ACF 79 — Lean 4 Proof Dossier

The Source

Substitution-Initiated Self-Separation of the Substituted-Chrysene Stock from a Polyaromatic Bulk, and the Supply It Delivers to the Protocell

Charles D. Schaper, Ph.D., Schaper Systems Press

Annals of the Chrysene Formalism, Vol. 1, Issue 4

Proof file: `acf-79-proof.lean` Verification harness: `verify_A79.py`

Kernel Certification

Item	Value
Proof file	`acf-79-proof.lean`
Lean version	`4.12.0` (core only, no Mathlib)
Namespace	`ChryseneFormalism.Article79`
Sections	S2–S14
Theorems	55
Proved by `decide`	55
Depending on no axioms	**54**
Depending on `propext` only	1
Depending on `Quot.sound` or `funext`	0
`sorry`	0
`native_decide`	0
Elaboration time	under one second

What This Certifies

The Lean 4.12.0 kernel independently verifies 55 finite arithmetic claims from Article 79. Each theorem is stated over `Nat` and discharged by `decide`: the goal is a decidable proposition, the kernel evaluates a verified Boolean decision procedure, and the proof is accepted only when that evaluation yields `true`. No external oracle is consulted; no `native_decide`; no `sorry`; no Mathlib import.

The axiom result is stronger than the usual one. An `#print axioms` audit over every theorem in the file reports *"does not depend on any axioms"* for 54 of the 55. The single exception is `demand_gcd`, which calls `Nat.gcd` and thereby inherits `propext` from the Lean core. `Quot.sound`, `funext` and `Classical.choice` appear nowhere. The dossier states this explicitly rather than quoting the customary three-axiom footprint, because the customary footprint would overstate what this file assumes.

The proof covers: the imported demand and its reduction to lowest terms; rounding certificates for the two demand ratios; the geometric arm ladder of H_{add} and the conclusion of Theorem 5.1 (Ordering); the exit distribution's normalisation and the design equation of Theorem 7.3; the matched efficiency against its lower bound; the cooperativity quotient; the initiator surplus; the window and completion thresholds of the contrast case; Burnside orbit counts for arm words; carbon per admitted unit and the molecular-formula balances under phosphorylation

and oxalate reduction; the cycle carbon ledger and the lipid coproduct; and the registry exponents.

Encoding

Rationals by cross-multiplication. Article 79's finite claims are exact rationals, and `Nat` has no rationals. A claim $a_1/b_1 = a_2/b_2$ is therefore certified in the form $a_1 b_2 = a_2 b_1$, which is equivalent to it and stays inside the natural numbers. The design equation of Theorem 7.3 is the leading example: $p_2/p_4 = R_D/R_T$ becomes

$$81\,450\,000\,000 \times 208 \;=\; 400\,000\,000 \times 42\,354,$$

both sides being 1.6927×10^{13}, checked by `decide`. Reduced to lowest terms the same claim reads $81\,450\,000\,000 \times 8 = 400\,000\,000 \times 1629$.

Distributions by a common denominator. The five exit probabilities are exact rationals over the common denominator $D = 81\,980\,087\,189$. Normalisation is then the `Nat` identity

$$8\,197\,189 + 81\,890\,000 + 81\,450\,000\,000 + 40\,000\,000 + 400\,000\,000 \;=\; D,$$

which is exact, not a tolerance.

Rounded figures by a rounding bound. Where the article quotes a rounded decimal, the certificate is the statement that the rounding is correct, not that some float agrees. The founder ratio $R_D/R_T = 42\,354/208$ is quoted as 204, and the certificate is

$$2 \left| 42\,354 - 204 \times 208 \right| \;=\; 156 \;<\; 208,$$

which says exactly that 204 is the nearest integer. The companion bracket theorem adds $203 \times 208 < 42\,354 < 204 \times 208$.

The arm ladder. Rate constants are scaled by 100 so that the ladder $e_n = e_0 q^n$ with $e_2 = 1$ is integral: $e = (1, 10, 100, 1000, 10\,000)$. Theorem 5.1's conclusion is then the `Nat` chain $1 < 10 < 100 < 1000 < 10\,000$, and the hypothesis H_{add} is the four equalities $e_{n+1} = e_n q$.

