The Journal of Open Source Software

Comrade: Composable Modeling of Radio Emission

Paul Tiede[1,2]

1 Black Hole Initiative at Harvard University **2** Center for Astrophysics | Harvard & Smithsonian

DOI: 10.xxxxxx/draft

Software

- Review
- Repository
- Archive

Editor:

Submitted: 22 March 2022
Published: unpublished

Summary

`Comrade` is a Bayesian modeling package, targeted for very-long-baseline interferometry (VLBI) and written in the Julia[1] programming language (Bezanson et al., 2015). `Comrade` aims at producing VLBI image of black holes and active galactic nuclei. Furthermore, it focuses on providing uncertainty quantification of image and physical source properties, such as a black hole's accretion state. The package has already been widely used within the Event Horizon Telescope Collaboration and will be useful for expert and novice VLBI researchers.

Statement of need

Radio interferometric measurements provide the highest resolution images ever produced, culminating in the first image of a black hole (Event Horizon Telescope Collaboration, 2019a, 2019b, 2019c). However, producing VLBI images is not straightforward. An ideal VLBI array samples the Fourier transform of the image, I:

$$V(u, v) = \int I(\alpha, \beta) e^{-2\pi i(u\alpha + v\beta)} d\alpha d\beta.$$

In general, VLBI data sets provide an incomplete sampling of $V(u_i, v_i)$ in the Fourier domain. Therefore, VLBI images are inherently uncertain and quantifying this uncertainty is fundamental to the VLBI imaging problem. This quantification is especially significant for the Event Horizon Telescope, which typically has only 5-8 distinct observing sites. To model this uncertainty, `Comrade` uses Bayesian inference and casts VLBI imaging as a Bayesian inverse problem.

Given the diverse nature of AGN and black holes, `Comrade` includes geometric models, such as Gaussians, disks, rings, crescents to extract relevant features. For non-parametric modeling/imaging, `Comrade` includes a rasterized image model similar to the one described in Broderick, Pesce, et al. (2020). Finally, `Comrade`'s flexible model interface, enables direct physical modeling of an accretion flow in curved spacetime.

Bayesian inference is numerically demanding relative to traditional VLBI imaging and modeling methods. Traditionally these computational demands have required writing large sections of the code in a lower-level language, e.g., `C/C++`. However, this approach comes at a productivity cost to the end-user and makes it difficult for researchers to add their own models. `Comrade` solves this problem by using the Julia programming language. Julia was designed to solve this two-language problem by having `C`-like performance while maintaining a Python-esque syntax and programming experience (Bezanson et al., 2018). Julia achieves these features using a just-ahead-of-time compiler, and code specialization based on multiple dispatch. The effectiveness of this approach has been demonstrated in, e.g, the Celeste project (Regier et al., 2016), where Julia was the first dynamically typed language to break the petaflops barrier.

[1] https://julialang.org

The Journal of Open Source Software

As Julia is a differentiable programming language, most `Comrade` models are natively differentiable. Utilizing gradient information to explore the parameter space quickly will be necessary to find reasonable image structures as image complexity grows. For images of the central black hole of AGN, this is the norm due to complicated accretion structure. `Comrade` is therefore well equipped to deal with the Bayesian VLBI imaging problem, even for large VLBI arrays.

To sample from the posterior `Comrade` has interfaces to nested sampling algorithms, `NestedSamplers` (Lucas et al., 2021) and `AdvancedHMC` (Xu et al., 2020) by default. Moreover, users can specify their own samplers. To make it easy to use `Comrade` with other posterior samplers, the `Comrade` includes functionality that transforms the posterior density from the parameter space to $\mathbb{R}^n$, as well as the unit hypercube. This functionality is needed for Hamiltonian Monte Carlo and nested sampling respectively.

