RUBIX: Differentiable forward modelling of galaxy spectral data cubes for gradient-based parameter estimation

Anna Lena Schaible[1,2], **Ufuk Çakır**[3], **Tobias Buck**[1,2], **Harald Mack**[1], **Aura Obreja**[1,2], **Nihat Oguz**[4], **William H. Oliver**[1,2], **Horea-Alexandru Cărămizaru**[5]
[1]Interdisciplinary Center for Scientific Computing, University of Heidelberg, D-69120 Heidelberg
[2]Zentrum für Astronomie, Institut für Theoretische Astrophysik, D-69120 Heidelberg, Germany
[3]Oxford Robotics Institute, Department of Engineering Science, University of Oxford, UK
[4]Institute for Theoretical Physics I, University of Stuttgart, D-70569 Stuttgart, Germany
[5]School of Engineering, The University of Edinburgh, Edinburgh, UK

Abstract

Although integral-field spectroscopy enables spatially resolved spectral studies of galaxies, bridging particle-based simulations to observations remains slow and non-differentiable. We present RUBIX, a JAX-based pipeline that models mock integral-field unit (IFU) cubes for galaxies end-to-end and calculates gradients with respect to particle inputs. Our implementation is purely functional, sharded, and differentiable throughout. We validate the gradients against central finite differences and demonstrate gradient-based parameter estimation on controlled setups. While current experiments are limited to basic test cases, they demonstrate the feasibility of differentiable forward modelling of IFU data. This paves the way for future work scaling up to realistic galaxy cubes and enabling machine learning workflows for IFU-based inference. The source code for the RUBIX software is publicly available under `https://github.com/AstroAI-Lab/rubix`.

1 Motivation

An IFU cube is a three-dimensional dataset obtained using an integral field unit spectrograph. Two dimensions record the spatial distribution of light in the sky, while the third dimension captures the spectrum at each position. Modern IFUs such as MUSE [1] and MaNGA [16] provide spatially resolved spectra across thousands of galaxies, offering unparalleled insight into stellar populations and dynamics. To compare these data with theoretical models, researchers typically forward model cosmological simulations into synthetic IFU cubes. Although existing software (e.g., SimSpin [12, 11], MaNGA data utilities [14, 17] or GalCraft [20]) can perform this mapping, it is often CPU-bound and does not expose derivatives. This restricts their application in contemporary machine learning pipelines, where gradients are vital for optimisation, gradient-assisted simulation-based inference [13, 22] and uncertainty quantification.

Differentiable programming offers a paradigm to close this gap: models are represented as computational graphs and paired with automatic differentiation to expose gradients end-to-end [4]. Frameworks like JAX [5] enable scientists to build such pipelines efficiently. In astrophysics, however, a practical, end-to-end differentiable simulator for IFU cubes has so far been missing.

`RUBIX` [6] addresses this by constructing IFU cubes of simulated galaxies within a single differentiable JAX pipeline. Every stage is functional, vectorized, and designed to scale across devices. In this work we focus on two showcase examples to establish a foundation for future developments towards gradient-assisted galaxy modelling. Crucially, it demonstrates that end-to-end gradients can be

1st Workshop on Differentiable Systems and Scientific Machine Learning @ EurIPS 2025.

obtained in IFU forward modelling, creating a starting point for subsequent integration with machine learning inference frameworks.

2 The RUBIX software

RUBIX is designed as a modular, purely functional pipeline that calculates IFU cubes from particle-level inputs. Its GPU accelerated capabilities were introduced in [6]. The overall flow is illustrated in Figure 1 as a clear sequence of stages and shows the individual physics processes to produce a realistic modelled IFU cube observation.

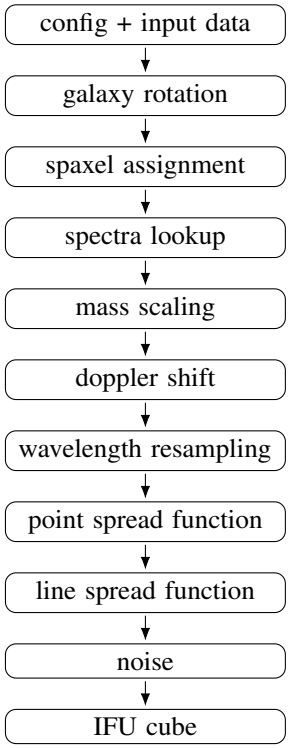

Figure 1: Different steps that happen inside the RUBIX pipeline. We start with a config file and input data, get a mock IFU cube and in between there are pure JAX functions.

