]).T
    )
    binning = observation.create_binner()
```

1.5.3 Retrieval

Now we can set up the retrieval, starting with the optimizer and pointing it to the observation and forward model.

[6]:
```
optimizer = None
solution = None

import time

if observation is None:
    root_logger.critical("No spectrum is defined!!")
    quit()

optimizer = pp.generate_optimizer()
optimizer.set_model(model)
optimizer.set_observed(observation)
pp.setup_optimizer(optimizer)
```

```
taurex.ParamParser - INFO - Setting up optimizer
taurex.ParamParser - INFO - Fitting: planet_radius
taurex.ParamParser - INFO - Fitting: T
taurex.ParamParser - INFO - Fitting: CO2
taurex.ParamParser - INFO - Fitting: CH4
```

[7]:
```
start_time = time.time()
solution = optimizer.fit()

end_time = time.time()

root_logger.info("Total Retrieval finish in %s seconds", end_time - start_time)

for _, optimized, _, _ in optimizer.get_solution():
    optimizer.update_model(optimized)
    break

result = model.model()
```

```
taurex.Emcee - INFO - Initializing parameters
taurex.Emcee - INFO - -------FITTING---------------
taurex.Emcee - INFO - Parameters to be fit:
taurex.Emcee - INFO - planet_radius: Value: 0.5 Type:Uniform Params:Bounds = [0.45,0.
↪55]
taurex.Emcee - INFO - T: Value: 1000.0 Type:Uniform Params:Bounds = [900.0,1100.0]
taurex.Emcee - INFO - CO2: Value: 0.0001 Type:LogUniform Params:Bounds = [-12.0,-1.0]
taurex.Emcee - INFO - CH4: Value: 0.0001 Type:LogUniform Params:Bounds = [-12.0,-1.0]
taurex.Emcee - INFO -
taurex.Emcee - INFO - ------------------------------------
taurex.Emcee - INFO - ------Retrieval Parameters-----------
taurex.Emcee - INFO - ------------------------------------
taurex.Emcee - INFO -
taurex.Emcee - INFO - Dimensionality of fit: 4
```

(continues on next page)

(continued from previous page)

```
taurex.Emcee - INFO -
taurex.Emcee - INFO -
Param            Value  Type        Args
-------------  -------  ----------  ----------------------
planet_radius      0.5  Uniform     Bounds = [0.45,0.55]
T               1000    Uniform     Bounds = [900.0,1100.0]
log_CO2           -4    LogUniform  Bounds = [-12.0,-1.0]
log_CH4           -4    LogUniform  Bounds = [-12.0,-1.0]


taurex.Emcee - INFO -
```

```
[autoemcee] finding starting points and running initial 100 MCMC steps
```

```
100
↪%|
↪100/100 [00:41<00:00,  2.39it/s]
100
↪%|
↪100/100 [00:38<00:00,  2.61it/s]
```

```
[autoemcee] rhat chain diagnostic: [1.29948991 1.35005165 1.2648493  1.25088209] (<1.
↪010 is good)
[autoemcee] not converged yet at iteration 1 after 1816 evals
[autoemcee] Running 200 MCMC steps ...
[autoemcee] Starting points chosen: {5}, L=-2656.1
[autoemcee] Starting at [0.50155868 0.29951435 0.73175664 0.7295726 ] +- [2.07316967e-
↪05 1.95106273e-03 2.15082485e-04 1.41191717e-04]
```

```
100
↪%|
↪100/100 [00:44<00:00,  2.27it/s]
100
↪%|
↪200/200 [01:28<00:00,  2.27it/s]
```

```
[autoemcee] Starting points chosen: {4}, L=-2656.1
[autoemcee] Starting at [0.49896974 0.95062315 0.68651548 0.70419042] +- [0.00024095 0.
↪01206181 0.0017155  0.00262025]
```

```
100
↪%|
↪100/100 [00:43<00:00,  2.28it/s]
100
↪%|
↪200/200 [01:27<00:00,  2.28it/s]
```

```
[autoemcee] Used 4832 calls in last MCMC run
[autoemcee] rhat chain diagnostic: [1.03861042 1.11989315 1.14887417 1.12795048] (<1.
↪010 is good)
[autoemcee] not converged yet at iteration 2 after 6648 evals
```

(continues on next page)

(continued from previous page)

```
[autoemcee] Running 400 MCMC steps ...
[autoemcee] Starting points chosen: {2}, L=-2656.1
[autoemcee] Starting at [0.50044448 0.4670975  0.72607207 0.72546587] +- [1.44640724e-
↪05 9.45368002e-04 1.16133036e-04 1.45492841e-05]


100
↪%|
↪200/200 [01:28<00:00,  2.27it/s]
100
↪%|
↪400/400 [02:53<00:00,  2.30it/s]
[autoemcee] Starting points chosen: {7}, L=-2656.1
[autoemcee] Starting at [0.49992209 0.53978566 0.72218774 0.72585682] +- [1.04131342e-
↪05 1.25879893e-03 1.65853827e-04 8.47646736e-05]


