Appendix: Free Energy and Active Inference

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

August 14, 2025

Contents

1 Appendix: The Free Energy Principle and Active Inference

1.1 Overview

The Free Energy Principle (FEP) posits that biological systems maintain their states by minimizing variational free energy, thereby reducing surprise via prediction and model updating. Active Inference extends this by casting action selection as inference under prior preferences. Background: see the concise overview on the Free energy principle and the monograph Active Inference (MIT Press).

This appendix emphasizes relationships among: (i) the four-fold partition of Active Inference, (ii) Quadrays (Fuller.4D) as a geometric scaffold for mapping this partition, and (iii) information-geometric flows (Einstein.4D analogy) that underpin perception–action updates. For the naming of 4D namespaces used throughout—Coxeter.4D (Euclidean E4), Einstein.4D (Minkowski spacetime analogy), Fuller.4D (Synergetics/Quadrays)—see `02_4d_namespaces.md`.

1.2 Mathematical Formulation and Equation Callouts (Equations linkage)

- Variational free energy (discrete states) — see Eq. (**??**) in the Equations appendix, implemented by `free_energy`:

$$\mathcal{F} = -\log P(o \mid s) + \mathrm{KL}\big[Q(s) \parallel P(s)\big] \tag{1}$$

  where $Q(s)$ is a variational posterior, $P(s)$ a prior, and $P(o \mid s)$ the likelihood. Lower $\mathcal{F}$ is better.

- Fisher Information Matrix (FIM) as metric — see Eq. (**??**) and `fisher_information_matrix`:

$$F_{i,j} = \mathbb{E}\left[\partial_{\theta_i} \log p(x;\theta)\ \partial_{\theta_j} \log p(x;\theta)\right]. \tag{2}$$

- Natural gradient descent under information geometry — see Eq. (**??**) and `natural_gradient_step`; overview: Natural gradient:

$$\theta \leftarrow \theta - \eta \, F(\theta)^{-1} \, \nabla_\theta L(\theta). \tag{3}$$

Figures: Figure **??**, Figure **??**, Figure **??**, Figure **??**.

Discrete variational optimization on the quadray lattice: `discrete_ivm_descent` greedily descends a free-energy-like objective over IVM moves, yielding integer-valued trajectories. See the path animation artifact `discrete_path.mp4` in `quadmath/output/`.

1.3 Four-Fold Partition and Tetrahedral Mapping (Quadrays; Fuller.4D)

Active Inference partitions the agent–environment system into four coupled states:

- Internal (μ) — agent's internal states
- Sensory (s) — observations
- Active (a) — actions
- External (ψ) — latent environmental causes

See, for an overview of this partition and generative process formulations, the Active Inference review and the general entry on Active inference.

Tetrahedral mapping via Quadrays (Fuller.4D): assign each state to a vertex of a tetrahedron, using Quadray coordinates `(A,B,C,D)` with non-negative components and at least one zero after normalization. One canonical mapping is `A \leftrightarrow Internal (\mu)`, `B \leftrightarrow Sensory (s)`, `C \leftrightarrow Active (a)`, `D \leftrightarrow External (\psi)`. The edges capture the pairwise couplings (e.g., `\mu\text{--}s` for perceptual inference; `a\text{--}\psi` for control). Integer tetravolume then quantifies the "coupled capacity" region spanned by jointly feasible states in a time slice; see `Quadray` and tetravolume methods in `03_quadray_methods.md`.

Interpretation note: this Quadray-based mapping is a didactic geometric scaffold. It is not standard in the Active Inference literature, which typically develops the four-state partition in probabilistic graphical terms. Our use highlights structural symmetries and discrete volumetric quantities available in Fuller.4D, building on the computational foundations developed in the 4dsolutions ecosystem for tetrahedral modeling and volume calculations.

Code linkage (no snippet): see `example_partition_tetra_volume` in `src/examples.py` and Figure 1.