All components are implemented in JAX, partially vectorized via vmap, and parallelized across devices using shard_map. This ensures end-to-end differentiability and GPU acceleration as shown in [6], while the modular structure allows individual blocks to be modified and to be selectively chosen during forward modelling via the yaml input file.

Although tailored for astrophysical applications, RUBIX' modular structure with end-to-end gradient support enables direct integration with variational inference frameworks, the embedding of neural networks inside the solver, solver in-the-loop setups [19] and efficient coupling to simulation-based inference pipelines [8].

3 Results

We present two illustrative examples that demonstrate that RUBIX' differentiability and its application to gradient-based parameter optimisation. Our model takes as input the positions and velocities of star particles with specified ages and chemical compositions (metallicities), and computes their stellar spectra using the spectral synthesis library FSPS [7] while consistently incorporating stellar dynamics.

3.1 Gradient validation and the physical interpretation of the gradients

We validate the differentiability of RUBIX by comparing JAX autodiff gradients with central finite differences on the spaxel spectra of a one-particle system with one spaxel[1]. A spaxel is a pixel with the spectral dimension. Figure 2 illustrates both the predicted spectrum and its parameter sensitivities. The top panel shows the luminosity generated by our model at 10 Gyrs and a metallicity of 1.4×10^{-3}. Prominent absorption features are clearly visible, reflecting the underlying stellar template FSPS [7]. The middle panel compares gradients of the luminosity with respect to stellar particle age, computed via automatic differentiation (solid blue) and finite differences (dashed orange). The maximum difference in a wavelength bin between finite difference and autodiff is 4.4×10^{-9} for the age and 3.9×10^{-6} for the metallicity. The two approaches show excellent agreement, validating the correctness of our autodiff implementation. Negative gradients for the age dominate across most wavelengths, indicating that increasing age systematically reduces flux, particularly in the blue region where younger stellar populations contribute most strongly. The bottom panel shows the corresponding gradients with respect to metallicity. Again, the autodiff and finite-difference methods are consistent and we find negative values across much of the optical wavelength range in agreement with physical expectations where increasing metallicity reduces flux in the blue part of the spectrum.

[1]The code to reproduce these results can be found at https://github.com/AstroAI-Lab/rubix/blob/eurips-anonymous/notebooks/gradient_age_metallicity_spectrum.ipynb.

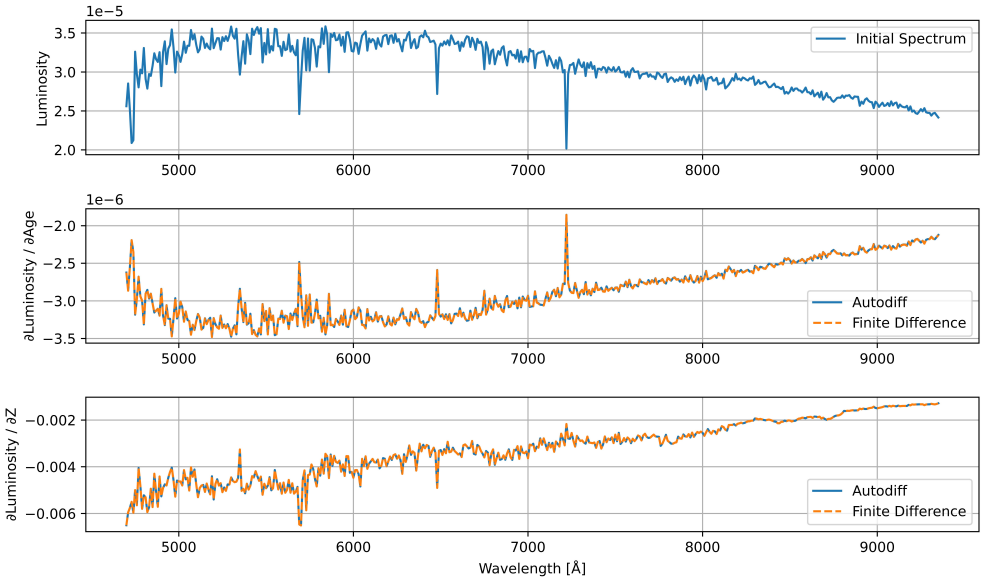

Figure 2: First row shows the spectrum of a single spaxel calculated by RUBIX, which contains one stellar particles with age of 10 Gyrs and metallicity of (1.4×10^{-3}). We calculate the gradient with respect to the stellar age in the second row and show the results for autograd (`jax.jacfwd`) and central finite difference. In the third row we show the gradient via autodiff and the finite difference with respect to the metallicity of the stellar particles.

