Supplementary Information

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

Raju Kumar Yadav, Ganesh Gautam

Department of Electronics and Computer Engineering,
Institute of Engineering, Pulchowk Campus, Tribhuvan University, Lalitpur, Nepal

This file contains the supporting tables and figures referred to in the main article. Every table is generated directly from its result artifact; no value is transcribed by hand. Notation follows the main article's Methods, and every item here is cited at the corresponding point in the main text.

Contents

S1–S3	**The game family and its structure.** Feature-by-feature comparison against Mendikot; branching factor by trick; exact information-set size as uncertainty collapses.
S4	**Tournament protocol and statistical control.** Family-wise error control over all 325 pairings.
S5–S11	**Determinized search.** Strength against measured decision cost; determinization waste by trick; the void-constraint ablation at matched samples and at matched wall-clock; single- against multiple-observer ISMCTS; exploitability by target.
S12–S13	**Endgame optimality.** Exact double-dummy solve cost by depth; endgame conversion with each agent generating its own positions.
S14–S19	**Learning agents.** Hyperparameter sensitivity; reward schemes paired by seed; DQN against PPO at matched budgets; the state-encoding ablation; permutation feature importance; opponent specificity.
S20	**Deal-conditioned outcomes.** Trump length against tens dealt, on properties of the deal fixed before play.
S21–S23	**Reproducibility, agents and compute.** Refactor invariance and bit-identical retraining; the hand-written heuristic ladder; compute and hardware.
Figs. S1–S6	Learning curves; reward design under DQN; the state-encoding ablation; distillation from ISMCTS; spread across random seeds; structural statistics of the game.

The game family and its structure

Feature	Dahal Jeet	Mendikot	Same
Number of players	four	four	yes
Partnership structure	two fixed pairs, partners opposite	two fixed pairs, partners opposite	yes
Deck	standard 52 cards	standard 52 cards	yes
Rank order	A high down to 2	A high down to 2	yes
Cards per player	thirteen	thirteen	yes
Deal pattern	five, then four, then four	five, then the remainder in fours	yes
Turn rotation	one seat at a time in a fixed direction	one seat at a time in a fixed direction	yes
Trump fixed by a random draw	yes, unseen, from the drawer's own first five	yes, drawn by the player to the dealer's right	yes
Seat that fixes trump	the seat after the dealer	the player to the dealer's right	yes
Trump suit public once fixed	yes	yes, under the random-draw method	yes
Trump card returns to hand and stays playable	yes	not stated in the source	**no**
Hidden-trump option (band hukum)	not played	yes	**no**
Trump-on-first-void option (cut hukum)	not played	yes	**no**
Who leads the first trick	the seat that drew the trump	the player to the dealer's right	yes
Following suit compulsory	yes	yes	yes
No obligation to trump when void	yes, any card may be played	yes, any card may be played	yes
Trick winner	highest trump, else highest of the led suit	highest trump, else highest of the led suit	yes
Capture timing	immediate, every trick	immediate, every trick	yes
Scoring cards	the four tens only	the four tens only	yes
Hand won by tens majority	three or more tens	three or four tens	yes
Tiebreak at two tens each	decided on trick count	the side with seven or more tricks	yes
All four tens is a named result	coat	mendikot	yes
All thirteen tricks is a named result	double coat	52-card mendikot, or whitewash	yes
Next dealer comes from the losing side	yes	yes	yes
A special result skips the deal to the loser's partner	yes, triggered by all four tens	yes, triggered by a whitewash	**no**
A double special result cancels that skip	yes, a double coat cancels it	no such rule	**no**

Supplementary Table S1: Feature-by-feature comparison against Mendikot, the closest documented game: 21 of 26 features agree. Mendikot is taken from Pagat [1]; Dahal Jeet from the specification in §4.1, as implemented and tested in the released engine.

Feature granularity is a judgement of ours, and splitting or merging rows would change the denominator, so the full list is given rather than the ratio alone.

Trick	Mean legal moves
1	5.715
2	5.465
3	5.269
4	5.004
5	4.737
6	4.477
7	4.221
8	3.841
9	3.465
10	2.993
11	2.453
12	1.796
13	1.000

Supplementary Table S2: Branching factor by trick, measured over 156,000 decisions of random play. Follow-suit narrows the choice steadily as the hand proceeds.

