Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Search-based agents outperform reinforcement learning in Dahal Jeet, a Nepaliimperfect-information card game

Raju Kumar Yadav
Pulchowk Campus
Ganesh Gautam
ganesh@pcampus.edu.np
Pulchowk Campus

Article

Keywords: Imperfect-information games, Monte Carlo tree search, Reinforcementlearning, Trick-taking card games, Benchmarking, Cultural heritage computing

Posted Date: November 2nd, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11016156/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Search-based agents outperform reinforcement learning in Dahal Jeet, a Nepali imperfect-information card game

Raju Kumar Yadav[1] and Ganesh Gautam[1*]

[1*]Department of Electronics and Computer Engineering, Institute of Engineering, Pulchowk Campus, Tribhuvan University, Lalitpur, Nepal.

*Corresponding author(s). E-mail(s): ganesh@pcampus.edu.np;
Contributing authors: 076bct100.raju@pcampus.edu.np;

Abstract

Dahal Jeet is a four-player partnership card game played daily across the Terai-Madhesh region of Nepal, in which only the four tens score. Its objective is lexicographic rather than additive: a majority of tens decides most hands, trick count the rest. The game belongs to the documented Mendikot branch of the Court Piece family, matching Mendikot on 21 of 26 compared features, but the Nepali variant has no published rule description, and the indexed literature returns one paper on the entire family. We contribute a formal specification of the ruleset, drawn from the first author's own play across seven districts, and a computational study built on it. 26 agents spanning hand-written heuristics, metaheuristic tuning, determinized search and seven learning families are compared across 325 pairings under duplicate-deal protocols with bootstrap intervals and family-wise error control. Search dominates: information-set Monte Carlo tree search leads and no learner reaches a tuned heuristic. An exact double-dummy solver shows search converts 0.923 of theoretically won endgames against 0.810 for the best heuristic. A minimal state encoding outperforms a rich hand-designed one, every deterministic heuristic is exploitable by a search that knows its policy, and the deal alone explains up to 0.391 of outcome variance.

Keywords: Imperfect-information games, Monte Carlo tree search, Reinforcement learning, Trick-taking card games, Benchmarking, Cultural heritage computing

1 Introduction

A game played every afternoon in Sarlahi has never been written down. Dahal Jeet is a trick-taking game of the Court Piece or Rang family, played by four players in two fixed partnerships, in which only the four tens score. The family is documented and we do not claim otherwise: this is a Nepali variant of Mendikot. What is absent is the variant. It is played daily in village gatherings across the Nepal Terai-Madhesh, yet the standard reference's own survey of card games played in Nepal lists no game of this branch [1], and the peer-reviewed literature contains a single paper on the whole family [2].

Systematic rule-matching against retrieved primary sources places Dahal Jeet inside the Court Piece family [3], in its tens-counting branch, closest to Mendikot [4] at 21 of 26 compared features (Supplementary Table S1), with a deferred-capture variant reproducing Dehla Pakad [5]. Vocabulary carries the correspondence further than the mechanics do: players in the Terai say *coat*, which is the South Asian *kot*, glossed on the Mendikot page as a win in which the opponents score nothing. What is unattested is narrow, and we state it narrowly: a double coat cancelling the coat's dealer-skip appears in no source we located. We searched OpenAlex, arXiv, Shodhganga and Nepal Journals Online across eight queries in English and Devanagari. None returns a record of this game. That is not the same as proving no record exists; much Nepali scholarship appears in print-only venues, and a game may be described under a local name none of our queries guessed.

The game is local; two of its properties are not. First, its objective is lexicographic and non-additive: a four-valued primary criterion with a thirteen-valued secondary that activates on one configuration of the primary and is otherwise ignored entirely, where every trick-taking game commonly used as a benchmark scores additively. Second, our principal negative result comes with a mechanism, the scoring density $\rho_{\text{score}} = 4/52$, and that mechanism yields a prediction the game family itself can falsify, since Court Piece scores every trick on a near-identical skeleton.

Determinized search is the methodological backbone of this study. Ginsberg's GIB established perfect-information Monte Carlo for bridge [6]; Cowling and colleagues introduced information-set Monte Carlo tree search, including the multiple-observer form we implement [7], and later measured ensemble determinization in Magic: The Gathering [8]. Solinas and colleagues combine search with supervised learning in trick-taking games [9]. On the partnership side, Amit and Markovitch model partner as well as opponent in bridge bidding [10], and Cohensius and colleagues study bidding in Spades [11]. The learning families we compare are standard: deep Q-networks [12] with the double [13] and duelling [14] refinements, PPO [15], A2C [16], MAPPO under centralised training with decentralised execution [17], policy distillation [18], and the self-play and regret-minimisation line running from CFR [19] through NFSP [20] to Deep CFR [21]. Reward shaping follows the potential-based formulation [22], and RLCard [23] is the closest existing card-game toolkit, though it does not include this family.

The closest work on a traditional Nepali card game is Malla's study of Dhumbal [24], which compares heuristic, search and learning agents and reports that a hand-written heuristic outperforms both. We note the parallel because our headline result

has the same shape on a different game, reached independently; the two studies share no code, no data and no game. Beyond cards, Nepali board games have received similar treatment in work on Baghchal [25, 26], and the question of how much of a traditional game is skill rather than chance has been quantified for South Asian games including Teen Patti [27], though not for this family. The framing of such work as cultural-heritage preservation has its own literature [28].

This paper contributes

(a) a formal specification of Dahal Jeet with numbered, individually testable rules, a documented provenance across seven districts, and a mathematical model of the ruleset;
(b) a reference engine with a behavioural test that no agent can access information its seat has not observed;
(c) a comparison of 26 agents under one protocol, with family-wise error control and a retrospective power analysis, in which every reported estimator is defined and 30 of 30 of the resulting identities are verified against the stored artifacts;
(d) an exact double-dummy endgame solver giving an optimality bound; and
(e) three negative or corrective results reported in full: reward shaping does not separate under PPO; a minimal encoding beats a rich one; and every deterministic heuristic tested is exploitable by a search that knows its policy, while random play is not.

2 Results

Every table in this section is generated directly from its result artifact; no value is transcribed by hand.

2.1 The structure of the game, and two objectives that conflict

There are 5.364×10^{28} distinct deals, exactly, and a seat's initial information set contains 8.448×10^{16} possibilities, also exactly (Methods, Equations (2) and (10)). The mean number of legal moves is 3.880, measured over 156,000 decisions, and it declines trick by trick (Supplementary Table S2). Evaluating Equation (10) at successive tricks, with inferred voids and the shown trump card applied, gives $10^{16.93}$ possible worlds at trick 1, $10^{12.23}$ by trick 4, and $10^{1.31}$ at trick 12 (Supplementary Table S3): fifteen orders of magnitude inside one hand. A fixed sampling budget therefore buys incomparable accuracy at trick 2 and trick 11, adequate coverage late and negligible coverage early.

