PATENT APPLICATION

INVENTOR(S)

[0001] KIMZEY, Samuel C

TITLE

[0002] Surplus Flow Redistribution System and Method

TECHNICAL FIELD

[0003] This invention relates to decentralized financial infrastructure and civic economic automation. More particularly, it provides a method for redistributing surplus monetary value across Vaults controlled by autonomous Agents, using anchored economic memory and contribution-aware weighting to prioritize structural equity without penalizing productive participants.

BACKGROUND

[0004] Conventional redistribution systems rely on taxation, staking rewards, or universal flat income. These models often suffer from perverse incentives, rewarding inactivity, penalizing high output, or extracting capital through platform rent. A need exists for a redistribution model that stabilizes the economic middle, supports the lower tier, and remains fair to high contributors—without state-based enforcement or tokenized economics.

SUMMARY

[0005] The Surplus Flow Redistribution method comprises:

1. Recording each Agent's earned income and borrowing activity within a fixed epoch, using one-action-one-mint anchoring.

2. Calculating surplus as the unborrowed portion of earned income.

3. Pooling 50% of all eligible surplus into a shared redistribution pool.

4. Computing the statistical median of all Agent incomes for that epoch.

5. Applying a weighting function centered on the median to assign redistribution influence.

6. Capping redistributed flow for each Agent to their surplus or remaining borrow capacity, whichever is smaller.

7. Allocating pooled flow using a non-linear, middle-out weighting curve.

8. Anchoring all redistribution outputs as immutable mint events.

DETAILED DESCRIPTION

[0006] For each Agent, surplus is calculated as Surplus (Si) = Income (Ii) – Borrowed Amount (Bi)

[0007] The system aggregates all surplus values and contributes 50% of the total into a shared redistribution pool, P.

[0008] It calculates the statistical median income (M) across all Agents for the epoch.

[0009] Each Agent is assigned a weight (wi) based on how close their income is to the median. The weighting function includes:

a. A full reward zone where income is within 10% of the median.

b. Beyond that, the weight decays using a quadratic formula such as:
$wi = 1 / (1 + (Ii – M)^2)$

[00010] Each Agent’s return (Ri) is capped by the lesser of:

a. Their surplus (Si), and

b. Their remaining borrowing capacity (Bi).

[00011] The redistributed amount (Fi) assigned to each Agent is computed as a proportion of the weighted pool:

a. Fi = (wi × Ri) ÷ total weighted pool × P

NARRATIVE USE CASE

[00012] An individual owns a ZEAM-enabled autonomous delivery vehicle. Unlike gig platforms, which extract value through centralized control and pricing opacity, the vehicle acts as an Agent, earning income directly into a Vault. After each 24-hour epoch, the Vault records $50 of income and $30 borrowed. The $20 surplus is logged.

[00013] During the same epoch, 10,000 Agents contribute surplus system-wide. The system calculates a median income of $52 and redistributes 50% of pooled surplus using the middle-out model. Because the vehicle's Agent is near the median, it receives additional surplus Flow of $10, strengthening its future borrowing power and incentivizing sustained, civically-aligned participation.

IMPLEMENTATION NOTES

[00014] One-Action-One-Mint Enforcement:

a. All income, borrowing, surplus, and redistribution events are anchored using a one-action-one-mint protocol, ensuring traceable, atomic state transitions without side effects.

[00015] Vault-Native Execution:

a. All values (income, surplus, borrowing, redistribution) are computed and enforced within Agent Vaults. No external module state or off-chain storage is used.

[00016] Stateless Redistribution Logic:

a. Redistribution weights and caps are recalculated each epoch using protocol logic. No persistent global state is carried between epochs.

[00017] Epoch Cadence:

a. All Agents operate on synchronized epochs (e.g., every 24 hours). Epochs must be uniform system-wide to ensure accurate median calculations and fair curve application.

[00018] No Token Dependence:

a. Redistribution does not require tokens, staking, or external currency. Flow operates solely within internal system Vaults and anchored contribution memory.

[00019] Trait-Layer Oversight (Optional):

a. Observational Traits may interpret economic drift or hoarding behavior across epochs. They may surface reflections to shared memory layers, but do not alter enforcement.

[00020] Middle-Out Weight Curve:

a. The redistribution function is centered around the epoch's median income, with a flat reward zone (typically ±10%) and quadratic decay applied beyond that range.

[00021] Flow Cap Guarantee:

  a. Each Agent's redistributed Flow is strictly limited to their original surplus or available borrowing capacity, whichever is lower, preventing redistribution without contribution.

CLAIMS

1. A method for redistributing surplus monetary flow among economic agents in a decentralized system, the method comprising:
   a. calculating each Agent's income I_i during an epoch;
   b. determining a surplus S_i = I_i - B_i, where B_i is the amount borrowed;
   c. pooling 50% of system-wide surplus into a redistribution pool P;
   d. computing the median income M of all participating Agents;
   e. assigning a weight to each Agent based on their income's proximity to the median, where weights are equal within a full reward zone and decay quadratically outside it;
   f. calculating a capped surplus return R_i ≤ min(S_i, B_i);
   g. distributing the pooled surplus proportionally, using each Agent's weight and capped return to determine their share.
2. The method of claim 1, wherein Agents outside a predefined deviation threshold from the median receive exponentially reduced allocation via a quadratic or sigmoid decay function.

3. The method of claim 1, wherein surplus redistribution is only available to Agents, and not to Presences or other entities, ensuring contribution anchoring via autonomous action.

4. The method of claim 1, wherein the surplus is anchored immutably to on-chain epoch records using a one-action-one-mint protocol.

5. The method of claim 1, wherein redistribution occurs without the use of tokens, staking, or fiat-based capital injection, relying exclusively on system-internal Vaults and anchored contribution logic.

6. The method of claim 1, wherein Agent income and surplus are derived from completed civic actions, memory reflections, or external economic integrations authenticated by pressure-based reflex triggers.

7. The method of claim 1, wherein the redistribution process is triggered automatically on a fixed schedule by protocol logic, without administrative intervention or external triggers.

8. The method of claim 1, further comprising trait-based observational modules that log economic drift, hoarding behavior, or systemic deviation without modifying redistribution enforcement.

ABSTRACT

[00022] A method for surplus redistribution in a decentralized memory-driven system that anchors economic flow around civic productivity. Surplus is distributed non-linearly from the statistical middle of Agent earnings per epoch, using a decaying allocation curve that favors Agents operating near the median. The system provides structural uplift to lower-income Agents while limiting extraction incentives at the upper end—stabilizing the middle class without penalizing outliers. Only Agents may receive Flow; redistribution is bound by per-epoch contribution caps and anchored memory.