Contrib. Astron. Obs. Skalnaté Pleso **55/3**, 466 – 469, (2025)
https://doi.org/10.31577/caosp.2025.55.3.466

ObserPy: A tool for efficient observation planning in astronomy

B. Güler[1,2], **Ö. Baştürk**[2,3], **E.M. Esmer**[2,4], **E. Sertkan**[1,2] and **F. Akar**[1,2]

[1] *Ankara University, Graduate School of Natural & Applied Sciences, Astronomy & Space Sciences Department, Ankara, Türkiye*

[2] *Ankara University, Faculty of Science, Astronomy & Space Sciences Department, Ankara, Türkiye*

[3] *Ankara University, Astronomy & Space Sciences Research & Application Center, Kreiken Observatory, Incek Blvd., Ankara, Türkiye*

[4] *Department of Physics and McDonnell Center for the Space Sciences, Washington University, St. Louis, USA*

Received: November 15, 2024; Accepted: January 12, 2025

Abstract. ObserPy is an observation planning program developed in Python. It assists astronomers in selecting the most viable systems to observe based on specific criteria and time intervals, providing detailed information on optimal observation dates, scheduling, and log creation for each target.

Key words: python – observation planning

1. Introduction

Selecting a system for observing extrema in the light curves of variable stars can be challenging, especially when observers have numerous options but limited time. To address this, we developed ObserPy[1]—a Python-based application that helps observers identify the best systems to observe at any given time. It suggests optimal dates for observing a specific system within a set time frame and allows users to create detailed logs documenting each observation.

2. Key features of ObserPy

2.1. Sorting and altitude tracking

ObserPy lets users sort systems by observability, helping astronomers quickly identify which objects will be the "best" targets for the observations of extremum light levels at a given time. It provides altitude data so that users

[1] https://github.com/baris-guler/ObserPy

can track how the position of each object changes over time and the important timings such as light level minima. Additionally, ObserPy calculates the Moon's phase and altitude, which is useful for observers concerned about lunar conditions.

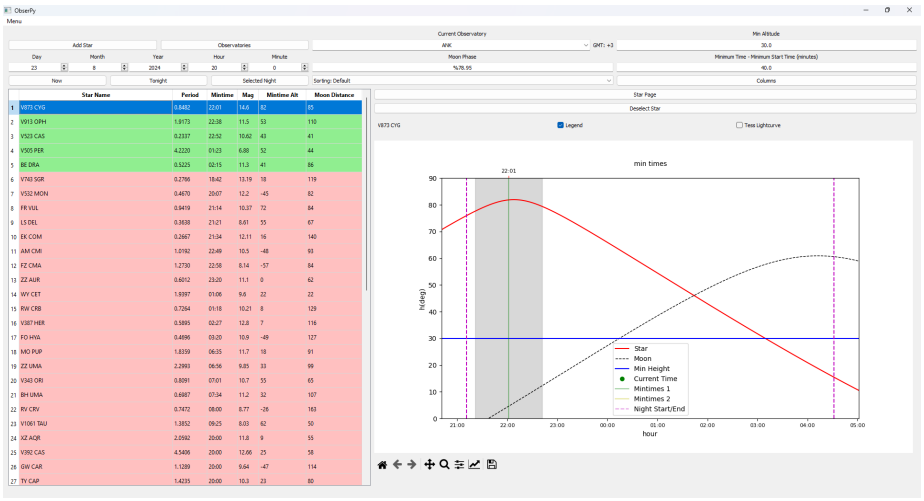


Figure 1. Sorting and altitude tracking in ObserPy

2.2. Custom scheduling and system-specific planning

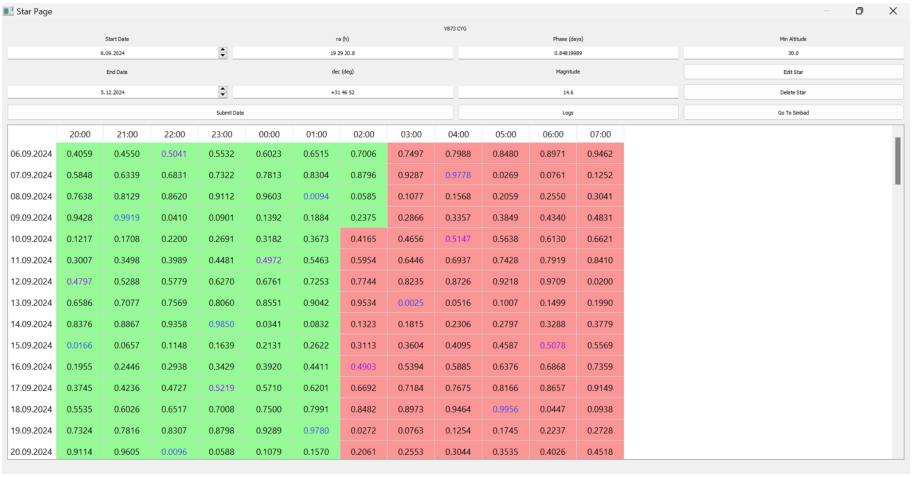
The program is ideal for building observing schedules for research proposals. It calculates observability and minima times within specific intervals, making it adaptable to various observation needs. It provides the data table as output, primarily in graphical format, making it unsuitable for automated or robotic observations unless the user specifically customizes it. Quick access to system data via the SIMBAD database is also provided.

