INTRANSITIVITY CAN RAISE THE MINIMUM ABSTRACTION SIZE

Anonymous authors
Paper under double-blind review

ABSTRACT

Large poker solvers group private hands into buckets before they solve. The number of buckets is a parameter set by experiment: coarse enough that the abstract game fits the solver, fine enough that the resulting strategy is not obviously exploitable. The theory of abstraction takes an abstraction as given. Lossless methods merge states that are interchangeable against every opponent, and lossy methods bound the exploitability of a coarsening already chosen. Neither answers how many buckets a game requires. The minimum also cannot be read off one equilibrium, which may distinguish hands it has no need to. We compute the exact minimum in one-card poker, a family small enough to enumerate. One player's strategy is a finite-state automaton reading the private card and then the public betting actions, so at each betting action a state is a bucket. When each post-card observation identifies one decision history, the fewest states attaining the equilibrium value equal the *joint width*, the fewest blocks of one partition of the deck that serves every decision the player faces. With one decision after the deal this is the *label-class width*; with several it can be larger, and the second player in four-card Kuhn poker needs four states although its two decisions need three and two. Sweeping the showdown relation exactly, we find a total order keeps width at most three on every deck from 3 to 40 cards, attained by a contiguous partition of card rank, with width three recurring with period ten. Reversing one comparison lifts width to 4 on five cards. Over all 456 seven-card isomorphism classes width reaches 7, one bucket per card, while three quarters of the classes need at most two. How many cards each card beats does not determine the width. Deck size alone does not set the bucket requirement: at seven cards the order needs two, the worst relation seven, others one.

1 INTRODUCTION

Large poker solvers group private hands into buckets before they solve. The number of buckets is a design choice made by experiment: coarse enough that the abstract game fits the solver, fine enough that the strategy is not obviously exploitable. The practice runs from the earliest testbed analyses (Billings et al., 1998; 2002) to the abstraction-based programs that surpassed human play in no-limit hold'em (Moravčík et al., 2017; Brown & Sandholm, 2018; 2019b), and in every one of them the number of buckets was a design choice. The one full-game solution, heads-up limit hold'em (Bowling et al., 2015), is what abstraction exists to avoid needing.

Two lines of work surround that choice. One builds the buckets: hand-designed features, clustering on equity, and distributions over future equity (Shi & Littman, 2000; Gilpin & Sandholm, 2006; 2007a; Li & Huang, 2025; Gilpin et al., 2007; Gilpin & Sandholm, 2008; Ganzfried & Sandholm, 2014), measured at scale in no-limit play (Johanson, 2013). The other prices an abstraction already given: lossless abstraction merges states provably interchangeable against every opponent strategy (Gilpin & Sandholm, 2007b), and lossy abstraction certifies the exploitability of a coarsening already chosen (Sandholm & Singh, 2012; Kroer & Sandholm, 2014; 2016; 2018). Neither line says how many buckets a game requires. The minimum also resists measurement from a single solution, which may distinguish hands no optimal strategy must distinguish.

We ask for the exact minimum: the fewest buckets that still attain the equilibrium value. In real poker this number is out of reach. Deciding whether a coarsening preserves the value means solving the

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.

coarsened game exactly, and the candidates grow as partitions of the hand space. We therefore ask it in a family small enough to enumerate, and ask what the minimum depends on. The family is one-card poker, which descends from Kuhn's two-person game (Kuhn, 1950) and is fixed by a deck and a showdown relation. One player's strategy is a finite-state automaton that reads its private card and then the public betting actions. It mixes once, before the deal, over the pure plans the machine admits. At each betting action its state is then a bucket of hands whose plans coincide, so the state count is the bucket count for the decisions that follow the deal. Infosets sharing a public history receive the same observation and can be separated only by the state. They form a *label class*. Its *width* is the fewest blocks into which the class can be cut while some strategy constant on the blocks still attains the equilibrium value (Section 3). Figure 1 walks through both on the five-card deck.

The largest width is a lower bound on the state count. When each post-card observation identifies one decision history, a condition every tree we build satisfies, the state count itself is the *joint width*. That is the fewest blocks of a single partition of the deck that serves every facing class at once (Theorem 4.2). When a seat has one facing class the two coincide, which is the case for player 1 on ninety of our ninety-two configurations, so there the identity of states and width is a theorem. With several facing classes the joint width can exceed the largest width, as it does on four-card Kuhn poker with player 2 restricted, one of the eight cases in which the seat has several facing classes. Width is then swept over the showdown relation. On a totally ordered deck it never exceeds three, from three cards to forty, attained by a threshold in card rank. Reversing showdown comparisons breaks the cap, up to one bucket per card on seven cards. Read against real poker, this gives a folklore hypothesis (Sklansky, 1999) a concrete shape: a handful of buckets where hand strength is close to a total order, and a rising requirement where blockers and competing draws leave the comparison intransitive.

We make three contributions.

- **We prove that the states a seat needs are exactly the joint width.** On every label-separated game of the family and for either seat, the fewest states attaining the equilibrium value equal the joint width, and equal the largest label-class width whenever the seat has one facing class (Theorem 4.2). We compute both sides exactly on $9 + 7 + 56 + 20 = 92$ configurations, ninety up to isomorphism, measuring the joint width directly on the eighty-eight raise-free configurations and through the theorem on the four raising games. We also exhibit the gap: with player 2 restricted, four-card Kuhn poker needs four states although its classes have widths 3 and 2 (Proposition 4.4).
- **We show that three buckets suffice on every ordered deck from 3 to 40 cards.** Width is 2 or 3 throughout, and 3 recurs exactly at the sizes congruent to $3, 5, 6, 8$ modulo 10; a contiguous partition of the rank order always attains the minimum. We reach forty cards with a linear program in $n + |P|$ variables that replaces the $2^n \times 2^n$ normal form and reproduces every recorded width (Theorem 5.1).
- **We show that intransitivity alone breaks the cap, up to one bucket per card.** Reversing a single showdown comparison lifts width to 4 on five cards, and on seven cards width reaches 7 over the 456 isomorphism classes that the 2^{21} relations reduce to, a reduction licensed because width is an isomorphism invariant. Three quarters of the classes need at most two buckets, so intransitivity makes high width possible while the relations that realise it form a thin set. We show that the score sequence, the number of cards each card beats, determines every other inexpensive summary in our predictor battery but not the width (Table 2).

2 Related Work

Compact strategies and state abstraction. Černỳ et al. (2020) introduced finite state machines as a compact representation of extensive-form strategies, with algorithms that find a good machine of a given size. Ours is the dual question, the size that the equilibrium value requires, and a lower bound on that size tells such a search where to start. Compactness has a longer history as the cost of representing and solving large games (Koller & Megiddo, 1992; Koller et al., 1994; Koller & Pfeffer, 1997), of small-support mixed strategies (Koller & Megiddo, 1996), and of equilibrium interpretation under bounded memory (Piccione & Rubinstein, 1997; Halpern & Pass, 2016). In sequential decision problems the same shape appears as state abstraction: model minimization (Dean & Givan, 1997), bisimulation (Givan et al., 2003) and its metric relaxations (Ferns et al., 2004), unified recently

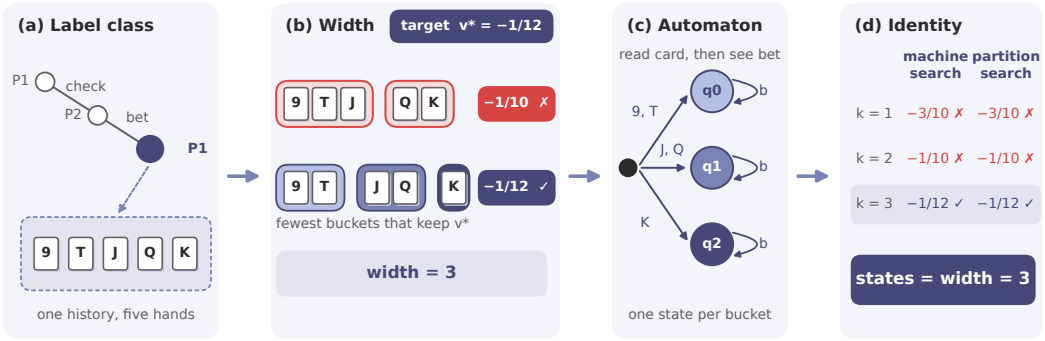


Figure 1: The three concepts on the five-card ordered deck, opening check forced ($v^\star = -1/12$). **(a)** The five hands facing a bet share one public history: a label class. **(b)** Its width is the fewest buckets that keep $v^\star$: two ($9TJ \mid QK$) fall to $-1/10$, three ($9T \mid JQ \mid K$) keep $-1/12$. **(c)** An automaton with one state per bucket reads the card, then the bet. **(d)** The search over machines and the search over partitions return the same ladder and first reach $v^\star$ at three states: $k^\star = \mathrm{w} = 3$, as Theorem 4.2 requires of a seat with one facing class.

(Abel, 2019) and carried to Markov games (Ishibashi et al., 2025). Those abstractions preserve transition structure or behavior; width preserves only the equilibrium value, so it may pool states that a structure-preserving merge must separate, and it answers the size question with a minimum.

Analytical poker models. The forced-check raise-free game is a discrete, without-replacement version of one-sided-betting poker: player 2 may bet or check on private strength and player 1 may only call or fold. Von Neumann's model (Von Neumann & Morgenstern, 2007) solves that game on a continuum of hands with a threshold structure: it bets from the top and the bottom of the range and calls from the top. Kuhn's three-card game (Kuhn, 1950) and the $[0, 1]$ games of Chen & Ankenman (2006) extend it to alternating bets and finite decks. Those solutions describe the optimal strategies. Width asks how few classes of hands an optimal strategy must distinguish, which a threshold description does not answer. The fold, mix, and call regimes of one optimal strategy are three intervals while the value is attained with two blocks for most deck sizes (Theorem 5.1).

Lossless and bounded abstraction. GameShrink (Gilpin & Sandholm, 2007b) merges nodes related by ordered game isomorphism, so that an equilibrium of the abstract game lifts back to one of the original game. Width merges under a weaker guarantee, that the equilibrium value survives. It is at most the number of isomorphism-distinct hands and often strictly smaller: on a totally ordered eleven-card deck, isomorphism distinguishes all eleven cards while width is 2. GameShrink merges as far as its equivalence criterion allows without looking at the solution. Width is the fewest blocks that survive when one is willing to look, and several incomparable partitions can attain it. The complementary line certifies the exploitability that a chosen lossy abstraction induces (Sandholm & Singh, 2012; Kroer & Sandholm, 2014; 2016; 2018), surveyed by Sandholm (2015). We fix the loss at zero and ask for the minimum size, which is a reference point for how tight such a bound could be. Abstraction practice also supplies tools: restricting the bet sizes themselves (Hawkin et al., 2011; 2012; Li et al., 2024) and translating between action spaces (Ganzfried & Sandholm, 2013). Limiting where a raise is offered builds a deeper decision tree without multiplying the deck.

Evaluating abstractions, and the benchmark. Waugh et al. (2009a) showed that making an abstraction finer can hurt: a two-sided refinement, solved and evaluated in the full game, can play worse. Our quantity is different in kind: the value of one restricted seat against an unrestricted opponent, monotone in the strategy set by definition. Johanson et al. (2013) argued that an abstraction should be judged by its own statistical properties, without solving it and measuring exploitability, and found imperfect-recall abstractions outperforming perfect-recall ones. Width is a complementary benchmark that does require the solve, and it prices the pooling that such abstractions perform. A machine that reads the exact card and then forgets it pools inside a label class, which is an imperfect-recall abstraction (Waugh et al., 2009b; Lanctot et al., 2012; Čermák, 2018). On the one-decision

games of Section 5 the same restriction equally amounts to dealing the bucket in place of the card. Appendix F places this automaton among the solution concepts that weaken perfect recall.

