Revista Mexicana de Astronomía y Astrofísica Serie de Conferencias (RMxAC), **57**, 42–46 (2024)

https://doi.org/10.22201/ia.14052059p.2024.57.10

III Workshop for Astronomy Beyond the Common Senses for Accessibility and Inclusion (3WAI) (November 29-30, 2023)
Editors: Santiago Vargas, Beatriz García, Johana Casado, Breezy Ocaña, and Angela Pérez - DOI: https://doi.org/10.22201/ia.14052059p.2024.57.10

EAR TO THE SKY: ASTRONOMICAL SONIFICATION FOR ACCESSIBLE OUTREACH, EDUCATION AND RESEARCH WITH STRAUSS

James W. Trayford[1], Samantha Youles[1], Nic Bonne[1], and Chris M. Harrison[2]

RESUMEN

El sesgo visual que impregna los campos STEM es especialmente notorio en astronomía, con un énfasis en la observación de fenómenos astronómicos, y una percepción pública construida en torno a *'imágenes bonitas'*. Sin embargo, el enfoque visual excluye a diversas personas que tienen preferencias no visuales para el aprendizaje y la comunicación. Actualmente, existen esfuerzos para explorar un enfoque multimodal y expresar datos de nuevas maneras. La sonificación, en particular, ofrece muchas posibilidades prometedoras para construir percepciones e intuiciones reales en conjuntos de datos astronómicos complejos, aprovechando el rango dinámico elevado de la audición humana (tanto en volumen como en tono). Desafiar los sesgos visuales arraigados la astronomía requiere pruebas extensas y sólidas de modalidades alternativas, junto con herramientas adecuadas y generales para sonificar datos. El proyecto *Ear to the Sky* tiene como objetivo abordar estos dos aspectos mediante la investigación de varias aplicaciones prometedoras de la sonificación en astronomía, al mismo tiempo que utiliza y desarrolla el código Python `strauss` como una herramienta generalizada para sonificar datos con fines de análisis, divulgación y educación.

ABSTRACT

The visual bias pervading STEM fields is particularly keen in astronomy, with an emphasis on observation and imaging of astronomical phenomena, and a public perception built on *"pretty pictures"*. However, visual focus is exclusionary for a range of people who have non-visual preferences for learning and communication. Efforts are growing to explore a multimodal approach and express data in new ways. Sonification in particular provides many promising possibilities of building real insights and intuition into complex astronomical data sets, harnessing the high dynamic range of human hearing (in terms of both volume and pitch). Challenging the entrenched visual biases in the presentation of astronomy requires widespread and robust testing of alternative modalities, alongside adequate and general tools to sonify data. The *Ear to the Sky* project aims to tackle these two aspects through investigating a number of promising applications of sonification in astronomy, while using and developing the `strauss` Python code as a generalised tool for sonifying data for analysis, outreach and education.

Key Words: sonification — multimodality — data analysis

1. INTRODUCTION & BACKGROUND

Visual modes and methods have long predominated STEM[3] fields, and are particularly emphasised astronomy. Telescope imagery presents the cosmos as a visually beautiful, but typically silent and static place. While such imagery can be very effective in inspiring people and drawing them into astronomy, relying purely on visual modes is exclusionary. In particular, this can be inaccessible for people who are *Blind or Visually Impaired* (BVI), or who have non-visual learning styles. Perhaps the beauty and complexity of this data can be expressed in other ways.

Sonification provides an alternative to visualisation, representing data using non-verbal sound (Kramer et al. 1999). This is far less widespread than visualisation processes, but has the potential to provide an accessible channel to otherwise inaccessible fields, and give all of us new perspectives on data that could provide deeper intuition and even lead to novel discovery in the data. Sonification also has potential as one facet of a *multimodal* approach to education; where different sensory inputs work synergistically together to enhance our understanding of a subject (Lim et al. 2022).

Alongside the presentation of astronomical data in educational and outreach contexts, sonification can also be used in more analytic and research con-

[1]Institute of Cosmology and Gravitation, University of Portsmouth, Dennis Sciama Building, Burnaby Road,Portsmouth, UK, (james.trayford@port.ac.uk).

