The Geometry of Motion and the Architecture of Time

From the Rod and the Spinning Top Thought Experiment to Nested Angular Progress and a Galactic Chronometric Atlas

S. M. H. Emamifar

Independent Research Collaboration on
Black Hole and Cosmology Concepts (IRCBHC)

ORCID: 0009-0007-6257-0163

July 2026

Abstract

This work joins two constructions that were originally developed separately. The first is the Rod and the Spinning Top Thought Experiment, which asks whether the geometry of an observed trajectory uniquely identifies the mechanism that produced it. A gravitational circular orbit and a rigid rotational constraint can both produce curved motion, although their angular rates, force transmission, energies, and angular momenta obey different laws. The thought experiment therefore establishes the methodological distinction

$$\text{trajectory} \neq \text{mechanism}.$$

The second construction is a gravity-independent kinematic chronometric architecture. A signed, unwrapped ledger records spin, orbital, parent-system, and Galactic phases. These phases are embedded geometrically rather than added as bare scalar angles. If

$$\boldsymbol{x}_P = \boldsymbol{x}_P(q^1, \ldots, q^n), \qquad \mathsf{G}^{(P)}_{ab} = \frac{\partial \boldsymbol{x}_P}{\partial q^a} \cdot \frac{\partial \boldsymbol{x}_P}{\partial q^b},$$

then the phase-generated speed and flat-spacetime kinematic proper time are

$$v_P^2 = \mathsf{G}^{(P)}_{ab} \dot{q}^a \dot{q}^b, \qquad \frac{\mathrm{d}\tau_P}{\mathrm{d}t} = \sqrt{1 - \frac{v_P^2}{c^2}}.$$

The mechanical thought experiment supplies a transparent example of a possible dynamical closure, while the phase metric supplies the independent map from angular progression to velocity and clock rate.

For comparison across scales, the local dimensionless chronometric label is

$$\mathscr{K}_P = \gamma_P \frac{v_P^2}{c^2} = \gamma_P - \frac{1}{\gamma_P}, \qquad \gamma_P = (1 - v_P^2/c^2)^{-1/2}.$$

For a uniform rotating ring it equals the one-winding holonomy divided by the native rotation period. The canonical terrestrial equatorial ring gives

$$\mathscr{K}_\oplus \simeq 2.4068743 \times 10^{-12}, \qquad \mathcal{H}^*_\oplus = \mathscr{K}_\oplus T_{\mathrm{sid},\oplus} \simeq 207.386\,\mathrm{ns}.$$

The value 207.386 ns is therefore a terrestrial one-winding translation, not a universal quantum. The full counter-propagating Sagnac proper-time difference is twice this one-direction quantity.

A second scale-normalised signature follows by dividing the one-winding holonomy by the circumference $C = 2\pi R$:

$$\mathscr{S} \equiv \frac{c|\mathcal{H}^*|}{C} = \gamma \frac{v}{c} = \gamma\beta.$$

Thus $\mathscr{K} = |\mathcal{H}^*|/T = \gamma\beta^2$ compares a ring with its native period, while $\mathscr{S}$ compares the same holonomy with its path length. This distinction formalises the comparison between terrestrial and Galactic rotation scales without confusing a dimensional interval with a frame-declared kinematic label.

The construction is finally extended to a quality-controlled Galactocentric phase-space atlas. Each object receives a frame- and epoch-declared state

$$\mathfrak{A}_X(t_0 \mid \mathcal{F}_{\mathrm{G}}) = (t_0; \boldsymbol{x}_X, \boldsymbol{v}_X; \boldsymbol{\Sigma}_X, \mathcal{Q}_X),$$

and only a measured or explicitly model-completed velocity may be converted into $\mathscr{K}_X$, $\mathscr{S}_X$, a clock factor, or a local-observer translation. The resulting architecture separates four levels that should not be merged: dynamical closure, geometric embedding, chronometric readout, and catalogue provenance. It is not a replacement for gravitational clock calculations; it is a unified language for the kinematic organisation of nested rotation from a mechanical thought experiment to a Milky Way atlas.

Keywords: rod thought experiment; spinning top; angular constraint; nested phase ledger; Sagnac signature; kinematic proper time; dimensionless chronometric label; Galactic phase space; Local Observer Translation Map.

Epistemic status. The document deliberately separates four categories. The special-relativistic proper-time relation, vector velocity composition, rotating-ring Sagnac relations, and coordinate transformations are standard kinematics. The rod and spinning-top construction is a controlled mechanical thought experiment. The rod–flywheel capacity and spin-transfer relations are conditional closure models, not measured universal laws. The phase ledger, phase metric, chronometric labels, circumference-normalised Sagnac signature, local-observer translation protocol, and catalogue quality system form the proposed architecture. No gravitational potential or gravitational redshift is used inside the kinematic conversion, so the result is not the complete clock rate of a real gravitating system. The canonical value 207.386 ns is the terrestrial one-winding translation of $\mathscr{K}_\oplus$; the corresponding two-direction Sagnac difference is twice that value.

Contents

1 Introduction **6**

2 Mechanical Prelude: The Rod and the Spinning Top 6
2.1 The inverse problem of motion . . . 6
2.2 Gravitational circular orbit and rigid angular constraint . . . 7
2.3 The Earth–Neptune rod experiment . . . 7
2.4 Mechanical cost and dynamical non-equivalence . . . 8

3 The Connection: From Angular Constraint to Phase Architecture 8

4 Scope and Reference Frames 9
4.1 Chosen reference time . . . 9
4.2 Meaning of "gravity-independent . . . 9

5 The Nested Angular-Progress Ledger 9
5.1 Immediate-parent chain . . . 9
5.2 Signed recurrence relations . . . 10

6 Geometric Embedding of the Ledger 10
6.1 Position as a function of angular phases . . . 10
6.2 The phase metric . . . 11

7 Local Kinematic Chronometry 11
7.1 Angular-progress norm . . . 11
7.2 Weak-speed approximation . . . 12

8 Dimensionless Chronometric Labels and Local Observer Translation 12
8.1 The local dimensionless label . . . 12
8.2 Kinematic equivalence classes . . . 13
8.3 Uniform rotating ring and one-winding translation . . . 13
8.4 Canonical Earth calibration . . . 14
8.5 Earth-normalised comparison and translation . . . 14
8.6 Scale-normalised Sagnac signatures . . . 15
8.7 Illustrative terrestrial–Galactic ring comparison . . . 15

9 Galactic Phase-Space Addressing and the Chronometric Atlas 16
9.1 Declared Galactocentric frame . . . 16
9.2 Six-dimensional Galactic chronometric address . . . 16
9.3 Why position alone does not determine speed . . . 17
9.4 Nested local refinement inside a Galactic address . . . 17
9.5 Atlas label, clock factor, and Earth-language coordinate . . . 18
9.6 Catalogue quality classes . . . 18
9.7 Observational input and Gaia-compatible ingestion . . . 18
9.8 Uncertainty propagation . . . 19
9.9 Demonstration of address and velocity non-uniqueness . . . 20
9.10 Atlas construction algorithm . . . 20

10 A Planetary Specialisation **21**
10.1 Centre motion plus body rotation . . . 21
10.2 Planar circular approximation . . . 21

11 Any Point on or Above a Rotating Planet **22**
11.1 Latitude and altitude . . . 22
11.2 Full-rotation average . . . 22
11.3 Difference between two altitudes . . . 23

12 Multiple Nested Motions **23**
12.1 General two-phase form . . . 23
12.2 Moon, planet, star, and galactic layers . . . 23
12.3 Non-commuting axes . . . 24

13 Conditional Rod–Flywheel Closure **24**
13.1 Role of the closure . . . 24
13.2 Radial-capacity relation . . . 24
13.3 Spin-transfer law . . . 25
13.4 Finite propagation and the limit of the rigid-rod picture . . . 25
13.5 Insertion into the chronometric map . . . 25

14 Earth as a Numerical Demonstration **25**
14.1 Reference values . . . 25
14.2 Orbital centre motion . . . 26
14.3 Sea level and a higher co-rotating point . . . 27
14.4 Latitude dependence . . . 27
14.5 Illustrative rescaled-Earth translation . . . 27

15 Natural Celestial Examples in Ledger Form **28**
15.1 Mars . . . 28
15.2 Venus . . . 28
15.3 The Moon . . . 28
15.4 Differentially rotating bodies . . . 28

16 Operational Calculation Protocol **28**

17 What Is Established and What Remains Conditional **29**
17.1 Established within the stated kinematic framework . . . 29
17.2 Not established by the chronometric construction . . . 30

18 Discussion **31**

19 Conclusion **32**

A Derivation in Turn-Count Coordinates **33**

B Exact Two-Phase Example **33**

C Relation to a Rotating-Ring Holonomy **34**

D Stable Numerical Formulae **34**

E Reference Python Implementations **35**

F On the Novelty, Architecture, and Applications of the Framework **54**
F.1 Standard building blocks, novel architecture 54
F.2 What the framework is not 55
F.3 Intended applications 55
F.4 Epistemic hierarchy of quantities 55

G Galactic Reference Chronometry and Local Observer Translation **56**
G.1 Galactic reference state 56
G.2 Frame-declared dimensionless kinematic label 56
G.3 Local observer translation 56

1 Introduction

Natural clocks are read from recurrent motion, yet an observed path contains less information than is often assumed. From a trajectory $\boldsymbol{x}(t)$ one may calculate velocity and acceleration, but the inverse map

$$\boldsymbol{x}(t) \longrightarrow \text{physical cause}$$

is generally non-unique. A circular trajectory may be maintained by a field, a material constraint, electromagnetic confinement, or another mechanism capable of supplying the same instantaneous inward acceleration.

This observation provides the entrance to the chronometric problem. Motion has at least three logically distinct descriptions:

(1) a *dynamical layer*, which determines why particular rates occur;

(2) a *geometric layer*, which composes nested phases into a world line;

(3) a *chronometric layer*, which converts the resulting velocity into a local kinematic clock factor and comparison label.

A fourth catalogue layer records the reference frame, epoch, uncertainty, and whether the velocity was measured or supplied by a closure model.

The unified chain developed here is

$$\boxed{\begin{array}{c}\text{measured rates or declared dynamical closure} \longrightarrow \text{signed nested phase ledger} \\ \longrightarrow \text{geometric embedding and phase metric} \longrightarrow \text{resultant point velocity} \\ \longrightarrow \begin{cases}\text{kinematic proper time,} \\ \text{dimensionless labels and Sagnac signatures,} \\ \text{frame-declared Galactic atlas record.}\end{cases}\end{array}} \tag{1}$$

The rod and spinning-top experiment is therefore not an ornamental analogy. It is the simplest explicit demonstration of why the first arrow in equation (1) must remain separate from the arrows that follow.

2 Mechanical Prelude: The Rod and the Spinning Top

2.1 The inverse problem of motion

Let the directly observed quantity be the trajectory

$$\boldsymbol{x}(t). \tag{2}$$

It determines

$$\boldsymbol{v}(t) = \dot{\boldsymbol{x}}(t), \qquad \boldsymbol{a}(t) = \ddot{\boldsymbol{x}}(t), \tag{3}$$

but does not by itself identify the interaction or constraint that generated that acceleration. Symbolically,

$$\boxed{\boldsymbol{x}(t) \not\Rightarrow \text{unique dynamics.}} \tag{4}$$

This is the methodological content of the thought experiment and is closely related to the general inverse-problem caution emphasised in classical discussions of mechanics [19].

2.2 Gravitational circular orbit and rigid angular constraint

For a Newtonian circular orbit of a body of mass m around a central mass M [17, 18],

$$\frac{GMm}{r^2} = m\omega^2 r, \tag{5}$$

so

$$\boxed{\omega_{\text{grav}}^2 = \frac{GM}{r^3}, \qquad v_{\text{grav}} = \sqrt{\frac{GM}{r}}.} \tag{6}$$

Therefore $\omega_{\text{grav}} \propto r^{-3/2}$ and $v_{\text{grav}} \propto r^{-1/2}$.

Now replace the field explanation with an idealised rod connected to one rotating axis. The defining constraint is

$$\boxed{\omega_{\text{rod}}(r) = \Omega, \qquad v_{\text{rod}}(r) = \Omega r, \qquad F_{\text{rod}} = m\Omega^2 r.} \tag{7}$$

The circular geometry can look similar, but the radial scalings and mechanism are different:

$$\boxed{\text{similar trajectory} \neq \text{identical dynamics.}} \tag{8}$$

2.3 The Earth–Neptune rod experiment

Take two terminal locations at

$$r_N \simeq 30 r_E. \tag{9}$$

If a single idealised rod forces both locations to share Earth's annual angular rate, then

$$\omega_N = \omega_E = \Omega, \qquad \boxed{\frac{v_N}{v_E} = \frac{r_N}{r_E} \simeq 30.} \tag{10}$$

The outer terminal must therefore possess a much larger tangential speed even though the two terminals complete the same number of turns.

A second version of the experiment fixes a radial transport capacity $F_\star$:

$$F_\star = m\Omega^2 r. \tag{11}$$

At fixed $F_\star$ and Ω,

$$mr = \text{constant}. \tag{12}$$

Thus the Neptune terminal requires

$$\boxed{\frac{m_N}{m_E} = \frac{r_E}{r_N} \simeq \frac{1}{30}.} \tag{13}$$

For equal mean density, $m \propto R^3$, and therefore

$$\boxed{\frac{R_N}{R_E} = \left(\frac{1}{30}\right)^{1/3} \simeq 0.322.} \tag{14}$$

This result is conditional on the declared capacity law. It is not a property of the actual Earth–Neptune system.

2.4 Mechanical cost and dynamical non-equivalence

For a gravitational circular orbit,

$$E_{\text{grav}} = -\frac{GMm}{2r}, \qquad L_{\text{grav}} = m\sqrt{GMr}, \tag{15}$$

whereas the rigidly constrained terminal has

$$E_{\text{rod}} = \frac{1}{2}m\Omega^2 r^2, \qquad L_{\text{rod}} = m\Omega r^2. \tag{16}$$

Hence

$$E_{\text{grav}} \propto r^{-1}, \quad L_{\text{grav}} \propto r^{1/2}, \qquad E_{\text{rod}} \propto r^2, \quad L_{\text{rod}} \propto r^2. \tag{17}$$

A common path shape therefore does not imply common force transmission, energy architecture, or angular-momentum scaling.

3 The Connection: From Angular Constraint to Phase Architecture

The rod experiment imposes angular coherence directly. Every terminal shares the same unwrapped angular progression

$$q(t) = q(t_0) + \Omega(t - t_0), \tag{18}$$

while its tangential speed depends on the embedding radius. This is precisely the distinction required by the phase-ledger construction: the turn count is not yet a clock reading, and the angular rate is not yet a velocity until the geometry is supplied.

The unified state of a point is therefore represented schematically by

$$\boxed{\mathfrak{S}_P = \big(\boldsymbol{q}_P, \dot{\boldsymbol{q}}_P; \boldsymbol{x}_P(\boldsymbol{q}_P), \boldsymbol{v}_P; \mathcal{F}, t_0, \mathcal{Q}_P\big).} \tag{19}$$

The rod or any other dynamical model may supply $\dot{\boldsymbol{q}}_P$. The geometric embedding supplies $\boldsymbol{v}_P$. The chronometric map supplies $\mathrm{d}\tau_P/\mathrm{d}t$, $\mathscr{K}_P$, and the Sagnac-derived signatures. The catalogue layer records how each input was obtained.

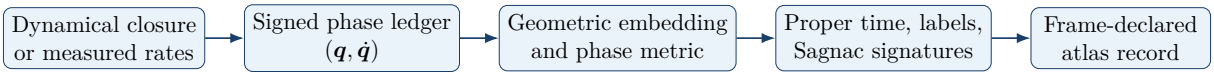

Figure 1: Engineered connection between the mechanical thought experiment and the chronometric atlas. Each layer has a distinct epistemic role.

4 Scope and Reference Frames

4.1 Chosen reference time

Let t denote the coordinate time of a specified quasi-inertial frame over the domain of the calculation. For Solar-System applications this may be an idealised barycentric inertial frame or a sufficiently accurate local inertial approximation. The speed of a point P in that frame is

$$\boldsymbol{v}_P(t) = \frac{\mathrm{d}\boldsymbol{x}_P}{\mathrm{d}t}. \tag{20}$$

In the flat-spacetime kinematic sector,

$$\boxed{\mathrm{d}\tau_P = \mathrm{d}t\sqrt{1 - \frac{v_P^2}{c^2}}.} \tag{21}$$

Equation (21) is the only clock law assumed in the main derivation.

