KAIST · CS377 · TEAM 5

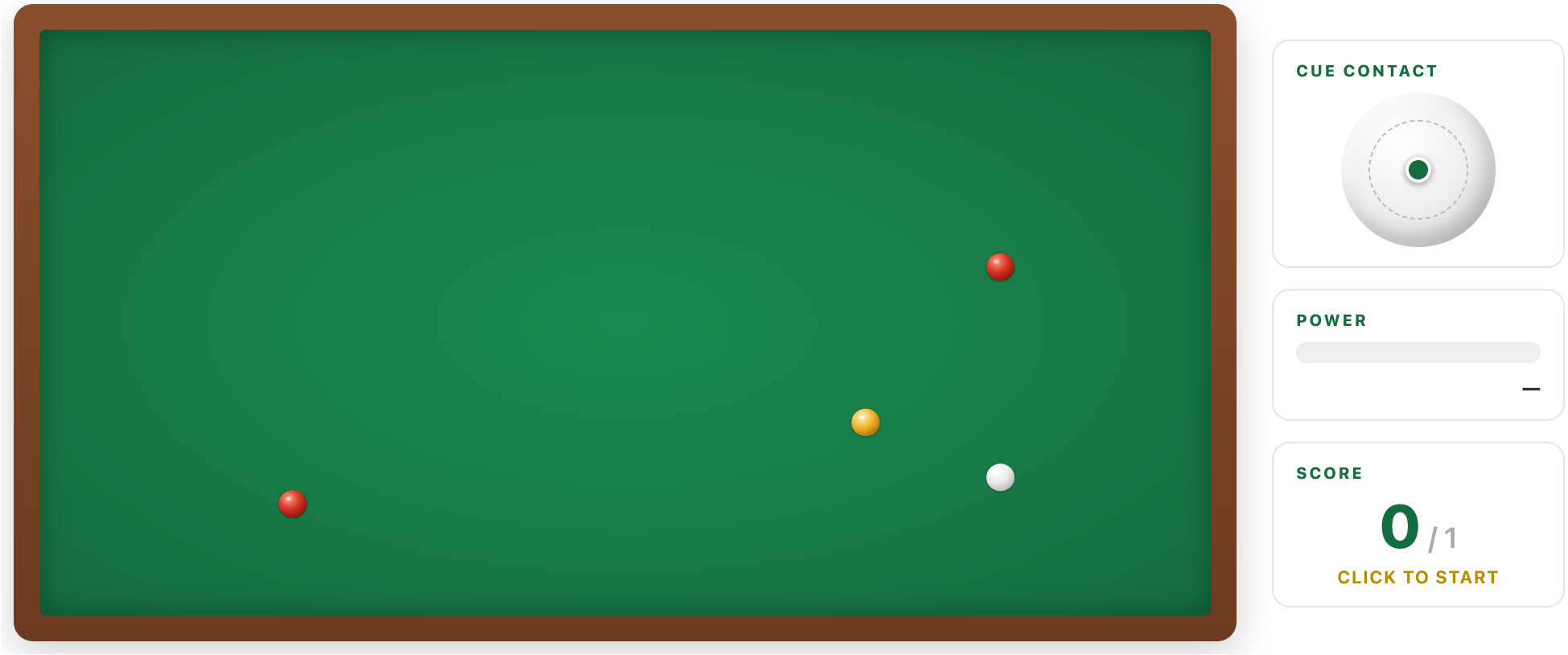
Korean 4-Ball Billiards

A Continuous, Deterministic, Sparse-Reward Benchmark Solved by Inference-Time Search

Doyeol Oh · Byungmo Kang · Seojun Park — KAIST

The game

Hit both red balls in one shot to score. The opponent's cue ball is a foul.



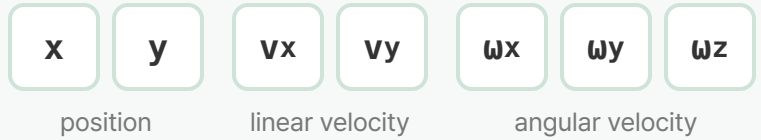
click the table — a 1-second aim preview (direction · power · contact), then the shot

The environment

A NumPy billiards simulator, exposed as a Gymnasium env.