By Equation (6) a hand is decided on one of two channels, the tens majority or the trick tiebreak at two each, and the channels partition the hands. Measured over 10,000 hands, pure ten-hunting won 0.6460 of tens-decided hands but only 0.1922 of tiebreaks; pure trick-taking did the reverse; and the two sit at 0.5024 head to head rather than one dominating (Fig. 1). The mixture identity of Equation (26) is one of the identities checked against the artifact. An agent facing this objective must decide, at every trick, which of two partially incompatible objectives it is currently playing for, and the answer depends on a state variable, the tens split, that it does not control.

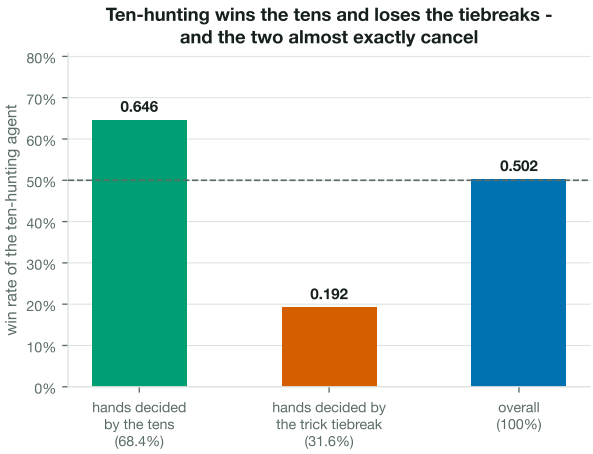


Fig. 1 The two objectives pull against each other. Pursuing tens wins most tens-decided hands but loses most two-each tiebreaks, and the two effects very nearly cancel.

2.2 Determinized search leads a full round-robin of 26 agents

All 26 agents played a complete round-robin of 325 pairings at 300 duplicate deals each, 390,000 hands. Determinized search takes the top of the table outright (Table 1, Fig. 2). ISMCTS250 leads at Elo 1663.2 and PIMC16 follows, and the gap from there down to third place is wider than the gap between any other adjacent pair near the top. Third place goes to the metaheuristically tuned heuristic, Tuned at 1563.4, ahead of every network we trained. MAPPO is the strongest of the learners and the only one that reaches the band the better hand-written heuristics occupy. NFSP sits second from bottom, above only random play: a method built for imperfect-information games finished below heuristics that fit in a page of code. At the budgets used here, no learning method reaches the tuned heuristic, and determinized search is stronger than both. That gap is the study's principal negative result.

Of the 325 pairings, 263 are nominally significant against the 16.2 expected by chance alone at $\alpha = 0.05$; 208 survive Holm–Bonferroni and 260 survive Benjamini–Hochberg (Supplementary Table S4). We report corrected values throughout. Substituting the observed per-deal standard deviation, 0.2778, into Equation (16) puts the detectable difference at 300 deals at roughly 0.05 for 80% power; resolving 0.02 would take 1514 deals, five times the tournament. Differences close to 0.02 are therefore beyond what this design can resolve, and we call them inconclusive rather than ties.

The ranking is not perfectly transitive. The cyclic energy of Equation (18) comes to $\kappa = 0.0518$, against a noise floor of 0.0154 simulated under a perfectly transitive Bradley–Terry truth at the measured intra-deal correlation $\rho = 0.097$, giving $p = 0.005$; 66 of the 2,600 triples are three-cycles. Which null is chosen decides that verdict, and Remark 2 gives the reason. The agents form a mostly clean ladder with a modest cyclic component on top.

Rank	Agent	Elo	95% CI
1	ISMCTS250	1663.2	[1654.2, 1673.2]
2	PIMC16	1643.2	[1634.4, 1652.4]
3	Tuned	1563.4	[1555.7, 1571.7]
4	AlphaDJ60	1553.2	[1547.8, 1558.5]
5	mappo_potential	1549.2	[1542.4, 1555.1]
6	TensThenTricks	1547.4	[1540.2, 1556.7]
7	PartnerAware	1539.1	[1532.0, 1548.2]
8	distill	1537.0	[1530.3, 1543.3]
9	GreedyTricks	1530.2	[1523.2, 1537.7]
10	ppo_trick_shaped	1519.5	[1512.1, 1526.5]
11	AdaptiveGreedy	1516.0	[1507.9, 1524.0]
12	TenAware	1514.6	[1506.2, 1522.7]
13	Adaptive	1511.7	[1503.3, 1519.2]
14	ppo_potential	1504.5	[1497.5, 1511.3]
15	ppo_terminal	1500.9	[1492.2, 1508.0]
16	ppo_ten_shaped	1490.4	[1483.5, 1497.3]
17	deepcfr	1463.7	[1456.4, 1471.8]
18	double_dueling	1458.9	[1451.8, 1465.5]
19	dqn_potential	1441.2	[1433.5, 1447.7]
20	dqn_terminal	1440.8	[1434.0, 1447.6]
21	dqn_ten_shaped	1439.6	[1431.7, 1446.2]
22	double	1438.0	[1430.2, 1444.2]
23	a2c	1437.4	[1428.8, 1446.0]
24	dqn_trick_shaped	1436.7	[1428.6, 1444.9]
25	nfsp	1387.0	[1378.1, 1395.2]
26	Random	1373.0	[1366.4, 1381.6]

Table 1 Full round-robin ranking, all 26 agents, 300 duplicate deals per pairing. Bradley–Terry ratings on an Elo scale; intervals bootstrapped over deals. Ratings are relative to this pool only.

2.3 Strength is bought with compute, and with an opponent model

Both search families gain strength monotonically as the budget rises, and at comparable budgets across a full hand information-set search is the stronger of the two (Fig. 3, Supplementary Table S5). Sampling worlds uniformly ignores the play history, and most such worlds contradict it: waste rises from 1.4% at trick 1 to 64% by trick 11 under random play (Supplementary Table S6). Whether constraining the sampler pays is a separate question and the answer is equivocal. At matched samples the constrained sampler wins at every budget (Supplementary Table S7); at matched wall-clock, which is the operative comparison since constraining costs time, the advantage narrows to 0.02–0.03, inside the inconclusive band above, and only one of eight arms survives Holm–Bonferroni (Supplementary Table S8). We report one significant arm of eight, not a general result.