1.4 How the 4D namespaces relate here

- Fuller.4D (Quadrays): geometric embedding of the four-state partition on a tetrahedron; integer tetravolumes and IVM moves provide discrete combinatorial structure.
- Coxeter.4D (Euclidean E4): exact Euclidean measurements (e.g., Cayley–Menger determinants) for tetrahedra underlying volumetric comparisons and scale relations.
- Einstein.4D (Minkowski analogy): information-geometric flows (natural gradient, metric-aware updates) supply a continuum picture for perception–action dynamics.

The three roles are complementary: Fuller.4D encodes partition structure, Coxeter.4D provides exact metric geometry for static comparisons, and Einstein.4D guides dynamical descent.

1.5 Joint Optimization in the Tetrahedral Framework (Methods linkage)

- Perception: update μ to minimize prediction error on s under the generative model (descending $\nabla_\mu F$).
- Action: select a that steers ψ toward preferred outcomes (descending $\nabla_a F$).

Continuous-time flows (Einstein.4D analogy for metric/geodesic intuition): see `perception_update` and `action_update` in `src/information.py`. Discrete Quadray moves connect to these flows via greedy descent on a local free-energy-like objective; see `discrete_ivm_descent` in `src/discrete_variational.py` and the path artifacts in `quadmath/output/`.

1.6 Neuroscience and Predictive Coding

Under Active Inference, cortical circuits minimize free energy through recurrent exchanges of descending predictions and ascending prediction errors, aligning with predictive coding accounts. See the neural dynamics framing in Active Inference neural dynamics (arXiv:2001.08028).