[2]School of Mathematics, Statistics & Physics, Newcastle University, Herschel Building, Newcastle upon Tyne, UK

[3]i.e. Science, Technology, Engineering & Mathematics

III Workshop for Astronomy Beyond the Common Senses for Accessibility and Inclusion (3WAI) (November 29-30, 2023)
Editors: Santiago Vargas, Beatriz García, Johana Casado, Breezy Ocaña, and Angela Pérez - DOI: https://doi.org/10.22201/ia.14052059p.2024.57.10

texts. While sonification is broadly underused, there are notable examples of its scientific use in astronomy (see e.g. Zanella et al. 2022). Work is ongoing to demonstrate the utility of sonification for specific analysis tasks in astronomy (e.g. Diaz-Merced 2013; Tucker Brown et al. 2022; Trayford et al. 2023).

In order for sonification approaches to have comparable usefulness to visualisation, it is important to have effective tools to sonify data. Increasingly, efforts have gone into producing software for sonification including graphical user interfaces (e.g., Potluri et al. 2023; Casado et al. 2023, 2024), scripting tools and libraries (e.g., Worrall et al. 2007; Tucker Brown et al. 2022; Reinsch & Hermann 2023), and also online applications (e.g., Cantrell et al. 2021; De la Vega et al. 2023).

`STRAUSS`[4] (Trayford & Harrison 2023) is one such tool currently in open development, intended as a *"high ceiling, low barrier"* Python package for general data sonification. By using abstraction and an object-oriented structure, with possible analogies to popular data visualisation or plotting libraries (e.g. *matplotlib* Hunter & Dale 2007), `STRAUSS` is intended to be integrated into the workflow of analysts to flexibly produce audio representations of data. The code is explained further in the following section. These tools can then be applied to increasingly complex and unwieldy data sets emerging from astronomical facilities and modelling, as the field becomes ever more data intensive (e.g. Zhang & Zhao 2015). We look to demonstrate how sound can provide an alternative sensory channel to this data, bringing alternative ways of handling data and encouraging broader inclusion and forms of contribution to scientific discovery.

2. THE STRAUSS CODE

The `STRAUSS` code is a free and open-source (FOSS), object-oriented python package[5], for the general sonification of data. Full details of features & structure of the code, alongside the philosophy behind it are detailed in Trayford & Harrison (2023). `STRAUSS` is comprised of a few primary classes that outline its conceptual structure, summarised briefly below.

The *`Source`* object handles the mapping between the input data and expressive properties of sound (such as pitch, volume, timbre, modulation, etc). Sub-classes of *`Source`* used to treat different types of data include *`Events`* and *`Objects`*; *`Events`* can be thought of as *data points* associated with sets of single values intended to have some occurrence time in the sonification, $t_{\rm occ}$. Supernovae observations could be an example of this; they have a detonation time, a proximity, a direction, a type, etc. This data could be mapped to things like $t_{\rm occ}$, volume, stereo position, choice of pitch, respectively. Conversely, *`Objects`* can persist and evolve through time, so may have multiple *data series* associated with them. Galaxies in a cosmological simulation could be an example of this, they can have properties such as star formation rate and metal composition, all varying as a function of time. As an example these could be mapped to time varying pitch and timbre, respectively. With these mappings in place, the next relevant class is the *`Score`* (as in musical score), which can be used to set limitations on these mappings. The *`Score`* can be thought of as mainly aesthetic choices, such as the choice of musical notes or keys, or rhythmic quantisation. This separation is intended to make it easier for the sonifier to control what aspects of the sounds are being *imposed* versus what aspects are *emerging* from the data itself. We note that aesthetic choices should not be regarded as unimportant in scientific or technical contexts, however, as they can be important for clarity and the user experience of the sonification.

Next is the *`Generator`* class, which generates the sounds themselves. Using the mappings for each *`Source`* and choices made in *`Score`*, the *`Generator`* creates the waveforms associated with each, either via the inbuilt *`Synthesizer`* (mathematically generated waveforms) or the *`Sampler`* (manipulating real recorded sounds). The *`Channels`* class then routes these sounds into different audio channels, depending on a chosen audio *setup* and the positional information provided for each source. For example, a `'stereo'` setup will have left and right channels, while `'mono'` will have only one (chosen, for example, if directional information is unimportant). `STRAUSS` can also support surround-sound and ambisonic formats. Finally, the *`Sonification`* class acts as a wrapper for the underlying classes and provides high level methods, such as rendering the sonification itself.