ONE BALL

7 FEATURES



× 4 balls

= 28D - STATE

ACTION

`(θ, p, a, b)` — aim, power, two spin offsets

SIMULATOR

≈ 5 ms per shot — cheap enough to verify

REWARD

1 only if the cue ball caroms **both reds** (no foul), else **0**

FOUL

cue ball hits the opponent ball — penalties off by default

BOUNDED EVAL

100 random-start innings · `max_shots = 10`

CHAIN EVAL

terminate-on-miss innings · long-horizon search only

Physics: standard cue-impact, rolling/sliding, ball–ball contact & cushion-rebound models [Marlow 1995 · Han 2005 · Alciatore 2004]

A random shot
almost never scores.

Continuous action | Deterministic physics | Sparse reward

Which part actually solves it?

01

Model-free learning

→ low ceiling

02

Domain knowledge

→ competent proposer

03

Reward shaping

→ little effect

04

Inference-time search

→ beyond the horizon

LEVER ① · MODEL-FREE LEARNING

First — can standard RL just learn to score?

METHOD · PLAIN RL

SAC, TD3, PPO — how they differ

TD3 OFF-POLICY

0 | +1 | 0 | +1 ↻ replay (reuse)

- **deterministic** policy + target smoothing
- twin clipped-Q · delayed updates

highest mean · high seed variance

SAC OFF-POLICY

+1 | 0 | 0 | +1 ↻ replay (reuse)

- **stochastic** policy + **entropy** bonus
- twin-Q · auto temperature

most stable → our default

PPO ON-POLICY

+1 | 0 | 0 → 🗑 used once

- policy gradient on **fresh rollouts**
- **stochastic** policy · value baseline

no replay → weak on sparse reward

Only **TD3 / SAC replay** the rare **+1** shots (~4% of all transitions) — that reuse is why they clear PPO by ~2.5×.

RESULT

Off-policy beats PPO by ~2.5×

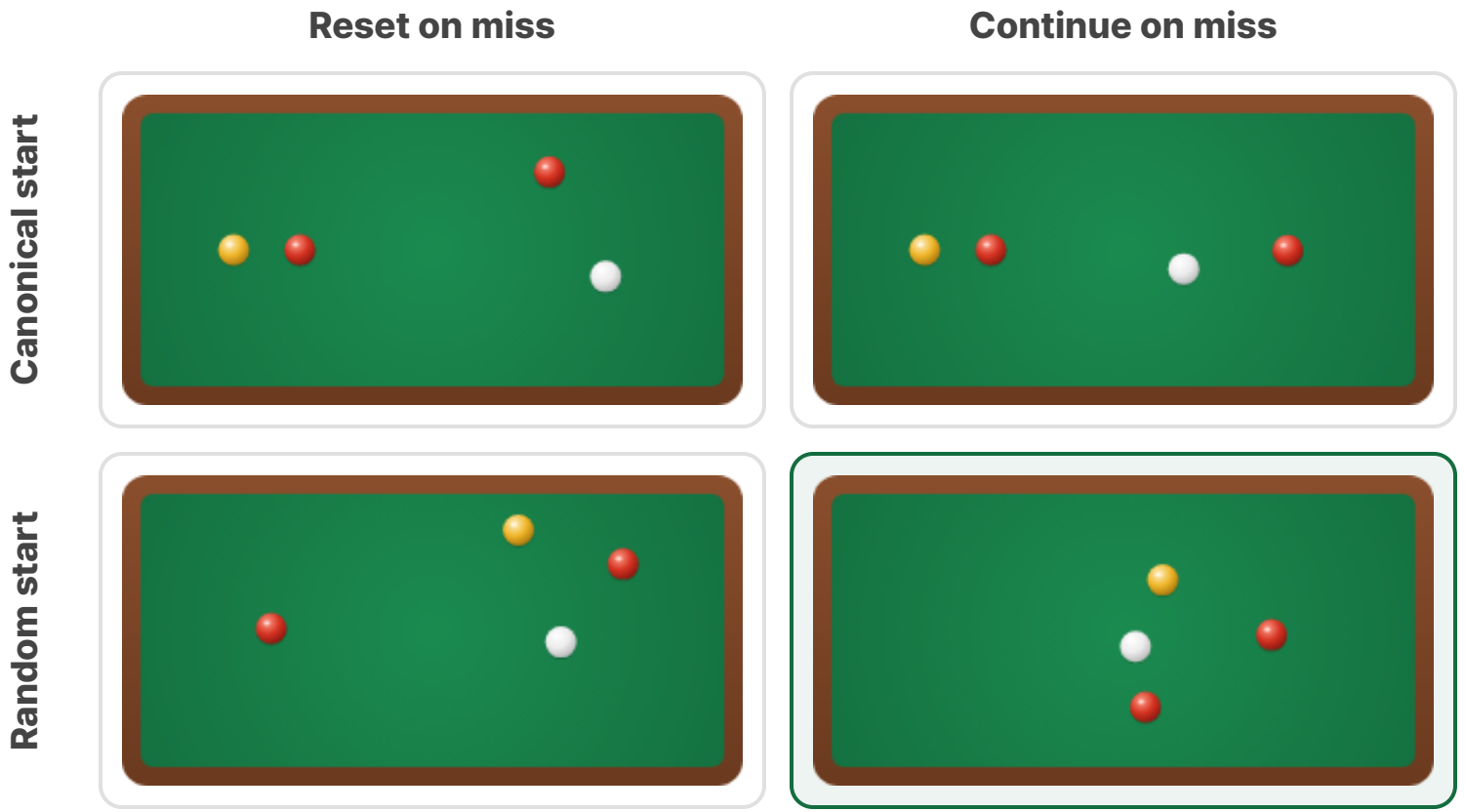
TABLE 2 · ALGORITHM COMPARISON · 400K STEPS · 5 SEEDS