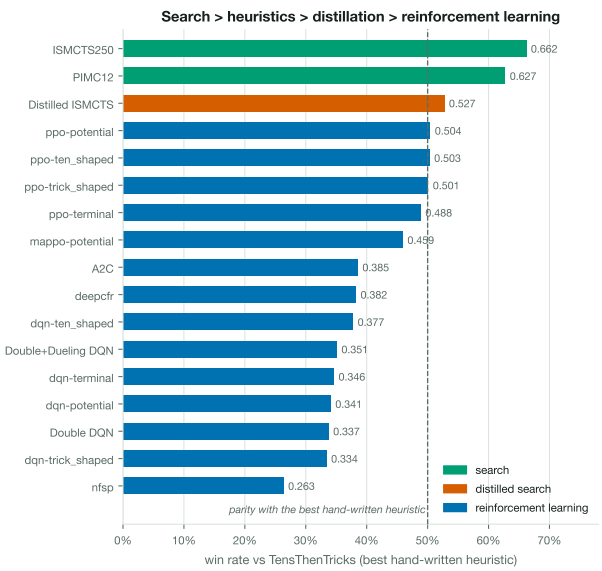


Fig. 2 Final ranking across all 325 pairings. Search occupies the top, the metaheuristically tuned heuristic is third, and no learning method exceeds it.

Correcting the single-observer backup of Equation (29) is worth between 0.0124 and 0.0500 win rate depending on search budget, and wins head to head at the largest budget tested (Supplementary Tables S9 and S10).

Knowing the opponent is worth more. Of 5 targets, 4 are significantly exploitable by an otherwise identical search that simulates the target's true policy in the opponent seats (Fig. 4, Supplementary Table S11). Random play is not (+0.0033, $p = 0.846$), which is the sanity result: there is no pattern in a random policy to exploit, so Equation (20) should return zero, and it does. Every deterministic heuristic is exploitable, including the strongest one (+0.0517).

2.4 Search plays endgames close to optimally

Exact double-dummy solving is affordable only in the endgame: cost grows roughly tenfold per additional trick, and whole-hand solving is not computable within any budget we could justify (Supplementary Table S12). We therefore report an endgame bound, defined by Equations (24) and (25).

The protocol matters. When each agent generates its own positions, conversion rates are not comparable, because a stronger agent reaches different endgames; its theoretical win rate ranges from 0.383 to 0.530 (Supplementary Table S13). On a common position set the theoretical rate is 0.5000 for every agent by construction and the comparison is clean (Table 2): PIMC converts 0.923 of theoretically won endgames and information-set search 0.897, against 0.810 for the best heuristic and 0.630 for random play. Search therefore plays endgames close to optimally; heuristics leave roughly one in five won positions on the table. This is the study's only absolute rather than relative yardstick.

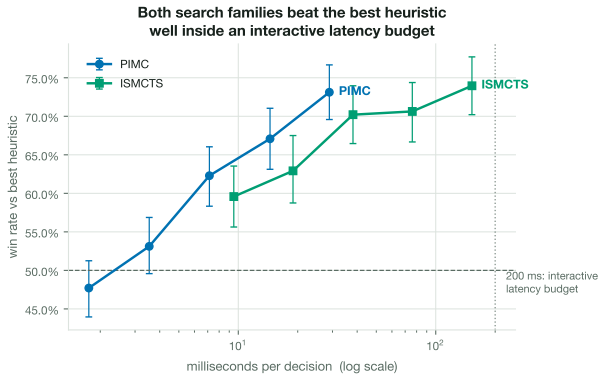


Fig. 3 Strength against measured decision cost for both search families. No cross-game reference point is plotted: latency depends on the action space and the rules searched, so a figure from another game is not commensurable.

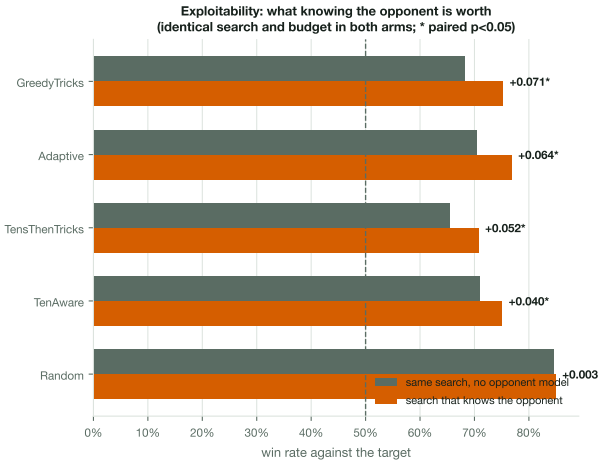


Fig. 4 Exploitability. Both arms are the same search at the same budget; only the opponent model differs.

Agent	Theoretical	Actual	Conversion	Steal	Gap
Random	0.5000	0.3683	0.630	0.107	+0.1317
GreedyTricks	0.5000	0.4350	0.713	0.157	+0.0650
TenAware	0.5000	0.4817	0.773	0.190	+0.0183
Adaptive	0.5000	0.4833	0.780	0.187	+0.0167
TensThenTricks	0.5000	0.5000	0.810	0.190	+0.0000
PIMC:16	0.5000	0.6200	0.923	0.317	-0.1200
ISMCTS:200	0.5000	0.5900	0.897	0.283	-0.0900

Table 2 Endgame optimality on a common position set: one fixed policy drives every seat to the handover, so all agents face identical positions. The theoretical rate is identical by construction, which is the protocol's own check. These are the comparable numbers.

2.5 No learning method reaches a tuned heuristic

The strongest learner is distill at 0.5270 ± 0.0096 (Table 3). Untrained networks of the same architectures score 0.260–0.267, and this is the floor against which trained results should be read: a randomly initialised network is not random play, because the architecture and action masking already impose structure. The learning curves flatten well before the training budget is exhausted (Supplementary Fig. S1), so the shortfall against search is not simply insufficient training. Checkpoints for the distilled policy are selected on win rate rather than on validation loss, because its accuracy stays flat while playing strength peaks and then decays (Supplementary Fig. S4). The spread across random seeds is what makes three runs the reporting unit throughout (Supplementary Fig. S5). Varying learning rate, hidden width and discount around the operating point moves the win rate against the best heuristic by 0.0637 from best to worst (Supplementary Table S14), which is smaller than the gap between the learners and the tuned heuristic; the headline shortfall is therefore not an artefact of a poorly chosen setting.

Method	Seeds	vs Random	vs best heuristic
distill	3	0.7358 ± 0.0068	0.527 ± 0.0096
ppo-potential	3	0.6921 ± 0.0038	0.5039 ± 0.0086
ppo-ten_shaped	3	0.6853 ± 0.0039	0.5028 ± 0.004
ppo-trick_shaped	3	0.6946 ± 0.0083	0.5006 ± 0.0064
ppo-terminal	3	0.6844 ± 0.0065	0.4883 ± 0.01
mappo-potential	3	0.7187 ± 0.0026	0.4589 ± 0.0186
a2c	3	0.6218 ± 0.0245	0.385 ± 0.0291
deepcfr	3	0.6314 ± 0.0085	0.382 ± 0.013
dqn-ten_shaped	3	0.6103 ± 0.0128	0.3772 ± 0.0193
double_dueling	3	0.6132 ± 0.005	0.3506 ± 0.0109
dqn-terminal	3	0.599 ± 0.0211	0.3458 ± 0.0284
dqn-potential	3	0.614 ± 0.0061	0.3405 ± 0.0094
double	3	0.5936 ± 0.0126	0.337 ± 0.0137
dqn-trick_shaped	3	0.5906 ± 0.0137	0.3336 ± 0.0057
nfsp	3	0.5333 ± 0.0022	0.263 ± 0.0034

Table 3 All learning methods, mean ± standard deviation across seeds.