3 PRELIMINARIES

The game family. Two players ante one unit each and draw one card from a deck of n named cards, every ordered deal of two distinct cards equally likely. Player 1 acts first and may either check or bet one unit. A player who faces a bet may fold, call, or, when a raise is offered, raise by one unit. A fold ends the hand and the folder loses what it has contributed; a call ends it at a showdown. If player 1 checks, player 2 may check, ending the hand at a showdown, or bet one unit, which player 1 then faces. At a showdown the winner takes the other's contribution, so with each side having put in s units the payoff to player 1 is $\pm s$. The number of raises is capped at r beyond the opening bet, and restricting the public histories at which a raise is offered is an action abstraction that varies decision depth without varying n. Kuhn poker is the case $r = 0$ on three cards. The showdown relation is a parameter: a function $\mathrm{beats}(c, c')$ returning which card wins. The default is comparison of rank, a total order; we also use relations obtained by reversing chosen comparisons, which are tournaments: complete, antisymmetric, and not necessarily transitive. Payoffs to player 2 are the negatives, and the value $v^\star$ is always player 1's. Cards carry their rank as a name, with T for ten, so the deck $9\mathrm{TJQK}$ runs from the nine to the king in increasing strength. On an ordered deck of n cards we write $v^\star_n$ for the value. Appendix B lists every configuration used.

One variant matters enough to name. In *forced-check* play, player 1's opening bet is removed and only the check remains. A free opening lets player 1's opening strategy filter which cards reach the later decision; forcing the check removes that filter and sends every card there, making bucket demand a question about the whole deck.

Exact arithmetic. Every payoff and solution value is a rational number computed with exact fraction arithmetic; floating point is never used. Equilibrium values $v^\star$ (Nash Jr, 1950) come from the linear program of the zero-sum matrix game over pure strategies (Von Neumann & Morgenstern, 2007; von Stengel, 2007), solved by an exact rational simplex. Every result below is an equality between values or a claim that some partition does *not* attain one; rational arithmetic certifies both.

Automaton strategies. One seat is the *restricted* player and the other is unrestricted; a configuration together with the choice of restricted seat is a *seat-game*. Player 1 is restricted unless stated otherwise, and Section 4 repeats the identity measurements with the seats swapped. The restricted player's position in the game is a string of observations. For player 1 the first observation is its private card, read at the opening decision, and every later one is the opponent's last public action, b or r. Player 2 has no opening decision. It reads its card first, as an observation that precedes any action, and then player 1's last action, k, b, or r. The check symbol is therefore an observation for player 2 and never for player 1.

The object we minimize is built in five steps.

1. A *transition skeleton* δ with k states maps (state, observation) to a next state. Writing δ^* for the extension of δ to observation strings, an infoset I with observation string $o_1, \ldots, o_m$ is read in the state $q_\delta(I) = \delta^*(0, o_1, \ldots, o_{m-1})$; the action at I is keyed on the pair $(q_\delta(I), o_m)$, *before* the transition on o_m is applied.
2. The *admissible plans* of δ are the pure strategies constant across infosets that share their last observation and their q_δ; write S_δ for this set.
3. The player randomizes once, before the deal, over S_δ, and the *value of* δ is the value of the matrix game with rows S_δ and columns the opponent's pure strategies. Pooling gives the restricted player imperfect recall, under which ex-ante mixtures and behavioral strategies need not coincide; every value here is of the former kind. On the forced-check raise-free games of Section 5 the player has one decision per card, and there the two notions coincide.
4. The *target value* of the restricted seat is its own equilibrium value, $v^\star$ or $-v^\star$ according to the seat; "attains $v^\star$" below means attains that target, and the payoff table is the seat's.
5. $k^\star$ is the fewest states k for which some δ attains the target value. The skeleton is what the state count charges for; the plan selected before the deal and the opening decisions, whose observation is the card itself, are not charged.

Definition 3.1 (Label class)**.** The *label* of an infoset is the observation the automaton reads at it. The *label classes* are the equivalence classes of the player's infosets under equal labels.

Infosets in one class receive the same current observation and can be told apart only through the state, so an automaton can only pool infosets *inside* a label class. At player 1's opening the observation is the card itself, so opening classes are singletons. On every betting tree used here each post-card label occurs at exactly one public history of the restricted seat (Appendix B). At every later decision the class is therefore the set of infosets sharing a public betting history and differing only in the private card, a *facing class*. We call such a tree *label-separated*. The general family is wider: with a free opening and two raises player 1 faces r after bet–raise and after check–bet–raise–raise. In a facing class the state reached after reading the card partitions the hands into at most k blocks, so within it an automaton state is a card bucket. For a partition P of a class C, let $S(P)$ be the set of pure plans that are constant on every block of P and unconstrained elsewhere. The *value of* P, written $V(P)$, is the value of the matrix game whose rows are $S(P)$, the best the player can do while mixing only over plans that respect P.

Definition 3.2 (Width)**.** The *width* $\mathrm{w}(C)$ of a label class is the smallest number of blocks in a partition P of C whose value is $v^\star$, all other classes left unpooled.

We write a partition by listing its blocks separated by $|$, as in $JT \mid QK$.

A mixture μ over pure plans *pools* two infosets of C if every plan in the support of μ prescribes the same action at both; the blocks of this relation are the *behavior blocks* of μ on C. This is a statement about the plans mixed, not about action probabilities (Appendix A).

Lemma 3.3 (Width is a minimum over optimal strategies)**.** $\mathrm{w}(C)$ *is the minimum, over mixtures attaining* $v^\star$*, of the number of behavior blocks into which the mixture pools* C*.*

Width is therefore not a property of one equilibrium, which may distinguish hands it has no need to. Every width we report is obtained by exhaustive search over partitions with the value recomputed exactly for each candidate, the ordered decks beyond eleven cards excepted, where a complete reduction to pure strategies replaces the enumeration (Section 5). The state count $k^\star$ is searched separately over transition functions.

Definition 3.4 (Joint width)**.** For a label-separated tree, a partition D of the deck cuts every facing class C into the blocks $\{I \in C : \mathrm{card}(I) \in B\}$, $B \in D$. The *joint width* J is the smallest $|D|$ such that the plans constant on every one of these blocks at once, opening classes left unpooled, have value $v^\star$.

Width cuts one class and frees the rest; joint width cuts all facing classes by the same deal partition. Hence $\max_C \mathrm{w}(C) \le J$, with equality whenever the seat has a single facing class, and Section 4 shows that the automaton computes exactly J.

4 Main Results

Table 1 reports both quantities on the nine ordered games spanning three deck sizes, three betting structures, forced and free opening, and decision depths two and three. They agree in every row. On the eighty-five configurations with $k^\star > 1$, the best machine with $k^\star - 1$ states falls strictly short, so $k^\star$ is where the value is first reached. The two sides are computed by separate routines (Section 3); their agreement cross-checks the software, and on most of these games the equality is a theorem. This section proves what the automaton computes, identifies exactly when that quantity collapses to the largest width, and shows a game on which it does not. The theorem identifies the resource the state count charges for with a partition of the deck, so the sweeps of Section 5 measure bucket demand by partition search alone; the characterization does not itself become a search algorithm (Appendix D).

Lemma 4.1 (Lower bound)**.** *If a* k*-state automaton attains* $v^\star$*, then every label class has width at most* k*. Consequently* $k^\star \ge \max_C \mathrm{w}(C)$*.*

Proof. See Appendix A. Grouping the infosets of C by the state in which the machine reads their shared observation gives a partition into at most k blocks on which every admissible plan is constant. □

Table 1: On the nine ordered games, $k^\star$ and the largest label-class width agree in every row. Player 1 has one facing class in every row but the sixth and the ninth; in those two Corollary 4.3 does not apply, and the agreement is a check on the two routines.

Game	$v^\star$	depth	width	$k^\star$
Kuhn, 3 cards	$-1/18$	2	2	2
Kuhn, 4 cards	$-1/24$	2	2	2
Kuhn, 3 cards, forced	$-1/18$	2	3	3
Kuhn, 4 cards, forced	$-1/12$	2	2	2
Kuhn, 5 cards, forced	$-1/12$	2	3	3
1 raise, 3 cards	$-1/18$	2	2	2
1 raise, 3 cards, forced	$-1/18$	2	3	3
1 raise, 4 cards, forced	$-1/12$	2	2	2
2 raises, 3 cards, forced	$-1/18$	3	3	3

The automaton reads the card once, and two cards sent to the same state follow the same trajectory on every public history. The partition a machine induces on each facing class is therefore coarser than, or equal to, the deal partition its first transition defines. That is the next statement.

Theorem 4.2 (States equal the joint width)**.** *On every label-separated game of the family and for either seat, $k^\star = J$.*

Proof. See Appendix A. A k-state machine attaining $v^\star$ mixes only over plans constant, in every facing class, on the blocks of the deal partition $D = \{\{c : \delta(0, c) = q\}\}_q$, so $J \le |D| \le k$. Conversely the J-state machine that reads a card into the state of its block and holds that state on every public symbol realizes a joint witness exactly. □

Corollary 4.3 (Single facing class)**.** *If the tree is label-separated and the restricted seat has one facing class, $k^\star = \max_C \mathrm{w}(C)$.*

For player 1 this covers ninety of the ninety-two configurations. Every raise-free game has card-specific singleton opening classes and a single facing class, so on those the agreement in Table 1 and Figure 4 is automatic. The identity is tested only on the two raising games with a second facing class, where it holds. The joint width can exceed the largest width: nothing forces the witnesses of different classes to be compatible.

Proposition 4.4 (Joint width above the largest width)**.** *In Kuhn poker on four cards, $TJQK$, with player 2 restricted, the facing-bet class has width 2, with witness $JT \mid QK$, and the facing-check class width 3, with witness $J \mid K \mid QT$, yet $J = 4$: no three-block deal partition has value above 0, while player 2's equilibrium value is $1/24$.*

The two widths are the exhaustive partition search of Section 3; the joint claim is the certificate of Appendix B. It gives, for each of the six three-block partitions of the deck, a player-1 mixture against which no admissible player-2 plan earns more than the listed value. All six values are at most 0, and each is checkable without a solver. The two classes cut the cards along different lines, and one memory of the card cannot serve both.

What is measured. The identity is computed on both sides on $9 + 7 + 56 + 20 = 92$ configurations, ninety games up to isomorphism, and holds at every width the seven-card deck admits (Figure 4, Appendix B). A second routine cuts the deck directly and returns $J = k^\star$ on both seats of every raise-free configuration.

The other seat. The state count belongs to a seat. Player 2 has a single facing class on every forced-check raise-free game, so Corollary 4.3 applies there, and multiple facing classes on the six free or raising configurations. With the seats swapped on all ninety-two configurations, $k^\star$ equals the largest width on ninety-one; the exception is Proposition 4.4. The widths themselves are seat-specific: they differ between the seats on thirty-five of the ninety-two configurations, higher for player 2 on twenty-five and lower on ten, from five to two on one six-card class and from six to seven on two seven-card classes. The phenomenon carries over: with player 2 restricted, intransitive relations lift width to one bucket per card and ordered decks stay within three buckets per class (Appendix B).