Algorithm	Mean ± std	Seed range	Foul/sh	Success/sh
TD3	0.460 ± 0.114	0.33–0.62	19.5%	4.6%
SAC	**0.418 ± 0.038**	**0.37–0.48**	**20.2%**	**4.2%**
PPO	0.170 ± 0.040	0.11–0.22	14.4%	1.7%

> Off-policy ≈ **2.5×** PPO · TD3 highest mean, but SAC is the **stable default**.

METHOD · PLAIN RL

Start state × miss handling



Continue-on-miss keeps playing from each resulting state → exposes many mid-rack situations per episode.

RESULT

Random-start + continue generalizes best

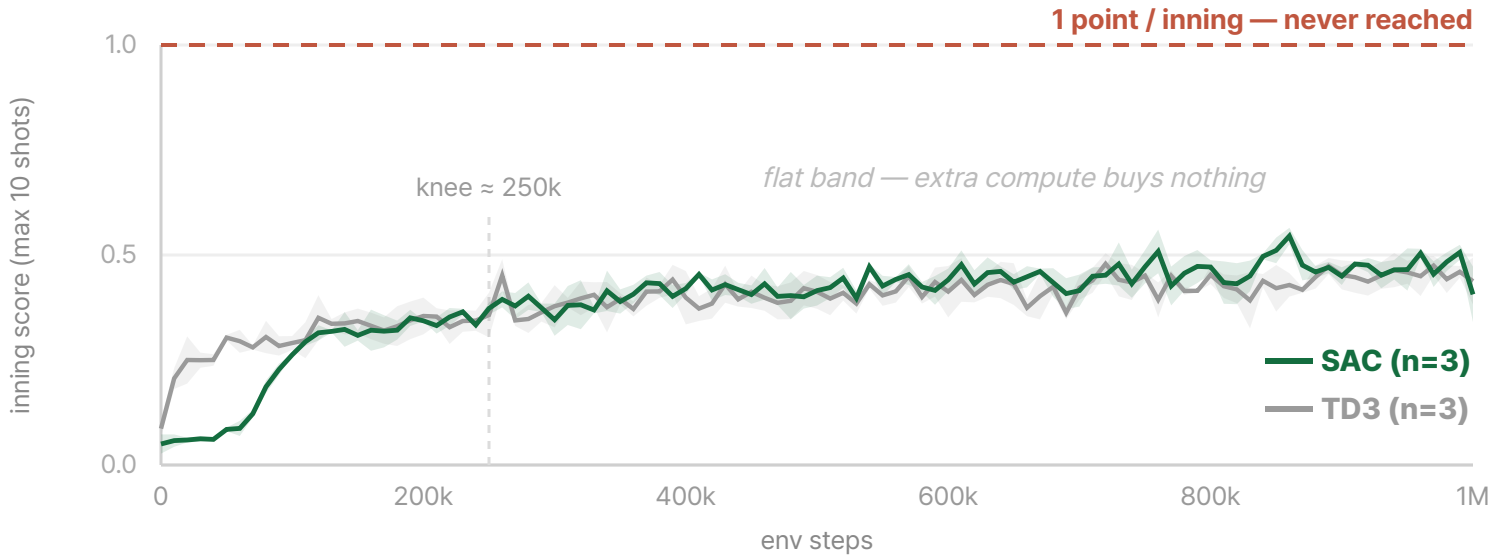
TABLE 3 · FOUR TRAINING PARADIGMS · RANDOM-START = MAIN METRIC

