Equal degree and type do not determine a code-loop representation

Myeongjin Kang

Abstract

Two actual rank-seven full-support reduced doubly-even binary subspaces on Fin50, each with sorted type fifteen ones/seven fives, represent the SAME explicitly nonassociative Moufang loop by genuine source-sign multiplicative bijections, but no arbitrary Fin50 coordinate permutation carries one code to the other. Refutes the broad printed2019Conjecture1, with all factor, weight, quotient-type, sign, rank and transport bridges included.

AI-assisted research preprint. The complete accepted technical proof follows. Independent internal AI reviews are not external journal peer review. Exact mathematical hypotheses, formal coverage, source-version boundaries, trust limits, and bounded prior-art qualifications remain controlling. No worldwide-priority certification is asserted.

mathdb:2611.00114v1 17 Nov 2026

Equal degree and type do not determine a code-loop representation

MathDB 343790. A degree-50, rank-seven counterexample to the broad printed 2019 conjecture. This paper completes the documentary details supplied and checked in the independent GPT-6.1 Sol review. The fifteen immutable Lean production modules are unchanged. Campaign accounting and the bounded scientific/process qualifications are recorded separately by root; this paper awards no credit by itself.

mathdb:2611.00114v1 17 Nov 2026

Statement and source scope

Pires, Grishkov and Rasskazova [1], Conjecture 1, assert that two representations of one code loop with equal degree and type are isomorphic codes. A representation is a doubly even binary subspace with a factor set whose associated multiplication gives that loop. Coordinate isomorphism means an arbitrary coordinate permutation carrying the entire code to the other. Coordinate classes consist of positions agreeing on EVERY codeword; a representation is reduced when every such class has size less than eight. Its type is the sorted list of class sizes. Rank means vector-space dimension.

We give two full-support, reduced, doubly even subspaces of F2^50, each of rank seven and type fifteen ones followed by seven fives. They represent the SAME explicitly nonassociative Moufang loop, but no permutation of the fifty positions carries one code to the other. Thus the printed universal assertion is false. Minimality is a separate source definition. No minimal-only, center-order-two or rank-three/four uniqueness claim is refuted here. No new underlying Moufang loop is claimed; the result concerns simultaneous constraints on representations.

The two codes and their actual type

Write W=F2^4 and X=F2^3, with coordinates numbered from zero. Let H1 be the nonzero part of a3=0. Let H2 be the union of the affine plane a3=1,a2=0 and the nonzero part of the plane a1=a3=0. Each has seven nonzero vectors and vector sum zero. For H1 this follows by counting each free coordinate four times. For H2, the four affine-plane vectors and three punctured-plane vectors each sum to zero; their last coordinates separate the two blocks. H2 contains the third and fourth standard unit vectors (one-based unit names), while differences of its affine-plane vectors supply the first two. It therefore spans W.

For H equal to H1 or H2, use every nonzero a in W as a column, five times if a is in H and once otherwise. This gives 43 positions. Add the seven nonzero columns b in X on disjoint positions. Define the evaluation word by the entries w dot a on the W positions and x dot b on the X positions. Let D0 use H1 and D1 use H2. Each code is the actual linear-map range, not a list of prescribed words or invariants. All unit columns occur, proving injectivity; hence each range has dimension 4+3=7. Every column is nonzero, so every position is used.

Positions with the same label agree on every word. Distinct W labels, or distinct X labels, are separated by a unit input. Labels on different summands are separated by an input supported on just one summand. Thus agreement on every codeword is EXACTLY equality of labels. Seven classes have size five; the remaining fifteen have size one. All sizes are below eight. The actual quotient classes therefore give the required reduced type. The formal quotient-to-label and fiber-to-copy equivalences prove this statement, and coordinate transport gives actual submodules D50(0), D50(1) of F2^(Fin 50).

Common loop and factor

All arithmetic in the following factor formulas is in F2. For x,y in X, set

f(x,y) = sum_(i=0..2) x_i y_i + x0 y1 + x0 y2 + x1 y2
+ x0 x1 y2 + x0 x2 y1 + x1 x2 y0.

The diagonal is

q(x) = 1 + (1+x0)(1+x1)(1+x2).

