

Resources

Daniel Ari Friedman
ORCID: 0000-0001-6232-9096
Email: daniel@activeinference.institute

August 14, 2025

Resources (References and Further Reading)

Quadrays and Synergetics (core starting points)

- **Quadray coordinates (intro and conversions):** Urner – Quadray intro, Urner – Quadrays and XYZ
- **Synergetics background and IVM:** Synergetics (Fuller, overview)

4dsolutions (Kirby Urner) — repositories and key artifacts

- **Organization overview:** 4dsolutions (GitHub org) — Python-centered explorations of Quadrays and synergetic geometry.
- **Math for Wisdom (m4w):** m4w (repo)
 - Quadray vectors and conversions: `qrays.py` (Qvector, SymPy-aware)
 - Synergetic tetravolumes and modules: `tetravolume.py` - PdF/CM vs native IVM, BEAST
- **School_of_Tomorrow (notebooks/code):** School_of_Tomorrow (repo)
 - Tom Ace 5×5 determinant: `Qvolume.ipynb`
 - Bridging vs native tetravolumes: `VolumeTalk.ipynb`
- **Historical variants:** `qrays.py` also appears in Python5 (archive).

Context: These materials popularize the IVM/CCP/FCC framing of space, integer tetravolumes, and projective Quadray normalization. They inform the methods in this paper and complement the `src/` implementations (see `quadray.py`, `cayley_menger.py`, `linalg_utils.py`).

Comprehensive index of 4dsolutions artifacts (selected)

Primary hub: School_of_Tomorrow (Python + notebooks)

- **Repository:** School_of_Tomorrow

- **Core modules:**
 - `grays.py`: Quadray implementation with normalization, conversions, and vector ops (`grays.py` source — [School_of_Tomorrow](#))
 - `quadcraft.py`: POV-Ray scenes for CCP/IVM arrangements, animations, and tutorials (`quadcraft.py` source — [School_of_Tomorrow](#))
 - `flextegrity.py`: Polyhedron framework, concentric hierarchy, POV-Ray export (`flextegrity.py` source — [School_of_Tomorrow](#))
 - Additional: `polyhedra.py`, `identities.py`, `smod_play.py` (synergetic modules)
- **Notebooks:**
 - `QuadCraft_Project.ipynb`: Interactive tutorials; CCP navigation and tetra demos (`QuadCraft_Project.ipynb` — [School_of_Tomorrow](#))
 - `Qvolume.ipynb`: Tom Ace 5×5 determinant; random-walk IVM volumes (`Qvolume.ipynb` — [School_of_Tomorrow](#))
 - `VolumeTalk.ipynb`: Bridging (CM/PdF) vs native (Ace/GdJ) tetravolumes (`VolumeTalk.ipynb` — [School_of_Tomorrow](#))
 - `TetraBook.ipynb`, `CascadianSynergetics.ipynb`, `Rendering_IVM.ipynb`, `SphereVolumes.ipynb` (visual and curricular materials)

Additional repositories

- **tetravolumes**: algorithms and pedagogy for tetra volumes
 - Repo: `tetravolumes`
 - Code: `tetravolume.py`
 - Notebooks: `Atoms R Us.ipynb`, `Computing Volumes.ipynb`
- **rusty_rays**: Rust port highlighting cross-language consistency
 - Repo: `rusty_rays`
 - Sources: Rust library implementation, Rust command-line interface
- **synmods**: Clojure/functional approach to Quadrays and synergetic modules
 - Repo: `synmods`
 - Sources: `grays.clj`, `ramping_up.clj`
- **BookCovers**: VPython for interactive educational animations
 - Repo: `BookCovers`
 - Examples: `bookdemo.py`, `stickworks.py`, `tetravolumes.py`

Additional educational resources

- **Oregon Curriculum Network (OCN)**: OCN portal
- **Python for Everyone**: `pymath` page
- **Python5 notebooks**: Polyhedrons 101.ipynb

Media and publications

- **YouTube demonstrations**: Synergetics talk 1, Synergetics talk 2, Additional
- **Academia profile**: Kirby Urner at [Academia.edu](#)

- **Fuller Institute:** BFI — Big Ideas: Synergetics

Background and community materials

- **RW Gray projects — Synergetics text:** rwgrayprojects.com (synergetics)
- **Fuller FAQ:** C. J. Fearnley's Fuller FAQ
- **Synergetics resource list:** C. J. Fearnley's resource page
- **Wikieducator:** Synergetics hub
- **Quadray animation:** [Quadray.gif](#) (Wikimedia Commons)

Geometry and volumes (Coxeter.4D context)

- **Regular polytopes (Euclidean E):** H. S. M. Coxeter, Regular Polytopes (Dover ed.), p. 119 clarifies Euclidean 4D vs spacetime.
- **Sphere packings and lattices:** J. H. Conway & N. J. A. Sloane, Sphere Packings, Lattices and Groups (Springer)
- **Cayley–Menger determinant:** Cayley–Menger determinant (reference)
- **Tetrahedron — volume:** Tetrahedron: volume (reference)
- **Bareiss algorithm (exact determinants):** Bareiss algorithm (reference)

Optimization and information geometry

- **Nelder–Mead method:** Nelder–Mead (reference)
- **Fisher information:** Fisher information (reference) — see also Eq. (??)
- **Natural gradient:** Natural gradient (reference) — see Eq. (??)

Active Inference

- **Free energy principle:** Free energy principle (reference)
- **Comprehensive review:** Active Inference — recent review (UCL Discovery, 2023)

Community discussions and context

- **Math4Wisdom:** IVM XYZ conversions (curated page)
- **synergeo (groups.io):** Synergetics discussion archive
- **GeodesicHelp:** Geodesic computations archive (Google Groups)

Related projects and applications

- **QuadCraft:** Tetrahedral voxel engine using Quadrays
- **Flextegrity:** Generating the Flextegrity Lattice (academia.edu)

Tooling

- **GCC libquadmath (binary128)**: Official GCC libquadmath documentation

Cross-language and cross-platform validation

- **Rust** (`rusty_rays`) and **Clojure** (`synmods`) mirror the Python algorithms for vector ops and tetravolumes, serving as independent checks on correctness and performance comparisons.
- **POV-Ray** (`quadcraft.py`) and **VPython** (**BookCovers**) demonstrate rendering pipelines for CCP/IVM scenes and educational animations.