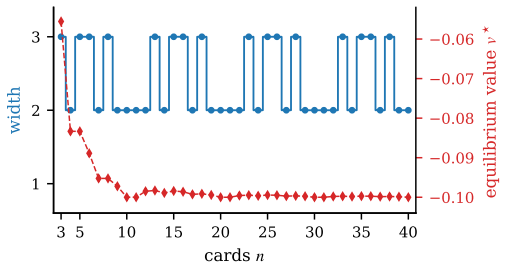


Figure 2: Width and equilibrium value on the totally ordered deck, forced check, $n = 3, \ldots, 40$. Width stays in $\{2, 3\}$ and returns to 3 with period ten while $v^\star$ oscillates within $[-1/10, -1/18]$ and equals $-1/10$ exactly at $n = 10, 11, 20, 21, 30, 31, 40$; the cap is not a plateau of the value.

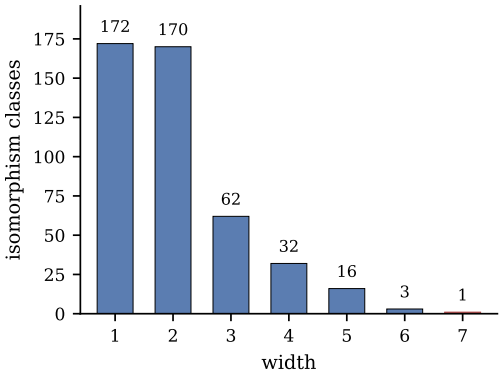


Figure 3: Width over all 456 isomorphism classes of seven-card relations. The scale is linear; three quarters of classes need at most two buckets, and width 7, shown dark, is a single class in which no two hands may share a bucket without loss.

Exact computation. On forced-check, raise-free games every value is an exact rational, computed by the linear program in $n + |P|$ variables. It replaces the $2^n \times 2^n$ normal form of the earlier sweeps and reproduces all $1024 + 56 + 456 + 9 = 1545$ recorded values and widths. The identity also fixes which side of the computation is tractable. Searching $k^\star$ directly ranges over $k^{k|\Sigma|}$ transition functions, each re-scored against $v^\star$, so the cost compounds in the state count. Searching a partition ranges over deal partitions scored jointly on the facing classes, which on a single facing class are the partitions of that class, each re-scored against the value the game already yields. In the trees we build the observation strings are short: only the start state is asked about a card, and a state is asked for a public entry only when a string transitions on it, so at most $n + k$ entries matter rather than $k(n + 2)$, and the exhaustive enumeration stays cheap (Appendix D). Raising and free-opening games go through the normal form (Appendix B). Below the width every partition is enumerated, and beyond eleven cards on the ordered decks Section 5 describes the complete reduction used. The code release ships checked records of these computations and solver-free certificates for Proposition 4.4 and the width-seven relation (Appendix B).

5 Experiments

Width is measured under two showdown regimes: a total order, where it stays bounded by three, and a tournament obtained by reversing chosen comparisons, where intransitivity can lift it.

Over the ordered deck the deck size varies by a factor of thirteen, and the equilibrium value settles towards $-1/10$, rising from $v^\star_{11}$ to $v^\star_{12} = -13/132$ (Figure 2).

Theorem 5.1 (Three buckets on an ordered deck)**.** *On the forced-check, raise-free one-card poker whose showdown is a total order, for each deck size $n = 3, \ldots, 40$, the facing-bet class has width two or three, and at least one minimum-size partition attaining $v^\star$ is an interval of card rank. Width is three exactly at $n \in \{3, 5, 6, 8, 13, 15, 16, 18, 23, 25, 26, 28, 33, 35, 36, 38\}$, the residues $3, 5, 6, 8$ modulo 10.*

Proof. The claim is a finite computation. For $n \leq 11$ the normal-form search behind Figure 2 returns an interval partition into at most three blocks of value $v^\star$, for example 9 | TJ | QK at five cards and 789 | TJQK at seven, and enumerates every partition into fewer blocks, none of which attains $v^\star$. Beyond that the value of a partition is a linear program in $n + |P|$ variables, the direct program (Appendix B). Three exact facts close each size. The one-block value is below $v^\star$. In every optimal strategy the weakest card folds and the strongest calls (Lemma A.6, Appendix A), so a two-block witness must be a pure strategy. Every pure strategy that can be optimal is enumerated exactly from the range of each call probability over the optimal face, the set of call-probability vectors attaining $v^\star$

(Appendix B). The enumeration is confirmed against all 2^n pure strategies for $n \leq 22$. When no pure strategy attains $v^\star$, an interval three-block witness does. The direct program reproduces every value and width of the normal-form route on $n \leq 11$. □

The betting tree moves width in one place. Table 1 varies the number of raises, the depth, and whether the check is forced; only the last moves width, from 2 to 3, and only at three cards. Depth does not, because the deepest decision is reached only by raising, which an optimal strategy of the depth-three game does with the strongest card alone; the class's other infosets are never reached by its own play, and its measured width is one.

The width is also the fewest distinct coordinate values of a call-probability vector on the optimal face of the direct program (Proposition A.5, Appendix A). The equivalence is constructive both ways: cards grouped by the value of an optimal vector form a partition of the same value, and a width-attaining partition admits an optimal vector constant on its blocks.

Player 1 has an interval best response to any fixed betting strategy of player 2 in the raise-free game, a contiguous partition of the rank order into fold, mix, call (Lemma A.1, Appendix A).

Not every minimum-size partition is an interval: at five cards $9J \mid T \mid QK$ attains $v^\star$ although 9 and J are separated in rank by T (Proposition A.2, Appendix A). The minimum is always *attainable* by a threshold, which is what the cap needs.

The sizes through eleven cards suggested that width settles at 2 from $n = 9$ on; it does not. Width returns to 3 at $n = 13$ and then at every size congruent to $3, 5, 6, 8$ modulo 10 up to 40. The equilibrium value is exactly $-1/10$ at $n \in \{10, 11, 20, 21, 30, 31, 40\}$, a period-ten structure that no run of two sizes reveals. The number of pure optimal strategies at the width-2 sizes runs through $1, 2, 5, 14, 42, \ldots$, the Catalan numbers (Table 5). We conjecture that width is at most three for every deck size and that the residue pattern persists.

Intransitive relations. A rotationally symmetric cyclic relation collapses width to 1 and the value to 0, since a card-independent strategy then attains the value (Appendix B). On five cards we enumerate all $2^{10} = 1024$ tournaments: the maximum width is 4, above the cap, attained at 120 relations, so one flipped edge breaks the cap. Flipping an edge between non-adjacent ranks introduces a cyclic triple, and a tournament is a total order exactly when it has none (Moon, 1968). Flipping an adjacent pair produces another total order. A concrete witness flips the pair 9 and Q: letting 9 beat Q raises the width from 3 to 4 at value $-3/23$, verified by the same search.

Lemma 5.2 (Isomorphism invariance)**.** *Let R and R' be showdown relations on the same deck and σ a bijection of the cards such that c beats c' under R exactly when $\sigma(c)$ beats $\sigma(c')$ under R'. Then the games with showdown R and R' have the same value and the same width on every label class.*

Proof. See Appendix A: relabelling by σ is a game isomorphism preserving label classes. □

Width is constant on each isomorphism class of tournaments, and all $5! = 120$ relabellings partition the 1024 five-card relations into exactly 12 classes, which licenses the sweep that follows. On seven cards there are 2^{21} tournaments, beyond reach, and 456 up to isomorphism, which is affordable.

The full seven-card sweep. Figure 3 plots width over all 456 classes, with exact counts in Appendix B. Width reaches 5, 6, then 7, the definitional ceiling, and at that relation the width-attaining partition is all singletons: no two hands may share a bucket without loss. The value there is $-2578/22015$ and the class is unique. A partition into fewer than seven blocks refines to one that merges a single pair of cards, so the 21 partitions merging one pair bound every coarser candidate. All 21 values fall below $v^\star$; the largest, merging J with K, is $-270/2303$, short by $988/7242935 \approx 1.4 \times 10^{-4}$. Rational arithmetic certifies the gap, so the width records an exact statement about this family (Appendix B).

Width does not track how far the relation is from a total order. Measuring that distance as the fewest comparisons whose reversal restores an order, the largest width on seven cards at distances 0 through 7 is $2, 4, 6, 6, 7, 3, 1, 1$. The width-7 class is four reversals from the order and fragile: no single reversal of it keeps width 7 (Table 9). The relations farthest from the order are nearly regular and measure width 1, with the cyclic relation at the limit, where card-independent play is provably optimal. High width needs a relation that breaks the order in a particular way, and not too far from it.

Table 2: Cheap predictors of width. "classes" is the number of distinct values a predictor takes, "collide" how many hold more than one width. Columns range over all 1024 five-card relations and over the 56 and 456 isomorphism classes. Folding five-card relations to their classes changes no count: every predictor and the width are isomorphism invariants (Lemma 5.2). Only the isomorphism class is exact.

	5 cards		6 cards		7 cards	
predictor	**classes**	**collide**	**classes**	**collide**	**classes**	**collide**
score sequence	9	1	22	7	59	27
cyclic triples	6	3	9	6	15	12
isomorphism class	12	0	56	0	456	0

Both ends of the range are measured, from the ordered deck to a seven-card relation where no pooling is possible. The frontier $4 \to 5 \to 7$ is a maximum over each deck's relations; no fixed construction tracks it across deck sizes (Table 6, Appendix B).

Three quarters of the isomorphism classes need at most two buckets, and one class needs seven (Table 7). Intransitivity makes high width *possible*, and the relations that realise it form a thin, identifiable set; the count weights every isomorphism class equally.

Tolerance. Width asks for the value exactly, so we also record the *approximate width* $w_\epsilon(C)$, the fewest blocks whose best restricted value comes within ϵ of $v^\star$. It is read off the ladder $L(k)$ of the best value over k-block partitions. Figure 5 (Appendix B) plots the loss $v^\star - L(k)$ against k for the twenty classes of width at least five and the ordered decks. The distribution over the 456 seven-card classes and the ordered decks is in Table 10 (Appendix B); on the ordered rows the ladders are exact through ten cards and interval-restricted beyond. At $\epsilon = 10^{-3}$ eleven classes still need at least five buckets and the largest demand is 5, so the excess over the ordered cap survives a tolerance on the value. The width-seven relation is the brittle instance: four buckets come within 10^{-3} of its value. The typical class is robust: among the 284 classes of width at least two, dropping one bucket costs a median 4.8% of the value.

The score sequence does not determine width. Table 2 tests invariants of the showdown tournament over all 1024 five-card relations and over the 56 six-card and 456 seven-card isomorphism classes. The other cheap statistics are functions of the score sequence: the cyclic-triple count by Proposition A.3, and the strong-component statistics and the component count, whose boundaries sit where the sorted scores sum to $\binom{k}{2}$ (Proposition A.4, Appendix A). They collide wherever the score sequence does. The battery asks whether the score sequence determines width. It does not. The sequence comes closest on five cards, where one value class holds two widths. It degrades as the deck grows: of 22 six-card sequences 7 hold more than one width, and on seven cards 2223345 holds $\{1, 2, 3, 4, 5\}$ while 2223336 holds $\{1, 6, 7\}$. The same summary describes both a one-bucket relation and the unique seven-bucket relation.