Three further results bear on why. First, reward design separates under DQN, where ten-shaped rewards win consistently across seeds, but under PPO nothing separates: every pairwise difference is small and only two of six are consistent in direction across seeds (Supplementary Table S15, Supplementary Fig. S2). The deliberately misaligned control, which rewards trick-taking, does not underperform under PPO. A plausible reason, rather than a resolution: a two-each split of the tens is decided on trick count, so trick-taking is less misaligned with the true objective than the label

suggests. Matching on episodes gives DQN and PPO very different optimisation budgets, so we matched on wall-clock and again on gradient steps; PPO wins under both (Supplementary Table S16).

Second, the state encoding inverts. No single block's removal changes performance materially, yet removing almost all of it, leaving only the agent's own hand and the current trick, improves performance by +0.0644, and the seeds of the two conditions do not overlap (Supplementary Table S17, Supplementary Fig. S3). The observation was designed to hand the objective to the network explicitly, so the direction of this result runs against the encoding's own rationale. It is also non-monotonic: removing the play history alone is the worst option, while removing it along with everything else is the best, which rules out both "more features help" and "fewer features help". We report it unresolved. Permutation importance on an already-trained policy is consistent with it: the current trick and the play history matter most, while the explicit ten-status block, the one designed to hand over the objective, is barely used (Supplementary Table S18).

Third, every learner is trained against one fixed opponent, which leaves open whether it learned the game or that opponent. Read through Equation (21), which cancels opponent strength by differencing two win rates against the same opponent, the advantage of having trained on a specific opponent is $+0.0323 \pm 0.0174$ and the agents largely learned the game (Supplementary Table S19). The naive alternative of Equation (22) reports +0.185 on the same data, but that compares performance against different opponents and mostly measures that search agents are stronger than heuristics.

2.6 Much of the game is luck, and trump length outweighs tens dealt

Because every deal is played from all four seat assignments, Equation (17) applies and the question can be answered rather than argued. For TensThenTricks against Random the cards alone account for 0.391 of outcome variance and seat assignment for 0.000; across the pairings decomposed in Table 4 the deal's share runs from 0.290 to 0.391.

Agent A	Agent B	Deal	Seat	Residual	Mean
TensThenTricks	Random	0.391	0.000	0.609	0.7417
TensThenTricks	TenAware	0.290	0.002	0.708	0.5208

Table 4 Variance decomposition of the outcome. 'Deal' is the cards themselves, i.e. luck; 'seat' is structural seat advantage, which duplicate dealing removes; 'residual' is how the matchup played out.

Conditioning on properties of the deal fixed before play, over 1,200 deals against TensThenTricks, exposes an asymmetry. In a game where only tens score, trump length predicts the outcome far more strongly than the number of tens dealt. A team holding

eight or more trumps wins 0.785 of hands (1,276 hands) against 0.627 for a team dealt all four tens (284 hands), and the win rate moves 0.592 across the trump-length buckets against 0.254 across the tens-dealt buckets (Supplementary Table S20). Long trumps confer control over which side captures the tens, and that control matters more than starting possession of them.

3 Discussion

Determinized search handles this game's structure; every learning method we trained does not. The likely reason is the scoring density of Equation (9): with $\rho_{\text{score}} = 4/52$, a whole hand of thirteen tricks yields at most four events that bear on the objective. Search can plan directly toward those events, because the objective is cheap to evaluate at a leaf. A learning agent has to discover them through a signal that is almost everywhere zero. That reading is a hypothesis about a mechanism, not a measurement, but it is a falsifiable one: Court Piece scores every trick on a near-identical skeleton, so raising ρ_{score} from $4/52$ to 1 while holding the rest of the game fixed should narrow or reverse the gap between search and learning. If it does not, the density reading is wrong. We have measured one point on that axis and offer the prediction rather than the result.

The game is regional; its objective is not. A lexicographic, non-additive objective sets this family against the additive scoring of every benchmark trick-taking game. Where those permit a policy to trade a loss on one dimension for a larger gain on another and optimise a single scalar, this objective does not: no number of tricks compensates for a third ten, and no number of tens is worth anything once the split is two each. Measured, the constraint bites, with the two channels pulling in opposite directions hard enough to cancel. Dahal Jeet is a small and cheap instance of that structural problem, with thirteen tricks, a mean branching factor of 3.880, and an exact endgame solver, and we offer it as a testbed on those grounds. Void-constrained determinization transfers more narrowly: it applies to any follow-suit game, and our result there is useful precisely because it is negative, since an engine author working on bridge, skat or hearts can take it as a reason to measure before paying for the constraint. Separately, the pipeline this paper follows, from insider participation through numbered individually testable rules to an executable reference engine and then a benchmark, transfers to any traditional game lacking a written specification.

Several limitations bound these claims. No human evaluation was performed. Every result here establishes strength relative to other agents in this benchmark, and none of it establishes strength against human players; a determinized search that dominates a field of programmed opponents may still be legible to an experienced player, and the exploitability result is a reason to expect exactly that, since every deterministic policy tested was exploitable by a search that knew it. Measuring against people is the natural next step and is not attempted here. The learning results are a statement about these algorithms at these budgets, not about learning in general, and they are reported as a baseline rather than a solution; exploitability is a lower bound only, for the reason set out in Remark 3. The encoding inversion is unexplained. The transfer claims above are hypotheses supported by this game's structure rather than measurements across

games, and the scoring-density prediction is untested at a single measured point. The specification covers one ruleset, one region and one language community, and its provenance is retrospective, without contemporaneous field notes.

4 Methods

4.1 The game

This paper documents the single-hand, immediate-capture variant, which is what the observed player communities play; a deferred-capture variant exists in the wider family and is out of scope.

Seats 1 and 3 play against seats 2 and 4. Each player receives thirteen cards, dealt in batches of five, four and four. Nobody chooses trump. Once the first five are out, the seat after the dealer draws one of its own five at random without looking, and that card's suit is trump for the rest of the hand. All four players see the card. It then returns to the hand it came from, where it stays playable like any other. Following suit is compulsory for a player who can; a player who cannot may play anything and is never obliged to trump. The trick falls to the highest trump in it, or, where nobody trumped, to the highest card of the suit led, and its winner takes the cards there and then.