Trick	Information-set size	Positions sampled
1	$10^{16.93}$	300
4	$10^{12.23}$	300
7	$10^{7.85}$	300
10	$10^{3.69}$	300
12	$10^{1.31}$	300

Supplementary Table S3: Exact information-set size from one seat's view, with inferred voids and the publicly shown trump card applied, counted by dynamic programming over eligibility patterns.

Uncertainty collapses by roughly fifteen orders of magnitude across a single hand, which is why determinized search is effective here and why sampling budget matters most early.

Tournament protocol and statistical control

Quantity	of 325 pairings
Nominally significant ($\alpha = 0.05$)	263
Expected false positives if uncorrected	16.2
Survive Holm–Bonferroni	208
Survive Benjamini–Hochberg	260

Supplementary Table S4: Family-wise error control over the full round-robin. The family is declared as all pairings, before analysis.

Determinized search

Family	Budget	Win rate	95% CI	ms/decision
PIMC	2	0.4771	[0.4396, 0.5125]	1.75
PIMC	4	0.5312	[0.4958, 0.5687]	3.54
PIMC	8	0.6229	[0.5833, 0.6604]	7.14
PIMC	16	0.6708	[0.6312, 0.7104]	14.48
PIMC	32	0.7312	[0.6958, 0.7667]	28.92
ISMCTS	50	0.5958	[0.5563, 0.6354]	9.49
ISMCTS	100	0.6292	[0.5875, 0.6750]	18.93
ISMCTS	200	0.7021	[0.6646, 0.7396]	38.10
ISMCTS	400	0.7063	[0.6667, 0.7438]	76.09
ISMCTS	800	0.7396	[0.7021, 0.7771]	152.53

Supplementary Table S5: Search strength against measured decision cost, versus the strongest heuristic. Budget is worlds for PIMC and iterations for ISMCTS.

Trick	Wasted (random play)	Wasted (skilled play)
1	0.0142	0.0112
2	0.0544	0.0675
3	0.1304	0.1215
4	0.2408	0.2406
5	0.3477	0.3408
6	0.4219	0.4854
7	0.5273	0.5731
8	0.6015	0.6321
9	0.6400	0.6596
10	0.6444	0.6773
11	0.6402	0.6277
12	0.5417	0.5150
13	0.1683	0.2017

Supplementary Table S6: Fraction of unconstrained determinizations that contradict the observed play. Measured under both random and skilled play, since the rate at which voids are revealed differs.

Family	Budget	Sampler	Win rate	95% CI	ms/dec
PIMC	4	constrained	0.5363	[0.5169, 0.5569]	3.24
PIMC	4	unconstrained	0.5238	[0.5025, 0.5444]	3.14
PIMC	8	constrained	0.6075	[0.5869, 0.6275]	6.56
PIMC	8	unconstrained	0.5806	[0.5613, 0.6000]	6.33
PIMC	16	constrained	0.6556	[0.6344, 0.6763]	13.19
PIMC	16	unconstrained	0.6231	[0.6031, 0.6450]	12.76
PIMC	32	constrained	0.7025	[0.6819, 0.7244]	26.63
PIMC	32	unconstrained	0.6725	[0.6512, 0.6937]	25.72
ISMCTS	50	constrained	0.5631	[0.5444, 0.5837]	8.63
ISMCTS	50	unconstrained	0.5406	[0.5206, 0.5600]	7.41
ISMCTS	100	constrained	0.6181	[0.5969, 0.6394]	17.28
ISMCTS	100	unconstrained	0.6075	[0.5869, 0.6281]	14.76
ISMCTS	200	constrained	0.6706	[0.6500, 0.6900]	34.42
ISMCTS	200	unconstrained	0.6212	[0.6019, 0.6406]	29.32
ISMCTS	400	constrained	0.6694	[0.6500, 0.6894]	68.75
ISMCTS	400	unconstrained	0.6494	[0.6294, 0.6694]	58.64

Supplementary Table S7: Void-constraint ablation, matched samples: identical world and iteration counts for both samplers.