It is zero at zero and one at every nonzero x. The complete polynomial gives f(0,y)=f(x,0)=0 and f(x,x)=q(x). Its symmetric polarization is q(x+y)+q(x)+q(y). Each bilinear term has zero coboundary; the three cubic terms yield the six distinct-coordinate monomials, the alternating triple form. These statements hold for all inputs, not only a basis.

On F2 x W x X define

(s,w,x) * (t,v,y) = (s+t+f(x,y), w+v, x+y).

Zero is the identity. To solve a*b=c, the W and X components give b_W=c_W-a_W and b_X=c_X-a_X; then b_s=c_s-a_s-f(a_X,b_X). To solve b*a=c use the same vector components and b_s=c_s-a_s-f(b_X,a_X). These explicit formulas prove existence and uniqueness of both divisions.

Expanding the displayed factor gives

f(x,y)+f(z,x)+f(x+y,z+x)
= f(y,z)+f(y+z,x)+f(x,y+z+x).

This proves (ab)(ca)=a((bc)a); the other components are additive. Consequently the multiplication is Moufang. At the three unit simplex inputs, with zero W and sign components, the associator signs differ. The loop is therefore genuinely nonassociative. CodeLoopLoop proves the identity, unique divisions, this Moufang law and the actual nonassociative witness directly; no factor-set-to-Moufang theorem is assumed.

mathdb:2611.00114v1 17 Nov 2026

Factor identities and source representations

For nonzero w in W, translation by a unit column with functional value one exchanges the zero and one fibers. Exactly eight W columns evaluate to one. If h_H(w) counts the ones on H, the repeated-column word has weight 8+4h_H(w). The vector sum of H is zero, so h_H(w) is even. The zero input has weight zero, and every W word has weight divisible by eight. The X simplex word has weight zero or four by the corresponding fiber translation. Thus D0 and D1 are doubly even.

For arbitrary binary words on any finite coordinate set, the checked identities are

wt(u+v) + 2 pair(u,v) = wt(u) + wt(v),

wt(u+v+z) + wt(u) + wt(v) + wt(z)
= wt(u+v) + wt(u+z) + wt(v+z) + 4 triple(u,v,z).

Here pair and triple are actual simultaneous-one intersection cardinalities. Dividing under the established divisibility hypotheses and reducing modulo two gives the pair-half and triple parity identities. The W part contributes a multiple of eight, so wt(d_i(w,x))/4 modulo two is q(x). Polarizing q supplies precisely the symmetry and coboundary identities of f. All source factor laws consequently hold for every actual codeword, including zero, equal and dependent inputs.

The inverse evaluation linear equivalence defines the factor on the actual code. The maps (s,w,x) -> (s,d_i(w,x)) are bijections and preserve the displayed multiplication, so both codes represent this same nonassociative loop. The explicit map from F2 to the integer signs {1,-1} sends addition to multiplication and n modulo two to (-1)^n. CodeLoopSigns proves these identities, the four literal multiplicative factor laws and the source signed multiplication law. Thus the additive encoding is transported to the actual source convention, not substituted for it without proof.

The linear equivalences onto Fin 50 carry the submodules, all codewords, rank, support, weights and pair/triple counts. Quotient-class and position-fiber equivalences carry class sizes, reducedness and the sorted type. Factors and multiplicative representations are transported by the same inverse maps. No unproved source or finite-coordinate bridge remains in the certified chain.

Arbitrary permutation obstruction and exact endpoint

The fourth W unit vector annihilates H1, giving a word of weight eight in D0. Since H2 spans W, every nonzero functional has a positive even count on H2. Its W word therefore has weight at least sixteen. The X summand adds zero or four, so D1 has no weight-eight word. Every arbitrary coordinate permutation preserves Hamming weight. Hence no permutation carries the entire first code to the second. This is not merely an obstruction to generator, label-preserving or chosen-coordinate maps.

The formal universal Literal2019Uniqueness quantifies over arbitrary finite coordinate types, actual binary submodules, source factor sets, reduced classes and representations of one Moufang loop. Its conclusion is full coordinate equivalence. The theorem literal2019_refuted_on_fin50 negates it using the actual fifty-position codes. The separate theorem common_isMoufangLoop proves the common-loop Moufang assertion. The exported full50_counterexample packages the negation, common nonassociativity, ranks, doubly even/full support/reduced/type conditions, representations and arbitrary-permutation inequivalence. It does NOT repeat common_isMoufangLoop as an additional explicit conjunct; that theorem is separately proved and used by the universal-negation proof.

mathdb:2611.00114v1 17 Nov 2026

Bounded prior comparison and scope

The target is the broad printed Conjecture 1 in the 2019 body [1]. The 2026 revision [2] retains general definitions and minimal rank-three/four classification but omits the explicit conjecture. Omission is not an authenticated correction, retraction or earlier public refutation. The construction has a substantial central W factor and is nonminimal; the rigid minimal/center-two questions remain open to this result. These limits accompany the claim.