For Galactic applications, the term "Galactocentric frame" does not denote one universally fixed modern standard. The origin, axis orientation, Solar position, Solar velocity, epoch, and transformation parameters must therefore be stated explicitly. The atlas developed below treats this parameter set as part of the address rather than hiding it in software defaults.

4.2 Meaning of "gravity-independent

No term of the form U/c^2, $GM/(rc^2)$, or any metric-specific gravitational potential is introduced. The word "gravity-independent" therefore means that the construction is a self-contained kinematic sector. It does not mean that gravitational effects are physically absent from actual astronomical clocks.

When two real clocks occupy different gravitational potentials, their total observed rate difference generally contains both kinematic and gravitational components. The present paper isolates the former so that the rotational and orbital geometry can be studied without reversing, reinterpreting, or absorbing it into a gravitational term.

5 The Nested Angular-Progress Ledger

5.1 Immediate-parent chain

Let P be a material point, surface marker, atmospheric parcel, or ideal clock associated with a natural celestial body. Introduce an immediate-parent chain

$$P \longrightarrow B_0 \longrightarrow B_1 \longrightarrow \cdots \longrightarrow B_m \longrightarrow \mathcal{F}_{\mathrm{ref}}, \tag{22}$$

where B_0 is the host body and $\mathcal{F}_{\text{ref}}$ is an operational orientation frame, such as the ICRF when appropriate [3, 4].

Each link has an oriented phase q^a and an unwrapped turn count N_a:

$$q^a(t) = q^a(t_0) + 2\pi N_a(t), \qquad N_a(t) \in \mathbb{R}. \tag{23}$$

The sign of $\dot{N}_a$ records prograde or retrograde progression.

Definition 1 (Nested angular-progress ledger)**.** *The ledger of point P is*

$$\boxed{\boldsymbol{N}_P(t) = \Big(N_1(t), N_2(t), \ldots, N_n(t)\Big).} \tag{24}$$

Its rate vector is

$$\boxed{\dot{\boldsymbol{N}}_P(t) = \Big(\nu_1(t), \ldots, \nu_n(t)\Big), \qquad \nu_a = \dot{N}_a = \frac{\omega_a}{2\pi}.} \tag{25}$$

The ledger is an address in nested cycle space. It is not a clock reading and its entries need not be commensurate.



Figure 2: Immediate-parent phase chain. The phase at each level is defined relative to the next parent rather than being referred ambiguously to one common origin.

5.2 Signed recurrence relations

In a planar common-axis approximation, unwrapped angles add and subtract exactly. If a body has signed sidereal spin rate ω_{sid} and orbital phase rate ω_{orb}, the apparent solar rate is

$$\omega_{\text{sol}} = \omega_{\text{sid}} - \omega_{\text{orb}}. \tag{26}$$

Using signed periods P^*,

$$\boxed{\frac{1}{P^*_{\text{sol}}} = \frac{1}{P^*_{\text{sid}}} - \frac{1}{P_{\text{orb}}}.} \tag{27}$$

Synodic recurrence and spin–orbit locking are analogous differences between selected ledger components. In three dimensions, however, finite rotations must be composed as ordered products in SO(3) rather than as scalar angle sums.

6 Geometric Embedding of the Ledger

6.1 Position as a function of angular phases

Let the spatial position of point P be a differentiable function of the angular coordinates:

$$\boxed{\boldsymbol{x}_P = \boldsymbol{x}_P(q^1, q^2, \ldots, q^n; t).} \tag{28}$$

An explicit t dependence is allowed for translations, slowly changing axes, deformation, precession parameters, or other prescribed non-angular inputs.

Differentiating gives

$$\boldsymbol{v}_P = \frac{\partial \boldsymbol{x}_P}{\partial t} + \sum_{a=1}^{n} \frac{\partial \boldsymbol{x}_P}{\partial q^a} \dot{q}^a. \tag{29}$$

Define

$$\boldsymbol{v}_0^{(P)} \equiv \frac{\partial \boldsymbol{x}_P}{\partial t}, \qquad \boldsymbol{e}_a^{(P)} \equiv \frac{\partial \boldsymbol{x}_P}{\partial q^a}. \tag{30}$$

Then

$$\boldsymbol{v}_P = \boldsymbol{v}_0^{(P)} + \boldsymbol{e}_a^{(P)} \dot{q}^a, \tag{31}$$

where repeated phase indices are summed.

6.2 The phase metric

Definition 2 (Point-dependent phase metric)**.** *The Gram matrix of angular displacement vectors is*

$$\boxed{\mathsf{G}_{ab}^{(P)} \equiv \boldsymbol{e}_a^{(P)} \cdot \boldsymbol{e}_b^{(P)} = \frac{\partial \boldsymbol{x}_P}{\partial q^a} \cdot \frac{\partial \boldsymbol{x}_P}{\partial q^b}.} \tag{32}$$

The diagonal entries encode the squared effective radii associated with each phase. The off-diagonal entries encode the instantaneous geometric alignment between different phase-generated velocities.

Proposition 1 (Angular composition of speed)**.** *If $\partial \boldsymbol{x}_P / \partial t = 0$ in the chosen embedding, then*

$$\boxed{v_P^2 = \mathsf{G}_{ab}^{(P)} \dot{q}^a \dot{q}^b.} \tag{33}$$

For a general embedding,

$$\boxed{v_P^2 = |\boldsymbol{v}_0^{(P)}|^2 + 2\boldsymbol{v}_0^{(P)} \cdot \boldsymbol{e}_a^{(P)} \dot{q}^a + \mathsf{G}_{ab}^{(P)} \dot{q}^a \dot{q}^b.} \tag{34}$$

Proof. Take the Euclidean inner product of equation (31) with itself and use the definition (32). □

This is the precise meaning of "adding angular progressions." The correct composition is not

$$\omega_1 + \omega_2 + \cdots, \tag{35}$$

but a point-dependent quadratic form whose cross terms retain directional information.

7 Local Kinematic Chronometry

7.1 Angular-progress norm

Define the dimensionless angular-progress norm

$$\boxed{\mathcal{A}_P(t) \equiv \frac{v_P^2(t)}{c^2}.} \tag{36}$$

When the position is generated entirely by the phase coordinates,

$$\mathcal{A}_P = \frac{1}{c^2}\mathsf{G}^{(P)}_{ab}\dot{q}^a\dot{q}^b = \frac{4\pi^2}{c^2}\mathsf{G}^{(P)}_{ab}\nu_a\nu_b. \tag{37}$$

Definition 3 (Local kinematic clock factor)**.**

$$\boxed{\alpha_P(t) \equiv \frac{\mathrm{d}\tau_P}{\mathrm{d}t} = \sqrt{1-\mathcal{A}_P(t)}.} \tag{38}$$

The accumulated kinematic proper time between t_0 and t_1 is

$$\boxed{\tau_P(t_1) - \tau_P(t_0) = \int_{t_0}^{t_1}\sqrt{1-\mathcal{A}_P(t)}\,\mathrm{d}t.} \tag{39}$$

Master relation. Given the nested angular phases, their rates, and the geometric embedding of a point, the gravity-independent kinematic time of that point is

$$\boxed{\mathrm{d}\tau_P = \mathrm{d}t\sqrt{1-\frac{1}{c^2}\mathsf{G}^{(P)}_{ab}\dot{q}^a\dot{q}^b}} \tag{40}$$

when the embedding has no additional explicit translation, and is obtained from equations (21) and (34) in the general case.

7.2 Weak-speed approximation

For $v_P \ll c$,

$$\alpha_P = 1 - \frac{\mathcal{A}_P}{2} + \mathcal{O}(\mathcal{A}_P^2), \tag{41}$$

so that

$$\Delta\tau_P \simeq \Delta t - \frac{1}{2c^2}\int v_P^2(t)\,\mathrm{d}t. \tag{42}$$

The clock deficit is therefore governed by the time integral of the squared resultant velocity, not by the arithmetic total of traversed angle.

8 Dimensionless Chronometric Labels and Local Observer Translation

8.1 The local dimensionless label

The angular-progress norm $\mathcal{A}_P = v_P^2/c^2$ already supplies a dimensionless measure of motion. For translation and comparison it is useful to introduce a second, exactly invertible label that is also the period-normalised rotating-ring holonomy.

Definition 4 (Dimensionless chronometric label)**.** *For a point P with $0 \le \mathcal{A}_P < 1$, define*

$$\boxed{\mathscr{K}_P \equiv \gamma_P\mathcal{A}_P = \gamma_P\frac{v_P^2}{c^2}, \qquad \gamma_P = (1-\mathcal{A}_P)^{-1/2}.} \tag{43}$$

Using $\alpha_P = 1/\gamma_P$, the same label has the exact identities

$$\boxed{\mathscr{K}_P = \gamma_P - \frac{1}{\gamma_P} = \frac{1}{\alpha_P} - \alpha_P.} \tag{44}$$

Proposition 2 (Exact label–rate correspondence)**.** *For $\mathscr{K}_P \geq 0$, the local kinematic clock factor is uniquely recovered by*

$$\boxed{\alpha_P = \frac{\sqrt{\mathscr{K}_P^2 + 4} - \mathscr{K}_P}{2} = \frac{2}{\sqrt{\mathscr{K}_P^2 + 4} + \mathscr{K}_P}.} \tag{45}$$

Proof. Equation (44) gives $\alpha_P^2 + \mathscr{K}_P \alpha_P - 1 = 0$. Only the positive root can represent a future-directed clock-rate factor. The second expression in equation (45) is algebraically equivalent and is numerically stable for small labels. □

In the weak-speed regime,

$$\mathscr{K}_P = \mathcal{A}_P + \mathcal{O}(\mathcal{A}_P^2), \qquad \alpha_P = 1 - \frac{\mathscr{K}_P}{2} + \mathcal{O}(\mathscr{K}_P^2). \tag{46}$$

8.2 Kinematic equivalence classes

Two specified points A and B, evaluated in the same reference frame and under the same convention, are *instantaneously kinematically chronometric-equivalent* when

$$\boxed{\mathscr{K}_A(t) = \mathscr{K}_B(t).} \tag{47}$$

Then $\alpha_A(t) = \alpha_B(t)$ follows exactly from equation (45). Equality of one averaged label is not, by itself, sufficient to guarantee equality of accumulated proper time when the labels vary, because the inverse map is nonlinear. Equality of the complete functions $\mathscr{K}_A(t)$ and $\mathscr{K}_B(t)$ over a common interval does guarantee equal accumulated kinematic proper time over that interval.

The label is not an intrinsic name detached from measurement convention. A reproducible label must be accompanied by the selected reference frame, point or canonical ring, latitude, altitude, epoch, and any centre-motion contribution. These metadata define what is being compared.

8.3 Uniform rotating ring and one-winding translation

For a uniformly rotating ring,

$$\beta = \frac{|\Omega| R}{c}, \qquad \mathscr{K}_{\mathrm{ring}} = \gamma \beta^2. \tag{48}$$

The one-direction slow-transport holonomy satisfies

$$\boxed{|\mathcal{H}^*| = \mathscr{K}_{\mathrm{ring}} T_{\mathrm{rot}}, \qquad T_{\mathrm{rot}} = \frac{2\pi}{|\Omega|}.} \tag{49}$$

Thus $\mathscr{K}$ is the unit-independent dimensionless content of this particular ring observable, while multiplication by a selected period translates it into a time interval.

8.4 Canonical Earth calibration

Adopt the WGS 84 equatorial radius and the terrestrial sidereal rotation period as the canonical Earth ring. Its dimensionless label is

$$\boxed{\mathscr{K}_\oplus \equiv \gamma_\oplus \frac{\Omega_\oplus^2 R_\oplus^2}{c^2} \simeq 2.4068743 \times 10^{-12}.} \tag{50}$$

Define the Earth-cycle translation constant

$$\boxed{\Theta_\oplus \equiv \mathscr{K}_\oplus T_{\mathrm{sid},\oplus} \simeq 207.386\,\mathrm{ns}.} \tag{51}$$

The number $\Theta_\oplus$ is therefore the terrestrial time-language rendering of $\mathscr{K}_\oplus$ under the canonical equatorial sidereal-cycle convention. It is not the fundamental label itself.

8.5 Earth-normalised comparison and translation

For any point or canonical body state X, define the Earth-normalised ratio

$$\boxed{\mathscr{R}_{X/\oplus} \equiv \frac{\mathscr{K}_X}{\mathscr{K}_\oplus}.} \tag{52}$$

Its Earth-language translated interval is

$$\boxed{\Theta_X^{(\oplus)} \equiv \mathscr{K}_X T_{\mathrm{sid},\oplus} = \mathscr{R}_{X/\oplus}\Theta_\oplus.} \tag{53}$$

When expressed in terrestrial nanoseconds,

$$\boxed{\Theta_X^{(\oplus)}[\mathrm{ns}_\oplus] = \frac{\mathscr{K}_X}{\mathscr{K}_\oplus} \times 207.386\ldots} \tag{54}$$

This is a comparison coordinate in the Earth calibration language. It must not be silently identified with the physical one-winding holonomy of X. That identification is valid only when X is itself represented by a uniform ring and its own native period is used.

For a uniform ring X, define the native-cycle translation

$$\boxed{\Theta_X^{(X)} \equiv \mathscr{K}_X T_X = |\mathcal{H}_X^*|.} \tag{55}$$

The coefficient that converts the native-cycle representation to the Earth-cycle representation is

$$\boxed{C_{X\to\oplus} \equiv \frac{T_{\mathrm{sid},\oplus}}{T_X}, \qquad \Theta_X^{(\oplus)} = C_{X\to\oplus}\Theta_X^{(X)}.} \tag{56}$$

Remark 1. *The translation map changes the numerical language, not the underlying kinematic label. Equal Local Observer Translation Maps imply equal labels because the same*

positive calibration period is used. Equal native-cycle intervals do not necessarily imply equal labels, since the native periods may differ.

8.6 Scale-normalised Sagnac signatures

For a uniform ring of circumference $C = 2\pi R$, native period $T = C/v$, and $\beta = v/c$, the one-winding proper-time holonomy used in this work is

$$\boxed{|\mathcal{H}^*| = \gamma \frac{2\pi\Omega R^2}{c^2} = \gamma \frac{Cv}{c^2}.} \tag{57}$$

It produces two complementary dimensionless normalisations:

$$\boxed{\frac{|\mathcal{H}^*|}{T} = \gamma\beta^2 = \mathscr{K}, \qquad \mathscr{S} \equiv \frac{c|\mathcal{H}^*|}{C} = \gamma\beta.} \tag{58}$$

The label $\mathscr{K}$ is period normalised. The signature $\mathscr{S}$ is circumference normalised. They encode different questions and should not be interchanged.

For two counter-propagating light signals received by the co-rotating detector, the proper-time Sagnac difference is

$$\boxed{\Delta\tau_{\leftrightarrow} = 2|\mathcal{H}^*|, \qquad \frac{c\Delta\tau_{\leftrightarrow}}{C} = 2\mathscr{S} = 2\gamma\beta.} \tag{59}$$

Consequently the canonical terrestrial value 207.386 ns is the one-direction quantity $|\mathcal{H}^*_\oplus|$. The associated counter-propagating proper-time difference is approximately 414.772 ns. Stating which convention is used removes the otherwise persistent factor-of-two ambiguity.

8.7 Illustrative terrestrial–Galactic ring comparison

A galaxy is not a rigid wheel, so no single global angular velocity is assigned here. Instead, consider a declared canonical Galactic ring at radius R_G with local azimuthal speed $v_{\phi,G}$. Its ring labels are

$$\boxed{\mathscr{K}_G = \gamma_G \left(\frac{v_{\phi,G}}{c}\right)^2, \qquad \mathscr{S}_G = \gamma_G \frac{v_{\phi,G}}{c}, \qquad |\mathcal{H}^*_G| = \frac{C_G}{c}\mathscr{S}_G.} \tag{60}$$

For the explicitly illustrative values

$$R_G = 50{,}000\,\mathrm{ly}, \qquad v_{\phi,G} = 220\,\mathrm{km/s}, \tag{61}$$

one obtains approximately

$$\boxed{|\mathcal{H}^*_G| \simeq 230.5\,\mathrm{yr}, \qquad \Delta\tau_{\leftrightarrow,G} \simeq 461.1\,\mathrm{yr}.} \tag{62}$$

These large dimensional intervals arise mainly from the enormous circumference. Dividing

by circumference isolates the speed-controlled comparison:

$$\boxed{\frac{\mathscr{S}_G}{\mathscr{S}_\oplus} \simeq \frac{v_{\phi,G}}{v_{\mathrm{eq},\oplus}} \simeq 4.73 \times 10^2} \tag{63}$$

in the weak-speed regime. By contrast, the period-normalised labels obey

$$\frac{\mathscr{K}_G}{\mathscr{K}_\oplus} \simeq 2.24 \times 10^5. \tag{64}$$

The two ratios differ because $\mathscr{S} \propto \beta$ while $\mathscr{K} \propto \beta^2$ at low speed.

