CERTIFIED RESOLVING AT EVERY SEAT: CONSTRUCTIVE SAFETY CERTIFICATES FOR MULTIPLAYER SUBGAME RESOLVING

Anonymous authors
Paper under double-blind review

ABSTRACT

Exact solving of an imperfect-information game does not scale, so the standard recipe computes a blueprint offline by regret minimisation and recomputes the strategy inside the subgame play actually reaches. In two-player zero-sum games a gadget makes resolving safe: it lets the opponent cash in, at every entry, the counterfactual value it was guaranteed under the blueprint, so any equilibrium of it refines the blueprint without raising its exploitability. That guarantee rests on the zero-sum identity, which is gone as soon as a third seat is added: an unguarded update can leave the profile more exploitable than the blueprint. What survives is a per-player charge against the blueprint, anchored at the opponents' subgame best-response values. The charge is a maximum of affine functions of the resolver's sequence-form strategy, so the bound on the profile's Nash gap, the largest best-response gain over the players, is convex and piecewise linear; minimising it or maximising the resolver's own payoff under a Nash-gap budget is a linear program. The minimum-certificate program is the value of a two-player zero-sum gadget whose meta-adversary picks the victim, solvable by counterfactual regret minimisation. This gives a subgame-local resolver for independent strategies in multiplayer games whose every update carries an explicit certificate on the Nash gap of the whole profile. In constant-sum games the certified gain is at most $(n-1)\tau$ plus the opponents' Nash-gap slack minus the shift of their best-response values, and three-player Kuhn instances attain it to six decimal places. A two-sided variant equals the exact safety constraint whenever no opponent acts before the subgame. Every seat can run the same program, and a ledger of certified gains keeps the joint bound monotone under round-robin certified resolving. Across 3-player Kuhn, 3-player Leduc, and random games with up to six players, $2{,}244$ certified updates never exceed their certificate beyond the solver tolerance of 10^{-9}; re-anchoring cuts the Nash gap by 41–57% on 3-player Kuhn. Unsafe resolving raises the Nash gap in 18–70% of configurations, by up to 0.17, while under the common clipped budget the certified gain matches or beats the unsafe gain in 176 of the 214 poker configurations. Safety becomes an objective a resolver can optimise against, priced by the cap identity and the gap decomposition.

1 INTRODUCTION

Solving an imperfect-information extensive-form game means finding a profile whose best-response gains are small, and computing one exactly does not scale to the games that matter. The established recipe splits the problem in two (Burch et al., 2014). Offline, counterfactual regret minimisation and its sampling variants compute a blueprint over an abstraction of the whole game (Zinkevich et al., 2007; Lanctot et al., 2009; Tammelin et al., 2015). Online, when play reaches a public subgame that the abstraction resolves badly, the agent recomputes its strategy inside that subgame alone and keeps the blueprint elsewhere. Subgame resolving, the online half of this recipe, is what carries DeepStack, Libratus and Student of Games (Moravčík et al., 2017; Brown & Sandholm, 2018; Schmid et al., 2023).

In two-player zero-sum games resolving is safe. The gadget of Burch et al. (2014) lets the opponent stop at any entry and collect the counterfactual value it was guaranteed under the blueprint, so any

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

equilibrium of the gadget refines the blueprint without increasing its exploitability; max-margin refining (Moravcik et al., 2016) and reach-aware and nested variants (Brown & Sandholm, 2017; Brown et al., 2018) carry that guarantee through repeated resolves and depth-limited lookahead. Each of them rests on one identity: a zero-sum game makes the two seats' gains cancel, so bounding one bounds the other. Once a third player sits at the table the identity fails: even when the total payoff is conserved, an opponent's gain can come out of another opponent's payoff instead of the resolver's. Three-player Kuhn already admits a family of equilibrium profiles that shift payoff between seats (Szafron et al., 2013), and Pluribus, the six-player agent, resolves with no guarantee at all (Brown & Sandholm, 2019). Extending the two-player gadget under-charges the several opponents it must price at once (Section 7).

What survives beyond two players is a post-hoc bound on a single update. A unilateral update by one seat inside a public subgame raises the profile's Nash gap, the largest best-response gain over the players, by at most the larger of two terms: the seat's own payoff loss, and, over the opponents, the rise of that opponent's best-response value inside the subgame, net of its payoff change. It certifies an update after delivery and says nothing about how to produce one with a small certificate, so the natural question is whether a resolver can target the certificate directly.

It can, and the reason is structural. Once the margins are anchored at the opponents' subgame best-response values against the blueprint, each deficit is a maximum of affine functions of the resolver's sequence-form strategy, so the certificate, and the form of it that reads the blueprint's own gains, are convex and piecewise linear. Minimising the certificate, or maximising the resolver's own payoff under a Nash-gap budget, is a linear program. The minimum-certificate program is also the value of a two-player zero-sum game: a meta-adversary picks one opponent to impersonate and plays that opponent's terminate-or-follow gadget, while the resolver plays one strategy shared across all branches, which puts the gadget within reach of regret minimisation. The program reads only the blueprint and public information, so the seats can run it in turn, each against the profile the seats before it produced, and the per-seat certificates telescope into a joint bound.

We evaluate with full-tree best responses on 3-player Kuhn, 3-player Leduc and stage games with three to six players, 374 configurations over six budgets. Unsafe resolving raises the Nash gap in 18–70% of them, by up to 0.17; the $2{,}244$ certified updates never exceed their certificate beyond the solver tolerance of 10^{-9}; under the budget the unsafe update itself induces, the certified gain matches or beats it in 176 of the 214 poker configurations; and re-anchored round-robin resolving cuts the Nash gap by 41–57% on 3-player Kuhn.

Our contributions are three:

- **We turn the safety certificate into an optimisation target.** We prove that, anchored at the opponents' subgame best-response values against the blueprint, the certificate is convex and piecewise linear in the resolver's sequence-form strategy (Proposition 2). Minimising it, or maximising the resolver's payoff under a budget on it, is a linear program whose every constraint lives inside the subgame. We further show that the minimum-certificate program is the value of a two-player zero-sum gadget, so that counterfactual regret minimisation solves it, and that finitely many penalised solves of the same gadget recover the budgeted program (Section 4).
- **We price certified safety exactly.** We prove a cap identity for constant-sum games: a τ-safe update by one seat gains at most $(n-1)\tau$ plus the opponents' slack minus the shift of their best-response values, with equality exactly when every opponent sits on the cap (Theorem 2). We show that a two-sided certificate coincides with the exact whole-game safety constraint whenever no opponent acts before the subgame (Corollary 1), and we decompose what the one-sided certificate gives up otherwise into three non-negative terms, clipping, reach and re-optimisation (Proposition 4).
- **We certify every seat at once.** We show that seats running the same program in turn produce one bound on the whole profile, that a ledger keeps it monotone under round-robin resolving, and that whenever the passing rule stops the dynamics it stops at a certificate-stable profile that need not be an equilibrium (Section 6). We find, on three game families under exact best responses, that no certificate is ever violated, that re-anchored round-robin resolving lowers the Nash gap where unsafe resolving raises it, and that under a common budget the certified update gives up little of the unsafe gain (Section 7).

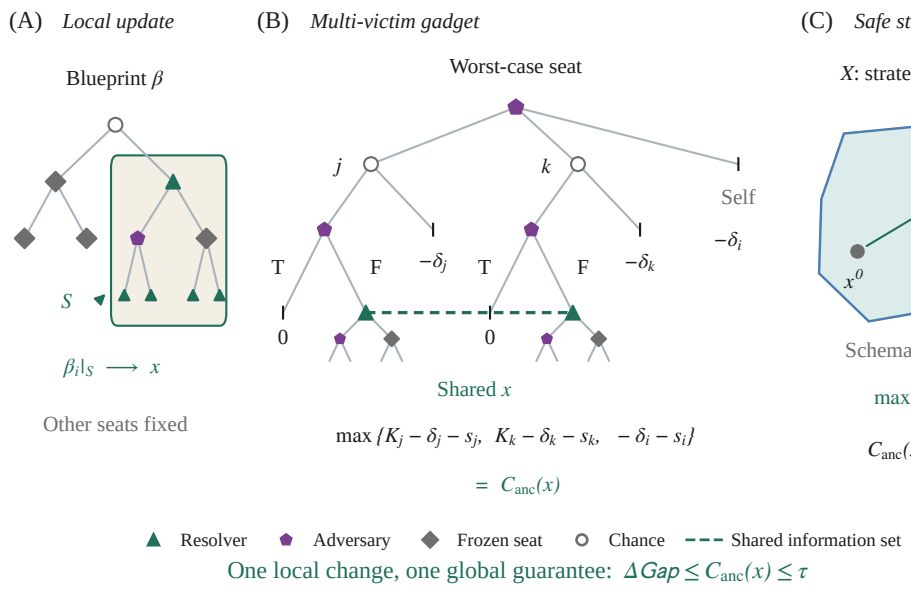


Figure 1: **One shared strategy, every seat protected.** **(A)** The resolver changes its strategy inside the public subgame S and nowhere else. **(B)** The meta-adversary selects a victim or self. Victim branches combine a terminate-or-follow margin with a payoff-change charge; dashed links tie corresponding resolver information sets across copies. The worst branch is exactly the value of the certificate. **(C)** A slice of the sequence-form polytope shows the gain-maximisation problem under a ceiling on how far the Nash gap may rise; the cross marks an excluded candidate.

2 Related Work

Safe resolving in two-player zero-sum games. Counterfactual regret minimisation (Zinkevich et al., 2007; Lanctot et al., 2009; Tammelin et al., 2015) solves the blueprint, and resolving refines it. The gadget of Burch et al. (2014) lets the opponent terminate for the blueprint's counterfactual value, max-margin refining (Moravcik et al., 2016) maximises the smallest margin, and reach-aware and nested variants (Brown & Sandholm, 2017; Brown et al., 2018) carry the guarantee through repeated resolves and depth-limited lookahead. Related formulations resolve without a public-state cut (Zhang & Sandholm, 2021), do so domain-independently and continually (Šustr et al., 2019), use generative subgame solving on the earliest-reached nodes (Ge et al., 2023), limit the opponent's search (Liu et al., 2023).

Each of these bounds the resolver's exploitability, a two-player best-response value (Johanson et al., 2011). Our gadget keeps the terminate-or-follow structure and replaces the identity with a per-opponent anchored deficit and a payoff-change term.

Beyond two players. Endgame solving in multiplayer games (Ganzfried & Sandholm, 2015) and the search of Pluribus (Brown & Sandholm, 2019) carry no guarantee, and depth-limited value functions are harder to define away from two players (Kovařík et al., 2023). The earlier multiplayer line made no safety claim either: regret minimisation (Gibson & Szafron, 2011a; Burch et al., 2012), stitching across subgames (Gibson & Szafron, 2011b), and stochastic-game equilibria (Ganzfried & Sandholm, 2009). Grafting separately solved strategies preserves no such guarantee (Waugh et al., 2009).

Budgets and weaker safety concepts. Robust counter-strategies (Johanson et al., 2007), their data-biased variants (Johanson & Bowling, 2009), opponent modelling (Ganzfried & Sandholm, 2011), safe opponent exploitation (Ganzfried & Sandholm, 2012), adaptation safety (Ge et al., 2024) and safe exploitation search (Liu et al., 2022) guard one player's own payoff against an opponent who switches strategy. Our object is the whole profile's Nash gap under exact best responses of every seat, allowed to rise by at most one budget τ shared by every victim.

Lossy abstraction carries explicit bounds (Kroer & Sandholm, 2018; Johanson et al., 2013), and Šustr et al. (2021) call an output sound when it provably matches its advertised guarantee. Constrained regret minimisation (Davis et al., 2019) and regret circuits (Farina et al., 2019) supply the solver for a convex constraint on one seat's sequence form; the anchored certificate is that constraint for the whole profile, and it is checked from inside S.

Guarantees for multiplayer resolving otherwise attach to a different object than the Nash gap of an independent profile: a mediator (Farina et al., 2018; Celli et al., 2019; Zhang et al., 2021) or a team maximiser (Von Stengel & Koller, 1997; Basilico et al., 2017), refined safely by Ling & Fang (2022) and Zhang et al. (2022). We keep independent strategies and the Nash gap, so our guarantee is a bound with an explicit price.