What is not encoded. Three blocks of the article are not finite and do not appear in the Lean file: the transcendental root x^* of $x = 2(1 - e^{-x})$, the Perron limit of the column transfer matrix, and the registry exponentials $e^{-2\nu\Delta}$. These are covered by the Python harness and labelled FLOATING there. Nothing in the Lean file depends on them.

Theorem Index

Section	Name	Statement (informal)
S3	`R_T_val`	$R_T = L + n_{\mathrm{ICL}} = 208$
	`demand_ratio`	$R_D \cdot 8 = R_T \cdot 1629$
	`demand_gcd`	$\gcd(R_D, R_T) = 26$
	`rho_founder_rounds_204`	$2\lvert R_D - 204R_T\rvert < R_T$
	`rho_founder_bracket`	$203R_T < R_D < 204R_T$
	`rho_two_rounds_94`	ratio at $\alpha = 2/100$ rounds to 94
S4	`ladder_geometric`	$e_{n+1} = e_n q$ for $n < 4$
	`q_exceeds_one`	$1 < q$
	`ordering`	**Thm 5.1**: $P_0 < P_1 < P_2 < P_3 < P_4$
	`ladder_span`	$e_4 = e_0 q^4$
S5	`exit_sums_to_one`	numerators sum to D
	`design_equation`	**Thm 7.3**: $p_2 R_T = p_4 R_D$
	`design_ratio_reduced`	$p_2 \cdot 8 = p_4 \cdot 1629$
	`p3_is_p4_over_q`	$p_3 q = p_4$
S6	`eta_is_p2_plus_p4`	$\eta = p_2 + p_4$
	`eta_meets_bound`	$\eta \geq$ the stated lower bound
	`eta_below_one`	$\eta < 1$
	`bound_below_one`	bound < 1
	`eta_above_998`	$\eta > 0.998$
S7	`coop_floor`	$203\,625/11 = 18\,511$
	`coop_not_exact`	$203\,625 \bmod 11 = 4$
	`coop_bracket`	$18511 \cdot 11 < 203625 < 18512 \cdot 11$
S8	`initiator_surplus_exceeds_demand`	$c_I \cdot 5 < 104$
	`initiator_surplus_twentyfold`	$20 \cdot 5 \leq 104$
S9	`window_threshold`	$\lfloor 81/10 \rfloor = 8$
	`window_threshold_exact`	$10 \cdot 8 + 1 = 81$
	`completion_threshold`	$q \cdot 9 = 90$
	`completion_dominates_window`	$81 < 90 \cdot 10$
S10	`orbits_two`	$(3, 8, 10)$ at $m = 2$
	`orbits_three`	$(6, 27, 45)$ at $m = 3$
	`types_two_total`	$3 + 8 + 10 = 21$
	`types_three_total`	$6 + 27 + 45 = 78$
	`orbT_burnside_two`	$2T(2) = 2^4 + 2^2$
	`orbT_burnside_three`	$2T(3) = 3^4 + 3^2$
S11	`carbon_parent`	$C(0) = 18$
	`carbon_D`, `carbon_I`, `carbon_T`	22, 24, 26
	`carbon_strict_mono`	C strictly increasing in n
	`formula_T_e`	$C_{26}H_{28}O_8$
	`phosphorylation_adds_no_carbon`	C unchanged
	`phosphorylation_balance`	$+H_1O_3P_1$ per arm

Section	Name	Statement (informal)
	`formula_D_oxalate`	$C_{22}H_{12}O_8$
S12	`carbon_cycle_val`	$22R_D + 26R_T = 937\,196$
	`lipids_val`	$\nu L/2 + \ell_{\mathrm{ICL}} = 6\,677$
	`coproduct_val`	$4 \times 6677 = 26\,708$
	`lipid_halving_exact`	νL even: no truncation
	`coproduct_small`	$30\times$ coproduct $<$ cycle
	`initiator_carbon`	$c_I \cdot 24 = 24$
S13	`registry_nu_values`	$\nu_D = 2$, $\nu_T = 4$
	`registry_T_double`	$\nu_T = 2\nu_D$
	`registry_two_contacts`	column exponents 4 and 8
S14	`demand_dominates_pore`	$100R_T < R_D$
	`demand_within_three_orders`	$R_D < 1000R_T$
	`icl_contribution_small`	ICL bilayer is a small share