This comparison formalises the useful intuition that a dimensional Sagnac interval should not be compared across radically different path sizes without also reporting its native period or circumference normalisation.

9 Galactic Phase-Space Addressing and the Chronometric Atlas

9.1 Declared Galactocentric frame

A Galactic atlas requires a declared reference construction. Let

$$\boxed{\mathcal{F}_{\mathrm{G}} = \left(O_{\mathrm{G}}, \hat{\boldsymbol{e}}_x, \hat{\boldsymbol{e}}_y, \hat{\boldsymbol{e}}_z, t_{\mathrm{G}}; \boldsymbol{\Pi}_{\mathrm{G}}\right)} \tag{65}$$

contain the adopted Galactic centre, oriented axes, coordinate time, and the parameter set $\boldsymbol{\Pi}_{\mathrm{G}}$ used to transform observations into that frame. The latter may include the Solar distance from the Galactic centre, the Solar height above the adopted midplane, and the Solar velocity vector. There is no single modern standard Galactocentric frame with universally fixed numerical parameters; consequently, reproducible science code must record them explicitly rather than relying on a changing default [15].

This Galactocentric frame is distinct from the conventional *Galactic* sky-coordinate system. The latter has its three-dimensional origin at the Solar-System barycentre, whereas the former is constructed with the Galactic centre as origin. Confusing these origins changes both the position and the velocity attached to an object.

9.2 Six-dimensional Galactic chronometric address

Definition 5 (Galactic phase-space address)**.** *For an object X at epoch t_0, define*

$$\boxed{\mathfrak{A}_X(t_0 \mid \mathcal{F}_{\mathrm{G}}) = (t_0; \boldsymbol{x}_X, \boldsymbol{v}_X; \boldsymbol{\Sigma}_X, \mathcal{Q}_X)\,,} \tag{66}$$

where $\boldsymbol{x}_X$ and $\boldsymbol{v}_X$ are its Galactocentric position and velocity, $\boldsymbol{\Sigma}_X$ is the adopted covariance or uncertainty description, and $\mathcal{Q}_X$ records whether the state is measured, transformed, incomplete, or model-completed.

In Cartesian coordinates,

$$\boldsymbol{x}_X = (x_X, y_X, z_X), \qquad \boldsymbol{v}_X = (v_{x,X}, v_{y,X}, v_{z,X}). \tag{67}$$

The corresponding cylindrical quantities are

$$R_X = \sqrt{x_X^2 + y_X^2}, \qquad \phi_X = \operatorname{atan2}(y_X, x_X), \tag{68}$$

$$v_{R,X} = \frac{x_X v_{x,X} + y_X v_{y,X}}{R_X}, \qquad v_{\phi,X} = \frac{x_X v_{y,X} - y_X v_{x,X}}{R_X}, \tag{69}$$

for $R_X > 0$. Therefore

$$\boxed{v_X^2 = v_{x,X}^2 + v_{y,X}^2 + v_{z,X}^2 = v_{R,X}^2 + v_{\phi,X}^2 + v_{z,X}^2.} \tag{70}$$

Equivalently, when the path is differentiable,

$$\boxed{v_X^2 = \dot{R}_X^2 + R_X^2 \dot{\phi}_X^2 + \dot{z}_X^2.} \tag{71}$$

The Galactic azimuthal phase ϕ_X is thus one ledger component, with rate $\dot{\phi}_X = v_{\phi,X}/R_X$, but the radial and vertical components must also be retained.

9.3 Why position alone does not determine speed

A single position $\boldsymbol{x}_X(t_0)$ does not determine $\boldsymbol{v}_X(t_0)$. Distinct world lines can intersect the same event in configuration space with different tangent vectors. Hence no unique inference

$$\boldsymbol{x}_X \longmapsto \boldsymbol{v}_X \tag{72}$$

exists without additional information.

There are only two legitimate routes from position to speed in this framework:

(1) observe the position at multiple epochs or provide the full measured phase-space state;

(2) declare a conditional velocity-field closure

$$\boxed{\boldsymbol{v}_X = \boldsymbol{\mathcal{V}}_{\mathrm{MW}}(\boldsymbol{x}_X, t; \boldsymbol{\eta}),} \tag{73}$$

where $\boldsymbol{\eta}$ records all model parameters.

Examples of conditional closures include a constant tangential-speed field, a solid-rotation field, a tabulated rotation curve, or a fitted non-axisymmetric velocity map. A velocity supplied this way is not labelled "measured" in the atlas.

9.4 Nested local refinement inside a Galactic address

Let X denote the centre of a star, planet, moon, cluster, or other host object, and let P be a point attached to or moving within it. Write

$$\boldsymbol{x}_P^{\mathrm{G}} = \boldsymbol{x}_X^{\mathrm{G}} + \mathsf{R}_X(t)\boldsymbol{r}_P, \tag{74}$$

where $\mathsf{R}_X(t) \in \mathrm{SO}(3)$ embeds the local coordinates into $\mathcal{F}_{\mathrm{G}}$. Differentiation gives

$$\boldsymbol{v}_P^{\mathrm{G}} = \boldsymbol{v}_X^{\mathrm{G}} + \dot{\mathsf{R}}_X \boldsymbol{r}_P + \mathsf{R}_X \dot{\boldsymbol{r}}_P. \tag{75}$$

For a fixed point co-rotating rigidly with the host,

$$\boxed{\boldsymbol{v}_P^{\mathrm{G}} = \boldsymbol{v}_X^{\mathrm{G}} + \boldsymbol{\Omega}_X^{\mathrm{G}} \times \boldsymbol{r}_P^{\mathrm{G}}.} \tag{76}$$

Additional moon–planet, planet–star, binary, cluster, or stream layers are inserted through the same ordered phase embedding used in the earlier sections. The centre address is therefore not a replacement for the nested ledger; it is its outer Galactic layer.

9.5 Atlas label, clock factor, and Earth-language coordinate

For every complete phase-space address, define

$$\beta_X^{\mathrm{G}} = \frac{|\boldsymbol{v}_X^{\mathrm{G}}|}{c}, \qquad \gamma_X^{\mathrm{G}} = \left(1 - (\beta_X^{\mathrm{G}})^2\right)^{-1/2}, \tag{77}$$

$$\boxed{\mathscr{K}_X^{\mathrm{G}} = \gamma_X^{\mathrm{G}} (\beta_X^{\mathrm{G}})^2, \qquad \alpha_X^{\mathrm{G}} = \frac{1}{\gamma_X^{\mathrm{G}}}.} \tag{78}$$

The Earth-language atlas coordinate is

$$\boxed{\Theta_{X,\mathrm{G}}^{(\oplus)} = \mathscr{K}_X^{\mathrm{G}} T_{\mathrm{sid},\oplus} = \frac{\mathscr{K}_X^{\mathrm{G}}}{\mathscr{K}_\oplus} \Theta_\oplus.} \tag{79}$$

The full catalogue record is

$$\boxed{\mathfrak{C}_X = \left(\mathrm{ID}_X, t_0, \mathcal{F}_{\mathrm{G}}, \boldsymbol{x}_X, \boldsymbol{v}_X, \mathscr{K}_X, \alpha_X, \Theta_X^{(\oplus)}, \boldsymbol{\Sigma}_X, \mathcal{Q}_X\right).} \tag{80}$$

The translated interval is a coordinate in the declared terrestrial calibration language. It is not a claim that the object physically performs an Earth-length cycle or possesses a closed-loop holonomy equal to that interval.

9.6 Catalogue quality classes

This classification prevents a position-only map from being silently promoted into a measured time map. The catalogue may contain all classes, but only complete M6, T6, or explicitly conditional C6 rows receive $\mathscr{K}$.

9.7 Observational input and Gaia-compatible ingestion

A complete stellar input can be formed from sky position, distance, two proper-motion components, radial velocity, epoch, and their covariance. Gaia DR3 provides astrometry for a very large source set and combined radial velocities for more than 3.3×10^7 stars, making a large six-dimensional implementation feasible for the measured subset [12, 13]. The

Table 1: Minimum provenance classes for a Galactic chronometric atlas.

Code	State	Interpretation
M6	Measured six-dimensional	Position and all three velocity components are observational inputs in a declared frame.
T6	Transformed six-dimensional	A complete observed sky state has been transformed into the declared Galactocentric frame.
P5	Position plus transverse motion	Radial velocity is absent; no unique total-speed label is issued.
P3	Position only	The atlas stores the address but leaves the chronometric label undefined.
C6	Conditional six-dimensional	Missing velocity components are supplied by a named model closure and are never presented as measurements.

Gaia archive documents the fields `ra`, `dec`, `parallax`, `pmra`, `pmdec`, and `radial_velocity`, including their reference-frame and uncertainty conventions [14].

For a source with distance d in kiloparsecs and total proper motion μ in milliarcseconds per year, the transverse-speed magnitude is

$$\boxed{v_\perp[\mathrm{km\,s^{-1}}] = 4.74047\,\mu[\mathrm{mas\,yr^{-1}}]\,d[\mathrm{kpc}].} \tag{81}$$

The complete vector transformation must nevertheless preserve the directions of both proper-motion components and add the line-of-sight velocity before transforming to $\mathcal{F}_{\mathrm{G}}$. The accompanying code offers an optional Astropy path for this transformation, with the Galactocentric parameters supplied explicitly rather than inherited silently from software defaults [15].

9.8 Uncertainty propagation

Let $\boldsymbol{y}$ denote the measured astrometric and velocity variables with covariance $\boldsymbol{\Sigma}_y$. To first order,

$$\boxed{\sigma^2_{\mathscr{K}} \simeq \nabla_{\boldsymbol{y}}\mathscr{K}^{\mathsf{T}}\boldsymbol{\Sigma}_y\nabla_{\boldsymbol{y}}\mathscr{K}\,.} \tag{82}$$

For a directly measured velocity vector, with $v = |\boldsymbol{v}|$,

$$\frac{\mathrm{d}\mathscr{K}}{\mathrm{d}v} = \frac{1}{c}\left(2\beta\gamma + \beta^3\gamma^3\right), \qquad \nabla_{\boldsymbol{v}}\mathscr{K} = \frac{\mathrm{d}\mathscr{K}}{\mathrm{d}v}\frac{\boldsymbol{v}}{v}. \tag{83}$$

The Earth-language uncertainty is simply

$$\boxed{\sigma_{\Theta^{(\oplus)}} = T_{\mathrm{sid},\oplus}\sigma_{\mathscr{K}}.} \tag{84}$$

For nonlinear distance transformations, asymmetric distance posteriors, or correlated astrometric errors, Monte Carlo propagation from the full input covariance is preferable to a diagonal first-order approximation.

9.9 Demonstration of address and velocity non-uniqueness

Table 2 uses an explicit illustrative Galactocentric convention with a Solar-frame velocity vector $(12.9, 245.6, 7.78)\,\mathrm{km/s}$. These values demonstrate the pipeline; they do not define a unique Galactic standard. The second row occupies the same position as the first but carries an additional velocity component, whereas the third is placed elsewhere with the same velocity magnitude as the first.

Table 2: Illustrative Galactic atlas records. $\Theta^{(\oplus)}$ is the Earth-language comparison coordinate.

Record	v (km/s)	$\mathscr{K}$	$\mathscr{K}/\mathscr{K}_\oplus$	$\Theta^{(\oplus)}$ (ms)
Illustrative Solar-frame state	246.062	6.73669×10^{-7}	2.79894×10^{5}	58.046
Same position, extra component	270.419	8.13640×10^{-7}	3.38049×10^{5}	70.107
Different position, equal speed	246.062	6.73669×10^{-7}	2.79894×10^{5}	58.046
Position only	–	–	–	–

The first and third rows belong to the same instantaneous kinematic chronometric equivalence class despite different positions. The first and second do not, despite identical positions. This is the operational reason that the atlas is fundamentally a phase-space atlas rather than a position-only map.

9.10 Atlas construction algorithm

A catalogue implementation follows this sequence:

A1: Freeze the Galactocentric frame parameters, axis convention, and epoch.

A2: Ingest either Cartesian phase-space rows or complete sky-coordinate rows.

A3: Transform every complete state to the same $\mathcal{F}_{\mathrm{G}}$; retain the full covariance when available.

A4: Assign the provenance class M6, T6, P5, P3, or C6.

A5: Compute cylindrical coordinates and verify $v^2 = v_R^2 + v_\phi^2 + v_z^2$.

A6: Calculate $\mathscr{K}$, α, $\mathscr{K}/\mathscr{K}_\oplus$, and $\Theta^{(\oplus)}$ only for complete states.

A7: Propagate uncertainties and attach quality flags rather than dropping them from the map.

A8: Store the result as a versioned table and visualise position separately from colour- or contour-coded chronometric labels.

A9: Never mix labels calculated in different frames, epochs, or Solar-parameter conventions without an explicit transformation.

10 A Planetary Specialisation

10.1 Centre motion plus body rotation

Let C be the centre of a rotating planet. Let $\boldsymbol{V}_C(t)$ be its velocity in the chosen quasi-inertial frame, and let $\boldsymbol{r}_P(t)$ be the position of point P relative to the centre. For rigid rotation with angular velocity $\boldsymbol{\Omega}(t)$,

$$\frac{\mathrm{d}\boldsymbol{r}_P}{\mathrm{d}t} = \boldsymbol{\Omega} \times \boldsymbol{r}_P. \tag{85}$$

Therefore

$$\boxed{\boldsymbol{v}_P = \boldsymbol{V}_C + \boldsymbol{\Omega} \times \boldsymbol{r}_P.} \tag{86}$$

The exact local kinematic clock factor is

$$\boxed{\alpha_P = \sqrt{1 - \frac{|\boldsymbol{V}_C + \boldsymbol{\Omega} \times \boldsymbol{r}_P|^2}{c^2}}.} \tag{87}$$

Expanding,

$$\boxed{|\boldsymbol{v}_P|^2 = V_C^2 + |\boldsymbol{\Omega} \times \boldsymbol{r}_P|^2 + 2\boldsymbol{V}_C \cdot (\boldsymbol{\Omega} \times \boldsymbol{r}_P).} \tag{88}$$

The three terms have distinct meanings:

(1) motion of the planetary centre through the selected frame;

(2) local spin motion around the planetary axis;

(3) instantaneous orbital–spin alignment or opposition.

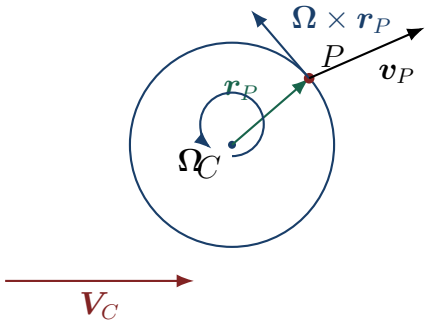

Figure 3: Velocity composition for a point on a rotating planet. The total velocity is the vector sum of centre motion and local spin motion.

10.2 Planar circular approximation

For a circular planetary orbit of radius a, orbital angular rate ω_{orb}, and local cylindrical spin radius ρ, let

$$V_{\mathrm{orb}} = a|\omega_{\mathrm{orb}}|, \qquad v_{\mathrm{spin}} = \rho|\Omega_{\mathrm{spin}}|. \tag{89}$$

If $\psi(t)$ is the instantaneous angle between the orbital and spin velocity vectors,

$$\boxed{v_P^2 = V_{\mathrm{orb}}^2 + v_{\mathrm{spin}}^2 + 2V_{\mathrm{orb}}v_{\mathrm{spin}}\cos\psi(t).} \tag{90}$$

Thus

$$\boxed{\alpha_P(t) = \sqrt{1 - \frac{V_{\text{orb}}^2 + v_{\text{spin}}^2 + 2V_{\text{orb}}v_{\text{spin}}\cos\psi(t)}{c^2}}.} \tag{91}$$

The cross term is required for instantaneous chronometry. It cannot be recovered from the magnitudes of the two angular rates alone; their geometric orientation must also be known.