100
↪%|
↪200/200 [01:26<00:00,  2.30it/s]
100
↪%|
↪400/400 [02:53<00:00,  2.30it/s]
[autoemcee] Used 9632 calls in last MCMC run
[autoemcee] rhat chain diagnostic: [1.00378234 1.02887037 1.02722475 1.0225528 ] (<1.
↪010 is good)
[autoemcee] not converged yet at iteration 3 after 16280 evals
[autoemcee] Running 800 MCMC steps ...
[autoemcee] Starting points chosen: {4}, L=-2656.1
[autoemcee] Starting at [0.50017916 0.37760545 0.7399924  0.7328286 ] +- [7.93067988e-
↪06 1.34987117e-03 3.00956367e-04 1.59212525e-04]


100
↪%|
↪400/400 [02:53<00:00,  2.30it/s]
100
↪%|
↪800/800 [05:49<00:00,  2.29it/s]
[autoemcee] Starting points chosen: {4}, L=-2656.1
[autoemcee] Starting at [0.50077536 0.47452216 0.71904538 0.72308628] +- [1.58826252e-
↪05 1.58216012e-03 2.94652435e-04 1.17993914e-04]


100
↪%|
↪400/400 [02:56<00:00,  2.27it/s]
100
↪%|
↪800/800 [05:48<00:00,  2.29it/s]
```

```
[autoemcee] Used 19232 calls in last MCMC run
[autoemcee] rhat chain diagnostic: [1.00320846 1.00373857 1.00746311 1.00648991] (<1.
↪010 is good)
[autoemcee] converged!!!
```

```
taurex.Emcee - INFO - Sampling time 1935.527314901352 s
taurex.Emcee - INFO - Post-processing - Generating spectra and profiles
taurex.Emcee - INFO - Computing solution 0
taurex.TransmissionCudaModel - INFO - Computing pressure profile
taurex.ChemistryModel - INFO - Initializing chemistry model
taurex.Absorption - INFO - Recomputing active gas CO2 opacity
taurex.Absorption - INFO - Recomputing active gas CH4 opacity
taurex.Absorption - INFO - Done
taurex.CIA - INFO - Done
taurex.Rayleigh - INFO - Done
taurex.TransmissionCudaModel - INFO - Computing pressure profile
taurex.ChemistryModel - INFO - Initializing chemistry model
taurex.Absorption - INFO - Recomputing active gas CO2 opacity
taurex.Absorption - INFO - Recomputing active gas CH4 opacity
taurex.Absorption - INFO - Done
taurex.CIA - INFO - Done
taurex.Rayleigh - INFO - Done
taurex.TransmissionCudaModel - INFO - Computing pressure profile
taurex.ChemistryModel - INFO - Initializing chemistry model
taurex.TransmissionCudaModel - INFO - Modelling each contribution...
taurex.Absorption - INFO - Recomputing active gas CO2 opacity
taurex.TransmissionCudaModel - INFO -   Absorption---CO2 contribtuion
taurex.Absorption - INFO - Recomputing active gas CH4 opacity
taurex.TransmissionCudaModel - INFO -   Absorption---CH4 contribtuion
taurex.TransmissionCudaModel - INFO -   CIA---H2-He contribtuion
taurex.TransmissionCudaModel - INFO -   CIA---H2-H2 contribtuion
taurex.TransmissionCudaModel - INFO -   Rayleigh---CO2 contribtuion
taurex.TransmissionCudaModel - INFO -   Rayleigh---CH4 contribtuion
taurex.TransmissionCudaModel - INFO -   Rayleigh---H2 contribtuion
taurex.TransmissionCudaModel - INFO -   Rayleigh---He contribtuion
taurex.TransmissionCudaModel - INFO - Computing pressure profile
taurex.ChemistryModel - INFO - Initializing chemistry model
taurex.Absorption - INFO - Recomputing active gas CO2 opacity
taurex.Absorption - INFO - Recomputing active gas CH4 opacity
taurex.Absorption - INFO - Done
taurex.CIA - INFO - Done
taurex.Rayleigh - INFO - Done
taurex.Emcee - INFO - ------------Variance generation step-----------------
taurex.Emcee - INFO - We are sampling 1280 points for the profiles
taurex.Emcee - INFO - I will only iterate through partitioned 1280 points (the rest␣
↪is in parallel)
taurex.Emcee - INFO - Progress 0.78125%
taurex.Emcee - INFO - Progress 1.5625%
taurex.Emcee - INFO - Progress 2.34375%
```

(continues on next page)

(continued from previous page)