Forty-eight of the fifty-two cards are worth nothing. Three tens or more takes the hand. At two each the tens stop deciding anything and trick count settles it, which it always can, since thirteen tricks cannot split evenly. All four tens is a *coat*; all four tens together with all thirteen tricks is a *double coat*. The losing side deals next. A coat pushes that deal past the seat it would otherwise fall to and on to the loser's partner; a double coat cancels the push. This last rule is the one we could not find attested in any documented source.

4.2 Formal model

This section states the rules of §4.1 as mathematical objects, so that the quantities reported above have definitions rather than descriptions. Let $N = \{1, 2, 3, 4\}$ be the seats in turn order, $T_0 = \{1, 3\}$ and $T_1 = \{2, 4\}$ the two fixed partnerships, $\mathcal{C}$ the 52-card deck with $\mathrm{suit}(c)$ and $\mathrm{rank}(c)$, and $\mathcal{T} \subset \mathcal{C}$ the four tens. Write H_i for the cards seat i still holds, t for the trump suit, ℓ for the suit led, τ for the current trick as a set of seat and card pairs, m_k and k_k for the tens and tricks captured by team k, and V_i for the suits seat i is provably void in. Every equation here is implemented in the released engine.

Dahal Jeet is a finite extensive-form game with imperfect information, chance moves and two teams,

$$G = \big(N, \{T_0, T_1\}, \mathcal{C}, \mathcal{H}, \mathcal{Z}, \varrho, \{\mathcal{I}_i\}_{i \in N}, u\big), \tag{1}$$

where $\mathcal{H}$ is the set of histories, $\mathcal{Z} \subset \mathcal{H}$ the terminal ones, ϱ the chance measure, $\mathcal{I}_i$ the information sets of seat i and u the payoff of Equation (6). Partnerships are fixed and known, so the teams are part of the game rather than a coalition structure to be discovered.

There are two chance nodes, not one. The deal is an ordered partition of the deck into four labelled hands of thirteen, and the trump suit is a second chance move: the seat after the dealer d draws uniformly from its own first five cards, written $H_{d+1}^{(1:5)}$, without looking, so

$$|\mathcal{D}| = \frac{52!}{(13!)^4} = 5.364 \times 10^{28}, \qquad t \sim \mathrm{Unif}\big\{\mathrm{suit}(c) : c \in H_{d+1}^{(1:5)}\big\}. \tag{2}$$

The consequence separates this family from most studied trick-taking games: trump is drawn, never chosen or bid, so no part of the bidding literature applies and the trump suit carries no information about anyone's hand beyond the one card shown.

Following suit is compulsory when possible and discretionary otherwise, with no obligation to trump when void:

$$L(H_i, \ell) = \begin{cases} \{c \in H_i : \mathrm{suit}(c) = \ell\} & \text{if that set is non-empty,} \\ H_i & \text{otherwise.} \end{cases} \tag{3}$$

The second branch generates the paper's central piece of hidden information. A seat that takes it is provably void thereafter,

$$V_i \;\leftarrow\; V_i \cup \{\ell\} \quad \text{whenever seat } i \text{ plays } c \text{ with } \mathrm{suit}(c) \neq \ell, \tag{4}$$

and V_i is public: every seat can draw the same inference.

A trick is won by the highest trump, or failing that the highest card of the led suit. Both cases collapse into one lexicographic argmax by scoring trumps above the whole rank order:

$$\varphi(c) = \begin{cases} 52 \cdot \mathbf{1}[\mathrm{suit}(c) = t] + \mathrm{rank}(c) & \text{if } \mathrm{suit}(c) \in \{t, \ell\}, \\ -\infty & \text{otherwise,} \end{cases} \qquad w(\tau) = \operatorname*{argmax}_{(i,c)\in\tau} \varphi(c). \tag{5}$$

Cards neither of the led suit nor of trump can never win, which is why φ sends them to $-\infty$ rather than to a low finite value.

Only the four tens score, and a two-each split is decided on trick count. With m_0 tens and k_0 tricks to T_0,

$$u(z) = \mathbf{1}[m_0 \geq 3] \;+\; \mathbf{1}[m_0 = 2] \cdot \mathbf{1}[k_0 \geq 7]. \tag{6}$$

Remark 1. Equation (6) is lexicographic and non-additive: a four-valued primary criterion, with a thirteen-valued secondary that activates on exactly the two-each split and is otherwise ignored entirely. The trick-taking games that serve as standard benchmarks all score additively, so an agent can trade a small loss on one dimension against a larger gain on another. Here it cannot.

Since there are thirteen tricks, $k_0 \neq k_1$ always, so Equation (6) admits no draw. For the winning team w the two named results are

$$\mathrm{coat}(z) = \mathbf{1}[m_w = 4], \qquad \mathrm{double}(z) = \mathbf{1}[m_w = 4] \cdot \mathbf{1}[k_w = 13]. \tag{7}$$

A double coat is therefore a coat, which is why the cancellation rule is unambiguous. Writing $\pi(i)$ for i's partner and δ_2 for the first seat of the losing team in turn order from d, the deal passes as

$$\delta\big(d, \bar{w}, \mathrm{coat}, \mathrm{double}\big) = \begin{cases} \pi(\delta_2) & \text{if coat} = 1 \text{ and double} = 0, \\ \delta_2 & \text{otherwise.} \end{cases} \tag{8}$$

Play continues indefinitely and what accumulates is a social tally, not a target score. Formally the session is a Markov chain on the dealer and tally whose transition is Equation (8), and it has no absorbing state. Every episode boundary in this paper, one hand, is therefore a modelling artifact imposed by the experiment rather than a rule of the game.

The property that drives the paper's principal result is one ratio. Writing ρ_{score} for the fraction of the deck that scores and $E_{\max}$ for the largest number of credit-bearing events a hand can contain,

$$\rho_{\mathrm{score}} = \frac{|\mathcal{T}|}{|\mathcal{C}|} = \frac{4}{52} \approx 0.0769, \qquad E_{\max} = |\mathcal{T}| = 4 \text{ over } 52 \text{ plies.} \tag{9}$$

4.3 Provenance

The specification is a formalisation of the first author's own knowledge of the game, acquired by playing it over many years as a member of the communities that play it; this insider or emic stance is standard in ethnography and folklore studies. The settings were a college peer group of approximately twenty players drawn from six districts, and village gatherings of daily players spanning two generations, with four to five separate groups in a single village. The same ruleset held in all of them, across seven districts, and that consistency is what licenses describing this as a regional variant rather than one village's house rules.

This study involved no human research participants. No person was recruited, enrolled, interviewed, or asked to perform any task for this work; no protocol or instrument was administered to anyone; and no data about any person were collected, held or analysed. What is formalised here is the first author's own recollection of games he has played, written down as numbered rules and then tested against the engine. No individual is identified anywhere in this paper, and the game is attributed at district level only.

Insider status provides access and fluency, and is also a bias: familiarity can render local variation invisible. The record is retrospective, and there are no contemporaneous field notes.

