# REPAIR DEPTH OF MULTI-ORIENTATION EDGE-MINIMUM BROADCAST REPAIR IN DENSE EISENSTEIN–JACOBI NETWORKS: DEPTH $t+1$ SUFFICES FOR TWO FAULTS AT EVERY DIAMETER $4 \le t \le 19$

KOREA SUPERINTELLIGENCE LABS

Abstract. The dense Eisenstein–Jacobi network $H_t$ is the Cayley graph of $\mathbb{Z}[\omega]/(\alpha)$, $\alpha = (t+1)+t\omega$, on the six units; it has $N = 3t^2+3t+1$ vertices and diameter $t$. Albader (arXiv:2606.19834) repairs a source-rooted broadcast tree of $H_t$ after the failure of at most two vertices by choosing one of fifteen coordinate-reduction orientations, deleting the faults, and reconnecting the surviving components with the minimum number $c-1$ of external edges (EJ-MOEM). He proves that a valid repair of depth at most $t+2$ always exists, observes in an exhaustive table that depth $t+1$ sufficed for every one- and two-fault placement at $t = 2, \dots, 12, 14, 16, 18$ except at $t = 3$, and asks whether depth $t+1$ is universally sufficient for $t \ge 4$. We answer the question for every diameter $4 \le t \le 19$: for each of the 2,871,160 fault sets $F$ with $|F| \le 2$ some orientation admits a valid non-redundant repair of depth at most $t+1$ (the diameters $13, 15, 17, 19$ lie outside the source's table), and for $4 \le t \le 18$ the bound is attained by an explicit two-fault placement that admits no valid repair of depth $\le t$ in any orientation, the minimum being taken over *all* valid repairs and not over the output of one algorithm. At $t = 3$ the exceptional placements are exactly the six pairs of adjacent vertices at distance one from the source, with repair depth exactly $5 = t+2$; every other placement has depth at most 4. Restricting to *star* repairs, in which every repair edge leaves the source component, the optimum is exactly $2t-1$ for $4 \le t \le 8$, so at those diameters chaining detached components is necessary and its absence costs $t-2$ units of depth. Two errata in the source's algebra are recorded: the displayed root of unity is a cube root where the model needs a sixth root, and the printed vertex label is not well defined on $\mathbb{Z}[\omega]/(\alpha)$. Every finite statement is certified by compiled evaluation of a certificate checker inside Lean 4; the errata rest on identities the Lean kernel proves for every $t$.

## 1. Introduction

**The objects.** Let $\omega = e^{i\pi/3}$, a primitive sixth root of unity, so that $\omega^2 = \omega - 1$ and $\mathbb{Z}[\omega]$ is the ring of Eisenstein integers with norm $\mathrm{N}(x+y\omega) = x^2+xy+y^2$. For an integer $t \ge 1$ put $\alpha = (t+1)+t\omega$; then $\mathrm{N}(\alpha) = 3t^2+3t+1 =: N$ and the quotient ring $\mathbb{Z}[\omega]/(\alpha)$ has $N$ elements. The *dense Eisenstein–Jacobi network* $\Gamma_t$ is the Cayley graph of the additive group of $\mathbb{Z}[\omega]/(\alpha)$ with respect to the six units $\pm 1, \pm\omega, \pm\omega^2$. Writing $x+y\omega$ as the axial pair $(x,y) \in \mathbb{Z}^2$, the six units are

$$E = (1,0), \quad SE = (1,-1), \quad S = (0,-1), \quad W = (-1,0), \quad NW = (-1,1), \quad N = (0,1),$$

and the hexagonal distance from the origin is $\rho(x,y) = \max\{|x|, |y|, |x+y|\}$. The radius-$t$ ball $H_t = \{(x,y) \in \mathbb{Z}^2 : \rho(x,y) \le t\}$ has exactly $3t^2+3t+1 = N$ points and is an exact set of representatives of $\mathbb{Z}[\omega]/(\alpha)$ — this is what "dense" means (Martínez, Stafford, Beivide and Gabidulin [4]; Flahive and Bose [5]). So the vertices of $\Gamma_t$ are the points of $H_t$, a step in a unit direction that stays inside the ball is an edge of the hexagonal grid, and a step that leaves the ball wraps to the unique point of $H_t$ congruent to it modulo $\alpha$. The graph is 6-regular, has diameter $t$, and $\rho$ is the layer (the distance from the source $s = (0,0)$).

A *coordinate-reduction orientation* $\theta$ is a priority list of the six directions; it assigns to every $v \ne s$ the parent $p_\theta(v) = v + \delta$ for the first direction $\delta$ in the list with $\rho(v+\delta) = \rho(v) - 1$, and the resulting parent map is a spanning tree $T_\theta$ of $\Gamma_t$ rooted at $s$ whose depth function is $\rho$; a one-to-all broadcast along $T_\theta$ reaches every vertex in $t$ steps. The source's method, EJ-MOEM, uses the fifteen orientations $\Theta = \{C_0, \dots, C_5, R_0, \dots, R_5, A_0, A_1, A_2\}$ of its Table 1 (Section 2). When a set $F$ of at most two vertices fails, $T_\theta - F$ is a forest with $c$ components, and a *valid repair* is a spanning tree of the healthy subgraph $\Gamma_t - F$, rooted at $s$, that keeps every surviving edge of $T_\theta$; it necessarily uses exactly $c-1$ new edges, one per detached component. Its *depth* is the eccentricity of $s$ in it, and the quantity studied here is

$$\mathrm{D}(F) = \min_{\theta\in\Theta} \min\{\text{depth of } P :\ P \text{ a valid repair of } T_\theta - F\},$$

the best repair depth EJ-MOEM can certify for the fault set $F$.

**The source and its open question.** The source is Albader's arXiv:2606.19834 [1]. Its Theorem 3 ("EJ-MOEM $t+2$ depth theorem") reads, verbatim:

> "For every dense EJ network $H_t$ with $t \geq 2$ and every fault set $F \subseteq H_t \setminus \{s\}$ with $|F| \leq 2$, the EJ-MOEM orientation family contains a valid repair of depth at most $t+2$. If $|F| = 1$, depth at most $t+1$ is sufficient."

Its Table 4 ("Extended exhaustive EJ-MOEM validation") enumerates every one- and two-fault placement for $t = 2, \ldots, 12, 14, 16, 18$ — up to $N = 1027$ vertices and 525,825 two-fault placements at $t = 18$ — and prints, for each diameter, the maximum depth of the repair its Algorithm 2 returned: $t+1$ everywhere except at $t = 3$ with $|F| = 2$, where it prints $5 = t+2$. Its Section 9 (Discussion) draws the conclusion and poses the question, verbatim:

> "The $t+2$ bound is a worst-case theorem. The validation run shows that the bound is attained in the smallest nontrivial two-fault case $t = 3$, but no overhead-two case was observed for $t \geq 4$ in the extended exhaustive run. A natural open question is whether depth $t+1$ is universally sufficient for all $t \geq 4$."

(Statement, table, example, section and equation numbers of [1] are those of the PDF compiled from the version-1 e-print and its `.aux` file; see the bibliography.) Two further points of the source matter for what follows. Its Algorithm 1, step 6, returns "the valid candidate of minimum depth; ties prefer fewer repair edges, then lower maximum out-degree, then deterministic edge order", and its Algorithm 2 selects among orientations lexicographically by "(success, depth, repair edges, maximum out-degree, orientation rank)" (its equation (16)); so the *depth* the source reports is a minimum and does not depend on any tie-break, which is why $\mathrm{D}(F)$ is the right object and why no unstated edge order has to be guessed. And the code behind Table 4 is not public ("available from the corresponding author upon reasonable request"), so the table's values rest on a run nobody outside can replay.