```
taurex.Emcee - INFO - Progress 3.125%
taurex.Emcee - INFO - Progress 3.90625%
taurex.Emcee - INFO - Progress 4.6875%
taurex.Emcee - INFO - Progress 5.46875%
taurex.Emcee - INFO - Progress 6.25%
taurex.Emcee - INFO - Progress 7.03125%
taurex.Emcee - INFO - Progress 7.8125%
taurex.Emcee - INFO - Progress 8.59375%
taurex.Emcee - INFO - Progress 9.375%
taurex.Emcee - INFO - Progress 10.15625%
taurex.Emcee - INFO - Progress 10.9375%
taurex.Emcee - INFO - Progress 11.71875%
taurex.Emcee - INFO - Progress 12.5%
taurex.Emcee - INFO - Progress 13.28125%
taurex.Emcee - INFO - Progress 14.0625%
taurex.Emcee - INFO - Progress 14.84375%
taurex.Emcee - INFO - Progress 15.625%
taurex.Emcee - INFO - Progress 16.40625%
taurex.Emcee - INFO - Progress 17.1875%
taurex.Emcee - INFO - Progress 17.96875%
taurex.Emcee - INFO - Progress 18.75%
taurex.Emcee - INFO - Progress 19.53125%
taurex.Emcee - INFO - Progress 20.3125%
taurex.Emcee - INFO - Progress 21.09375%
taurex.Emcee - INFO - Progress 21.875%
taurex.Emcee - INFO - Progress 22.65625%
taurex.Emcee - INFO - Progress 23.4375%
taurex.Emcee - INFO - Progress 24.21875%
taurex.Emcee - INFO - Progress 25.0%
taurex.Emcee - INFO - Progress 25.78125%
taurex.Emcee - INFO - Progress 26.5625%
taurex.Emcee - INFO - Progress 27.34375%
taurex.Emcee - INFO - Progress 28.125%
taurex.Emcee - INFO - Progress 28.90625%
taurex.Emcee - INFO - Progress 29.6875%
taurex.Emcee - INFO - Progress 30.46875%
taurex.Emcee - INFO - Progress 31.25%
taurex.Emcee - INFO - Progress 32.03125%
taurex.Emcee - INFO - Progress 32.8125%
taurex.Emcee - INFO - Progress 33.59375%
taurex.Emcee - INFO - Progress 34.375%
taurex.Emcee - INFO - Progress 35.15625%
taurex.Emcee - INFO - Progress 35.9375%
taurex.Emcee - INFO - Progress 36.71875%
taurex.Emcee - INFO - Progress 37.5%
taurex.Emcee - INFO - Progress 38.28125%
taurex.Emcee - INFO - Progress 39.0625%
taurex.Emcee - INFO - Progress 39.84375%
taurex.Emcee - INFO - Progress 40.625%
```

(continues on next page)

(continued from previous page)

```
taurex.Emcee - INFO - Progress 41.40625%
taurex.Emcee - INFO - Progress 42.1875%
taurex.Emcee - INFO - Progress 42.96875%
taurex.Emcee - INFO - Progress 43.75%
taurex.Emcee - INFO - Progress 44.53125%
taurex.Emcee - INFO - Progress 45.3125%
taurex.Emcee - INFO - Progress 46.09375%
taurex.Emcee - INFO - Progress 46.875%
taurex.Emcee - INFO - Progress 47.65625%
taurex.Emcee - INFO - Progress 48.4375%
taurex.Emcee - INFO - Progress 49.21875%
taurex.Emcee - INFO - Progress 50.0%
taurex.Emcee - INFO - Progress 50.78125%
taurex.Emcee - INFO - Progress 51.5625%
taurex.Emcee - INFO - Progress 52.34375%
taurex.Emcee - INFO - Progress 53.125%
taurex.Emcee - INFO - Progress 53.90625%
taurex.Emcee - INFO - Progress 54.6875%
taurex.Emcee - INFO - Progress 55.46875%
taurex.Emcee - INFO - Progress 56.25%
taurex.Emcee - INFO - Progress 57.03125%
taurex.Emcee - INFO - Progress 57.8125%
taurex.Emcee - INFO - Progress 58.59375%
taurex.Emcee - INFO - Progress 59.375%
taurex.Emcee - INFO - Progress 60.15625%
taurex.Emcee - INFO - Progress 60.9375%
taurex.Emcee - INFO - Progress 61.71875%
taurex.Emcee - INFO - Progress 62.5%
taurex.Emcee - INFO - Progress 63.28125%
taurex.Emcee - INFO - Progress 64.0625%
taurex.Emcee - INFO - Progress 64.84375%
taurex.Emcee - INFO - Progress 65.625%
taurex.Emcee - INFO - Progress 66.40625%
taurex.Emcee - INFO - Progress 67.1875%
taurex.Emcee - INFO - Progress 67.96875%
taurex.Emcee - INFO - Progress 68.75%
taurex.Emcee - INFO - Progress 69.53125%
taurex.Emcee - INFO - Progress 70.3125%
taurex.Emcee - INFO - Progress 71.09375%
taurex.Emcee - INFO - Progress 71.875%
taurex.Emcee - INFO - Progress 72.65625%
taurex.Emcee - INFO - Progress 73.4375%
taurex.Emcee - INFO - Progress 74.21875%
taurex.Emcee - INFO - Progress 75.0%
taurex.Emcee - INFO - Progress 75.78125%
taurex.Emcee - INFO - Progress 76.5625%
taurex.Emcee - INFO - Progress 77.34375%
taurex.Emcee - INFO - Progress 78.125%
taurex.Emcee - INFO - Progress 78.90625%
```

(continues on next page)

(continued from previous page)

