DOI: 10.61782/fa.2025.0507

FORUM ACUSTICUM EURONOISE 2025

IOTA: A POSITIONAL DATA PROCESSING SOFTWARE FOR AIRCRAFT NOISE SIMULATIONS

Thomas Ramseier[1*] **Mateja Gligorijevic**[1] **Stefan Plüss**[1] **Beat Schäffer**[1]

[1] Empa, Swiss Federal Laboratories for Materials Science and Technology, Dübendorf, Switzerland

ABSTRACT

Aircraft noise simulations rely on accurate positional data to achieve precise noise mapping based on real air traffic. This data can come from different sources, such as radar data or ADS-B. In most cases, the raw data requires processing to represent the real flown trajectories more closely. Indeed, such data often displays large amount of scatter, missing information and/or unusable or unavailable data close to the runway. For such data processing, the novel program IOTA was developed in Python, which can process different types of trajectory data. It was developed, since the current processing software SELFA2 of Empa is getting difficult to maintain and contains processing steps, which are not required nowadays due to improved radar data quality. In this contribution, IOTA with its processing steps, including Kalman smoothing, addition of data on the runway and horizontal correction, will be presented. Further, comparisons with SELFA2 will be shown. The comparisons reveal small differences in the resulting positional data and velocity profiles, mostly occurring close to the runways and for narrow turns. Resulting differences in computed sound pressure levels are then analyzed for different classes of aircraft types (airliners, propeller aircraft, private jets) for the Swiss airports of Geneva and Zürich.

Keywords: *Aircraft noise, radar data, acoustic simulation, Kalman smoothing, noise mapping.*

****Corresponding author***: *thomas.ramseier@empa.ch.*

1. INTRODUCTION

The current program used by the Empa for positional data processing, SELFA2, has been used now for 20 years to process raw radar data provided by the airports of Geneva (GVA) and Zürich (ZRH), Switzerland, for yearly aircraft noise calculations. As SELFA2 became difficult to maintain and as radar data improved over the years, rendering certain processing steps unnecessary, a new program for positional data processing, IOTA, was developed in Python. The aim of developing IOTA was to enable efficient maintenance and development in the long term and adding new features and improvements to radar data processing. For example, different types of inputs can be given to the program such as ADS-B or flight data recorder (FDR) data. The processing of the trajectories has also been improved using Kalman smoothing, instead of the previously used B-splines, to better reflect the raw trajectories and enable intuitive setting of the parameters depending on the aircraft type. Finally, IOTA enables the visualization of each processing step, which eases the identification of potential problems and tuning of the settings to achieve the desired outcome.

2. METHOD

In this section, the main steps are first summarized and then the processing step is explained in more detail, as it is the central part of IOTA.

- **Reading of the radar trajectories:** The radar data is read into a standardized format in the form of a Python dictionary. A table is also saved, which contains the meta data (Take-off/landing time, runway, route, destination, ...) of these trajectories.

- **Matching the list of movements and radar data:** The table containing meta information mentioned before is then compared with a list of movement provided by the airport. it contains the same information as the previous meta data table, but additionally also information on the identification number and type of aircraft, among others. It is used to identify the individual flights contained in the raw radar data.

- **Filtering of the trajectories:** The individual trajectories need to fulfill minimal quality requirements before going through further processing steps. Those include, among others, that no large gaps are present in the data, that the data has points close enough from the runway and that the trajectory goes through certain gates (defined by two horizontal coordinates and an upper and lower altitude) set by the user. Other quality criteria count the number of certain defects in the data, such as duplicate positions or negative time steps. If there are too many of these points, the trajectory is deemed unusable and gets rejected.

- **Correction of the trajectories:** This step of the program corrects, some of the issues (if present) detected before. Those corrections include the removal of duplicate data points or successive data points with negative time steps. This correction is only done, if the amount of problematic data points remains acceptable, as explained before.

- **Processing of the trajectories:** Here, IOTA processes each trajectory to improve its quality. This part of IOTA is explained in more detail in the following section, as it is the most complex part of the program.

