The Cell Protocol: Formal Corrections and Implementation Specification

Errata, Proofs, and Build Plan for a Zero-Sum Mutual Credit Protocol for Resilient Communities

R. Roberts

February 2026

Abstract

The Cell is a mutual credit protocol for small-scale community economies (60–100 households), designed to provide essential goods and services through reciprocal exchange under a zero-sum conservation invariant. A critical review of the protocol specification identified three mathematical gaps and five specification gaps in the original treatment. This paper formalises corrections for all three mathematical gaps—cell account conservation, compound federation exposure, and emergency mode de-escalation—together with two extensions: a time-dependent extraction bound that strengthens Sybil resistance, and a refined cooperation condition that separates time preference from continuation probability. Each correction is accompanied by a formal proof that the protocol's safety properties are preserved. The paper concludes with a phased implementation plan for the Wexford pilot, mapping specification sections to concrete engineering tasks with dependency ordering and time estimates. This document serves both as an academic errata and as a build guide for implementors.

Contents

1 Introduction

The Cell protocol specifies a mutual credit system for communities of 60–100 households. Credits are not minted or destroyed; every transaction is a paired transfer that preserves a global zero-sum invariant. The protocol provides:

1. **Conservation** — the sum of all balances is identically zero at all times;
2. **Bounded extraction** — no identity can extract more than its balance plus credit limit;
3. **Cooperation incentives** — a repeated-game condition under which rational agents prefer continued participation to defection;
4. **Federation** — bilateral links between cells with exposure caps that bound contagion;
5. **Emergency mode** — automatic parameter tightening under stress.

A critical review of the specification revealed three mathematical gaps (Sections 3–5) and identified the need for two extensions (Sections 6–7). Five additional specification gaps are catalogued in Section 8. Section 9 maps the corrected specification to a phased implementation plan for the Wexford pilot deployment.

Scope. This paper addresses the formal protocol layer. The existing Connect Again application implements a simplified Level 1 interface (individual balances, basic exchange matching, −10 debt floor) atop Supabase. The corrections and build plan herein target the full Level 2 specification: household accounts, cell accounts, meal rota settlement, repair subsidies, and federation.

2 System Model and Invariants

We establish notation and restate the core invariants from the original specification, against which all corrections are verified.

2.1 Entities and Balances

Definition 2.1 (Cell)**.** A cell $\mathcal{C}_k$ at time t consists of:

- A set of households $H_k(t) = \{h_1, \ldots, h_{N(t)}\}$, where $N(t) \in [N_{\min}, N_{\max}]$.
- Each household h has balance $b_h(t) \in \mathbb{R}$, credit limit $L_h > 0$, and reserve $r_h(t) \geq 0$.
- A governance council elected from member households.

Definition 2.2 (Credit Limit)**.** For household h with n_h adults and d_h dependents:

$$L_h = n_h \cdot L_{\text{adult}} + \varphi(d_h) \cdot L_{\text{dep}}$$

where $\varphi : \mathbb{N} \to \mathbb{R}_{\geq 0}$ is sublinear with $\varphi(d) = \min(d,\, d_{\max})$.

Definition 2.3 (Transaction)**.** A transaction $T(h_i, h_j, v)$ with $v > 0$ executes:

$$b_{h_i} \leftarrow b_{h_i} - v, \qquad b_{h_j} \leftarrow b_{h_j} + v$$

subject to the feasibility constraint $b_{h_i} - r_{h_i} - v \geq -L_{h_i}$.

2.2 Core Invariants

Invariant 2.1 (Conservation — I1)**.**

$$\sum_{h \in H_k(t)} b_h(t) = 0 \qquad \forall\, t$$

This holds by construction: every transaction subtracts v from one household and adds v to another. No other operation modifies balances.

Invariant 2.2 (Bounded Extraction — I2)**.** *For any household h with static limit L_h, the maximum one-shot extraction is:*

$$G_h \le b_h + L_h$$

This follows from the floor constraint: spending reduces b_h until $b_h = -L_h$, at which point no further transactions are feasible.

2.3 Federation

Definition 2.4 (Federation Link)**.** A bilateral link l between cells $\mathcal{C}_j$ and $\mathcal{C}_k$ carries a net position $B_{k,l}(t) \in \mathbb{R}$, where $B_{k,l} > 0$ means $\mathcal{C}_k$ is a net creditor to $\mathcal{C}_j$ on link l. The aggregate external position is $B_k(t) = \sum_{l \in \text{links}(k)} B_{k,l}(t)$.