6 Conclusion

Bucket count is a budget that practitioners fix by experiment. On label-separated one-card poker it becomes an exact quantity: the minimum abstraction size of a seat is the joint width, the fewest blocks of one deal partition that preserve that seat's equilibrium value at every decision it faces (Theorem 4.2). The joint width carries a cost that the width of a single decision class misses. When one partition must serve several decisions, the cheapest partition for each need not be available at once: four-card Kuhn poker with player 2 restricted needs four states although its classes fit in three and two (Proposition 4.4). Measured exactly, the quantity separates two regimes. Every ordered deck from three to forty cards needs at most three buckets, while on seven cards the widest relation needs one bucket per card. A smaller deck can therefore demand the finest partition there is, and the requirement is a property of the showdown relation itself (Appendix C).

AI USE STATEMENT

In this work, we used generative AI tools for drafting and editing the text and for assisting with the experiments. We have reviewed all AI-assisted work and take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work is a theoretical and computational study of finite games; it involves no human subjects, no data release, and no foreseeable harmful applications or conflicts of interest.

REPRODUCIBILITY STATEMENT

All definitions, theorems, and proofs appear in the main text and the appendix (Appendices A, B, and D). Every value reported is computed with exact rational arithmetic by exhaustive search, with no convergence tolerance. The anonymous supplementary code package runs every measurement of the paper, one program per measurement in pipeline order, each reading the JSON written by the programs it depends on but recomputing its numbers from the game definitions. It also ships the per-configuration records and the solver-free certificates the appendix cites.

REFERENCES

David Abel. A theory of state abstraction for reinforcement learning. In *AAAI*, pp. 9876–9877. AAAI Press, 2019.

Robert J Aumann, Sergiu Hart, and Motty Perry. The absent-minded driver. *Games and Economic Behavior*, 20(1):102–116, 1997.

Ratip Emin Berker, Emanuel Tewolde, Ioannis Anagnostides, Tuomas Sandholm, and Vincent Conitzer. The value of recall in extensive-form games. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 13631–13640, 2025.

Darse Billings, Denis Papp, Jonathan Schaeffer, and Duane Szafron. Poker as a testbed for ai research. In *Conference of the Canadian Society for Computational Studies of Intelligence*, pp. 228–238. Springer, 1998.

Darse Billings, Aaron Davidson, Jonathan Schaeffer, and Duane Szafron. The challenge of poker. *Artificial Intelligence*, 134(1-2):201–240, 2002.

Michael Bowling, Neil Burch, Michael Johanson, and Oskari Tammelin. Heads-up limit hold'em poker is solved. *Science*, 347(6218):145–149, 2015.

George W Brown. Iterative solution of games by fictitious play. *Act. Anal. Prod Allocation*, 13(1): 374, 1951.

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. Solving imperfect-information games via discounted regret minimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 1829–1836, 2019a.

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

Noam Brown, Adam Lerer, Sam Gross, and Tuomas Sandholm. Deep counterfactual regret minimization. In *International conference on machine learning*, pp. 793–802. PMLR, 2019.

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.

Colin F Camerer, Teck-Hua Ho, and Juin-Kuan Chong. A cognitive hierarchy model of games. *The quarterly journal of economics*, 119(3):861–898, 2004.

Jiří Čermák. *Solving Extensive-Form Games Using Imperfect Recall Abstraction*. PhD thesis, Czech Technical University, 2018.

Jirí Cermák, Branislav Bosanský, Karel Horák, Viliam Lisý, and Michal Pechoucek. Approximating maxmin strategies in imperfect recall games using a-loss recall property. *Int. J. Approx. Reason.*, 93:290–326, 2018.

Jakub Černỳ, Branislav Bosanskỳ, and Bo An. Finite state machines play extensive-form games. In *Proceedings of the 21st ACM Conference on Economics and Computation*, pp. 509–533, 2020.

Bill Chen and Jerrod Ankenman. *The mathematics of poker*. ConJelCo LLC Pittsburgh, PA, 2006.

Thomas Dean and Robert Givan. Model minimization in markov decision processes. In *Proceedings of the Fourteenth National Conference on Artificial Intelligence (AAAI-97)*, pp. 106–111. AAAI Press, 1997.

Gabriele Farina, Christian Kroer, Noam Brown, and Tuomas Sandholm. Stable-predictive optimistic counterfactual regret minimization. In *International conference on machine learning*, pp. 1853–1862. PMLR, 2019.

Gabriele Farina, Christian Kroer, and Tuomas Sandholm. Faster game solving via predictive blackwell approachability: Connecting regret matching and mirror descent. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 5363–5371, 2021.

Norm Ferns, Prakash Panangaden, and Doina Precup. Metrics for finite markov decision processes. In David Maxwell Chickering and Joseph Y. Halpern (eds.), *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence (UAI 2004)*, pp. 162–169. AUAI Press, 2004.

Sam Ganzfried and Tuomas Sandholm. Action translation in extensive-form games with large action spaces: Axioms, paradoxes, and the pseudo-harmonic mapping. In *IJCAI*, pp. 120–128, 2013.

Sam Ganzfried and Tuomas Sandholm. Potential-aware imperfect-recall abstraction with earth mover's distance in imperfect-information games. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 28, 2014.

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.

Andrew Gilpin and Tuomas Sandholm. A texas hold'em poker player based on automated abstraction and real-time equilibrium computation. In *Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems*, pp. 1453–1454, 2006.

Andrew Gilpin and Tuomas Sandholm. Better automated abstraction techniques for imperfect information games, with application to texas hold'em poker. In *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*, pp. 1–8, 2007a.

Andrew Gilpin and Tuomas Sandholm. Lossless abstraction of imperfect information games. *Journal of the ACM (JACM)*, 54(5):25–es, 2007b.

Andrew Gilpin and Tuomas Sandholm. Expectation-based versus potential-aware automated abstraction in imperfect information games: An experimental comparison using poker. In *AAAI*, pp. 1454–1457, 2008.

Andrew Gilpin, Tuomas Sandholm, and Troels Bjerre Sørensen. Potential-aware automated abstraction of sequential games, and holistic equilibrium analysis of texas hold'em poker. In *Proceedings of the National Conference on Artificial Intelligence*, volume 22, pp. 50, 2007.

Robert Givan, Thomas Dean, and Matthew Greig. Equivalence notions and model minimization in markov decision processes. *Artificial Intelligence*, 147(1-2):163–223, 2003.

Joseph Y Halpern and Rafael Pass. Sequential equilibrium in games of imperfect recall. In *KR*, pp. 278–287, 2016.

Joseph Y Halpern and Leandro C Rêgo. Extensive games with possibly unaware players. In *Proceedings of the fifth international joint conference on autonomous agents and multiagent systems*, pp. 744–751, 2006.

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

John Hawkin, Robert Holte, and Duane Szafron. Automated action abstraction of imperfect information extensive-form games. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 25, pp. 681–687, 2011.

John Hawkin, Robert Holte, and Duane Szafron. Using sliding windows to generate action abstractions in extensive-form games. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 26, pp. 1924–1930, 2012.

Johannes Heinrich and David Silver. Deep reinforcement learning from self-play in imperfect-information games. *arXiv preprint arXiv:1603.01121*, 2016.

Johannes Heinrich, Marc Lanctot, and David Silver. Fictitious self-play in extensive-form games. In *International conference on machine learning*, pp. 805–813. PMLR, 2015.

Hiroki Ishibashi, Kenshi Abe, and Atsushi Iwasaki. Approximate state abstraction for markov games. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 17555–17563, 2025.

Michael Johanson. Measuring the size of large no-limit poker games. *arXiv preprint arXiv:1302.7008*, 2013.

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.

Daphne Koller and Nimrod Megiddo. The complexity of two-person zero-sum games in extensive form. *Games and economic behavior*, 4(4):528–552, 1992.

Daphne Koller and Nimrod Megiddo. Finding mixed strategies with small supports in extensive form games. *International Journal of Game Theory*, 25(1):73–92, 1996.

Daphne Koller and Avi Pfeffer. Representations and solutions for game-theoretic problems. *Artificial intelligence*, 94(1-2):167–215, 1997.

Daphne Koller, Nimrod Megiddo, and Bernhard Von Stengel. Fast algorithms for finding randomized strategies in game trees. In *Proceedings of the twenty-sixth annual ACM symposium on Theory of computing*, pp. 750–759, 1994.

Christian Kroer and Tuomas Sandholm. Extensive-form game abstraction with bounds. In *Proceedings of the fifteenth ACM conference on Economics and computation*, pp. 621–638, 2014.

Christian Kroer and Tuomas Sandholm. Imperfect-recall abstractions with bounds in games. In *Proceedings of the 2016 ACM Conference on Economics and Computation*, pp. 459–476, 2016.

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

Harold W. Kuhn. A simplified two-person poker. In Harold W. Kuhn and Albert W. Tucker (eds.), *Contributions to the Theory of Games, Volume I*, volume 24 of *Annals of Mathematics Studies*, pp. 97–103. Princeton University Press, 1950.

Nicolas S. Lambert, Adrian Marple, and Yoav Shoham. On equilibria in games with imperfect recall. *Games Econ. Behav.*, 113:164–185, 2019.

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.

Marc Lanctot, Richard G. Gibson, Neil Burch, Martin Zinkevich, and Michael Bowling. No-regret learning in extensive-form games with imperfect recall. In *Proceedings of the 29th International Conference on Machine Learning (ICML)*, pp. 65–72. icml.cc / Omnipress, 2012.

Boning Li and Longbo Huang. Efficient online pruning and abstraction for imperfect information extensive-form games. In *The Thirteenth International Conference on Learning Representations*, 2025.

Boning Li and Longbo Huang. Real-time parallel counterfactual regret minimization. *arXiv preprint arXiv:2605.19928*, 2026.

Boning Li, Zhixuan Fang, and Longbo Huang. Rl-cfr: improving action abstraction for imperfect information extensive-form games with reinforcement learning. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 27752–27770, 2024.

Boning Li, Yu Chen, and Longbo Huang. Correlated chance sampling for monte carlo counterfactual regret minimization. *arXiv preprint arXiv:2607.27035*, 2026.

Richard D McKelvey and Thomas R Palfrey. Quantal response equilibria for normal form games. *Games and economic behavior*, 10(1):6–38, 1995.

Richard D McKelvey and Thomas R Palfrey. Quantal response equilibria for extensive form games. *Experimental economics*, 1(1):9–41, 1998.

John W Moon. *Topics on tournaments*. Holt, Rinehart and Winston, 1968.

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.

Rosemarie Nagel. Unraveling in guessing games: An experimental study. *The American economic review*, 85(5):1313–1326, 1995.

John F Nash Jr. Equilibrium points in n-person games. *Proceedings of the national academy of sciences*, 36(1):48–49, 1950.

Akira Okada. Complete inflation and perfect recall in extensive games. *International Journal of Game Theory*, 16(2):85–91, 1987.

Michele Piccione and Ariel Rubinstein. On the interpretation of decision problems with imperfect recall. *Games and Economic Behavior*, 20(1):3–24, 1997.

I Romanovskii. Reduction of a game with complete memory to a matrix game. *Soviet Mathematics*, 3:678–681, 1962.

Tuomas Sandholm. Abstraction for solving large incomplete-information games. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29, 2015.

Tuomas Sandholm and Satinder Singh. Lossy stochastic game abstraction with bounds. In *Proceedings of the 13th ACM Conference on Electronic Commerce*, pp. 880–897, 2012.

R Selten. Reexamination of the perfectness concept for equilibrium points in extensive games. *International Journal of Game Theory*, 4(1):25–55, 1975.

Jiefu Shi and Michael L Littman. Abstraction methods for game theoretic poker. In *International Conference on Computers and Games*, pp. 333–345. Springer, 2000.