Proof Listing: `acf-79-proof.lean`

`acf-79-proof.lean` — complete source

```
/-
  acf-79-proof.lean
  Annals of the Chrysene Formalism, Vol. 1, Issue 5.
  Article 79 -- "The Source: Substitution-Initiated Self-Separation of the
  Substituted-Chrysene Stock from a Polyaromatic Bulk, and the Supply It
  Delivers to the Protocell"

  Lean 4 formal proof file.  Core only: no Mathlib, no `sorry`, no
  `native_decide`.  Every theorem is discharged by `decide`: the goal is a
  decidable proposition over Nat, and the kernel evaluates a verified
  Boolean decision procedure and confirms `true`.

  ENCODING.  Article 79's finite claims are exact rationals.  A rational
  a/b is certified here by cross-multiplication over Nat, which is the
  faithful encoding: `a₁ * b₂ = a₂ * b₁` says exactly `a₁/b₁ = a₂/b₂`
  without leaving the natural numbers.  Where the paper quotes a rounded
  decimal, the certificate is the rounding bound `2*|n - r*d| < d`, which
  says exactly that `n/d` rounds to `r`.  No floating point appears.

  Build:  lean acf-79-proof.lean
-/

namespace ChryseneFormalism.Article79

/-! ## S2.  Scaled corpus constants

    Rate constants are scaled by 100 so that the arm ladder
    e n = e₀ qⁿ with e₂ = 1 is integral: e = (1, 10, 100, 1000, 10000)
    in units of 10⁻² .  Lengths are in bases, counts in units. -/

def L            : Nat := 200          -- genome length, bases
def nICL         : Nat := 8            -- pore T₀ units, once per founder
def ellICL       : Nat := 77           -- bilayer lipids of the minimal ICL
def nu           : Nat := 66           -- lipids per base (Article 65)
def cI           : Nat := 1            -- initiators drawn per strand-period

/-- Arm multiplier q of H_add: each polar arm multiplies the export rate
    by q.  Article 79 fixes q = 10. -/
def q            : Nat := 10

/-- The export ladder e n = e₀ qⁿ, scaled by 100 so e₂ = 100 ↔ 1. -/
def e : Nat → Nat
  | 0 => 1
  | 1 => 10
  | 2 => 100
  | 3 => 1000
  | _ => 10000

/-- Imported demand, Article 65 Thm 8.5 plus the Article 78 ICL bilayer,
    at bud fraction α = 1/10. -/
def R_D          : Nat := 42354
/-- Pore demand, drawn once per founder: L + nICL. -/
def R_T          : Nat := 208
```

```
/-! ## S3.  Imported demand -/

theorem R_T_val : R_T = L + nICL := by decide

/-- The demand ratio in lowest terms: R_D / R_T = 1629 / 8. -/
theorem demand_ratio : R_D * 8 = R_T * 1629 := by decide

theorem demand_gcd : Nat.gcd R_D R_T = 26 := by decide

/-- R_D / R_T rounds to 204: twice the distance to 204 is below the
    denominator, which is exactly the statement that 204 is the nearest
    integer. -/
theorem rho_founder_rounds_204 : 2 * (204 * R_T - R_D) < R_T := by decide

/-- and 204 is an over-estimate, 203 an under-estimate: the true ratio
    lies strictly between. -/
theorem rho_founder_bracket : 203 * R_T < R_D ∧ R_D < 204 * R_T := by decide

/-- At the contrasting bud fraction α = 2/100 the demand is 136478/7, so
    the ratio to R_T is 136478 / 1456, which rounds to 94. -/
theorem rho_two_rounds_94 : 2 * (94 * 1456 - 136478) < 1456 := by decide

/-! ## S4.  The arm ladder and the ordering theorem

    Article 79 Hypothesis H_add gives P n = P₀ qⁿ with q > 1; Theorem 5.1
    (Ordering) concludes P₀ < P₁ < P₂ < P₃ < P₄.  On the scaled ladder the
    conclusion is a chain of Nat inequalities. -/

theorem ladder_geometric :
    e 1 = e 0 * q ∧ e 2 = e 1 * q ∧ e 3 = e 2 * q ∧ e 4 = e 3 * q := by
      decide

theorem q_exceeds_one : 1 < q := by decide

/-- Theorem 5.1, Ordering, on the scaled ladder. -/
theorem ordering : e 0 < e 1 ∧ e 1 < e 2 ∧ e 2 < e 3 ∧ e 3 < e 4 := by
  decide

/-- The ladder spans four decades: P₄ / P₀ = q⁴. -/
theorem ladder_span : e 4 = e 0 * q * q * q * q := by decide

/-! ## S5.  The exit distribution under the adopted kinetics

    With k₁ = k₂ = k₄ = 100 (scaled) and k₃ set by the design equation,