3. ACCESSIBLE USE OF STRAUSS

Accessibility considerations come into `STRAUSS` development in a number of different areas;

1. The BVI accessibility of using the code itself, such that BVI analysts can work with data as part of an accessible workflow (§3.1, §3.2, §4).

[4] Sonification Tools & Resources for Analysis Using Sound Synthesis

[5] `https://github.com/james-trayford/strauss`

III Workshop for Astronomy Beyond the Common Senses for Accessibility and Inclusion (3WAI) (November 29-30, 2023)
Editors: Santiago Vargas, Beatriz García, Johana Casado, Breezy Ocaña, and Angela Pérez - DOI: https://doi.org/10.22201/ia.14052059p.2024.57.10

2. The accessibility of the code conceptually, such that it easy for everyone to understand and use across different levels (§3.1, §3.2).

3. Demonstration of code applications in different contexts, encouraging data sonification in the field for accessibility and inclusion of BVI people, non-visual learners, and broader SEND[6] communities (§4, Harrison et al. (2024)).

The *"low-barrier, high-ceiling"* concept for `STRAUSS` means that the code should be easy to get started with for a novice (new to python and/or sonification as a concept), but providing ample options for low-level control and customisation for the expert. Low level control is typically afforded by the modular aspect of the code itself, where parameters and settings can be manipulated easily in python. For lowering barriers to entry, some of the approaches are detailed below.

3.1. *Balancing Jupyter Notebook Integration*

One important aspect of lowering the barrier to entry for a code like `STRAUSS` is making it compatible with interactive coding environments like `Jupyter`. Jupyter notebooks allow us to program various examples that demonstrate the use of `STRAUSS` in differing contexts. These are an important aspect of `STRAUSS` development, embracing a *Tutorial Driven Development* (TDD) approach where these examples are designed to demonstrate prospective code features, and code is developed so these work as intuitively as possible. These are maintained inside the code repository[4] in the `examples` directory. We also maintain Jupyter-specific methods in the code, supporting in-line plotting and playback, intended to be efficiently embedded into a given workflow, where code cells can be run and re-run systematically to generate sonifications. Further, we have adapted more extended and pedagogical examples to the Google `Colab` platform, which allows `STRAUSS` to be run entirely in-browser. Examples of this were developed for the *Audible Universe* workshop series (Harrison et al. 2022b; Misdariis et al. 2022, 2023), and can be found at the Audio Universe website[7]; This removes barriers, like needing to setup and install python and the requisite libraries on your machine or host files, and these notebooks have been tested and used effectively and undergraduate and high-school levels.

However, it should be noted that the current accessibility status of Jupyter from a BVI perspective leaves a lot to be desired, as these are essentially inscrutable using screen readers, such that Jupyter integration only addresses points 2 and 3 above. To address point 1 it is important to maintain parallel functionality in plaintext, screen-reader friendly contexts. `STRAUSS` was used in this way by BVI astronomer Dr Enrique Pérez-Montero to communicate their own work[8] Efforts are underway to improve Jupyter accessibility and provide general guidelines on notebook accessibility (Gozman 2023).

3.2. *Presetting and Preconfiguration*

Another aspect to lowering barriers to entry is the use of presetting. For example, while the generator classes in `STRAUSS` (e.g. `Synthesizer`, `Sampler`) can be fully customised in detail (e.g. adding multiple different oscillators, filtering, detuning, looping options etc), the expected use for most people is to choose a configuration *'preset'* for the generator. For instance, the `Synthesizer` generator has a `"pitch_mapper"` preset that can be used for the common pitch sonification of 1D data series. This is designed to be easy to hear[9] and programmed with a 3 octave pitch shift range for good contrast.

Presetting also comes up elsewhere in the code, for example the audio channels con be configured in a completely custom way, but common setups like `"mono"`, `"stereo"`, `"5.1"` etc are preset.

