e Null-Prefix Problem

A Formal Open Question in Computability Theory

Version 1.4 Changes from v1.3

(1) T-BUF (Buffer Overflow Theorem) added to Section V. P_0 is formally identified as the buffer-full condition: $K(x|n)/n = 1$ means no shorter representation of the null state exists. The contained program (L-CONTAIN) can no longer be represented in compressed form. Its execution in full is the only remaining representation. That execution is the Snap. The cause and the event are the same thing. Step 5 is now a proof target, not an open gap.

(2) R-DL (Density-at-Lower-Limit remark) added. P_0 applies specifically to the $\perp \rightarrow \varepsilon_0$ boundary. It is a local density condition at the null-to-existence transition, not a global property of zero's entire structure. This sharpens T-BUF significantly.

(3) AX-1 status updated from NEAR-DERIVABLE to DERIVABLE PENDING T-BUF. The derivation chain is complete in sketch form. T-BUF is the proof target.

I. AX-1 — Current Status

Axiom AX-1 — Binary Snap Causality | Status: DERIVABLE PENDING T-BUF [changed from NEAR-DERIVABLE in v1.4]

When the configuration string x of the Null State reaches the incompressibility threshold P_0, the Null State undergoes the Binary Snap.

Derivation status: The complete derivation chain is now established in sketch form. T-BUF is the proof target. Once T-BUF is formally proven in ZP-C v1.4, AX-1 becomes a derived theorem and ceases to be an axiom.

Informal restatement (T-BUF model): P_0 is not a trigger. P_0 is the buffer filling. The Snap is the flush. Before P_0, a shorter representation of zero exists — execution is deferred. At P_0, no shorter representation exists — the program must execute in full. The Snap is zero's contained program being forced to run.

II. Definitions (All Unchanged)

D-TQ-1 — Self-Description Program

$p \in SD(\varepsilon)$ iff $U(p) = \varepsilon$.

D-TQ-2 — Partial Output

$out_t(p)$ = output tape content by step t.

D-TQ-3 — Null-Throughout

$\forall$ t: $out_t(p) = \varepsilon$.

CC-TQ — Computability Constraint

$CV(\varepsilon) = \{p \in SD(\varepsilon) : \exists\, t < T(p),\ c_t \neq c_0\}$.

L-RUN — Running is State Change

$c_0 \to c_1$ is non-null state change by AX-B1. Near-complete pending DA-1.

L-CONTAIN — Zero Contains Every Executor

$K(p|0)=0$ for all $p \in SD(\varepsilon)$. Candidate pending DA-L.

R-HW — Hardware Inversion

Software already contained in hardware. Zero does not wait for its executor.

III. The Question (Unchanged)

TQ-IH — The Null-Prefix Problem | Status: OPEN

$NT(\varepsilon) \cap CV(\varepsilon) = $ ▯ ?

IV. T-BUF — The Buffer Overflow Theorem

This is the formal statement that closes Step 5 in sketch form. P_0 is not an external trigger. It is the buffer-full condition. The Snap is the flush.

Theorem T-BUF — Buffer Overflow at P_0 | Status: PROOF TARGET — sketch complete, formal proof required in ZP-C v1.4 [NEW v1.4]

Claim: At P_0, the already-contained self-description program p ∈ CV(ε) (from L-CONTAIN) cannot be represented in compressed form. Its execution in full is the only remaining representation of zero's state. That execution is the Snap.

Argument:

Step A — Before P_0: $K(x|n)/n < 1$. A program shorter than x exists that produces the null state. The system can represent zero’s state without running the full self-description program p. The buffer has room. Execution is deferred.

Step B — At P_0: $K(x|n)/n = 1$. No program shorter than x itself produces the null state. The compressed representation is exhausted. The buffer is full. The only remaining representation of zero’s state is the execution of p itself. The buffer must flush.

Step C — The flush is the Snap: The execution of p (L-CONTAIN: already contained) produces a non-null intermediate state (L-RUN + DA-1: running is state change by AX-B1). That non-null intermediate state is the First Atomic State. The Snap is not triggered by P_0 — the Snap IS the flush at P_0. Cause and event are the same.

Formal proof requirement: Prove that $K(x|n)/n = 1$ implies no compressed proxy for p remains — that is, the incompressibility of x at P_0 entails that p must execute in full rather than being represented by a shorter program. This connects ZP-C D1 to L-CONTAIN via the Kolmogorov coding theorem (I-KC).

4.1 Remark R-DL — Density at the Lower Limit

Remark R-DL — P_0 is Local to the $\perp \rightarrow \varepsilon_0$ Boundary | Status: Remark [NEW v1.4]

P_0 as the Snap trigger is not a global property of zero's entire structure. It is specifically the density condition at the $\perp \rightarrow \varepsilon_0$ boundary — the transition from the lower limit of the lattice to the first element above it.

Higher state transitions are governed by ZP-E T5 (Iterative Forcing Theorem: $\alpha_n = \varepsilon(S_n)$ at each step). P_0 does not need to be reached everywhere in zero's structure simultaneously. It needs to be reached only at this specific boundary.

Consequence for T-BUF: the buffer whose overflow constitutes the Snap is the buffer at the $\perp \rightarrow \varepsilon_0$ boundary specifically. The flush is local. This makes the formal proof of T-BUF tractable: it only needs to hold at the lower limit, not universally.

V. Complete Derivation Chain

Step	Claim	Status
1	Zero is internally infinite: $v_2(0)=+\infty$, $I\to\infty$, ZP-G T6	Established
2	SD(ε) is infinite: incompressibility, ZP-C D1	Established
3	L-CONTAIN: $K(p\|0)=0$ for all $p \in SD(\varepsilon)$	Candidate — DA-L needed
4a	CC-TQ: valid self-descriptions must be actual computations	Established
4b	L-RUN: $c_0\to c_1$ is non-null state change by AX-B1	Near-complete — DA-1
4c	DA-1: configuration change entails non-null output-tape event	Open — expected straightforward
5	T-BUF: P_0 = buffer full = forced flush = the Snap	PROOF TARGET — sketch complete
5a	R-DL: P_0 applies locally at $\bot\to\varepsilon_0$ boundary	Remark — sharpens T-BUF
6	AX-1 derived	Follows from Steps 1–5

VI. Open Items Register v1.4

Item	Status	Description
AX-1	DERIVABLE PENDING T-BUF	No longer an axiom in substance. Proof target is T-BUF in ZP-C v1.4.
T-BUF	PROOF TARGET — sketch complete	Prove $K(x\|n)/n=1$ at $\bot\to\varepsilon_0$ forces execution of contained program. Key: connect incompressibility (ZP-C D1) to L-CONTAIN via I-KC coding theorem.
DA-L	Open — expected straightforward	Formally connect $K(p\|0)=0$ to ZP-C D1 and I-KC.
DA-1	Open — expected straightforward	Configuration change at t=1 entails $out_1(p)\neq\varepsilon$ per D-TQ-2.
L-CONTAIN	Candidate	$K(p\|0)=0$ for all $p \in SD(\varepsilon)$. Needs DA-L.
TQ-IH	OPEN	NT(ε) ∩ CV(ε) = ▯? Follows from T-BUF + L-RUN.
BT-M / AX-⊥-M	Conditional on T-BUF	If T-BUF: AX-⊥-M derivable as corollary. No axiom needed.

ZP-C v1.4	Next document	Formalize L-CONTAIN, DA-L, T-BUF, R-DL. Derive AX-1 as theorem.
M2/M3 preference	CLOSED — CC-TQ	v1.1.