TensorCash AI Compute Derivatives: Native, Cash-Settled Exposure to the Compute Cycle

Imosuke Takakuni[1]

9 June 2026

ABSTRACT

TensorCash ties block production to verified language-model inference: the network's mining difficulty is, in effect, a live on-chain measure of how much AI compute the world is committing to the chain. This paper specifies **AI Compute Derivatives** — native, cash-settled contracts whose payoff is a deterministic function of that difficulty at a committed future block. They let a miner or AI provider hedge revenue when compute floods in, and let an external party take financial exposure to the AI-compute cycle without operating a single GPU. The instrument is a margined, **capped** contract-for-difference: each side posts initial margin into its own Taproot vault and can lose at most that margin. A symmetric, zero-cost-to-enter form gives **delta-one** (linear) exposure; an optional day-one premium turns it into a **covered call or put** on compute. Settlement is enforced entirely by consensus — two narrow opcodes, active from genesis — with no general-purpose virtual machine, no oracle, and no intermediary. Cooperative settlement is private: on chain it is indistinguishable from an ordinary payment.

TABLE OF CONTENTS

1. "Imosuke Takakuni" is a pseudonym. Contact: takakuni@tensorcash.org

I. Difficulty Is the AI-Compute Index

In a conventional proof-of-work chain, mining difficulty tracks how much otherwise-useless hashing secures the network. In TensorCash, the work

is

AI inference: miners run registered open-source models, and the same forward pass that answers a paying user also competes to extend the chain. Difficulty therefore rises when more — or more capable — AI compute joins the network, and eases when compute leaves. It is a continuously updated, tamper-resistant, publicly readable index of aggregate AI-compute supply and demand.

That index is economically meaningful to several parties:

- **Miners and AI providers** are structurally *short* the compute cycle: when difficulty rises, each unit of their hardware earns a smaller share of the block subsidy, compressing margins. They have a natural need to hedge.
- **Investors and funds** want exposure to the growth of AI compute, but buying and operating hardware is capital-intensive, illiquid, and operationally heavy. A financial instrument referencing difficulty offers that exposure synthetically.
- **Treasuries and counterparties** who price long-dated compute commitments want a transparent reference and a way to lock in cost.

AI Compute Derivatives turn that index into a position anyone can take, settled natively by the chain rather than by a centralised venue.

II. The Instrument

An AI Compute Derivative is a **margined, cash-settled contract-for-difference (CFD)** on network difficulty, fixed at a committed future block height `H` (the *fixing*). It is bilateral: a long and a short. Each side posts **initial margin (IM)** into a separate on-chain vault, and the realized payoff is bounded by that margin — the maximum loss is the IM, never more. There are no margin calls and no liquidation cascades: the contract is fully pre-funded at open.

i. Capped payoff as a fraction of margin

Each vault settles *only its own posted margin*, paid out as a dimensionless fraction of itself. Let the strike difficulty and the realized difficulty at `H` define a proportional move. A vault pays its counterparty only when its owner is on the losing side of that move:

- The **long** loses as difficulty falls below strike; the **short** loses as difficulty rises above strike.
- The losing fraction is `leverage × (proportional move)`, clamped to the range 0–100% of the vault's margin.
- The owner keeps whatever is not paid out.

Because everything is denominated as a fraction of the vault's own margin, leverage can be set independently on each leg, and the two legs need not even use the same collateral asset — there is no cross-asset price for consensus to know. The chain enforces only the fractions; the parties choose the leverage that makes the trade fair at the price they agreed off-chain.

ii. Delta-one and optional forms

- **Delta-one (swap / CFD).** Both sides post margin; entry is zero-cost. Payoff is (capped) linear in the difficulty move. This is the building block for going long or short the compute cycle, and for a miner offsetting revenue risk.
- **Option (covered call / put).** A non-refundable premium is paid on day one, inside the same atomic opening transaction. In the one-sided shape, only the *writer* funds a collateral vault and the *buyer* posts no settlement margin: the buyer's maximum loss is the premium, and the upside is up to the writer's full margin. This is a covered difficulty call (buyer profits if compute rises) or put (buyer profits if compute falls). The premium may be paid in any asset — it is not part of the settlement covenant.

The same primitive thus spans linear hedges and asymmetric, premium-priced optionality without any change to the consensus rules.

iii. Worked examples

Each side posts 10 units of margin at 10× leverage. "Move" is the proportional change in difficulty versus the strike at fixing.

Scenario	Difficulty move	Long ends with	Short ends with
Compute rises +5%	+5%	15	5
Compute rises +10%	+10%	20	0 (short fully used)
Compute rises +12%	+12% (clamped)	20	0
Flat or falling	≤ 0	10	10
Compute falls −5%	−5%	5	15

No party loses margin merely for being offline; margin is at risk only to the extent the realized move consumes it, and the loss is hard-capped at the posted amount.