Instances of presetting have precedent in the sort of visualisation and plotting libraries that `STRAUSS` development draws influence from - for example the choice of a pre-existing `colormap` in `matplotlib`, aiding with points 1 and 2 above, rather than configuring your own (Hunter 2007). We plan to extend this to other areas of the code, for example musical choices in the *`Score`*, avoiding the need to understand musical theory.

4. STRAUSS APPLICATIONS & FUTURE

The *Ear to the Sky* project is intended to not just provide `STRAUSS` as a tool, but also demonstrate its utility in a number of different applications (point 3 above). A primary direction for this has been the investigation of Spectral Audification (SA) approaches as a means of rapidly inspecting spectra, By converting the frequencies of electromagnetic radiation to audible frequencies of sound. This has been explored in a proof-of-concept study (Trayford et al.

[6] Special Educational Needs & Disabilities
[7] `https://www.audiouniverse.org/research/strauss/getting-started`
[8] `https://tinyurl.com/8n3t254e`.
[9] i.e. a triangle waveform balancing frequency coverage with the dominance of a fundamental frequency.

2023). We are close to a prototype for the browser-based *'listener'* application for hyperspectral (IFU) datacubes described in this study, providing a novel exploratory tool for this complex data, alongside an accessible interface and workflow for BVI people.

In addition to research applications, `STRAUSS` has been used to sonify data for outreach. In particular, the *Audio Universe: Tour of the Solar System* (AUTotSS) planetarium show used `STRAUSS` to sonify data throughout, creating the first fully BVI accessible planetarium show, with strongly positive feedback from BVI audiences, alongside broader SEND student and general audiences (Harrison et al. 2022a, 2024). We have adapted some of these sequences, alongside new ones, in virtual reality, making use of the spatial audio capabilities of the code[10]. This can act as a *'personal planetarium'* and improve accessibility and inclusion along different vectors (i.e. those who have access to a VR headset but not a local planetarium showing the *AUTotSS*). A third application is accessible educational resources. It seems reasonable that students could build similar intuition for data sonification as they do with visualisation if incorporated into education programs, particularly for children with BVI or non-visual learning preferences. We are developing specific high-school level Jupyter notebooks as a low-barrier entry point to sonifying data. These can be exported in HTML as a non-interactive, but BVI accessible teaching aid (see §3.1, e.g. Gozman 2023). Future plans aim to use `STRAUSS` to make sonifications accompanying broader school curricula.

While research, outreach and education are distinct threads of the project, there are aspects of these that we hope to converge. Clear, listenable and intuitive sonifications are desired in all three contexts. As with visualisation, we hope that good sonic representations of data can be appreciated on many levels, just as an expert and novice can get different things out of the same detailed image of a galaxy or nebula. To realise the potential of sonification as an accessible tool, it is important to evaluate how effectively they are received in these differing contexts. Ultimately, it is a combination of these threads that will lead to a more accessible and inclusive field.

Acknowledgements

JT acknowledges the support of UKRI STFC, *Early Stage Research & Development Grant*, reference ST/X004651/1. CH acknowledges the United Kingdom Research and Innovation grant (code: MR/V022830/1).

[10] Particularly, rendering to arbitrary order *Ambisonic* formats

REFERENCES

Cantrell, S. J., Walker, B. N., & Moseng, Ø. (2021). Highcharts Sonification Studio: An online, open-source, extensible, and accessible data sonification tool. In Proceedings of the Georgia Institute of Technology

Casado, J., de la Vega, G., & García, B. (2024). SonoUno development: A User-Centered Sonification software for data analysis. Journal of Open Source Software, 9(93), 5819. https://doi.org/10.21105/joss.05819

Casado, J., García, B., Gandhi, P., & Díaz-Merced, W. (2023). arXiv e-prints. Retrieved from `https://ui.adsabs.harvard.edu/abs/2023arXiv230200085C` DOI: 10.48550/arXiv.2302.00085

De La Vega, G., Exequiel Dominguez, L. M., Casado, J., & García, B. (2023). SonoUno web: an innovative user centred web interface. arXiv e-prints. DOI: 10.48550/arXiv.2302.00081