**Results.**

- **Depth $t+1$ suffices,** $4 \leq t \leq 19$ (Theorem 4.1). For every diameter $4 \leq t \leq 19$ and every $F \subseteq H_t \setminus \{s\}$ with $|F| \leq 2$ — $1 + (N-1) + \binom{N-1}{2}$ fault sets per diameter, 2,871,160 in all — some orientation of $\Theta$ admits a valid repair of $T_\theta - F$ of depth at most $t+1$. For $t \in \{4, \ldots, 12, 14, 16, 18\}$ the value is the source's own (its Table 4 prints maximum overhead 1); what is added is a certificate that can be replayed. For $t \in \{13, 15, 17, 19\}$ — 149,332, 259,561, 421,822 and 650,371 fault sets — the source has no value at all: its exhaustive table skips 13, 15 and 17 and stops at 18, and its structured tests at $t = 15$ (its Table 5) cover seven hand-picked fault families totalling 7,589 trials, not the 258,840 two-fault placements.
- **The bound is exact,** $4 \leq t \leq 18$ (Theorem 4.2, Corollary 4.3). The two-fault placement $F_t = \{(-t, 0),\ (1-t, t-1)\}$ — the boundary vertex $t \cdot W$ and the layer-$(t-1)$ vertex $(t-1)\cdot NW$ — admits no valid repair of depth $\leq t$ in any orientation, the minimum being taken over all valid repairs and not over the candidates of an algorithm, while it admits one of depth $t+1$. Hence $\max_{|F|\leq 2} \mathrm{D}(F) = t+1$ exactly for $4 \leq t \leq 18$. The source's Table 4 prints "Max depth $= t+1$" for the diameters it covers, but that number is the depth of the tree its algorithm returned; minimality over all valid repairs, and the value at $t = 13, 15, 17$, are new.
- **The $t = 3$ exception, characterised** (Theorem 5.1). At $t = 3$, among the 667 fault sets with $|F| \leq 2$, exactly six have $\mathrm{D}(F) = 5 = t+2$, namely the six pairs of *adjacent* layer-1 vertices

$$\{(-1,0),(-1,1)\},\quad \{(-1,0),(0,-1)\},\quad \{(-1,1),(0,1)\},$$
$$\{(0,-1),(1,-1)\},\quad \{(0,1),(1,0)\},\quad \{(1,-1),(1,0)\},$$

and every other fault set has $\mathrm{D}(F) \leq 4$. The source says only that the bound "is attained in the smallest nontrivial two-fault case $t = 3$".

- **Chained attachment is necessary, and star repairs cost** $2t - 1$ (Theorem 6.1). Call a valid repair a *star* repair when every repair edge joins the source component to a detached component. For $4 \leq t \leq 8$ every fault set with $|F| \leq 2$ admits a star repair of depth at most $2t-1$, and the placement $\{(-1,0),(-1,1)\}$ admits none of depth $\leq 2t-2$ in any orientation, so the star optimum is exactly $2t-1$ for $4 \leq t \leq 8$: at those diameters the $t+1$ bound cannot be reached by star certificates at all, and the step at which the source's two-fault lemmas attach the second detached component to the first is not a weakness of the proof but a necessity.
- **Two errata** (Propositions 7.1 and 7.2). The source's equation (2) displays $\omega = (-1 + i\sqrt{3})/2$, a primitive *cube* root of unity, for which $\mathrm{N}(\alpha) = t^2 + t + 1 < 3t^2 + 3t + 1$; its direction set, its distance and its order formula all need the sixth root. Its equation (10) prints the vertex label $\phi(x + y\omega) \equiv tx + (2t+1)y \pmod{N}$, which sends $\alpha$ to $3t^2 + 2t \not\equiv 0$ and so is not a function on $\mathbb{Z}[\omega]/(\alpha)$; the author's companion preprint arXiv:2606.21133 [3] prints $tx - (t+1)y$ for the same $\alpha$ and $N$, which is. Neither erratum touches the algorithm, Theorem 3 or the validation, because the label is never used by EJ-MOEM and the root of unity is only a display.

**What is new and what is not.** The network family, its density and its diameter are due to Martínez, Stafford, Beivide and Gabidulin [4] and Flahive and Bose [5]; the fifteen-orientation family, the edge-minimum repair theorem ($c - 1$ external edges are necessary and sufficient), the $t + 2$ depth theorem and the exhaustive table for $t = 2, \ldots, 12, 14, 16, 18$ are the source's [1]. The affirmative answer at $t \in \{13, 15, 17, 19\}$, the minimality over all valid repairs, the count and characterisation of the $t = 3$ exceptions, the star result and the two errata are claimed here. As of 2026-09-07 the source has no citing work in OpenAlex, and neither do the author's later preprints [2, 3]; a full-text sweep of the local arXiv mirror (376 shards, coverage through the September 2026 announcements) for the phrase "Eisenstein–Jacobi" — case-insensitively, with zero, one or two hyphens and an optional space — matches 34 shards and, e-print by e-print inside them, 51 distinct preprints. Thirteen of the 51 are the author's own June 2026 cluster. The other 38 — modular-form papers on Eisenstein–Jacobi *series* and Jacobi forms, coding theory over Eisenstein–Jacobi *integers* in the sense of Huber [9], perfect codes in circulant graphs citing Thomson and Zhou [6] or Hussain, Bose and Al-Dhelaan [7], and bibliography entries — were read one at a time, and not one of them concerns fault-tolerant broadcast repair in any network. Of the author's cluster, arXiv:2606.21132 [2] proves the same $t + 1/t + 2$ bounds for a constant-time selector and validates it exhaustively for $t = 2, \ldots, 8$ only (52,395 cases, which is the full count of one- and two-fault placements there); arXiv:2606.21133 [3] repeats word for word the observation of the source's Section 7.3 that "overhead two appears only in the small $t = 3$ exhaustive case, while all larger exhaustive, structured, and random tests observed maximum overhead at most one", and proves a depth bound of $2t + 1$ in a node-and-link fault model; neither settles the question or claims to. The nearest earlier work on trees in these networks, Hussain, AboElFotoh and AlBdaiwi [8], pre-builds independent spanning trees and does not treat post-fault repair.

**Verification.** Every finite statement below is certified in Lean 4: a certificate checker, written in a module that imports nothing, verifies that a proposed parent map is a valid repair and recomputes its depth; an exhaustive search over all valid repairs decides the lower bounds; and the Lean compiler evaluates the resulting Boolean sweeps in native code, the result being admitted by the kernel through Lean's native-evaluation axiom. The passage from the Boolean sweep to the quantified sentence ("for every fault set there is an orientation and a checked repair of depth $\leq k$") is a kernel-checked theorem without native evaluation, and the two errata are kernel-checked for all $t$ with no native evaluation at all. Section 9 says precisely what is trusted; Appendix A reproduces the formal statements. Everything that the text says beyond its formal counterpart — the reading of the source's parent rule in Remark 2.1, the reconstruction of the source's "Min good" column and the observations of Section 8, the mechanism described in Remark 4.4, and the independent re-computations of Section 9 — is labelled as such where it occurs.

## 2. Preliminaries

2.1. **The network.** Fix $t \geq 1$ and let $\omega$, $\alpha$, $N$, the six directions, $\rho$ and $H_t$ be as in the introduction. The ideal $(\alpha)$ is the $\mathbb{Z}$-module spanned by $\alpha = (t+1, t)$ and $\alpha\omega = -t + (2t+1)\omega = (-t, 2t+1)$; its six shortest vectors are $\pm(t+1, t)$, $\pm(-t, 2t+1)$, $\pm(-2t-1, t+1)$. For $v \in H_t$ and a direction $\delta$ the neighbour of $v$ in direction $\delta$ is $v + \delta$ if $\rho(v + \delta) \leq t$, and otherwise the unique $u \in H_t$ with $u \equiv v + \delta \pmod{\alpha}$, which is $v + \delta$ minus one of the six shortest vectors. This is the graph $\Gamma_t$. The source (its Section 3) states the same model with the caveats of Section 7: its equation (2) displays the cube root of unity and its equation (10) a label that is not a function on the quotient. Since $\mathbb{Z}[e^{i\pi/3}] = \mathbb{Z}[(-1 + i\sqrt{3})/2]$ as subrings of $\mathbb{C}$, the object is the same and only the printed formulas differ.

2.2. **Orientations and their trees.** Index the directions cyclically, $\delta_0 = E$, $\delta_1 = SE$, $\delta_2 = S$, $\delta_3 = W$, $\delta_4 = NW$, $\delta_5 = N$, indices modulo 6. The fifteen orientations of the source's Table 1 are

$$C_i = (\delta_i, \delta_{i+1}, \ldots, \delta_{i+5}), \qquad R_i = (\delta_i, \delta_{i-1}, \ldots, \delta_{i-5}), \qquad 0 \leq i < 6,$$
$$A_i = (\delta_i, \delta_{i+3}, \delta_{i+1}, \delta_{i+4}, \delta_{i+2}, \delta_{i+5}), \qquad 0 \leq i < 3,$$

taken in the order $C_0, \ldots, C_5, R_0, \ldots, R_5, A_0, A_1, A_2$; for instance $A_0 = (E, W, SE, NW, S, N)$. For an orientation $\theta$ and $v \in H_t \setminus \{s\}$, the parent $p_\theta(v)$ is $v+\delta$ for the first $\delta$ in the list $\theta$ with $\rho(v+\delta) = \rho(v)-1$, the test being made on the point $v + \delta \in \mathbb{Z}^2$; since $\rho(v + \delta) < t$ that point lies in $H_t$ and no wraparound is involved. Every $v \neq s$ has such a $\delta$, so $p_\theta$ is a spanning tree $T_\theta$ rooted at $s$ with depth function $\rho$ (the source's Lemma 1); Proposition 3.1 checks this for $2 \leq t \leq 12$.