4.4 Engine and evaluation protocol

Each numbered rule has a corresponding test. A separate behavioural test verifies the absence of information leakage: redistributing the hidden cards among opponents, consistently with everything the acting seat has observed, must not change the card the agent plays.

Evaluating on independently shuffled hands makes deal luck a large variance component. Every deal is therefore played four times with the seat assignment rotated, so both agents hold identical cards from every seat and the comparison is paired. That is also what makes the deal, not the hand, the independent unit throughout. Contrary to a common assumption, the trump-drawing seat is not structurally advantaged: across 4,800 self-play hands the side that drew the trump won 0.5008 of the 2,400 hands in which it did (95% CI [0.4808, 0.5208]), and the seat term in a variance decomposition is 0.000. Rotation is worthwhile because it makes the comparison paired, not because it removes an advantage that exists.

A refactor and a performance optimisation were applied after results already existed, and neither altered any result; a checkpoint trained by the original code is reproduced bit-for-bit by the current code (Supplementary Table S21).

The exact size of a seat's information set follows from Equation (4) by a multinomial count over the eligibility pattern of the unseen cards,

$$|\mathcal{I}_i| = \sum_{\text{patterns}} \binom{n_u}{n_1,\, n_2,\, n_3}, \tag{10}$$

evaluated by dynamic programming over which unseen cards each opponent may legally hold given the observed voids. Waste from unconstrained sampling is estimated by drawing M worlds and testing them against the observed voids, giving at trick θ

$$\hat{W}_\theta = 1 - \frac{1}{M|\mathcal{S}_\theta|} \sum_{h \in \mathcal{S}_\theta} \sum_{m=1}^{M} \mathbf{1}\big[\text{world } m \text{ consistent with } V(h)\big], \tag{11}$$

over the decisions $\mathcal{S}_\theta$ observed at that trick.

4.5 Estimands and estimators

Every headline quantity above is an estimate of something, and this section says what. The estimators are stated in the form the code implements. Wherever an identity asserted here is checkable by arithmetic over the stored artifacts, `verify_equations.py` recomputes it, re-running no experiment and reading only files already on disk: 30 of 30 checks pass, and the run is itself an artifact.

Writing $Y_{jr} \in \{0, 1\}$ for whether team A won deal j under rotation r, and $D_j = \sum_{r=0}^{3} Y_{jr}$ for the per-deal win count,

$$\hat{w}_A = \frac{1}{4n} \sum_{j=1}^{n} D_j. \tag{12}$$

All 325 pairwise comparisons use a two-sided sign-flip permutation test on the per-deal margin, which is exact for this design rather than assuming normality. With

$$T = \frac{1}{n}\sum_{j=1}^{n}\big(D_j - 2\big), \qquad T^*(\varepsilon) = \frac{1}{n}\sum_{j=1}^{n}\varepsilon_j\big(D_j - 2\big), \quad \varepsilon_j \overset{\text{iid}}{\sim} \text{Unif}\{-1,+1\}, \tag{13}$$

the p-value over $B = 20{,}000$ draws is

$$\hat{p} = \frac{1 + \#\{b : |T_b^*| \geq |T|\}}{B+1}. \tag{14}$$

The validity argument is earned by the duplicate design: under the null that the two agents are equally strong, relabelling them maps $D_j \mapsto 4 - D_j$, so $D_j - 2$ is symmetric about zero and sign-flipping is a valid randomisation. That symmetry holds only because both agents played the identical cards from every seat.

With $m = 325$ simultaneous tests, both corrections are reported because they answer different questions. For raw p-values sorted $p_{(1)} \leq \cdots \leq p_{(m)}$,

$$\tilde{p}^{\,\text{Holm}}_{(k)} = \max_{j \leq k} \ \min\big\{1,\ (m-j+1)\,p_{(j)}\big\}, \qquad \tilde{p}^{\,\text{BH}}_{(k)} = \min_{j \geq k} \ \min\Big\{1,\ \tfrac{m}{j}\,p_{(j)}\Big\}. \tag{15}$$

Holm controls the family-wise error rate and BH the false-discovery rate, so $\tilde{p}^{\,\text{Holm}} \geq \tilde{p}^{\,\text{BH}}$ pointwise, which is why the surviving counts satisfy $208 \leq 260$.

All confidence intervals are percentile bootstrap intervals that resample deals, over $B = 2{,}000$ replicates, which keeps the four rotations of a deal together. Retrospective power uses the paired per-deal margin with standard deviation σ_D,

$$n_{\min} = \frac{(z_{1-\alpha/2} + z_{1-\beta})^2\,\sigma_D^2}{\delta^2}, \qquad \delta_{\min}(n) = \frac{(z_{1-\alpha/2} + z_{1-\beta})\,\sigma_D}{\sqrt{n}}. \tag{16}$$

Duplicate dealing turns the question of how much is luck into a two-way layout with one observation per deal and seat assignment, giving variance shares

$$\eta^2_{\text{deal}} = \frac{\sigma^2_{\text{deal}}}{\sigma^2_{\text{deal}} + \sigma^2_{\text{seat}} + \sigma^2_{\varepsilon}}, \tag{17}$$

with σ^2_{deal} and σ^2_{seat} estimated from the corresponding mean squares, Intransitivity is measured by the Hodge decomposition of the pairwise margin matrix into a gradient and a curl part, with cyclic energy

$$\kappa = \frac{\|\text{curl part}\|^2}{\|\text{gradient part}\|^2 + \|\text{curl part}\|^2}, \tag{18}$$

and its null distribution simulated under a transitive Bradley–Terry truth at the measured intra-deal correlation

$$\rho = \frac{\sigma^2_{\text{deal}}}{\sigma^2_{\text{deal}} + \sigma^2_{\varepsilon}}. \tag{19}$$

Remark 2. At $\rho = 0.097$ a binomial null understates the variance of the cycle count by a factor of $1 + 3\rho$, sets the bar too low, and would have declared this same data significant for the wrong reason. The beta-binomial null used here corrects for that over-dispersion.

Best-response exploitability is measured against an explicit oracle class, since Nash exploitability is only well defined for two-player zero-sum games. With $\mathfrak{B}_T$ the policies trained against target T and nothing else, and $\mathfrak{G}_T$ those that never trained against T,

$$e(T) = \max_{B \in \mathfrak{B}_T} w(B, T) \; - \; \max_{G \in \mathfrak{G}_T} w(G, T). \tag{20}$$

Remark 3. Equation (20) is a lower bound on the true best-response gap, and only that: it is exact only if $\mathfrak{B}_T$ contains a genuine best response. An oracle class too weak to beat T makes the first term small and reports $e(T) \leq 0$ whatever T's real exploitability is. This is not hypothetical and not specific to one algorithm: earlier DQN and PPO oracles both finished weaker than the heuristics they were meant to exploit, returning -0.11 to -0.20 on these same targets, and those numbers had to be discarded rather than reported.