```
taurex.Emcee - INFO - Progress 79.6875%
taurex.Emcee - INFO - Progress 80.46875%
taurex.Emcee - INFO - Progress 81.25%
taurex.Emcee - INFO - Progress 82.03125%
taurex.Emcee - INFO - Progress 82.8125%
taurex.Emcee - INFO - Progress 83.59375%
taurex.Emcee - INFO - Progress 84.375%
taurex.Emcee - INFO - Progress 85.15625%
taurex.Emcee - INFO - Progress 85.9375%
taurex.Emcee - INFO - Progress 86.71875%
taurex.Emcee - INFO - Progress 87.5%
taurex.Emcee - INFO - Progress 88.28125%
taurex.Emcee - INFO - Progress 89.0625%
taurex.Emcee - INFO - Progress 89.84375%
taurex.Emcee - INFO - Progress 90.625%
taurex.Emcee - INFO - Progress 91.40625%
taurex.Emcee - INFO - Progress 92.1875%
taurex.Emcee - INFO - Progress 92.96875%
taurex.Emcee - INFO - Progress 93.75%
taurex.Emcee - INFO - Progress 94.53125%
taurex.Emcee - INFO - Progress 95.3125%
taurex.Emcee - INFO - Progress 96.09375%
taurex.Emcee - INFO - Progress 96.875%
taurex.Emcee - INFO - Progress 97.65625%
taurex.Emcee - INFO - Progress 98.4375%
taurex.Emcee - INFO - Progress 99.21875%
taurex.Emcee - INFO - Computing derived parameters...
taurex.Emcee - INFO - Post-processing - Complete
taurex.Emcee - INFO -
taurex.Emcee - INFO - ------------------------------------
taurex.Emcee - INFO - ------Final results------------------
taurex.Emcee - INFO - ------------------------------------
taurex.Emcee - INFO -
taurex.Emcee - INFO - Dimensionality of fit: 4
taurex.Emcee - INFO -
taurex.Emcee - INFO -
---Solution 0------
taurex.Emcee - INFO -
Param                  MAP       Median
-------------  -----------  -----------
planet_radius     0.499972     0.500018
T              1002.76      1001.28
log_CO2          -3.98031     -4.02948
log_CH4          -3.96889     -4.02388


taurex - INFO - Total Retrieval finish in 2023.2468461990356 seconds
taurex.TransmissionCudaModel - INFO - Computing pressure profile
taurex.ChemistryModel - INFO - Initializing chemistry model
```

(continues on next page)

(continued from previous page)

```
taurex.Absorption - INFO - Recomputing active gas CO2 opacity
taurex.Absorption - INFO - Recomputing active gas CH4 opacity
taurex.Absorption - INFO - Done
taurex.CIA - INFO - Done
taurex.Rayleigh - INFO - Done
```

1.5.4 Plots

Let's plot the spectrum first.

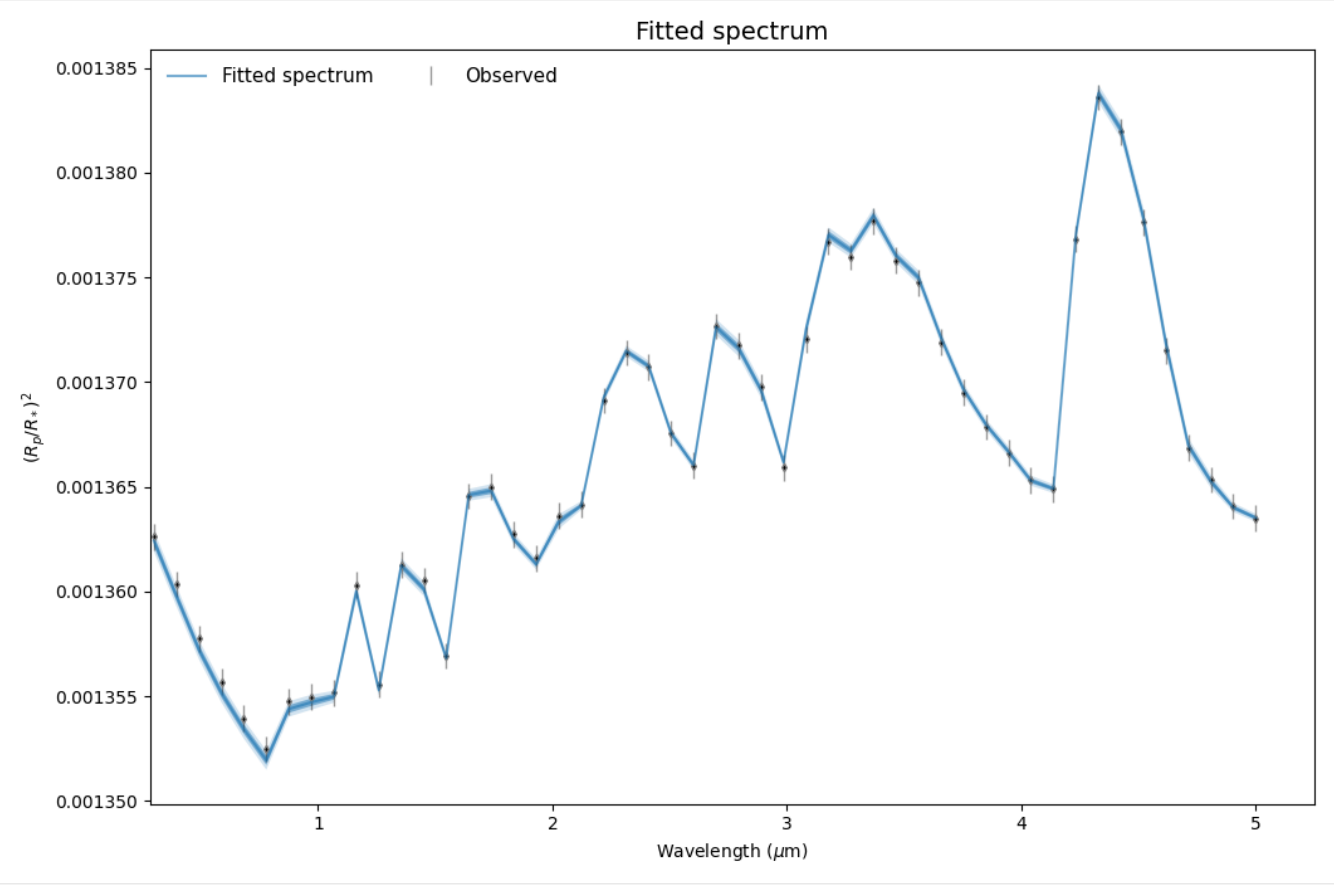
[8]:
```
modelAxis = {
    "TransmissionModel": "$(R_p/R_*)^2$",
    "TransmissionCudaModel": "$(R_p/R_*)^2$",
    "EmissionModel": "$F_p/F_*$",
    "EmissionCudaModel": "$F_p/F_*$",
    "DirectImageModel": "$F_p$",
    "DirectImageCudaModel": "$F_p$",
}
```

[9]:
```
fig = plt.figure(figsize=(10.6, 7.0))

