StakeTime: Time–Lock Staking Multiplier with Epochal Vesting

Open Source Draft v0.2 – June 2025

Abstract

Stakers mint synthetic governance tokens by locking a base asset for a chosen period t (maximum four years). A multiplier curve $M(t)$ translates lock length into mint amount. All curve parameters are stored on-chain and may be tuned by governance without redeploying code. Rewards emitted every epoch enter a vesting escrow and may be cut—partially or fully—by a multisig or on-chain DAO if arbitration proves validator misconduct within a configurable look-back window. This paper provides the mathematical foundations, pallet interface, economic rationale and security proofs, presented in continuous prose rather than bullet lists for heightened readability.

Contents

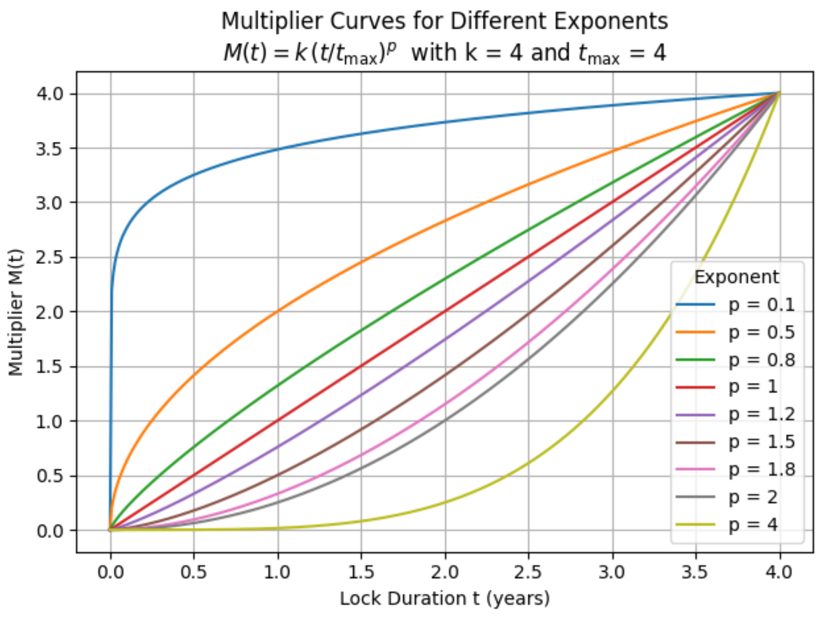

Figure 1: Staketime Multipliers

1 Core Symbols and Governance Parameters

Table 1 summarises all symbols exposed to governance. The multiplier's upper bound k is four in the canonical deployment, while the exponent p defaults to one. The epoch length ϵ is expressed in blocks; rewards vest linearly across ν epochs, and a look-back period λ allows retroactive slashing.

Symbol	Meaning
A	Tokens locked by a staker (base denomination)
t	Lock duration, measured in blocks or seconds ($0 < t \leq t_{\max}$)
$t_{\max}$	Maximum lock duration, fixed initially at four years
$M(t)$	Multiplier converting base tokens to synthetic supply
S	Synthetic tokens minted for a position; see Eq. (2)
k	Maximum possible multiplier (governance-controlled)
p	Curve exponent controlling convexity (governance-controlled)
ϵ	Epoch length in blocks (constant across runtime)
ν	Vesting span in epochs for newly emitted rewards
λ	Arbitration look-back window in epochs

Table 1: Configurable parameters manipulated by DAO or multisig.

The system employs a continuously differentiable curve

$$M(t) = k\left(\frac{t}{t_{\max}}\right)^p, \qquad 0 < t \leq t_{\max}, \tag{1}$$

where $k > 0$ provides the ceiling and $p > 0$ shapes concavity. Setting $p = 1$ yields a linear schedule, whereas values above unity strongly favour longer commitments; values below unity privilege short-term locks. Governance may modify k, p or even swap the functional form entirely by a runtime call `dao::set_curve`.

2 Minting and Lock Lifecycle

Upon calling `create_lock`, a participant transfers A tokens to an escrow vault. The runtime immediately computes S according to

$$S = A\,M(t), \tag{2}$$

and credits the account with S synthetic governance tokens. These synthetic units remain outstanding until unlock, at which point they are burned atomically with the return of the base deposit.