Opponent specificity differences two win rates against the same opponent, so that opponent strength cancels. Writing π_o for a policy trained against opponent o and $w(\pi, o)$ for its win rate against o,

$$\sigma_o = w(\pi_o,\, o) \; - \; \frac{1}{|O| - 1} \sum_{o' \neq o} w(\pi_{o'},\, o), \tag{21}$$

against the naive alternative, which differences across opponents instead,

$$\text{naive}_o = w(\pi_o,\, o) \; - \; \frac{1}{|B|} \sum_{b \in B} w(\pi_o,\, b), \qquad B = \text{held-out opponents}. \tag{22}$$

Permutation importance replaces observation block b, at every decision, with the same slice taken from an unrelated real game state, and reports the drop

$$\Delta_b = w(\pi) - w(\pi \circ \Pi_b). \tag{23}$$

Splicing from a real state rather than zeroing the block is deliberate: zeroing changes the block's marginal distribution as well as its relationship to the position, and pushes the input off the data manifold.

For the endgame bound, let $V^*(z) \in \{0, 1\}$ be the double-dummy value of a handover position for the agent's team; the solver optimises the win indicator of Equation (6) directly, which is what makes it tractable, since a boolean value set lets minimax short-circuit. Then

$$\theta = \Pr\big(V^* = 1\big), \qquad c = \Pr\big(\text{win} \mid V^* = 1\big), \qquad s = \Pr\big(\text{win} \mid V^* = 0\big), \tag{24}$$

and since V^* partitions the positions, the realised win rate a and the reported gap follow by the law of total probability,

$$a = c\,\theta + s\,(1 - \theta), \qquad \text{gap} = \theta - a. \tag{25}$$

A negative gap means the agent's steals outweighed its conversion losses.

4.6 Agents

Seven hand-written agents form a ladder that isolates one idea at a time: random play; greedy trick-taking; partnership awareness, never overtaking one's own partner; ten-awareness; two adaptive policies; and *TensThenTricks*, which pursues tens first and takes tricks as a strict secondary objective (Supplementary Table S22). The ladder exists so that any advantage of a complex method can be attributed to something specific rather than to sophistication in general. Hand-tuning the balance between the two channels failed, which is itself informative, so a parameterised agent with a numeric genome is tuned offline by hill-climbing against a fixed opponent panel under duplicate-deal evaluation.

Since the channels partition the hands, any agent's overall rate is a mixture,

$$\Pr(\text{win}) = \Pr(\text{win} \mid \text{tens})\Pr(\text{tens}) + \Pr(\text{win} \mid \text{tie})\Pr(\text{tie}), \qquad \Pr(\text{tens}) + \Pr(\text{tie}) = 1. \tag{26}$$

PIMC and ISMCTS both sample hypothetical worlds consistent with everything the acting seat has observed, respecting the inferred voids of Equation (4). Writing $\mathcal{W}(I)$ for the worlds consistent with information set I and σ for the rollout policy, PIMC scores each legal move by the mean payoff of playing it and rolling out,

$$\hat{Q}_{\text{PIMC}}(a) = \frac{1}{W} \sum_{k=1}^{W} u\big(\sigma \mid a, \omega_k\big), \qquad \omega_k \sim \text{Unif}\, \mathcal{W}(I), \tag{27}$$

and plays the argmax. Each ω_k is solved as though the world were common knowledge, so Equation (27) estimates the mean over worlds of a perfect-information value rather than the value of the information set itself; the two differ, and PIMC cannot value information-gathering as a result.

ISMCTS instead builds one tree keyed by the action sequence from the root, so nodes are information sets, and draws a fresh world per iteration. Selection is UCB1 over the moves legal in the current world, with the exploration term normalised by

how often each move was available rather than by the parent's visit count:

$$a^\star = \operatorname*{argmax}_{a \in L(H_i,\ell)} \left[\bar{Q}(a) + c\sqrt{\frac{\log \sum_{a' \in L(H_i,\ell)} n(a')}{n(a)}} \right], \qquad c = 0.7. \tag{28}$$

The availability denominator is the whole difference between Equation (28) and plain MCTS: a move that is legal in few sampled worlds is not penalised for the iterations in which it could not be chosen. A multiple-observer variant differs by one subscript, in what a visited node is credited with after the rollout reaches a terminal z:

$$\text{SO:} \quad v(h) \mathrel{+}= u_{T(\text{root})}(z), \qquad \text{MO:} \quad v(h) \mathrel{+}= u_{T(\text{seat}(h))}(z). \tag{29}$$

The single-observer form backs up the root team's payoff at every node, including opponents' decision nodes, and then maximises it there, which is to say it searches opponents as though they were helping the searcher. The multiple-observer variant keeps a separate tree per seat and credits each node with the payoff of the side that actually moves there.

The learning agents are DQN with double and duelling variants; PPO; A2C; MAPPO under centralised training with decentralised execution, which the fixed partnerships make natural; NFSP; Deep CFR; and a policy distilled from ISMCTS. The observation is seat-relative, so a policy learned in one seat transfers to all four. By Equation (9) the honest reward is close to unusable: one signal at the end of a hand and nothing in between. Rather than pick a shaping function and report whatever it gave, we treat the choice as the experiment. Writing $R(z) = +1$ if the learner's team won and -1 otherwise, Δm and Δk for the tens and tricks gained on a transition, and T' for the opposing team, the four schemes are

$$r_{\text{terminal}} = R(z)\,\mathbf{1}[\text{done}], \tag{30}$$

$$r_{\text{ten}} = 0.25\,\Delta m_T - 0.25\,\Delta m_{T'} + R(z)\,\mathbf{1}[\text{done}], \tag{31}$$

$$r_{\text{potential}} = \Phi(s') - \Phi(s) + R(z)\,\mathbf{1}[\text{done}], \quad \Phi(s) = 0.66\,\frac{m_T - m_{T'}}{4} + 0.34\,\frac{k_T - k_{T'}}{13}, \tag{32}$$

$$r_{\text{trick}} = 0.05\,\Delta k_T + R(z)\,\mathbf{1}[\text{done}]. \tag{33}$$

Only Equation (32) has the potential-based form with $\gamma = 1$, so by Ng and colleagues [22] only it is guaranteed policy-invariant: it changes credit assignment without moving the optimal policy. Equations (31) and (33) are not of that form and can distort the objective, which is why they are included. The weights inside Φ approximate the measured channel split of Equation (26), which runs from 0.66 to 0.71 on the tens channel across the pairings we decomposed; they are an approximation to that measurement, not the measurement itself. Equation (33) is a deliberately misaligned control: trick count is the wrong objective under Equation (6), since 13.5% of hands are won by the side taking fewer tricks (Supplementary Fig. S6), so if the reward ablation separates anything this arm should lose.