obs_spectrum = optimizer._observed.spectrum
error = optimizer._observed.errorBar
wlgrid = optimizer._observed.wavelengthGrid

plt.errorbar(
    wlgrid,
    obs_spectrum,
    error,
    lw=1,
    color="black",
    alpha=0.4,
    ls="none",
    zorder=0,
    label="Observed",
)

for solution_idx, solution_val in solution.items():
    binned_grid = solution_val["Spectra"]["binned_wlgrid"][...]
    native_grid = solution_val["Spectra"]["native_wngrid"][...]

    plt.scatter(
        wlgrid,
        obs_spectrum,
        marker="d",
        zorder=1,
        **{"s": 10, "edgecolors": "grey", "color": "black"}
    )
```

(continues on next page)

(continued from previous page)

```
    binned_spectrum = solution_val["Spectra"]["binned_spectrum"][...]
    binned_error = solution_val["Spectra"]["binned_std"][...]

    color = "C0"
    label = "Fitted spectrum"
    plt.plot(wlgrid, binned_spectrum, label=label, color=color, alpha=0.6)
    if binned_error is not None:
        # 1 sigma
        plt.fill_between(
            wlgrid,
            binned_spectrum - binned_error,
            binned_spectrum + binned_error,
            alpha=0.5,
            zorder=-2,
            color=color,
            edgecolor="none",
        )

        # 2 sigma
        plt.fill_between(
            wlgrid,
            binned_spectrum - 2 * binned_error,
            binned_spectrum + 2 * binned_error,
            alpha=0.2,
            zorder=-3,
            color=color,
            edgecolor="none",
        )


plt.xlim(np.min(wlgrid) - 0.05 * np.min(wlgrid), np.max(wlgrid) + 0.05 * np.
↪max(wlgrid))
plt.xlabel(r"Wavelength ($\mu$m)")
plt.ylabel(modelAxis[model.__class__.__name__])

if np.max(wlgrid) - np.min(wlgrid) > 5:
    plt.xscale("log")
    plt.tick_params(axis="x", which="minor")
plt.legend(loc="best", ncol=2, frameon=False, prop={"size": 11})