2.4 Emergency Mode

The protocol defines a stress index:

$$\sigma(t) = \max\bigl(S_M(t),\, S_E(t)\bigr)$$

where S_M is the membership stress index and S_E is the energy stress index. Upward transitions: Normal → Stressed when $\sigma \ge \theta_S = 1.0$; Stressed → Panic when $\sigma \ge \theta_P = 1.2$.

3 Correction 1: Cell Account Formalisation

3.1 Problem Statement

The protocol references several cell-level accounts without formal treatment:

- **Meal Production Account** (Addendum 8): intermediary for food rota settlement.
- **Cell Maintenance Account** b_M (Track B, Policy B2.2): funds repair subsidies.
- **Tool Library Account** b_T (Track B, Policy B5): holds tool deposit compensations.

These accounts participate in credit transfers but have no defined limits, funding rules, or conservation treatment. A cell account that can go arbitrarily negative becomes a mechanism for creating unbacked credit, violating bounded extraction.

3.2 Definitions

Definition 3.1 (Cell Account)**.** Let cell $\mathcal{C}_k$ have a set of cell accounts $\mathcal{A}_k = \{a_1, \ldots, a_P\}$. Each cell account a_p has:

- Balance: $b_{a_p}(t) \in \mathbb{R}$.
- Floor limit: $b_{a_p}(t) \ge -L_{a_p}$.

- Ceiling limit: $b_{a_p}(t) \leq U_{a_p}$ (optional; prevents accumulation).

Cell accounts are **not members**. They cannot initiate transactions, hold commitments, or vote. They are passive ledger entries that receive and disburse credits under governance-approved rules.

Invariant 3.1 (Extended Conservation — I1′)**.**

$$\sum_{h \in H_k(t)} b_h(t) \; + \; \sum_{p=1}^{P} b_{a_p}(t) = 0 \qquad \forall\, t$$

3.3 Funding Mechanism

Cell accounts cannot create credit. They must be funded by explicit transfers from households.

Definition 3.2 (Levy)**.** A levy is a periodic transfer from each household to a cell account:

$$\ell_p \;:\; \text{credits per household per period for account } a_p$$

Execution for each household h:

$$\begin{aligned} b_h &\leftarrow b_h - \ell_p \\ b_{a_p} &\leftarrow b_{a_p} + \ell_p \end{aligned}$$

Feasibility: $b_h - r_h - \ell_p \geq -L_h$. If a household cannot pay (too close to floor), the levy is *deferred*—not waived—creating a receivable settled when the household next earns.

3.4 Recommended Limits

Meal Production Account.

$$L_{\text{meal}} = N \cdot c_E, \qquad U_{\text{meal}} = 2 \cdot N \cdot c_E$$

where c_E is the essential bundle cost per household and N is cell size.

Cell Maintenance Account.

$$L_M = R_{\max} \cdot (1 - \gamma), \qquad U_M = 4 \cdot L_M$$

where $R_{\max}$ is the largest expected Class 2 repair value and γ is the household co-pay fraction.

Tool Library Account.

$$L_T = 0, \qquad U_T = \sum_u d(u)$$

where $d(u)$ is the deposit value for tool u. The tool account should never go negative; it is funded entirely by deposits.

3.5 Conservation Proof

Proposition 3.1 (Conservation Under Cell Account Operations)**.** *If every cell account operation is a paired transfer (one side household, one side cell account), and cell accounts participate in the conservation sum, then Invariant I1′ is preserved.*

Proof. Let T be any cell account operation. T takes one of two forms:

Case (a): Household → Cell Account. $b_h \leftarrow b_h - v, \quad b_{a_p} \leftarrow b_{a_p} + v.$

$$\sum b'_h + \sum b'_{a_p} = \left(\sum b_h - v\right) + \left(\sum b_{a_p} + v\right) = \sum b_h + \sum b_{a_p} = 0. \checkmark$$

Case (b): Cell Account → Household. $b_{a_p} \leftarrow b_{a_p} - v, \quad b_h \leftarrow b_h + v.$

$$\sum b'_h + \sum b'_{a_p} = \left(\sum b_h + v\right) + \left(\sum b_{a_p} - v\right) = \sum b_h + \sum b_{a_p} = 0. \checkmark$$