    the exit probabilities are exact rationals over the common
    denominator D below.  They are certified by their numerators. -/

def D_exit  : Nat := 81980087189
def p0_num  : Nat := 8197189
def p1_num  : Nat := 81890000
def p2_num  : Nat := 81450000000
def p3_num  : Nat := 40000000
def p4_num  : Nat := 400000000

/-- The exit distribution is a probability distribution: the numerators
    sum to the common denominator. -/
theorem exit_sums_to_one :
```

```
    p0_num + p1_num + p2_num + p3_num + p4_num = D_exit := by decide

/-- The design equation of Theorem 7.3: the disubstituted and
    tetrasubstituted exits stand in exactly the ratio of their demands,
    p₂ / p₄ = R_D / R_T.  Certified by cross-multiplication. -/
theorem design_equation : p2_num * R_T = p4_num * R_D := by decide

/-- Equivalently, in lowest terms p₂ / p₄ = 1629 / 8. -/
theorem design_ratio_reduced : p2_num * 8 = p4_num * 1629 := by decide

/-- The trisubstituted channel is a tenth of the tetrasubstituted one:
    p₃ / p₄ = 1/10 = 1/q. -/
theorem p3_is_p4_over_q : p3_num * q = p4_num := by decide

/-! ## S6.  Matched efficiency and its lower bound

    η = p₂ + p₄ = 81850000000 / D.  Article 79 bounds it below by
    1 - e₀/k₁ - e₁/k₂ - (R_T/(R_D+R_T))(e₃/k₄) = 16343993 / 16370000. -/

def eta_num   : Nat := 81850000000
def bound_num : Nat := 16343993
def bound_den : Nat := 16370000

theorem eta_is_p2_plus_p4 : eta_num = p2_num + p4_num := by decide

/-- η ≥ the stated lower bound, by cross-multiplication. -/
theorem eta_meets_bound : bound_num * D_exit ≤ eta_num * bound_den := by
    decide

/-- Both lie strictly below 1: the source is not perfectly efficient. -/
theorem eta_below_one : eta_num < D_exit := by decide

theorem bound_below_one : bound_num < bound_den := by decide

/-- η exceeds 0.998: scaled, 1000 * eta_num > 998 * D. -/
theorem eta_above_998 : 998 * D_exit < 1000 * eta_num := by decide

/-! ## S7.  The cooperativity trade-off

    k₄ / k₃ = 203625 / 11.  The paper quotes 18 511. -/

theorem coop_floor : 203625 / 11 = 18511 := by decide

theorem coop_not_exact : 203625 % 11 = 4 := by decide

theorem coop_bracket : 18511 * 11 < 203625 ∧ 203625 < 18512 * 11 := by
    decide

/-! ## S8.  The initiator surplus

    Theorem 7.3 requires that the initiator actually drawn, c_I, not
    exceed the initiator made per period, Φ p₃ = 104/5 = 20.8. -/

def Isurplus_num : Nat := 104
def Isurplus_den : Nat := 5

theorem initiator_surplus_exceeds_demand :
```

```
    cI * Isurplus_den < Isurplus_num := by decide

/-- The surplus is twentyfold and more. -/
theorem initiator_surplus_twentyfold :
    20 * Isurplus_den ≤ Isurplus_num := by decide

/-! ## S9.  The contrast case: window and completion thresholds

    At tolerance ε = 1/10 the axis window threshold is
    (1-ε)² / (q ε²) = 81/10 and the completion threshold is
    q (1-ε) / ε = 90. -/

theorem window_threshold : 81 / 10 = 8 := by decide

theorem window_threshold_exact : 10 * 8 + 1 = 81 := by decide

theorem completion_threshold : q * 9 = 90 := by decide

/-- The completion threshold exceeds the window threshold by an order of
    magnitude: 90 * 10 > 81. -/
theorem completion_dominates_window : 81 < 90 * 10 := by decide

/-! ## S10.  Arm words within Article 77's families

    Orbit counts under the inversion ι (3↔9, 6↔12):
      D on {6,12}      : m(m+1)/2
      I on {3,6,12}    : m³          (ι acts trivially on the family)
      T on {3,6,9,12}  : (m⁴+m²)/2
    for an alphabet of m arm types. -/

def orbD (m : Nat) : Nat := m * (m + 1) / 2
def orbI (m : Nat) : Nat := m * m * m
def orbT (m : Nat) : Nat := (m*m*m*m + m*m) / 2

theorem orbits_two : orbD 2 = 3 ∧ orbI 2 = 8 ∧ orbT 2 = 10 := by decide
theorem orbits_three : orbD 3 = 6 ∧ orbI 3 = 27 ∧ orbT 3 = 45 := by decide

theorem types_two_total : orbD 2 + orbI 2 + orbT 2 = 21 := by decide
theorem types_three_total : orbD 3 + orbI 3 + orbT 3 = 78 := by decide