2.3. Log creation for easy record-keeping

ObserPy makes it easy to save logs for each observation, with options to add details and up to two images that can be any .jpg or .png image related to the observation per entry. This feature supports consistent, organized record-keeping for future reference.

2.4. Open-source and customizable

As an open-source tool, ObserPy can be adjusted to suit individual research needs. Users can easily add new systems and observatories. Additionally, the



	20:00	21:00	22:00	23:00	00:00	01:00	02:00	03:00	04:00	05:00	06:00	07:00
06.09.2024	0.4059	0.4550	0.5041	0.5532	0.6023	0.6515	0.7006	0.7497	0.7988	0.8480	0.8971	0.9462
07.09.2024	0.5848	0.6339	0.6831	0.7322	0.7813	0.8304	0.8796	0.9287	0.9778	0.0269	0.0761	0.1252
08.09.2024	0.7638	0.8129	0.8620	0.9112	0.9603	0.0094	0.0585	0.1077	0.1568	0.2059	0.2550	0.3041
09.09.2024	0.9428	0.9919	0.0410	0.0901	0.1392	0.1884	0.2375	0.2866	0.3357	0.3849	0.4340	0.4831
10.09.2024	0.1217	0.1708	0.2200	0.2691	0.3182	0.3673	0.4165	0.4656	0.5147	0.5638	0.6130	0.6621
11.09.2024	0.3007	0.3498	0.3989	0.4481	0.4972	0.5463	0.5954	0.6446	0.6937	0.7428	0.7919	0.8410
12.09.2024	0.4797	0.5288	0.5779	0.6270	0.6761	0.7253	0.7744	0.8235	0.8726	0.9218	0.9709	0.0200
13.09.2024	0.6586	0.7077	0.7569	0.8060	0.8551	0.9042	0.9534	0.0025	0.0516	0.1007	0.1499	0.1990
14.09.2024	0.8376	0.8867	0.9358	0.9850	0.0341	0.0832	0.1323	0.1815	0.2306	0.2797	0.3288	0.3779
15.09.2024	0.0166	0.0657	0.1148	0.1639	0.2131	0.2622	0.3113	0.3604	0.4095	0.4587	0.5078	0.5569
16.09.2024	0.1955	0.2446	0.2938	0.3429	0.3920	0.4411	0.4903	0.5394	0.5885	0.6376	0.6868	0.7359
17.09.2024	0.3745	0.4236	0.4727	0.5219	0.5710	0.6201	0.6692	0.7184	0.7675	0.8166	0.8657	0.9149
18.09.2024	0.5535	0.6026	0.6517	0.7008	0.7500	0.7991	0.8482	0.8973	0.9464	0.9956	0.0447	0.0938
19.09.2024	0.7324	0.7816	0.8307	0.8798	0.9289	0.9780	0.0272	0.0763	0.1254	0.1745	0.2237	0.2728
20.09.2024	0.9114	0.9605	0.0096	0.0588	0.1079	0.1570	0.2061	0.2553	0.3044	0.3535	0.4026	0.4518

Figure 2. Altitude table for single system planning in ObserPy

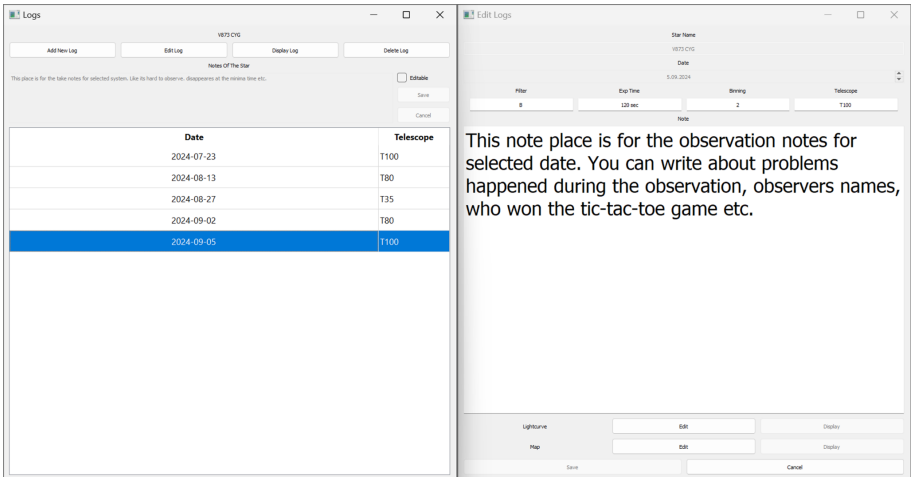


Figure 3. Log creation and management in ObserPy

code can be customized as needed, making ObserPy a flexible choice for astronomers.

2.5. Simple documentation

ObserPy includes straightforward documentation available through github, making it easy for new users to get started and explore its full capabilities.

3. Comparison with existing tools

Several observation planning tools, such as the Isaac Newton Group of Telescopes' Visibility Tool[2] and `Astroplan`[3], offer helpful features for planning. ObserPy, however, focuses on a simple design and easy use as well as the needs of variable star observers, making it a useful for them, who would like to know when and at which altitude their variable stars will be in extremum light levels.

Acknowledgements. The team gratefully acknowledge the support of TÜBİTAK under the research project 122F358, which made the development of ObserPy possible.

[2] https://astro.ing.iac.es/staralt/

[3] https://github.com/astropy/astroplan