No other operation type is admissible for cell accounts. □

Proposition 3.2 (Cell Account Floor Preserves Bounded Loss). *With cell account floors enforced, the maximum value extractable via cell account subsidies is bounded by L_{a_p} per account beyond its current balance.*

Proof. A cell account disburses credits to households. Each disbursement reduces b_{a_p}. The floor constraint $b_{a_p} \geq -L_{a_p}$ bounds total disbursements beyond initial balance:

$$\text{Total disbursements} \leq b_{a_p}(t_0) + L_{a_p}.$$

An attacker who captures governance and attempts to drain the cell account via fraudulent subsidies can extract at most $b_{a_p}(t_0) + L_{a_p}$ before the account hits its floor. With $b_{a_p}(t_0) \leq U_{a_p}$, the absolute worst case is $U_{a_p} + L_{a_p}$.

For the maintenance account with recommended limits:

$$\text{Max extraction} = U_M + L_M = 4L_M + L_M = 5R_{\max}(1 - \gamma).$$

This is a bounded, known, auditable quantity. □

3.6 Settlement Mechanics: Food Rota

For a meal event with total worker-hours W and N_r receiving households:

Worker credits. For each worker household w contributing h_w hours:

$$b_{\text{meal}} \leftarrow b_{\text{meal}} - h_w$$
$$b_w \leftarrow b_w + h_w$$

Receiver debits. For each receiving household r:

$$\text{debit}_r = W/N_r \quad \text{(equal share)}$$
$$b_r \leftarrow b_r - \text{debit}_r$$
$$b_{\text{meal}} \leftarrow b_{\text{meal}} + \text{debit}_r$$

Verification.

$$\Delta\Big(\sum b_h\Big) = \sum h_w - \sum \text{debit}_r, \qquad \Delta(b_{\text{meal}}) = -\sum h_w + \sum \text{debit}_r$$

$$\text{Total } \Delta = 0. \checkmark$$

The meal account is a pass-through. Over time, if total debits equal total credits, b_{meal} stays near zero. Small imbalances (rounding, partial attendance) are absorbed by the account's limit.

4 Correction 2: Compound Federation Exposure

4.1 Problem Statement

The original specification constrains the *aggregate net* external position:

$$|B_k(t)| \leq \beta \cdot \Lambda_k(t)$$

where $B_k = \sum_{l \in \mathrm{links}(k)} B_{k,l}$ and Λ_k is aggregate credit capacity. However, B_k is a signed sum. Offsetting positions across links mean the individual positions $|B_{k,l}|$ could each be large while $|B_k|$ remains small. Under partial severance (losing some links while retaining others), the remaining aggregate position could exceed the cap if the severed links were providing offsetting positions.

4.2 Corrected Constraint

Constraint 4.1 (Aggregate Absolute Exposure — F1)**.**

$$\sum_{l \in \mathrm{links}(k)} |B_{k,l}(t)| \leq \beta \cdot \Lambda_k(t)$$

This is strictly tighter than the original.

Constraint 4.2 (Per-Link Sub-Cap — F2 (recommended))**.**

$$|B_{k,l}(t)| \leq \frac{\beta}{d_k} \cdot \Lambda_k(t) \qquad \forall\, l \in \mathrm{links}(k)$$

where $d_k = |\mathrm{links}(k)|$ *is the federation degree.*

Remark 4.1. F2 implies F1:

$$\sum_l |B_{k,l}| \leq \sum_l \frac{\beta}{d_k} \cdot \Lambda_k = d_k \cdot \frac{\beta}{d_k} \cdot \Lambda_k = \beta \cdot \Lambda_k. \checkmark$$

F2 is more restrictive (prevents concentrating exposure in one link) but simpler to enforce: each link only checks its own position against a fixed sub-cap.

4.3 Compound Severance Bound

Proposition 4.1 (Compound Severance Bound Under F1)**.** *Under Constraint F1, if an arbitrary subset* $S \subseteq \mathrm{links}(k)$ *of federation links is simultaneously severed, the total value at risk for cell* $\mathcal{C}_k$ *is bounded by* $\beta \cdot \Lambda_k(t)$*.*

Proof. Upon severance of link l, the position $B_{k,l}$ freezes. The cell cannot settle this position and must absorb it as a loss (if $B_{k,l} > 0$, the cell is owed value it cannot collect; if $B_{k,l} < 0$, the cell owes value it may need to write off).