*Remark* 2.1 (the parent rule, as read). The source defines the rule in its Section 3.1 ("chooses, for every non-source vertex $v \in H_t$, one neighbor $p(v)$ satisfying $\rho(p(v)) = \rho(v) - 1$") and its Example 2 ("An orientation priority list decides which inward neighbor is selected"). Read literally on the quotient, a boundary vertex $v$ (layer $t$) and an outward direction $\delta$ would also have to be tested, the neighbour then being the wrapped representative of $v + \delta$. We test $\rho$ on the unwrapped point, so that outward directions are never parent directions. Whichever reading is intended, the parent map is a tree with depth function $\rho$, since the layer decreases by one along every parent edge. The two readings can differ only if the wrapped representative of some $v + \delta$ with $\rho(v) = t$ and $\rho(v + \delta) = t + 1$ lay on layer $t - 1$. Computations outside the formal development (Section 9) find no such pair for $2 \leq t \leq 40$: every wrapped neighbour of a boundary vertex is again a boundary vertex. Over that range — which contains the whole certified range $2 \leq t \leq 19$ — the two readings therefore produce the same fifteen trees, vertex by vertex. That is a computation and not a proof: for $t \geq 41$ nothing here excludes a difference, and the theorems below are in any case statements about the unwrapped-rule trees, which are the trees the formal development builds.

2.3. **Faults, repairs, depth.** A fault set is $F \subseteq H_t \setminus \{s\}$ with $|F| \leq 2$; there are $1 + (N-1) + \binom{N-1}{2}$ of them, the empty set included. Deleting $F$ from $T_\theta$ leaves a forest $T_\theta - F$ with $c = c_\theta(F)$ components, the *source component* $C_s \ni s$ and $c - 1$ *detached* components.

**Definition 2.2.** A *valid repair* of $T_\theta - F$ is a map $P$ assigning to every healthy vertex $v \neq s$ a parent $P(v)$ such that

(1) $P(v)$ is healthy and adjacent to $v$ in $\Gamma_t$;
(2) following $P$ from any healthy vertex reaches $s$, so that $P$ is a spanning tree of $\Gamma_t - F$ rooted at $s$ (every healthy vertex receives the message exactly once, no faulty vertex is used, there is no cycle);
(3) every edge of $T_\theta$ with both ends healthy is an edge of $P$;
(4) exactly $c - 1$ edges of $P$ are not edges of $T_\theta$.

Its *depth* is the maximum over healthy $v$ of the distance from $s$ to $v$ in $P$. $P$ is a *star* repair if each of its $c - 1$ new edges has an endpoint in $C_s$.

Conditions (1)–(3) say that $P$ is a spanning tree of the healthy subgraph containing the surviving forest, and (4) is then automatic — it is the source's Lemma 2 and Theorem 1 that $c - 1$ crossing edges are necessary and sufficient — but the checker verifies it separately. Equivalently, a valid repair is a rooted spanning tree of the component graph together with one healthy crossing edge per attachment: the detached components are hung, one at a time, off the part already attached, and a vertex $v$ of a component entered at $b$ through the edge $(a, b)$ receives depth $d(a)+1+\operatorname{dist}(b, v)$, the distance measured inside the component. This is exactly the accounting of the source's equation (15) and its Lemma 3.

**Definition 2.3.** For a fault set $F$ let $\mathrm{D}_\theta(F)$ be the minimum depth of a valid repair of $T_\theta - F$ ($\infty$ if none exists), and $\mathrm{D}(F) = \min_{\theta\in\Theta} \mathrm{D}_\theta(F)$. Define $\mathrm{D}^\star_\theta(F)$ and $\mathrm{D}^\star(F)$ in the same way over star repairs.

The source's Theorem 3 says $\mathrm{D}(F) \le t + 2$ for $|F| \le 2$ and $\mathrm{D}(F) \le t + 1$ for $|F| \le 1$, at every $t \ge 2$ — in particular $\mathrm{D}(F) < \infty$, so $\Gamma_t - F$ is connected and every orientation's component graph is connected — and its open question is whether $\max_{|F|\le 2} \mathrm{D}(F) \le t + 1$ for all $t \ge 4$. The depth its Algorithm 2 returns for $F$ is at least $\mathrm{D}(F)$, for the plain reason that every candidate its Algorithm 1 validates is a valid repair in the sense of Definition 2.2, and $\mathrm{D}_\theta(F)$ is the minimum over all of those. The inequality can be strict: step 4 of its Algorithm 1 reads, verbatim, "Enumerate the rooted component-level spanning choices of $\mathcal{C}_\theta$; for each ordered component attachment, use the stored crossing edge with minimum attachment value", so the attachment orderings are enumerated but only one crossing edge per ordered attachment — the one minimising the predicted value of its equation (15) — is ever tried. Upper bounds on D certified by an explicit repair are therefore statements about the repair problem that the source's algorithm can realise, and lower bounds on D are statements about the problem that no algorithm can beat.

2.4. **What is computed.** The formal development consists of a model of $\Gamma_t$ and of the fifteen trees, an *untrusted* greedy construction that hangs the detached components one at a time through the crossing edge minimising $d(a)+1+\operatorname{ecc}(b)$ (the source's predicted attachment value, its equation (15)), a *certificate checker* `checkRepair` that decides conditions (1)–(4) of Definition 2.2 for a proposed parent map and recomputes its depth by a breadth-first search from $s$, and an *exhaustive search* `existsRepairLe` that decides whether a valid repair of depth $\le k$ exists at all. The search proceeds attachment by attachment; at each step its candidates are the pairs (entry vertex $b$ of a still-detached component, offset $d(a) + 1$ for an already-attached neighbour $a$ of $b$), every distinct pair being tried, so that every rooted spanning tree of the component graph and every choice of crossing edge is reachable, and it uses no dominance argument and no heuristic bound. With the star flag set, both the construction and the search require the attached endpoint $a$ to lie in $C_s$. For a diameter $t$ and a bound $k$ the *sweep* `sweep` $t$ $k$ is the Boolean "for every fault set $F$ with $|F| \le 2$ some orientation's constructed repair passes the checker with depth $\le k$", and `sweepStar` is the same with the star flag. A kernel-checked bridge (`sweep_spec`, Appendix A) turns `sweep` $t$ $k$ $=$ `true` into the quantified sentence.

## 3. The model is the source's

**Proposition 3.1** (non-vacuity of the model)**.** *For every $t \in \{2, \dots, 12\}$ the following hold, by compiled evaluation.*

(i) *The vertex set built has $3t^2 + 3t + 1$ elements and the source has layer 0; for every vertex $v$ and direction $\delta_d$ the neighbour $u$ of $v$ in direction $\delta_d$ is a vertex different from $v$, and the neighbour of $u$ in direction $\delta_{d+3}$ is $v$.*

(ii) *The corrected label $(x, y) \mapsto (t+1)x + (2t+1)y \bmod N$ of Proposition 7.2 is injective on $H_t$ and $|H_t| = N$; since that label is a $\mathbb{Z}$-linear map $\mathbb{Z}^2 \to \mathbb{Z}/N$ vanishing on $(\alpha)$ and $|\mathbb{Z}^2/(\alpha)| = N$, the composite $H_t \to \mathbb{Z}^2/(\alpha)$ is a bijection, so $H_t$ is an exact set of representatives of $\mathbb{Z}[\omega]/(\alpha)$, the wraparound edges are well defined and the network is dense. (The corrected labels are isomorphisms of additive groups, not ring homomorphisms: they send 1 to $t + 1$ and to $t$, not to 1.)*

(iii) *There are fifteen orientation trees; in each, the source has no parent and every other vertex has a parent one layer nearer the source, so each is a spanning tree rooted at $s$ with depth function $\rho$, and the maximum layer is $t$.*

(iv) *The fault-set enumeration has exactly $1 + (N - 1) + \binom{N-1}{2}$ entries, each a canonical pair of distinct non-source vertices (or one, or none).*

Nothing in Proposition 3.1 is claimed as new: the density is [4, 5] and (iii) is the source's Lemma 1. It exists so that the sweeps below are known to be about the source's object.