11 Any Point on or Above a Rotating Planet

11.1 Latitude and altitude

For a spherical reference body of radius R, a point at altitude h and latitude λ has cylindrical distance from the spin axis

$$\boxed{\rho(h,\lambda) = (R+h)\cos\lambda.} \tag{92}$$

The local spin speed is

$$\boxed{v_{\text{spin}}(h,\lambda) = |\Omega_{\text{spin}}|(R+h)\cos\lambda.} \tag{93}$$

Substitution into equation (91) gives a direct kinematic time field over the body and its co-rotating atmosphere:

$$\alpha(h,\lambda,t) = \Bigg[1 - \frac{1}{c^2}\Big(V_{\text{orb}}^2 + \Omega_{\text{spin}}^2(R+h)^2\cos^2\lambda + 2V_{\text{orb}}|\Omega_{\text{spin}}|(R+h)\cos\lambda\cos\psi(t)\Big)\Bigg]^{1/2}. \tag{94}$$

This expression contains no gravitational variable. Its inputs are the selected reference-frame motion of the centre, the spin rate, the point's position relative to the axis, and the instantaneous orientation between the nested motions.

11.2 Full-rotation average

If $\boldsymbol{V}_C$ changes negligibly over one body rotation and the spin velocity explores a complete symmetric cycle, then

$$\langle \boldsymbol{V}_C \cdot (\boldsymbol{\Omega} \times \boldsymbol{r}_P)\rangle_{\text{spin}} = 0. \tag{95}$$

Hence

$$\boxed{\left\langle v_P^2 \right\rangle_{\text{spin}} = V_C^2 + \Omega_{\text{spin}}^2\rho^2.} \tag{96}$$

In the weak-speed limit,

$$\boxed{\langle \alpha_P \rangle_{\text{spin}} \simeq 1 - \frac{V_C^2}{2c^2} - \frac{\Omega_{\text{spin}}^2\rho^2}{2c^2}.} \tag{97}$$

The first deficit is common to points sharing the same centre motion. The second distinguishes latitude and altitude.

11.3 Difference between two altitudes

At the same latitude, compare h_1 and h_2. The spin-only difference in squared speed is

$$\Delta v_{\text{spin}}^2 = \Omega_{\text{spin}}^2 \left[(R+h_2)^2 - (R+h_1)^2\right] \cos^2\lambda. \tag{98}$$

For $|h_2 - h_1| \ll R$,

$$\Delta v_{\text{spin}}^2 \simeq 2\Omega_{\text{spin}}^2 R \Delta h \cos^2\lambda. \tag{99}$$

The full-spin-cycle averaged kinematic clock-rate difference is therefore

$$\boxed{\Delta \left\langle \frac{\mathrm{d}\tau}{\mathrm{d}t} \right\rangle_{\text{kin}} \simeq -\frac{\Omega_{\text{spin}}^2 R \Delta h \cos^2\lambda}{c^2}.} \tag{100}$$

Within this gravity-independent sector, the higher co-rotating point is farther from the axis, has greater spin speed, and accumulates slightly less proper time over the same coordinate interval.

12 Multiple Nested Motions

12.1 General two-phase form

For two angular phases q^1 and q^2,

$$\boxed{v_P^2 = \mathsf{G}_{11}(\dot{q}^1)^2 + \mathsf{G}_{22}(\dot{q}^2)^2 + 2\mathsf{G}_{12}\dot{q}^1\dot{q}^2.} \tag{101}$$

Writing

$$\mathsf{G}_{11} = r_1^2, \qquad \mathsf{G}_{22} = r_2^2, \qquad \mathsf{G}_{12} = r_1 r_2 \cos\psi, \tag{102}$$

recovers

$$\boxed{v_P^2 = r_1^2\omega_1^2 + r_2^2\omega_2^2 + 2r_1 r_2 \omega_1\omega_2 \cos\psi.} \tag{103}$$

The sign of each ω_a and the geometry contained in $\cos\psi$ jointly determine whether two phase-generated velocities reinforce or oppose each other.

12.2 Moon, planet, star, and galactic layers

For a lunar surface point, the position may contain at least the following motions:

(1) lunar spin about the lunar axis;

(2) lunar orbital motion about the Earth–Moon barycentre;

(3) barycentric motion of the Earth–Moon system about the Solar-System barycentre;

(4) optional higher motion relative to a Galactic or extragalactic reference construction.

The ledger stores these phase rates separately. The embedding $\boldsymbol{x}_P(\boldsymbol{q})$ determines their exact vector composition. Adding more layers changes the dimension of $\mathsf{G}_{ab}^{(P)}$ but not the chronometric law (40).

12.3 Non-commuting axes

Finite orientations in three dimensions satisfy

$$R_{P/\mathrm{ref}}(t) = R_n(q^n)R_{n-1}(q^{n-1})\cdots R_1(q^1), \qquad R_a \in \mathrm{SO}(3). \tag{104}$$

The order is physical. The position embedding is built from the ordered product, after which the phase metric is computed by differentiation. The quadratic chronometric relation remains valid even though the finite angles themselves do not commute.

13 Conditional Rod–Flywheel Closure

13.1 Role of the closure

The mechanical prelude and the chronometric construction answer different questions. The closure layer asks

> What determines the angular rates and how are stresses, torque, or phase locks transmitted?

The chronometric layer asks

> Given the rates and geometry, what velocity, clock factor, Sagnac signature, and catalogue label follow?

Keeping these questions separate prevents a kinematic identity from being mistaken for a dynamical explanation.

13.2 Radial-capacity relation

Suppose an idealised rod–flywheel channel assigns a radial transport capacity $F_\star$ to a terminal body of inertial mass m at lever length a:

$$\boxed{F_\star = m\omega_{\mathrm{orb}}^2 a.} \tag{105}$$

Then

$$\boxed{\omega_{\mathrm{orb}} = \sqrt{\frac{F_\star}{ma}}.} \tag{106}$$

For uniform density ϱ and body radius R,

$$m = \frac{4\pi}{3}\varrho R^3, \tag{107}$$

so

$$\boxed{\omega_{\mathrm{orb}} = \sqrt{\frac{3F_\star}{4\pi\varrho R^3 a}}.} \tag{108}$$

At fixed $F_\star$, ϱ, and ω_{orb},

$$\boxed{R \propto a^{-1/3}.} \tag{109}$$

Equation (14) is the corresponding Earth–Neptune example.

13.3 Spin-transfer law

Orbital angular progression does not by itself determine the terminal body's spin. Introduce an independent dimensionless transfer function

$$\boxed{\Omega_{\mathrm{spin}} = \omega_{\mathrm{orb}}\,\Phi\!\left(\frac{a}{R};\boldsymbol{\eta}\right),} \tag{110}$$

where $\boldsymbol{\eta}$ may contain moment-of-inertia coefficients, coupling fractions, density ratios, compliance parameters, or phase-lock indices. The function Φ must be derived from a mechanical model or inferred from data; it is not supplied by the phase ledger.

13.4 Finite propagation and the limit of the rigid-rod picture

An exactly rigid rod is an idealisation. In any physical completion, stress and torque propagate at finite speed and the outer terminal cannot be updated instantaneously. The useful content of the thought experiment is therefore not literal rigidity but a declared rule that correlates angular phases across scales. A finite-stiffness completion would replace the algebraic constraint by dynamical fields or coupled phase variables and would test equilibrium, stability, and spin-up.

13.5 Insertion into the chronometric map

Once a closure or observation supplies ω_{orb} and Ω_{spin}, the ledger stores them separately. The embedding then constructs the resultant velocity with all cross terms, after which $\mathscr{K}$, $\mathscr{S}$, and the local clock factor follow. No rod force, mass, or body size is inserted directly into the special-relativistic clock law; those quantities affect chronometry only through the motion generated by the declared closure.

14 Earth as a Numerical Demonstration

14.1 Reference values

For illustration, adopt the WGS 84 equatorial radius

$$R_{\oplus} = 6\,378\,137\,\mathrm{m}, \tag{111}$$

and the mean sidereal day

$$T_{\mathrm{sid}} = 86\,164.090\,54\,\mathrm{s}, \tag{112}$$

so that [9, 6]

$$\Omega_{\oplus} = \frac{2\pi}{T_{\mathrm{sid}}} \simeq 7.292\,115 \times 10^{-5}\,\mathrm{rad/s}. \tag{113}$$

The equatorial spin speed is

$$\boxed{v_{\mathrm{eq}} = \Omega_{\oplus} R_{\oplus} \simeq 465.101\,\mathrm{m/s}.} \tag{114}$$

The spin-only dimensionless progress norm at the equator is

$$\mathcal{A}_{\mathrm{spin,eq}} = \frac{v_{\mathrm{eq}}^2}{c^2} \simeq 2.4068743 \times 10^{-12}. \tag{115}$$

The exact canonical Earth label is

$$\boxed{\mathscr{K}_\oplus = \gamma_\oplus \mathcal{A}_{\mathrm{spin,eq}} \simeq 2.4068743 \times 10^{-12}.} \tag{116}$$

The difference between $\mathscr{K}_\oplus$ and $\mathcal{A}_{\mathrm{spin,eq}}$ begins only beyond the displayed precision. Multiplication by the sidereal period gives the Earth-language calibration

$$\boxed{\Theta_\oplus = \mathscr{K}_\oplus T_{\mathrm{sid}} \simeq 207.386\,\mathrm{ns}.} \tag{117}$$

The corresponding proper-time deficit of the co-rotating equatorial clock over one sidereal rotation is a different quantity,

$$\delta\tau_{\mathrm{spin,eq}} = T_{\mathrm{sid}} \left(1 - \frac{1}{\gamma_\oplus}\right) \simeq 103.7\,\mathrm{ns}. \tag{118}$$

At weak speed, this deficit is approximately one half of the one-direction holonomy translation $\Theta_\oplus$.

14.2 Orbital centre motion

Using the exact astronomical unit and the sidereal year listed by JPL [6], a circular mean-speed estimate is

$$V_{\mathrm{orb},\oplus} \simeq \frac{2\pi\,\mathrm{au}}{T_{\mathrm{year}}} \simeq 29.785\,\mathrm{km/s}. \tag{119}$$

This common centre-motion contribution produces a weak-speed kinematic deficit of approximately

$$\frac{V_{\mathrm{orb},\oplus}^2}{2c^2} \simeq 4.94 \times 10^{-9}, \tag{120}$$

or approximately 425 µs over one sidereal day in the selected barycentric flat-spacetime approximation. It is common to nearby terrestrial points and therefore largely cancels in comparisons between them.

The instantaneous cross term in equation (88) can be much larger than the spin-squared term, reaching an equatorial magnitude of order

$$\frac{V_{\mathrm{orb},\oplus} v_{\mathrm{eq}}}{c^2} \simeq 1.54 \times 10^{-10}, \tag{121}$$

but it changes sign through the daily cycle and averages to zero under the symmetric approximation (95).

14.3 Sea level and a higher co-rotating point

At the equator, compare sea level with a point 1000 m higher. Equation (100) gives

$$\Delta \left\langle \frac{\mathrm{d}\tau}{\mathrm{d}t} \right\rangle_{\mathrm{kin}} \simeq -3.77 \times 10^{-16}. \tag{122}$$

Over one sidereal day,

$$\boxed{\Delta\tau_{\mathrm{kin}} \simeq -32.5\,\mathrm{ps}.} \tag{123}$$

The sign is negative because the higher co-rotating point has the greater tangential spin speed. This number is intentionally only the rotational kinematic contribution. It is not the complete observed clock difference between the two altitudes.

14.4 Latitude dependence

At latitude λ, the spin-only contribution scales as

$$\mathcal{A}_{\mathrm{spin}}(\lambda, h) = \frac{\Omega_\oplus^2 (R_\oplus + h)^2 \cos^2 \lambda}{c^2}. \tag{124}$$

It is maximal at the equator and vanishes on the rotation axis. The orbital centre-motion term does not vanish at the poles; only the local spin component does.

14.5 Illustrative rescaled-Earth translation

Consider the conditional scaling discussed in the rod–flywheel model,

$$a' = 27 a_\oplus, \qquad R' = \frac{R_\oplus}{3}, \qquad \Omega' = 9\Omega_\oplus. \tag{125}$$

At leading weak-speed order, the native one-winding interval remains equal to the terrestrial value because

$$\Omega' R'^2 = 9\Omega_\oplus \frac{R_\oplus^2}{9} = \Omega_\oplus R_\oplus^2. \tag{126}$$

However, the equatorial speed becomes $v' = 3v_\oplus$, so the dimensionless label becomes, to leading order,

$$\mathscr{K}' \simeq 9\mathscr{K}_\oplus. \tag{127}$$

Its Local Observer Translation Map is therefore

$$\Theta^{(\oplus)}_{\mathrm{scaled}} \simeq 9\Theta_\oplus \simeq 1866.48\,\mathrm{ns}_\oplus, \tag{128}$$

while its native-cycle translation remains approximately 207.386 ns because its native period is $T' = T_\oplus/9$. This example demonstrates why a native interval and an Earth-language comparison value must be kept distinct.

15 Natural Celestial Examples in Ledger Form

15.1 Mars

For a point on Mars, the same equations apply after replacing the terrestrial inputs by the Martian centre trajectory, rotation axis, angular rate, reference radius, latitude, altitude, and local phase. The mean Martian solar day follows from the signed phase difference

$$\omega_{\mathrm{sol,Mars}} = \omega_{\mathrm{sid,Mars}} - \omega_{\mathrm{orb,Mars}}. \tag{129}$$

The local kinematic time field is not tied to the solar day definition; it is computed from the resultant velocity in the chosen reference frame.

15.2 Venus

Venus has a retrograde signed sidereal spin period. In the ledger, $\Omega_{\mathrm{spin}} < 0$. The local spin velocity $\boldsymbol{\Omega} \times \boldsymbol{r}_P$ therefore reverses direction, changing the instantaneous cross term with orbital motion. The proper-time factor remains positive because it depends on the squared magnitude of the resultant velocity.

15.3 The Moon

For a lunar surface point, synchronous rotation is expressed by near equality between the lunar spin rate and the Moon's orbital phase rate about Earth. These two rates are distinct ledger entries even when their difference is nearly zero. The local velocity still contains the Moon-centre motion plus the surface spin term; phase locking does not erase either physical motion.

15.4 Differentially rotating bodies

For gas giants and stars, Ω_{spin} may depend on latitude, depth, or pattern. Then

$$\Omega_{\mathrm{spin}} = \Omega_{\mathrm{spin}}(\lambda, h, t), \tag{130}$$

and equation (94) remains valid pointwise after the appropriate local angular rate is supplied. A unique body-level spin number requires an explicit convention, whereas the local chronometric field does not.

16 Operational Calculation Protocol

A reproducible calculation for one point, one object, or an atlas catalogue proceeds as follows:

Step 1: Choose the quasi-inertial or Galactocentric reference frame, freeze all frame parameters, and state the coordinate time and epoch.

Step 2: Construct the immediate-parent phase chain and define signed unwrapped phases.

Step 3: For an atlas row, store the position, velocity, covariance, and provenance class separately; do not infer a unique velocity from position alone.

Step 4: Specify the position embedding $\boldsymbol{x}_P(\boldsymbol{q}; t)$, including all relevant radii, axes, inclinations, translations, and ordered rotations.

Step 5: Differentiate the embedding to obtain $\boldsymbol{v}_P$, or compute the phase metric $\mathsf{G}^{(P)}_{ab}$.

Step 6: Evaluate the angular-progress norm $\mathcal{A}_P = v_P^2/c^2$.

Step 7: Compute the dimensionless chronometric label $\mathscr{K}_P = \gamma_P \mathcal{A}_P$ and recover the local clock factor from equation (45).

Step 8: If an Earth-language comparison is required, compute $\mathscr{R}_{P/\oplus}$ and $\Theta_P^{(\oplus)} = \mathscr{K}_P T_{\mathrm{sid},\oplus}$.

Step 9: Integrate α_P over the desired coordinate-time interval when accumulated proper time, rather than an instantaneous label, is required.

Step 10: For comparisons of altitude, latitude, or nested motions, retain instantaneous vector cross terms unless a complete symmetric-cycle average is explicitly intended.

Step 11: If velocity or angular rates are not measured inputs, name and parameterise the separate closure law and mark the output as conditional.

Step 12: Propagate uncertainty from the input covariance or by Monte Carlo sampling.

Step 13: Record the frame, epoch, point convention, latitude, altitude, closure status, and whether the label is spin-only, centre-motion-only, or fully nested.

Step 14: Do not interpret the result as a complete gravitational clock prediction unless a metric-specific gravitational sector is added separately.

17 What Is Established and What Remains Conditional

17.1 Established within the stated kinematic framework

The following statements follow from the geometry and standard relativistic kinematics used here:

(1) Nested celestial cycles can be represented by a signed, unwrapped phase ledger.

(2) A point's resultant velocity is obtained from the geometric embedding of those phases.

(3) The correct combination of angular rates is a point-dependent quadratic form with cross terms.