The maximum total at-risk value is:

$$\sum_{l \in S} |B_{k,l}| \leq \sum_{l \in \mathrm{links}(k)} |B_{k,l}| \leq \beta \cdot \Lambda_k \qquad \text{(by F1)}.$$

This bound holds regardless of $|S|$—whether one link or all links are severed simultaneously. □

4.4 Implementation Note

The existing federation engine enforces $|B_k| \leq \beta\Lambda_k$ on the aggregate net position. To enforce F1, each inter-cell transaction must additionally check:

$$\sum_l |B_{k,l} + \Delta B_{k,l}| \leq \beta \cdot \Lambda_k.$$

This requires tracking per-link positions $B_{k,l}$, which the engine already maintains. The validation change is: sum the absolute values of all per-link positions and compare against $\beta\Lambda_k$.

5 Correction 3: Emergency Mode De-Escalation

5.1 Problem Statement

The specification defines upward transitions (Normal → Stressed → Panic) based on the stress index $\sigma(t)$ exceeding thresholds, with hysteresis to prevent flapping. Downward transitions are not specified. Without explicit de-escalation rules:

- A cell that enters Panic may never exit, even after conditions improve.
- Parameters tightened during emergency (L, β) are never restored.
- The cell permanently operates under constrained conditions.

5.2 De-Escalation Thresholds

Define downward thresholds with hysteresis gap $h \in [0.05, 0.10]$ (default $h = 0.05$):

$$\begin{aligned} \theta_{S\downarrow} &= \theta_S - h = 1.0 - h && \text{(Stressed} \rightarrow \text{Normal trigger)} \\ \theta_{P\downarrow} &= \theta_P - h = 1.2 - h && \text{(Panic} \rightarrow \text{Stressed trigger)} \end{aligned}$$

Define cooldown periods τ (in settlement periods, e.g. weeks):

$$\tau_S = 2, \qquad \tau_P = 3.$$

5.3 State Machine

PANIC → STRESSED. Requires:

$$\sigma(t-j) < \theta_{P\downarrow} \qquad \forall\, j \in \{0, 1, \ldots, \tau_P - 1\}$$

i.e., the stress index must remain below $\theta_P - h$ for τ_P consecutive periods.

Upon transition, restore parameters to stressed-mode values:

$$\begin{aligned} \beta &\leftarrow \beta_{\text{stressed}} \\ L_i &\leftarrow \min(L_i/\lambda_2,\ L_{\max}) \\ \text{Admission} &\leftarrow \textsc{Bonded} \\ \text{Commitments} &\leftarrow \textsc{Escrow-Essentials} \\ \text{Scheduler} &\leftarrow \textsc{Essentials-First} \end{aligned}$$

STRESSED → NORMAL. Requires:

$$\sigma(t-j) < \theta_{S\downarrow} \qquad \forall\, j \in \{0, 1, \ldots, \tau_S - 1\}$$

Upon transition, restore all parameters to normal-mode values.

5.4 Gradual Parameter Restoration

Instant restoration risks snapback instability. For parameter P with emergency value P_e and normal value P_n, at period j after de-escalation ($j = 0, \ldots, R-1$):

$$P(j) = P_e + (P_n - P_e) \cdot \frac{j}{R}$$

Recommended restoration periods:

$$R_\beta = 4 \text{ periods}, \qquad R_L = 3 \text{ periods}.$$

After leaving Panic, full return to normal takes approximately $\tau_P + R \approx 7$ periods. This is intentionally slow—premature relaxation invites defection.

5.5 Re-Escalation During Restoration

If $\sigma(t)$ rises above the relevant upward threshold during the restoration period, the cell immediately re-enters the higher state. The cooldown timer resets. There is no protection from re-escalation during restoration.

5.6 Safety Proof

Proposition 5.1 (De-Escalation Preserves Safety)**.** *De-escalation with hysteresis and cooldown does not weaken the bounded-loss property.*

Proof. During de-escalation, parameters are restored gradually. At every point in the restoration:

(a) Conservation (I1′) is maintained because de-escalation modifies caps and modes, not balances.

(b) The debt floor (I2) is maintained: L_i increases during restoration, which *relaxes* the floor (allows more debt). This does not violate I2; it expands the feasible region.

(c) Bounded extraction: during restoration, L_i increases from L_e to L_n. The extraction bound $G_i \leq b_i + L_i$ also increases. An attacker who waited for de-escalation could extract more than during emergency. However: (i) the attacker must wait $\tau_P + R$ periods of cooperative behaviour, (ii) maximum extraction is still bounded by L_n (the normal limit), and (iii) the cooperation condition was designed for L_n.