Training method	Random mean ± std	Canonical mean	Foul/sh	Success/sh
canonical, reset	0.073 ± 0.045	1.00 ± 0.82	9.3%	0.7%
canonical, continue	0.353 ± 0.101	1.33 ± 0.47	20.4%	3.5%
random, reset	0.407 ± 0.046	0.67 ± 0.47	18.6%	4.1%
random, continue	**0.453 ± 0.012**	**0.67 ± 0.94**	**20.3%**	**4.5%**

> Best mean & **lowest variance** — training curves alone would mis-rank it.

PLAIN RL · BUDGET CHECK

The score plateaus early — and stays low



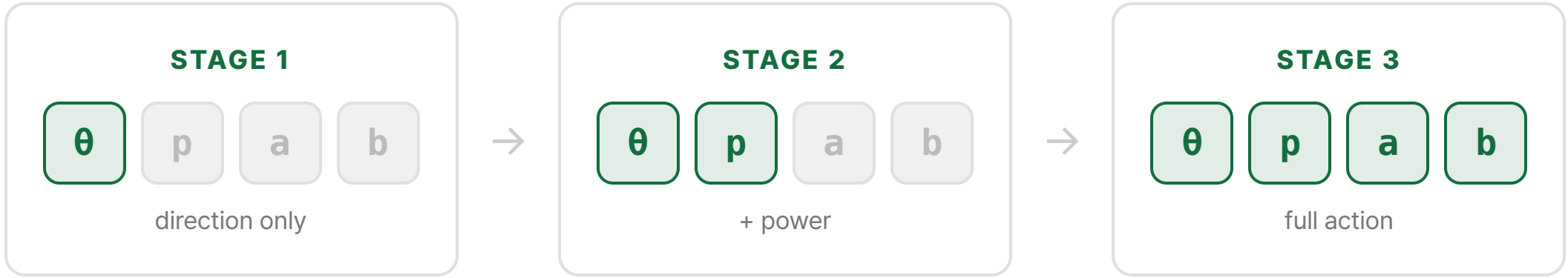
3-seed mean ±std. Both knee by **~250k**, then flatten in the **0.40–0.47** band — well under 1 point.
Not compute-limited — it **lacks structure**.

LEVER ① · IMPROVING THE BASELINE

Plain RL stalls. Can we help it without new structure?