Family	Budget	Constrained	Matched budget	Unconstrained	Δ	Time error	Paired p
ISMCTS	50	0.5631	58	0.5687	-0.0056	-0.8%	0.7107
ISMCTS	100	0.6181	117	0.6144	+0.0037	-0.0%	0.8257
ISMCTS	200	0.6706	236	0.6325	+0.0381	+0.2%	0.0025
ISMCTS	400	0.6694	473	0.6462	+0.0232	+0.1%	0.0716
PIMC	4	0.5363	4	0.5238	+0.0125	-3.6%	0.3702
PIMC	8	0.6075	8	0.5806	+0.0269	-3.7%	0.0487
PIMC	16	0.6556	16	0.6231	+0.0325	-3.5%	0.0146
PIMC	32	0.7025	33	0.6787	+0.0238	-0.8%	0.0590

Supplementary Table S8: Void-constraint ablation, matched wall-clock. The unconstrained sampler is actually run at a budget chosen to match the constrained arm's measured time; both arms play the same deals, so the test is paired.

Eight simultaneous tests. Under Holm–Bonferroni only the ISMCTS-200 arm survives at $\alpha = 0.05$.

Variant	Iterations	Win rate	95% CI	ms/dec
SO-ISMCTS	100	0.6138	[0.5875, 0.6400]	19.93
MO-ISMCTS	100	0.6262	[0.5975, 0.6525]	20.01
SO-ISMCTS	200	0.6450	[0.6175, 0.6725]	39.07
MO-ISMCTS	200	0.6950	[0.6663, 0.7212]	38.88
SO-ISMCTS	400	0.6700	[0.6412, 0.7000]	76.14
MO-ISMCTS	400	0.7050	[0.6750, 0.7338]	76.66

Supplementary Table S9: Single-observer versus multiple-observer ISMCTS. The single-observer form backs up the root team's payoff at every node, so opponents are searched as though they were helping the searcher.

Iterations	MO win rate vs SO	95% CI	Verdict
100	0.5038	[0.4775, 0.5288]	tie
200	0.5225	[0.4988, 0.5475]	tie
400	0.5375	[0.5138, 0.5625]	MO wins

Supplementary Table S10: Multiple- against single-observer ISMCTS, head to head on identical deals.

Target	Best response	Generic search	Exploitability	Paired p
GreedyTricks	0.7525	0.6817	+0.0708	<0.001
Adaptive	0.7683	0.7042	+0.0642	<0.001
TensThenTricks	0.7075	0.6558	+0.0517	<0.001
TenAware	0.7508	0.7108	+0.0400	0.0093
Random	0.8492	0.8458	+0.0033	0.8465

Supplementary Table S11: Exploitability: a search that simulates the target's true policy for the opponent seats, against the same search at the same budget with no opponent model. Identical algorithm in both arms, paired on identical deals.

Endgame optimality

Tricks left	Solved	Mean s (solved only)	Censored lower bound	Nodes	Biased
1	8/8	0.000	0.000	1	
2	8/8	0.000	0.000	4	
3	8/8	0.000	0.000	32	
4	8/8	0.000	0.000	119	
5	8/8	0.003	0.003	916	
6	8/8	0.006	0.006	1,678	
7	8/8	0.046	0.046	13,276	
8	8/8	0.840	0.840	202,685	
9	7/8	2.705	4.867	631,237	yes
10	3/8	13.961	17.735	3,019,164	yes
11	3/8	2.822	13.558	604,228	yes
12	1/8	9.519	18.690	1,841,947	yes
13	1/8	1.183	17.648	282,687	yes

Supplementary Table S12: Exact double-dummy solve cost by depth. Cost grows roughly tenfold per additional trick.

Where positions abort, the solved-only mean is subject to survivorship bias and can fall with depth, because only easy positions finish. Use the censored lower bound.

Agent	Theoretical	Actual	Conversion	Steal	Gap
Random	0.3833	0.2533	0.548	0.070	+0.1300
GreedyTricks	0.4883	0.4450	0.747	0.156	+0.0433
TenAware	0.4533	0.4233	0.713	0.183	+0.0300
Adaptive	0.4533	0.4250	0.721	0.180	+0.0283
TensThenTricks	0.5000	0.5000	0.810	0.190	+0.0000
PIMC:16	0.5300	0.6467	0.931	0.326	-0.1167
ISMCTS:200	0.5083	0.6300	0.902	0.349	-0.1217

Supplementary Table S13: Endgame optimality with each agent generating its own positions. Conversion rates are not comparable across agents here, because a stronger agent reaches different endgames.