De-escalation restores the normal security posture. It does not create a new vulnerability. □

5.7 Governance Override

The council may:

- **Delay de-escalation**: vote to remain in a higher state even if stress indices qualify for de-escalation.
- **Accelerate de-escalation**: vote to skip cooldown if conditions are clearly safe (requires supermajority).
- **NOT bypass upward escalation**: if $\sigma(t) \geq \theta_P$, the system enters Panic regardless of council preference.

This asymmetry is deliberate: escalation is automatic (safety), de-escalation requires evidence (caution).

6 Extension 1: Time-Dependent Extraction Bound

6.1 Problem Statement

Proposition 3.2 and Invariant I2 bound one-shot extraction assuming static L_i. If an attacker can get L_i raised—via governance capture, reputation growth, or de-escalation—the effective extraction bound depends on the *maximum* L_i achieved during the attack window.

6.2 Dynamic Bound

Proposition 6.1 (Time-Dependent Extraction Bound)**.** *Let identity i exist from time t_0 to $t_0 + T$. Let $L_i(t)$ be the credit limit at time t, subject to rate-limiting $|L_i(t+1) - L_i(t)| \leq \eta$. Then maximum net extraction over the period is:*

$$G_i(T) \leq b_i(t_0) + L_i(t_0) + \eta \cdot T.$$

Proof. At any time t, the extraction bound is $G_i \leq b_i(t) + L_i(t)$. The limit at time t satisfies:

$$L_i(t) \leq L_i(t_0) + \eta \cdot (t - t_0)$$

by repeated application of the rate limit.

The attacker's optimal strategy is to first raise L_i as fast as possible (waiting T periods gains ηT additional limit), then extract at the maximum limit. The attacker cannot simultaneously extract and raise L_i (extracting reduces b_i, which does not affect L_i but reduces remaining extractable value).

At time $t_0 + T$, the worst case is:

- $b_i(t_0 + T) = b_i(t_0)$ (attacker cooperated to build trust);
- $L_i(t_0 + T) = L_i(t_0) + \eta T$;
- Single extraction: $G_i = b_i(t_0) + L_i(t_0) + \eta T$.

□

6.3 Implications for Sybil Attacks

For S identities each operating for T periods:

$$G_A(T) \leq S \cdot \big(b_0 + L_0 + \eta T\big).$$

With newcomers starting at $b_0 = 0$ and $L_0 = L_{\text{new}} \ll L$:

$$G_A(T) \leq S \cdot \big(L_{\text{new}} + \eta T\big).$$

Slow privilege (small η) is the primary defence against patient Sybil attacks. Even with many identities, if η is small, the attack requires proportionally more time, increasing detection probability.

6.4 Recommended Parameters

From the Wexford pilot parameters: $L = 20$ credits (one week of essentials), $L_{\text{new}} = 5$ credits. If an attacker should need at least 26 weeks to reach full limit:

$$\eta \leq \frac{L - L_{\text{new}}}{26} = \frac{15}{26} \approx 0.58.$$

Rounded down: $\eta = 0.5$ credits per week.

An attacker controlling $S = 3$ Sybil identities for $T = 26$ weeks extracts at most:

$$G_A = 3 \cdot (5 + 0.5 \times 26) = 3 \times 18 = 54 \text{ credits}.$$

This is roughly 3 person-weeks of essential labour—significant but survivable for an 80-person cell, and it required 6 months of sustained cooperative behaviour from 3 fake identities.

7 Extension 2: Cooperation Condition Refinement

7.1 Problem Statement

The original cooperation condition uses a single discount factor δ that conflates:

- **Time preference** ρ: how much the agent values future utility relative to present.
- **Continuation probability** π: probability the cell persists to the next period.

These behave differently under stress. In crisis, π may drop sharply (the cell might dissolve) while ρ is relatively stable. Collapsing them hides the mechanism by which crises destabilise cooperation.

7.2 Refined Utility Model

Definition 7.1 (Effective Discount Factor)**.**

$$\delta(t) = \rho \cdot \pi(t)$$

where $\rho \in (0, 1)$ is pure time preference and $\pi(t) \in (0, 1)$ is continuation probability at time t.