Certificates and dynamics. Certificates for very large games (Zhang & Sandholm, 2020) and the self-play bounds of MacQueen & Wright (2023) for near constant-sum polymatrix games share our currency, best-response gains, but certify a profile; ours certifies an update, tightly enough to be optimised. We measure with exact best responses, so the numbers of Section 7 carry no estimation error against the certificate; learned (Timbers et al., 2022) and sampled (Ponsen et al., 2011) estimates would add one.

Exploitability descent (Lockhart et al., 2019), carried to multiplayer games by projection (Ganzfried, 2026), lowers a best-response measure by gradient steps over the whole game; the minimum-certificate resolve lowers an upper bound on the same measure by one program confined to S. Exact multiplayer equilibrium is computed globally (Zhang et al., 2023). No-external-regret dynamics reach coarse correlated, and with swap regret correlated, but not Nash equilibria in general games (Hart & Mas-Colell, 2000; Daskalakis et al., 2021; Farina et al., 2022); gradient and best-response dynamics cycle (Balduzzi et al., 2018; Heinrich et al., 2023), and CFR variants that drop Nash as the objective play better (Ju et al., 2025), while last-iterate convergence to an exact equilibrium is known only in the limit and under hypomonotonicity (Lu et al., 2025). Our round-robin dynamics keeps a bound monotone, and whenever it stops before the step cap it stops at an ε-certificate-stable profile (Observation 1).

3 Preliminaries and the Anchored Certificate

Games, sequence form, Nash gap. An n-player perfect-recall extensive-form game has terminal payoffs $u_p(z)$, chance moves and information sets. For a profile σ, write $u_p(\sigma)$ for player p's expected payoff, $\mathrm{BRV}_p(\sigma) = \max_{y_p} u_p(y_p, \sigma_{-p})$ for its best-response value, $r_p(\sigma) = \mathrm{BRV}_p(\sigma) - u_p(\sigma)$ for its best-response gain, and $\mathsf{Gap}(\sigma) = \max_p A_p(\sigma)$ for the profile's Nash gap, the smallest ε for which σ is an ε-Nash equilibrium. At $n = 2$ in a zero-sum game $r_1 + r_2$ is the profile's NashConv, and two-player safe resolving bounds the resolver's own exploitability, hence the sum; it need not preserve the maximum (Appendix A).

A player's behaviour is represented by realisation weights, one per sequence, with the empty sequence fixed to one and flow conservation at every information set (Koller et al., 1996); every payoff is multilinear in them. Write X for seat i's sequence-form polytope inside S and $Y_j(\eta)$ for opponent j's below a group η.

A public subgame S is the set of histories below a public state, and its entries e are the earliest histories inside S. Each entry carries a chance weight c_e and a reach probability $\pi_p(e)$, the product of p's own action probabilities on the path to e. Write $[x]_+ = \max(x, 0)$ and $[x]_- = \max(-x, 0)$ for the positive and negative parts.

Resolving. A blueprint β is given. Seat i resolves S by replacing its strategy inside S with some $x \in X$, and the resolved profile β' agrees with β elsewhere. Write $\delta_p = u_p(\beta') - u_p(\beta)$ for the realised payoff change and $s_j = \mathsf{Gap}(\beta) - r_j(\beta) \geq 0$ for opponent j's slack, the room it has before the profile cap.

For an opponent j the entries of S are sorted into entry groups η by the information j holds at the entry: its private observations and the information sets and actions it has passed, and each group has weight $\omega_\eta = \sum_{e \in \eta} c_e \prod_{q \neq j} \pi_q(e)$. Since $\pi_j(e)$ is a product over exactly those information sets and actions, j's own reach is the same at every entry of a group under every strategy of j; and since j

observes the public state and has perfect recall, every information set of j inside S descends from the entries of a single group and none straddles the boundary of S.

Anchor and anchored deficit. Let $\mathrm{CBV}_j(\eta;\sigma) = \max_{y_j} v_j(\eta; y_j, \sigma_{-j})$ be j's best-response counterfactual value at η against σ_{-j} inside S, where $v_j(\eta;\sigma)$ is j's expected payoff below η when play enters with the entry distribution $c_e \prod_{q\neq j} \pi_q(e)/\omega_\eta$, so the weight ω_η sits outside; groups with $\omega_\eta = 0$ are omitted throughout, from every sum and every program. Anchor j at its value against the blueprint, $\mathrm{CBV}^0_j(\eta) = \mathrm{CBV}_j(\eta;\beta)$.

Definition 1 (Anchored deficit)**.** $K_j(\beta') = \sum_\eta \omega_\eta \left[\mathrm{CBV}_j(\eta;\beta') - \mathrm{CBV}^0_j(\eta)\right]_+$.

The deficit is zero at the blueprint, and it charges j only where the resolve raised j's best continuation value above its anchor. Two-player resolving anchors at the blueprint's counterfactual value $\mathrm{CFV}^0_j(\eta)$ instead (Burch et al., 2014). Since $\mathrm{CBV}^0_j \geq \mathrm{CFV}^0_j$, our deficit is never larger, and the anchor is what makes the blueprint certifiable (Lemma 1).

Lemma 1 (Anchoring)**.** *For every opponent j,* $\mathrm{BRV}_j(\beta') - \mathrm{BRV}_j(\beta) \leq K_j(\beta')$.

The proof is in Appendix A.

Write $D_j = \mathrm{BRV}_j(\beta') - \mathrm{BRV}_j(\beta)$ for the realised shift of j's best-response value. Lemma 1 says $D_j \leq K_j$, the identity $r_j(\beta') = r_j(\beta) + D_j - \delta_j$ holds for every opponent, and the resolver's own best-response value does not move, so $r_i(\beta') = r_i(\beta) - \delta_i$.

Proposition 1 (Anchored certificate)**.** *For any unilateral resolve β' of β by seat i in S,*

$$\begin{aligned}\mathsf{Gap}(\beta') - \mathsf{Gap}(\beta) \;&\leq\; C_{\mathrm{anc}}(\beta') := \max\Big(r_i(\beta) - \delta_i,\; \max_{j\neq i}\big[r_j(\beta) + K_j - \delta_j\big]\Big) - \mathsf{Gap}(\beta)\\ &\leq\; C_{\mathrm{raw}}(\beta') := \max\Big(-\delta_i,\; \max_{j\neq i}[K_j - \delta_j]\Big).\end{aligned}$$

Proof. $r_i(\beta') = r_i(\beta) - \delta_i$ and $r_j(\beta') \leq r_j(\beta) + K_j - \delta_j$ by Lemma 1; take the maximum over players and subtract $\mathsf{Gap}(\beta)$. The second inequality drops $r_p(\beta) \leq \mathsf{Gap}(\beta)$. □

Both forms vanish at $x = \beta_i$, since then $\delta \equiv 0$ and $K_j \equiv 0$. An opponent's gain can rise because its best response improves, charged by K_j, or because its own payoff falls, charged by $-\delta_j$; C_{raw} charges both in full, while C_{anc} also reads the blueprint's gains $r_p(\beta)$, computed once, so a seat below the cap absorbs part of the charge and the online constraint is subgame-local once those full-game gains are cached. We say that β' is τ-*certified* if $C_{\mathrm{anc}}(\beta') \leq \tau$.

4 Certified Resolving: Linear Program and Gadget

The certificate is convex in the resolver's strategy. Fix β_{-i} and let x range over X, seat i's sequence-form polytope inside S. Every payoff is multilinear in the players' realisation weights, so with all seats but i frozen at β each $\delta_p(x)$ is affine in x. For a group η of opponent j, $\mathrm{CBV}_j(\eta; x) = \max_{y\in Y_j(\eta)} \langle x, U_\eta y\rangle$ is a maximum of affine functions of x over j's sequence-form polytope below η, where U_η pairs i's sequences and j's sequences by payoff, and a maximum of affine functions is convex. The positive part and the weighted sum preserve convexity.

Proposition 2 (Convexity)**.** *K_j, C_{raw} and C_{anc} are convex piecewise-linear functions of $x \in X$. Consequently*

$$\min_{x\in X} C_{\mathrm{anc}}(x) \qquad \textit{and} \qquad \max_{x\in X}\; u_i(x) \quad \textit{s.t.} \quad C_{\mathrm{anc}}(x) \leq \tau$$

are linear programs, feasible for every $\tau \geq 0$ since $x = \beta_i$ has $C_{\mathrm{anc}} = 0$.

Its variables are the resolver's sequences, one best-response value per opponent information set inside S and one slack per entry group; its constraints are the resolver's sequence-form equalities, one inequality per opponent sequence inside S and one per entry group, so every opponent dual is confined to S. We call the minimiser of the first program the *minimum-certificate resolve* and the maximiser of the second at budget τ the *τ-certified resolve*; $\tau = 0$ asks for the most profitable update that provably does not raise the Nash gap.

The multi-victim gadget. The minimum-certificate program is also the value of a two-player zero-sum game, which gives a regret-minimisation formulation for subgames beyond an LP solver. The resolver is player one and a meta-adversary is player two. At the root the adversary picks a branch in $\{\text{victim } j : j \neq i\} \cup \{\text{self}\}$, and the branch of victim j splits by a fair coin into a margin part and a payoff part, both with terminals doubled.

In the margin part chance draws a group η with probability ω_η/Ω_j, where $\Omega_j = \sum_\eta \omega_\eta$. The meta-adversary then chooses *terminate*, ending with payoff 0, or *follow*. Under follow, chance draws an entry of η with probability proportional to its weight, and play continues inside a copy of S where j acts freely, every seat other than i and j is frozen at β, and the resolver acts at its own information sets; the terminal payoff to the adversary is $\Omega_j\,(u_j(z) - \mathrm{CBV}_j^0(\eta))$.

In the payoff part chance draws an entry e with its absolute blueprint probability $P_0(e) = c_e \prod_p \pi_p^\beta(e)$ and pays the adversary the blueprint minus the realised payoff below e (Appendix A); the self branch does the same for i's payoff. Each branch adds the constant $r_j(\beta) - \mathsf{Gap}(\beta)$ or $r_i(\beta) - \mathsf{Gap}(\beta)$ to its terminals. The resolver's information sets keep their labels in every copy, so it plays one strategy x everywhere (Fig. 1B).

Theorem 1 (Gadget)**.** *The gadget is a two-player zero-sum perfect-recall extensive-form game, the meta-adversary's best-response value against any resolver strategy x equals $C_{\mathrm{anc}}(x)$, and its game value equals $\min_{x\in X} C_{\mathrm{anc}}(x)$. Replacing the constants by zero gives C_{raw}.*

Proof sketch. The root maximum realises the outer $\max$ over victims and self, terminate-or-follow realises $[\mathrm{CBV}_j(\eta; x) - \mathrm{CBV}_j^0(\eta)]_+$ because the follow subtree's value against x is j's best response below η, and the chance halves average the margin and payoff parts; perfect recall holds because the adversary observes its own branch, group and choice and the resolver's information sets keep their labels. Appendix A has the details. □

Counterfactual regret minimisation (Zinkevich et al., 2007) on the gadget converges to $\min_x C_{\mathrm{anc}}(x)$ at the usual rate (Appendix B). The budgeted program has the single constraint $C_{\mathrm{anc}}(x) \le \tau$, and its value is the lower envelope $\min_{\lambda\ge 0}[P(\lambda) + \lambda\tau]$ of a family of penalised gadget games $P(\lambda)$; Appendix A recovers a τ-certified resolve of largest gain from finitely many penalised solves (Proposition 7).

What localisation costs. With only seat i's strategy free, $\mathrm{BRV}_j(x) = \max_{y_j} u_j(x, y_j, \beta_{-ij})$ is a maximum of affine functions of x over the *whole* game. The program $\max_{x\in X} u_i(x)$ subject to $r_p(x) \le \mathsf{Gap}(\beta) + \tau$ for every p is therefore a linear program too, feasible at $x = \beta_i$ for every $\tau \ge 0$, and its best-response duals range over every information set of every opponent: on Leduc the median exact-safe program has 3,111 variables and 2,981 rows against 218 and 159 for the certified one, and its HiGHS call takes 15.3 ms against 4.7 (Appendix B). We call its value, less $u_i(\beta)$, the *exact-safe* gain.

The certificate replaces each full-game program by one confined to S, and the anchor CBV_j^0 is what keeps the replacement feasible at the blueprint. The replacement is one-sided: it charges a rise of CBV_j at a group and credits no fall, because the reach with which j's best response enters the group is decided outside S and may be zero.