**Proposition 3.2** (controls). (a) *The checker is not vacuous. Starting from a repair that it accepts ($t = 4$, one fault at $(2, 1)$, orientation $C_0$), it rejects each of four perturbations: a vertex re-parented to the faulty vertex; a vertex made its own parent; a vertex re-parented to a non-adjacent vertex; a surviving tree edge deleted by re-parenting one endpoint.*

(b) *The source's worked examples reproduce. For its Example 4 ($t = 4$, $f = (2, 1)$, orientation $C_0$) the constructed repair passes the checker with depth $4 = t$, matching the source's certificate value $3 + 1 + 0 = 4$; for its Example 6 ($t = 3$, $F = \{(1, 0), (2, 0)\}$, orientation $A_0$) it passes with depth $4 = t + 1$, and the forest has two components, so one repair edge is used, matching the source's $3 + 1 + 0 = 4 = t + 1$.*

(c) *The $t+1$ claim is genuinely false at $t = 3$ and cannot be lowered: the sweep* `sweep` 3 4 *evaluates to* `false` *and* `sweep` 3 5 *to* `true`.

Part (c) is only about the greedy construction; that no valid repair of depth 4 exists for six placements at $t = 3$ is Theorem 5.1.

## 4. Depth $t + 1$ suffices, and is optimal

**Theorem 4.1** (depth $t + 1$, $4 \le t \le 19$). *Let $4 \le t \le 19$ and let $F \subseteq H_t \setminus \{s\}$ with $|F| \le 2$. Then some orientation $\theta \in \Theta$ admits a valid repair of $T_\theta - F$ of depth at most $t + 1$; that is, $\mathrm{D}(F) \le t + 1$. Explicitly, for each such $t$ and $F$ the constructed repair of some orientation passes the certificate checker with depth $\le t + 1$. The three groups of diameters are certified separately:*

(a) $t = 4, \ldots, 12$*:* 1,831*,* 4,096*,* 8,002*,* 14,197*,* 23,437*,* 36,586*,* 54,616*,* 78,607 *and* 109,747 *fault sets respectively;*

(b) $t = 14, 16, 18$*:* 198,766*,* 333,337 *and* 526,852 *fault sets;*

(c) $t = 13, 15, 17, 19$*:* 149,332*,* 259,561*,* 421,822 *and* 650,371 *fault sets.*

*Proof sketch.* For each $t$ the Boolean `sweep` $t$ $(t + 1)$ is evaluated by compiled code and found to be `true`: the enumeration of Proposition 3.1(iv) is run in full, and for each fault set the fifteen orientations are tried in the order of the source's Table 1 until one constructed repair passes the checker with depth $\le t + 1$. The bridge theorem then yields, for every $F$, an orientation $\theta$, a natural number $d \le t + 1$ and the fact that the constructed parent map satisfies (1)–(4) of Definition 2.2 with depth $d$. Groups (a), (b), (c) are the sweeps of the modules `SweepLow`, `SweepHiA` together with `SweepHiB`, and `SweepGap` together with `SweepTop` (Section 9). Nothing rests on the greedy construction being good; it is only a witness generator, and a fault set for which it produced a bad witness in every orientation would have made the sweep `false`. □

Groups (a) and (b) are the diameters of the source's Table 4, whose "Max over." column prints 1 for each of them; there the value is the source's and the contribution is the replayable certificate. Group (c) is outside the source's validation entirely: $t = 19$ ($N = 1141$) is one diameter beyond the largest it enumerated, and its structured tests jump from $t = 15$ to $t = 20$ with seven hand-picked fault families each.

**Theorem 4.2** ($t + 1$ is optimal, $4 \le t \le 18$). *Let $4 \le t \le 18$ and $F_t = \{(-t, 0), (1 - t, t - 1)\}$, the boundary vertex on the $W$ ray and the layer-$(t - 1)$ vertex on the $NW$ ray. Then $\mathrm{D}(F_t) = t + 1$: for every orientation $\theta \in \Theta$ there is no valid repair of $T_\theta - F_t$ of depth $\le t$, the search ranging over all valid repairs, while for some $\theta$ the constructed repair passes the checker with depth $\le t + 1$.*

*Proof sketch.* For each $t$ the exhaustive search `existsRepairLe` with bound $k = t$ is run for all fifteen orientations and returns `false` each time; then the constructed repair is checked at bound $t+1$. The search enumerates every rooted spanning tree of the component graph and every crossing edge (Section 2.4), so its negative answer is a lower bound on the repair problem, not on the source's Algorithm 1. □

**Corollary 4.3.** *For $4 \le t \le 18$, $\max\{\mathrm{D}(F) : F \subseteq H_t \setminus \{s\},\ |F| \le 2\} = t + 1$; at $t = 19$ the maximum is at most $t + 1$. In particular the answer to the source's question is affirmative at every diameter*

$4 \le t \le 19$, *and in that range the bound* $t+1$ *cannot be lowered to* $t$*. The question as the source poses it, for all* $t \ge 4$*, is not settled here.*

*Remark* 4.4 (the mechanism). The lower bound is visible in the geometry. The vertex $u = (t-1)\cdot NW = (1-t, t-1)$ is the parent, in every orientation, of the boundary vertex $t \cdot NW = (-t, t)$: from $(-t, t)$ the directions $E$ and $S$ lead to layer-$t$ points, $W$, $NW$ and $N$ leave the ball, and $SE$ alone leads one layer inward, to $u$. Once $u$ fails, every healthy neighbour of $(-t, t)$ lies on layer $t$, so in any valid repair $(-t, t)$ hangs below a vertex of depth at least $t$ and has depth at least $t+1$. The one thing this argument needs beyond the coordinates is that the three wrapped neighbours of $(-t, t)$ are themselves boundary vertices; the computations reported in Section 9 check that for every boundary vertex at every $2 \le t \le 40$ (Remark 2.1), and it is not proved here for general $t$. Theorem 4.2 does not depend on it: the theorem rests on the exhaustive search alone, and the paragraph above is stated only to say where the extra unit of depth comes from. The same computations confirm that the single fault $\{u\}$ already has $\mathrm{D}(\{u\}) = t+1$ for $4 \le t \le 8$, that the boundary fault $\{(-t, 0)\}$ alone has depth $t$, and that every one of the fifteen orientations has $\mathrm{D}_\theta(F_t) = t+1$ exactly, at those diameters.

*Remark* 4.5. The source's "Max depth" column is the maximum over placements of the depth of the tree its Algorithm 2 returned, and for each $F$ that depth is at least $\mathrm{D}(F)$. Theorem 4.2 shows that at $F_t$ no valid repair of depth $\le t$ exists at all, so the number the column prints, $t+1$, is the true value of $\max_{|F|\le 2} \mathrm{D}(F)$ for $t \in \{4, \dots, 12, 14, 16, 18\}$: on that column the source's run is not merely an upper bound. Whether its algorithm attains $\mathrm{D}(F)$ at each individual $F$ is a different question and is not settled here. For $t = 13, 15, 17$ no value existed.

## 5. The $t = 3$ exception

At $t = 3$ the network has $N = 37$ vertices, 667 fault sets with $|F| \le 2$, and the source's Table 4 prints maximum overhead 2 for two faults.