plt.title("Fitted spectrum", fontsize=14)
plt.tight_layout()
plt.show()
```



Very nice!

Now let's plot the posterior distribution.

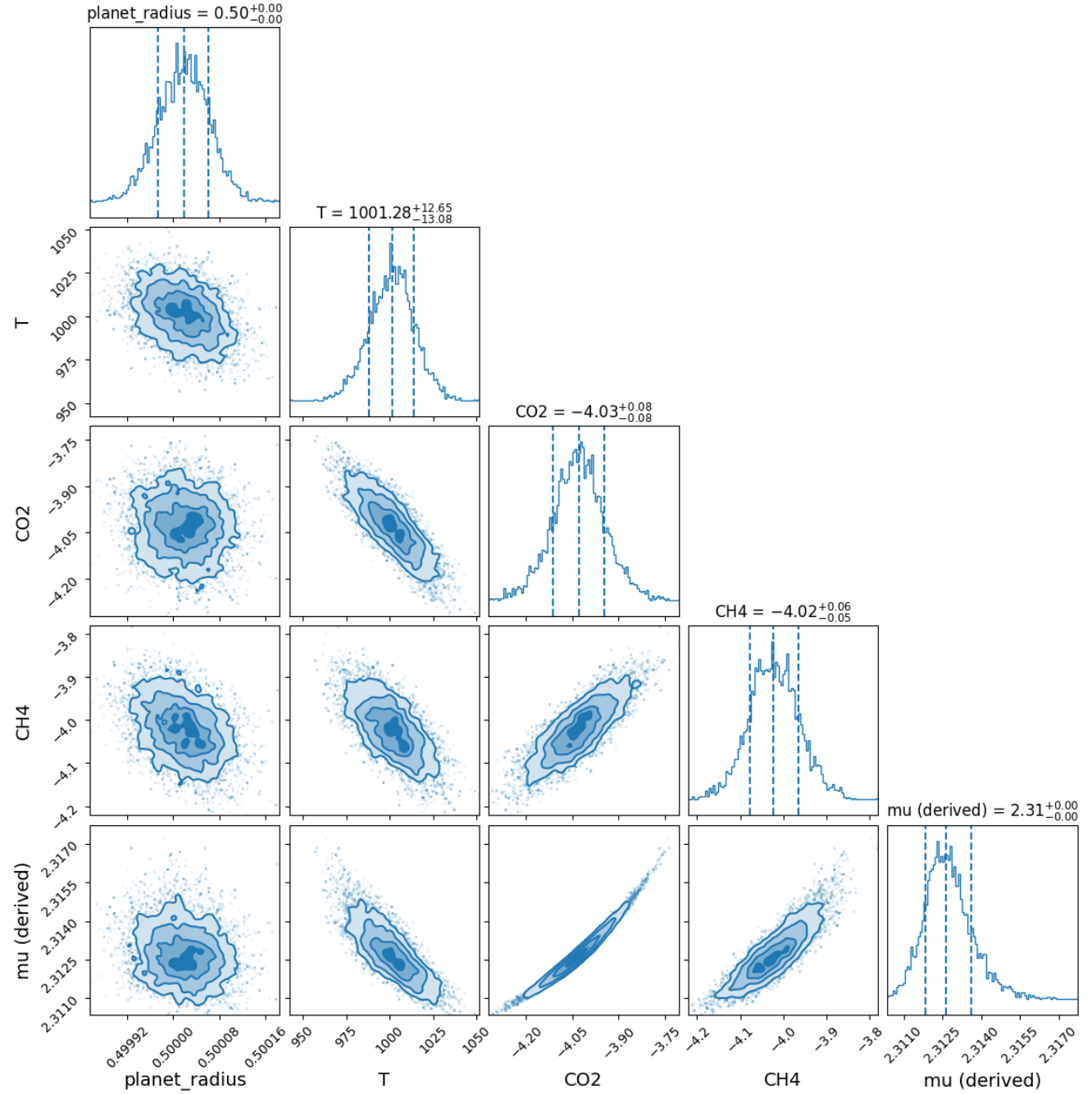
[10]:
```
def get_derived_parameters(solution):
    if 'derived_params' in solution:
        return [c for k, c in solution['derived_params'].items()]
    else:
        return [solution['fit_params']['mu_derived']]
```

[11]:
```
fittingNames = [param[0] for param in optimizer.fitting_parameters]

figs = []
fig = plt.figure(figsize=(12, 12))

for solution_idx, solution_val in solution.items():
    tracedata = solution_val["tracedata"]
    weights = solution_val["weights"]
    indices = np.array([fittingNames.index(x) for x in fittingNames])

    mu_derived = get_derived_parameters(solution_val)
    _tracedata = np.column_stack((tracedata, mu_derived[0]["trace"]))
    fittingNames.append("mu (derived)")

    figure_past = fig
```

(continues on next page)

(continued from previous page)

```
    plt.rc("xtick", labelsize=10)  # size of individual labels
    plt.rc("ytick", labelsize=10)
    plt.rc("axes.formatter", limits=(-4, 5))  # scientific notation..