Diaz-Merced, W. (2013). Sound for the exploration of space physics data (PhD thesis). School of Computing Science, University of Glasgow. An optional note

Gozman, K. (2023). Notebooks For All: Making Jupyter Notebooks More Accessible

Harrison, C., Trayford, J., Harrison, L., & Bonne, N. (2022a). Audio universe: tour of the solar system. Astronomy and Geophysics, 63(2), 2.38-2.40. doi: `10.1093/astrogeo/atac027`

Harrison, C., Zanella, A., Bonne, N., Meredith, K., & Misdariis, N. (2022b). Audible universe. Nature Astronomy, 6, 22-23. doi: `10.1038/s41550-021-01582-y`

Harrison, C. M., Zanella, A., & Girdhar, A. (2024). Evaluation and insights from a sonification-based planetarium show intended for improving inclusivity. Communicating Astronomy with the Public, 33, 6-12

Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. Computing in Science & Engineering, 9(3), 90-95. Publisher: IEEE COMPUTER SOC. DOI: 10.1109/MCSE.2007.55

Hunter, J., & Dale, D. (2007). The matplotlib user's guide. Matplotlib 0.90. 0 user's guide

Kramer, G., Walker, B., Bonebright, T., Cook, P., Flowers, J., Miner, N., ... Tipei, S. (1999). The sonification report: Status of the field and research agenda. Report prepared for the National Science Foundation by members of the International Community for Auditory Display

Lim, F. V., Toh, W., & Nguyen, T. T. H. (2022). Multimodality in the English language classroom: A systematic review of literature. Linguistics and Education, 69, 101048. Elsevier

Misdariis, N., Özcan, E., Grassi, M., Pauletto, S., Barrass, S., Bresin, R., & Susini, P. (2022). Sound experts' perspectives on astronomy sonification projects. Nature Astronomy, 6(11), 1249-1255

Misdariis, N., Pauletto, S., Bonne, N., Harrison, C., Meredith, K., & Zanella, A. (2023). The audible universe workshop: An interdisciplinary approach to the design and evaluation of tools for astronomical data

III Workshop for Astronomy Beyond the Common Senses for Accessibility and Inclusion (3WAI) (November 29-30, 2023)
Editors: Santiago Vargas, Beatriz García, Johana Casado, Breezy Ocaña, and Angela Pérez - DOI: https://doi.org/10.22201/ia.14052059p.2024.57.10

sonification. In International Conference on Auditory Displays

Potluri, V., Thompson, J., Devine, J., Lee, B., Morsi, N., De Halleux, P., Hodges, S., & Mankoff, J. (2022). PSST: Enabling Blind or Visually Impaired Developers to Author Sonifications of Streaming Sensor Data

Reinsch, D., & Hermann, T. (2023). sonecules: a Python Sonification Architecture

Trayford, J. W., Harrison, C. M., Hinz, R. C., Kavanagh Blatt, M., Dougherty, S., & Girdhar, A. (2023). Inspecting spectra with sound: proof-of-concept and extension to datacubes. RAS Techniques and Instruments, 2(1), 387-392. doi:10.1093/rasti/rzad021

Trayford, J. W., & Harrison, C. M. (2023). Introducing STRAUSS: A flexible sonification Python package. arXiv e-prints

Tucker Brown, J., Harrison, C. M., Zanella, A., & Trayford, J. (2022). Evaluating the efficacy of sonification for signal detection in univariate, evenly sampled light curves using ASTRONIFY. Monthly Notices of the Royal Astronomical Society, 516(4), 5674-5683

Worrall, D., Bylstra, M., Barrass, S., & Dean, R. (2007). SoniPy: The design of an extendable software framework for sonification research and auditory display. In Proceedings of the International Conference on Auditory Display (ICAD)

Zanella, A., Harrison, C. M., Lenzi, S., Cooke, J., Damsma, P., & Fleming, S. W. (2022). Sonification and sound design for astronomy research, education and public engagement. Nature Astronomy, 6, 1241-1248

Zhang, Y., & Zhao, Y. (2015). Astronomy in the big data era. Data Science Journal, 14, 11-11