Learning agents

Learning rate	Hidden	γ	vs best heuristic
0.0001	512	0.99	0.3887
0.0003	512	0.99	0.3525
0.001	512	0.99	0.3275
0.0003	256	0.99	0.3688
0.0003	1024	0.99	0.3400
0.0003	512	0.95	0.3525
0.0003	512	1.0	0.3250

Supplementary Table S14: Hyperparameter sensitivity at 20,000 episodes. Spread 0.0637.

Algo	A	B	Pairs	mean(A−B)	SD	Consistent
dqn	dqn-potential	dqn-ten_shaped	3	-0.0367	0.0133	yes
dqn	dqn-potential	dqn-terminal	3	-0.0053	0.0191	no
dqn	dqn-potential	dqn-trick_shaped	3	+0.0069	0.0041	yes
dqn	dqn-ten_shaped	dqn-terminal	3	+0.0314	0.0204	yes
dqn	dqn-ten_shaped	dqn-trick_shaped	3	+0.0436	0.0143	yes
dqn	dqn-terminal	dqn-trick_shaped	3	+0.0122	0.0232	no
ppo	ppo-potential	ppo-ten_shaped	3	+0.0011	0.0110	no
ppo	ppo-potential	ppo-terminal	3	+0.0155	0.0131	no
ppo	ppo-potential	ppo-trick_shaped	3	+0.0033	0.0147	no
ppo	ppo-ten_shaped	ppo-terminal	3	+0.0144	0.0069	yes
ppo	ppo-ten_shaped	ppo-trick_shaped	3	+0.0022	0.0041	no
ppo	ppo-terminal	ppo-trick_shaped	3	-0.0122	0.0096	yes

Supplementary Table S15: Reward schemes compared seed by seed. Pairing on the seed removes the seed's own contribution; 'consistent' means every seed agreed on the direction.

Algo	Matched on	Target	Achieved	Episodes	Updates	vs best heuristic
dqn	wall clock	300	355	14,559	94,134	0.3387
dqn	wall clock	300	370	14,559	94,134	0.3337
ppo	wall clock	300	263	51,402	2,600	0.4800
ppo	wall clock	300	260	51,402	2,600	0.4713
dqn	wall clock	900	1071	43,678	283,408	0.3387
dqn	wall clock	900	1027	43,678	283,408	0.3837
ppo	wall clock	900	772	154,208	7,808	0.4938
ppo	wall clock	900	773	154,208	7,808	0.5100
dqn	gradient steps	5000	5234	882	5,234	0.3438
dqn	gradient steps	5000	5234	882	5,234	0.3175
ppo	gradient steps	5000	5424	107,142	5,424	0.5062
ppo	gradient steps	5000	5424	107,142	5,424	0.5050
dqn	gradient steps	15000	16700	2,646	16,700	0.3362
dqn	gradient steps	15000	16700	2,646	16,700	0.3613
ppo	gradient steps	15000	16272	321,428	16,272	0.5138
ppo	gradient steps	15000	16272	321,428	16,272	0.4913

Supplementary Table S16: DQN and PPO compared at matched wall-clock and again at matched gradient steps. Matching on episodes, as elsewhere in the literature, gives the two very different optimisation budgets.

DQN performs 122× more gradient steps per episode than PPO.

Blocks zeroed	Mean	SD	Seeds	Per-seed
minimal	0.4273	0.0262	3	0.4081, 0.4094, 0.4644
no_scalars	0.3671	0.0091	3	0.3550, 0.3694, 0.3769
no_voids	0.3642	0.0112	3	0.3513, 0.3625, 0.3787
no_ten_status	0.3634	0.0217	3	0.3394, 0.3588, 0.3919
none	0.3629	0.0182	3	0.3494, 0.3506, 0.3887
no_history	0.3569	0.0227	3	0.3375, 0.3444, 0.3887

Supplementary Table S17: State-encoding ablation at 120,000 episodes, matched to the headline agents' budget.

No single block's removal changes performance materially, yet removing almost everything improves it, with no overlap between the seeds of 'minimal' and the full encoding.