**Theorem 5.1** (the six exceptional placements)**.** *Let* $t = 3$ *and let* $\mathcal{E}$ *be the set of six two-fault placements*

$$\{(-1,0),(-1,1)\}, \quad \{(-1,0),(0,-1)\}, \quad \{(-1,1),(0,1)\},$$
$$\{(0,-1),(1,-1)\}, \quad \{(0,1),(1,0)\}, \quad \{(1,-1),(1,0)\}.$$

*Then:*

- (i) $\mathcal{E}$ *is exactly the set of pairs of adjacent layer-*1 *vertices, i.e. pairs of neighbours of* $s$ *that are adjacent to each other (both inclusions and both cardinalities are checked);*
- (ii) *for every* $F \in \mathcal{E}$ *and every orientation* $\theta \in \Theta$ *there is no valid repair of* $T_\theta - F$ *of depth* $\le 4$*, over all valid repairs, while for some* $\theta$ *the constructed repair passes the checker with depth* $\le 5$*; so* $\mathrm{D}(F) = 5 = t+2$*;*
- (iii) *for every fault set* $F \notin \mathcal{E}$ *with* $|F| \le 2$ *— the other* 624 *two-fault placements, the* 36 *one-fault placements and the empty set — some orientation's constructed repair passes the checker with depth* $\le 4$*; so* $\mathrm{D}(F) \le 4$*.*

*Proof sketch.* (i) is a finite comparison of two lists. (ii) runs the exhaustive search at bound 4 for the six placements and all fifteen orientations, and checks a depth-5 witness for each. (iii) is the sweep at bound 4 over the complement of $\mathcal{E}$. □

The source states only that its bound "is attained in the smallest nontrivial two-fault case $t = 3$"; it does not say how many placements attain it or which, and its statement concerns the output of its algorithm. The configuration that is exceptional here — two adjacent neighbours of the source — is exactly the one whose star cost is computed in the next section. A cheap lower bound, the eccentricity of $s$ in the graph formed by the surviving forest edges together with all healthy crossing edges, gives only 4 for these six placements; the value 5 needs the search over all valid repairs.

## 6. Star repairs cost $2t - 1$

**Theorem 6.1** (star repairs)**.** *Let $4 \le t \le 8$ and $R = \{(-1,0), (-1,1)\}$, two adjacent layer-1 vertices. Then:*

- (i) *for every $F \subseteq H_t \setminus \{s\}$ with $|F| \le 2$, some orientation's constructed* star *repair passes the checker with depth $\le 2t-1$; so $\mathrm{D}^\star(F) \le 2t-1$;*
- (ii) *for every orientation $\theta \in \Theta$ there is no star repair of $T_\theta - R$ of depth $\le 2t-2$, over all star repairs;*
- (iii) *for some orientation there is a star repair of $T_\theta - R$ of depth $\le 2t-1$.*

*Consequently* $\max_{|F|\le 2} \mathrm{D}^\star(F) = 2t-1$ *for $4 \le t \le 8$, whereas* $\max_{|F|\le 2} \mathrm{D}(F) = t+1$ *by Corollary 4.3.*

*Proof sketch.* (i) is the star sweep `sweepStar` $t$ $(2t-1)$; the construction with the star flag attaches every detached component through an edge whose attached endpoint lies in $C_s$, so its witnesses are star repairs by construction (the checker verifies validity and depth; the star property is guaranteed by the construction, not re-verified). (ii) and (iii) are the exhaustive search with the star flag at bounds $2t-2$ and $2t-1$. □

In the source's proof of Theorem 3, the two-fault lemmas lose their second unit of depth at a chained attachment. Lemma 10 (separated-sector faults) and Lemma 11 (adjacent-sector faults) say so in words — Lemma 11 reads "attach the component with the unblocked outer side first with value at most $t+1$; after it is attached, the second component has value at most $t+2$" — and in Lemma 9 (same-ray and opposite-ray faults) the outer suffix is entered "at depth at most $b+1$", which is the depth the first attached component reaches. Theorem 6.1 says the chaining is forced: at $4 \le t \le 8$, forbidding it costs not one unit but $t-2$, and the placement that exhibits this is the configuration that is exceptional at $t = 3$. The nearest statement in the author's cluster is Lemma 4 of arXiv:2606.21133 [3], a bound of $2t+1$ under a "shallowest-layer entry" rule in a node-and-link fault model; in the node-only model here the exact star value is $2t-1$ at $4 \le t \le 8$, so bounds of that shape are nearly tight for star-shaped certificates while being far from the truth of the repair problem.

## 7. Two errata in the source's algebra

Both statements below are proved for every integer $t$ by the Lean kernel using ordinary ring arithmetic; no compiled evaluation is involved. Neither affects the source's algorithm, its Theorem 3 or its validation.

**Proposition 7.1** (the root of unity)**.** *For every integer $t$,*

$$(t+1)^2 + (t+1)t + t^2 = 3t^2 + 3t + 1, \qquad (t+1)^2 - (t+1)t + t^2 = t^2 + t + 1,$$

*and $t^2+t+1 < 3t^2+3t+1$ whenever $t \ge 1$. Moreover, as polynomial identities in $x, y, X$: $(x+yX)(x+y(1-X)) = (x^2+xy+y^2) - y^2(X^2-X+1)$ and $(x+yX)(x+y(-1-X)) = (x^2-xy+y^2) - y^2(X^2+X+1)$.*

Consequently, with $\omega^2 = \omega - 1$ (the sixth root $e^{i\pi/3}$) the norm form is $x^2 + xy + y^2$ and $\mathrm{N}(\alpha) = 3t^2 + 3t + 1$, the order the source states in its equations (1) and (9); with $\omega^2 = -1 - \omega$ (the cube root $(-1+i\sqrt{3})/2$ that its equation (2) displays) the norm form is $x^2 - xy + y^2$ and $\mathrm{N}(\alpha) = t^2+t+1$, which is strictly smaller for every $t \ge 1$. With the displayed $\omega$ the six units are $\pm 1, \pm\omega, \pm(1+\omega)$, in coordinates $\pm(1,0), \pm(0,1), \pm(1,1)$, whereas the source's direction set (its equation (3)) is $\pm(1,0), \pm(0,1), \pm(1,-1)$ and its distance (its equation (5)) is $\max\{|x|, |y|, |x+y|\}$, both of which belong to the sixth root. So the displayed root cannot be the one the paper uses. The same display recurs in the author's later preprints (equation (2) of [3], Section 3.2 of [2]).

**Proposition 7.2** (the vertex label)**.** *For every integer $t$:*

- (i) *$t(t+1) + (2t+1)t = 3t^2 + 2t$, and $0 < 3t^2 + 2t < 3t^2 + 3t + 1$ whenever $t \ge 1$;*
- (ii) *$(t+1)(t+1) + (2t+1)t = 3t^2 + 3t + 1$ and $(t+1)(-t) + (2t+1)(2t+1) = 3t^2 + 3t + 1$;*
- (iii) *$t(t+1) - (t+1)t = 0$ and $t(-t) - (t+1)(2t+1) = -(3t^2 + 3t + 1)$.*

*At $t = 3$: $3 \cdot 4 + 7 \cdot 3 = 33$ and $33 \not\equiv 0 \pmod{37}$, while $3 \cdot 4 + 33 \cdot 3$, $3 \cdot (-3) + 33 \cdot 7$, $4 \cdot 4 + 7 \cdot 3$ and $4 \cdot (-3) + 7 \cdot 7$ are all $\equiv 0 \pmod{37}$.*

The source's equation (10) prints $\phi(x+y\omega) \equiv tx+(2t+1)y \pmod N$ as "the integer label associated with an axial coordinate", and its Example 1 instantiates it at $t=3$ as $\phi(1,0)=3$, $\phi(0,1)=7$. A label of the vertices of $\mathbb{Z}[\omega]/(\alpha)$ is a $\mathbb{Z}$-linear map $\mathbb{Z}^2 \to \mathbb{Z}/N$ vanishing on the ideal, i.e. on its $\mathbb{Z}$-module generators $\alpha=(t+1,t)$ and $\alpha\omega=(-t,2t+1)$ (the identity $((t+1)+tX)X = (-t+(2t+1)X)+t(X^2-X+1)$ is also kernel-checked). By (i) the printed map sends $\alpha$ to a residue strictly between 0 and $N$, so it is not constant on the cosets of $(\alpha)$ and is not a function on the quotient; concretely, at $t=3$ the point $(4,3)=\alpha$ is congruent to the source $(0,0)$, yet the printed formula gives it the label 33 and the source the label 0. Exactly one coefficient has to move and either repair works: by (ii), $(t+1)x+(2t+1)y$ sends both generators to $N$; by (iii), $tx-(t+1)y$ sends them to 0 and $-N$. The second form is what the author's companion preprint arXiv:2606.21133 [3] prints, in its Section 3 (its equation (8)) and in the caption of its Figure 1 ("Integer labels use $\phi(x+y\omega)=3x-4y \pmod{37}$" at $t=3$), for the same $\alpha$ and the same $N$; so the source's $(2t+1)$ is a slip in the $y$-coefficient with the $x$-coefficient left at $t$. The author's two preprints of the following day disagree with each other here — [2] repeats the source's map in its equation (10), [3] prints the corrected one — which is itself evidence that the printed form is a slip rather than a convention. The source's Example 1 values cannot both hold under any valid label: since $t$ is invertible modulo $N$, fixing $\phi(1,0)=3$ forces $\phi(0,1)\equiv -4 \equiv 33$, and fixing $\phi(0,1)=7$ forces $\phi(1,0)=4$; the four residues displayed above are the kernel-checked instances.