The value of cooperation in the stationary case (u^C and π constant):

$$V^C = \frac{u^C}{1 - \rho\pi}$$

The defection payoff:

$$V^D = \alpha L + w$$

where α is the conversion factor from extracted credits to utility and w is the outside option.

7.3 Refined Cooperation Condition

Cooperation is rational when $V^C \geq V^D$:

$$\frac{u^C}{1 - \rho\pi} \geq \alpha L + w$$

Rearranging for the maximum permissible L:

$$L \leq \frac{u^C - (1 - \rho\pi) \cdot w}{\alpha \cdot (1 - \rho\pi)} \tag{1}$$

7.4 Comparative Statics

(a) Continuation probability π drops (crisis).

$$\frac{\partial \,\text{RHS}}{\partial \pi} = \frac{\rho(u^C - w)}{\alpha(1 - \rho\pi)^2}$$

If $u^C > w$ (the cell provides more than the outside option), then $\partial\text{RHS}/\partial\pi > 0$—a drop in π *tightens* the permissible limit.

Worked example. Let $\rho = 0.95$, $u^C = 15$, $w = 2$, $\alpha = 1$.

$$\pi = 0.95 \implies L \leq \frac{15 - 0.0975 \times 2}{1 \times 0.0975} = \frac{14.805}{0.0975} \approx 152$$
$$\pi = 0.70 \implies L \leq \frac{15 - 0.335 \times 2}{1 \times 0.335} = \frac{14.33}{0.335} \approx 43$$

The permissible L drops by ~72%. With $L = 20$, normal conditions have ample headroom; crisis conditions still hold ($43 > 20$) but with much thinner margin.

(b) Cell utility u^C drops (essentials shortage).

$$\frac{\partial \, \text{RHS}}{\partial u^C} = \frac{1}{\alpha(1 - \rho\pi)} > 0$$

A drop in u^C directly tightens the permissible L.

(c) Outside option w rises (external aid appears).

$$\frac{\partial \, \text{RHS}}{\partial w} = -\frac{1}{\alpha} < 0$$

External generosity *destabilises* mutual credit—the aid dependency problem.

7.5 Interaction with Emergency Mode

Emergency mode applies $L \leftarrow \lambda L$ with $\lambda < 1$. This is correct if λL stays below the RHS of (1). The protocol should verify:

$$\lambda L \leq \frac{u^C_{\text{min}} - (1 - \rho\pi_{\text{min}}) \cdot w_{\text{max}}}{\alpha \cdot (1 - \rho\pi_{\text{min}})}$$

where u^C_{min}, π_{min}, w_{max} are worst-case values. If this inequality fails, cooperation breaks down even with emergency tightening, and the cell should enter a **dissolution protocol** rather than persisting under Panic mode.

7.6 Empirical Calibration

The parameters u^C, π, w, α are empirical. The Wexford pilot should measure:

1. Weekly confidence survey: "Would you stay?" → proxy for π.
2. Essential bundle delivery rate → proxy for u^C.
3. External price comparison for equivalent services → proxy for w.
4. Exit interview data → direct measurement of defection reasoning.

8 Specification Gaps

Beyond the mathematical corrections, the critical review identified seven specification gaps requiring resolution before or during pilot operation.

Specification Gap 8.1 (Cell Account Mechanics)**.** Multiple addendums reference cell accounts but they lack formal treatment. **Resolution**: Section 3 of this paper.

Specification Gap 8.2 (Multi-Cell Member Migration)**.** The protocol assumes each member belongs to exactly one cell. Transfer of reputation and balance upon migration is unspecified. **Recommendation**: Transfer request → cooldown → balance settlement → departure → admission at new cell. Outstanding commitments must be settled before transfer. Reputation: portable summary vs. fresh start (governance decision per cell).

Specification Gap 8.3 (Cell Splitting and Merging)**.** As cells grow beyond $N_{\max}$ or shrink below $N_{\min}$, they must split or merge. No protocol exists for balance allocation, commitment transfer, or federation link inheritance during these operations.

Specification Gap 8.4 (Dispute Resolution Timelines)**.** The specification states disputes should have "fixed timelines" and "simple outcomes" but provides no concrete deadlines. **Recommendation**: Evidence submission 48h, council ruling 7 days, appeal window 72h, auto-escalation to cell-wide vote if council is deadlocked.