Model	Block permuted	Features	Win rate	Drop
rl_ppo_potential_s0.pt	hand	52	0.5119	-0.0044
rl_ppo_potential_s0.pt	history	208	0.4906	+0.0169
rl_ppo_potential_s0.pt	ten_status	12	0.5062	+0.0012
rl_ppo_potential_s0.pt	current_trick	52	0.4694	+0.0381
rl_ppo_potential_s0.pt	voids	16	0.4988	+0.0087
rl_ppo_potential_s0.pt	scalars	5	0.5038	+0.0037
rl_dqn_ten_shaped_s0.pt	hand	52	0.3731	-0.0119
rl_dqn_ten_shaped_s0.pt	history	208	0.3063	+0.0550
rl_dqn_ten_shaped_s0.pt	ten_status	12	0.3506	+0.0106
rl_dqn_ten_shaped_s0.pt	current_trick	52	0.3319	+0.0294
rl_dqn_ten_shaped_s0.pt	voids	16	0.3425	+0.0187
rl_dqn_ten_shaped_s0.pt	scalars	5	0.3631	-0.0019

Supplementary Table S18: Permutation feature importance. Each block is replaced at every decision by the same slice from an unrelated real state, which destroys its relationship to the position while preserving its marginal distribution.

Algo	Opponent	Trained on it	Trained elsewhere	Specificity
dqn	Random	0.5690	0.5793	-0.0102
dqn	GreedyTricks	0.3697	0.3280	+0.0418
dqn	TenAware	0.3689	0.3443	+0.0246
dqn	Adaptive	0.3817	0.3497	+0.0320
dqn	TensThenTricks	0.3436	0.3049	+0.0387
ppo	Random	0.7626	0.6994	+0.0632
ppo	GreedyTricks	0.4822	0.4464	+0.0358
ppo	TenAware	0.5364	0.5095	+0.0268
ppo	Adaptive	0.5467	0.5149	+0.0318
ppo	TensThenTricks	0.4995	0.4608	+0.0386

Supplementary Table S19: Opponent specificity: the advantage from having trained against this specific opponent, with opponent strength cancelled because both terms are measured against the same opponent.

Mean +0.0323 ± 0.0174 over 10 opponents. A naive gap (own minus held-out) would instead report roughly +0.30, but that compares performance against different opponents and mostly measures their relative strength.

Deal-conditioned outcomes

Condition	Value	Win rate	Hands
tens_dealt	0	0.373	284
tens_dealt	1	0.433	1,182
tens_dealt	2	0.500	1,868
tens_dealt	3	0.567	1,182
tens_dealt	4	0.627	284
trump_length	2	0.000	24
trump_length	3	0.061	98
trump_length	4	0.193	342
trump_length	5	0.249	812
trump_length	6	0.454	1,124
trump_length	7	0.546	1,124
trump_length	8+	0.785	1,276
holds_trump	False	0.499	2,400
holds_trump	True	0.501	2,400
decided_by	tens	0.500	3,092
decided_by	tiebreak	0.500	1,708

Supplementary Table S20: Outcomes conditioned on properties of the deal fixed before play, for TensThenTricks in self-play. Trump length counts the team's trumps between them and its top bucket is eight or more. A low win rate in a bucket is a property of the situation, not of how it was played.

Reproducibility, agents and compute

Check	Scope	Result	max $\|\Delta\|$
Buffer optimisation, all configurations	24 configs	bit-identical	0.0e+00
Stored checkpoint reproduced (120,000 episodes)	rl_dqn_potential_s0.pt	bit-identical	0.0e+00

Supplementary Table S21: A refactor and a performance optimisation applied mid-project altered no result. In the second row, a checkpoint trained by the original code is reproduced exactly by the current code, which is what licenses reusing earlier artifacts.

Rank	Heuristic	Mean win rate
1	Tuned	0.6147
2	TensThenTricks	0.5979
3	GreedyTricks	0.5446
4	AdaptiveGreedy	0.5446
5	Balanced	0.5281
6	TenAware	0.5268
7	Adaptive	0.5243
8	PartnerAware	0.5193
9	TenHunter	0.4514
10	TenDefender	0.4363
11	Random	0.3571
12	Lowest	0.3550