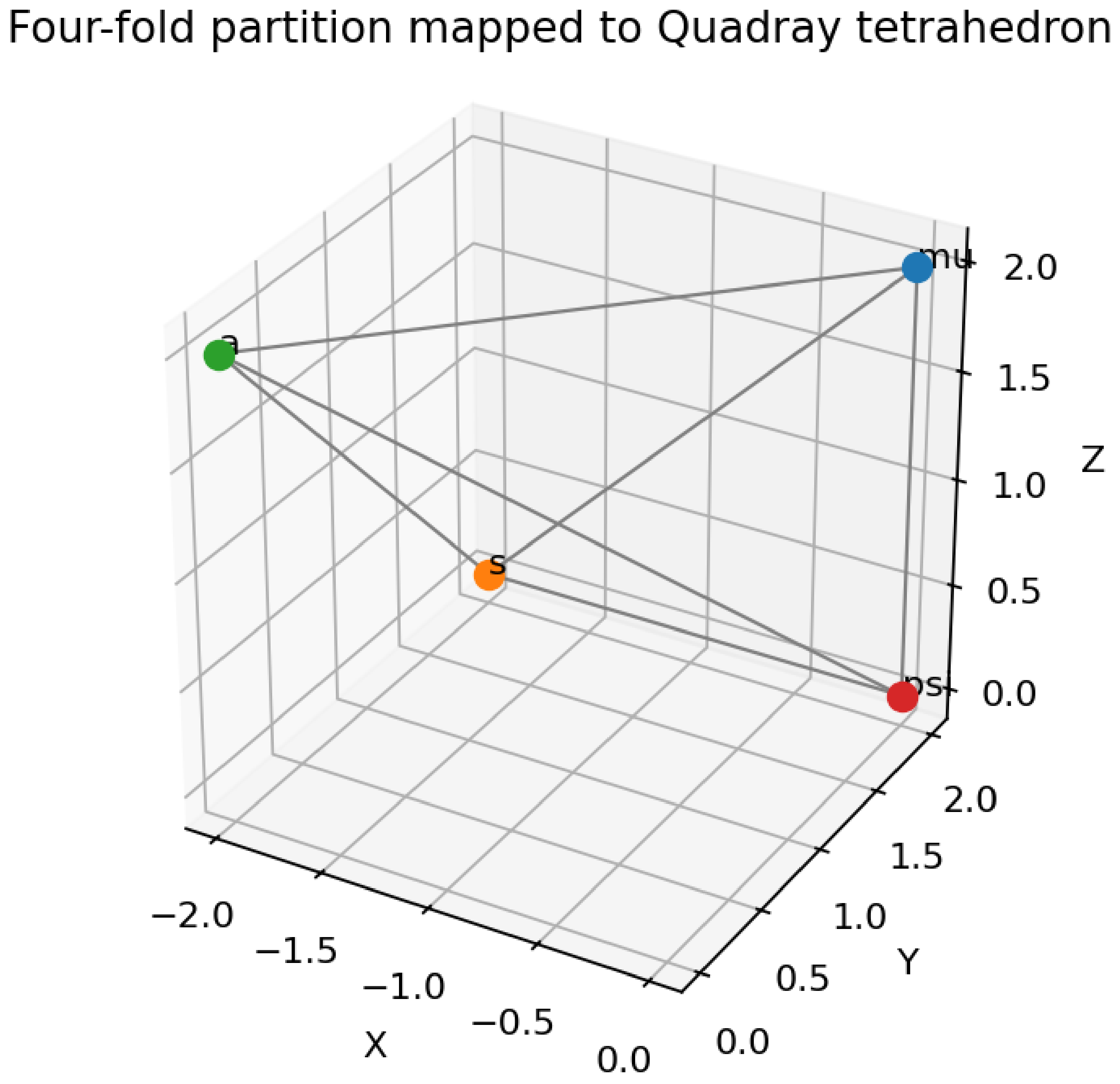

Figure 1: Four-fold partition mapped to a Quadray tetrahedron — vertices labeled as Internal (μ), Sensory (s), Active (a), External (ψ). Edges depict pairwise couplings: (μ–s) perceptual inference, (a–ψ) control, and cross-couplings capturing active perception and sensorimotor contingencies.

1.7 Relation to Reinforcement Learning and Control

Active Inference replaces explicit value functions with prior preferences over outcomes and transitions, balancing exploration (epistemic value) and exploitation (pragmatic value) via expected free energy. See Active Inference and RL (arXiv:2002.12636). Connections to optimal control arise when minimizing expected free energy plays the role of a control objective; cf. Optimal control.

1.8 Links to Other Theories

- Bayesian Brain hypothesis: Bayesian brain
- Predictive Coding: Predictive coding
- Information Geometry: Fisher information, Natural gradient

1.9 Implications for AI and Robust Computation

FEP/Active Inference provide algorithms that unify perception and action under uncertainty, offering biologically plausible alternatives to standard RL with adaptive exploration and robust decision-making. See applications in AI (arXiv:1907.03876).

1.10 Code, Reproducibility, and Cross-References

– Equation references: Eq. (Free Energy), Eq. (FIM), Eq. (Natural Gradient) in `08_equations_appendix.md`.
– Code anchors (for readers who want to run experiments): `free_energy`, `fisher_information_matrix`, `natural_gradient_step`, `perception_update`, `action_update`, and `discrete_ivm_descent` in `src/information.py` and `src/discrete_variational.py`.

Demo and figures generated by `quadmath/scripts/information_demo.py` output to `quadmath/output/`:

- **Visualizations**: `fisher_information_matrix.png`, `fisher_information_eigenspectrum.png`, `natural_gradient_pat` `free_energy_curve.png`, `partition_tetrahedron.png`.
- **Raw data**: `fisher_information_matrix.csv`, `fisher_information_matrix.npz` (F, grads, X, y, w_true, w_est), `fisher_information_eigenvalues.csv`, `fisher_information_eigensystem.npz`.
- **External validation**: Cross-reference with volume calculations in `Qvolume.ipynb` and tetrahedral modeling tools from the 4dsolutions ecosystem.