Specification Gap 8.5 (Offline Sync Conflict Resolution)**.** The architecture requires offline-first operation, meaning concurrent edits to the ledger can occur. The protocol does not specify conflict resolution (CRDT, last-writer-wins, manual merge). **Recommendation**: Timestamp + vector clock ordering; both transactions valid if both feasible; reject later if infeasible; first-commit-wins for commitment conflicts.

Specification Gap 8.6 (Reputation Signal Aggregation)**.** Sybil resistance references reputation signals (transaction consistency, fulfilment rate, dispute involvement, network participation, tenure) but provides no formal aggregation function.

Specification Gap 8.7 (Federation Link Formation Protocol)**.** Who initiates federation links? The specification says "bilateral proposals" but the negotiation protocol—information exchange, due diligence, trial period—is underspecified. **Recommendation**: Proposal with cell metadata exchange → trial period with reduced β → full activation → periodic health checks.

9 Implementation Status and Build Plan

9.1 Current State

The existing implementation comprises ∼28,700 lines across the cell protocol engine. Table 1 summarises coverage.

Known defects.

1. Member churn calculation is a placeholder (always returns 0) in the emergency engine.
2. Rare escrow safety violations under certain operation sequences in invariant tests (seed-dependent, documented).

9.2 Phased Build Plan

Table 2 presents the implementation phases ordered by pilot criticality. The critical path for pilot launch (Phases 0–4 plus 6) is approximately 15–23 days of implementation work.

9.3 Priority Ordering for Wexford Pilot

If the goal is a working pilot, the ordering should be:

1. Phase 0 — fix known bugs (prerequisite for everything).
2. Phase 1 — household accounts (dependents cannot participate without this).

Table 1: Implementation status vs. specification.

Specification Section	**Status**	**Notes**
Core Ledger (I1–I3)	Complete	All invariants enforced
Transactions (T1)	Complete	Ed25519 signatures
Commitments (C0/C1)	Complete	9 task categories
Identity / Admission	Complete	Sybil hooks
Governance	Complete	Council, proposals, disputes
Federation	Complete	Bilateral links, exposure caps
Emergency Mode	Complete	3 states, 6 stress indicators
Survival Scheduler	Complete	Coverage analysis, matching
Energy Layer	Complete	5 carriers, rationing, stress
Sybil Resistance	Complete	Sponsor/service bonds
Hardening / Testing	Complete	Property tests, adversarial
Household Accounts	Not started	Addendum 4
Cell Accounts	Not started	This paper, §3
MVC Gate	Not started	Addendum 5
Meal Rota UX	Not started	Addendum 8
Repair / Tool Library	Not started	Track B PRD
Cell Splitting / Merging	Not specified	Gaps 8.3
Member Migration	Not specified	Gap 8.2

Table 2: Phased build plan with dependency ordering.

Phase	**Scope**	**Days**	**Depends On**
0	Fix known bugs (churn calc, escrow)	1–2	—
1	Household accounts	3–5	Phase 0
2	Cell accounts (§3)	2–3	Phase 0
3	MVC gate (role coverage validation)	2–3	Phase 1
4	Meal rota / food track	5–7	Phases 1, 2
5	Repair & tool library	5–7	Phase 2
6	Dispute resolution SLAs	2–3	Phase 0
7	Emergency de-escalation (§5)	1–2	Phase 0
8	Offline sync protocol	3–5	Phase 0
9	Federation hardening (§4)	2–3	Phase 0
10	Cell lifecycle (split/merge/migrate)	3–5	All above

3. Phase 2 — cell accounts (needed for subsidies in Phases 4/5).

4. Phase 4 — meal rota (*this is the pilot's first activity*).

5. Phase 3 — MVC gate (validate cell viability).

6. Phase 6 — dispute SLAs (needed before real disputes arise).

7. Phase 7 — emergency de-escalation (robustness).

8. Phase 5 — repair/tool library (second vertical, post-launch).

9. Phase 8 — offline sync (important but online-first is acceptable for v1).

10. Phase 9 — federation hardening (only when multiple cells exist).

11. Phase 10 — cell lifecycle (only at scale).

Critical path for pilot launch: Phases 0 → 1 → 2 → 4 → 3 → 6 ≈ 15–23 days.

9.4 Phase Detail: Key Tasks

Phase 1: Household Accounts.