Processing of the trajectories: In a first step, a horizontal shift is applied to the trajectory, where necessary, so that it lies exactly on the runway centerline, once it is close to or directly on the runway. This step is necessary due to the observed lack of precision of radar data for this part of the flight. The correction consists of two adjustments. First, the part of the trajectory lying on the runway and up to a chosen distance away from the end of it, is projected onto the runway centerline. From this distance and up to a maximal distance d_m defined by the user, a linearly decreasing lateral adjustment is applied to the data points. Thus, the lateral correction is only applied in the vicinity of the runway. Beyond d_m, the data points remain unchanged.

In a second step, the scatter of the positional data is smoothed to obtain more plausible trajectories overall. Smoothing also helps to increase the robustness of further steps of the process, such as the extrapolation or computations of derivatives described below, as they are sensitive to noise. An appropriate method for such a task is Kalman smoothing [1]. The Kalman smoother, in contrast to the Kalman Filter, provides a more accurate estimate by using "future data" (in the sense of data from subsequent time steps), if available, which is the case for the present application.

In a third step, the trajectories need to be extrapolated onto the runway, since radar and ADS-B trajectories often have unusable or missing data below the altitude of a few hundred meters above runway. The extrapolation is done by first computing a 3D regression line using the N_p last position points before the runway. The intersection between this 3D line and the 2D plane approximating the runway is then computed. This point approximates the take-off/touchdown point. The trajectory is then extended until the beginning (break release point for departures) or end of the runway (last point on the runway for landings), and a constant acceleration is enforced to meet the requirements of aircraft noise simulation regulations in Switzerland [2]. This step also retrieves the exact height above the runway by interpolating elevation values from a Digital Terrain Model (DTM) and adding the aircraft height. Finally, the transition between the points on the ground and in the air, i.e. around the take-off/touchdown point, is smoothed using a B-spline to ensure a realistic transition to the flight part of the trajectory.

3. VALIDATION

This section shows samples of the tests and comparisons done to (i) ensure that the code works as expected and (ii) to analyze if and how it differs from the current program SELFA2. The results are shown exemplarily first for single flights. In this part of the validation, so-called footprints are computed for the same flight using SELFA2 and IOTA for the processing of the trajectory. A footprint is a computation of sound immission (we use the A-weighted sound exposure level L_{AE} in this study) on all receiver points (arranged as a grid). We then analyze in the second part of the validation, footprints for a bundle of flights (of one aircraft type on a specific air route during either departure or approach). The footprints from bundle of flights are obtained by summing up energetically the individual footprints of each flight. In annual calculations, the individual footprints are normally weighted with the number of corresponding movements of each route and aircraft type combination, which is in this study set to one for each combination, as real traffic movements are not analyzed here. The bundle of flights simulations are particularly important, since they indicate if systematic differences between both programs are to be expected when a representative sample of flights is accounted for. As IOTA is aimed at processing positional data for yearly calculations, it is important to know, what discrepancies could occur for such simulation scenarios.

3.1 Data and simulation tool used

For the validation of IOTA, radar data from GVA and ZRH from the year 2022 was used. The aircraft noise simulation software FLULA2 [3] was chosen to compare IOTA with SELFA2 in terms of resulting acoustical footprints. In this contribution, examples from GVA are shown.

3.2 Comparison of processed (IOTA and SELFA2) radar data with FDR data

To ensure that the output of IOTA is close to the real flown trajectories, comparisons of trajectories for which radar and independent FDR data were available, were undertaken. Since FDR data are considered the most accurate and reliable aircraft position data, as they rely on the aircraft's sensors and are provided in 1 s time steps, instead of 4 s for radar data, they were used as a reference. Flights of Airbus A320neo and A321neo were used for that purpose. For most examples, the absolute differences in altitude remained within an acceptable margin for aircraft noise of around 10 to 20 m. Figure 1 shows, exemplarily for one departure, that both IOTA and SELFA2 match well in terms of altitude and velocity profiles, when compared to FDR data.