David Sklansky. *The theory of poker*. Two plus two publishing LLC, 1999.

Dale O. Stahl and Paul W. Wilson. On players' models of other players: Theory and experimental evidence. *Games and Economic Behavior*, 10(1):218–254, 1995. doi: 10.1006/game.1995.1031.

Oskari Tammelin. Solving large imperfect information games using cfr+. *arXiv preprint arXiv:1407.5042*, 2014.

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

Emanuel Tewolde, Caspar Oesterheld, Vincent Conitzer, and Paul W Goldberg. The computational complexity of single-player imperfect-recall games. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, pp. 2878–2887, 2023.

Emanuel Tewolde, Brian Hu Zhang, Caspar Oesterheld, Manolis Zampetakis, Tuomas Sandholm, Paul Goldberg, and Vincent Conitzer. Imperfect-recall games: equilibrium concepts and their complexity. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pp. 2994–3004, 2024.

John Von Neumann and Oskar Morgenstern. Theory of games and economic behavior: 60th anniversary commemorative edition. In *Theory of games and economic behavior*. Princeton university press, 2007.

Bernhard von Stengel. Equilibrium computation for two-player games in strategic and extensive form. In *Algorithmic Game Theory*, pp. 53–78. Cambridge University Press, 2007.

Kevin Waugh, David Schnizlein, Michael H Bowling, and Duane Szafron. Abstraction pathologies in extensive games. In *Proceedings of the 8th International Conference on Autonomous Agents and Multiagent Systems*, volume 2, pp. 781–788, 2009a.

Kevin Waugh, Martin Zinkevich, Michael Johanson, Morgan Kan, David Schnizlein, and Michael H Bowling. A practical use of imperfect recall. In *SARA*, 2009b.

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

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 TECHNICAL PROOFS

Why a threshold is the natural shape. In the raise-free game player 1 has an interval best response to any fixed betting strategy of player 2 (Lemma A.1): below a fold boundary a card folds, above a call boundary it calls, and on the interval between them every mixture is optimal. An interval partition encodes exactly these three regimes, a contiguous partition of the rank order into fold, mix, call. The lemma concerns a best response to one opponent strategy, and it motivates the interval witnesses that Theorem 5.1 reports without proving that one must exist; the cap itself rests on the computation.

Lemma A.1 (The interval best response)**.** *On a totally ordered deck, at the facing-bet decision of a raise-free game, where player 1 may only call or fold, fix any mixed betting strategy β of player 2, with $\beta(d)$ the probability that player 2 bets when holding card d, and $B = \sum_d \beta(d) > 0$. Assume $\beta(c) < B$ for every card c, so that the conditioning below is well-posed. Conditional on player 1 holding c and facing a bet, the continuation after a check with c whether or not player 1's own play checks there, let*

$$C(c) = \frac{2\left(\sum_{d<c}\beta(d) - \sum_{d>c}\beta(d)\right)}{B - \beta(c)}$$

be the value of calling, so that calling ties folding exactly when $C(c) = -1$. Define $S(c) = 4\sum_{d<c}\beta(d) + \beta(c)$. Then S is non-decreasing in card rank, fold is strictly best for c exactly when $S(c) < B$, calling is strictly best exactly when $S(c) > B$, and the two tie exactly when $S(c) = B$.

Proof. Conditional on facing a bet, player 2's card is $d \neq c$ with probability $\beta(d)/(B - \beta(c))$. A call wins 2 against every bet card below c and loses 2 against every bet card above c, so its expectation is $C(c)$; folding always yields -1, the contribution already committed. Calling is therefore strictly best, indifferent, or strictly worse than folding as $C(c)$ is greater than, equal to, or less than -1. With $N(c) = \sum_{d<c}\beta(d) - \sum_{d>c}\beta(d)$ a direct expansion gives $2N(c) + (B - \beta(c)) = S(c) - B$, so the strict fold set is $\{c : S(c) < B\}$ and the strict call set is $\{c : S(c) > B\}$. Finally $S(c+1) - S(c) = 3\beta(c) + \beta(c+1) \geq 0$, so S is non-decreasing: the fold set is a prefix of the rank order, the call set a suffix, and the indifference set $\{c : S(c) = B\}$ the interval between them, on which any mixture is optimal. The hypothesis $\beta(c) < B$ is what makes the conditioning well-posed: $\beta(c) = B$ means c is the only card player 2 ever bets, so player 1 holding c faces a bet with probability zero and its action there is irrelevant to the value; the formula $S(c) - B = 2N(c) + (B - \beta(c))$ still holds, with $N(c) = 0$ and $B - \beta(c) = 0$ it places c at $S(c) = B$, where any action is optimal, so adjoining that degenerate case changes nothing. □

Proposition A.2 (Not every minimizer is a threshold)**.** *On the forced-check, raise-free game over the ordered five-card deck* 9TJQK*, the facing-bet class has minimum-size partitions attaining $v^\star$ that are not intervals of card rank; the partition $9J \mid T \mid QK$ attains $v^\star$ although* 9 *and* J *are separated in rank by* T*.*

Proof. The minimum size is three: every partition of the facing-bet class into one or two blocks is enumerated and none attains $v^\star$. Enumerating the three-block partitions finds five that attain $v^\star$, three intervals and two not; each is re-scored by an exact linear program, so the non-interval members attain $v^\star$ as exact rationals. The partition $9J \mid T \mid QK$ attains $v^\star$ although 9 and J are separated in rank by T, which is the claimed witness. □

Proposition A.3 (Three-cycle count)**.** *In a tournament on n cards, the number of cyclic triples is $\binom{n}{3} - \sum_v \binom{d_v}{2}$, where d_v is the number of cards that v beats (Moon, 1968). In particular it is a function of the score sequence (d_v).*

Proof. A triple is transitive exactly when one member beats both others, so summing $\binom{d_v}{2}$ over the unique winner counts the transitive triples and the cyclic triples are the remainder. □

Proposition A.4 (Strong components from the score sequence)**.** *Let $d_1 \leq \cdots \leq d_n$ be the score sequence of a tournament in non-decreasing order. The indices k with $\sum_{i \leq k} d_i = \binom{k}{2}$ are exactly the cumulative sizes of the strongly connected components listed from the one that loses to all others upward. The number of components, their sizes, and the largest size are therefore functions of the score sequence, and so is every combination in Table 8.*

Proof. For any set K of k cards, $\sum_{v \in K} d_v \geq \binom{k}{2}$, since the $\binom{k}{2}$ games inside K each contribute one win to the sum, with equality exactly when no card of K beats a card outside K. The condensation of a tournament is a transitive tournament on its strong components, so the components are linearly ordered, every card in a lower component loses to every card in a higher one, and the score of every card in a lower component is below the score of every card in a higher one. Hence the k lowest-scoring cards form a union of the lowest components whenever k is a cumulative component size, and then equality holds; conversely, equality for the k lowest-scoring cards says that this set beats nothing outside itself, so it is closed under the beats relation and is a union of lowest components, which makes k a cumulative component size. The three component statistics are read off these indices, and the three-cycle count is a function of the score sequence by Proposition A.3. □

Proof of Lemma 5.2. Relabelling the cards by σ is an isomorphism from the game with showdown R to the game with showdown R': it preserves every payoff and maps label classes to label classes. The facing classes, whose labels are public betting symbols, are mapped onto the facing classes with the same labels; the opening classes are singletons, mapped among themselves, with width one in both games. It therefore carries optimal strategies to optimal strategies and width-attaining partitions to width-attaining partitions, class by class. □

Width is a minimum over optimal strategies. The following lemma justifies measuring width by partition search: reading behaviors off a single equilibrium overcounts whenever that equilibrium distinguishes hands it has no need to. The natural shortcut is to solve the game once and count the distinct behaviors the resulting equilibrium uses inside each class. An infoset the player's own play never reaches contributes nothing to the value regardless of the behavior assigned there, so it can join any block for free, yet a particular equilibrium may still give it a behavior of its own. The consequence is concrete in Section 5: the cap need not be visible if one reads behaviors off a single equilibrium, because the overcount can hide it.

The one-state machine of Section 3 makes the pooling relation explicit. It has $q_\delta = 0$ at every infoset, so its admissible plans call a bet with every card or fold with every card. Mixing the always-call plan with weight $\frac{1}{2}$ and the always-fold plan with weight $\frac{1}{2}$ calls with probability $\frac{1}{2}$ on each of the three cards, and both plans in the support are constant on the facing class, so the mixture pools the class into one behavior block.

Proof of Lemma 3.3. The width-attaining partition is realized by a strategy constant on each of its blocks that attains $v^\star$, so the value is reached with $\mathrm{w}(C)$ blocks and the least count is at most $\mathrm{w}(C)$. Conversely, let a mixture μ attain $v^\star$ and pool C into t behavior blocks. Every plan in the support of μ is constant on each block by the definition of pooling, so μ mixes only over plans in $S(P)$ for the partition P of C into those t blocks, and the value of P is at least the value of μ, which is $v^\star$; it cannot exceed $v^\star$. Hence $\mathrm{w}(C) \leq t$, and taking t minimal gives the reverse inequality. □

Width reads the optimal face. On a forced-check, raise-free game a strategy is a call-probability vector x, the value $V(x)$ of Appendix B is concave in x, and the width question becomes a question about V's optimal face.

Proposition A.5 (Width as the distinct coordinate values of the optimal face)**.** *On a forced-check, raise-free game the value of any partition P of the facing class is $V(P) = \max\{V(x) : x \text{ constant on the blocks of } P\}$, and the facing-class width is $\min_{x \in F^\star} \big|\{x_c : c \text{ a card}\}\big|$, the fewest distinct coordinate values of a call-probability vector on the optimal face $F^\star = \{x : V(x) = v^\star\}$.*

Proof. Against every pure strategy of player 2 the payoff of a mixture over pure plans depends on the plans only through the call probability x_c it induces at each card, because one hand gives player 1 one decision; so a mixture and its marginal vector x have the same value. Every mixture over $S(P)$ has a marginal constant on the blocks of P, and conversely a vector x constant on the blocks is the marginal of the mixture that draws the action of each block independently, calling with probability the value of x on that block. This gives the first claim. For the second, write $t(x)$ for the number of distinct values of x and group the cards by value, giving a partition into $t(x)$ blocks on which x is constant, whose value is at least $V(x)$, which is $v^\star$ for x on the face, so the width is at most $\min_{F^\star} t(x)$. Conversely a width-attaining partition has an optimal vector constant on its blocks, which has at most w distinct values, so the minimum is at most the width. □

Lemma A.6 (The ends of the deck)**.** *On the forced-check, raise-free game over a totally ordered deck of $n \geq 3$ cards, $v^\star \leq -\frac{2n-5}{3n(n-1)} < 0$, and every optimal call vector has $x_c = 0$ at the weakest card and $x_c = 1$ at the strongest.*

Proof. Let player 2 bet with the strongest card, bet with the weakest card with probability $\frac{1}{3}$, and check otherwise; measure player 1's payoff on each deal against the showdown it would get if player 2 checked, which sums to zero. A middle card faces a bet from the strongest card with weight 1 and from the weakest with weight $\frac{1}{3}$, and cannot tell them apart: calling changes the payoff by $-1 + \frac{1}{3}$, folding by $0 - \frac{2}{3}$, both $-\frac{2}{3}$. The strongest card faces only the bluff and gains $\frac{1}{3}$ by calling; the weakest faces only the strong bet and gains nothing. So every reply earns at most $\big(-\frac{2}{3}(n-2)+\frac{1}{3}\big)/(n(n-1))$, which is the bound and is negative for $n \geq 3$. Now let $\beta^\star$ be an optimal betting strategy of player 2 and x an optimal call vector; the pair is an equilibrium, so x earns exactly $v^\star$ against $\beta^\star$ and no vector earns more. If $\beta^\star$ bet with no card but the strongest, the best reply would earn 0; with no card but the weakest, at least 0; both exceed $v^\star$, so $\beta^\star$ bets with some card other than each end, and each end faces a bet with positive probability. Facing a bet the strongest card wins 2 by calling and loses 1 by folding, so $x_c < 1$ there would let a change of that coordinate alone earn more than $v^\star$ against $\beta^\star$; hence $x_c = 1$. The weakest card loses 2 by calling and 1 by folding, so $x_c = 0$ by the same argument. □

The identity. Section 4 delegates three proofs here. Every deal partition of size J whose value is $v^\star$ is called a *joint witness*.