*Remark* 7.3 (outside the formal development). The printed map is injective on the ball $H_t$ for $1 \le t \le 12$ (checked in Section 9), so as a numbering of the $N$ points of $H_t$ it is a bijection; its defect is entirely at the wraparound. At $t=3$ the $E$-neighbour of the boundary vertex $(3,0)$ is $(0,-3)$, whose printed label 16 differs from $9+3$; under the printed label 9 of the 111 edges of $\Gamma_3$ receive a difference outside $\{\pm 3, \pm 4, \pm 7\}$, whereas under $tx-(t+1)y$ every $E$-edge is the jump $+t$. This is what "not well defined on the quotient" means for the network.

## 8. Measurements outside the formal development

The statements of this section are computations, not theorems of the formal development; they are recorded because they locate the certified results inside the source's own table and extend it. An independent C implementation, written from the source's definitions before any Lean file, reproduces every printed entry of the source's Table 4 — all 26 rows, all seven data columns (Cases, Success, Max depth, Max over., Max repair, Max comp., Min good) — for $t=2,\ldots,12,14,16,18$ and both $|F|=1$ and $|F|=2$, and extends the same table to $t=13,15,17,19,20$. Table 1 shows the two-fault rows.

Three points. First, the source never defines its "Min good" column. The reconstruction that reproduces it exactly is: the number of orientations whose own candidate meets the proved bound $t+2$ — not the number of orientations that succeed, which is always 15 (Section 2), and not the number that attain the optimum. With that definition the column reads 15 at $t=2,3$ and, for every $t$ from 4 to 20 without exception, 12 for $|F|=1$ and 7 for $|F|=2$. The source defines neither the column nor what makes an orientation "good", so this is a reverse-engineering of the printed values and a reading of the table, not a fact about the source; what recommends it is that it reproduces all 26 printed entries and nothing else tried does. Second, these statistics describe the candidates of an algorithm, not the repair problem: an exhaustive search over all valid repairs (Section 9) finds that at $t=4$ every one of the fifteen orientations admits a valid repair of depth $\le t+1$ for every fault set with $|F| \le 2$, and at $t=5$ every orientation admits one of depth $\le t+2$ for every fault set while at least 12 admit one of depth $\le t+1$. Third, $t=20$ ($N=1261$, 794,431 fault sets) behaves exactly like every $t \ge 4$: max depth $21=t+1$ with the cheap lower bound (the eccentricity of $s$ in the graph formed by the surviving forest edges and all healthy crossing edges, minimised over orientations and maximised over placements) also equal to 21, so $t+1$ is optimal there as well; this diameter has no Lean certificate and is recorded as data only. The constancy of the whole profile from $t=4$ to $t=20$ is the strongest evidence available here that $t+1$ is a theorem for all $t \ge 4$ and not a coincidence of the range; a proof would have to close the gap in the source's two-fault lemmas (its Lemmas 9–11), which give up their unit of depth at the second attachment; and Theorem 6.1 shows, at the diameters where it is proved, that the gap has to be closed by a better *order* of attachment and not by a better choice of entry edges.

| $t$ | $N$ | two-fault placements | max depth | over. | max repair | max comp. | Min good | in [1] | certified by |
|---|---|---|---|---|---|---|---|---|---|
| 2 | 19 | 153 | 3 | 1 | 3 | 4 | 15 | yes | – |
| 3 | 37 | 630 | 5 | 2 | 3 | 4 | 15 | yes | 5.1 |
| 4 | 61 | 1,770 | 5 | 1 | 4 | 5 | 7 | yes | 4.1(a), 4.2, 6.1 |
| 5 | 91 | 4,005 | 6 | 1 | 4 | 5 | 7 | yes | 4.1(a), 4.2, 6.1 |
| 6 | 127 | 7,875 | 7 | 1 | 4 | 5 | 7 | yes | 4.1(a), 4.2, 6.1 |
| 7 | 169 | 14,028 | 8 | 1 | 4 | 5 | 7 | yes | 4.1(a), 4.2, 6.1 |
| 8 | 217 | 23,220 | 9 | 1 | 4 | 5 | 7 | yes | 4.1(a), 4.2, 6.1 |
| 9 | 271 | 36,315 | 10 | 1 | 4 | 5 | 7 | yes | 4.1(a), 4.2 |
| 10 | 331 | 54,285 | 11 | 1 | 4 | 5 | 7 | yes | 4.1(a), 4.2 |
| 11 | 397 | 78,210 | 12 | 1 | 4 | 5 | 7 | yes | 4.1(a), 4.2 |
| 12 | 469 | 109,278 | 13 | 1 | 4 | 5 | 7 | yes | 4.1(a), 4.2 |
| 13 | 547 | 148,785 | 14 | 1 | 4 | 5 | 7 | **no** | 4.1(c), 4.2 |
| 14 | 631 | 198,135 | 15 | 1 | 4 | 5 | 7 | yes | 4.1(b), 4.2 |
| 15 | 721 | 258,840 | 16 | 1 | 4 | 5 | 7 | **no** | 4.1(c), 4.2 |
| 16 | 817 | 332,520 | 17 | 1 | 4 | 5 | 7 | yes | 4.1(b), 4.2 |
| 17 | 919 | 420,903 | 18 | 1 | 4 | 5 | 7 | **no** | 4.1(c), 4.2 |
| 18 | 1027 | 525,825 | 19 | 1 | 4 | 5 | 7 | yes | 4.1(b), 4.2 |
| 19 | 1141 | 649,230 | 20 | 1 | 4 | 5 | 7 | **no** | 4.1(c) |
| 20 | 1261 | 793,170 | 21 | 1 | 4 | 5 | 7 | **no** | – |

TABLE 1. Two-fault rows of the source's Table 4 as reproduced by the C implementation, extended to $t = 13, 15, 17, 19, 20$. "max depth" is the maximum over placements of the depth of the greedy candidate selected over the fifteen orientations; "over." is max depth minus $t$; "max repair" and "max comp." are the maximum number of repair edges and of components of the selected candidate; "Min good" is defined in the text. The last column names the theorems of this paper that certify the row's *depth* column, in the stronger form $\max_{|F|\le 2} \mathrm{D}(F) = t+1$ (for $t = 3$, $= t+2$ on six placements); the other columns are statistics of the constructed candidates and have no formal counterpart. The one-fault rows (not shown) have max depth $t+1$, max repair 1, max comp. 2 and Min good 15 at $t = 2, 3$ and 12 at every $t \ge 4$, again matching the source where it has a row. For $t \ge 4$ the whole profile is constant in $t$ up to $t = 20$. Every "yes" row matches the source's printed row entry by entry.

## 9. Verification

The formal development is ten Lean 4 modules, 1,245 lines in all, checked with Lean 4.32.0. `Engine.lean` (569 lines) imports nothing, not even the standard library's extended tactics, and defines the network (`mkNet`), the fifteen trees (`mkPar`, `orients`), the untrusted greedy construction (`buildRepair`), the certificate checker (`checkRepair`), the exhaustive search (`existsRepairLe`), the sweeps (`sweep`, `sweepStar`, `sweepExcept`) and the two label maps; because it imports nothing it is compiled to a shared library and every sweep is evaluated as native code rather than by Lean's interpreter (measured speed-up about 27× on this code). `Spec.lean` proves the bridge from the Boolean sweep to the quantified statement (`sweep_spec`, `sweepStar_spec`) by ordinary kernel reasoning; its axioms are `propext` and `Quot.sound` only. `Model.lean` holds Propositions 3.1 and 3.2 (`model_sane`, `controls`, `t3_not_t_add_one`); `Optimal.lean` holds Theorems 4.2, 5.1 and 6.1 (`t_add_one_optimal`, `t3_exceptional_set`, `star_necessary`); `SweepLow`, `SweepHiA`, `SweepHiB`, `SweepGap` and `SweepTop` hold Theorem 4.1(a), (b), (b), (c), (c) as the sweeps `sweep_low`, `sweep_hi_a`, `sweep_hi_b`, `sweep_gap`, `sweep_top` together with their quantified corollaries `sweep_low_spec`, etc.; and `Errata.lean` holds Propositions 7.1 and 7.2 (`alpha_norm`, `label_erratum`, `label_erratum_example`) using *Mathlib*'s `ring`, `nlinarith` and `norm_num` [10, 11].