3.2 Parameter estimation via gradient-based optimization

To demonstrate the utility of RUBIX for inference[2], we constructed with RUBIX a synthetic cube consisting of a single spaxel generated from one stellar particle with fixed stellar age ($a = 3.16$ Gyr) and metallicity ($m = 1.4 \times 10^{-2}$). This target cube (yellow star in Figure 3) is compared against cubes forward-modelled with RUBIX from arbitrary initial age and metallicity values (white circles in Figure 3). The discrepancy between target and model cube was quantified with a scalar loss ℓ per spaxel, $\ell(a, m) = \text{cosine_distance}\Big(\text{IFU}_{\text{target}},\ \text{IFU}_{\text{model}}(a, m)\Big)$, using the `optax.cosine_distance` implementation [9]. We adopted cosine distance because it provided faster convergence and less oscillation than alternative metrics tested.

Since RUBIX is implemented as a pure JAX pipeline, the gradient of the loss with respect to age a and metallicity m can be computed directly. These gradients are then used to update the input parameters with the Adam optimizer [15] (learning rate 5×10^{-3}, tolerance 10^{-10}). All experiments were run on an Apple M3 CPU. The optimization converged within ~ 50 iterations (right panel of Figure 3), yielding spectra that agree closely with the target, with residuals two orders of magnitude smaller than the flux values.

Figure 3 summarizes the process. The left panel shows the log loss landscape in age–metallicity space. Optimization trajectories from different initializations rapidly descend into the L-shaped minimum region, sometimes overshooting metallicity before settling near the true parameters. The right panel shows loss histories for three runs, all decreasing steeply before flattening at low values. Oscillations around the optimum can be reduced by lowering the learning rate, at the cost of more iterations.

4 Discussion and limitations

The experiments presented here highlight both the opportunities and current boundaries of RUBIX. Our validation focused on controlled single-spaxel, one-particle setups, where both target and model come from the same forward model. These controlled setups are deliberately minimal: they provide clean

[2]The code to reproduce these results can be found at `https://github.com/AstroAI-Lab/rubix/blob/main/notebooks/gradient_age_metallicity_adamoptimizer_multi.ipynb`.

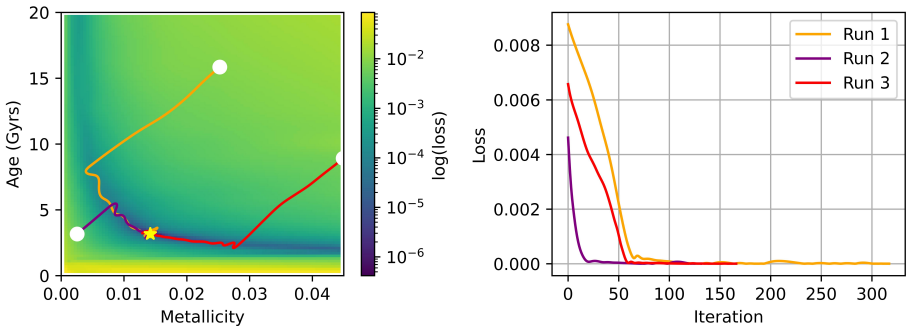

Figure 3: Left: loss landscape and optimization trajectories. The true parameters are marked by the yellow star; initializations are shown as white circles. Right: loss history across iterations for three runs, showing rapid convergence.

benchmarks for testing differentiability and optimization, but they do not yet capture the complexity of realistic IFU cubes containing hundreds of spaxels. Extending RUBIX to full-galaxy simulations is technically feasible but introduces significant challenges related to memory consumption, non smooth loss landscapes and stable gradient propagation across devices.

Ill-posed inverse problems: The loss landscape often contains extended valleys or L-shaped minima, which explains overshooting and oscillations in the recovered parameters. Although RUBIX provides accurate gradients for the synthetic example, these alone are insufficient to resolve intrinsic degeneracies. Addressing them will require more advanced optimization strategies, additional physical constraints (e.g. deep inverse priors [21, 18]), or regularization across many spaxels.

Future applications: Looking ahead, the differentiable design of RUBIX suggests multiple directions for expansion. Because gradients can be propagated through the full pipeline, one can in principle begin from observational data and optimize an underlying particle or distribution-function model that reproduces the observed cube. This goes beyond classical approaches based solely on forward simulations, since gradients naturally capture nonlinear relationships between model parameters and observables. Extending the current examples to realistic galaxies and testing them against survey data such as GECKOS [10] is a natural next step. Importantly, RUBIX is not limited to particle-level inference. It can be coupled with differentiable distribution function models [3, 2], where optimization targets a smaller set of global parameters describing stellar distributions, kinematics, or structural properties. This reduces the dimensionality of the inference problem and provides a more interpretable connection between model parameters and astrophysical quantities of interest.