Proof of Lemma 4.1. Let δ be a k-state skeleton attaining $v^\star$. Group the infosets of C by the state $q_\delta(I)$ in which the machine reads the shared observation; this is a partition P of C into at most k blocks. Every admissible plan in S_δ prescribes one action per block, so $S_\delta \subseteq S(P)$ and the value of P is at least the value of δ, which is $v^\star$; it cannot exceed $v^\star$. Hence $\mathrm{w}(C) \leq k$ by definition. □

Proof of Theorem 4.2. Label separation is used once: it makes every post-card label class a facing class, the infosets of one public history, so that a deal partition cuts classes into blocks on which an automaton that has the same state at both cards must prescribe the same action, and so that the machine built below, which holds its state on every public symbol, is never asked for two actions at one label. Let A be a k-state automaton attaining $v^\star$ and let $D = \{\{c : \delta(0, c) = q\} : q\}$ be the partition of the deck by the state reached after the card is read, which has at most k blocks. Every later observation is a public symbol, so two cards in one block of D are in the same state at every public history and every pure plan A admits is constant, in every facing class, on the blocks that D cuts; opening classes are singletons because the opening observation is the card itself. The plans A mixes therefore lie in the set that defines the value of D, so that value is at least $v^\star$, hence equal to it, and $J \leq |D| \leq k$. Taking $k = k^\star$ gives $J \leq k^\star$. Conversely let D^* be a deal partition of size J whose value is $v^\star$, and build a J-state automaton: index the states by the blocks of D^*, identify the start state with the state of one block (a card transition is taken only from the start state, so the others are never traversed), set $\delta(0, c) = i$ for $c \in B_i$, and $\delta(q, \sigma) = q$ on every public symbol. Its admissible plans are exactly the plans constant on the blocks D^* cuts, so it attains $v^\star$ and $k^\star \leq J$. For player 2 the card is the first observation and the argument is unchanged. □

Proof of Corollary 4.3. With one facing class C, a deal partition cuts only C, so the value of D is the value of the partition of C it induces and $J = \mathrm{w}(C)$; opening classes are singletons and have width one. Theorem 4.2 gives the rest. □

B Scope of the Measurements

This appendix records what the results are stated over. Table 3 separates the measured quantities by how each is obtained; the paragraphs below give the details.

The trees in full. Every number is measured on the one-card poker of Section 3, restricting player 1 unless the seat is named. The nine ordered games of Table 1 are: Kuhn poker ($r = 0$, free opening) on decks JQK and $TJQK$; its forced-check version on JQK, $TJQK$, $9TJQK$; one raise ($r = 1$, offered at every history the cap allows) on JQK with free opening and on JQK, $TJQK$ with the

Table 3: What each quantity rests on. A checked record is re-solved by the exact program before writing, one per configuration; the joint widths and the ladders carry a record of their own, and the two solver-free certificates are the width-seven one and the Proposition 4.4 one.

quantity	games	how obtained	record
values $v^\star$	the 92 configurations	exact rational simplex on the normal form; the direct $n+\lvert P\rvert$ program re-solves every forced-check, raise-free value	checked records
class widths	the 92 configurations, both seats	exhaustive partition search with exact re-scoring; interval witnesses from twelve ordered cards on	checked records
machine minima $k^\star$	the 92 configurations, both seats	exhaustive enumeration of realizable state assignments	checked records
joint width J	the 88 raise-free configurations, both seats (176 seat-games)	independent search over deal partitions, every class re-scored at once	joint-width
joint width of the four raising games	4 configurations, both seats	inferred through Theorem 4.2	none
approximate ladders	the 456 seven-card classes; the ordered decks of 3 to 16 cards	every partition to ten cards, interval partitions from eleven	ladder

Table 4: The equilibrium value $v^\star$ and width of the ordered deck for $n = 3, \ldots, 11$, the sizes of the normal-form sweep; Table 5 continues to $n = 40$ and Figure 2 plots both.

cards n	3	4	5	6	7
$v^\star$	$-\frac{1}{18}$	$-\frac{1}{12}$	$-\frac{1}{12}$	$-\frac{4}{45}$	$-\frac{2}{21}$
width	3	2	3	3	2
cards n	8	9	10	11	
$v^\star$	$-\frac{2}{21}$	$-\frac{7}{72}$	$-\frac{1}{10}$	$-\frac{1}{10}$	
width	3	2	2	2	

check forced. The ninth is two raises on JQK with the check forced, the raise offered only after the public histories check–bet and check–bet–raise, so player 1 can act three times. Every other game in the paper is forced-check Kuhn poker ($r = 0$) on a deck of n cards, named by rank (the names are immaterial: only the showdown relation enters the payoffs). The cap sweep uses the total order; the five-card sweep and the nested families use the total order with listed comparisons reversed; and the six- and seven-card class sweeps use a tournament given as an edge set. Player 1's infosets are (card, public history); its label is the card at the opening decision and the opponent's last action, b or r, afterwards. When player 2 is restricted its observation string is the card, then player 1's last action, k, b, or r. The state after the card read is the deal partition, and the transition on the first public symbol is taken from that state. The label of each infoset is that symbol, so the action at an infoset with observation σ is keyed on the state *before* $\delta(q, \sigma)$ is applied. The convention holds for either seat, and the state-assignment enumeration uses it, so the facing-check and facing-bet infosets holding one card share the pre-symbol state and lie in different classes. Player 2's payoff table is its own, and the value it attains is compared with $-v^\star$. The cap is an exact statement about this family. Leduc hold'em is outside what exact enumeration reaches: the normal-form payoff matrix has 2^g rows in the number of groups being distinguished, so those tables stop being buildable at roughly twenty groups, and the eleven-card arm of Figure 2 spends 34 minutes on a 2048×2048 table. On the forced-check raise-free games the same value is a linear program in $n + |P|$ variables (below), which is how the ordered deck is carried to forty cards; the normal-form route remains the one that handles raises and free openings. Appendix C draws out what these exact statements suggest for large-solver practice.

The cap numbers. Table 4 reports the values behind Figure 2.

The width sequence is $3, 2, 3, 3, 2, 3, 2, 2, 2$: bounded by three and not monotone. Deck size moves width both ways, never past three.

The direct linear program. On a forced-check raise-free game player 1 has one decision per card, whether to call the bet, so a strategy is a vector $x \in [0,1]^n$ of call probabilities and ex-ante mixing adds nothing. With $s_{cd} = +1$ if c beats d and -1 otherwise, a bet by player 2 with card d shifts player 1's unnormalized payoff by $D_d(x) = \sum_{c \neq d}\big[-1 - s_{cd} + (1 + 2s_{cd})x_c\big]$ relative to a check, and since check–check pays zero in aggregate the value of x is $V(x) = \frac{1}{n(n-1)} \sum_d \min\{0, D_d(x)\}$. A partition P of the deck identifies the x_c inside each block. Maximizing V over the block variables is a linear program with $n + |P|$ variables (one auxiliary $t_d \leq \min\{0, D_d\}$ per opponent card and one call probability per block, all bounded in $[0,1]$ or shifted to be non-negative) and $2n + |P|$ constraints, solved by the same exact rational simplex. This implementation shares nothing with the normal-form route but the simplex. Recomputing the equilibrium value and the exact width of every recorded relation with it, all 1024 five-card relations, the 56 six-card and 456 seven-card classes, and the ordered decks $n = 3, \ldots, 11$, reproduces every value and every width: 1545 of 1545 agree, in 37 seconds on eight cores, against hours for the normal-form sweeps.

The ordered deck to forty cards. With the direct program the ordered-deck width is exact for $n = 3, \ldots, 40$ (Table 5), by the following procedure. (i) The one-block value is computed and is below $v^\star$. (ii) For each card c the range $[\ell_c, u_c]$ of its call probability over the set of optimal strategies is computed exactly. Maximize $\sum_d t_d + \varepsilon x_c$ (and $-\varepsilon x_c$) over the same polytope, halving ε from 10^{-4} until the t-part of the optimum equals the unrestricted optimum. At that point the solution lies on the optimal face, and any face point with a larger x_c would have a larger perturbed objective, so the returned x_c is the exact extreme. On every $n \leq 40$, $[\ell, u] = [0, 0]$ for the weakest card and $[1, 1]$ for the strongest, as Lemma A.6 requires. (iii) A two-block partition with the weakest card at 0 and the strongest at 1 is a pure strategy. A pure strategy can be optimal only if $x_c = 0$ implies $\ell_c = 0$ and $x_c = 1$ implies $u_c = 1$. Every pure strategy meeting these constraints is enumerated and its value evaluated in closed form, $V(x) = \frac{1}{n(n-1)} \sum_d \min\{0, D_d(x)\}$, so the enumeration is complete and exact. For $n \leq 22$ the count of pure optimal strategies is recomputed over all 2^n pure strategies and agrees. (iv) When no pure strategy attains $v^\star$, width is at least 3 and an interval three-block partition attaining $v^\star$ is exhibited. At every width-two size exactly one of the pure optima is a threshold, folding the weakest m cards and calling the rest, so a minimum-size interval witness exists at every $n \leq 40$; Table 5 lists the block sizes. The result is width 3 exactly at $n \in \{3, 5, 6, 8, 13, 15, 16, 18, 23, 25, 26, 28, 33, 35, 36, 38\}$, the residues $3, 5, 6, 8 \pmod{10}$, and width 2 at every other $n \leq 40$. The pure-optimum counts at the width-2 sizes are the Catalan numbers $1, 2, 5, 14, 42, 132, 429, 1430, 4862, 16796, 58786, 208012$, each holding for one to three consecutive width-2 sizes. The value has period-ten structure as well: $v^\star = -1/10$ exactly at $n \in \{10, 11, 20, 21, 30, 31, 40\}$, and $v^\star_{12} = -13/132 > v^\star_{11}$, so the value is not monotone in n.

The predictor tests of Section 5 close the shortcut: the only exact predictor of width among the invariants we tested is the isomorphism class itself, and on this family width was obtained by solving. The practical consequence is that width is a diagnostic that has to be solved for on a tractable abstraction of a game.

The intransitive sweeps carry a control. The cyclic construction, card i beating the next $(n-1)/2$ cards around a circle, is rotationally symmetric, so a card-independent strategy attains the value and card-dependent play is unnecessary. It is measured at three, five, and seven cards, and returns width 1 and value 0 at each. The ordered relation is measured on the same code path at the same deck sizes, returning widths 3, 3, 2 and values $-1/18$, $-1/12$, $-2/21$, which agree with Figure 2. Every intransitive sweep carries the ordered arm alongside it as a cross-check. The transitive relation is one of the 456 classes, the total order on seven cards. It is where this sweep meets the ordered-deck one of Section 5: the edge-set enumeration used here and the forced-check sweep behind Figure 2 agree on it, width 2 and value $-2/21$ at $n = 7$. The ordered game is the relation whose width the cap theorem predicts, and the prediction holds inside the enumeration.