METHOD · STRUCTURE-FREE ATTEMPT ①

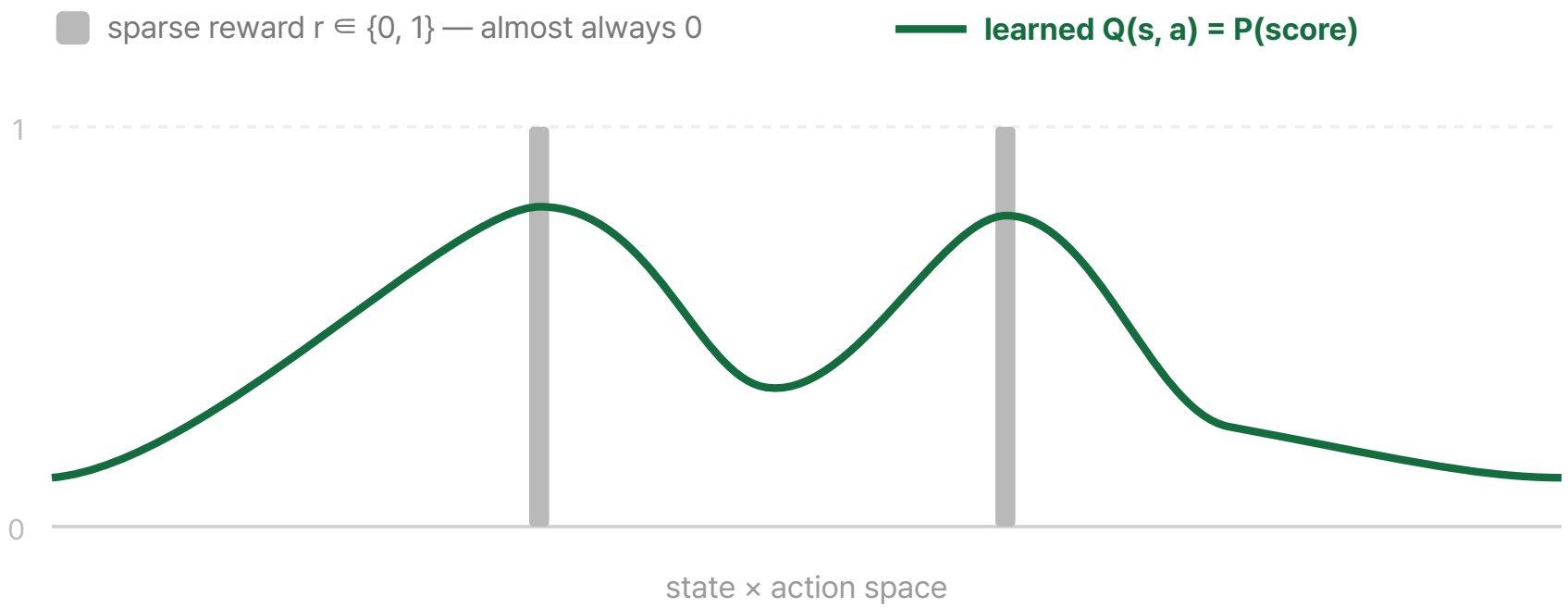
Action curriculum



Unlock the action **one dimension at a time** — direction → +power → full — warm-started across stages.

METHOD · STRUCTURE-FREE ATTEMPT ②

Reward-model guidance



A small MLP `Q(s,a) = P(score | s,a)` trained offline (BCE) adds a **dense bonus between the rare rewards**; evaluation still uses the true carom score.

RESULT

Neither breaks the ceiling

TABLE 5 · IMPROVING-BASELINE ATTEMPTS · 1M STEPS · 3 SEEDS

Method	Mean ± std	Max	Foul/sh	Success/sh
plain SAC baseline	0.487 ± 0.097	4	20.5%	4.9%
staged action curriculum	0.407 ± 0.052	4	18.4%	4.1%
RM-guided SAC	**0.840 ± 0.000**	**4**	**18.7%**	**8.4%**

> Even the best (0.840) stays **under 1** → the gain must come from **domain knowledge**.