Uncertainty quantification via variational inference: Beyond point estimation, the differentiable structure of RUBIX naturally enables uncertainty quantification. Integration with modern machine learning inference frameworks, such as simulation-based inference or variational inference, would allow posterior distributions to be approximated rather than single best-fit values.

5 Conclusion

RUBIX demonstrates that differentiable programming can be successfully applied to challenging astrophysical forward modelling. By expressing the entire IFU modelling pipeline in JAX, we obtain end-to-end gradients and GPU-accelerated execution. Our results show that gradients are numerically consistent with finite differences and can be exploited for gradient-based parameter estimation. The present experiments establish a very promising proof of concept for differentiable IFU cube simulation. The framework is designed with scalability in mind, and future work will address realistic galaxies and multi-spaxel cubes.

Overall, RUBIX provides a foundation for combining astrophysical modelling with modern machine learning methodology. Its differentiable structure opens the path toward variational inference, solver in-the-loop setups and simulation-based inference, ultimately enabling new ML workflows for IFU data analysis.

Acknowledgments and Disclosure of Funding

The authors thank Kate Harborne for the support during the development of `RUBIX`. The authors thank the Scientific Software Center at Heidelberg University for the support. This work is funded by the Carl-Zeiss-Stiftung through the NEXUS program. This work was supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy EXC 2181/1 - 390900948 (the Heidelberg STRUCTURES Excellence Cluster). We acknowledge the usage of the AI-clusters Tom and Jerry funded by the Field of Focus 2 of Heidelberg University.

References

[1] R. Bacon, M. Accardo, L. Adjali, H. Anwand, S. Bauer, I. Biswas, J. Blaizot, D. Boudon, S. Brau-Nogue, J. Brinchmann, P. Caillier, L. Capoani, C. M. Carollo, T. Contini, P. Couderc, E. Daguisé, S. Deiries, B. Delabre, S. Dreizler, J. Dubois, M. Dupieux, C. Dupuy, E. Emsellem, T. Fechner, A. Fleischmann, M. François, G. Gallou, T. Gharsa, A. Glindemann, D. Gojak, B. Guiderdoni, G. Hansali, T. Hahn, A. Jarno, A. Kelz, C. Koehler, J. Kosmalski, F. Laurent, M. Le Floch, S. J. Lilly, J. L. Lizon, M. Loupias, A. Manescau, C. Monstein, H. Nicklas, J. C. Olaya, L. Pares, L. Pasquini, A. Pécontal-Rousset, R. Pelló, C. Petit, E. Popow, R. Reiss, A. Remillieux, E. Renault, M. Roth, G. Rupprecht, D. Serre, J. Schaye, G. Soucail, M. Steinmetz, O. Streicher, R. Stuik, H. Valentin, J. Vernet, P. Weilbacher, L. Wisotzki, and N. Yerle. The MUSE second-generation VLT instrument. In Ian S. McLean, Suzanne K. Ramsay, and Hideki Takami, editors, *Ground-based and Airborne Instrumentation for Astronomy III*, volume 7735 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, page 773508, July 2010.

[2] James Binney. More dynamical models of our Galaxy. , 426(2):1328–1337, October 2012.

[3] James Binney and Scott Tremaine. *Galactic Dynamics: Second Edition*. 2008.

[4] Mathieu Blondel and Vincent Roulet. The Elements of Differentiable Programming. *arXiv e-prints*, page arXiv:2403.14606, March 2024.

[5] James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: composable transformations of Python+NumPy programs, 2018.

[6] Ufuk Çakır, Anna Lena Schaible, and Tobias Buck. Fast GPU-Powered and Auto-Differentiable Forward Modeling of IFU Data Cubes. *arXiv e-prints*, page arXiv:2412.08265, December 2024.

[7] Charlie Conroy, James E. Gunn, and Martin White. The Propagation of Uncertainties in Stellar Population Synthesis Modeling. I. The Relevance of Uncertain Aspects of Stellar Evolution and the Initial Mass Function to the Derived Physical Properties of Galaxies. , 699(1):486–506, July 2009.

[8] Kyle Cranmer, Johann Brehmer, and Gilles Louppe. The frontier of simulation-based inference. *Proceedings of the National Academy of Science*, 117(48):30055–30062, December 2020.