The games behind the identity. Figure 4 is stated over ninety-two configurations, each measured on both sides. The nine ordered games of Table 1 and seven five-card relations obtained by reversing comparisons, two at each of widths 1, 2, 4 and one at width 3, make sixteen. The rest are all 56 six-card isomorphism classes, at which widths $1, 2, 3, 4, 5$ occur $5, 36, 9, 3, 3$ times, and the 20

Table 5: The ordered deck, $n = 3, \ldots, 40$: width, the number of pure optimal strategies (zero exactly at width three), the block sizes of an interval witness from the weakest card up (a two-block witness folds the first block and calls the second; a three-block witness folds, mixes, calls), and the equilibrium value.

n	width	pure	blocks	$v^\star$	n	width	pure	blocks	$v^\star$
3	3	0	1|1|1	$-\frac{1}{18}$	22	2	429	9|13	$-\frac{23}{231}$
4	2	1	2|2	$-\frac{1}{12}$	23	3	0	3|7|13	$-\frac{151}{1518}$
5	3	0	1|2|2	$-\frac{1}{12}$	24	2	429	10|14	$-\frac{55}{552}$
6	3	0	1|2|3	$-\frac{4}{45}$	25	3	0	3|8|14	$-\frac{179}{1800}$
7	2	2	3|4	$-\frac{2}{21}$	26	3	0	3|8|15	$-\frac{97}{975}$
8	3	0	1|3|4	$-\frac{2}{21}$	27	2	1430	11|16	$-\frac{35}{351}$
9	2	5	4|5	$-\frac{7}{72}$	28	3	0	3|9|16	$-\frac{113}{1134}$
10	2	5	4|6	$-\frac{1}{10}$	29	2	4862	12|17	$-\frac{81}{812}$
11	2	5	5|6	$-\frac{1}{10}$	30	2	4862	12|18	$-\frac{1}{10}$
12	2	14	5|7	$-\frac{13}{132}$	31	2	4862	13|18	$-\frac{1}{10}$
13	3	0	2|4|7	$-\frac{23}{234}$	32	2	16796	13|19	$-\frac{99}{992}$
14	2	14	6|8	$-\frac{9}{91}$	33	3	0	4|10|19	$-\frac{79}{792}$
15	3	0	2|5|8	$-\frac{31}{315}$	34	2	16796	14|20	$-\frac{56}{561}$
16	3	0	2|5|9	$-\frac{71}{720}$	35	3	0	4|11|20	$-\frac{178}{1785}$
17	2	42	7|10	$-\frac{27}{272}$	36	3	0	4|11|21	$-\frac{377}{3780}$
18	3	0	2|6|10	$-\frac{91}{918}$	37	2	58786	15|22	$-\frac{133}{1332}$
19	2	132	8|11	$-\frac{17}{171}$	38	3	0	4|12|22	$-\frac{421}{4218}$
20	2	132	8|12	$-\frac{1}{10}$	39	2	208012	16|23	$-\frac{74}{741}$
21	2	132	9|12	$-\frac{1}{10}$	40	2	208012	16|24	$-\frac{1}{10}$

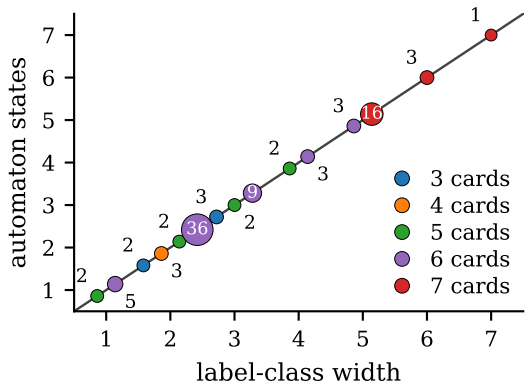


Figure 4: The identity on all ninety-two configurations, player 1 restricted: one marker per deck size at each width, its area proportional to the game count printed beside it; every marker lies on the diagonal, where $k^\star$ equals the width, up to width 7.

seven-card classes of width 5 or more, 16, 3, 1 at widths 5, 6, 7. The five-card sweep also re-measured the ordered five-card forced game, which is a row of Table 1 and is counted once. Two configurations duplicate others up to relabelling. The width-3 five-card relation reverses the adjacent pair 9, T and is the total order. The two width-4 relations (reversing 9, Q; reversing 9, T and T, Q) are one isomorphism class, the only one of width 4 on five cards. The ninety-two configurations are therefore ninety games up to isomorphism. The six-card scan includes the transitive class, which returns width 3 and value $-4/45$ on the tournament code path, matching Figure 2 at $n = 6$. With player 1 restricted, every class other than the widest has width one on all ninety-two configurations. On ninety of them, the eighty-eight raise-free games and the two forced-check one-raise games, the widest is the only facing class.

Figure 4 reaches width 4 at $v^\star = -3/23$ and widths 5, 6, 7 at values $-4/49$, $-284/2751$, $-2578/22015$. Above width four the check runs over every isomorphism class there is.

Table 6: Width of three fixed families of relations carried across deck sizes 5 to 8 (constructions in Appendix B); no row tracks the frontier $4, 5, 7$.

	$n = 5$	$n = 6$	$n = 7$	$n = 8$
champion recursion	4	5	3	2
single bottom flip	2	2	2	2
ceiling recursion	n.a.	n.a.	7	7

Table 7: Number of isomorphism classes of seven-card relations at each width, the counts behind Figure 3.

width	1	2	3	4	5	6	7
classes	172	170	62	32	16	3	1

The joint width, both seats. The joint width is measured directly, by cutting the deck and re-scoring every facing class at once. The measurement covers the eighty-eight raise-free configurations with each seat restricted in turn, and equals the recorded $k^\star$ on all 176 seat-games. On the four raising configurations $J = k^\star$ by Theorem 4.2 and $k^\star$ equals the largest width on both seats. Eight seat-games have more than one facing class: for player 1 the free one-raise game and the two-raise game (classes b and r), and for player 2 the two free Kuhn games (classes k and b) and the four raising games (classes k, r, and in free play b). On seven of the eight the joint width equals the largest width; the eighth is Proposition 4.4. The certificate for that game: player 2's equilibrium value in free four-card Kuhn poker is $1/24$; its facing-bet class has width 2 with witness $JT \mid QK$ and its facing-check class width 3 with witness $J \mid K \mid QT$; the six three-block deal partitions have values $TJ \mid Q \mid K$: 0, $T \mid JQ \mid K$: 0, $T \mid J \mid QK$: 0, $TQ \mid J \mid K$: $-1/12$, $TK \mid J \mid Q$: $-1/12$, $T \mid JK \mid Q$: $-1/10$, none $1/24$; the best two-block partition, $TJ \mid QK$, has value $-1/48$; and the best three-state machine has value 0, the four-state one $1/24$. For each three-block partition the file records a mixture over player 1's pure strategies, which makes the exclusion checkable in exact arithmetic alone, without a linear program: no admissible player-2 plan earns more than the listed value against it. The file also records a player-2 mixture on the four-block partition that earns $1/24$ against every player-1 pure strategy, together with those pure strategies. The seat-indexed record of the eighty-eight raise-free configurations, with J, a deal partition attaining it, and the best value at every smaller size for each seat, is the joint-width record of the code release.

The five-card maximum 4 is not the number replacing 3 once the order breaks: six cards reach 5 and seven reach 7, so deck size bounds width and intransitivity only releases it; forty ordered cards need two buckets while seven intransitive ones need seven.

The nested families. Maxima need not move together. Three fixed families, one construction per deck size, leave the frontier $4 \to 5 \to 7$ untouched; that frontier is a maximum over each deck's relations, and no single construction tracks it across deck sizes (Table 6).

The three rows of Table 6 are one construction each. Champion recursion starts from the five-card width-4 relation, the total order with the weakest card beating the second strongest, and at each step adds a new card that beats every card already present; its widths are $4, 5, 3, 2$ at values $-3/23$, $-1/15$, $-4/63$, $-1/14$. Single bottom flip is the total order with the weakest card beating the strongest, which stays at width 2 from five to eight cards at values $-2/15$, $-2/15$, $-1/7$, $-1/7$. Ceiling recursion starts from the unique seven-card width-7 class and adds a card beating every card present; the eight-card relation keeps width 7 at value $-5723/44030$. The transitive order was run on the same code path alongside them and returns $3, 3, 2, 3$, the widths of Figure 2 at $n = 5, \ldots, 8$.

The seven-card histogram. Table 7 reports the counts behind Figure 3.

The combinations. Table 2 reports three predictors singly; Table 8 reports four of their combinations, on the 1024 five-card relations and the 456 seven-card classes. The strong-component size multiset, folded out of Table 2, takes $4, 6, 8$ values on five, six, and seven cards and collides on $3, 4, 6$ classes; the largest component size and the component count each take $4, 5, 6$ values and collide on $3, 4, 5$.

Table 8: The four predictor combinations beyond the single predictors of Table 2, on the 1024 five-card relations and the 456 seven-card classes.

	5 cards		7 cards	
predictor	classes	collide	classes	collide
score seq. + strong comp. sizes	9	1	59	27
score seq. + largest comp. size	9	1	59	27
3-cycles + strong comp. sizes	6	3	23	19
score seq. + 3-cycles + strong comp. sizes	9	1	59	27

Table 9: The extremal relations. "distance" is the fewest reversals to a total order; the last column lists, as width:count, the widths of the relations obtained by reversing one comparison (10 on five cards, 21 on seven). Edge sets are in the code release.

class	**score sequence**	**width**	**distance**	**widths after one reversal**
five cards, width 4	1 1 2 2 4	4	1	1:2, 2:2, 3:3, 4:3
seven cards, width 6	1 2 2 2 3 5 6	6	2	2:8, 3:2, 4:4, 5:6, 6:1
seven cards, width 6	2 2 2 3 3 3 6	6	3	1:5, 2:8, 3:1, 4:2, 6:4, 7:1
seven cards, width 6	2 2 2 3 3 3 6	6	4	1:5, 2:3, 3:1, 4:6, 5:1, 6:1, 7:4
seven cards, width 7	2 2 2 3 3 3 6	7	4	1:7, 2:5, 3:1, 4:1, 5:2, 6:5

Every combination containing the score sequence takes exactly the score sequence's values and collides exactly where it does, as Proposition A.4 requires; the table is kept as the computed confirmation of that proposition.

The extremal relations. Table 9 describes the five-card class of width 4 and the four seven-card classes of width 6 or 7: the score sequence, the reversal distance, and the widths obtained by reversing each single comparison of the relation. The reversal distance is the fewest comparisons whose reversal turns the relation into a total order. Every one of the five has a trivial automorphism group. Width is not monotone in the reversal distance: on seven cards the largest width at distances $0, \dots, 7$ is $2, 4, 6, 6, 7, 3, 1, 1$, so the widest relations sit at distance two to four from the order, and the relations farthest from it, which are the most nearly regular, carry no information in the card and have width 1. The width-7 class is fragile: every single reversal of it lowers the width (Table 9).