III. On-Chain Architecture

i. Two self-settling vaults

Each side's margin lives in one **Taproot output**. The two vaults are independent: neither needs to know anything about the other, which is what preserves asymmetric margin, asymmetric leverage, and per-leg collateral. Each vault offers two ways to settle:

- **Cooperative (key path).** Both parties hold an aggregate signing key (MuSig2) and co-sign the outcome. This path can settle any mutually agreed result — including early close, mark-to-market, novation, or a fully bespoke non-linear payoff — and can merge both vaults into a single transaction. On chain it looks exactly like an ordinary single-signature Taproot spend: it reveals nothing. This is the expected, everyday path.
- **Unilateral (script path).** If cooperation fails, the contract's formula is enforced by a covenant leaf that anyone — either party, or a third-party keeper — can execute once the fixing is final. All economic terms are committed inside the leaf, so the only outcome it can produce is the correct one.

ii. The underlying, and why it is buried

The contract reads difficulty at a **committed ancestor height** `H`, never at the block that confirms the settlement transaction. Reading the confirming block would let the in-the-money party time settlement around a retargeting boundary; reading a fixed, already-buried ancestor makes the payoff deterministic the moment block `H` exists. A maturity-depth burial requirement, enforced both in relay policy and in the covenant's timelock, ensures the fixing is final before settlement can be mined.

iii. No virtual machine

Settlement is two narrow consensus opcodes — one that reads the difficulty target at a height, one that verifies the capped payoff and binds it to exact outputs — active from genesis. There is no general-purpose programmable layer, no contract storage, no loops, and no external oracle: the underlying is the chain's own header data. This is the same conservative posture as the rest of TensorCash's settlement layer, and it removes the class of bugs that has produced most high-profile DeFi losses.

iv. Settlement liveness

The winning party is incentivised to settle both vaults — recovering its own margin and claiming its winnings. The losing party need never come online and still receives `margin − loss` at its committed address. Because the unilateral path is covenant-only, a keeper can settle on either party's behalf; keepers can only effect the *correct* outcome, so there is no griefing surface beyond timing, which the timelock fixes.

IV. Privacy

The privacy outcome follows the settlement path. A cooperative keyspend is indistinguishable from any other Taproot payment, so a contract opened and closed cooperatively never discloses that a derivative existed. The unilateral fallback reveals only the single leg it settles, and even then only its committed terms — not the counterparty's leg. Taproot keeps the term sheet as a hash commitment until, and unless, the script path is used.

V. Collateral and Settlement Assets

In the first version, the **margin is settled in the native unit (TSC)**, because the unilateral covenant settles native outputs. The architecture already supports per-leg, non-native collateral; a later asset-aware settlement opcode lifts the native-only restriction without changing the two-vault design. The **day-one premium has no such restriction** even today: because it is an ordinary output in the opening transaction and not part of the settlement covenant, it can be paid in any asset — native or issued.

VI. Use Cases

- **Provider revenue hedge.** A miner expecting difficulty to climb shorts the compute cycle: if difficulty rises and operating margins compress, the short pays out, offsetting the revenue lost to competition.
- **Synthetic compute exposure.** A fund takes the long leg to gain capped, liquid exposure to AI-compute growth with no hardware, no cloud contracts, and no operational risk beyond the posted margin.
- **Asymmetric views.** A buyer pays a premium for a covered call on compute, capping downside at the premium while retaining upside to the writer's full collateral.
- **Treasury reference and cost-locking.** Counterparties pricing forward compute commitments use difficulty as a transparent, manipulation-resistant reference and lock in exposure bilaterally.

VII. What It Is and Is Not

AI Compute Derivatives are a **native protocol capability**: bilateral, fully collateralised, self-custodied contracts settled by consensus. There is no pool, no fund, no issuer, no managed product, and no counter-party desk — the chain only enforces the agreed formula against margin each party posted itself. This document describes a protocol mechanism for informational purposes; it is not an offer, solicitation, or recommendation of any financial product, and difficulty exposure is volatile and may result in the loss of posted margin.

VIII. Roadmap

- **v1.** Native-margin (TSC) settlement; delta-one and premium-priced option forms; cooperative private settlement plus unilateral, keeper-settleable fallback; difficulty read at a buried fixing height.
- **v2.** Non-native (issued-asset) margin settlement; a fixed native bounty for keepers; both legs settled unilaterally in a single transaction; bucketed / piecewise non-linear unilateral payoffs (the cooperative path already supports arbitrary non-linear settlement today).