The Hirajoshi Wave

Methodological Note - Version 1.1 - August 2026

Purpose. This visualizer links musical pitch, periodic motion, phase, and recurrence without conflating audible acoustic frequency with the visibly animated pendulum frequency. The system is designed for classroom exploration, repeatable evidence capture, and transparent numerical inspection.

1. Scientific model

The default C Hirajoshi software preset uses C4, D4, Eb4, G4, and Ab4 under 12-tone equal temperament with A4 = 440 Hz. Frequency is calculated from MIDI note number n as f = 440 x 2^((n - 69)/12). The application exposes the exact audible frequency for every active note.

A macroscopic pendulum cannot visibly oscillate at hundreds of cycles per second. The visual layer therefore divides every audible frequency by the same power-of-two divisor, 1024. Because the divisor is common to all notes, frequency ratios are preserved exactly even though the absolute time scale is altered.

For each visible frequency f_v, the ideal small-angle pendulum length is reported as L = g / (2 pi f_v)^2. These modeled lengths describe the visible-motion layer only. They are not claims about physical pendula that produce the corresponding audible pitches.

Note	Audible Hz	Visible Hz	Visible period (s)
C4	261.6256	0.255494	3.9137
D4	293.6648	0.286782	3.4870
Eb4	311.1270	0.303835	3.2913
G4	391.9954	0.382808	2.6123
Ab4	415.3047	0.405571	2.4657

2. Motion, phase, and recurrence

Each sphere follows an analytic angular trajectory theta_i(t) = A cos(2 pi f_i t + phi_i). Its visible Cartesian position is derived from its modeled visual length. This direct analytic update is intentional: it prevents numerical integration error from changing the musical ratios or center-crossing times.

Rapier3D integration

Rapier3D is present as the project physics layer and mirrors each analytic sphere with a kinematic rigid body and collider. The physics world is stepped with the rendered scene, but it does not overwrite the analytically calculated position in version 1.1. This separation preserves the exact pedagogical frequency model while making the project ready for later collision, grabbing, damping, and coupled-motion experiments.

Center-crossing audio

A musical event is generated when a sphere changes sign around the center line. Simultaneous repeated spheres in the same note family are grouped to prevent artificial gain multiplication. A short duplicate-trigger guard and sphere-count normalization keep 25-sphere configurations from becoming dramatically louder than 5-sphere configurations.

Recurrence

Equal-tempered intervals generally do not yield a short exact common period. The visualizer therefore reports a numerical near-recurrence using a phase-coherence threshold. The threshold is visible to the learner, and the output is labeled approximate rather than exact.

> **Interpretive constraint. A visually striking return does not itself establish an exact mathematical recurrence. Students should record the threshold, elapsed model time, and configuration when using recurrence as evidence.**

3. Rendering and high-fidelity audio

Three.js scene

Version 1.1 uses a local Three.js module for WebGL rendering. The scene includes a curved suspension rail, physically based sphere materials, high-resolution sphere meshes, lighting, shadows, additive halos, trails, a center reference plane, labels, orbit controls, adaptive pixel ratio, and a preserved drawing buffer for evidence-frame capture. A reduced-motion mode lowers trail density and render load.

Audio architecture

The sound engine is built directly on the browser Web Audio API. Audio activation requires an explicit user action, matching browser autoplay rules. Each voice is routed through stereo positioning, a dry path, a synthesized stereo room impulse, dynamics compression, and a master level. The system never changes a note frequency to create a timbre; timbre is produced by partial structure, transient noise, envelope, and reverberation.

Instrument set. Temple Bell is the default. Struck Glass, Marimba, Plucked String, Pure Sine, Koto-like Synthesis, and Stella Nova Layer are selectable. Every instrument has a Preview control that plays the active five-note scale before the pendulum is started.

Instrument	Synthesis intent
Temple Bell	Long-decay inharmonic modal partials; controlled transient; room tail
Struck Glass	Sparse bright inharmonic resonances; longer wet field
Marimba	Short mallet transient plus resonant bar modes
Plucked String	Broadband pluck transient plus harmonic body
Pure Sine	Single-frequency reference oscillator
Koto-like	Bridge-like transient, harmonic body, short resonant decay
Stella Nova Layer	Fundamental-focused bell plus wider secondary layer

4. Student evidence notebook

The Evidence Notebook turns the live visualizer into an inspectable classroom instrument. A student can enter a research question, hypothesis, observation, and interpretation, then capture the current visual frame together with the complete numerical configuration. The record includes scale, sphere count, instrument, tuning, elapsed modeled time, phase coherence, recurrence state, note frequencies, visible frequencies, periods, modeled lengths, and active comparative model.