*What is trusted.* Eleven theorems are Boolean equalities proved by `native_decide`: the Lean compiler compiles the definitions of `Engine.lean` to C and then to a shared library, the sweep is evaluated as native code, and the kernel admits the resulting equality on the strength of Lean's native-evaluation axiom, which appears in the axiom print of each theorem as `<name>._native.native_decide.ax_1_1`. The library is not a re-implementation: the `.olean` that the sweep files import and the `.c` that becomes the library are emitted by one and the same invocation of `lean` on `Engine.lean`, so the code that runs is the compiled form of exactly the definitions the statements name. The axiom print of every one of these eleven theorems, re-run read-only for this paper, is exactly `propext`, `Quot.sound` and its own native axiom; each of the five quantified corollaries inherits exactly the native axiom of its own sweep and nothing else; and the two bridge theorems `sweep_spec` and `sweepStar_spec`, which are what turn a Boolean into a quantified sentence, are proved by the kernel with axioms `propext` and `Quot.sound` only — no native evaluation enters the bridge. So the trust base is Lean's kernel together with the Lean compiler and the C toolchain behind it, plus the correctness of the definitions in `Engine.lean`: that `checkRepair` decides (1)–(4) of Definition 2.2 and recomputes the depth, and, for the lower bounds, that `existsRepairLe` really enumerates every valid repair (Section 2.4) — a property of the code as written, not a Lean theorem, and the reason for the controls of Proposition 3.2 and for the independent re-computations below. The greedy `buildRepair` is not trusted at all: it only proposes witnesses, and every witness is checked. The three errata theorems use no native evaluation; their axioms are `propext`, `Classical.choice`, `Quot.sound`. No theorem depends on `sorryAx`.

*Cost.* As recorded when the development was written, on a loaded twenty-core machine: the sweeps of Theorem 4.1 took 2 min 9 s ($t = 4, \ldots, 12$), 6 min 19 s ($t = 14, 16$), 7 min 58 s ($t = 18$), 10 min 34 s ($t = 13, 15, 17$) and 10 min 30 s ($t = 19$); `Model` 7.5 s, `Optimal` 12.1 s, `Errata` 4.1 s; peak resident set at most 2.2 GB (the errata module, which imports *Mathlib*); about 50 processor-minutes in all. The cost of a sweep grows like $t^6$; $t = 20$ is priced at about 15 minutes and was not run.

*Independent re-computations.* Three implementations independent of the Lean code were used. The C implementation of Section 8 (a greedy plus a checker, and a separate exhaustive minimiser) was written first and the Lean engine was cross-checked against it at every diameter; one disagreement during development was a defective C build that reported star repairs of depth $t + 1$, caught by the Lean check (`sweepStar` 4 5 evaluates to `false`), which is how the value $2t-1$ of Theorem 6.1 was found. Two further implementations, both in Python, both written from the source's printed definitions and independently of each other and of the C and Lean code (in both, the wraparound reduces modulo the $\mathbb{Z}$-module $(\alpha)$ by search rather than by a stored list of short vectors), re-derived: the errata arithmetic for $1 \le t \le 12$, including the injectivity on $H_t$ of both corrected labels and of the printed one, and the fact that $\rho$ is the graph distance from $s$, so that the diameter is $t$; the source's Examples 4 and 6; at $t = 3$, the exact minimum $\mathrm{D}(F)$ over all valid repairs and all orientations for all 667 fault sets, finding exactly the six placements of Theorem 5.1 with $\mathrm{D}(F) = 5$, confirming that they are the adjacent layer-1 pairs, and $\mathrm{D}(F) \le 4$ for the other 661; at $t = 4, \ldots, 19$, $\mathrm{D}(F_t) = t + 1$ with $\mathrm{D}_\theta(F_t) = t + 1$ for every one of the fifteen $\theta$ — Theorem 4.2 certifies $4 \le t \le 18$ and the value at $t = 19$ is recorded here as a computation only, so that outside the formal development $\max_{|F|\le 2} \mathrm{D}(F) = t + 1$ at $t = 19$ too — and, at $t = 4, \ldots, 8$, $\mathrm{D}(\{(1-t, t-1)\}) = t+1$ and $\mathrm{D}(\{(-t, 0)\}) = t$; the full enumerations with exact minima at $t = 4$ and $t = 5$ reported in Section 8, whose maxima are $t + 1$; at $t = 4, 5, 6$, the star optimum in full, $\mathrm{D}^\star(R) = 2t - 1$ for the placement $R$ of Theorem 6.1 and $\max_{|F|\le 2} \mathrm{D}^\star(F) = 2t - 1$ attained at $R$; and, for $2 \le t \le 40$, that every wrapped neighbour of a boundary vertex is again a boundary vertex, so that over that range the two readings of the parent rule give the same fifteen trees (Remark 2.1) and the mechanism of Remark 4.4 applies. Every value agreed with the Lean side. The development is currently in a private repository: repository reference to be added at submission.

## Appendix A. The formal statements

The Lean statements corresponding to the results above, verbatim (statements only; proofs, docstrings and attributes omitted; the namespace `EjMoemDepth` is omitted). Here `mkNet t` builds $\Gamma_t$, `mkCtx t` builds $\Gamma_t$ with its fifteen trees (`.pars`, in the order of the source's Table 1), `faultSets` enumerates the fault sets with $|F| \le 2$ as index pairs (the value `n` in a slot meaning "no fault"), `repairDepth net p star f g` runs the construction for orientation tree `p` and faults `f`, `g` and returns `some d` exactly when the checker accepts the result with depth `d`, `caseOK c star k F` says some orientation's constructed repair is accepted with depth $\le$ `k`, and `someOrientationRepairLe c star k F` runs the exhaustive search at bound `k` over all orientations. `vertexAt net x y` is the index of the axial point $(x, y)$.

**The bridge (Section 2.4).**

```
def sweep (t k : Nat) : Bool :=
  let c := mkCtx t
  (faultSets c.net).all (caseOK c false k)

theorem sweep_spec {t k : Nat} (h : sweep t k = true) :
    ∀ F ∈ faultSets (mkCtx t).net, ∃ p ∈ (mkCtx t).pars,
      ∃ d, repairDepth (mkCtx t).net p false F.1 F.2 = some d ∧ d ≤ k
```

**Theorem 4.1.**

```
theorem sweep_low : ([4,5,6,7,8,9,10,11,12] : List Nat).all (fun t => sweep t (t + 1)) = true
theorem sweep_low_spec {t : Nat} (ht : t ∈ ([4,5,6,7,8,9,10,11,12] : List Nat)) :
    ∀ F ∈ faultSets (mkCtx t).net, ∃ p ∈ (mkCtx t).pars,
      ∃ d, repairDepth (mkCtx t).net p false F.1 F.2 = some d ∧ d ≤ t + 1
theorem sweep_hi_a : ([14,16] : List Nat).all (fun t => sweep t (t + 1)) = true
theorem sweep_hi_b : ([18] : List Nat).all (fun t => sweep t (t + 1)) = true
theorem sweep_gap : ([13,15,17] : List Nat).all (fun t => sweep t (t + 1)) = true
theorem sweep_top : sweep 19 20 = true
theorem sweep_top_spec :
    ∀ F ∈ faultSets (mkCtx 19).net, ∃ p ∈ (mkCtx 19).pars,
      ∃ d, repairDepth (mkCtx 19).net p false F.1 F.2 = some d ∧ d ≤ 20
```

(`sweep_hi_a_spec`, `sweep_hi_b_spec` and `sweep_gap_spec` have the same shape as `sweep_low_spec` over their own lists.)

**Theorem 4.2.**

```
def witPair (t : Nat) : Nat × Nat :=
  let net := mkNet t
  let a := vertexAt net (-(Int.ofNat t)) 0
  let b := vertexAt net (1 - Int.ofNat t) (Int.ofNat t - 1)
  (min a b, max a b)

theorem t_add_one_optimal :
    ([4,5,6,7,8,9,10,11,12,13,14,15,16,17,18] : List Nat).all
      (fun t => !someOrientationRepairLe (mkCtx t) false t (witPair t) &&
                caseOK (mkCtx t) false (t + 1) (witPair t)) = true
```

**Theorem 5.1.**

```
def exc3coords : List ((Int × Int) × (Int × Int)) :=
  [((-1, 0), (-1, 1)), ((-1, 0), (0, -1)), ((-1, 1), (0, 1)),
   ((0, -1), (1, -1)), ((0, 1), (1, 0)), ((1, -1), (1, 0))]

theorem t3_exceptional_set :
    exc3.length = 6 ∧ (ringAdjPairs 3).length = 6 ∧
    exc3.all (fun F => (ringAdjPairs 3).contains F) = true ∧
    (ringAdjPairs 3).all (fun F => exc3.contains F) = true ∧
    exc3.all (fun F => (faultSets (mkNet 3)).contains F) = true ∧
    exc3.all (fun F => !someOrientationRepairLe (mkCtx 3) false 4 F) = true ∧
    exc3.all (fun F => caseOK (mkCtx 3) false 5 F) = true ∧
    sweepExcept 3 4 exc3 = true
```

(`exc3` is `exc3coords` converted to canonical index pairs; `ringAdjPairs t` lists the pairs $u < v$ of adjacent layer-1 vertices; `sweepExcept t k skip` is the sweep over the fault sets not in `skip`.)