(4) The local flat-spacetime kinematic proper time follows exactly from the resultant speed.

(5) Latitude, altitude, longitude, epoch, spin direction, and parent-system motion enter through explicit geometric variables.

(6) A complete spin-cycle average removes the centre–spin cross term under the stated symmetry assumptions.

(7) No gravitational potential is required to calculate this isolated kinematic sector.

(8) The dimensionless label $\mathscr{K}_P$ uniquely reconstructs the local kinematic clock factor in the selected frame.

(9) For a uniform rotating ring, $\mathscr{K} = |\mathcal{H}^*|/T_{\mathrm{rot}}$ exactly.

(10) The Earth-language map is an algebraically reproducible calibration once the canonical terrestrial ring is fixed.

(11) A complete Galactocentric six-dimensional state uniquely determines the isolated kinematic label in the declared frame.

(12) Cartesian and cylindrical representations give the same speed and therefore the same label when transformed consistently.

(13) A versioned catalogue can retain measured, incomplete, and conditionally completed rows without conflating their provenance.

17.2 Not established by the chronometric construction

The present framework does not establish that

(1) a scalar arithmetic sum of angles is identical to physical time;

(2) planetary spin and orbital rates are determined by distance and diameter alone;

(3) the rod–flywheel capacity law is an observed universal law of planetary systems;

(4) one universal function $\Phi(a/R)$ explains every planetary day;

(5) gravitational clock effects are absent from real measurements;

(6) strong-field compact objects can be treated by the flat-spacetime formula without metric-specific corrections;

(7) the selected quasi-inertial frame is an absolute state of rest;

(8) 207.386 ns is a universal time quantum or an observer-independent number;

(9) the Local Observer Translation Map of a general point is automatically its physical one-winding holonomy;

(10) equality of native-cycle intervals alone guarantees equality of dimensionless labels or clock rates;

(11) one instantaneous Galactic position uniquely determines an object's velocity;

(12) a conditional Galactic rotation field is observationally measured merely because it supplies a velocity for a position-only row;

(13) a Galactocentric kinematic label is frame-independent or a complete replacement for Galactic gravitational dynamics.

18 Discussion

The main gain of the phase-metric formulation is conceptual economy. Spin, orbit, parent motion, and higher cycles are not forced into one arbitrary scalar. They remain separate in the ledger, while their physical effect on a selected point is combined by the geometry of the point's actual path.

This resolves three recurrent ambiguities.

First, equal angular velocities do not imply equal linear velocities. The local contribution of a phase is weighted by the appropriate effective radius contained in G_{aa}.

Second, the relative direction of two motions matters. The off-diagonal entries G_{ab} retain reinforcement and opposition through the cross terms.

Third, a body-level period does not by itself define the time rate at every location. The local radius from the rotation axis, latitude, altitude, and phase orientation must be supplied.

The framework is therefore naturally local while remaining nested. It can describe a surface clock, a mountain clock, a co-rotating atmospheric clock, a lunar marker, or a point in a multi-body chain with the same master equation.

The optional rod–flywheel closure addresses a different problem: how angular rates might be related to mass, size, and lever length under a specified transport capacity. Keeping this closure separate prevents a tentative mechanical hypothesis from being confused with the exact kinematic map from motion to proper time.

The dimensionless label introduces a second kind of economy. Instead of attaching a dimensional interval directly to every body, the framework first records $\mathscr{K}_X$. The same label can then be displayed in its native-cycle language, in the canonical Earth-cycle language, or in another declared calibration. The translated number changes with the chosen calibration period; the underlying label does not. In this precise sense, 207.386 ns is a terrestrial translation of the canonical Earth label rather than the label itself.

This translation architecture also defines equivalence classes. Bodies or points with the same $\mathscr{K}$, evaluated under the same frame and point convention, have the same kinematic clock factor even when their radii, angular rates, and native periods differ. Conversely, equal one-cycle dimensional intervals can conceal different clock factors if the native cycle durations are different.

The Galactic extension changes the scale but not the logic. A catalogue row is first a phase-space address, not a scalar time tag. Only after a velocity has been measured or conditionally supplied does the row acquire $\mathscr{K}$. This prevents a colour map of position from being mistaken for a measured map of time rates. The resulting product is best understood as a *kinematic chronometric atlas*: a versioned map of frame-dependent phase-space labels, Local Observer Translation Maps, and uncertainties.

The atlas also provides a practical test of the framework. Objects at different positions but equal resultant speeds must occupy the same instantaneous $\mathscr{K}$ class in the selected frame. Objects at the same position but different velocities must not. Large catalogues such as Gaia DR3 allow these classifications to be generated for measured six-dimensional subsets, while incomplete rows remain visibly incomplete or are processed under explicitly named model closures.

19 Conclusion

The combined construction begins with a simple methodological fact:

$$\boxed{\text{trajectory} \neq \text{mechanism.}} \tag{131}$$

The rod and spinning-top thought experiment demonstrates this by producing circular, phase-coherent motion through a mechanical constraint whose force, energy, and angular-momentum scalings differ from those of a gravitational orbit. It also supplies a transparent conditional closure in which mass, lever length, density, and a transfer law determine angular rates.

The chronometric architecture begins only after those rates are measured or supplied. A signed nested phase ledger and its geometric embedding produce

$$\mathsf{G}^{(P)}_{ab} = \frac{\partial \boldsymbol{x}_P}{\partial q^a} \cdot \frac{\partial \boldsymbol{x}_P}{\partial q^b}, \qquad v_P^2 = \mathsf{G}^{(P)}_{ab} \dot{q}^a \dot{q}^b, \tag{132}$$

and therefore

$$\boxed{\mathrm{d}\tau_P = \mathrm{d}t \sqrt{1 - \frac{v_P^2}{c^2}}.} \tag{133}$$

For a point on or above a rotating body,

$$\boxed{\boldsymbol{v}_P = \boldsymbol{V}_C + \boldsymbol{\Omega} \times \boldsymbol{r}_P.} \tag{134}$$

This expression preserves the instantaneous vector cross term rather than adding orbital and spin angles as scalars.

For a uniform rotating ring, the architecture supplies two complementary scale-independent observables:

$$\boxed{\mathscr{K} = \frac{|\mathcal{H}^*|}{T} = \gamma\beta^2, \qquad \mathscr{S} = \frac{c|\mathcal{H}^*|}{C} = \gamma\beta.} \tag{135}$$

The canonical terrestrial one-winding translation is 207.386 ns, while the complete counter-propagating Sagnac proper-time difference is twice that value. At Galactic scale, a canonical ring must be declared by its radius, azimuthal speed, frame, and epoch; the Milky Way is not treated as a rigid wheel. Dimensional holonomies, period-normalised labels, and circumference-normalised signatures are stored separately so that scale and speed are not conflated.

The final atlas record is

$$\boxed{\mathfrak{A}_X(t_0 \mid \mathcal{F}_{\mathrm{G}}) = (t_0; \boldsymbol{x}_X, \boldsymbol{v}_X; \boldsymbol{\Sigma}_X, \mathcal{Q}_X)\,.} \tag{136}$$

Position alone does not determine a speed or chronometric label. A velocity must be measured, transformed, or supplied under an explicitly named closure. The complete architecture is therefore modular:

$$\boxed{\begin{array}{c}\text{dynamics or observation} \to \text{phase ledger} \to \text{geometry} \\ \to \text{chronometry} \to \text{catalogue provenance.}\end{array}} \tag{137}$$

The rod gives a mechanism for asking how angular coherence might be imposed. The phase metric tells how nested angular coherence becomes motion. The chronometric labels tell how that motion is read as a local kinematic clock state. The Galactic atlas tells how those states can be compared without hiding the reference frame, scale, uncertainty, or origin of the velocity.

A Derivation in Turn-Count Coordinates

Since $q^a = 2\pi N_a$,

$$\frac{\partial \boldsymbol{x}_P}{\partial N_a} = 2\pi \frac{\partial \boldsymbol{x}_P}{\partial q^a}. \tag{138}$$

Define the turn-count metric

$$\widetilde{\mathsf{G}}^{(P)}_{ab} \equiv \frac{\partial \boldsymbol{x}_P}{\partial N_a} \cdot \frac{\partial \boldsymbol{x}_P}{\partial N_b} = 4\pi^2 \mathsf{G}^{(P)}_{ab}. \tag{139}$$

Then

$$\boxed{v_P^2 = \widetilde{\mathsf{G}}^{(P)}_{ab} \nu_a \nu_b, \qquad \nu_a = \dot{N}_a.} \tag{140}$$

The proper-time law may be written directly in ledger rates:

$$\boxed{\mathrm{d}\tau_P = \mathrm{d}t \sqrt{1 - \frac{\widetilde{\mathsf{G}}^{(P)}_{ab} \nu_a \nu_b}{c^2}}.} \tag{141}$$

B Exact Two-Phase Example

Let

$$\boldsymbol{x}(q^1, q^2) = r_1 \begin{pmatrix} \cos q^1 \\ \sin q^1 \end{pmatrix} + r_2 \begin{pmatrix} \cos q^2 \\ \sin q^2 \end{pmatrix}. \tag{142}$$

Then

$$\frac{\partial \boldsymbol{x}}{\partial q^1} = r_1 \begin{pmatrix} -\sin q^1 \\ \cos q^1 \end{pmatrix}, \tag{143}$$

$$\frac{\partial \boldsymbol{x}}{\partial q^2} = r_2 \begin{pmatrix} -\sin q^2 \\ \cos q^2 \end{pmatrix}. \tag{144}$$

Therefore

$$\mathsf{G} = \begin{pmatrix} r_1^2 & r_1 r_2 \cos(q^1 - q^2) \\ r_1 r_2 \cos(q^1 - q^2) & r_2^2 \end{pmatrix}, \tag{145}$$

and

$$v^2 = r_1^2\omega_1^2 + r_2^2\omega_2^2 + 2r_1 r_2 \omega_1 \omega_2 \cos(q^1 - q^2). \tag{146}$$

This explicit example shows that the cross term is generated by the geometry of the phase embedding and not added by hand.

C Relation to a Rotating-Ring Holonomy

For one uniformly rotating ring of radius R and angular velocity Ω, the local speed is $v = |\Omega|R$ and

$$\frac{\mathrm{d}\tau}{\mathrm{d}t} = \sqrt{1 - \frac{\Omega^2 R^2}{c^2}}. \tag{147}$$

A slow directed transport around the same ring yields the standard one-winding time holonomy

$$\mathcal{H}^* = -\gamma \frac{2\pi\Omega R^2}{c^2}, \qquad \gamma = \left(1 - \frac{\Omega^2 R^2}{c^2}\right)^{-1/2}. \tag{148}$$

Since $T_{\mathrm{rot}} = 2\pi/|\Omega|$, the magnitude obeys

$$\boxed{\frac{|\mathcal{H}^*|}{T_{\mathrm{rot}}} = \gamma \frac{\Omega^2 R^2}{c^2} = \mathscr{K}_{\mathrm{ring}}.} \tag{149}$$

Thus the dimensionless chronometric label is exactly the period-normalised one-direction holonomy for a uniform ring. The phase-metric framework is more general because it applies to open intervals, arbitrary points, multiple nested phases, and non-closed trajectories. For those more general cases, $\mathscr{K}_P$ remains a local rate label, whereas multiplying it by a calibration period is a declared translation rather than an automatically realised loop observable.

D Stable Numerical Formulae

For a known speed v with $\beta = v/c \ll 1$,

$$1 - \sqrt{1 - \beta^2} = \frac{\beta^2}{1 + \sqrt{1 - \beta^2}}, \tag{150}$$

which avoids subtractive loss of precision. For a time series sampled at t_i, the accumulated proper time may be approximated by

$$\tau_P \simeq \sum_i \Delta t_i \sqrt{1 - \frac{v_P^2(t_i)}{c^2}}, \tag{151}$$

or by a higher-order quadrature when ephemeris accuracy requires it.

For a label $\mathscr{K}$, the stable inverse clock factor is

$$\alpha = \frac{2}{\sqrt{\mathscr{K}^2 + 4} + \mathscr{K}}. \tag{152}$$

The Local Observer Translation Map in nanoseconds is

$$\Theta_X^{(\oplus)}[\mathrm{ns}_\oplus] = 10^9\, T_{\mathrm{sid},\oplus} \mathscr{K}_X. \tag{153}$$

E Reference Python Implementations

The accompanying source package includes three executable files. The first, `chronometric_label_calculator.py`, computes canonical spin-only and local nested labels. The second, `rod_phase_sagnac_calculator.py`, implements the engineered bridge from a declared rod-capacity closure or a uniform-ring state to $\mathscr{K}$, $\mathscr{S}$, the one-winding holonomy, and the full counter-propagating Sagnac proper-time difference. The third, `galactic_chronometric_atlas.py`, ingests single phase-space states or catalogue CSV files, converts Cartesian addresses to cylindrical components, distinguishes measured from conditionally completed rows, computes $\mathscr{K}$, and writes local-observer atlas tables. Its optional ICRS mode uses Astropy with explicit Galactocentric parameters. The package also includes Cartesian and Gaia-style CSV templates plus example measured and conditional outputs. None of the implementations includes a gravitational clock sector.

Listing 1: Local dimensionless chronometric-label and Local Observer Translation Map calculator.

```
#!/usr/bin/env python3
"""
Dimensionless chronometric-label and Earth-language translation calculator.

The calculator implements the gravity-independent kinematic sector developed in
"Nested Angular Progress and Local Kinematic Time".

Core definitions
----------------
    beta = v / c
    gamma = 1 / sqrt(1 - beta**2)
    K = gamma * beta**2 = gamma - 1/gamma
    alpha = d_tau/dt = 1/gamma

For Earth's canonical equatorial rotating ring,
    K_E = gamma_E * (Omega_E R_E / c)**2
and the Earth-language calibration interval is
    Theta_E = K_E * T_E ~= 207.386 ns.

For another object or point X, its Earth-language translation is
    Theta_X^(E) = K_X * T_E
                = (K_X / K_E) * Theta_E.

This translated interval is a comparison convention. It is not automatically
the physical one-winding holonomy of X. For a uniformly rotating canonical
ring, the native one-winding holonomy magnitude is K_X * T_X.

No gravitational potential or gravitational redshift is included.
"""

from __future__ import annotations