PLAIN RL · STRUCTURE-FREE VERDICT

Why both were set aside

✕ Staged curriculum

0.407 vs 0.487 plain

Below plain

exploration wasn't the bottleneck

✕ Reward-model guidance

0.840 best, still < 1

Expensive

no compounding with geometry

5 ms sim **verifies exactly** → RM redundant

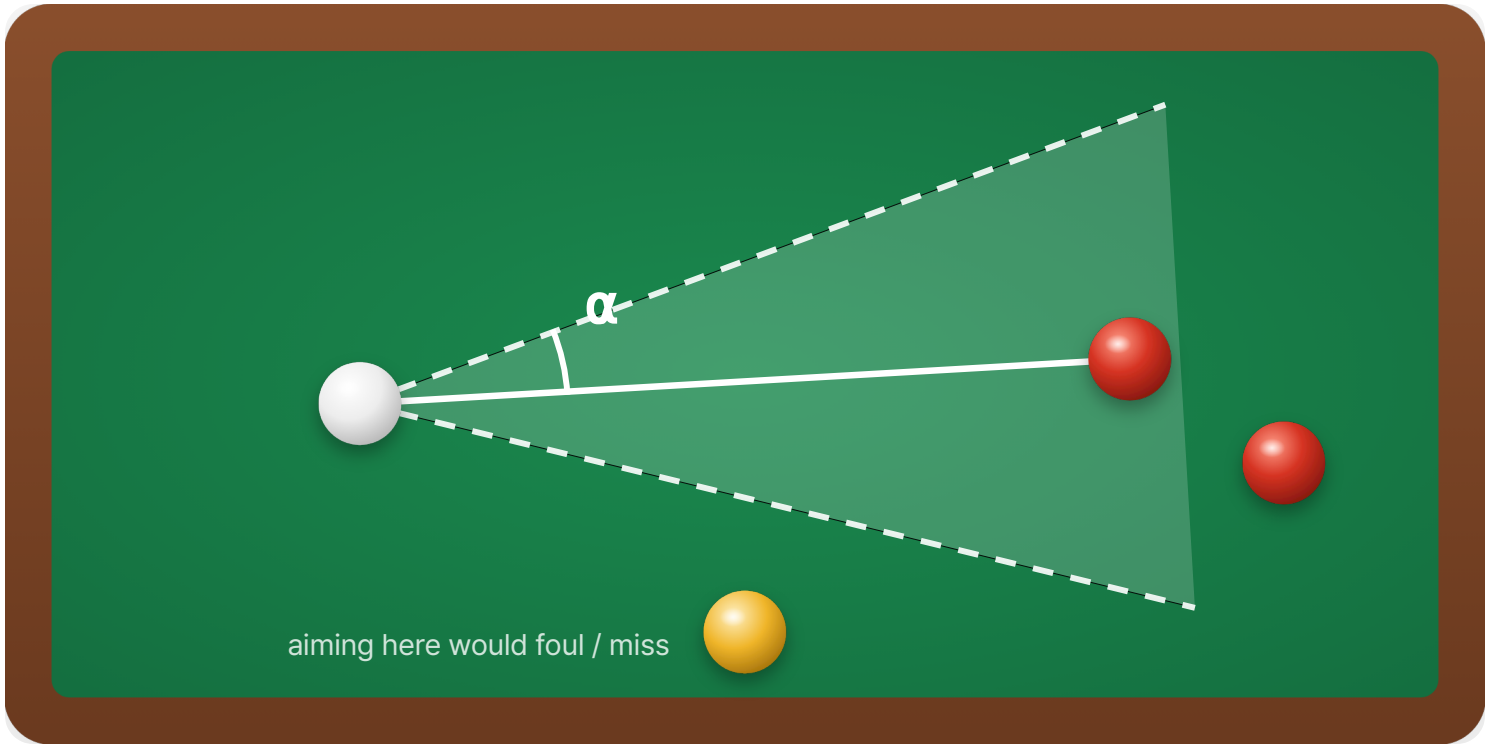
Negative results · no structure → no ceiling break → **domain knowledge next**

LEVER ② · DOMAIN KNOWLEDGE

What if we hand it the geometry?

METHOD · DOMAIN KNOWLEDGE ⓘ

Aim constraint — a guaranteed-contact cone



The policy aims **inside the cone** (where), never outside it (whether).

```
θ = target ± α·offset,  α = arcsin(2r / d)
```