When the replacement is exact. Write $d_\eta(x) = \mathrm{CBV}_j(\eta; x) - \mathrm{CBV}_j^0(\eta)$ for the shift of j's continuation value at η, and $\pi_j^{y^*}(\eta) \in [0, 1]$ for j's own reach to η under a best response y^* to β'_{-j}, the product of its action probabilities on the path, which S does not decide. Opponent j *acts before* S if some information set of j lies on a path from the root into S.

Definition 2 (Two-sided deficit)**.** $\widetilde{K}_j(\beta') = \sum_\eta \omega_\eta \left[\mathrm{CBV}_j(\eta; \beta') - \mathrm{CBV}_j^0(\eta)\right]$, the sum of Definition 1 without the positive part.

Proposition 3 (Exact shift)**.** *If opponent j does not act before S, then $D_j(\beta') = \widetilde{K}_j(\beta') \le K_j(\beta')$ for every unilateral resolve β', and $\widetilde{K}_j$ is a convex piecewise-linear function of $x \in X$. Consequently the constraint $r_j(\beta) + \widetilde{K}_j(x) - \delta_j(x) \le \mathsf{Gap}(\beta) + \tau$ is the exact-safe constraint $r_j(x) \le \mathsf{Gap}(\beta) + \tau$, and Proposition 1 holds with $\widetilde{K}_j$ in place of K_j for every such j.*

Corollary 1 (Zero localisation gap)**.** *If no opponent acts before S, the two-sided certified program $\max_{x\in X} u_i(x)$ s.t. $\max\big(r_i(\beta) - \delta_i, \max_{j\neq i}[r_j(\beta) + \widetilde{K}_j - \delta_j]\big) \le \mathsf{Gap}(\beta) + \tau$ has the same feasible*

set and the same value as the exact-safe program, while its dual variables range over the opponents' information sets inside S only.

The stage games of Section 7 and every poker subgame entered before an opponent has acted satisfy the hypothesis.

What the positive part pays for.

Proposition 4 (Gap decomposition)**.** *Fix an opponent j, a best response y^* of j to β'_{-j}, and let y' agree with y^* outside S and best-respond to β_{-j} inside. Then $K_j(\beta') - D_j(\beta')$ is the sum of the clipping term $\sum_\eta \pi_j^{y^*}(\eta)\omega_\eta[d_\eta]_-$, the reach term $\sum_\eta(1-\pi_j^{y^*}(\eta))\omega_\eta[d_\eta]_+$ and the re-optimisation term $\mathrm{BRV}_j(\beta) - u_j(y', \beta_{-j})$, each non-negative; the last two vanish if j does not act before S.*

Clipping is the part of j's continuation value that the resolve lowers and the positive part ignores. Reach is the part it raises on entries that its best response never visits. Re-optimisation is what j loses against the blueprint when, outside S, it keeps the policy of its best response to the resolve (Table 1).

Table 1: The three resolving programs. *Certified*: the bound is computed inside the subgame. *Exact-safe*: the feasible set is the exact whole-game constraint.

Program	Certified?	Equals exact-safe?	Where used
One-sided local (Proposition 2)	always	no (Proposition 8)	Tables 2 and 6
Two-sided/one-sided hybrid (Proposition 3)	always	when no victim acts before S	Section 7
Whole-game exact-safe	always	by definition	oracle in Section 7

5 The Price of Certified Safety

In a two-player zero-sum game safe resolving never raises the opponent's best-response value, $D_j \le 0$, so the resolved strategy is no more exploitable than the blueprint (Burch et al., 2014). At three or more seats a certified update has a price, and constant-sum games pin that price down exactly.

Theorem 2 (Cap identity)**.** *Let the game be constant-sum, $\sum_p u_p \equiv c$. Any unilateral resolve β' of β by seat i with $\mathsf{Gap}(\beta') \le \mathsf{Gap}(\beta) + \tau$ satisfies*

$$\delta_i \;\le\; (n-1)\tau + \sum_{j\neq i} s_j - \sum_{j\neq i} D_j,$$

with equality if and only if every opponent sits on the cap, $r_j(\beta') = \mathsf{Gap}(\beta) + \tau$ for all $j \neq i$.

Proof. For each opponent, $r_j(\beta') = r_j(\beta) + D_j - \delta_j \le \mathsf{Gap}(\beta) + \tau$ gives $\delta_j \ge D_j - s_j - \tau$. Sum over $j \neq i$ and use $\delta_i = -\sum_{j\neq i}\delta_j$. □

The bound is an accounting identity over the cap constraints, valid for any τ-safe update, certified or not.

Corollary 2 (Certified cap)**.** *A τ-certified resolve satisfies $\delta_i \le (n-1)\tau + \sum_{j\neq i} s_j - \sum_{j\neq i} K_j$, by the proof of Theorem 2 with the certified constraint $r_j(\beta) + K_j - \delta_j \le \mathsf{Gap}(\beta) + \tau$ in place of the cap. The gap to Theorem 2 is $\sum_{j\neq i}(K_j - D_j) \ge 0$, the slack of the certificate.*

Corollary 3 (Homogeneous opponents)**.** *If $s_j = 0$ and $D_j \ge 0$ for every opponent, then $\delta_i \le (n-1)\tau$; at $\tau = 0$ no update with a positive gain is τ-safe.*

Both bounds are attained. In the five Kuhn configurations of Table 6 the τ-certified resolve meets Corollary 2 to 10^{-6}. In four of them the whole certificate slack $K_j - D_j$ is the clipping term of Proposition 4. In the fifth, $(T{=}1000, resolver2)$, all three terms vanish: both victims reach the cap and Theorem 2 holds with equality.

6 Every Seat: Joint Updates and Certified Dynamics

Joint updates and a ledger. Proposition 1 bounds one unilateral update. Deploy one agent at every seat, all running the same program on the same public subgame. The joint update is certifiable as long

as the program is deterministic and depends only on the blueprint, the public state and a fixed seat order: seat k can then reproduce every earlier resolve, so the profile it faces is common knowledge and Proposition 1 applies step by step.

Proposition 5 (Joint certificate)**.** *Let seats resolve S in a fixed order, seat k resolving against the profile produced by the seats before it, each step t being τ_t-certified with certificate B_t computed against the current profile. Then* $\mathsf{Gap}(\beta^{(n)}) - \mathsf{Gap}(\beta) \le \sum_t \max(B_t, 0)$.

The proof is the telescoping sum of Proposition 1. If every seat instead solves its own program against the original blueprint and the updates are spliced together at once, no bound applies (Appendix B).

Repeated certified resolving needs the gains r_p of the current profile, and recomputing them costs one full best response per player. Proposition 1 supplies a cheaper object: a ledger $\bar{r}^t \in \mathbb{R}^n$ with $\bar{r}^0 = r(\beta)$ that updates as $\bar{r}_i^{t+1} = \bar{r}_i^t - \delta_i$ and $\bar{r}_j^{t+1} = \bar{r}_j^t + K_j - \delta_j$ for $j \neq i$, after seat i resolves at step t with realised changes δ and deficits K.

Proposition 6 (Monotone potential)**.** $r_p(\beta^t) \le \bar{r}_p^t$ *for every p and t. If every step is* 0*-certified against the ledger, that is* $\max_p \bar{r}_p^{t+1} \le \max_p \bar{r}_p^t$*, then* $\Phi_t = \max_p \bar{r}_p^t$ *is non-increasing and* $\mathsf{Gap}(\beta^t) \le \Phi_t$.

The first claim is Lemma 1 applied inductively, the second the definition of a certified step.

Saturation, re-anchoring, equalisation. The ledger is an upper bound, and the resolve exploits it: the entry of every binding victim reaches the cap while its true gain stays below it.

Re-anchoring and frugal resolving remove the saturation. *Re-anchoring* resets $\bar{r}^t = r(\beta^t)$ at the start of every round, at the cost of n best responses. Then $\Phi_t = \mathsf{Gap}(\beta^t)$ at every round start, so by Proposition 6 no step of a round raises the true Nash gap above its opening value; the Nash gap falls from 0.037 to 0.016–0.022 within at most 60 steps in all six seat orders. *Frugal* resolving shades the objective toward the victims' margins, $\max_{x \in X} u_i(x) - \lambda \sum_j K_j(x)$, so it spends less of their slack (Appendix B).

The fixed points of the re-anchored dynamics have a structure that Corollary 3 predicts. A profile is *ε-certificate-stable* if, with anchors and gains computed from that profile by exact best responses, no seat has a 0-certified resolve of gain above ε in any public subgame. Every Nash equilibrium is 0-certificate-stable. The converse fails, and the failure is systematic.

Observation 1 (Equalisation). On 3-player Kuhn, re-anchored round-robin resolving from blueprints after 100 and $1{,}000$ CFR iterations, over all six seat orders, stops within 60 steps in 4 of 12 runs, the other eight still moving when the cap is reached. All four stop where the three best-response gains agree to 2×10^{-5}. At the stopped profiles every seat still has an unconstrained profitable update of at least 8×10^{-4}, and up to 1.3×10^{-2}, inside some public subgame, while the largest certified gain left to any seat, recomputed with exact anchors, is 7.3×10^{-8}.

In a constant-sum game, Corollary 2 at $\tau = 0$ says a 0-certified update with positive gain needs $\sum_{j \neq i} s_j > \sum_{j \neq i} K_j \ge 0$: some victim must sit strictly below the cap and lose payoff. The runs that stop do so at equalised profiles, but a seat below the cap whose payoff the resolver cannot lower holds slack nobody can cash, so certificate-stability is not equalisation (Appendix A). Certified resolving therefore need not converge to equilibrium: when the passing rule stops it, the profile is certificate-stable, not necessarily Nash.

7 Experiments

Setup. Three families: 3-player Kuhn poker, 3-player Leduc poker, and simultaneous-move stage games with binary private types for $n \in \{3, 4, 5, 6\}$ players, half with random payoffs and half within 0.15 of a constant-sum polymatrix game (MacQueen & Wright, 2023). Kuhn is resolved at every public subgame by every seat with an information set inside it, Leduc at 31 public states, giving 54 and 160 configurations; the stage games are resolved at the post-deal stage by one seat, 160 configurations.

Four arms are compared: the *certified* resolve of Proposition 2 at $\tau \in \{0, .002, .005, .01, .02, .05\}$; the *unsafe* arm, which runs full-continuation multiplayer CFR inside S from the blueprint's entry distribution and splices the result, in the pattern of the real-time search of Brown & Sandholm (2019); the *robust gadget*, a naïve n-player extension of the two-player gadget; and the blueprint.

Table 2: Certified resolving against unsafe resolving and the robust gadget, every update scored exactly (Johanson et al., 2011); $\delta_i > 0$ is the share of configurations whose certified gain at $\tau = 0$ exceeds 10^{-6}, W/T/L compares certified against unsafe gain at the clipped budget $\tau^\star$, *resolve* is the median wall time of one certified resolve at $\tau = 0$ on one core; the remaining definitions and tallies are in Appendix B.

		certified		unsafe		robust gadget			
Game	subg.	cert-viol	$\delta_i > 0$	raised	max	raised	max	W/T/L	resolve (ms)
Kuhn (3p)	54	0	78%	70%	.167	89%	.104	29/1/24	13
Leduc (3p)	160	0	71%	33%	.153	48%	.187	115/31/14	1012
Stage games (n=3)	40	0	55%	25%	.015	35%	.220	22/0/18	95
Stage games (n=4)	40	0	60%	18%	.014	28%	.078	25/0/15	7
Stage games (n=5)	40	0	70%	38%	.009	40%	.132	37/0/3	29
Stage games (n=6)	40	0	50%	55%	.027	60%	.161	37/0/3	136

Certified updates keep their certificate. Table 2 covers $2{,}244$ certified updates across 374 configurations and six budgets. None exceeds its certificate. On Leduc 41 updates exceed their own budget τ, by at most 2.3×10^{-9}, about twice the solver tolerance. A strictly positive certified gain at $\tau = 0$ exists in at least half of the configurations of every family, and the gain is non-decreasing in the budget (Fig. 3). Run the same program the other way and it lowers the Nash gap: the minimum-certificate resolve drives C_{anc} below -10^{-6} in 28 of the 54 Kuhn and 111 of the 160 Leduc configurations (Appendix B). Solved on the gadget by counterfactual regret minimisation instead of by the linear program, the same values are reached to 10^{-4} in 159 of the 160 instances, in medians of 200 to 700 iterations.

Unsafe resolving raises the Nash gap. Table 2 also reports the two baselines. The unsafe resolve raises the true Nash gap in most Kuhn configurations and in a third of the Leduc ones, and the robust gadget raises it more often still on every family. The robust gadget gives each opponent one terminate-or-follow choice at its entry group and has no adversary that prices the several opponents at once, the charge the meta-adversary of Section 4 supplies.