```

```python
import argparse
import math
from dataclasses import dataclass, asdict
from typing import Any

C = 299_792_458.0  # m/s, exact SI value

# Canonical Earth calibration adopted in the article.
EARTH_RADIUS_M = 6_378_137.0
EARTH_SIDEREAL_PERIOD_S = 86_164.09054


@dataclass(frozen=True)
class ChronometricRecord:
    """Computed dimensionless label and translation quantities."""

    name: str
    speed_m_s: float
    beta: float
    gamma: float
    clock_factor_alpha: float
    label_K: float
    earth_normalized_ratio: float
    earth_language_seconds: float
    earth_language_nanoseconds: float
    native_period_s: float | None
    native_cycle_equivalent_seconds: float | None
    native_cycle_equivalent_nanoseconds: float | None
    native_to_earth_conversion: float | None


def angular_velocity(rotation_period_s: float) -> float:
    """Return signed angular velocity in rad/s."""
    if rotation_period_s == 0.0:
        raise ValueError("rotation_period_s cannot be zero")
    return 2.0 * math.pi / rotation_period_s


def cylindrical_radius(
    body_radius_m: float,
    altitude_m: float = 0.0,
    latitude_deg: float = 0.0,
) -> float:
    """Distance from the rotation axis: rho = (R + h) cos(latitude)."""
    if body_radius_m <= 0.0:
        raise ValueError("body_radius_m must be positive")
    if body_radius_m + altitude_m <= 0.0:
        raise ValueError("body_radius_m + altitude_m must be positive")
    if not -90.0 <= latitude_deg <= 90.0:
        raise ValueError("latitude_deg must lie between -90 and +90")
    return (body_radius_m + altitude_m) * math.cos(math.radians(latitude_deg))


def spin_speed(
    body_radius_m: float,
    rotation_period_s: float,
    altitude_m: float = 0.0,
    latitude_deg: float = 0.0,
) -> float:
    """Return signed local spin speed Omega*rho in m/s."""
    omega = angular_velocity(rotation_period_s)
    rho = cylindrical_radius(body_radius_m, altitude_m, latitude_deg)
    return omega * rho


def resultant_speed(
    center_speed_m_s: float,
    local_spin_speed_m_s: float,
    alignment_angle_deg: float,
```

```
) -> float:
    """
    Magnitude of V_center + v_spin using their included angle psi.

    v^2 = V^2 + v_spin^2 + 2 V v_spin cos(psi)

    The sign of local_spin_speed_m_s is retained, so retrograde rotation can be
    represented either by a negative rotation period or by the chosen angle.
    """
    if center_speed_m_s < 0.0:
        raise ValueError("center_speed_m_s must be non-negative")
    psi = math.radians(alignment_angle_deg)
    v2 = (
        center_speed_m_s**2
        + local_spin_speed_m_s**2
        + 2.0
        * center_speed_m_s
        * local_spin_speed_m_s
        * math.cos(psi)
    )
    # Protect against tiny negative roundoff in an exact cancellation case.
    return math.sqrt(max(v2, 0.0))


def dimensionless_label_from_speed(speed_m_s: float) -> tuple[float, float, float, float]:
    """Return beta, gamma, alpha, and K for a speed magnitude."""
    if speed_m_s < 0.0:
        raise ValueError("speed_m_s must be non-negative")
    beta = speed_m_s / C
    if beta >= 1.0:
        raise ValueError("speed must be smaller than c")
    gamma = 1.0 / math.sqrt(1.0 - beta * beta)
    alpha = 1.0 / gamma
    label_k = gamma * beta * beta
    return beta, gamma, alpha, label_k


def clock_factor_from_label(label_k: float) -> float:
    """Exact inverse map alpha(K) for K >= 0."""
    if label_k < 0.0:
        raise ValueError("label_k must be non-negative")
    # Stable form of (sqrt(K^2 + 4) - K)/2.
    return 2.0 / (math.sqrt(label_k * label_k + 4.0) + label_k)


def earth_calibration() -> dict[str, float]:
    """Return Earth's canonical equatorial calibration constants."""
    v_earth = abs(
        spin_speed(
            body_radius_m=EARTH_RADIUS_M,
            rotation_period_s=EARTH_SIDEREAL_PERIOD_S,
        )
    )
    beta, gamma, alpha, label_k = dimensionless_label_from_speed(v_earth)
    theta_s = label_k * EARTH_SIDEREAL_PERIOD_S
    return {
        "speed_m_s": v_earth,
        "beta": beta,
        "gamma": gamma,
        "alpha": alpha,
        "label_K": label_k,
        "translation_seconds": theta_s,
        "translation_nanoseconds": theta_s * 1.0e9,
    }


EARTH = earth_calibration()
EARTH_LABEL_K = EARTH["label_K"]
EARTH_TRANSLATION_S = EARTH["translation_seconds"]
```

```
EARTH_TRANSLATION_NS = EARTH["translation_nanoseconds"]


def make_record(
    *,
    name: str,
    speed_m_s: float,
    native_period_s: float | None = None,
) -> ChronometricRecord:
    """Create a complete label and Earth-translation record from speed."""
    beta, gamma, alpha, label_k = dimensionless_label_from_speed(speed_m_s)
    ratio = label_k / EARTH_LABEL_K
    earth_s = label_k * EARTH_SIDEREAL_PERIOD_S

    if native_period_s is None:
        native_s = None
        native_ns = None
        conversion = None
    else:
        period = abs(native_period_s)
        if period == 0.0:
            raise ValueError("native_period_s cannot be zero")
        native_s = label_k * period
        native_ns = native_s * 1.0e9
        conversion = EARTH_SIDEREAL_PERIOD_S / period

    return ChronometricRecord(
        name=name,
        speed_m_s=speed_m_s,
        beta=beta,
        gamma=gamma,
        clock_factor_alpha=alpha,
        label_K=label_k,
        earth_normalized_ratio=ratio,
        earth_language_seconds=earth_s,
        earth_language_nanoseconds=earth_s * 1.0e9,
        native_period_s=abs(native_period_s) if native_period_s is not None else None,
        native_cycle_equivalent_seconds=native_s,
        native_cycle_equivalent_nanoseconds=native_ns,
        native_to_earth_conversion=conversion,
    )


def canonical_rotating_body_record(
    *,
    name: str,
    radius_m: float,
    rotation_period_s: float,
    latitude_deg: float = 0.0,
    altitude_m: float = 0.0,
) -> ChronometricRecord:
    """
    Canonical spin-only record for a point fixed to a rotating body.

    For the standard body label, use latitude_deg=0 and altitude_m=0.
    """
    v_spin = abs(
        spin_speed(
            body_radius_m=radius_m,
            rotation_period_s=rotation_period_s,
            altitude_m=altitude_m,
            latitude_deg=latitude_deg,
        )
    )
    return make_record(
        name=name,
        speed_m_s=v_spin,
        native_period_s=rotation_period_s,
    )
```

```
240
241
242 def local_resultant_record(
243     *,
244     name: str,
245     radius_m: float,
246     rotation_period_s: float,
247     center_speed_m_s: float,
248     alignment_angle_deg: float,
249     latitude_deg: float = 0.0,
250     altitude_m: float = 0.0,
251 ) -> ChronometricRecord:
252     """Instantaneous local label including centre motion and local spin."""
253     v_spin = spin_speed(
254         body_radius_m=radius_m,
255         rotation_period_s=rotation_period_s,
256         altitude_m=altitude_m,
257         latitude_deg=latitude_deg,
258     )
259     v_total = resultant_speed(
260         center_speed_m_s=center_speed_m_s,
261         local_spin_speed_m_s=v_spin,
262         alignment_angle_deg=alignment_angle_deg,
263     )
264     # The native-cycle field here is only a translation convention. If the
265     # alignment changes during a cycle, it is not an accumulated physical loop
266     # observable; integrate alpha(t) for that calculation.
267     return make_record(
268         name=name,
269         speed_m_s=v_total,
270         native_period_s=rotation_period_s,
271     )
272
273
274 def print_record(record: ChronometricRecord) -> None:
275     """Print one record in a compact scientific format."""
276     print("=" * 72)
277     print(record.name)
278     print("=" * 72)
279     print(f"speed                          = {record.speed_m_s:.12g} m/s")
280     print(f"beta                           = {record.beta:.12e}")
281     print(f"gamma                          = {record.gamma:.16g}")
282     print(f"clock factor alpha             = {record.clock_factor_alpha:.16g}")
283     print(f"dimensionless label K          = {record.label_K:.12e}")
284     print(f"Earth-normalized ratio         = {record.earth_normalized_ratio:.12e}")
285     print(
286         "Earth-language translation     = "
287         f"{record.earth_language_nanoseconds:.12g} ns_E"
288     )
289     if record.native_period_s is not None:
290         print(f"native period                  = {record.native_period_s:.12g} s")
291         print(
292             "native-cycle equivalent       = "
293             f"{record.native_cycle_equivalent_nanoseconds:.12g} ns"
294         )
295         print(
296             "native -> Earth coefficient   = "
297             f"{record.native_to_earth_conversion:.12e}"
298         )
299     print()
300
301
302 def record_as_dict(record: ChronometricRecord) -> dict[str, Any]:
303     """Return a serialization-friendly dictionary."""
304     return asdict(record)
305
306
307 def build_argument_parser() -> argparse.ArgumentParser:
308     parser = argparse.ArgumentParser(
```

```
        description=(
            "Calculate a dimensionless kinematic chronometric label and its "
            "Earth-language translation."
        )
    )
    parser.add_argument("--name", default="Custom body")
    parser.add_argument("--radius-km", type=float, required=True)
    parser.add_argument(
        "--rotation-period-hours",
        type=float,
        required=True,
        help="Use a negative value for retrograde rotation.",
    )
    parser.add_argument("--latitude-deg", type=float, default=0.0)
    parser.add_argument("--altitude-km", type=float, default=0.0)
    parser.add_argument(
        "--center-speed-km-s",
        type=float,
        default=0.0,
        help="Optional centre-of-body speed in the selected reference frame.",
    )
    parser.add_argument(
        "--alignment-angle-deg",
        type=float,
        default=90.0,
        help="Angle between centre velocity and signed local spin velocity.",
    )
    return parser


def main() -> None:
    parser = build_argument_parser()
    args = parser.parse_args()

    radius_m = args.radius_km * 1_000.0
    period_s = args.rotation_period_hours * 3_600.0
    altitude_m = args.altitude_km * 1_000.0
    center_speed = args.center_speed_km_s * 1_000.0

    print("Earth calibration")
    print(f"K_E       = {EARTH_LABEL_K:.12e}")
    print(f"Theta_E   = {EARTH_TRANSLATION_NS:.12f} ns_E")
    print()

    if center_speed == 0.0:
        record = canonical_rotating_body_record(
            name=args.name,
            radius_m=radius_m,
            rotation_period_s=period_s,
            latitude_deg=args.latitude_deg,
            altitude_m=altitude_m,
        )
    else:
        record = local_resultant_record(
            name=args.name,
            radius_m=radius_m,
            rotation_period_s=period_s,
            center_speed_m_s=center_speed,
            alignment_angle_deg=args.alignment_angle_deg,
            latitude_deg=args.latitude_deg,
            altitude_m=altitude_m,
        )

    print_record(record)


if __name__ == "__main__":
    main()
```

Listing 2: Integrated rod–phase–Sagnac calculator.

```
#!/usr/bin/env python3
"""
Integrated rod--phase--Sagnac calculator.

This utility implements the executable bridge used in the combined manuscript:

1. A declared mechanical closure may provide an angular rate.
2. The angular rate and radius determine the tangential speed.
3. The speed determines the dimensionless chronometric label K.
4. A uniform ring also has a one-winding holonomy H*, a full
   counter-propagating Sagnac proper-time difference, and a circumference-
   normalised signature S.

Definitions
-----------
    beta = v / c
    gamma = 1 / sqrt(1 - beta**2)
    K = gamma * beta**2
    S = gamma * beta
    |H*| = K * T = S * C_ring / c
    Delta_tau_counter = 2 * |H*|

For the conditional rod-capacity closure:
    F_star = m * omega**2 * lever
    omega = sqrt(F_star / (m * lever))

The code does not include gravitational redshift or a universal dynamical law.
"""

from __future__ import annotations

import argparse
import math
from dataclasses import asdict, dataclass
from typing import Any

C = 299_792_458.0
EARTH_RADIUS_M = 6_378_137.0
EARTH_SIDEREAL_PERIOD_S = 86_164.09054


@dataclass(frozen=True)
class RingSignature:
    name: str
    radius_m: float
    circumference_m: float
    angular_velocity_rad_s: float
    period_s: float
    speed_m_s: float
    beta: float
    gamma: float
    clock_factor_alpha: float
    label_K: float
    circumference_signature_S: float
    one_winding_holonomy_s: float
    counter_propagating_sagnac_s: float
    earth_normalized_K: float
    earth_language_translation_s: float


def _validate_radius(radius_m: float) -> None:
    if radius_m <= 0.0:
        raise ValueError("radius_m must be positive")


def _validate_speed(speed_m_s: float) -> None:
    if not 0.0 <= speed_m_s < C:
```

```
        raise ValueError("speed magnitude must satisfy 0 <= v < c")


def earth_label() -> float:
    speed = 2.0 * math.pi * EARTH_RADIUS_M / EARTH_SIDEREAL_PERIOD_S
    beta = speed / C
    gamma = 1.0 / math.sqrt(1.0 - beta * beta)
    return gamma * beta * beta


EARTH_LABEL_K = earth_label()


def ring_from_speed(*, name: str, radius_m: float, speed_m_s: float) -> RingSignature:
    """Construct all ring signatures from radius and tangential speed."""
    _validate_radius(radius_m)
    speed = abs(speed_m_s)
    _validate_speed(speed)
    if speed == 0.0:
        raise ValueError("speed must be non-zero for a finite native period")

    circumference = 2.0 * math.pi * radius_m
    omega = speed / radius_m
    period = circumference / speed
    beta = speed / C
    gamma = 1.0 / math.sqrt(1.0 - beta * beta)
    alpha = 1.0 / gamma
    label_k = gamma * beta * beta
    signature_s = gamma * beta
    one_winding = label_k * period
    two_way = 2.0 * one_winding
    earth_translation = label_k * EARTH_SIDEREAL_PERIOD_S

    return RingSignature(
        name=name,
        radius_m=radius_m,
        circumference_m=circumference,
        angular_velocity_rad_s=omega,
        period_s=period,
        speed_m_s=speed,
        beta=beta,
        gamma=gamma,
        clock_factor_alpha=alpha,
        label_K=label_k,
        circumference_signature_S=signature_s,
        one_winding_holonomy_s=one_winding,
        counter_propagating_sagnac_s=two_way,
        earth_normalized_K=label_k / EARTH_LABEL_K,
        earth_language_translation_s=earth_translation,
    )


def ring_from_period(*, name: str, radius_m: float, period_s: float) -> RingSignature:
    """Construct all ring signatures from radius and native rotation period."""
    _validate_radius(radius_m)
    period = abs(period_s)
    if period == 0.0:
        raise ValueError("period_s must be non-zero")
    speed = 2.0 * math.pi * radius_m / period
    return ring_from_speed(name=name, radius_m=radius_m, speed_m_s=speed)


def rod_capacity_ring(
    *,
    name: str,
    force_capacity_n: float,
    terminal_mass_kg: float,
    lever_m: float,
) -> RingSignature:
```

```
    """Apply F_star = m omega^2 a and convert the result to ring signatures."""
    if force_capacity_n <= 0.0:
        raise ValueError("force_capacity_n must be positive")
    if terminal_mass_kg <= 0.0:
        raise ValueError("terminal_mass_kg must be positive")
    _validate_radius(lever_m)

    omega = math.sqrt(force_capacity_n / (terminal_mass_kg * lever_m))
    speed = omega * lever_m
    return ring_from_speed(name=name, radius_m=lever_m, speed_m_s=speed)


def equal_density_radius_ratio(distance_ratio: float) -> float:
    """At fixed capacity and angular rate, return R2/R1 = distance_ratio^(-1/3)."""
    if distance_ratio <= 0.0:
        raise ValueError("distance_ratio must be positive")
    return distance_ratio ** (-1.0 / 3.0)


def record_as_dict(record: RingSignature) -> dict[str, Any]:
    return asdict(record)


def print_record(record: RingSignature) -> None:
    print("=" * 78)
    print(record.name)
    print("=" * 78)
    print(f"radius                          = {record.radius_m:.12g} m")
    print(f"circumference                   = {record.circumference_m:.12g} m")
    print(f"speed                           = {record.speed_m_s:.12g} m/s")
    print(f"angular velocity                = {record.angular_velocity_rad_s:.12e} rad/s")
    print(f"native period                   = {record.period_s:.12g} s")
    print(f"beta                            = {record.beta:.12e}")
    print(f"gamma                           = {record.gamma:.16g}")
    print(f"clock factor alpha              = {record.clock_factor_alpha:.16g}")
    print(f"period-normalised label K       = {record.label_K:.12e}")
    print(f"circumference signature S       = {record.circumference_signature_S:.12e}")
    print(f"one-winding holonomy |H*|       = {record.one_winding_holonomy_s:.12e} s")
    print(f"counter-propagating Delta_tau   = {record.counter_propagating_sagnac_s:.12e} s")
    print(f"Earth-normalised K ratio        = {record.earth_normalized_K:.12e}")
    print(f"Earth-language translation      = {record.earth_language_translation_s:.12e} s_E")


def build_parser() -> argparse.ArgumentParser:
    parser = argparse.ArgumentParser(description=__doc__)
    sub = parser.add_subparsers(dest="mode", required=True)

    ring = sub.add_parser("ring", help="Calculate signatures for a uniform ring")
    ring.add_argument("--name", default="Uniform ring")
    ring.add_argument("--radius-m", type=float, required=True)
    source = ring.add_mutually_exclusive_group(required=True)
    source.add_argument("--speed-m-s", type=float)
    source.add_argument("--period-s", type=float)

    rod = sub.add_parser("rod", help="Apply the conditional rod-capacity closure")
    rod.add_argument("--name", default="Rod terminal")
    rod.add_argument("--force-n", type=float, required=True)
    rod.add_argument("--mass-kg", type=float, required=True)
    rod.add_argument("--lever-m", type=float, required=True)
    rod.add_argument(
        "--distance-ratio",
        type=float,
        help="Also report the equal-density terminal radius ratio r^(-1/3).",
    )
    return parser


def main() -> None:
    args = build_parser().parse_args()
```

```
206
207     if args.mode == "ring":
208         if args.speed_m_s is not None:
209             record = ring_from_speed(
210                 name=args.name,
211                 radius_m=args.radius_m,
212                 speed_m_s=args.speed_m_s,
213             )
214         else:
215             record = ring_from_period(
216                 name=args.name,
217                 radius_m=args.radius_m,
218                 period_s=args.period_s,
219             )
220     else:
221         record = rod_capacity_ring(
222             name=args.name,
223             force_capacity_n=args.force_n,
224             terminal_mass_kg=args.mass_kg,
225             lever_m=args.lever_m,
226         )
227
228     print_record(record)
229
230     if args.mode == "rod" and args.distance_ratio is not None:
231         ratio = equal_density_radius_ratio(args.distance_ratio)
232         print(f"equal-density terminal radius ratio = {ratio:.12g}")
233
234
235 if __name__ == "__main__":
236     main()
```