Evidence workflow

1. Ask. Define a question that can be investigated with a change in scale, sphere count, timing, or mathematical model. **2. Predict.** State a hypothesis before changing the system. **3. Observe.** Run the visualizer and record what is seen and heard. **4. Capture.** Store a frame and numerical state together. **5. Interpret.** Explain whether the evidence supports, complicates, or fails to resolve the hypothesis. **6. Export.** Save JSON for complete machine-readable evidence, CSV for tabular analysis, or generate the bordered printable report.

Evidence records are session data in version 1.1 and are not written to the public annotation log. This prevents student evidence from being confused with public scholarly commentary and avoids replacing existing server-side annotations during an installer update.

Comparative models

The same active scale is carried into ten comparative views where appropriate: pendulum phase and recurrence, Lissajous figures, harmonic and subharmonic series, beats and interference, coupled oscillators, standing waves, Fourier construction, a Chladni/cymatic mathematical approximation, orbital resonance, and polyrhythmic cycle wheels. Approximation and analogy are explicitly labeled when the model is not a literal physical simulation.

5. Reproducibility, caching, and limitations

Cache control

All application URLs are emitted with a semantic release version and file modification token. Static assets are configured for long immutable caching because a changed file receives a changed URL; PHP and JSON endpoints are explicitly no-store. The local Three.js and Rapier module URLs are also versioned. This prevents a browser from silently mixing JavaScript or CSS from different releases.

Local vendor policy

The runtime imports Three.js and Rapier only from the project assets/vendor directory. The release pins Three.js 0.185.1 and @dimforge/rapier3d-compat 0.20.0. If a pinned vendor file is absent after installation, the server bootstrap retrieves that exact version into the local vendor directory before the browser import map is emitted. There is no browser-side CDN fallback.

Limitations

The visible pendula are a time-scaled mathematical representation, not acoustic oscillators. Rapier mirrors rather than controls the exact-frequency motion in this release. The Web Audio instruments are synthesis models rather than sampled reconstructions of specific historical instruments. The named scale presets are operational software definitions and should not be treated as a claim that a single tuning or interval set exhausts the historical or regional use of a named musical scale.

Reproducibility. A valid student result should identify the project version, scale/root, tuning reference, sphere count, motion mode, instrument, speed, recurrence threshold when relevant, and the captured numerical state. The evidence export preserves these fields.

6. References and standards

Fletcher, N. H., & Rossing, T. D. (1998). *The Physics of Musical Instruments* (2nd ed.). Springer. [Acoustics, resonance, musical-instrument physics.]

Strogatz, S. H. (2015). *Nonlinear Dynamics and Chaos* (2nd ed.). Westview Press. [Oscillation, phase, synchronization, recurrence concepts.]

Sethares, W. A. (2005). *Tuning, Timbre, Spectrum, Scale* (2nd ed.). Springer. [Relations among tuning, spectrum, and perceived musical structure.]

National Research Council. (2012). *A Framework for K-12 Science Education: Practices, Crosscutting Concepts, and Core Ideas*. National Academies Press. https://doi.org/10.17226/13165. [Evidence-centered scientific practice.]

W3C. (2021). *Web Audio API*. W3C Recommendation. [Browser audio scheduling and processing model.]

Three.js project. *Three.js documentation and source*, pinned application dependency 0.185.1. [WebGL scene rendering.]

Dimforge. *Rapier JavaScript documentation and source*, pinned application dependency @dimforge/rapier3d-compat 0.20.0. [Rigid-body physics layer.]

Version note

Methodological Note v1.1 accompanies The Hirajoshi Wave repair update from managed version 1.1.0 to 1.1.1. Scientific behavior remains anchored to the original distinction between exact audible frequencies and ratio-preserving visible motion; this release upgrades rendering, local dependency management, audio inspection, cache correctness, and student evidence collection.