Gain under a common budget. To compare the two arms on equal terms, we re-solve the certified program at the budget $\tau^\star = \max(0, \Delta\mathsf{Gap}_{\mathrm{unsafe}})$ the unsafe update induces; both then satisfy the common ceiling $\mathsf{Gap}(\beta) + \tau^\star$, a difference below 10^{-6} counting as a tie (Table 2); where the unsafe update lowers the Nash gap, $\tau^\star = 0$ and the two arms share a ceiling, not a reduction. Where the certified resolve loses, the cause is the certificate's conservatism: its gain rises from a median of 0.37 of the exact-safe gain at $\tau = 0$ to 0.92 at $\tau = 0.005$ on Kuhn, and from 0.81 to 0.99 on Leduc (Appendix B).

The two-sided certificate recovers the gap. Dropping the positive part for a victim that does not act before S costs nothing (Proposition 3). On the stage games, where no opponent acts before the post-deal subgame, the two-sided program *is* the exact-safe program and wins or ties every configuration. On Kuhn and Leduc it wins one more configuration at the matched budget and lifts the median ratio to the exact-safe gain there (Appendix B).

8 Conclusion

Resolving is a convex program over one seat's sequence form, and its solution carries a self-contained, explicitly capped bound on the profile's Nash gap. Because the certificate is the value of a two-player zero-sum gadget, regret minimisation reaches it without a linear-programming solver, and the budgeted program is recovered from a penalised family of that gadget. On constant-sum games the cap identity prices safety exactly, and the gap decomposition accounts for the three terms the one-sided certificate gives up. Telescoping certificates keep the bound monotone under round-robin certified resolving, which on 3-player Kuhn stops at equally exploitable profiles: the three best-response gains agree to 2×10^{-5} in every re-anchored run that stops within the step cap.

REPRODUCIBILITY STATEMENT

Every lemma, proposition, corollary and theorem is proved in the main text or in Appendix A, with its hypotheses stated; Appendix B lists the game instances, blueprint iteration counts, CFR iteration counts, and solver tolerances. Every reported quantity is an exact full-game computation: payoffs by tree walks, best responses by linear-time backward induction, certificates by HiGHS. The supplementary material is an anonymized code package: one entry point per experiment, the fixed-seed logs and the configuration manifest behind every number in Table 2, and the compiled modules (Linux x86-64, CPython 3.12) those entry points call; re-running an entry point regenerates its logs from the shipped blueprints.

AI USE STATEMENT

We used generative AI tools to develop and refine theoretical results, to design and review the experimental methodology, to interpret results, to edit the prose, and to write the code for the experiments and figures. Every argument and proof was checked by hand, every passage read for correctness, and every experiment run by us before its numbers were reported; the authors answer for all of it.

REFERENCES

David Balduzzi, Sebastien Racaniere, James Martens, Jakob Foerster, Karl Tuyls, and Thore Graepel. The mechanics of n-player differentiable games. In *International Conference on Machine Learning*, pp. 354–363. PMLR, 2018.

Nicola Basilico, Andrea Celli, Giuseppe De Nittis, and Nicola Gatti. Team-maxmin equilibrium: Efficiency bounds and algorithms. In *AAAI*, pp. 356–362. AAAI Press, 2017.

Noam Brown and Tuomas Sandholm. Safe and nested subgame solving for imperfect-information games. *Advances in neural information processing systems*, 30, 2017.

Noam Brown and Tuomas Sandholm. Superhuman ai for heads-up no-limit poker: Libratus beats top professionals. *Science*, 359(6374):418–424, 2018.

Noam Brown and Tuomas Sandholm. Superhuman ai for multiplayer poker. *Science*, 365(6456): 885–890, 2019.

Noam Brown, Tuomas Sandholm, and Brandon Amos. Depth-limited solving for imperfect-information games. *Advances in neural information processing systems*, 31, 2018.

Neil Burch, Marc Lanctot, Duane Szafron, and Richard Gibson. Efficient monte carlo counterfactual regret minimization in games with many player actions. *Advances in neural information processing systems*, 25, 2012.

Neil Burch, Michael Johanson, and Michael Bowling. Solving imperfect information games using decomposition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 28, pp. 602–608, 2014.

Andrea Celli, Alberto Marchesi, Tommaso Bianchi, and Nicola Gatti. Learning to correlate in multi-player general-sum sequential games. In *NeurIPS*, pp. 13055–13065, 2019.

Constantinos Daskalakis, Maxwell Fishelson, and Noah Golowich. Near-optimal no-regret learning in general games. *Advances in Neural Information Processing Systems*, 34:27604–27616, 2021.

Trevor Davis, Kevin Waugh, and Michael Bowling. Solving large extensive-form games with strategy constraints. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pp. 1861–1868, 2019.

Gabriele Farina, Andrea Celli, Nicola Gatti, and Tuomas Sandholm. Ex ante coordination and collusion in zero-sum multi-player extensive-form games. In *NeurIPS*, pp. 9661–9671, 2018.

Gabriele Farina, Christian Kroer, and Tuomas Sandholm. Regret circuits: Composability of regret minimizers. In *ICML*, Proceedings of Machine Learning Research, pp. 1863–1872. PMLR, 2019.

Gabriele Farina, Ioannis Anagnostides, Haipeng Luo, Chung-Wei Lee, Christian Kroer, and Tuomas Sandholm. Near-optimal no-regret learning dynamics for general convex games. *Advances in Neural Information Processing Systems*, 35:39076–39089, 2022.

Sam Ganzfried. Projected exploitability descent for nash equilibrium computation in multiplayer imperfect-information games. *arXiv preprint arXiv:2606.29169*, 2026.

Sam Ganzfried and Tuomas Sandholm. Computing equilibria in multiplayer stochastic games of imperfect information. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence*, pp. 140–146, 2009.

Sam Ganzfried and Tuomas Sandholm. Game theory-based opponent modeling in large imperfect-information games. In *The 10th International Conference on Autonomous Agents and Multiagent Systems-Volume 2*, pp. 533–540, 2011.

Sam Ganzfried and Tuomas Sandholm. Endgame solving in large imperfect-information games. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, pp. 37–45, 2015.

Samuel Ganzfried and Tuomas Sandholm. Safe opponent exploitation. In *EC*, pp. 587–604. ACM, 2012.

Zhenxing Ge, Zheng Xu, Tianyu Ding, Wenbin Li, and Yang Gao. Efficient subgame refinement for extensive-form games. *Advances in Neural Information Processing Systems*, 36:8280–8291, 2023.

Zhenxing Ge, Zheng Xu, Tianyu Ding, Linjian Meng, Bo An, Wenbin Li, and Yang Gao. Safe and robust subgame exploitation in imperfect information games. In *Forty-first International Conference on Machine Learning*, 2024.

Richard Gibson and Duane Szafron. Regret minimization in multiplayer extensive games. In *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, volume 22, pp. 2802, 2011a.

Richard Gibson and Duane Szafron. On strategy stitching in large extensive form multiplayer games. *Advances in Neural Information Processing Systems*, 24, 2011b.

Sergiu Hart and Andreu Mas-Colell. A simple adaptive procedure leading to correlated equilibrium. *Econometrica*, 68(5):1127–1150, 2000.

Torsten Heinrich, Yoojin Jang, Luca Mungo, Marco Pangallo, Alex Scott, Bassel Tarbush, and Samuel Wiese. Best-response dynamics, playing sequences, and convergence to equilibrium in random games. *International Journal of Game Theory*, 52(3):703–735, 2023.

Michael Johanson and Michael Bowling. Data biased robust counter strategies. In *Artificial Intelligence and Statistics*, pp. 264–271. PMLR, 2009.

Michael Johanson, Martin Zinkevich, and Michael Bowling. Computing robust counter-strategies. *Advances in neural information processing systems*, 20, 2007.

Michael Johanson, Kevin Waugh, Michael Bowling, and Martin Zinkevich. Accelerating best response calculation in large extensive games. In *IJCAI*, volume 11, pp. 258–265, 2011.

Michael Johanson, Neil Burch, Richard Valenzano, and Michael Bowling. Evaluating state-space abstractions in extensive-form games. In *Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems*, pp. 271–278, 2013.

Qi Ju, Thomas Tellier, Meng Sun, Zhemei Fang, and YunFeng Luo. Preference-CFR: Beyond nash equilibrium for better game strategies. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=beBNOZP5Tk.

Daphne Koller, Nimrod Megiddo, and Bernhard Von Stengel. Efficient computation of equilibria for extensive two-person games. *Games and economic behavior*, 14(2):247–259, 1996.

Vojtěch Kovařík, Dominik Seitz, Viliam Lisý, Jan Rudolf, Shuo Sun, and Karel Ha. Value functions for depth-limited solving in zero-sum imperfect-information games. *Artificial Intelligence*, 314: 103805, 2023.

Christian Kroer and Tuomas Sandholm. A unified framework for extensive-form game abstraction with bounds. *Advances in Neural Information Processing Systems*, 31, 2018.

Marc Lanctot, Kevin Waugh, Martin Zinkevich, and Michael Bowling. Monte carlo sampling for regret minimization in extensive games. *Advances in neural information processing systems*, 22, 2009.

Chun Kai Ling and Fei Fang. Safe subgame resolving for extensive form correlated equilibrium. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 5116–5123, 2022.

Mingyang Liu, Chengjie Wu, Qihan Liu, Yansen Jing, Jun Yang, Pingzhong Tang, and Chongjie Zhang. Safe opponent-exploitation subgame refinement. *Advances in Neural Information Processing Systems*, 35:27610–27622, 2022.

Weiming Liu, Haobo Fu, Qiang Fu, and Yang Wei. Opponent-limited online search for imperfect information games. In *International Conference on Machine Learning*, pp. 21567–21585. PMLR, 2023.

Edward Lockhart, Marc Lanctot, Julien Pérolat, Jean-Baptiste Lespiau, Dustin Morrill, Finbarr Timbers, and Karl Tuyls. Computing approximate equilibria in sequential adversarial games by exploitability descent. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pp. 464–470, 2019.

Runyu Lu, Yuanheng Zhu, and Dongbin Zhao. Divergence-regularized discounted aggregation: Equilibrium finding in multiplayer partially observable stochastic games. In *The Thirteenth International Conference on Learning Representations*, 2025. URL `https://openreview.net/forum?id=KD5nJUgeW4`.

Revan MacQueen and James Wright. Guarantees for self-play in multiplayer games via polymatrix decomposability. *Advances in Neural Information Processing Systems*, 36:20230–20259, 2023.

Matej Moravcik, Martin Schmid, Karel Ha, Milan Hladik, and Stephen Gaukrodger. Refining subgames in large imperfect information games. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30, 2016.

Matej Moravčík, Martin Schmid, Neil Burch, Viliam Lisý, Dustin Morrill, Nolan Bard, Trevor Davis, Kevin Waugh, Michael Johanson, and Michael Bowling. Deepstack: Expert-level artificial intelligence in heads-up no-limit poker. *Science*, 356(6337):508–513, 2017.

Marc Ponsen, Steven De Jong, and Marc Lanctot. Computing approximate nash equilibria and robust best-responses using sampling. *Journal of Artificial Intelligence Research*, 42:575–605, 2011.

Martin Schmid, Matej Moravčík, Neil Burch, Rudolf Kadlec, Josh Davidson, Kevin Waugh, Nolan Bard, Finbarr Timbers, Marc Lanctot, G Zacharias Holland, Elnaz Davoodi, Alden Christianson, and Michael Bowling. Student of games: A unified learning algorithm for both perfect and imperfect information games. *Science Advances*, 9(46):eadg3256, 2023.

Michal Šustr, Vojtěch Kovařík, and Viliam Lisỳ. Monte carlo continual resolving for online strategy computation in imperfect information games. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, pp. 224–232, 2019.

Michal Šustr, Martin Schmid, Matej Moravčík, Neil Burch, Marc Lanctot, and Michael Bowling. Sound algorithms in imperfect information games. In *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems*, pp. 1674–1676, 2021.

Duane Szafron, Richard G Gibson, and Nathan R Sturtevant. A parameterized family of equilibrium profiles for three-player kuhn poker. In *AAMAS*, volume 13, pp. 247–254, 2013.

Oskari Tammelin, Neil Burch, Michael Johanson, and Michael Bowling. Solving heads-up limit texas hold'em. In *IJCAI*, volume 15, pp. 645–652, 2015.