Listing 3: Galactic phase-space chronometric atlas generator.

```
1  #!/usr/bin/env python3
2  """
3  Galactic phase-space chronometric atlas generator.
4
5  This script implements the gravity-independent kinematic sector developed in
6  "Nested Angular Progress, Galactic Phase-Space Addressing, and Earth-Language
7  Chronometric Labels".
8
9  The fundamental catalogue input is a declared Galactocentric phase-space state
10 at one epoch:
11
12     (x, y, z, vx, vy, vz)
13
14 Position alone does not determine velocity.  Rows without measured velocities
15 are therefore either marked INCOMPLETE or completed by an explicitly selected
16 conditional velocity-field closure.
17
18 For each complete row the script calculates:
19
20     R, phi, z, v_R, v_phi, v_z, |v|,
21     beta, gamma, alpha=d_tau/dt,
22     K=gamma*beta^2,
23     K/K_Earth,
24     Theta_Earth=K*T_sidereal_Earth.
25
26 The Earth-language interval is a comparison coordinate.  It is not claimed to
27 be a native physical loop holonomy unless a separate closed-loop construction
28 applies to the object being represented.
29
30 No gravitational potential or gravitational redshift is included.
31
32 Optional dependency:
33     astropy -- only required for ICRS/Gaia-style sky-coordinate transformation.
34
```

```
Author: S. M. H. Emamifar
Affiliation: IRCBHC
"""

from __future__ import annotations

import argparse
import csv
import json
import math
import random
import statistics
from dataclasses import asdict, dataclass
from pathlib import Path
from typing import Any, Iterable

C_M_S = 299_792_458.0
KPC_M = 3.085_677_581_491_367e19
KM_S_TO_M_S = 1_000.0

EARTH_RADIUS_M = 6_378_137.0
EARTH_SIDEREAL_PERIOD_S = 86_164.09054
EARTH_OMEGA_RAD_S = 2.0 * math.pi / EARTH_SIDEREAL_PERIOD_S
EARTH_EQUATORIAL_SPEED_M_S = EARTH_OMEGA_RAD_S * EARTH_RADIUS_M

# Explicit illustrative Galactocentric frame parameters.  They match the
# Astropy v4.0 example convention and are not asserted to be a unique standard.
DEFAULT_GALCEN_DISTANCE_KPC = 8.122
DEFAULT_Z_SUN_PC = 20.8
DEFAULT_V_SUN_KM_S = (12.9, 245.6, 7.78)


@dataclass(frozen=True)
class Vector3:
    x: float
    y: float
    z: float

    def norm(self) -> float:
        return math.sqrt(self.x * self.x + self.y * self.y + self.z * self.z)

    def __add__(self, other: "Vector3") -> "Vector3":
        return Vector3(self.x + other.x, self.y + other.y, self.z + other.z)

    def scale(self, factor: float) -> "Vector3":
        return Vector3(self.x * factor, self.y * factor, self.z * factor)


@dataclass(frozen=True)
class LabelState:
    speed_m_s: float
    beta: float
    gamma: float
    alpha: float
    label_K: float
    earth_ratio: float
    earth_translation_s: float
    earth_translation_ns: float


@dataclass(frozen=True)
class GalacticChronometricRecord:
    name: str
    epoch: str
    frame_name: str
    quality_class: str
    closure_name: str
    x_kpc: float
    y_kpc: float
```

```
    z_kpc: float
    R_kpc: float
    phi_deg: float
    vx_km_s: float | None
    vy_km_s: float | None
    vz_km_s: float | None
    vR_km_s: float | None
    vphi_km_s: float | None
    speed_km_s: float | None
    beta: float | None
    gamma: float | None
    alpha: float | None
    label_K: float | None
    earth_ratio: float | None
    earth_translation_s: float | None
    earth_translation_ms: float | None
    label_K_sigma: float | None
    earth_translation_ms_sigma: float | None
    notes: str


def label_from_speed_m_s(speed_m_s: float) -> LabelState:
    """Calculate the exact dimensionless label and Earth translation."""
    if speed_m_s < 0.0:
        raise ValueError("speed_m_s must be non-negative")
    beta = speed_m_s / C_M_S
    if beta >= 1.0:
        raise ValueError("speed must be smaller than c")
    gamma = 1.0 / math.sqrt(1.0 - beta * beta)
    alpha = 1.0 / gamma
    label_k = gamma * beta * beta
    return LabelState(
        speed_m_s=speed_m_s,
        beta=beta,
        gamma=gamma,
        alpha=alpha,
        label_K=label_k,
        earth_ratio=label_k / EARTH_LABEL.label_K,
        earth_translation_s=label_k * EARTH_SIDEREAL_PERIOD_S,
        earth_translation_ns=label_k * EARTH_SIDEREAL_PERIOD_S * 1.0e9,
    )


# Bootstrap Earth after the function exists.  The temporary calculation is
# written explicitly to avoid recursive access to EARTH_LABEL.
def _earth_label() -> LabelState:
    beta = EARTH_EQUATORIAL_SPEED_M_S / C_M_S
    gamma = 1.0 / math.sqrt(1.0 - beta * beta)
    alpha = 1.0 / gamma
    label_k = gamma * beta * beta
    return LabelState(
        speed_m_s=EARTH_EQUATORIAL_SPEED_M_S,
        beta=beta,
        gamma=gamma,
        alpha=alpha,
        label_K=label_k,
        earth_ratio=1.0,
        earth_translation_s=label_k * EARTH_SIDEREAL_PERIOD_S,
        earth_translation_ns=label_k * EARTH_SIDEREAL_PERIOD_S * 1.0e9,
    )


EARTH_LABEL = _earth_label()


def cartesian_to_cylindrical(
    position_kpc: Vector3,
    velocity_km_s: Vector3 | None,
) -> tuple[float, float, float | None, float | None]:
```

```
    """Return R, phi(deg), v_R and v_phi in a right-handed frame."""
    x, y = position_kpc.x, position_kpc.y
    radius = math.hypot(x, y)
    phi_deg = math.degrees(math.atan2(y, x))
    if velocity_km_s is None:
        return radius, phi_deg, None, None
    if radius == 0.0:
        return radius, phi_deg, None, None
    v_r = (x * velocity_km_s.x + y * velocity_km_s.y) / radius
    v_phi = (x * velocity_km_s.y - y * velocity_km_s.x) / radius
    return radius, phi_deg, v_r, v_phi


def flat_rotation_closure(position_kpc: Vector3, speed_km_s: float) -> Vector3:
    """Conditional circular field with constant signed tangential speed."""
    radius = math.hypot(position_kpc.x, position_kpc.y)
    if radius == 0.0:
        raise ValueError("flat rotation closure is undefined at R=0")
    return Vector3(
        -speed_km_s * position_kpc.y / radius,
        speed_km_s * position_kpc.x / radius,
        0.0,
    )


def solid_rotation_closure(
    position_kpc: Vector3,
    omega_km_s_kpc: float,
) -> Vector3:
    """Conditional solid-rotation field v_phi=Omega*R."""
    return Vector3(
        -omega_km_s_kpc * position_kpc.y,
        omega_km_s_kpc * position_kpc.x,
        0.0,
    )


def apply_velocity_closure(
    position_kpc: Vector3,
    closure: str,
    flat_speed_km_s: float,
    solid_omega_km_s_kpc: float,
) -> tuple[Vector3 | None, str, str]:
    if closure == "none":
        return None, "P3_POSITION_ONLY", "none"
    if closure == "flat":
        return (
            flat_rotation_closure(position_kpc, flat_speed_km_s),
            "C6_MODEL_CLOSED",
            f"flat_rotation({flat_speed_km_s:g} km/s)",
        )
    if closure == "solid":
        return (
            solid_rotation_closure(position_kpc, solid_omega_km_s_kpc),
            "C6_MODEL_CLOSED",
            f"solid_rotation({solid_omega_km_s_kpc:g} km/s/kpc)",
        )
    raise ValueError(f"unsupported closure: {closure}")


def monte_carlo_label_sigma(
    velocity_km_s: Vector3,
    sigma_km_s: Vector3,
    samples: int,
    seed: int,
) -> tuple[float | None, float | None]:
    """Diagonal-error Monte Carlo. Full covariance should be used when available."""
    if samples <= 1:
        return None, None
```

```
    if sigma_km_s.x < 0 or sigma_km_s.y < 0 or sigma_km_s.z < 0:
        raise ValueError("velocity uncertainties must be non-negative")
    if sigma_km_s.norm() == 0.0:
        return 0.0, 0.0
    rng = random.Random(seed)
    labels: list[float] = []
    translations_ms: list[float] = []
    for _ in range(samples):
        draw = Vector3(
            rng.gauss(velocity_km_s.x, sigma_km_s.x),
            rng.gauss(velocity_km_s.y, sigma_km_s.y),
            rng.gauss(velocity_km_s.z, sigma_km_s.z),
        )
        state = label_from_speed_m_s(draw.norm() * KM_S_TO_M_S)
        labels.append(state.label_K)
        translations_ms.append(state.earth_translation_s * 1.0e3)
    return statistics.stdev(labels), statistics.stdev(translations_ms)


def build_record(
    *,
    name: str,
    epoch: str,
    frame_name: str,
    position_kpc: Vector3,
    velocity_km_s: Vector3 | None,
    quality_class: str,
    closure_name: str,
    velocity_sigma_km_s: Vector3 | None = None,
    mc_samples: int = 0,
    seed: int = 20260722,
    notes: str = "",
) -> GalacticChronometricRecord:
    R_kpc, phi_deg, v_r, v_phi = cartesian_to_cylindrical(
        position_kpc, velocity_km_s
    )

    if velocity_km_s is None:
        return GalacticChronometricRecord(
            name=name,
            epoch=epoch,
            frame_name=frame_name,
            quality_class=quality_class,
            closure_name=closure_name,
            x_kpc=position_kpc.x,
            y_kpc=position_kpc.y,
            z_kpc=position_kpc.z,
            R_kpc=R_kpc,
            phi_deg=phi_deg,
            vx_km_s=None,
            vy_km_s=None,
            vz_km_s=None,
            vR_km_s=None,
            vphi_km_s=None,
            speed_km_s=None,
            beta=None,
            gamma=None,
            alpha=None,
            label_K=None,
            earth_ratio=None,
            earth_translation_s=None,
            earth_translation_ms=None,
            label_K_sigma=None,
            earth_translation_ms_sigma=None,
            notes=notes,
        )

    speed_km_s = velocity_km_s.norm()
    state = label_from_speed_m_s(speed_km_s * KM_S_TO_M_S)
```

```
    if velocity_sigma_km_s is None:
        k_sigma, theta_sigma = None, None
    else:
        k_sigma, theta_sigma = monte_carlo_label_sigma(
            velocity_km_s,
            velocity_sigma_km_s,
            mc_samples,
            seed,
        )

    return GalacticChronometricRecord(
        name=name,
        epoch=epoch,
        frame_name=frame_name,
        quality_class=quality_class,
        closure_name=closure_name,
        x_kpc=position_kpc.x,
        y_kpc=position_kpc.y,
        z_kpc=position_kpc.z,
        R_kpc=R_kpc,
        phi_deg=phi_deg,
        vx_km_s=velocity_km_s.x,
        vy_km_s=velocity_km_s.y,
        vz_km_s=velocity_km_s.z,
        vR_km_s=v_r,
        vphi_km_s=v_phi,
        speed_km_s=speed_km_s,
        beta=state.beta,
        gamma=state.gamma,
        alpha=state.alpha,
        label_K=state.label_K,
        earth_ratio=state.earth_ratio,
        earth_translation_s=state.earth_translation_s,
        earth_translation_ms=state.earth_translation_s * 1.0e3,
        label_K_sigma=k_sigma,
        earth_translation_ms_sigma=theta_sigma,
        notes=notes,
    )


def parse_optional_float(row: dict[str, str], key: str) -> float | None:
    value = (row.get(key) or "").strip()
    if value == "":
        return None
    return float(value)


def records_from_cartesian_csv(
    input_path: Path,
    *,
    closure: str,
    flat_speed_km_s: float,
    solid_omega_km_s_kpc: float,
    frame_name: str,
    default_epoch: str,
    mc_samples: int,
) -> list[GalacticChronometricRecord]:
    records: list[GalacticChronometricRecord] = []
    with input_path.open("r", encoding="utf-8-sig", newline="") as handle:
        reader = csv.DictReader(handle)
        required = {"name", "x_kpc", "y_kpc", "z_kpc"}
        missing = required.difference(reader.fieldnames or [])
        if missing:
            raise ValueError(f"missing CSV columns: {sorted(missing)}")

        for index, row in enumerate(reader, start=1):
            position = Vector3(
                float(row["x_kpc"]),
                float(row["y_kpc"]),
```

```
                float(row["z_kpc"]),
            )
            vx = parse_optional_float(row, "vx_km_s")
            vy = parse_optional_float(row, "vy_km_s")
            vz = parse_optional_float(row, "vz_km_s")
            velocity_values = (vx, vy, vz)

            if all(value is not None for value in velocity_values):
                velocity = Vector3(float(vx), float(vy), float(vz))
                quality = "M6_MEASURED"
                closure_name = "none"
            elif all(value is None for value in velocity_values):
                velocity, quality, closure_name = apply_velocity_closure(
                    position,
                    closure,
                    flat_speed_km_s,
                    solid_omega_km_s_kpc,
                )
            else:
                velocity = None
                quality = "P5_PARTIAL_VELOCITY"
                closure_name = "none"

            svx = parse_optional_float(row, "vx_error_km_s") or 0.0
            svy = parse_optional_float(row, "vy_error_km_s") or 0.0
            svz = parse_optional_float(row, "vz_error_km_s") or 0.0
            sigma = Vector3(svx, svy, svz) if velocity is not None else None

            records.append(
                build_record(
                    name=(row.get("name") or f"row-{index}").strip(),
                    epoch=(row.get("epoch") or default_epoch).strip(),
                    frame_name=(row.get("frame_name") or frame_name).strip(),
                    position_kpc=position,
                    velocity_km_s=velocity,
                    quality_class=quality,
                    closure_name=closure_name,
                    velocity_sigma_km_s=sigma,
                    mc_samples=mc_samples,
                    seed=20260722 + index,
                    notes=(row.get("notes") or "").strip(),
                )
            )
    return records


def icrs_to_galactocentric(
    *,
    ra_deg: float,
    dec_deg: float,
    distance_pc: float,
    pmra_mas_yr: float,
    pmdec_mas_yr: float,
    radial_velocity_km_s: float,
    galcen_distance_kpc: float = DEFAULT_GALCEN_DISTANCE_KPC,
    z_sun_pc: float = DEFAULT_Z_SUN_PC,
    v_sun_km_s: tuple[float, float, float] = DEFAULT_V_SUN_KM_S,
) -> tuple[Vector3, Vector3]:
    """Transform one full ICRS phase-space state using explicit Astropy settings."""
    try:
        import astropy.coordinates as coord
        import astropy.units as u
    except ImportError as exc:
        raise RuntimeError(
            "Astropy is required for ICRS transformation. Install with: "
            "pip install astropy"
        ) from exc

    sky = coord.SkyCoord(
```

```
        ra=ra_deg * u.deg,
        dec=dec_deg * u.deg,
        distance=distance_pc * u.pc,
        pm_ra_cosdec=pmra_mas_yr * u.mas / u.yr,
        pm_dec=pmdec_mas_yr * u.mas / u.yr,
        radial_velocity=radial_velocity_km_s * u.km / u.s,
        frame="icrs",
    )
    velocity_components = tuple(
        component * u.km / u.s for component in v_sun_km_s
    )
    # Astropy versions before 8 use CartesianDifferential here, whereas
    # Astropy 8+ accepts CartesianRepresentation.  Keep both paths explicit.
    try:
        solar_velocity = coord.CartesianRepresentation(*velocity_components)
        frame = coord.Galactocentric(
            galcen_distance=galcen_distance_kpc * u.kpc,
            z_sun=z_sun_pc * u.pc,
            galcen_v_sun=solar_velocity,
        )
    except (TypeError, ValueError):
        solar_velocity = coord.CartesianDifferential(*velocity_components)
        frame = coord.Galactocentric(
            galcen_distance=galcen_distance_kpc * u.kpc,
            z_sun=z_sun_pc * u.pc,
            galcen_v_sun=solar_velocity,
        )
    gc = sky.transform_to(frame)
    position = Vector3(
        gc.x.to_value(u.kpc),
        gc.y.to_value(u.kpc),
        gc.z.to_value(u.kpc),
    )
    velocity = Vector3(
        gc.v_x.to_value(u.km / u.s),
        gc.v_y.to_value(u.km / u.s),
        gc.v_z.to_value(u.km / u.s),
    )
    return position, velocity


def records_from_icrs_csv(
    input_path: Path,
    *,
    frame_name: str,
    default_epoch: str,
) -> list[GalacticChronometricRecord]:
    records: list[GalacticChronometricRecord] = []
    with input_path.open("r", encoding="utf-8-sig", newline="") as handle:
        reader = csv.DictReader(handle)
        required = {
            "name",
            "ra_deg",
            "dec_deg",
            "distance_pc",
            "pmra_mas_yr",
            "pmdec_mas_yr",
            "radial_velocity_km_s",
        }
        missing = required.difference(reader.fieldnames or [])
        if missing:
            raise ValueError(f"missing ICRS CSV columns: {sorted(missing)}")
        for row in reader:
            position, velocity = icrs_to_galactocentric(
                ra_deg=float(row["ra_deg"]),
                dec_deg=float(row["dec_deg"]),
                distance_pc=float(row["distance_pc"]),
                pmra_mas_yr=float(row["pmra_mas_yr"]),
                pmdec_mas_yr=float(row["pmdec_mas_yr"]),
```

```
                radial_velocity_km_s=float(row["radial_velocity_km_s"]),
            )
            records.append(
                build_record(
                    name=row["name"].strip(),
                    epoch=(row.get("epoch") or default_epoch).strip(),
                    frame_name=frame_name,
                    position_kpc=position,
                    velocity_km_s=velocity,
                    quality_class="T6_TRANSFORMED",
                    closure_name="none",
                    notes=(row.get("notes") or "").strip(),
                )
            )
    return records


def write_records_csv(records: Iterable[GalacticChronometricRecord], path: Path) -> None:
    rows = [asdict(record) for record in records]
    if not rows:
        raise ValueError("no records to write")
    with path.open("w", encoding="utf-8", newline="") as handle:
        writer = csv.DictWriter(handle, fieldnames=list(rows[0].keys()))
        writer.writeheader()
        writer.writerows(rows)


def write_records_json(records: Iterable[GalacticChronometricRecord], path: Path) -> None:
    with path.open("w", encoding="utf-8") as handle:
        json.dump([asdict(record) for record in records], handle, indent=2)


def print_summary(records: list[GalacticChronometricRecord]) -> None:
    print("Earth canonical calibration")
    print(f"  K_Earth    = {EARTH_LABEL.label_K:.12e}")
    print(f"  Theta_Earth= {EARTH_LABEL.earth_translation_ns:.12f} ns_E")
    print()
    header = (
        f"{'name':24s} {'quality':25s} {'speed km/s':>12s} "
        f"{'K':>13s} {'K/K_E':>12s} {'Theta_E ms':>12s}"
    )
    print(header)
    print("-" * len(header))
    for record in records:
        if record.speed_km_s is None:
            speed = k_value = ratio = theta = "--"
        else:
            speed = f"{record.speed_km_s:.6g}"
            k_value = f"{record.label_K:.6e}"
            ratio = f"{record.earth_ratio:.6e}"
            theta = f"{record.earth_translation_ms:.6g}"
        print(
            f"{record.name[:24]:24s} {record.quality_class[:25]:25s} "
            f"{speed:>12s} {k_value:>13s} {ratio:>12s} {theta:>12s}"
        )


def build_parser() -> argparse.ArgumentParser:
    parser = argparse.ArgumentParser(
        description="Build a gravity-independent Galactic chronometric atlas."
    )
    subparsers = parser.add_subparsers(dest="command", required=True)

    cart = subparsers.add_parser("cartesian-csv", help="Process Galactocentric Cartesian CSV")
    cart.add_argument("input_csv", type=Path)
    cart.add_argument("--output-csv", type=Path, default=Path("atlas_output.csv"))
    cart.add_argument("--output-json", type=Path)
    cart.add_argument("--frame-name", default="Declared Galactocentric frame")
    cart.add_argument("--epoch", default="J2016.0")
```

```
    cart.add_argument("--closure", choices=("none", "flat", "solid"), default="none")
    cart.add_argument("--flat-speed-km-s", type=float, default=220.0)
    cart.add_argument("--solid-omega-km-s-kpc", type=float, default=27.0)
    cart.add_argument("--mc-samples", type=int, default=0)

    icrs = subparsers.add_parser("icrs-csv", help="Transform full ICRS CSV with Astropy")
    icrs.add_argument("input_csv", type=Path)
    icrs.add_argument("--output-csv", type=Path, default=Path("atlas_output.csv"))
    icrs.add_argument("--output-json", type=Path)
    icrs.add_argument("--frame-name", default="Astropy explicit Galactocentric frame")
    icrs.add_argument("--epoch", default="J2016.0")

    single = subparsers.add_parser("single", help="Calculate one Cartesian phase-space state")
    single.add_argument("--name", default="Custom object")
    single.add_argument("--epoch", default="J2016.0")
    single.add_argument("--frame-name", default="Declared Galactocentric frame")
    single.add_argument("--x-kpc", type=float, required=True)
    single.add_argument("--y-kpc", type=float, required=True)
    single.add_argument("--z-kpc", type=float, required=True)
    single.add_argument("--vx-km-s", type=float, required=True)
    single.add_argument("--vy-km-s", type=float, required=True)
    single.add_argument("--vz-km-s", type=float, required=True)
    return parser


def main() -> None:
    args = build_parser().parse_args()
    if args.command == "cartesian-csv":
        records = records_from_cartesian_csv(
            args.input_csv,
            closure=args.closure,
            flat_speed_km_s=args.flat_speed_km_s,
            solid_omega_km_s_kpc=args.solid_omega_km_s_kpc,
            frame_name=args.frame_name,
            default_epoch=args.epoch,
            mc_samples=args.mc_samples,
        )
        write_records_csv(records, args.output_csv)
        if args.output_json:
            write_records_json(records, args.output_json)
        print_summary(records)
        return

    if args.command == "icrs-csv":
        records = records_from_icrs_csv(
            args.input_csv,
            frame_name=args.frame_name,
            default_epoch=args.epoch,
        )
        write_records_csv(records, args.output_csv)
        if args.output_json:
            write_records_json(records, args.output_json)
        print_summary(records)
        return

    if args.command == "single":
        record = build_record(
            name=args.name,
            epoch=args.epoch,
            frame_name=args.frame_name,
            position_kpc=Vector3(args.x_kpc, args.y_kpc, args.z_kpc),
            velocity_km_s=Vector3(args.vx_km_s, args.vy_km_s, args.vz_km_s),
            quality_class="M6_MEASURED",
            closure_name="none",
        )
        print_summary([record])
        print(json.dumps(asdict(record), indent=2))
        return