3.3 Single flight comparison

An exemplary flight is shown in the following account to reveal what differences can be expected between both programs. Figure 2 shows that IOTA tends to follow the raw trajectory points more closely in terms of altitude (upper figure) and velocity profile (lower figure), while SELFA2 is using a stronger averaging. This may lead to some differences between the two programs, especially close to the ground. Further, the velocity on the runway can differ between both programs, as seen in this figure, despite

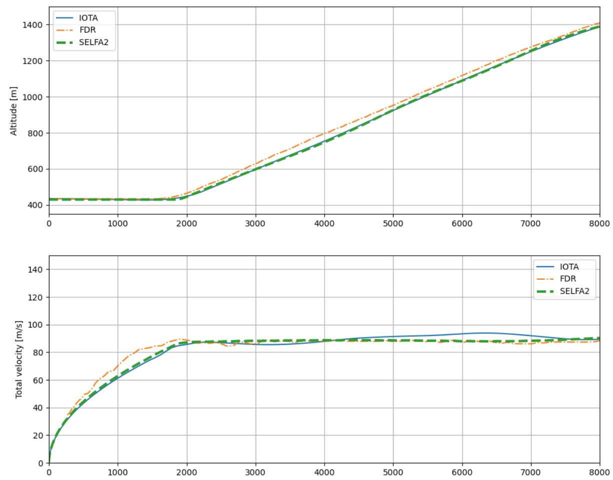

Figure 1. Comparison between processed radar data using SELFA2, IOTA and FDR. Above: distance vs. altitude. Below: distance vs. velocity.

both following the prescribed constant acceleration, when the aircraft is on the ground. This is due to the fact that the velocity of the first point in the air may be different due to the aforementioned reason. This will influence the constant acceleration value to reach that velocity in the available distance.

Resulting footprint differences when using IOTA and SELFA2 for the trajectory processing are shown in figure 3 for the above flight. The influence of the disparate velocities on the runway are clearly visible. In this example, IOTA yields faster velocities on the runway than SELFA2, which results in smaller L_{AE} values for that part of the flight with IOTA, than with SELFA2. Further away, both simulations coincide well. A slight increase in the L_{AE} of IOTA compared to SELFA2 can be seen approximately 2 km after the runway end, directly under the flight path. This is due to the fact that IOTA is briefly slower than SELFA2 in this part of the flight, as seen in figure 2; the small differences in altitude, in contrast, are hardly visible.

Figure 4 shows a further difference between both programs, regarding smoothing. In this example, IOTA smooths the trajectory stronger at the transition between rolling on the runway and flight. This smoothing can be, however, set differently which would also affect the differences with respect to SELFA2.

The altitude differences observed in this example af-

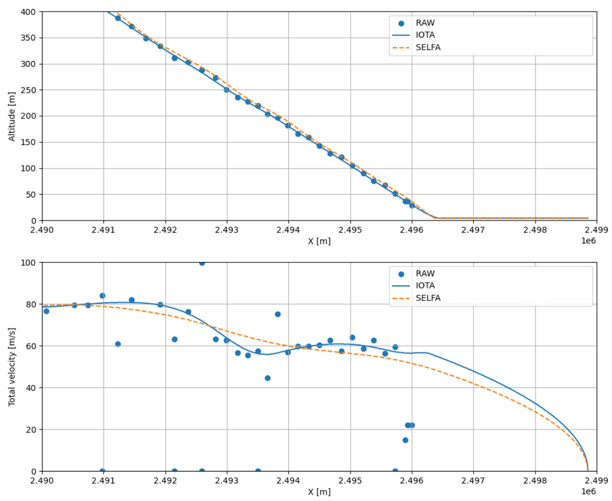

Figure 2. Comparison of raw radar data (blue dots), SELFA2 (dashed orange line) and IOTA (blue line) on an exemplary flight. Above: X position vs. altitude. Below: X position vs. velocity.

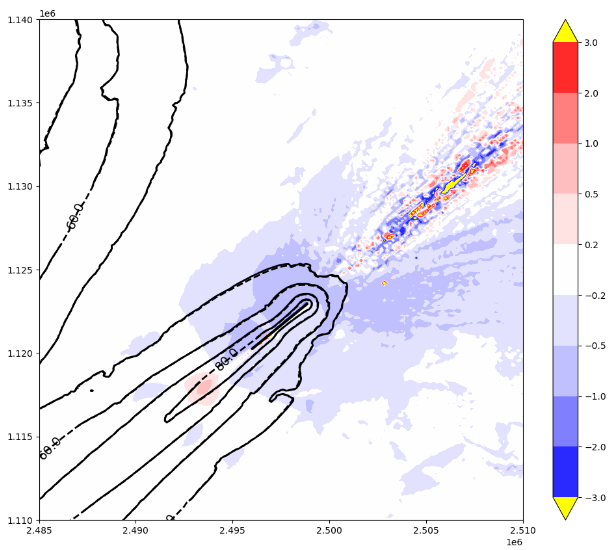

Figure 3. L_{AE} footprint difference IOTA-SELFA2 for the above flight.