    fig = corner.corner(
        _tracedata,
        weights=weights,
        labels=fittingNames,
        label_kwargs=dict(fontsize=14),
        smooth=1.5,
        scale_hist=True,
        quantiles=[0.16, 0.5, 0.84],
        show_titles=True,
        title_kwargs=dict(fontsize=12),
        # truths=truths,
        truth_color="black",
        ret=True,
        fill_contours=True,
        color=color,
        top_ticks=False,
        bins=100,
        fig=figure_past,
    )
    # fig.gca().annotate(
    #     "Posterior %s" % (solution_idx),
    #     xy=(0.5, 0.96),
    #     xycoords="figure fraction",
    #     xytext=(0, -5),
    #     textcoords="offset points",
    #     ha="center",
    #     va="top",
    #     fontsize=20,
    # )
plt.show()
```


planet_radius = 0.50$^{+0.00}_{-0.00}$
T = 1001.28$^{+12.65}_{-13.08}$
CO2 = −4.03$^{+0.08}_{-0.08}$
CH4 = −4.02$^{+0.06}_{-0.05}$
mu (derived) = 2.31$^{+0.00}_{-0.00}$
planet_radius
T
CO2
CH4
mu (derived)

Excellent!

[ ]:

Chapter 2

Developer guide

In this section we report some general guidelines for contributing to `taurex-emcee` development.

The section is inspired by the package ExoSim2.0.

2.1 Coding conventions

The `taurex-emcee` code has been developed following the PeP8 standard and the python Zen.

2.2 Documentation

Every `taurex-emcee` function or class should be documented using docstrings which follow numpydoc structure. This web page is written using the reStructuredText format, which is parsed by sphinx. If you want to contribute to this documentation, please refer to sphinx documentation first. You can improve this pages by digging into the *docs* directory in the source.

2.3 Testing

Unit-testing is very important to make sure that each code addition is tested and validated and the code never breaks. This shall be provided after *taurex-emcee v1.0.0*.

2.4 Versioning conventions

The versioning convention used (after *taurex-emcee v1.0.0*) shall be the one described in Semantic Versioning (semver) and shall be compliant to PEP440 standard. In the X.Y.Z scheme, for each modification to the previous release we increase one of the numbers.

- X
  increased only if the code in not compatible anymore with the previous version. This is considered a Major change.
- Y
  increased for minor changes. These are for the addition of new features that may change the results from previous versions. This are still hard edits, but not enough to justify the increase of an X.
- Z

the patches. This number should increase for any big fixed, or minor addition or change to the code. It won't affect the user experience in any way.

2.5 Source Control

The code is hosted on GitHub (https://github.com/ExObsSim/taurex-emcee) and structured as following.

The source has two main branches:

- `main`
  branch for stable and releases. It is the public branch and should be handled carefully.
- `develop`
  working branch where the new features are tested before they are moved to the *main* branch

2.5.1 Adding new features

New features can be added to the code following the schemes designed above.

If the contributor has writing rights to the repository, should create a new branch starting from the *develop* one. In the new *feature* branch the user should produce the new functionalities, according to the above guidelines. When the feature is ready, the branch can be merged into the official *develop* one.

To create the new feature starting from the current develop version, the contributor should run

```
$ git checkout develop
$ git checkout -b feature/<branchname>
```

The completed feature shall then be merged to the develop:

```
$ git checkout develop
$ git merge feature/<branchname>
$ git push
```

Once a feature is completed and merged, the contributor should archive the branch and remove it, to keep the repository clean. The usual procedure is:

```
$ git tag archive/<branchname> feature/<branchname>
$ git push --tags
$ git branch -d feature/<branchname>
```

Remember to delete the branch also from the remote repository. If needed, the feature branch can be restored as