RESULT · DOMAIN KNOWLEDGE ①

A 5.3× jump from one reparameterization

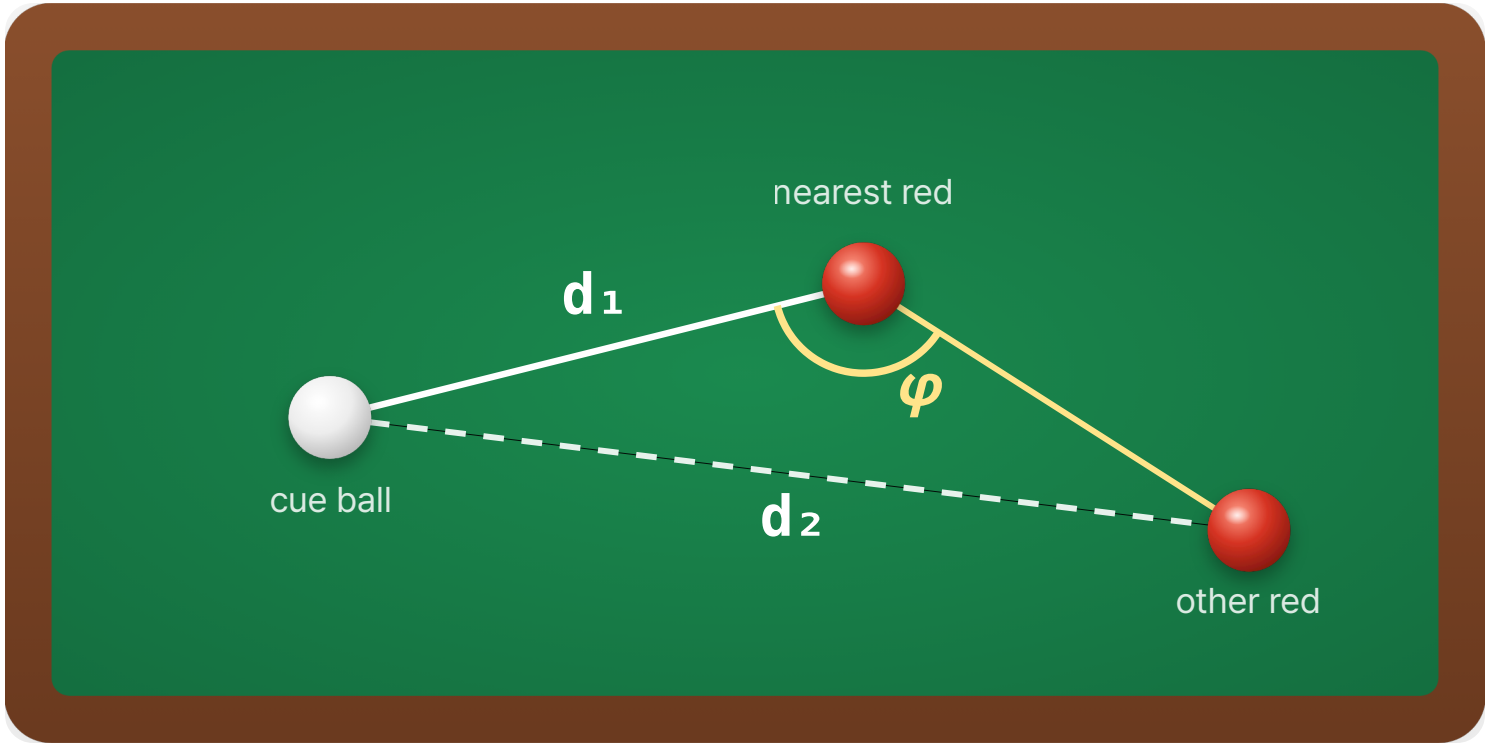
0.487 → **2.570**

plain baseline | + aim constraint

5.3× mean score · per-shot success 4.9% → **25.7%** · fouls drop too.

METHOD · DOMAIN KNOWLEDGE ②

Carom-geometry features in the observation



Append the carom geometry the policy would otherwise infer: `(d₁, d₂, sin φ, cos φ)`

RESULT · DOMAIN KNOWLEDGE ②

Geometry is the biggest lever

0.487 → **2.570** → **6.460**

plain baseline | + aim constraint | + geometry features

> per-shot success **4.9% → 25.7% → 64.6%** — explicit geometry beats changing the reward.