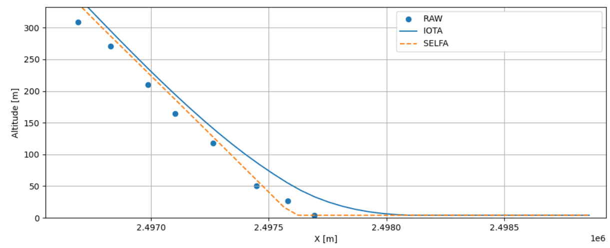

Figure 4. Comparison of raw radar data (blue dots), SELFA2 (dashed orange line) and IOTA (blue line) on an exemplary flight. The X position vs. the altitude at the transition between rolling on the runway and flight is shown.

fect resulting footprints locally, as seen in figure 5. A sound level reduction is observed similarly to figure 3 due to a higher velocity on the runway with IOTA. The IOTA trajectory is subsequently higher in the air than SELFA2 just after the rotation point. This induces a positive single sound level event difference due to the increased lateral attenuation [4] in the case of SELFA2, as it flies closer to the ground at this point. Below the flightpath, where the lateral attenuation is not present, IOTA generates lower sound levels, as it flies higher than SELFA2.

3.4 Bundle of flights comparison

After comparing footprints of individual flights, the resulting differences in footprints from bundle of flights are studied. Two examples are shown from the extensive comparisons that were undertaken for this project. The effect of filtering the trajectories differently between SELFA2 and IOTA is also considered, as the bundles of flights contain only the flights that each program retained after the filtering step (see section 2). Figure 6 exemplarily shows a FLULA2 simulation for $\simeq 35$ departures in GVA. Overall, differences are mostly negligible ($|\Delta L_{AE}| \leq 0.5$ dB) for the domains of interest ($L_{AE} \leq 50$ dB).

Figure 7 shows the differences between a superposition of 6 landings of B7672. The differences remain mostly negligible here as well, except for some areas with low noise exposure of $L_{AE} \leq 50$ dB or below, far from the airport and air routes, which are of limited interest. Note that the two examples are representative also of other comparisons undertaken during this study, which did not reveal any systematic differences between the two programs. Also, footprints may have many more flights (sev-

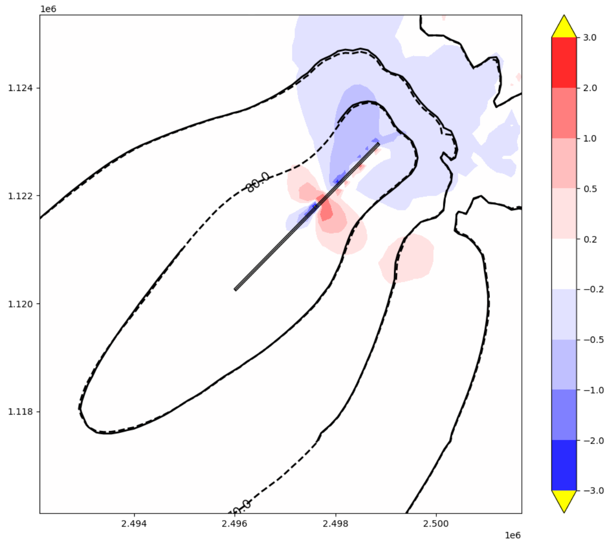

Figure 5. L_{AE} footprint difference IOTA-SELFA2 for the above flight.