```
$ git checkout -b <branchname> archive/<branchname>
```

If the contributor does not have writing rights to the repository, should use the Fork-and-Pull model. The contributor should fork the main repository and clone it. Then the new features can be implemented. When the code is ready, a pull request can be raised.

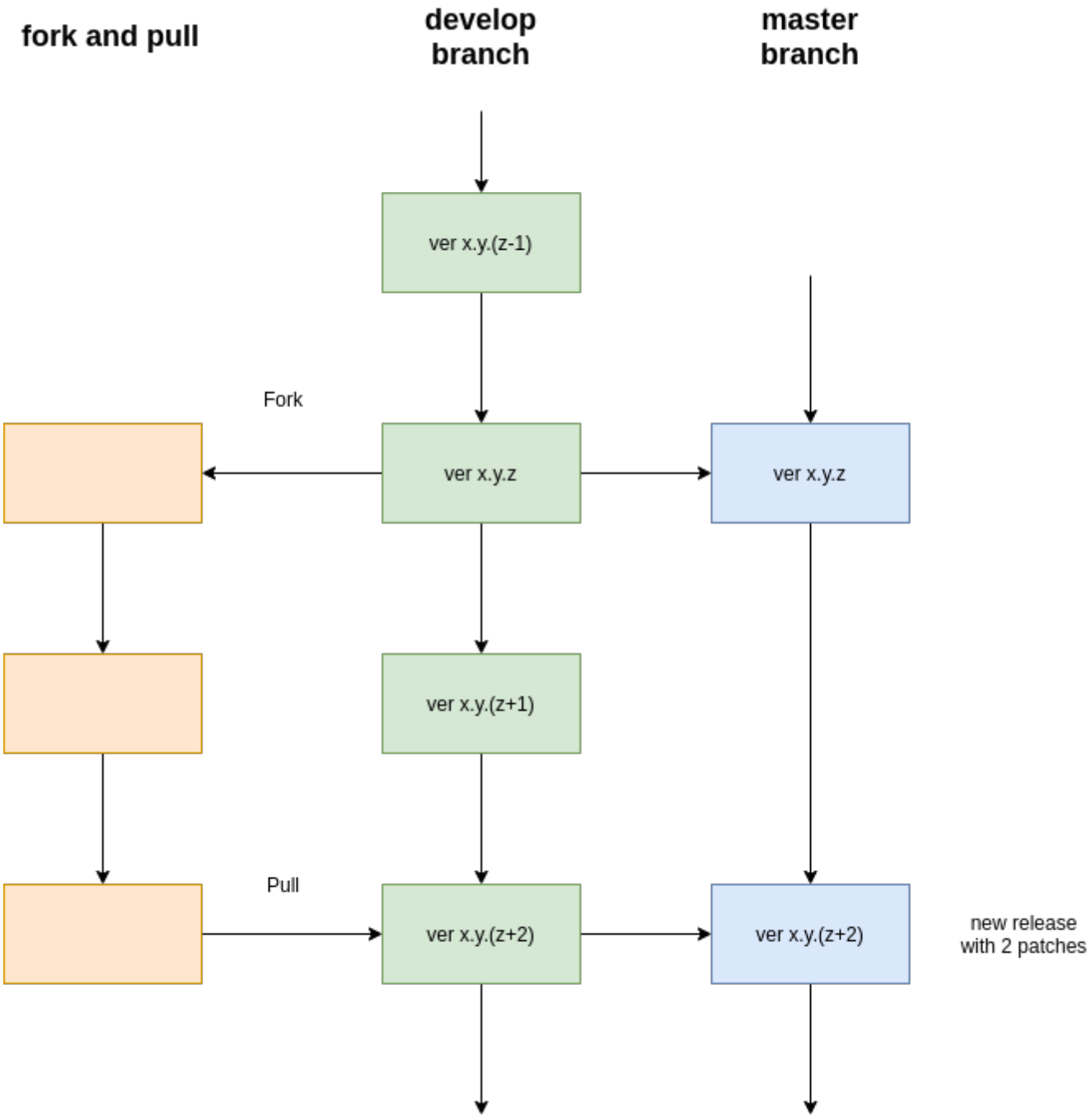


Fig. 2.1 – *Forking and pulling*

Chapter 3

API guide

3.1 EmceeSampler (`taurex_emcee.emcee_optimizer.EmceeSampler`)

Emcee sampler for TauREx3.1.

param observed
Sets the observation to optimize the model to
type observed
BaseSpectrum, optional
param model
The forward model we wish to optimize
type model
ForwardModel, optional
param sigma_fraction
Fraction of weights to use in computing the error. (Default: 0.1)
type sigma_fraction
float, optional
param num_global_samples
Number of samples to initially draw from the prior. Default is 10000
type num_global_samples
int
param num_chains
Number of independent ensembles to run. Default is 4
type num_chains
int
param num_walkers
Ensemble size. Default is max(100, 4 * dim)
type num_walkers
int
param max_ncalls
Maximum number of likelihood function evaluations. Default is 1000000
type max_ncalls
int
param growth_factor
Factor by which to increase the number of steps. Default is 10
type growth_factor
int

param max_improvement_loops
Number of times MCMC should be re-attempted. Default is 4
type max_improvement_loops
int
param num_initial_steps
Number of sampler steps to take in first iteration. Default is 100
type num_initial_steps
int
param min_autocorr_times
If positive, sets autocorelation as an additional convergence criterion. Default is 0
type min_autocorr_times
int
param rhat_max
Sets Gelman-Rubin diagnostic to converge. Default is 1.01
type rhat_max
float
param geweke_max
Sets Gelman-Rubin diagnostic to converge. Default is 2.0
type geweke_max
float
param progress
If True, show progress bars. Default is True
type progress
bool

Chapter 4

License

Chapter 5

Changelog

All notable changes to this project will be documented in this file.

The format is based on Keep a Changelog (keepachangelog), and this project adheres to Semantic Versioning (semver).

5.1 [Unreleased]

5.1.1 0.2.0 [27/11/2023]

First working version of `taurex-emcee`

5.1.2 0.3.0 [13/12/2023]

Integrated `autoemcee` into `taurex-emcee`

5.1.3 0.4.0-beta [15/12/2023]

5.1.3.1 Added

- Added quickstart

5.1.4 0.4.0-rc0 [27/01/2024]

5.1.4.1 Added

- Complete documentation in html and pdf

5.1.5 0.4.0-rc1 [28/01/2024]

5.1.5.1 Added

- package logo

Chapter 6

Acknowledgements

The development of `taurex-emcee` has been possible thanks to Andrea Bocchieri, Quentin Changeat, Lorenzo V. Mugnai, and Enzo Pascale.

Their work was supported by the Italian Space Agency (ASI) with Ariel grant n. 2021.5.HH.0.

Python Module Index

t

Index