Finbarr Timbers, Nolan Bard, Edward Lockhart, Marc Lanctot, Martin Schmid, Neil Burch, Julian Schrittwieser, Thomas Hubert, and Michael Bowling. Approximate exploitability: Learning a best response. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*, pp. 3487–3493. ijcai.org, 2022. doi: 10.24963/ijcai.2022/484.

Bernhard Von Stengel and Daphne Koller. Team-maxmin equilibria. *Games and Economic Behavior*, 21(1-2):309–321, 1997.

Kevin Waugh, Nolan Bard, and Michael Bowling. Strategy grafting in extensive games. *Advances in Neural Information Processing Systems*, 22, 2009.

Brian Zhang, Luca Carminati, Federico Cacciamani, Gabriele Farina, Pierriccardo Olivieri, Nicola Gatti, and Tuomas Sandholm. Subgame solving in adversarial team games. *Advances in Neural Information Processing Systems*, 35:26686–26697, 2022.

Brian Hu Zhang and Tuomas Sandholm. Small nash equilibrium certificates in very large games. In *NeurIPS*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/4fbe073f17f161810fdf3dab1307b30f-Abstract.html.

Brian Hu Zhang and Tuomas Sandholm. Subgame solving without common knowledge. *Advances in Neural Information Processing Systems*, 34:23993–24004, 2021.

Youzhi Zhang, Bo An, and Jakub Černỳ. Computing ex ante coordinated team-maxmin equilibria in zero-sum multiplayer extensive-form games. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 5813–5821, 2021.

Youzhi Zhang, Bo An, and Venkatramanan Subrahmanian. Computing optimal nash equilibria in multiplayer games. *Advances in Neural Information Processing Systems*, 36:21230–21254, 2023.

Martin Zinkevich, Michael Johanson, Michael Bowling, and Carmelo Piccione. Regret minimization in games with incomplete information. *Advances in neural information processing systems*, 20, 2007.

A Proofs

A.1 Proof of Lemma 1

Fix an opponent j and let y^* be a best response of j to β'_{-j}. Define y' to agree with y^* at every information set of j outside S and, inside S, to play at each entry group η a best response to β_{-j} below η. By the two group properties of Section 3, j's own reach is the same across each group and j's information sets inside S are partitioned by the groups, so y' is a well-defined perfect-recall strategy. Then

$$\mathrm{BRV}_j(\beta) \;\geq\; u_j(y', \beta_{-j}).$$

The profiles (y^*, β'_{-j}) and (y', β_{-j}) agree outside S: j plays y^* there in both, and $\beta'_{-j} = \beta_{-j}$ outside S. Inside S they differ, and the contribution of S to j's payoff decomposes over the groups reached, weighted by j's own reach $\pi_j^{y^*}(\eta)$ (the same for both profiles, as it is determined outside S) and the others' reach ω_η:

$$\begin{aligned} u_j(y^*, \beta'_{-j}) - u_j(y', \beta_{-j}) &= \sum_\eta \pi_j^{y^*}(\eta)\, \omega_\eta \left[v_j(\eta; y^*, \beta'_{-j}) - \mathrm{CBV}_j^0(\eta) \right] \\ &\leq \sum_\eta \pi_j^{y^*}(\eta)\, \omega_\eta \left[\mathrm{CBV}_j(\eta; \beta') - \mathrm{CBV}_j^0(\eta) \right], \end{aligned}$$

where the inequality uses $v_j(\eta; y^*, \beta'_{-j}) \leq \mathrm{CBV}_j(\eta; \beta')$. Dropping the factor $\pi_j^{y^*}(\eta) \leq 1$ on the positive terms and the non-positive terms altogether gives $\mathrm{BRV}_j(\beta') - \mathrm{BRV}_j(\beta) \leq \sum_\eta \omega_\eta [\mathrm{CBV}_j(\eta; \beta') - \mathrm{CBV}_j^0(\eta)]_+ = K_j(\beta')$. □

Comparison with the counterfactual-value anchor. With $\mathrm{CFV}_j^0(\eta) = v_j(\eta; \beta)$ in place of $\mathrm{CBV}_j^0(\eta)$ the same argument goes through with $y' = \beta_j$ inside S, giving the deficit used in two-player resolving. Since $\mathrm{CBV}_j^0 \geq \mathrm{CFV}_j^0$ pointwise, $K_j^{\mathrm{CBV}} \leq K_j^{\mathrm{CFV}}$; the two anchors differ, group by group, by j's regret inside S against the blueprint, which the counterfactual-value anchor charges to every update and the best-response anchor does not. At the blueprint itself $K_j^{\mathrm{CBV}}(\beta) = 0$, which is what makes $x = \beta_i$ feasible for every budget $\tau \geq 0$ in Proposition 2.

A.2 Proof of Proposition 2

Fix all seats but i at β. In sequence form the payoff of every player is $\langle x, U_p \otimes_{q \neq i} y_q \rangle$, linear in x for fixed y_{-i}, so $\delta_p(x) = u_p(x) - u_p(\beta)$ is affine. For a group η of opponent j, the value below η against (x, β_{-ij}) is $\langle x, U_\eta y \rangle$ for j's realisation weights y below η, and $\mathrm{CBV}_j(\eta; x) = \max_{y \in Y_j(\eta)} \langle x, U_\eta y \rangle$ is a supremum of affine functions, hence convex and piecewise linear; $Y_j(\eta)$ is a polytope, so the supremum is attained at a vertex and the function has finitely many pieces. Positive parts, non-negative combinations and maxima preserve convexity and piecewise linearity. The epigraph of a convex piecewise-linear function is a polyhedron described by finitely many linear inequalities; here the description is explicit through the dual of j's best response: with v_I for each information set I of j below η and v_η for the root,

$$v_{I(q)} \;\geq \sum_{z:\, s_j(z) = q} \mathrm{coef}_z \, x_{s_i(z)} + \sum_{I' \text{ child of } q} v_{I'} \quad \text{for every sequence } q \text{ of } j \text{ below } \eta,$$

and $\mathrm{CBV}_j(\eta; x) = \min v_\eta$ over feasible v by linear-programming duality. The constraints $\xi_\eta \geq v_\eta - \mathrm{CBV}^0_j(\eta)$, $\xi_\eta \geq 0$, $z \geq r_j(\beta) + \sum_\eta \omega_\eta \xi_\eta - \delta_j(x) - \mathsf{Gap}(\beta)$ and $z \geq r_i(\beta) - \delta_i(x) - \mathsf{Gap}(\beta)$ then express $C_{\mathrm{anc}}(x) \leq z$, and minimising z, or maximising $u_i(x)$ under $z \leq \tau$, is a linear program in (x, v, ξ, z). Feasibility: at $x = \beta_i$ every v_η can be set to $\mathrm{CBV}^0_j(\eta)$ and all slacks to zero, giving $z = 0$. □

A.3 Proof of Proposition 3

Let y^* be a best response of j to β'_{-j}. Since j does not act before S, every information set of j is either inside S or in a part of the tree disjoint from S and its ancestors; the latter part is untouched by the resolve, because x changes only seat i's strategy inside S, so the restriction of y^* to it is a best response against β_{-j} as well. Write $u_j = u_j^{\bar{S}} + u_j^S$ for the split of j's payoff into terminals outside and inside S. The first term is the same function of j's strategy under β'_{-j} and β_{-j}. For the second, j's reach to every entry is 1, so

$$u_j^S(y_j, \beta'_{-j}) = \sum_\eta \omega_\eta \, v_j(\eta; y_j, \beta'_{-j}),$$

and because j's information sets inside S are partitioned by the groups (Section 3), the maximum over y_j splits over η and equals $\sum_\eta \omega_\eta \mathrm{CBV}_j(\eta; \beta')$. The same computation against β_{-j} gives $\sum_\eta \omega_\eta \mathrm{CBV}^0_j(\eta)$. Subtracting, $\mathrm{BRV}_j(\beta') - \mathrm{BRV}_j(\beta) = \sum_\eta \omega_\eta [\mathrm{CBV}_j(\eta; \beta') - \mathrm{CBV}^0_j(\eta)] = \widetilde{K}_j(\beta')$, and $\widetilde{K}_j \leq K_j$ termwise. Convexity and piecewise linearity are those of $\mathrm{CBV}_j(\eta; x)$ (Proposition 2); the epigraph description drops the constraint $\xi_\eta \geq 0$, so the program stays a linear program. With $D_j = \widetilde{K}_j$ the constraint $r_j(\beta) + \widetilde{K}_j - \delta_j \leq \mathsf{Gap}(\beta) + \tau$ reads $r_j(\beta') \leq \mathsf{Gap}(\beta) + \tau$ by the identity $r_j(\beta') = r_j(\beta) + D_j - \delta_j$. □

A.4 Proof of Proposition 4

Since y^* best-responds inside S at every group it reaches with positive counterfactual weight, $v_j(\eta; y^*, \beta'_{-j}) = \mathrm{CBV}_j(\eta; \beta')$ whenever $\pi_j^{y^*}(\eta) > 0$ and $\omega_\eta > 0$; groups with $\omega_\eta = 0$ contribute nothing to any of the sums and are omitted, so the inequality in the proof of Lemma 1 is an equality: $\mathrm{BRV}_j(\beta') - u_j(y', \beta_{-j}) = \sum_\eta \pi_j^{y^*}(\eta) \omega_\eta d_\eta$. Hence $D_j = \sum_\eta \pi_j^{y^*}(\eta) \omega_\eta d_\eta - \big[\mathrm{BRV}_j(\beta) - u_j(y', \beta_{-j})\big]$, and subtracting from $K_j = \sum_\eta \omega_\eta [d_\eta]_+$ with $d_\eta = [d_\eta]_+ - [d_\eta]_-$ gives the three terms. Non-negativity: $\pi \in [0, 1]$, $[\cdot]_\pm \geq 0$, and y' is one strategy against β_{-j}. The last claim is the proof of Proposition 3. □

A.5 Proof of Theorem 1

Let x be any resolver strategy and consider the meta-adversary's best response. In the follow subtree below group η of victim j, the adversary controls exactly j's decisions inside S, the other seats are frozen at β and the resolver plays x, so the adversary's best-response value is $\max_y \langle x, U_\eta y \rangle - \mathrm{CBV}^0_j(\eta) = \mathrm{CBV}_j(\eta; x) - \mathrm{CBV}^0_j(\eta)$ times Ω_j; against terminate, worth 0, the best choice is worth

$\Omega_j[\mathrm{CBV}_j(\eta;x)-\mathrm{CBV}_j^0(\eta)]_+$. Chance over groups with weights ω_η/Ω_j gives $\sum_\eta \omega_\eta[\cdot]_+ = K_j(x)$. The payoff part draws an entry with its absolute blueprint probability $P_0(e) = c_e \prod_p \pi_p^\beta(e)$, ends with payoff 0 with the remaining probability $1-\sum_e P_0(e)$, and below the entry freezes all seats but i at β and pays the adversary $-(u_j(z)-u_j(e;\beta))$, where $u_j(e;\sigma)$ is j's expected payoff in the tree below e under σ, defined whatever $P_0(e)$; against x it is worth $-\sum_e P_0(e)\big(u_j(e;x,\beta_{-i})-u_j(e;\beta)\big) = -\delta_j(x)$, because β' and β differ only below the entries and the remaining probability carries payoff 0; drawing entries conditionally on reaching S would give $-\delta_j(x)/\sum_e P_0(e)$ instead. Groups with $\omega_\eta = 0$ are never drawn and are omitted from the margin part; if $\Omega_j = 0$ the victim's branch is its payoff part alone. The half-half chance node with doubled terminals gives $K_j(x)-\delta_j(x)$, plus the constant $r_j(\beta)-\mathsf{Gap}(\beta)$; the self branch is worth $-\delta_i(x)+r_i(\beta)-\mathsf{Gap}(\beta)$. The root maximum over branches is $C_{\mathrm{anc}}(x)$. Perfect recall: the adversary's information sets are labelled by branch, group and the victim's own information set inside S, so it observes exactly what j observes and never forgets its own moves; the resolver's information sets are the original ones, copied with their original action sets, and its histories inside different copies of S are indistinguishable by construction, which is a coarsening consistent with perfect recall because the resolver takes no action before entering a copy. The game is zero-sum by construction, so its value is $\min_x \max_{\text{adv}} = \min_x C_{\mathrm{anc}}(x)$. □

For an opponent that does not act before S the margin part of its branch drops the terminate option: chance draws η and play continues in the follow subtree, whose value against x is $\widetilde{K}_j(x)$ after averaging over groups, the factor Ω_j on the terminals cancelling the chance weights ω_η/Ω_j; Theorem 1 then holds with $\widetilde{K}_j$ in place of K_j for that victim.