[9] DeepMind, Igor Babuschkin, Kate Baumli, Alison Bell, Surya Bhupatiraju, Jake Bruce, Peter Buchlovsky, David Budden, Trevor Cai, Aidan Clark, Ivo Danihelka, Antoine Dedieu, Claudio Fantacci, Jonathan Godwin, Chris Jones, Ross Hemsley, Tom Hennigan, Matteo Hessel, Shaobo Hou, Steven Kapturowski, Thomas Keck, Iurii Kemaev, Michael King, Markus Kunesch, Lena Martens, Hamza Merzic, Vladimir Mikulik, Tamara Norman, George Papamakarios, John Quan, Roman Ring, Francisco Ruiz, Alvaro Sanchez, Laurent Sartran, Rosalia Schneider, Eren Sezener, Stephen Spencer, Srivatsan Srinivasan, Miloš Stanojević, Wojciech Stokowiec, Luyu Wang, Guangyao Zhou, and Fabio Viola. The DeepMind JAX Ecosystem, 2020.

[10] A. Fraser-McKelvie, J. van de Sande, D. A. Gadotti, E. Emsellem, T. Brown, D. B. Fisher, M. Martig, M. Bureau, O. Gerhard, A. J. Battisti, J. Bland-Hawthorn, B. Catinella, F. Combes, L. Cortese, S. M. Croom, T. A. Davis, J. Falcón-Barroso, F. Fragkoudi, K. C. Freeman, M. R. Hayden, R. McDermid, B. Mazzilli Ciraulo, J. T. Mendel, F. Pinna, A. Poci, T. H. Rutherford, C. de Sá-Freitas, L. A. Silva-Lima, L. M. Valenzuela, G. van de Ven, Z. Wang, and A. B. Watts. The GECKOS Survey: Identifying kinematic sub-structures in edge-on galaxies. *arXiv e-prints*, page arXiv:2411.03430, November 2024.

[11] K. E. Harborne, A. Serene, E. J. A. Davies, C. Derkenne, S. Vaughan, A. I. Burdon, C. del P. Lagos, R. McDermid, S. O'Toole, C. Power, A. S. G. Robotham, G. Santucci, and R. Tobar. SimSpin v2.6.0—constructing synthetic spectral IFU cubes for comparison with observational surveys. , 40:e048, October 2023.

[12] Katherine E. Harborne, Chris Power, and Aaron S. G. Robotham. SIMSPIN - Constructing mock IFS kinematic data cubes. , 37:e016, May 2020.

[13] Benjamin Holzschuh and Nils Thuerey. Flow Matching for Posterior Inference with Simulator Feedback. *arXiv e-prints*, page arXiv:2410.22573, October 2024.

[14] Héctor J. Ibarra-Medel, Vladimir Avila-Reese, Sebastián F. Sánchez, Alejandro González-Samaniego, and Aldo Rodríguez-Puebla. Optical integral field spectroscopy observations applied to simulated galaxies: testing the fossil record method. , 483(4):4525–4550, March 2019.

[15] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017.

[16] S. F. Sánchez, J. K. Barrera-Ballesteros, E. Lacerda, A. Mejía-Narvaez, A. Camps-Fariña, Gustavo Bruzual, C. Espinosa-Ponce, A. Rodríguez-Puebla, A. R. Calette, H. Ibarra-Medel, V. Avila-Reese, H. Hernandez-Toledo, M. A. Bershady, M. Cano-Diaz, and A. M. Munguia-Cordova. SDSS-IV MaNGA: pyPipe3D Analysis Release for 10,000 Galaxies. , 262(2):36, October 2022.

[17] Regina Sarmiento, Marc Huertas-Company, Johan H. Knapen, Héctor Ibarra-Medel, Annalisa Pillepich, Sebastián F. Sánchez, and Alina Boecker. MaNGIA: 10 000 mock galaxies for stellar population analysis. , 673:A23, May 2023.

[18] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep Image Prior. *arXiv e-prints*, page arXiv:1711.10925, November 2017.

[19] Kiwon Um, Robert Brand, Yun, Fei, Philipp Holl, and Nils Thuerey. Solver-in-the-Loop: Learning from Differentiable Physics to Interact with Iterative PDE-Solvers. *arXiv e-prints*, page arXiv:2007.00016, June 2020.

[20] Zixian Wang. GalCraft: Building integral-field spectrograph data cubes of the Milky Way. Astrophysics Source Code Library, record ascl:2410.001, October 2024.

[21] Haoyu Yang, Kamyar Azizzadenesheli, and Haoxing Ren. Deep Physics Prior for First Order Inverse Optimization. *arXiv e-prints*, page arXiv:2504.20278, April 2025.

[22] Justine Zeghal, Francois Lanusse, Alexandre Boucaud, Benjamin Remy, and Eric Aubourg. Neural Posterior Estimation with Differentiable Simulator. In *Machine Learning for Astrophysics*, page 52, July 2022.