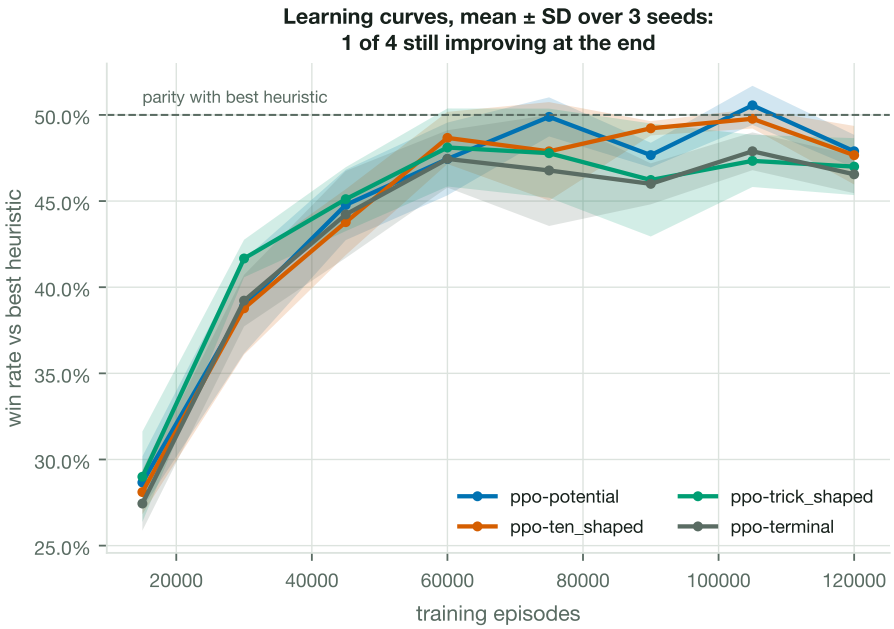
Supplementary Table S22: Hand-written heuristic ladder, 1200 duplicate deals per pairing. Each rung adds one idea to the rung below advantage can be attributed to something specific.

Quantity	Value
Processor	M1 Pro
Logical cores	10
Memory	16.0 GiB
Accelerator	none; CPU only
Result artifacts	81
Measured CPU-hours	134.9

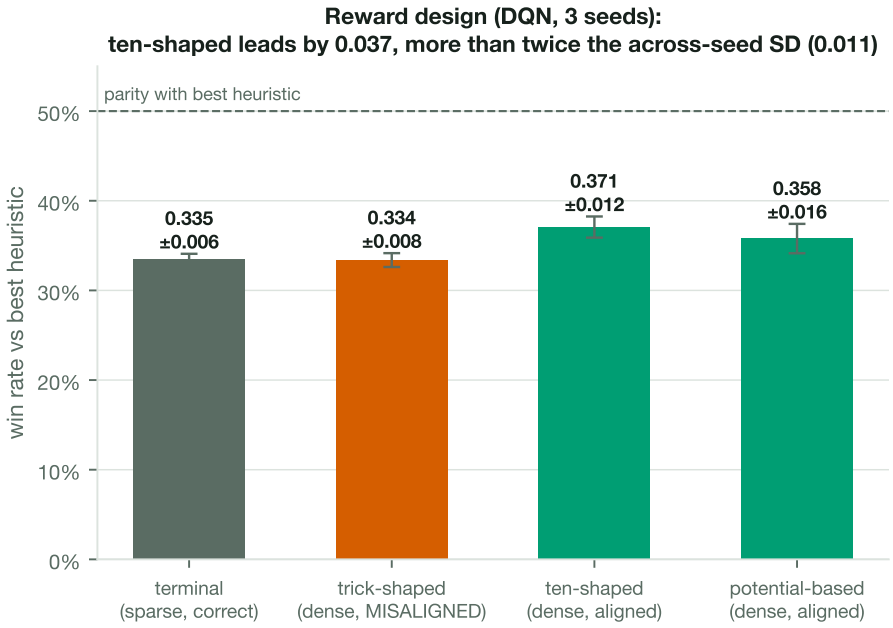
Supplementary Table S23: Compute and hardware. Every experiment ran on the CPU of a single laptop; no GPU acceleration was used.

The platform's integrated GPU backend was benchmarked and rejected: at batch size 1, which dominates reinforcement-learning rollouts, it is roughly four times slower than the CPU for networks of this size.