**Theorem 6.1.**

```
theorem star_necessary :
    ([4,5,6,7,8] : List Nat).all
      (fun t => sweepStar t (2 * t - 1) &&
                !someOrientationRepairLe (mkCtx t) true (2 * t - 2) (ringPair t) &&
                someOrientationRepairLe (mkCtx t) true (2 * t - 1) (ringPair t)) = true
```

(`ringPair t` is the canonical index pair of $\{(-1,0),(-1,1)\}$.)

**Propositions 3.1 and 3.2.**

```
theorem model_sane :
    ([2,3,4,5,6,7,8,9,10,11,12] : List Nat).all
      (fun t => netSane t && labelInjective t && treesSane t && faultSetsSane t) = true

theorem controls :
    example4 = some 4 ∧ example6 = some 4 ∧
    broken 0 = none ∧ broken 1 = none ∧ broken 2 = none ∧ broken 3 = none

theorem t3_not_t_add_one : sweep 3 4 = false ∧ sweep 3 5 = true
```

**Propositions 7.1 and 7.2.**

```
theorem alpha_norm (t : ℤ) :
    ((t + 1) ^ 2 + (t + 1) * t + t ^ 2 = 3 * t ^ 2 + 3 * t + 1) ∧
    ((t + 1) ^ 2 - (t + 1) * t + t ^ 2 = t ^ 2 + t + 1) ∧
    (1 ≤ t → (t + 1) ^ 2 - (t + 1) * t + t ^ 2 < 3 * t ^ 2 + 3 * t + 1)

theorem label_erratum (t : ℤ) :
    (t * (t + 1) + (2 * t + 1) * t = 3 * t ^ 2 + 2 * t) ∧
    (1 ≤ t → 0 < 3 * t ^ 2 + 2 * t ∧ 3 * t ^ 2 + 2 * t < 3 * t ^ 2 + 3 * t + 1) ∧
    ((t + 1) * (t + 1) + (2 * t + 1) * t = 3 * t ^ 2 + 3 * t + 1) ∧
    ((t + 1) * (-t) + (2 * t + 1) * (2 * t + 1) = 3 * t ^ 2 + 3 * t + 1) ∧
    (t * (t + 1) + (-(t + 1)) * t = 0) ∧
    (t * (-t) + (-(t + 1)) * (2 * t + 1) = -(3 * t ^ 2 + 3 * t + 1))

theorem label_erratum_example :
    (3 : ℤ) * 4 + 7 * 3 = 33 ∧ (33 : ℤ) % 37 ≠ 0 ∧
    ((3 : ℤ) * 4 + 33 * 3) % 37 = 0 ∧ ((3 : ℤ) * (-3) + 33 * 7) % 37 = 0 ∧
    ((4 : ℤ) * 4 + 7 * 3) % 37 = 0 ∧ ((4 : ℤ) * (-3) + 7 * 7) % 37 = 0
```

The two polynomial identities quoted after Proposition 7.1 and the identity for $\alpha\omega$ are the untagged lemmas `norm_sixth_root`, `norm_cube_root` and `alpha_mul_omega` of the same module, each closed by `ring`.

## References


[1] B. A. Albader, *Multi-Orientation Edge-Minimum Repair for Non-Redundant Fault-Tolerant Broadcasting in Dense Eisenstein–Jacobi Networks*, arXiv:2606.19834v1 [cs.DC], submitted 18 June 2026. Read in full from the version-1 e-print (a single `main.tex` and four figures), the only version as of 2026-09-07 (live abstract page checked on that date; primary category cs.DC read off the page). Statement, table, example, section and equation numbers are those of the PDF compiled with `tectonic` from the unmodified e-print (19 pages) and of its `.aux` file: equations (1)–(3), (5), (9), (10), (15), (16); Lemma 1, Lemma 2, Theorem 1, Lemma 3, Lemma 9, Lemma 11, Theorem 3; Algorithms 1 and 2; Examples 1, 2, 4, 6; Tables 1, 4, 5; Sections 3, 3.1, 7.3, 9. The quotations in Section 1 are verbatim from that PDF.

[2] B. A. Albader, *Constant-Time Certificate Selection for Local Broadcast Repair in Dense Gaussian and Eisenstein–Jacobi Networks*, arXiv:2606.21132v1 [cs.DC], 19 June 2026. Read from the PDF compiled with `tectonic` (27 pages) for its statement of the EJ bounds, the range of its exhaustive validation ($t = 2, \ldots, 8$, 52,395 cases, which is exactly the number of one- and two-fault placements at those diameters), and its Section 3.2 ("Dense Eisenstein–Jacobi Networks"), whose equation (10) repeats the vertex label of Proposition 7.2.

[3] B. A. Albader, *Re-Rooting-Assisted Edge-Minimum Runtime Repair for Node and Link Failures in Dense Eisenstein–Jacobi Broadcast Networks*, arXiv:2606.21133v1 [cs.DC], 19 June 2026. Read from the e-print compiled with `tectonic` (29 pages): its equations (2) and (8) and the caption of its Figure 1 (all in its Section 3, "Dense EJ Network Model"), its Lemma 4, and the sentence quoted in Section 1.

[4] C. Martínez, E. Stafford, R. Beivide and E. M. Gabidulin, *Modeling hexagonal constellations with Eisenstein–Jacobi graphs*, Problems of Information Transmission **44** (2008), no. 1, 1–11. Bibliographic data checked against Crossref (doi:10.1134/S0032946008010018); the network family. Not read for this work.

[5] M. Flahive and B. Bose, *The topology of Gaussian and Eisenstein–Jacobi interconnection networks*, IEEE Transactions on Parallel and Distributed Systems **21** (2010), no. 8, 1132–1142. Bibliographic data checked against Crossref (doi:10.1109/TPDS.2009.132); the density of the ball $H_t$. Not read for this work.

[6] A. Thomson and S. Zhou, *Frobenius circulant graphs of valency six, Eisenstein–Jacobi networks, and hexagonal meshes*, European Journal of Combinatorics **38** (2014), 61–78. Bibliographic data checked against Crossref (doi:10.1016/j.ejc.2013.11.004). Cited as the main non-Albader occurrence of the phrase in the corpus sweep; not read for this work.

[7] Z. Hussain, B. Bose and A. Al-Dhelaan, *Edge disjoint Hamiltonian cycles in Eisenstein–Jacobi networks*, Journal of Parallel and Distributed Computing **86** (2015), 62–70. Bibliographic data checked against Crossref (doi:10.1016/j.jpdc.2015.08.003); not read for this work.

[8] Z. Hussain, H. AboElFotoh and B. AlBdaiwi, *Independent Spanning Trees in Eisenstein–Jacobi Networks*, arXiv:2101.09797 [cs.DC], 2021. Title, authors and category checked against the arXiv API; the source's reference [8]; not read for this work.

[9] K. Huber, *Codes over Eisenstein–Jacobi integers*, in: Finite Fields: Theory, Applications, and Algorithms, Contemporary Mathematics **168**, American Mathematical Society, 1994, pp. 165–179. Bibliographic data checked against Crossref (doi:10.1090/conm/168/01696). Cited only to name the coding-theoretic use of the phrase that dominates the corpus sweep; not read for this work.

[10] L. de Moura and S. Ullrich, *The Lean 4 theorem prover and programming language*, in: Automated Deduction – CADE 28, Lecture Notes in Computer Science **12699**, Springer, 2021, pp. 625–635.

[11] The mathlib Community, *The Lean mathematical library*, in: Proceedings of the 9th ACM SIGPLAN International Conference on Certified Programs and Proofs (CPP 2020), ACM, 2020, pp. 367–381.