/-- The Burnside halving is exact: the tetrasubstituted count is the
    average of all words and the ι-fixed words. -/
theorem orbT_burnside_two : 2 * orbT 2 = 2*2*2*2 + 2*2 := by decide
theorem orbT_burnside_three : 2 * orbT 3 = 3*3*3*3 + 3*3 := by decide

/-! ## S11.  Carbon per admitted unit and molecular formulae

    Article 77 Prop 3.2: a unit carrying n two-carbon arms has
    C = 18 + 2n.  The parent chrysene is C₁₈H₁₂. -/

def carbon (n : Nat) : Nat := 18 + 2 * n

theorem carbon_parent : carbon 0 = 18 := by decide
theorem carbon_D : carbon 2 = 22 := by decide
theorem carbon_I : carbon 3 = 24 := by decide
theorem carbon_T : carbon 4 = 26 := by decide
```

```
    /-- The carbon count is strictly monotone in arm number, so arm count is
        recoverable from carbon count. -/
    theorem carbon_strict_mono :
        carbon 0 < carbon 1 ∧ carbon 1 < carbon 2 ∧
226     carbon 2 < carbon 3 ∧ carbon 3 < carbon 4 := by decide

    /-- Formula of the tetra-(hydroxyethoxy) unit: C₂₆H₂₈O₈. -/
    theorem formula_T_e : carbon 4 = 26 ∧ 12 + 4*4 = 28 ∧ 2*4 = 8 := by decide

231 /-- Phosphorylation of an e arm adds exactly H₁O₃P₁ per arm and no
        carbon, so the tetra-phosphate is C₂₆H₃₂O₂₀P₄. -/
    theorem phosphorylation_adds_no_carbon : carbon 4 = 26 := by decide
    theorem phosphorylation_balance :
        28 + 4 = 32 ∧ 8 + 3*4 = 20 := by decide
236
    /-- The 6,12-bis(oxalate) acid: C₂₂H₁₂O₈.  An oxalate arm is net C₂O₄,
        so two arms add four carbons to C₁₈. -/
    theorem formula_D_oxalate :
        18 + 2*2 = 22 ∧ 10 + 2*1 = 12 ∧ 4*2 = 8 := by decide
241
    /-! ## S12.  The carbon ledger of one cycle

        Carbon in the D and T demand is 22 R_D + 26 R_T.
        The route-A lipid coproduct is four carbons per lipid, with
246     ν L / 2 + ellICL lipids. -/

    def carbon_cycle : Nat := 22 * R_D + 26 * R_T
    def lipids       : Nat := nu * L / 2 + ellICL
    def coproduct    : Nat := 4 * lipids
251
    theorem carbon_cycle_val : carbon_cycle = 937196 := by decide
    theorem lipids_val : lipids = 6677 := by decide
    theorem coproduct_val : coproduct = 26708 := by decide

256 /-- The lipid count halves exactly: ν L is even, so no unit is lost to
        truncation in ν L / 2. -/
    theorem lipid_halving_exact : nu * L % 2 = 0 := by decide

    /-- The coproduct carbon is a small fraction of the cycle carbon: more
261     than thirtyfold smaller. -/
    theorem coproduct_small : 30 * coproduct < carbon_cycle := by decide

    /-- The initiator carbon actually drawn is 24 per initiator, and the
        surplus made exceeds it twentyfold. -/
266 theorem initiator_carbon : cI * 24 = 24 := by decide

    /-! ## S13.  Column registry

        A guest that cannot present the arm pattern breaks ν registrations at
271     each of two contacts.  The disubstituted stream carries ν = 2, the
        tetrasubstituted ν = 4. -/

    def regNuD : Nat := 2
    def regNuT : Nat := 4
276
    theorem registry_nu_values : regNuD = 2 ∧ regNuT = 4 := by decide
```

```
    /-- The tetrasubstituted stream is protected by twice the exponent, so at
        equal Δ the suppression is the square. -/
281 theorem registry_T_double : regNuT = 2 * regNuD := by decide

    /-- Each contact pair doubles the exponent again: the column exponent is
        2ν. -/
    theorem registry_two_contacts :
286     2 * regNuD = 4 ∧ 2 * regNuT = 8 := by decide

    /-! ## S14.  Consistency of the demand with the genome

        R_T = L + nICL is the founder pore draw; the per-period D demand
291     exceeds it by two orders of magnitude. -/

    theorem demand_dominates_pore : 100 * R_T < R_D := by decide

    theorem demand_within_three_orders : R_D < 1000 * R_T := by decide
296
    /-- The ICL bilayer contribution 2*ellICL is small against the total. -/
    theorem icl_contribution_small : 200 * (2 * ellICL) < R_D * 2 := by decide

    end ChryseneFormalism.Article79
```