As a result of `Comrade`'s design, it makes it easy to quickly produce posteriors of VLBI data. Here we show an example program that reproduces results from Event Horizon Telescope Collaboration (2019c). Namely and produces posterior of the image structure of the black hole in M 87[2].

```
using Comrade
using Distributions
using Pathfinder
using AdvancedHMC
using Plots
# load eht-imaging we use this to load eht data
load_ehtim()
# To download the data visit https://doi.org/10.25739/g85n-f134
file = ”SR1_M87_2017_096_lo_hops_netcal_StokesI.uvfits”
obs = ehtim.obsdata.load_uvfits(file)
obs.add_scans()
# kill 0-baselines since we don't care about
# large scale structure and make scan-average data
obs = obs.flag_uvdist(uv_min=0.1e9).avg_coherent(0.0, scan_avg=true)
# extract log closure amplitudes and closure phases
dlcamp = extract_lcamp(obs; count=”min”)
dcphase = extract_cphase(obs, count=”min”)
# form the likelihood
lklhd = RadioLikelihood(dlcamp, dcphase)
# build the model: here we fit a ring with a azimuthal
# brightness variation and a Gaussian
function model(params)
  (;rad, wid, a, b, f, sig, asy, pa, x, y) = params
  ring = f*smoothed(stretched(MRing((a,), (b,)), rad, rad), wid)
  g = (1-f)*shifted(rotated(stretched(Gaussian(), sig*asy, sig), pa), x, y)
  return ring + g
end
# define the priors
uas2rad = pi/180.0/3600/1e6
prior = (
          rad = Uniform(uas2rad*(10.0), uas2rad*(30.0)),
          wid = Uniform(uas2rad*(1.0), uas2rad*(10.0)),
          a = Uniform(-0.5, 0.5), b = Uniform(-0.5, 0.5),
          f = Uniform(0.0, 1.0),
          sig = Uniform(uas2rad*(1.0), uas2rad*(40.0)),
          asy = Uniform(0.0, 0.75),
          pa = Uniform(0.0, 1pi),
```


[2] On an Intel i5-7200U chip, this finishes in 165s

```
            x = Uniform(-uas2rad*(80.0), uas2rad*(80.0)),
            y = Uniform(-uas2rad*(80.0), uas2rad*(80.0))
        )
# Now form the posterior
post = Posterior(lklhd, prior, model)
# We will use HMC to sample the posterior.
# First to reduce burn in we use pathfinder
q, phi, _ = multipathfinder(post, 100)
# now we sample using hmc
metric = DiagEuclideanMetric(dimension(post))
chain, stats = sample(post, HMC(;metric), 2000;
                      nadapts=1000, init_params=phi[1])
# plot a draw from the posterior
plot(model(chain[end]))
```

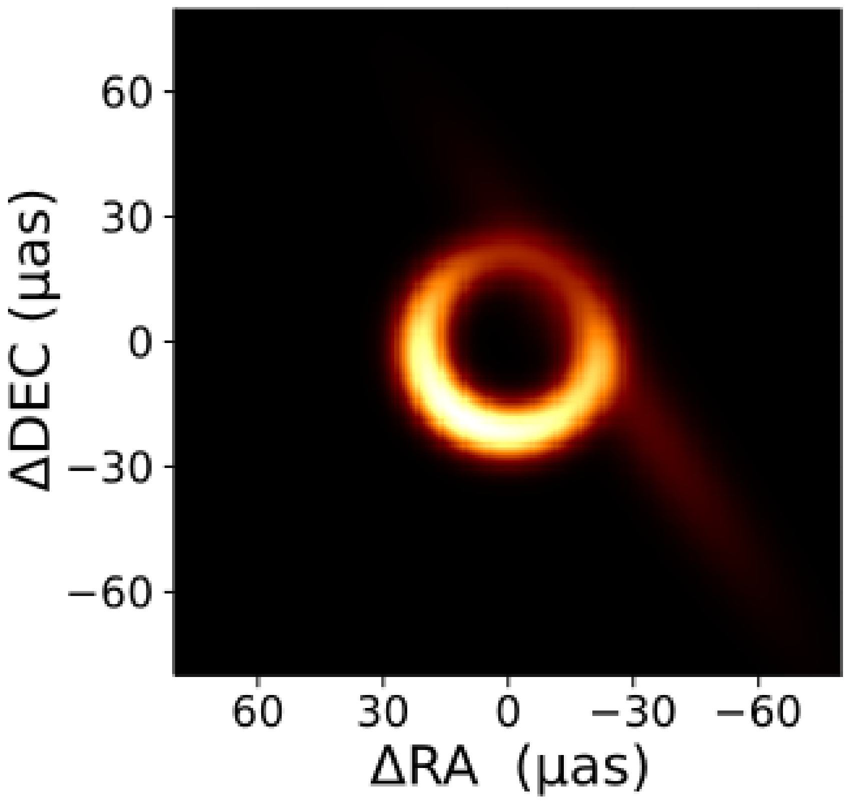

Figure 1: Output of the above code. The image is a random posterior draw for an image of M 87.

Similar Packages

- eht-imaging (Chael et al., 2018): Python general purpose EHT imaging package. It currently has a modeling submodule.
- eht-dmc (Pesce, 2021): Python Bayesian polarized imaging package that also fits calibration systematics by solving the radio interferometry measurement equation (Hamaker, J. P. et al., 1996).
- Galifray (Saurabh, 2022): Python modeling package that uses emcee as its sampler.
- InterferometricModels (Plavin, 2022): Recent Julia radio astronomy package with some similar features to Comrade

- THEMIS (Broderick, Gold, et al., 2020): A C++ parameter estimation package used by the EHT. It is currently a private repository for the EHT.