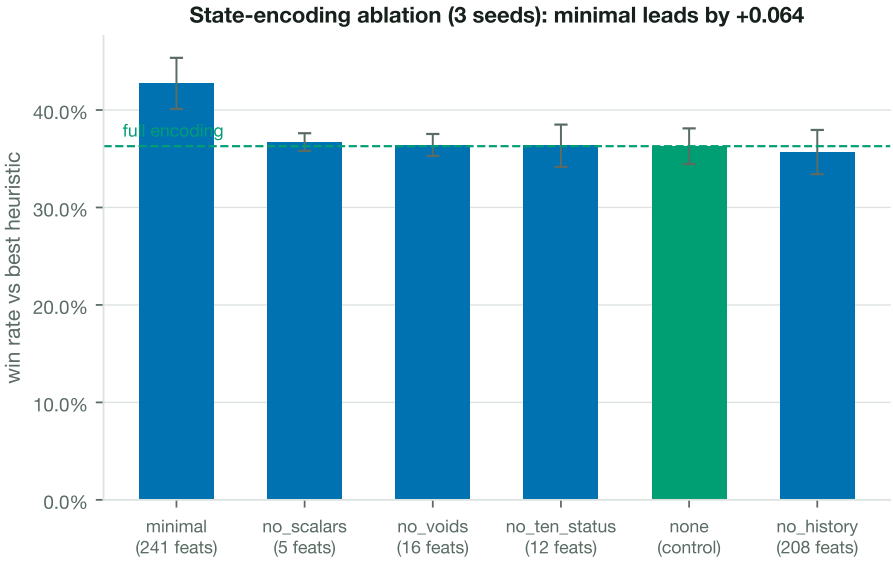
Supplementary figures



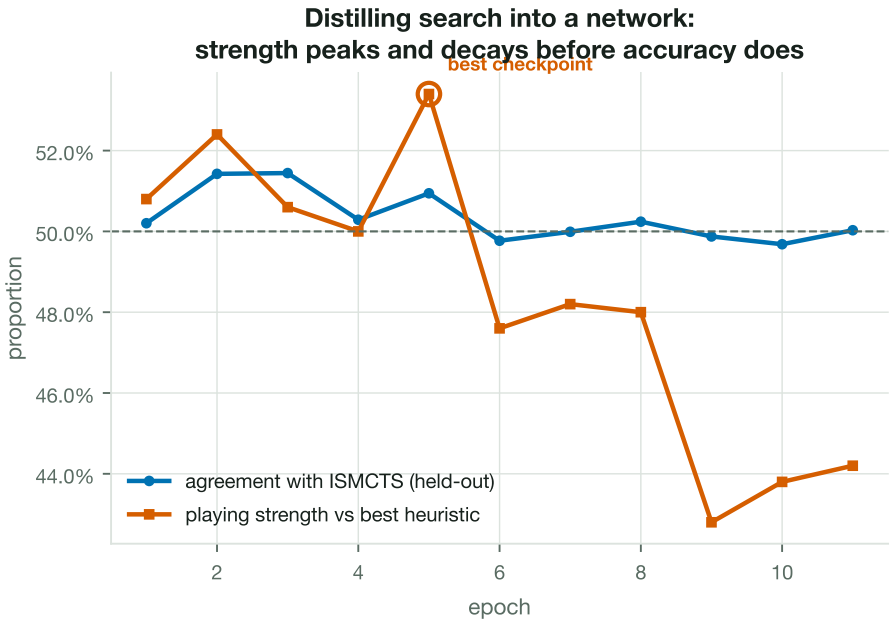
Supplementary Fig. S1: Learning curves, mean ± one standard deviation across seeds. The curves flatten well before the training budget is exhausted, so the shortfall against search is not simply insufficient training.