A.6 The penalised gadget and the budgeted program

For $\lambda \geq 0$ the penalised problem $P(\lambda) = \max_{x\in X}[u_i(x) - \lambda C_{\mathrm{anc}}(x)]$ is, up to the constant $u_i(\beta)$, the value of a two-player zero-sum game: chance opens with probability $\frac{1}{2}$ into the self payoff part, worth $\delta_i(x) = u_i(x) - u_i(\beta)$ to the resolver, and with probability $\frac{1}{2}$ into the gadget with its terminals scaled by λ, all terminals doubled; the value is $\max_x[\delta_i(x) - \lambda C_{\mathrm{anc}}(x)] = P(\lambda) - u_i(\beta)$, with the same maximisers, and CFR solves it. Because C_{anc} is a maximum of finitely many affine functions of x, the set $E = \{(x,t) : x \in X,\ C_{\mathrm{anc}}(x) \leq t \leq \bar{B}\}$ with $\bar{B} = \max_X C_{\mathrm{anc}}$ is a polytope, and for $\lambda > 0$, $P(\lambda) = \max_{(x,t)\in E}[u_i(x) - \lambda t]$, every maximiser having $t = C_{\mathrm{anc}}(x)$. An affine function on a polytope is maximised at a vertex, so with $\mathcal{V}$ the finite vertex set of E, $P(\lambda) = \max_{(x_v,t_v)\in\mathcal{V}}[u_i(x_v) - \lambda t_v]$: P is convex and piecewise linear in λ with at most $|\mathcal{V}|$ pieces.

The pieces come from vertices of E, not of X: on $X = [0,1]$ with $u_i(x) = x$ and $C_{\mathrm{anc}}(x) = |x - \frac{1}{2}|$ the maximiser for every $\lambda > 1$ is the interior point $\frac{1}{2}$. We solve $P(\lambda)$ as this linear program by a simplex method, which returns a vertex $(x_\lambda, C_{\mathrm{anc}}(x_\lambda))$ of E; for $\lambda < \lambda'$ adding the two optimality inequalities gives $C_{\mathrm{anc}}(x_{\lambda'}) \leq C_{\mathrm{anc}}(x_\lambda)$. For a vertex x of X the point $(x, C_{\mathrm{anc}}(x))$ is a vertex of E, so a vertex maximiser of u_i over X also names a piece of P. Let $V(\tau)$ be the value of the budgeted program at $\tau \geq 0$, and $R = \max_X u_i - \min_X u_i$, at most the range of seat i's terminal payoffs. For every $\tau \geq 0$, $V(\tau) = \min_{\lambda\geq 0}[P(\lambda) + \lambda\tau]$: the inequality $\leq$ is the chain $u_i(x) \leq u_i(x) - \lambda[C_{\mathrm{anc}}(x) - \tau] \leq P(\lambda) + \lambda\tau$ for feasible x, and for $\geq$ write $P(\lambda) + \lambda\tau = \max_{v\in\mathcal{V}}[u_i(x_v) - \lambda(t_v - \tau)]$, so that $\min_{\lambda\geq 0}[P(\lambda) + \lambda\tau]$ is a linear program in (z,λ) whose dual, $\max \sum_v \mu_v u_i(x_v)$ over $\mu \geq 0$, $\sum_v \mu_v = 1$, $\sum_v \mu_v t_v \leq \tau$, ranges over exactly the convex combinations of vertices lying in $E \cap \{t \leq \tau\}$ and has value $V(\tau)$; both are feasible ($\lambda = 0$; $(\beta_i, 0) \in E$), so the values agree. A single penalised gadget therefore has value $P(\lambda) - u_i(\beta)$, not $V(\tau)$; the multiplier $\hat{\lambda}$ found below attains the minimum, $V(\tau) = P(\hat{\lambda}) + \hat{\lambda}\tau$.

Proposition 7 (Primal recovery)**.** *Let $\lambda \geq 0$ and let x_- and x_+ be maximisers of $P(\lambda)$ with $C_{\mathrm{anc}}(x_+) \leq \tau \leq C_{\mathrm{anc}}(x_-)$. Every point of the segment $[x_-, x_+]$ maximises $P(\lambda)$, and any point x^τ of the segment with $C_{\mathrm{anc}}(x^\tau) = \tau$ is a τ-certified resolve of largest gain.*

Proof. $u_i - \lambda C_{\mathrm{anc}}$ is concave on the convex set X (affine minus a non-negative multiple of a convex function) and attains its maximum at both endpoints, hence on the whole segment. C_{anc} is continuous on the segment and takes values $\leq \tau$ and $\geq \tau$ at its ends, so some x^τ has $C_{\mathrm{anc}}(x^\tau) = \tau$. For any feasible x, $u_i(x) \leq u_i(x) - \lambda(C_{\mathrm{anc}}(x) - \tau) \leq P(\lambda) + \lambda\tau = u_i(x^\tau) - \lambda C_{\mathrm{anc}}(x^\tau) + \lambda\tau = u_i(x^\tau)$. □

Finding the pair. Solve $\max_X u_i$; if its vertex maximiser x_0 has $C_{\mathrm{anc}}(x_0) \le \tau$ the budget is inactive and x_0 is returned. Otherwise double λ from 1 until $C_{\mathrm{anc}}(x_\lambda) \le \tau$. This stops at the latest once $\lambda > \lambda_0(\tau) = \max\{[u_i(x_v) - V(\tau)]/[t_v - \tau] : (x_v, t_v) \in \mathcal{V},\ t_v > \tau\}$, because for such λ every vertex with $t_v > \tau$ has $u_i(x_v) - \lambda t_v < V(\tau) - \lambda\tau \le P(\lambda)$, the last inequality evaluated at a τ-certified resolve of gain $V(\tau)$, so the returned vertex has $t_v \le \tau$; for $\tau > 0$ it also stops once $\lambda \ge R/\tau$, since a maximiser with $C_{\mathrm{anc}}(x_\lambda) > \tau$ would give $P(\lambda) < \max_X u_i - \lambda\tau \le \min_X u_i \le u_i(\beta_i) - \lambda C_{\mathrm{anc}}(\beta_i) \le P(\lambda)$. The doubling therefore takes at most $2 + \lceil \log_2 \max(1, \bar\lambda) \rceil$ solves with $\bar\lambda = \min(\lambda_0(\tau), R/\tau)$, R/τ read as $+\infty$ at $\tau = 0$, a count that depends on the instance as well as on $|\mathcal{V}|$: $u_i(x) = x$, $C_{\mathrm{anc}}(x) = \kappa x$ on $[0, 1]$ at $\tau = 0$ has two pieces and $\lambda_0 = 1/\kappa$.

If $C_{\mathrm{anc}}(x_\lambda) = \tau$ at any solve, x_λ is returned, since for every feasible x, $u_i(x) \le u_i(x) - \lambda(C_{\mathrm{anc}}(x) - \tau) \le P(\lambda) + \lambda\tau = u_i(x_\lambda)$. Otherwise keep a bracket $\lambda_- < \lambda_+$ with returned vertices x_- of $P(\lambda_-)$ and x_+ of $P(\lambda_+)$ and $C_{\mathrm{anc}}(x_-) > \tau > C_{\mathrm{anc}}(x_+)$, initially the last two doubling solves, or x_0 with $\lambda_- = 0$ when the first solve is already feasible, and set $\hat\lambda = [u_i(x_-) - u_i(x_+)]/[C_{\mathrm{anc}}(x_-) - C_{\mathrm{anc}}(x_+)]$, the multiplier at which the two pieces through x_- and x_+ meet; the two optimality inequalities give $\lambda_- \le \hat\lambda \le \lambda_+$.

Solve $P(\hat\lambda)$. If $P(\hat\lambda) = u_i(x_-) - \hat\lambda C_{\mathrm{anc}}(x_-)$ then x_- and x_+ both maximise $P(\hat\lambda)$ and Proposition 7 applies with $\lambda = \hat\lambda$; the point is explicit: $u_i - \hat\lambda C_{\mathrm{anc}}$ is concave and maximal at both ends of the segment, hence constant on it, so for $\hat\lambda > 0$ the affinity of u_i makes C_{anc} affine on the segment and $x^\tau = (1 - \alpha)x_- + \alpha x_+$ with $\alpha = [C_{\mathrm{anc}}(x_-) - \tau]/[C_{\mathrm{anc}}(x_-) - C_{\mathrm{anc}}(x_+)]$; if $\hat\lambda = 0$ then $\lambda_- = 0$, $x_- = x_0$, and $u_i(x_+) = u_i(x_0) = \max_X u_i$, so the feasible point x_+ is returned. Otherwise the returned vertex $\hat x$ has $C_{\mathrm{anc}}(x_+) < C_{\mathrm{anc}}(\hat x) < C_{\mathrm{anc}}(x_-)$: were $C_{\mathrm{anc}}(\hat x) \ge C_{\mathrm{anc}}(x_-)$, optimality of x_- at $\lambda_- \le \hat\lambda$ would give $u_i(\hat x) - \hat\lambda C_{\mathrm{anc}}(\hat x) \le u_i(x_-) - \lambda_- C_{\mathrm{anc}}(x_-) - (\hat\lambda - \lambda_-)C_{\mathrm{anc}}(\hat x) \le u_i(x_-) - \hat\lambda C_{\mathrm{anc}}(x_-)$, against $P(\hat\lambda) > u_i(x_-) - \hat\lambda C_{\mathrm{anc}}(x_-)$, and symmetrically at x_+ with $\lambda_+ \ge \hat\lambda$. So $\hat x$ is a vertex whose C_{anc} lies strictly inside the bracket's interval, hence distinct from every vertex seen so far, all of which lie outside that open interval; replace the end on its side of τ, or return $\hat x$ if $C_{\mathrm{anc}}(\hat x) = \tau$. The interval shrinks strictly at every round and $\mathcal{V}$ is finite, so the secant phase takes at most $|\mathcal{V}|$ solves. On $X = [0, 1]$ with $u_i = C_{\mathrm{anc}} = x$ and $\tau = \frac{1}{2}$: $x_0 = 1$, the doubling stops at the first $\lambda \in \{1, 2\}$ whose returned vertex is $x = 0$, $\hat\lambda = 1$ certifies the pair $\{1, 0\}$, and the segment gives $\frac{1}{2}$.

Approximate solves. The procedure above assumes exact solves. With an ε-optimal $\hat x$ for $P(\lambda)$, $\lambda \ge 0$, the same chain gives, for every τ-certified x, $u_i(x) \le u_i(\hat x) + \varepsilon + \lambda[\tau - C_{\mathrm{anc}}(\hat x)]$, a computable upper bound on $V(\tau)$. When $C_{\mathrm{anc}}(\hat x) \le \tau$ the delivered update is τ-certified with gain within $\varepsilon + \lambda[\tau - C_{\mathrm{anc}}(\hat x)]$ of $V(\tau)$; when $C_{\mathrm{anc}}(\hat x) > \tau$, mixing it with a feasible x_+ as $\tilde x = (1 - \alpha)\hat x + \alpha x_+$, $\alpha = [C_{\mathrm{anc}}(\hat x) - \tau]/[C_{\mathrm{anc}}(\hat x) - C_{\mathrm{anc}}(x_+)]$, restores $C_{\mathrm{anc}}(\tilde x) \le \tau$ by convexity at a gain loss of at most $\alpha[u_i(\hat x) - u_i(x_+)]_+$, both quantities computed by exact re-evaluation of the certificate. The repair restores feasibility at a computable loss and bounds the achieved gain from above. Termination is proved for exact solves; the experiments solve the program exactly, and Appendix B compares the secant recovery with an exact oracle against the direct linear program.

Reading a certified resolve. A τ-certified resolve spends a budget of Nash gap on payoff. At $\tau = 0$ it moves only where a victim can absorb a deficit, that is where $r_j(\beta) + K_j - \delta_j \le \mathsf{Gap}(\beta)$, or where the victim's payoff falls without raising its best-response value. Shading the objective toward frugality, $\max u_i(x) - \lambda \sum_j K_j(x)$, prefers updates that leave the victims' margins intact; Section 6 uses it.