Kernel Axiom Audit

Axiom Inventory

Running `#print axioms` on each of the 55 theorems in `acf-79-proof.lean` yields:

Result	Meaning	Count
no axioms	The proof term reduces to primitive inference steps alone. Nothing is assumed beyond the type theory itself.	54
`propext`	Propositional extensionality, inherited from `Nat.gcd` in the Lean core by `demand_gcd`.	1
`Quot.sound`	Not used.	0
`funext`	Not used.	0
`Classical.choice`	Not used.	0
Extra axioms introduced by this proof:		**none**

Verification Note

The proof file makes no use of:

- `native_decide`, which bypasses the kernel by running compiled code;
- `sorry`, the placeholder for a missing proof;
- `Classical.choice` or `Classical.em`, neither of which is required for decidable `Nat` arithmetic;
- any Mathlib import.

The `decide` tactic in Lean 4 core reduces a goal to a kernel-evaluated Boolean expression; the kernel itself, and not an external plugin, performs the evaluation. This is a strictly stronger guarantee than `native_decide`, which trusts compiled Lean code outside the kernel.

To reproduce: install Lean 4.12.0 and run `lean acf-79-proof.lean`. A silent run with exit code 0 is the certificate; any failure is reported on standard error. The file has no dependencies beyond the Lean core, so no `lakefile` and no package fetch are required.

Scope and Honest Limits

A kernel-checked theorem certifies that a conclusion follows from the definitions written in the file. It does not certify that those definitions model the chemistry. In particular the scaled constants of section S2 are the article's *stipulated* nondimensional illustrations: the Lean file proves that the article's numbers follow from the article's rates, and is silent on whether those rates obtain. The regioselectivity premise of the bromination sequence, which Article 79 carries openly as an unmet obligation, is outside the reach of any arithmetic proof and is not addressed here.