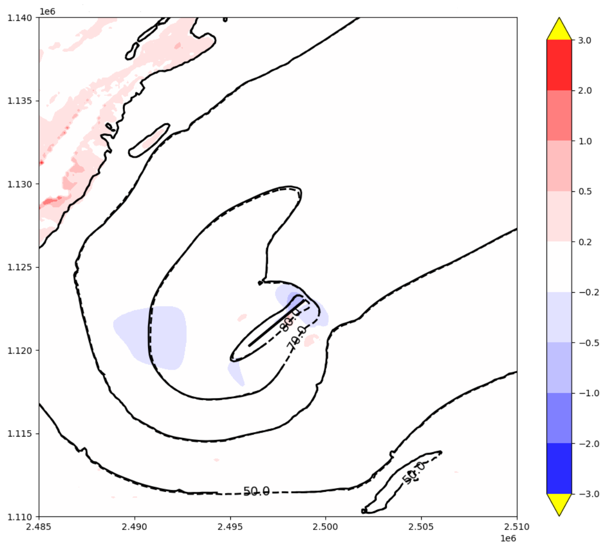

Figure 6. Differences between both programs on a noise footprint resulting from approximately $\simeq$ 35 departures (IOTA: 34, SELFA2: 35) of AT42 on runway 22 in GVA.

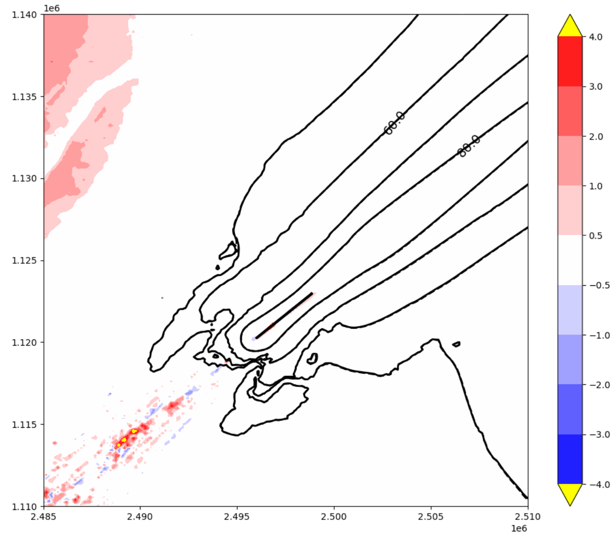

Figure 7. Differences between both programs on a grouping of 6 landings (for both programs) of B7672 on runway 22 in GVA.

eral hundreds), so that differences of individual flights between the programs equal even more out.

4. CONCLUSION

A new software for processing aircraft positional data has been written in Python, called IOTA. It replaces SELFA2, which is the current program used by Empa. Some improvements have been added in this new implementation, such as Kalman smoothing, to enable more refined smoothing of the trajectories. The program was tested on radar data from 2022 for GVA and ZRH airports. IOTA with FLULA2, has resulted in sound pressure levels, which agree very well with the results of SELFA2. Both programs were compared in terms of trajectories and FLULA2 noise simulations in GVA and ZRH, for individual flights and groupings of flights. The study showed that the way in which positional data is processed, can have substantial effects on aircraft noise simulation results. In particular, the smoothing of the data should be carefully investigated and tuned. IOTA will now be validated on a larger scale, by using it in annual aircraft noise calculations for ZRH and GVA for the year 2024. A comparison with SELFA2 will be again done to identify the possible discrepancies between both programs.

5. ACKNOWLEDGMENTS

The authors are grateful to Stefan Zbinden, former radar data expert at Skyguide, for providing useful insights into the functioning of radar systems, ADS-B data and aircraft operations overall. Further, we would like to thank the ZRH and GVA airport for providing us the input radar data and in particular GVA, for enabling us to present our results at this conference. Finally, we thank Swiss International Air Lines, for approving the use of their FDR data for this study and to present exemplary results.

6. REFERENCES

[1] E. Brookner, "Tracking and Kalman Filtering Made Easy," 1998.

[2] BAFU, BAZL and GS VBS, "Leitfaden Fluglärm, Vorgaben für die Lärmermittlung. Umwelt-Vollzug, Lärm Nr. 1625," 2021.

[3] W. Krebs, R. Bütikofer, P. S., and G. Thomann, "FLULA2, Ein Verfahren zur Berechnung und Darstellung der Fluglärmbelastung. Technische Programm-Dokumentation. Version 4," report, Eidgenössische Materialprüfungs- und Forschungsanstalt (Empa), Abteilung Akustik / Lärmminderung, 2010.

[4] W. Krebs and G. Thomann, "Aircraft noise: New aspects on lateral sound attenuation," *Acta Acustica united with Acustica*, vol. 95, p. 1013–1023, Nov. 2009.