```

```
    raise RuntimeError("unreachable command")


if __name__ == "__main__":
    main()
```

F On the Novelty, Architecture, and Applications of the Framework

This appendix clarifies the nature of the contribution and the intended scope of the framework. It does not modify any mathematical result or physical claim derived in the main text. It provides a transparent description of the architectural organisation of standard kinematics, phase bookkeeping, and chronometric labelling used in this work.

F.1 Standard building blocks, novel architecture

The individual physical ingredients employed here are established: special-relativistic proper-time relations, vector composition of velocities generated by nested rotations and translations, and standard Galactic coordinate descriptions.

The contribution of this work is not the introduction of a new gravitational law or a replacement for General Relativity. Rather, it is the systematic organisation of these ingredients into a single operational framework consisting of:

1. a signed, unwrapped nested phase ledger separating spin, orbital, parent-system, and Galactic phases;

2. a point-dependent phase metric encoding the embedding geometry and the resulting velocity norm within the stated construction;

3. a dimensionless chronometric label

$$\mathscr{K}_X = \gamma_X \frac{v_X^2}{c^2},$$

   together with its inverse mapping to the local kinematic clock factor;

4. a Local Observer Translation Map convention

$$\Theta_X^{(\oplus)} = \mathscr{K}_X T_{\mathrm{sid},\oplus},$$

   which is a comparison coordinate and not a universal unit of time;

5. a quality-controlled Galactic phase-space atlas that distinguishes measured states, incomplete states, and model-completed states.

The framework therefore represents a new organisation of known kinematical territory rather than a claim of new fundamental dynamics.

F.2 What the framework is not

The following limitations are explicit:

- The framework is not a replacement for General Relativity or any metric theory of gravity. Gravitational redshift and strong-field corrections must be added separately when total observed clock rates are required.
- It does not determine orbital or spin rates from position and size alone. Angular rates entering the chronometric equations must be supplied by observation or by an explicitly declared dynamical closure.
- The quantity $\mathscr{K}_X$ is not an absolute scalar independent of convention. Reference frame, epoch, and point definition must always be specified.
- The local-observer quantity $\Theta^{(O)}$ is a calibration for comparison with terrestrial units, not a universal temporal quantum.

F.3 Intended applications

The architecture is intended for:

1. comparison of kinematic chronometric states of planets, stars, and catalogued Galactic sources;
2. separation of kinematic and gravitational contributions in precision timing problems;
3. pedagogical visualisation of the chain

$$\text{phase ledger} \to \text{phase geometry} \to \text{velocity} \to \mathscr{K} \to \text{local translation};$$

4. future construction of empirical chronometric atlases when reliable relations between measured positions and velocity fields are available;
5. enrichment of astronomical catalogues with dimensionless kinematic labels and terrestrial comparison coordinates.

F.4 Epistemic hierarchy of quantities

The framework separates quantities according to their origin:

Level	Quantity	Status
Observation	position, velocity, period	measured input
Derivation	$\mathscr{K}_X$, α_X	mathematical output
Translation	$\Theta_X^{(\oplus)}$	comparison convention
Closure model	position–velocity relation	explicit hypothesis

This hierarchy prevents confusion between observed data, mathematical consequences, and future modelling assumptions.

G Galactic Reference Chronometry and Local Observer Translation

The chronometric framework does not assign a privileged status to the Earth. The Earth is one possible local observer among many. The primary comparison structure is an operational Galactic reference frame $\mathcal{F}_{\mathrm{MW}}$, defined by a declared origin, orientation, and reference epoch.

This choice does not introduce an absolute cosmic time. It provides a common kinematic language for comparing celestial systems.

G.1 Galactic reference state

For an object X, define the complete state

$$\boxed{\mathcal{A}_X(t_0) = (\mathbf{x}_X, \mathbf{v}_X, \boldsymbol{N}_X, \mathcal{F}_{\mathrm{MW}}, t_0)} \tag{154}$$

where position, velocity, nested phase ledger, reference frame, and epoch are recorded together.

Position alone does not determine a chronometric label:

$$\mathbf{x}_X \not\Rightarrow \mathscr{K}_X. \tag{155}$$

A complete phase-space state is required:

$$(\mathbf{x}_X, \mathbf{v}_X) \Rightarrow \mathscr{K}_X. \tag{156}$$

G.2 Frame-declared dimensionless kinematic label

The dimensionless kinematic label is defined by

$$\boxed{\mathscr{K}_X = \gamma_X \beta_X^2} \tag{157}$$

with

$$\beta_X = \frac{|\mathbf{v}_X|}{c}, \qquad \gamma_X = (1 - \beta_X^2)^{-1/2}. \tag{158}$$

The associated kinematic clock factor is

$$\boxed{\alpha_X = \frac{\sqrt{\mathscr{K}_X^2 + 4} - \mathscr{K}_X}{2}} \tag{159}$$

G.3 Local observer translation

The dimensionless label is primary. A local observer O may translate it into the observer's own units:

$$\boxed{\Theta_X^{(O)} = \mathscr{K}_X T_O} \tag{160}$$

The Earth expression is only the special case $O = \oplus$:

$$\Theta_X^{(\oplus)} = \mathscr{K}_X T_\oplus. \tag{161}$$

Therefore terrestrial units are a calibration language, not the fundamental definition of the framework.

Author Declaration

This manuscript unifies a mechanical thought experiment with a phase-geometric chronometric and Galactic catalogue architecture. The rod and spinning-top model is presented as a controlled conceptual and conditional mechanical closure, not as a literal description of the Solar System or the Milky Way. The kinematic proper-time, velocity-composition, and rotating-ring Sagnac relations are used within their declared reference-frame assumptions. The values $\mathscr{K}$, $\mathscr{S}$, and local-observer translations are not absolute names detached from frame, epoch, path, or point convention. The value 207.386 ns denotes the canonical terrestrial one-winding translation; it is not a universal time quantum. Any Galactic position-to-velocity law remains an explicitly declared closure and must not be confused with measured six-dimensional phase-space data.

S. M. H. Emamifar
Independent Research Collaboration on Black Hole and Cosmology Concepts (IRCBHC)
ORCID: 0009-0007-6257-0163

References

[1] A. Einstein, "Zur Elektrodynamik bewegter Koerper," *Annalen der Physik* **322**, 891–921 (1905), doi:10.1002/andp.19053221004.

[2] W. Rindler, *Relativity: Special, General, and Cosmological*, 2nd ed., Oxford University Press (2006).

[3] G. Petit and B. Luzum (eds.), *IERS Conventions (2010)*, IERS Technical Note No. 36, Verlag des Bundesamts fuer Kartographie und Geodaesie, Frankfurt am Main (2010), https://iers-conventions.obspm.fr/.

[4] P. Charlot et al., "The Third Realization of the International Celestial Reference Frame by Very Long Baseline Interferometry," *Astronomy & Astrophysics* **644**, A159 (2020), doi:10.1051/0004-6361/202038368.

[5] NASA Jet Propulsion Laboratory, Solar System Dynamics, "Planetary Physical Parameters," https://ssd.jpl.nasa.gov/planets/phys_par.html.

[6] NASA Jet Propulsion Laboratory, Solar System Dynamics, "Astrodynamic Parameters," https://ssd.jpl.nasa.gov/astro_par.html.

[7] NASA Jet Propulsion Laboratory, Solar System Dynamics, "Orbits and Ephemerides," https://ssd.jpl.nasa.gov/orbits.html.

[8] B. A. Archinal, C. H. Acton, M. F. A'Hearn, et al., "Report of the IAU Working Group on Cartographic Coordinates and Rotational Elements: 2015," *Celestial Mechanics and Dynamical Astronomy* **130**, 22 (2018), doi:10.1007/s10569-017-9805-5.

[9] National Geospatial-Intelligence Agency, *World Geodetic System 1984: Its Definition and Relationships with Local Geodetic Systems*, NGA.STND.0036_1.0.0 (2014).

[10] N. Ashby, "Relativity in the Global Positioning System," *Living Reviews in Relativity* **6**, 1 (2003), doi:10.12942/lrr-2003-1.

[11] G. Sagnac, "L'ether lumineux demontre par l'effet du vent relatif d'ether dans un interferometre en rotation uniforme," *Comptes Rendus de l'Academie des Sciences* **157**, 708–710 (1913).

[12] Gaia Collaboration, A. Vallenari, A. G. A. Brown, et al., "Gaia Data Release 3: Summary of the Content and Survey Properties," *Astronomy & Astrophysics* **674**, A1 (2023), doi:10.1051/0004-6361/202243940.

[13] D. Katz, P. Sartoretti, A. Guerrier, et al., "Gaia Data Release 3: Properties and Validation of the Radial Velocities," *Astronomy & Astrophysics* **674**, A5 (2023), doi:10.1051/0004-6361/202244220.

[14] European Space Agency, Gaia Archive, "Gaia DR3 Source Catalogue Data Model," https://gea.esac.esa.int/archive/documentation/GDR3/Gaia_archive/chap_datamodel/sec_dm_main_source_catalogue/ssec_dm_gaia_source.html.

[15] Astropy Collaboration, "Description of the Galactocentric Coordinate Frame," *Astropy Coordinates Documentation*, https://docs.astropy.org/en/stable/coordinates/galactocentric.html.

[16] Gaia Collaboration, R. Drimmel, M. Romero-Gomez, et al., "Gaia Data Release 3: Mapping the Asymmetric Disc of the Milky Way," *Astronomy & Astrophysics* **674**, A37 (2023), doi:10.1051/0004-6361/202243797.

[17] I. Newton, *Philosophiae Naturalis Principia Mathematica*, London (1687).

[18] J. Kepler, *Harmonices Mundi*, Linz (1619).

[19] E. Mach, *The Science of Mechanics*, Open Court Publishing (1883).