- Define household types: entity (id, members, balance, limit, reserve), formation/modification events.
- Implement household engine: creation with governance approval, member add/remove, balance operations, limit calculation per Definition 2.2.
- Integrate with ledger: conservation over households ($\sum b_h = 0$), transaction feasibility.
- Integrate with scheduler: essential bundle debits to households, task credits to worker households.
- Test suite: bounded extraction ($G_h \leq L_h$), dependent inflation prevention, slow privilege enforcement.

Phase 2: Cell Accounts.

- Implement per Definition 3.1: maintenance account b_M, tool library account b_T, meal production account.
- Enforce limits (floor, ceiling) per Section 3.
- Levy mechanism with deferred collection.
- Prove conservation (Proposition 3.1) holds in test suite.

Phase 4: Meal Rota.

- Meal event state machine: Draft → Open → Locked → InProgress → Completed → Settled → Disputed.
- Role commitment with escrow (C1 integration).
- Settlement via cell meal account per Section 3.6.
- Tests: full lifecycle, missed commitment → dispute, near-floor household cannot overcommit, settlement preserves conservation.

10 Conclusion

This paper has formalised three corrections and two extensions to the Cell mutual credit protocol:

1. **Cell account formalisation** (Section 3): defines cell accounts as bounded passive ledger entries, extends the conservation invariant to include them, and proves that bounded extraction is preserved.
2. **Compound federation exposure** (Section 4): replaces the aggregate net exposure cap with an aggregate absolute exposure cap, preventing compound severance from exceeding $\beta\Lambda_k$.
3. **Emergency mode de-escalation** (Section 5): completes the emergency state machine with hysteresis-based downward transitions, gradual parameter restoration, and a proof that safety properties are preserved.

4. **Time-dependent extraction bound** (Section 6): extends the static extraction bound to account for limit growth over time, quantifying Sybil attack cost as a function of the slow-privilege rate η.

5. **Cooperation condition refinement** (Section 7): separates time preference from continuation probability, derives comparative statics showing how crises, essentials shortages, and external aid affect cooperation stability, and identifies the empirical measurements needed for calibration.

All corrections preserve the protocol's core safety properties: zero-sum conservation and bounded extraction. The phased build plan provides a concrete path from corrected specification to working pilot, with the critical path for the Wexford deployment estimated at 15–23 implementation days.

A Summary of Invariant Changes

Table 3: Invariant and constraint changes.

ID	Original	Corrected	Section
I1	$\sum_h b_h = 0$	$\sum_h b_h + \sum_p b_{a_p} = 0$	3
F-cap	$\lvert B_k\rvert \leq \beta\Lambda_k$	$\sum_l \lvert B_{k,l}\rvert \leq \beta\Lambda_k$	4
Emergency	Upward only	Upward + downward with hysteresis	5
Extraction	$G_i \leq b_i + L_i$ (static)	$G_i(T) \leq b_i + L_i + \eta T$	6
Cooperation	$\delta = \rho\pi$ (single)	$\delta(t) = \rho \cdot \pi(t)$ (separated)	7

B Notation Reference

Table 4: Symbol definitions.

Symbol	**Meaning**
$\mathcal{C}_k$	Cell k
$H_k(t)$	Set of households in cell k at time t
$b_h(t)$	Balance of household h at time t
L_h	Credit limit for household h
$r_h(t)$	Reserve (escrowed) for household h
$b_{a_p}(t)$	Balance of cell account a_p
L_{a_p}	Floor limit for cell account a_p
U_{a_p}	Ceiling limit for cell account a_p
ℓ_p	Levy rate for cell account a_p
c_E	Essential bundle cost per household per period
$R_{\max}$	Largest expected Class 2 repair value
γ	Household co-pay fraction for repairs
$B_{k,l}(t)$	Net position of cell k on federation link l
$\Lambda_k(t)$	Aggregate credit capacity of cell k
β	Federation exposure cap coefficient
d_k	Federation degree (number of links)
$\sigma(t)$	Combined stress index
S_M, S_E	Membership and energy stress indices
θ_S, θ_P	Upward thresholds (Stressed, Panic)
h	Hysteresis gap
τ_S, τ_P	Cooldown periods
η	Rate limit on credit limit changes
$G_i(T)$	Maximum extraction over T periods
ρ	Pure time discount factor
$\pi(t)$	Continuation probability at time t
$\delta(t)$	Effective discount factor ($\rho \cdot \pi(t)$)
u^C	Per-period utility from cooperation
α	Conversion factor (credits to utility)
w	Outside option utility
λ	Emergency limit tightening factor (< 1)