Vojtechovsky [3], Theorem 2.1 and Propositions 3.1-3.2, supplies doubly even realizations of prescribed cubic data by a block-append construction and derived-form congruences. That existence theorem does not guarantee two outputs with simultaneous degree 50, the same reduced type and support, and global permutation inequivalence. Fixed-code factor equivalence is not uniqueness of different codes. The present two column arrangements, simultaneous constraints and weight obstruction still require proof; the earlier theorem body reviewed does not provide a complete reduction to this target.

The actual 2026 attribution of the 2018 orbit classification concerns loops through characteristic vectors, not equal-type coordinate representations. The authenticated narrow rank-four thesis attribution is likewise not an equal-type representation-nonuniqueness claim. No concrete applicable earlier exact proof was found in the bounded primary and registry comparisons. This is not exhaustive worldwide novelty clearance. The 2018 original and older 1986/1990 bodies and thesis were not fully read; final publication/date of the independent realization author PDF was not authenticated. Those limits are retained rather than presented as source clearance.

mathdb:2611.00114v1 17 Nov 2026

Validation and evidence limits

All fifteen unchanged production modules passed the independent fresh source rebuild and module-name leanchecker under Lean 4.34.1, one worker, M3500 and warnings as errors. The build was observed empty before compiling; LEAN_PATH contains only the fresh build plus pinned upstream. Actual binary hashes match the recorded pins. Pinned upstream mathematical truth and prebuilt libraries are trusted, not re-proved or independently attested here.

The independent origin inventory has 429 root entries, representing 428 unique kernel declarations. Full transitive type AND available opaque-value traversal covers 13,203 constants. The complete serialization contains 25,386 type/value DAGs and 2,469,269 nodes. Structural validation checks node arities, unique contiguous IDs, resolved references, root reachability, no loose variables and complete declaration records. Extra generated IR-only names without kernel constants are explicitly recorded, not assigned fabricated proof bodies. The only axioms are propext, Classical.choice and Quot.sound.

Meaningful controls reject a deleted DAG root, an omitted full50 audit root, a false coordinate-permutation assertion and a forbidden existing constant introduced in audit tooling. All 619 frozen bindings remained unchanged. An incorrect initial module order and audit/validator API failures remain preserved, followed by diagnostic-driven corrections. Production source was never repaired in review.

The implementation and review disclosed helper/log transport reads outside the required wrapper. Their mathematical builds, checkers and audits used the fixed supplied runner, but no universal wrapped-reader or clean-whole-history claim is made. Historical modified GLM harness/retry logic was not used in the certified Sol execution chain. Resource observations are sampled, not continuous cgroup or aggregate concurrent CPU bounds. Root reporting changes only transcribe the independently checked equations, complete citations and correct packaging/count descriptions; all Lean production bytes remain unchanged. Acceptance/accounting is separate.

References

[1] Rosemary Miguel Pires, Alexandre Grishkov and Marina Rasskazova. Representations of code loops by binary codes. arXiv:1909.05143v1, 9 September 2019, Definitions 2-3 and Conjecture 1.

[2] The same authors. Representations of code loops by binary codes. arXiv:2601.02332v1, 5 January 2026, general definitions and reduced/minimal representation results. No correction/retraction is inferred from omission of the 2019 conjecture.

[3] Petr Vojtechovsky. Combinatorial Aspects of Code Loops. Theorem 2.1 and Propositions 3.1-3.2, actual author PDF inspected by the independent review; final publication/date not authenticated in the bounded comparison.

Certificate pins: Lean 4.34.1 commit5045d0056413266e57c625dcd7c365b10e377c52; mathlibd13f23b723b8a846827a245b89c10fc7d3f11612. Full independent review: code-loop-formal-independent-review-sol-r7/FINAL-REVIEW.md. Root acceptance record: code-loop-acceptance.json.

mathdb:2611.00114v1 17 Nov 2026