3 Epochal Emission and Vesting

At block heights divisible by the epoch length ϵ the chain mints a pool of new reward tokens, distributing them pro-rata to synthetic balances S. The payout does not settle immediately: instead each beneficiary receives a vesting stream spanning ν consecutive epochs. Thus only one-ν-th of the reward can be withdrawn after the first epoch, two-ν-ths after the second, and so forth until the amount is fully liquid. Maintaining a vesting buffer ensures that if arbitration later proves misconduct, yet-to-vest slices can be confiscated.

4 Retroactive Arbitration and Slashing on Pending Emissions

Validators supply attestations about off-chain activity performed by agents or application modules. Either the senate or a stake-weighted DAO can file an on-chain `slash_motion` referencing a specific module, the contested epoch index, and cryptographic evidence. Provided the motion is passed within the look-back window λ, the runtime automatically removes any unvested rewards associated with the disputed period and redirects them to a community treasury. Optionally the motion may impose an additional penalty on the modules's principal lock. This means that whatever is being locked by the module during registration as collateral can be potentially slashed. In either case, a second quorum needs to be resolved for the slash to finalize to add an extra buffer of precaution. We set this as a constant with respect to the vesting period. If the validator is flaged as acting maliciously during a previous arbitration, the locked stake it has as collateral is slashed. To avoid malicious slashing, the multisig will used to verify each slashing mechanism.

5 Parameter Changes via Governance and Staking DAO

Token holders may later submit proposals to change any of these values. If the proposal passes the prescribed quorum and threshold, the runtime executes an automated call that updates the parameter storage items; existing locks preserve their original multiplier, whereas new locks observe the revised curve. Emergency suspension of new locks is possible through a governance-controlled switch that disables the public extrinsic while leaving unlocks unaffected.

6 Module Futures

By reinterpreting the multiplier $M(t)$ as a *leverage function* $L(t)$, the StakeTime escrow can be repurposed into a cash-settled futures engine agnostic to the underlying asset. A trader opens a *futures lock* by depositing A units of the native token and specifying (i) a reference asset X (e.g. BTC, stETH, a commoditised index, or even an off-chain data feed), (ii) a bespoke vesting curve $V_X(t)$ that dictates how profit and loss (PnL) becomes withdrawable over time, and (iii) a leverage exponent p_L producing the exposure profile

$$L(t) = k_L \left(\frac{t}{t_{\max}} \right)^{p_L}, \qquad 0 < t \leq t_{\max}.$$

The runtime instantly mints synthetic long or short tokens whose notional value equals $A \cdot L(t)$. Mark-to-market gains stream to the holder according to $V_X(t)$, while unrealised losses are absorbed first by clawing back unvested rewards and, if necessary, by seizing the collateral principal. Because escrow, vesting, and PnL accounting live in the same pallet, the scheme is *chain-agnostic*: oracle relayers merely submit signed price updates, and wrapped versions of the synthetic futures tokens can circulate on any external DEX. Thus StakeTime extends beyond passive staking to a universal, permissionless derivatives layer where *any* asset—fungible or non-fungible—can be listed with tailor-made leverage curves, all secured by the native token's economic bandwidth and governed by the same DAO-controlled parameters that steer ordinary time-lock staking.

7 Security Considerations

Synthetic supply grants voting influence; therefore overly generous multipliers can threaten governance neutrality. The vest-then-slash construction mitigates short-term bribery attacks by ensuring that ill-gotten rewards remain claw-back-able for λ epochs. Fixed-point arithmetic employs 64-bit precision with saturation checks, preventing numeric overflow. To discourage griefing locks that bloat state, an exit before the agreed term burns not only the synthetic balance but also a configurable share of the underlying principal.

8 Conclusion

Time-lock staking combined with a parametrised multiplier curve aligns long-term commitment with token issuance. By draping every emission in a vesting veil that the DAO can slash retrospectively, the design reconciles capital efficiency with accountability. All numerical levers live in storage and may be tuned through standard governance motions, ensuring that economic policy evolves without disruptive code changes.