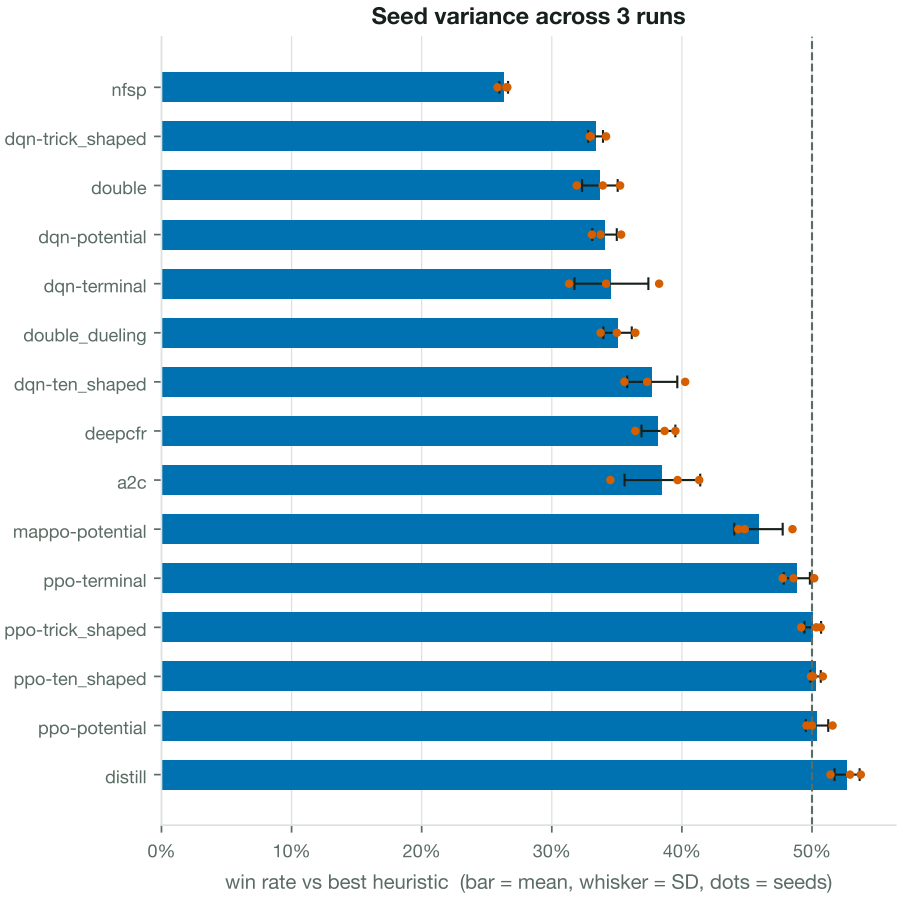
Supplementary Fig. S2: Reward design under DQN, mean ± standard deviation over three seeds. Ten-shaped rewards win consistently here, where under PPO nothing separates.



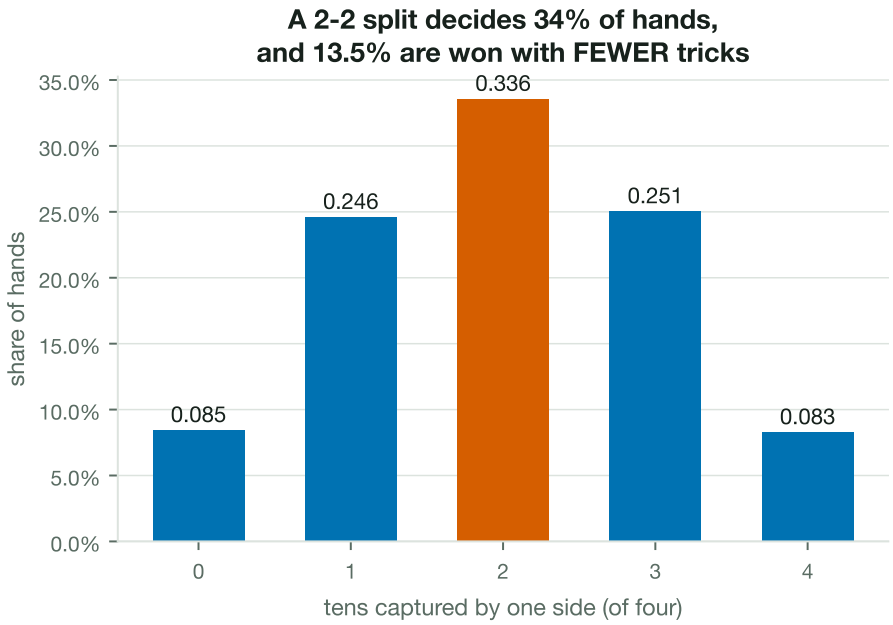
Supplementary Fig. S3: State-encoding ablation. No single block earns its place, yet removing almost all of them improves performance.



Supplementary Fig. S4: Distillation from ISMCTS. Validation accuracy is flat while playing strength peaks and then decays, which is why checkpoints are selected on win rate rather than on loss.



Supplementary Fig. S5: Spread across random seeds, which is why every learning result in the main article is reported over three seeds.



Supplementary Fig. S6: Structural statistics of Dahal Jeet measured over 30,000 random-play hands. The share of hands won by the side taking fewer tricks is what makes the trick-reward arm a misaligned control rather than a useless one.

Reference cited in a supplementary caption

[1] John McLeod. Mendikot. Pagat.com, Rules of Card Games, n.d. URL `https://www.pagat.com/pointtrk/mendikot.html`.