A.7 Proofs for Section 6

Proposition 5. Apply Proposition 1 at each step with the current profile as blueprint; the intermediate gaps cancel in the sum, and $\max(B_t, 0)$ replaces B_t because a negative certificate is not passed forward.

Proposition 6. Induction on t. At $t = 0$ the ledger equals $r(\beta)$. If $r_p(\beta^t) \le \bar r_p^t$ for all p and seat i resolves, then $r_i(\beta^{t+1}) = r_i(\beta^t) - \delta_i \le \bar r_i^t - \delta_i$ and, by Lemma 1 applied with β^t as blueprint, $r_j(\beta^{t+1}) \le r_j(\beta^t) + K_j - \delta_j \le \bar r_j^t + K_j - \delta_j$. A step is 0-certified against the ledger exactly when the new maximum does not exceed the old one. □

Certificate-stable is not equalised. Take three-player matching pennies with an inert third seat: seats 1 and 2 mix over $\{\pm 1\}$ with means $a, b \in [-1, 1]$, $u_1 = ab$, $u_2 = -ab$, $u_3 \equiv 0$, and let the public subgame be the whole game, so each opponent has one entry group of weight 1 and $K_j = [D_j]_+$. At $a = 1/2$, $b = -1/4$ the gains are $(r_1, r_2, r_3) = (3/8, 3/8, 0)$. Seat 1 has a safe update with positive gain: $a' = 0$ gives $g_1 = 1/8$, $r_2(\beta') = 0$ and $\mathsf{Gap}(\beta') = 1/4$. It has no 0-certified one: victim 2's constraint with $s_2 = 0$ reads $K_2 - g_2 \leq 0$, and constant sum gives $g_2 = -g_1$, hence $g_1 \leq -K_2 \leq 0$. The certificate cannot see the safe update because $D_2 = -1/2$ is a fall of victim 2's best-response value, and the deficit credits no falls. Seat 2 is blocked in the same way and seat 3's payoff is constant. The profile is certificate-stable and not equalised, and Corollary 2 is one-way: slack is necessary for a certified gain, not sufficient.

Proposition 8 (No multiplicative guarantee)**.** *There is a constant-sum three-player game, a blueprint, and a public subgame equal to the whole game at which the exact-safe gain at $\tau = 0$ is $1/5$ and the 0-certified gain of Proposition 2 is 0; the two-sided program of Proposition 3 attains $1/5$ there.*

The exact-safe program at $\tau = 0$ maximises $g_1 = (1/2 - a')/4$ subject to $r'_1 = 1/4 + a'/4 \leq 3/8$ and $r'_2 = |a'| - a'/4 \leq 3/8$; the second constraint gives $a' \geq -3/10$, so the optimum is $a' = -3/10$ with $g_1 = 1/5$, victim 2 on the cap and $r'_1 = 7/40$. No seat acts before the whole game, so by Proposition 3 $\widetilde{K}_2 = \mathrm{BRV}_2(\beta') - \mathrm{BRV}_2(\beta) = 3/10 - 1/2 = -1/5$, and the two-sided program attains the same $1/5$. This proves Proposition 8.

The two-player reading of Theorem 2. For $n = 2$ the theorem reads $\delta_i \leq \tau + s_j - D_j$, which with $\delta_j = -\delta_i$ is the cap condition itself. Two-player safe resolving guarantees $D_j \leq 0$ instead (Burch et al., 2014): it never raises $r_i + r_j$, but a gain δ_i moves up to δ_i of regret from the resolver to its opponent, and whether the maximum rises depends on which seat was the more exploitable one (Section 3).

B Experimental Details

Games. 3-player Kuhn: four cards, one ante, one bet of one chip, no raise; 48 information sets, 312 terminals, constant-sum. 3-player Leduc: six cards (two suits, three ranks), one private card each, one public card, two betting rounds with bets of 2 and 4 and at most one bet per round, pairs beat high cards, ties split; 2,232 information sets, 29,880 terminals, constant-sum. The 31 Leduc public states are the 11 first-round states and 20 second-round states drawn uniformly without replacement from the 432 under a fixed seed, listed in Table 3; the 10,000-iteration blueprint covers the 21 of them whose cluster jobs finished before the deadline, marked there. n-player stage games: binary private types dealt uniformly and independently, then one binary action per seat taken without observing the others; *random* payoffs draw an independent uniform $[-1, 1]^n$ vector for every (type, action) profile; *near-CSP* payoffs give every pair $i < j$ a 4×4 matrix with uniform $[-1, 1]$ entries indexed by (type, action), added to u_i and subtracted from u_j, plus 0.15 times an independent uniform $[-1, 1]$ tensor term per player; seeds 0–19 per (n, family), generator seeded by (n, seed), resolver = seed $\bmod n$. These games are not constant-sum, and the certificate of Proposition 1 applies unchanged.

Table 3: The 31 Leduc public states: first-round actions, then for second-round states the index of the public card among the six and the second-round actions (k check, b bet, c call, f fold). * marks the 21 states the 10,000-iteration blueprint covers.

First round (11, all*): b, bc, bf, k, kb, kbc, kbf, kk, kkb, kkbc, kkbf
Second round (20): bcc/4/kbc*, bcc/5/bf*, bcf/0/kb*, bcf/4/b*, bcf/5/k*, bcf/5/kb*, bfc/0/*, bfc/2/b*, bfc/3/*, bfc/4/*, bfc/4/k, bfc/5/kb, kbcc/1/kbc, kbcf/1/b, kbcf/5/, kkbcc/1/bc, kkbcf/4/k, kkbfc/5/kb, kkk/0/kkbf, kkk/2/kkb

Blueprints and arms. Blueprints are the plain average strategies of simultaneous-update multiplayer CFR with regret matching, using a per-iteration policy snapshot so that every node of an information set sees the same strategy within an iteration; the number of iterations is 100, 1,000 and 10,000 for Kuhn, 100 and 1,000 for Leduc (a 10,000-iteration Leduc blueprint is included for the 21 public states, 44 resolver configurations, whose cluster jobs completed), and 500 for the stage games. Unsafe resolving runs the same CFR inside S from the chance node over entries

weighted by $c_e \prod_p \pi_p^\beta(e)$, for 2,000 iterations on Kuhn, 500 on Leduc, and 1,000, 1,000, 400, 200 for $n = 3, \ldots, 6$; the depth limit and continuation strategies of the real-time search are dropped, and its blueprint is the same one the certified arm resolves against. The robust gadget is the n-player game whose root chance draws an entry e with probability proportional to $c_e\, \pi_i^\beta(e)$, the opponents' own reach excluded; at that entry every opponent chooses *terminate* or *follow* at its entry group inside the public state, in a fixed seat order, and no opponent observes an earlier choice. Below the choices play proceeds in a copy of S in which a terminating opponent is frozen at β and receives, at every terminal, its blueprint counterfactual value in place of the realised payoff, while a following opponent and the resolver act freely and receive realised payoffs. Multiplayer CFR runs on this game for the iteration counts above and the resolver's average strategy is spliced; it is an ablation of the meta-adversary, not a method from the literature. Certified resolves are the linear programs of Proposition 2, solved by HiGHS through SciPy with primal and dual feasibility tolerance 10^{-9}; when the budget-free program reports infeasibility because the blueprint sits exactly on the boundary, a fallback relaxes the budget by 10^{-7} and the certificate is re-evaluated exactly on the delivered strategy; the fallback never fired: the manifest records its flag for the 2,244 grid, 374 matched and 2,618 two-sided solves, all 0, and it fired in 0 of the 7,614 dynamics solves; the exact-safe and minimum-certificate programs have no relaxation branch, and the signed program reports infeasibility instead.

Evaluation. Every reported quantity is computed on the full game without sampling. Best responses use a linear-time backward induction over the player's information sets ordered by depth; on Kuhn they agree to machine precision with an independent implementation. The true change of the Nash gap, the deficits and the payoff changes are recomputed from the delivered strategy, independently of the program's own accounting; on Kuhn the two agree to 10^{-9} on random strategies. A violation is a delivered update whose true $\Delta\mathsf{Gap}$ exceeds its certificate by more than 10^{-9}; there are none. With the budget τ in place of the certificate, 41 Leduc updates exceed τ, by more than the 10^{-9} tolerance and by at most 2.3×10^{-9}; we report these separately; the two-sided program has 49 such Leduc points, also by at most 2.3×10^{-9}. In Table 2 *cert-viol* counts, over the six budgets, the updates whose true $\Delta\mathsf{Gap}$ exceeds their certificate, $\delta_i > 0$ is the share of configurations whose certified gain at $\tau = 0$ exceeds 10^{-6}, and *raised* is the share of updates whose true $\Delta\mathsf{Gap}$ exceeds 10^{-4}, with the largest such change. The configuration manifest is supplied with the supplementary material: one row per certified update, with its game, blueprint, subgame, resolver, budget, gain, deficits, certificate, true change, unsafe and robust arms, exact-safe gain, matched budget, the requested and effective budget, the solver status of every solve of every family (grid, exact-safe, matched, minimum-certificate, two-sided, signed and dynamics), and the relaxation and over-budget flags. On the 160 stage-game configurations, where no opponent acts before the post-deal subgame, the two-sided program of Proposition 3 and the whole-game exact-safe program return the same value, to within 5.99×10^{-14}. The share of configurations where the unsafe resolve raises the Nash gap rises with blueprint quality: on Leduc it is 27% at 100 iterations, 36% at 1,000 and 55% at 10,000, over the 44 configurations the 10,000-iteration blueprint covers. In Table 2, W/T/L compares the certified gain against the unsafe gain at the budget $\tau^\star = \max(0, \Delta\mathsf{Gap}_{\mathrm{unsafe}})$ induced by the unsafe update's own true $\Delta\mathsf{Gap}$, a difference below 10^{-6} counting as a tie and no configuration excluded, and *resolve* is the median wall time of one certified resolve at $\tau = 0$ on one core.

On the configurations where the unsafe update lowers the Nash gap, the signed budget $\tau = \Delta\mathsf{Gap}_{\mathrm{unsafe}} < 0$ makes the one-sided program infeasible in 6 of the 10 Kuhn, 15 of the 64 Leduc and 40 of the 75 stage-game configurations, while the exact-safe program, for which the unsafe strategy itself is feasible, wins, ties and loses $9/1/0$, $49/15/0$ and $75/0/0$. On the wider subgroup where the unsafe update does not raise the Nash gap, the 10 Kuhn, 92 Leduc and 75 stage-game configurations, the certified resolve at the clipped budget wins, ties and loses $4/0/6$, $62/21/9$ and $54/0/21$, and over all 374 configurations the exact-safe program wins, ties and loses $52/2/0$ on Kuhn, $127/33/0$ on Leduc and $160/0/0$ on the stage games.

Two-sided program. On the configurations where the one-sided certificate loses at the matched budget, the exact-safe gain exceeds the certified gain by a median of 1.9×10^{-4} on Kuhn, 2.4×10^{-4} on Leduc and 4.8×10^{-4} at $n = 3$. On the stage games, where no opponent acts before the post-deal subgame, the two-sided program of Proposition 3 wins or ties all 160 configurations, against 121 wins and 39 losses for the one-sided certificate. On Kuhn and Leduc it improves the certified gain in 46 of the 324 and 176 of the 960 (configuration, budget) points, takes the matched-budget W/T/L from $29/1/24$ to $30/1/23$ and from $115/31/14$ to $116/31/13$, and lifts the median ratio to the exact-safe

gain at that budget from 0.87 to 0.93 and from 0.93 to 0.98, over the configurations whose exact-safe gain there is numerically positive, all 54 on Kuhn and 159 of the 160 on Leduc. At the grid budgets the ratio of certified to exact-safe gain quoted in Section 7 is taken over the configurations whose exact-safe gain is positive: all 54 on Kuhn at both budgets, and on Leduc 157 of 160 at $\tau = 0$ and all 160 at $\tau = 0.005$, the three omitted configurations having exact-safe gain 0 at $\tau = 0$; its medians are 0.37 and 0.92 on Kuhn and 0.81 and 0.99 on Leduc. Section 4 tabulates the three programs.

Minimum certificate. The minimum-certificate resolve drives C_{anc} below -10^{-6} in 28 of the 54 Kuhn and 111 of the 160 Leduc configurations; in those, the resolving seat's own payoff falls below -10^{-6} in 11 and 58 cases (18 and 76 over all configurations), by 0.014 and 0.056 at worst. Plain multiplayer CFR continued from the same blueprint reaches a lower Nash gap after 400 further iterations, but not monotonically: it rises from 0.020 to 0.028 along the way.