4.7 Compute

Every experiment ran on the CPU of a single M1 Pro laptop, with no GPU. The study comprises 81 stored artifacts and 134.9 CPU-hours (Supplementary Table S23).

4.8 Use of generative AI

During the preparation of this work, the authors used Claude (Anthropic) in order to improve the grammar, punctuation, and readability of the manuscript. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication. No generative AI tool was used to generate the scientific content, experimental design, results, data analysis, or conclusions presented in this work. All references cited have been verified by the authors against their original sources.

Data Availability

The complete code base, all trained models, and every result artifact, including the per-deal outcome vectors for all 325 tournament pairings, are available from the corresponding author on reasonable request and will be released publicly on publication. Every statistic reported here can be recomputed from those files by a single script, and the identities asserted by the equations are rechecked against the artifacts by a second.

References

[1] McLeod, J. Card Games of Nepal. Pagat.com, National Card Games (n.d.). URL https://www.pagat.com/national/nepal.html.

[2] Juthani, A. Optimizing LLM strategies for playing Mendikot using prompt engineering. *International Journal for Multidisciplinary Research* **6** (2024).

[3] McLeod, J. Court Piece, Rang and Double Sir. Pagat.com, Rules of Card Games (n.d.). URL https://www.pagat.com/whist/rang.html.

[4] McLeod, J. Mendikot. Pagat.com, Rules of Card Games (n.d.). URL https://www.pagat.com/pointtrk/mendikot.html.

[5] McLeod, J. Dehla Pakad. Pagat.com, Rules of Card Games (n.d.). URL https://www.pagat.com/pointtrk/dehlapakad.html.

[6] Ginsberg, M. L. Gib: Imperfect information in a computationally challenging game. *Journal of Artificial Intelligence Research* **14**, 303–358 (2001). URL http://dx.doi.org/10.1613/jair.820.

[7] Cowling, P., Powley, E. & Whitehouse, D. Information set monte carlo tree search. *IEEE Transactions on Computational Intelligence and AI in Games* **4**, 120–143 (2012).

[8] Cowling, P. I., Ward, C. D. & Powley, E. J. Ensemble determinization in monte carlo tree search for the imperfect information card game magic: The gathering. *IEEE Transactions on Computational Intelligence and AI in Games* **4**, 241–257 (2012).

[9] Solinas, C., Rebstock, D. & Buro, M. Improving search with supervised learning in trick-based card games. *Proceedings of the AAAI Conference on Artificial Intelligence* **33**, 1158–1165 (2019). URL http://dx.doi.org/10.1609/aaai.v33i01.33011158.

[10] Amit, A. & Markovitch, S. Learning to bid in bridge. *Machine Learning* **63**, 287–327 (2006).

[11] Cohensius, G., Meir, R., Oved, N. & Stern, R. Bidding in spades (2020). URL https://arxiv.org/abs/1912.11323. arXiv:1912.11323.

[12] Mnih, V. *et al.* Human-level control through deep reinforcement learning. *Nature* **518**, 529–33 (2015).

[13] van Hasselt, H., Guez, A. & Silver, D. Deep reinforcement learning with double q-learning (2015). URL https://arxiv.org/abs/1509.06461. arXiv:1509.06461.

[14] Wang, Z. *et al.* Dueling network architectures for deep reinforcement learning (2016). URL https://arxiv.org/abs/1511.06581. arXiv:1511.06581.

[15] Schulman, J., Wolski, F., Dhariwal, P., Radford, A. & Klimov, O. Proximal policy optimization algorithms (2017). URL https://arxiv.org/abs/1707.06347. arXiv:1707.06347.

[16] Mnih, V. *et al.* Asynchronous methods for deep reinforcement learning (2016). URL https://arxiv.org/abs/1602.01783. arXiv:1602.01783.

[17] Bayen, A. *et al.* The surprising effectiveness of ppo in cooperative multi-agent games (2022).

[18] Hinton, G., Vinyals, O. & Dean, J. Distilling the knowledge in a neural network (2015). URL https://arxiv.org/abs/1503.02531. arXiv:1503.02531.

[19] Zinkevich, M., Johanson, M., Bowling, M. & Piccione, C. Regret minimization in games with incomplete information. *Proceedings of the 21st International Conference on Neural Information Processing Systems* 1729–1736 (2007).

[20] Heinrich, J. & Silver, D. Deep reinforcement learning from self-play in imperfect-information games (2016). URL https://arxiv.org/abs/1603.01121. arXiv:1603.01121.

[21] Brown, N., Lerer, A., Gross, S. & Sandholm, T. Deep counterfactual regret minimization (2019). URL https://arxiv.org/abs/1811.00164. arXiv:1811.00164.

[22] Ng, A. Y., Harada, D. & Russell, S. J. Policy invariance under reward transformations: Theory and application to reward shaping. *Proceedings of the Sixteenth International Conference on Machine Learning* 278–287 (1999).

[23] Zha, D. *et al.* Rlcard: A toolkit for reinforcement learning in card games (2020). URL https://arxiv.org/abs/1910.04376. arXiv:1910.04376.

[24] Malla, S. R. Heuristic optimization of AI agents for Dhumbal. *Scientific Reports* (2026).

[25] Raut, R., Subedi, A., Rai, S., Shakya, A. & Shakya, M. Deep reinforcement learning to master the asymmetric strategy of baghchal (2026). URL https://arxiv.org/abs/2607.18296. arXiv:2607.18296.

[26] Agarwal, S. & Iida, H. Analyzing thousand years old game tigers and goats is still alive. *Asia-Pacific Journal of Information Technology & Multimedia* **07**, 1–11 (2018).

[27] Banerjee, T., De, A., Maitra, S. & Mukherjee, D. Skill vs. chance quantification for popular card & board games (2025). URL https://arxiv.org/abs/2410.14363. arXiv:2410.14363.

[28] Anderson, E. F. *et al.* Developing serious games for cultural heritage: a state-of-the-art review. *Virtual Real.* **14**, 255–275 (2010).

Acknowledgements

Thanks to the many players across Sarlahi, Dhanusha, Mahottari, Siraha, Saptari, Parsa and Rautahat who taught the first author this game and played it with him over many years. They are unnamed here by their own preference.

Author contributions

R.K.Y. conceived the study, designed the methodology, wrote the software, curated the data, carried out the investigation and experiments, performed the formal analysis, produced the figures and tables, and drafted the original manuscript. **G.G.** supervised the project, provided conceptual guidance, contributed to the methodology, validated the research approach, interpreted the results, administered the project, and critically reviewed, revised and edited the manuscript. Both authors took part in discussing and interpreting the findings, appraised the manuscript critically for important intellectual content, and have read and approved the version submitted.

Additional Information

Competing interests. The authors declare no competing interests.

Funding. The authors received no funding for this work.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- 3SupplementaryInformationSearchbasedagentsoutperformreinforcementlearninginDahalJeetaNepaliimperfectinformationcardgame.pdf