Acknowledgements

P.T. thanks Michael Johnson, Dom Pesce, Lindy Blackburn, and Avery Broderick for their helpful discussions related to the development of this package and VLBI in general.

This work was supported by the Black Hole Initiative at Harvard University, which is funded by grants from the John Templeton Foundation and the Gordon and Betty Moore Foundation to Harvard University. Additional support was provided by Perimeter Institute for Theoretical Physics. Research at Perimeter Institute is supported by the Government of Canada through the Department of Innovation, Science and Economic Development Canada and by the Province of Ontario through the Ministry of Economic Develop- ment, Job Creation and Trade. PT's research is also supported by the National Science Foundation (AST-1935980 and AST-2034306) and additional financial support from the Natural Sciences and Engineering Research Council of Canada through a Discovery Grant.

References

Bezanson, J., Chen, J., Chung, B., Karpinski, S., Shah, V. B., Vitek, J., & Zoubritzky, L. (2018). Julia: Dynamism and performance reconciled by design. *Proc. ACM Program. Lang.*, *2*(OOPSLA). https://doi.org/10.1145/3276490

Bezanson, J., Edelman, A., Karpinski, S., & Shah, V. B. (2015). *Julia: A fresh approach to numerical computing*. https://doi.org/10.1137/141000671

Broderick, A. E., Gold, R., Karami, M., Preciado-López, J. A., Tiede, P., Pu, H.-Y., & al., et. (2020). THEMIS: A Parameter Estimation Framework for the Event Horizon Telescope. *897*(2), 139. https://doi.org/10.3847/1538-4357/ab91a4

Broderick, A. E., Pesce, D. W., Tiede, P., Pu, H.-Y., & Gold, R. (2020). Hybrid Very Long Baseline Interferometry Imaging and Modeling with THEMIS. *898*(1), 9. https://doi.org/10.3847/1538-4357/ab9c1f

Chael, A. A., Johnson, M. D., Bouman, K. L., Blackburn, L. L., Akiyama, K., & Narayan, R. (2018). Interferometric Imaging Directly with Closure Phases and Closure Amplitudes. *857*(1), 23. https://doi.org/10.3847/1538-4357/aab6a8

Event Horizon Telescope Collaboration. (2019a). First M87 Event Horizon Telescope Results. I. The Shadow of the Supermassive Black Hole. *875*(1), L1. https://doi.org/10.3847/2041-8213/ab0ec7

Event Horizon Telescope Collaboration. (2019b). First M87 Event Horizon Telescope Results. IV. Imaging the Central Supermassive Black Hole. *875*(1), L4. https://doi.org/10.3847/2041-8213/ab0e85

Event Horizon Telescope Collaboration. (2019c). First M87 Event Horizon Telescope Results. VI. The Shadow and Mass of the Central Black Hole. *875*(1), L6. https://doi.org/10.3847/2041-8213/ab1141

Hamaker, J. P., Bregman, J. D., & Sault, R. J. (1996). Understanding radio polarimetry. I. Mathematical foundations. *Astron. Astrophys. Suppl. Ser.*, *117*(1), 137–147. https://doi.org/10.1051/aas:1996146

Lucas, M., Kaur, S., Fjelde, T. E., Alexander, Ge, H., & Pfiffer, C. (2021). *TuringLang/NestedSamplers.jl: v0.8.1* (Version v0.8.1) [Computer software]. Zenodo. https://doi.org/10.5281/zenodo.5808196

Pesce, D. W. (2021). A D-term Modeling Code (DMC) for Simultaneous Calibration and Full-Stokes Imaging of Very Long Baseline Interferometric Data. *161*(4), 178. https://doi.org/10.3847/1538-3881/abe3f8

Plavin, A. (2022). InterferometricModels. In *GitLab repository*. GitLab. https://gitlab.com/aplavin/InterferometricModels.jl

Regier, J., Pamnany, K., Giordano, R., Thomas, R., Schlegel, D., McAuliffe, J., & Prabhat. (2016). Learning an Astronomical Catalog of the Visible Universe through Scalable Bayesian Inference. *arXiv e-Prints*, arXiv:1611.03404. https://arxiv.org/abs/1611.03404

Saurabh. (2022). Galifray: Geometric modelling and parameter estimation using black hole images for EHT including VLBI. In *GitHub repository*. GitHub. https://github.com/Relativist1/Gallifray

Xu, K., Ge, H., Tebbutt, W., Tarek, M., Trapp, M., & Ghahramani, Z. (2020). AdvancedHMC.Jl: A robust, modular and efficient implementation of advanced HMC algorithms. *Symposium on Advances in Approximate Bayesian Inference*, 1–10.