Round-robin dynamics. State at step t: profile β^t and ledger $\bar{r}^t$, with $\bar{r}^0 = r(\beta)$ from exact best responses. Step $t \geq 0$ reads $(\beta^t, \bar{r}^t)$, writes $(\beta^{t+1}, \bar{r}^{t+1})$, has acting seat $i = \mathrm{order}[t \bmod n]$, and

1. if $t \bmod n = 0$ and re-anchoring is on, resets $\bar{r}^t \leftarrow r(\beta^t)$ with n exact best responses (the *ledger reset*, at the first step of every round);
2. for every public subgame in which i acts, recomputes the anchors $\mathrm{CBV}_j^0(\eta) \leftarrow \mathrm{CBV}_j(\eta; \beta^t)$ of every opponent against the current profile (the *deficit anchor*, every step, subgame-local), and solves the budget-free program with $\bar{r}^t$ in place of $r(\beta)$;
3. applies the resolve of largest certified gain if it exceeds $\varepsilon = 10^{-7}$, else passes; the update is $\bar{r}_i^{t+1} = \bar{r}_i^t - \delta_i$, $\bar{r}_j^{t+1} = \bar{r}_j^t + K_j - \delta_j$.

The run stops when a round that opened with a ledger reset ends with all n seats passing, or at 60 steps; a run that stops by the passing rule ends at an ε-certificate-stable profile by construction, and we confirm it by a final audit with fresh anchors and exact gains; reaching the step cap carries no such guarantee. On the 100-iteration Kuhn blueprint every entry of the ledger sits at $\Phi_0 = 0.037$ within 17 steps in all six seat orders, and the true Nash gap ends between 0.017 and 0.036 while Φ never changes. Re-anchored resolving is not monotone step by step: the true Nash gap rose at 79 intermediate steps across the 24 re-anchored runs, never above its round's opening value. Lemma 1 applied with β^t as blueprint is what step 2 needs; step 1 lowers a valid upper bound and is optional. Frugal resolves use the objective $u_i(x) - \sum_j K_j(x)$; their stopping is relative to that objective and is not certificate stability. Combined with re-anchoring, on the 100-iteration blueprint they reach the same Nash gap as re-anchoring alone with the largest drop in the sum of gains, from 0.090 to 0.029, and they stop within 9–18 steps where the re-anchored dynamics stops in 4 of the 12 runs and otherwise reaches 60. The CFR comparison continues multiplayer CFR from the blueprint's iteration count and evaluates the average strategy every 20 iterations.

Joint updates. For each of the 11 public subgames of Kuhn, each of the six seat orders, and the 100- and $1{,}000$-iteration blueprints ($11 \times 6 \times 2 = 132$ cases), seats resolve in order with budget 0.005 against the ledger; the sequential certificate is $\sum_t \max(B_t, 0)$. The simultaneous arm solves each seat's program against the original blueprint and splices all resolves at once; its budget line is $n \times 0.005$. The sequential certificate holds in all 132 cases, with median slack 0.004, and 74% of the joint updates do raise the Nash gap, within the certificate. The simultaneous splice exceeds $n\tau$ in 6 of the 132 cases, by up to 2.7τ.

Scale. All Kuhn experiments run on four cores of a workstation; Leduc and the stage games on one CPU node of a cluster with 32 worker processes, in about one hour and fifteen minutes each. Timings in Table 2 are single-process wall times of one certified resolve including the exact certificate. The program solves in 1–2 ms on the n-player stage games; the delivered resolve, program plus exact certificate, has a median wall time of 13 ms on Kuhn, 7 to 136 ms on the stage games, and 1.0 s on Leduc, whose deepest subgame, with $1{,}117$ resolver sequences, takes 1.7 s. Solved back to back at $\tau = 0$ on the 214 poker configurations, the certified program is the smaller one everywhere and the HiGHS call alone takes a median of 4.7 ms on Leduc against 15.3 ms for the exact-safe program, and 1.7 ms either way on Kuhn, where both are small enough for the build and the exact certificate to dominate.

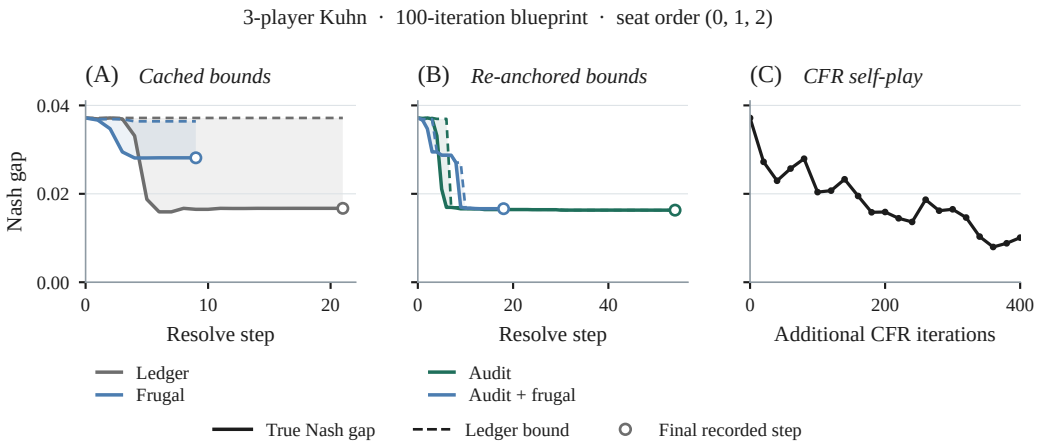


Figure 2: **Re-anchoring tightens the running certificate.** Deterministic trajectories on 3-player Kuhn from the same blueprint and seat order. **(A,B)** Solid curves show the true Nash gap, dashed curves the ledger bound Φ_t, and shading their gap. Open circles mark the final recorded step. **(C)** CFR self-play plotted against its own iteration count, so the horizontal axes of the three panels are on different scales.

Beyond the linear program. CFR$^+$ on the gadget reaches the program's value to 10^{-4} in 159 of the 160 instances, in per-family medians of 200–700 iterations and 0.2–0.3 s at $n = 3$ against 22–59 s at $n = 6$; one instance at $n = 6$ is stopped at 4×10^{-4} after 4,000 iterations. The gadget has five victim branches at $n = 6$ and every iteration traverses 2^n deals; the gadget is the formulation we would use for subgames beyond an LP solver; the experiments here do not reach that regime, and its cost includes anchor computation and the exact certificate audit. Table 4 gives the median program sizes at $\tau = 0$ over the 214 poker configurations: confining the dual variables to the subgame shrinks the program by an order of magnitude on Leduc.

Table 4: Median size of the certified program of Proposition 2 and of the whole-game exact-safe program, at $\tau = 0$ over the 214 poker configurations.

Game	Program	variables	rows	nonzeros
Kuhn (3p)	certified	36	31	180
	exact-safe	75	69	256
Leduc (3p)	certified	218	159	1,948
	exact-safe	3,111	2,981	8,984

Recovery from penalised solves. With HiGHS as the exact oracle, the secant recovery of Appendix A reproduces the direct program's gain to 1.4×10^{-6} on all 324 Kuhn and to 4.8×10^{-7} on all 960 Leduc (configuration, budget) points: the budget is inactive at 87 and 458 points and met to 10^{-9} at the remaining 237 and 502, with at most 14 and 9 doubling steps and 5 and 10 secant solves. The secant slope is a ratio of two solver values, so at 15 of the 960 points it fell outside its bracketing interval and was clamped to the nearer endpoint, by at most a relative 3.6×10^{-8}; the largest differences between the recovered and the direct gain, 1.4×10^{-6} on Kuhn and 4.8×10^{-7} on Leduc, occur at $\tau = 0$ points where the dual variable of the budget constraint times the solver's 10^{-9} feasibility tolerance is larger still, consistent with the multiplier amplifying that tolerance.

Equality instances. The slack column of Table 6 is the clipping term of Proposition 4 in every row: at each of the five configurations the reach and re-optimisation terms are exactly 0 for both victims, so there the three-term decomposition of the certificate slack reduces to its first term. Over all 2,244 (configuration, budget) points and 5,928 victim entries of Table 2 the identity holds to 8.8×10^{-16} at worst, and on the stage games, where no opponent acts before S, the reach and re-optimisation terms are exactly 0 for every victim, as Proposition 3 predicts. Best responses break ties toward the first

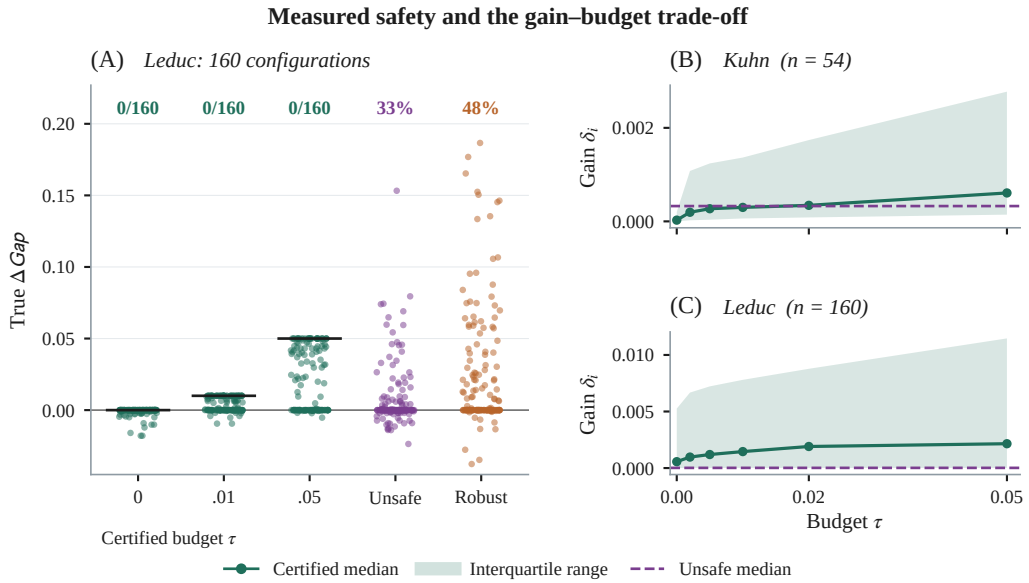


Figure 3: **A controlled Nash gap with a tunable gain. (A)** Every Leduc update in five arms; horizontal jitter separates points without changing their values. Black bars mark τ. Counts above certified arms use $\Delta\mathsf{Gap} > \tau + 10^{-8}$; percentages above baselines use $\Delta\mathsf{Gap} > 10^{-4}$. **(B,C)** Certified gain at all six measured budgets: median and interquartile range across configurations, with the unsafe median dashed. Shading is the interquartile range across configurations; connecting lines guide the eye.

action, which fixes the individual terms; their sum is invariant. Table 5 counts the victim entries at which each term is non-zero.

Table 5: Victim entries of Table 2 at which each term of the decomposition in Proposition 4 exceeds 10^{-9}, out of all victim entries on that family.

Victims	clipping	reach	re-optimisation
Kuhn (3p)	332/648	116/648	83/648
Leduc (3p)	473/1,920	409/1,920	352/1,920
Stage games	2,261/3,360	0/3,360	0/3,360

Table 6: All five instances of 3-player Kuhn (subgame after the first check, $\tau = 0.005$) at which the τ-certified resolve meets Corollary 2 to 10^{-6}. The column *cert bound* is the right-hand side of Corollary 2, $r_j(\beta') - \mathsf{Gap}(\beta) - \tau$ is each victim's distance to the cap and $K_j - D_j$ its certificate slack, one entry per victim. At T=1000, r=2 both victims sit on the cap with zero slack, so that row attains the cap identity of Theorem 2, with the three terms of Proposition 4 all zero for both victims.

blueprint	resolver	δ_i	cert bound	$r_j(\beta') - V - \tau$	$K_j - D_j$
T=100	r=1	0.01333	0.01333	-0.0037 / +0.0000	0.0037 / 0.0000
T=1000	r=0	0.00064	0.00064	-0.0000 / -0.0046	0.0000 / 0.0046
T=1000	r=1	0.00164	0.00164	-0.0044 / -0.0046	0.0044 / 0.0046
T=1000	r=2	0.00186	0.00186	+0.0000 / +0.0000	0.0000 / 0.0000
T=10000	r=1	0.00013	0.00013	-0.0029 / +0.0000	0.0029 / 0.0000