A standalone certificate for width seven. The width-7 class is the tournament on $8, 9, T, J, Q, K, A$ in which each card beats the cards listed after it: 8 beats all six others; 9 beats T, J, Q; T beats J, Q, K; J beats Q, A; Q beats K, A; K beats $9, J$; A beats $9, T, K$. Its value is $v^\star = -2578/22015$. Of the 21 partition values that bound every coarser candidate, the largest is $-270/2303$ (merging J with K), which falls short of $v^\star$ by $988/7242935$, and the smallest is $-1/7$ (merging 8 with T, or T with A). The width-seven certificate records the unrestricted optimal strategy and, for each of the 21 partitions, the restricted optimal strategy and a rational dual solution of the direct linear program, so dual feasibility, the bound $b^\top u$ and the primal values are all checkable in fraction arithmetic alone, without a solver. The six three-block values of Proposition 4.4 are re-solved by the normal-form program.

The approximate-width ladder. The ladder record lists $L(k) = \max_{|P|=k} V(P)$ for $k = 1$ through n on the 456 seven-card classes and on the ordered decks of 3 to 16 cards. Decks of at most ten cards are searched over every partition, so their ladders are exact. From eleven cards the search runs over interval partitions, which bounds the unrestricted ladder from below, $L_{\mathrm{int}}(k) \le L(k)$, and the approximate width from above, $w_{\epsilon,\mathrm{int}} \ge w_\epsilon$: an interval partition attaining the exact width at $k = k^\star$ need not maximise value at every smaller k. Every entry is an exact rational re-solved by the direct linear program, so the file is a checked record, every ladder is nondecreasing, and every ladder arrives at $v^\star$ at $k = k^\star$. On the width-seven relation the shortfall is $81/3145$ at one bucket, $13/3885$ at three, $914/2355605$ at five, and $988/7242935$ at six. On the ordered decks the cap holds under tolerance as well: among the decks of at most ten cards, where every partition is enumerated, only the eight-card deck drops below its exact width by $\epsilon = 10^{-2}$, and it needs two.

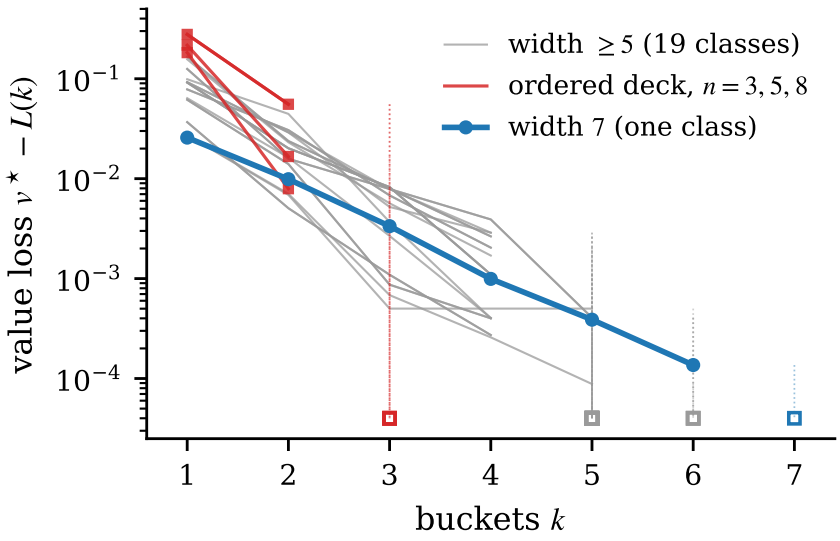


Figure 5: Value loss $v^\star - L(k)$ against the bucket count k, on a logarithmic scale: the twenty seven-card classes of width five or more (grey, the width-seven class highlighted) and the ordered decks $n = 3, 5, 8$ (red). Open squares mark exact attainment at the axis floor. At one bucket the ordered deck loses more than any of these intransitive classes; on the intransitive classes exactness arrives between five and seven buckets.

Table 10: The approximate width w_ϵ over the 456 seven-card classes and over the ordered decks of 3 to 16 cards. The exact width is the $\epsilon = 0$ column. The seven-card rows are exact; on the ordered-deck row the decks from eleven cards on are searched over interval partitions, which bounds the row from above, and the maximum of three is attained exactly at three cards.

ϵ	0	10^{-4}	10^{-3}	10^{-2}	5×10^{-2}
largest w_ϵ, seven cards	7	7	5	4	2
classes with $w_\epsilon \geq 5$	20	20	11	0	0
median w_ϵ, seven cards	2	2	2	2	1
largest w_ϵ, ordered decks	3	3	3	3	3

Checked records and certificates. Two kinds of file accompany the results and we keep the words apart. A *checked record* re-solves the optimization problems whose answers it lists: a width-attaining partition and its value certify that the partition attains $v^\star$, but the claim that every smaller partition falls short rests on the same solver that produced it. A *certificate* is checkable without a solver: it carries, for every excluded partition, a rational opponent strategy or dual solution against which no admissible plan reaches $v^\star$, and an arithmetic-only script verifies the bound. The width-seven relation and Proposition 4.4 have certificates; everything else is a checked record, and the ordered decks beyond eleven cards rest on the optimal-face procedure behind Table 5, whose optimal-face ranges and candidate lists are recorded for inspection.

The two sides of the identity are computed by separate routines. $k^\star$ is the least k at which the exhaustive enumeration of the state assignments realizable by a k-state transition function (Appendix D) attains $v^\star$; on every game with $k^\star > 1$ the best $(k^\star - 1)$-state machine falls strictly short. Width is the partition search of Section 3, in which the class under study is partitioned while every other class is left unpooled. The code release ships one machine-readable record per game, with player 1 as the restricted seat. It holds the deck, the showdown relation as an explicit edge set, the equilibrium value of that seat, the width and a width-attaining partition of every label class, the deal partition D^* constructed in Theorem 4.2, with its value, and the transition table of the resulting $k^\star$-state automaton. It also lists the exact value of the best k-state machine for every k up to $k^\star$. The script that writes a

record rebuilds the game from the listed configuration and re-scores every listed partition and the automaton by the exact linear program before writing, so each file is a checked record.

C Discussion

Read as buckets, the fewest that preserve the equilibrium value for one seat is the joint width. It equals the largest label-class width when the seat has one facing class, and can exceed it when the seat has several. In four-card Kuhn poker for player 2, each class fits three buckets, but the two classes want different cuts, and four states are needed (Proposition 4.4). For player 1 the measured minima equal the largest width on every configuration; on the ordered decks that is at most three per class, and intransitivity is what raises it. A bucket count is usually pushed until exploitability stops improving; on a game small enough to solve, width says where that happens, and the joint width says what a single bucketing of the cards costs when it must serve several decisions of one seat. The widths are seat-specific (Section 4), and a bucket scheme reused across seats therefore answers a third question, which we have not measured. Whether these findings carry to real poker is a hypothesis, since our showdown relations are arbitrary tournaments and a board shapes the real ones. If it does, the budget belongs on intransitivity's recognisable sources: blockers, competing draws, and board textures. The best-response threshold structure of Lemma A.1 motivates the contiguous partitions we observe on ordered decks without by itself bounding the maximin abstraction size. Whether that structure favours potential-aware and distribution-based abstractions (Gilpin et al., 2007; Ganzfried & Sandholm, 2014) over pure equity bucketing is an open question.

D Why Refinement Search Adds Nothing Here

The natural algorithmic reading of the identity of Section 4 is a refinement search: enumerate transition functions with more states, and split a state whenever a witness shows the pooling is costly. Three measured reasons say that search has nothing to add on this family.

First, the observation strings are short and nearly uniform. For player 1 in the deepest game we build, they take only the forms (card), (card, b), and (card, b, r). Position 0 therefore queries a card, and every later position queries a single facing symbol, b or r. Only the start state is ever asked about a card, and since the action at a string is chosen before the transition on its last symbol, the only public entry a state is ever asked for is its b entry. A k-state table thus has at most $n + k$ entries that matter rather than $k(n + 2)$, and the assignments to enumerate number at most k^{n+k} rather than $k^{k(n+2)}$. Branching remains, but at the state counts that occur here it stays affordable, as the next paragraph measures.

Second, distinct transition functions inducing the same partition of the infosets have the same value, so the count of linear programs to solve is the number of distinct partitions. The enumeration is a depth-first search over the observation strings of the restricted seat. Strings are processed in a fixed order, and each is followed from state 0 through the entries of a partial transition table. An entry not yet fixed is branched over all k targets, and when the string's last observation is reached its infoset is assigned the current state. Every completed assignment is realized by the partial table that produced it, and every realizable assignment is produced, because every entry a string queries is branched over all targets. Assignments are deduplicated as maps from infosets to states before the partition they induce, infosets grouped by state within each label class, is scored once. On player 1's free three-card Kuhn poker the search visits 8, 27, and 64 assignments at $k = 2, 3, 4$ and solves 4, 5, and 5 programs, the partitions of the facing class into at most k blocks; on four cards it visits 16 and 81 assignments and solves 8 and 14 programs at $k = 2, 3$. Exhaustive enumeration is simply cheap.

Third, depth adds nothing here. Reaching the third decision requires player 1 to raise, and an optimal strategy of the depth-three game raises with the strongest card alone. The deepest label class has one infoset per card, but only the strongest card's is reached by that strategy's own play, and an infoset its own play never reaches can take the action of any block without changing the payoff against any opponent strategy, so the class has width 1, as measured, and pooling it changes the value by zero. Building a deeper game does not build a harder instance.

The identity characterizes how much abstraction is required; it does not find it. On this family the characterization is the more useful of the two, because the enumeration it licenses is affordable.

Lemma 5.2 says isomorphism determines width, while degree information does not. What lies between, such as counts of small induced subtournaments, remains open.

E Solving the Abstract Game

This section carries the part of the related work that describes how the abstract game is solved once its size is fixed.

Once the abstraction is fixed, the abstract game is solved by iterative methods descended from regret matching (Hart & Mas-Colell, 2000) and fictitious play (Brown, 1951). Counterfactual regret minimization and its accelerated, discounted, and predictive variants (Zinkevich et al., 2007; Lanctot et al., 2009; Tammelin, 2014; Tammelin et al., 2015; Brown & Sandholm, 2019a; Farina et al., 2019; 2021; Li & Huang, 2026; Li et al., 2026) are scaled by function approximation (Brown et al., 2019; Heinrich et al., 2015; Heinrich & Silver, 2016) and decomposed by subgame techniques (Moravcik et al., 2016; Brown & Sandholm, 2017; Burch et al., 2014; Ganzfried & Sandholm, 2015; Zhang & Sandholm, 2021). Our numbers come from none of these: every value is an exact rational from a linear program (Section 3), not from sequence-form (Romanovskii, 1962) or iterative solving, so no tolerance enters any number.

F Equilibrium Under Failed Recall

This section carries the part of the related work that places the automaton among the solution concepts that weaken perfect recall.

What equilibrium means when recall fails has a long pedigree. It runs from the absent-minded driver (Aumann et al., 1997) and perfect recall as a structural condition (Okada, 1987), through refinements and awareness (Selten, 1975; Halpern & Rêgo, 2006), and extends to degraded recall and behavioral equilibria with complexity results (Cermák et al., 2018; Lambert et al., 2019; Tewolde et al., 2023; 2024; Berker et al., 2025). Our automaton is a proper subclass of general imperfect recall. Behavioral solution concepts, quantal response (McKelvey & Palfrey, 1995; 1998) and level-k (Nagel, 1995; Stahl & Wilson, 1995; Camerer et al., 2004), replace the benchmark, while width is defined against the equilibrium value itself.