RESULT · DOMAIN KNOWLEDGE ②

The largest single jump in the project

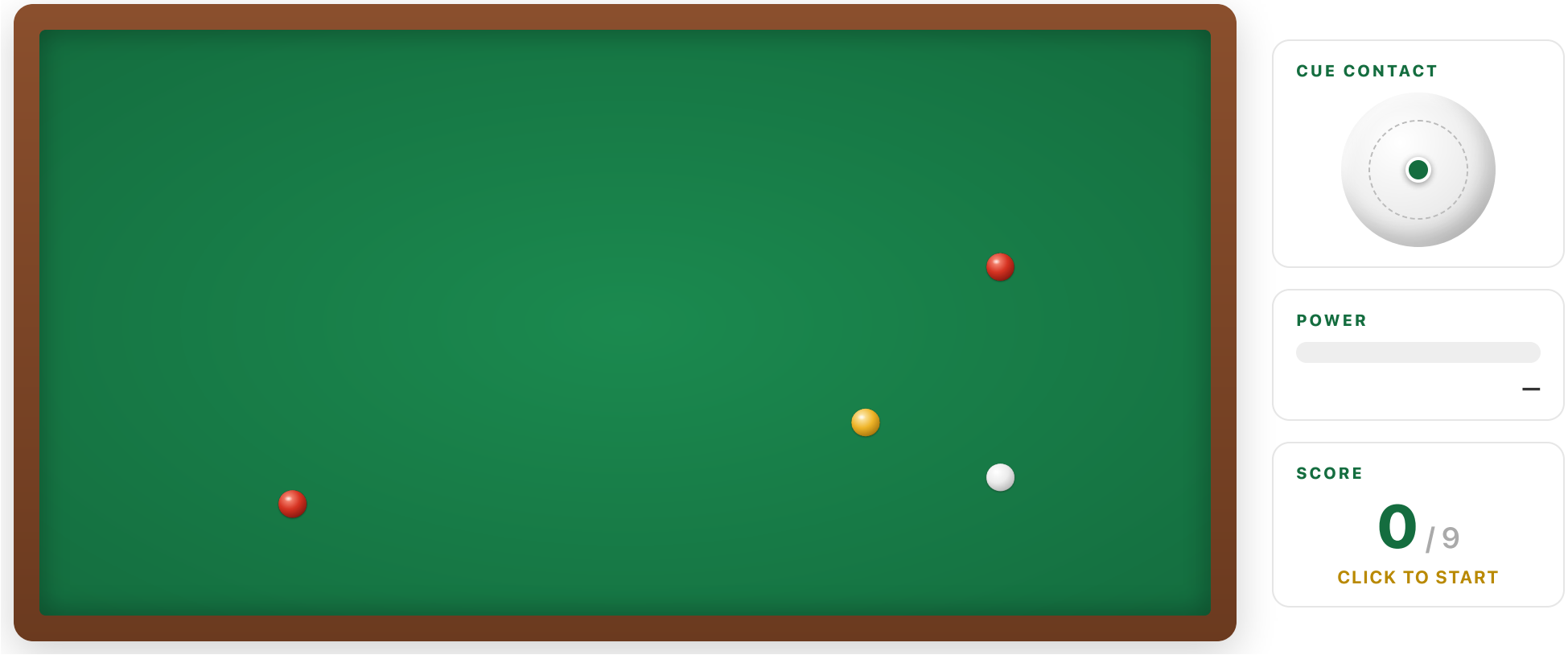
TABLE 6 · MAIN DOMAIN-KNOWLEDGE ABLATION · 1M STEPS

Variant	Mean ± std	Max	Foul/sh	Success/sh	Δ plain
plain	0.487 ± 0.097	4	20.5%	4.9%	—
constrain	2.570 ± 0.128	6	15.4%	25.7%	+2.083
constrain + extra	**6.460 ± 0.123**	**10**	**11.1%**	**64.6%**	**+5.973**

> 2.570 → **6.460** · per-shot success 25.7% → **64.6%**.

See it score ★

The constrain+extra policy chains shots — nine scores in a row.



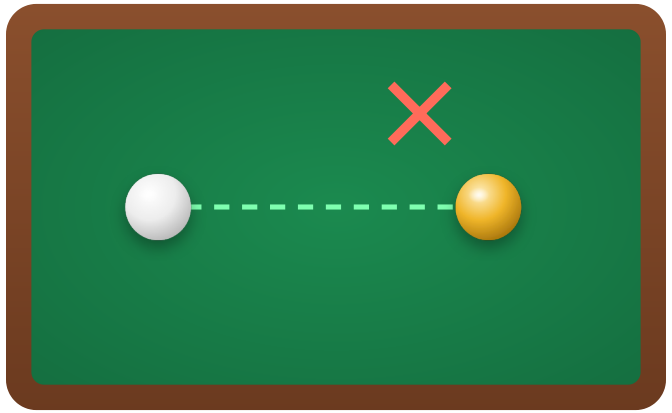
constrain+extra policy · 9-in-a-row — each shot previewed before it plays (click to start)

LEVER ③ · REWARD SHAPING

Could a richer reward push it further?

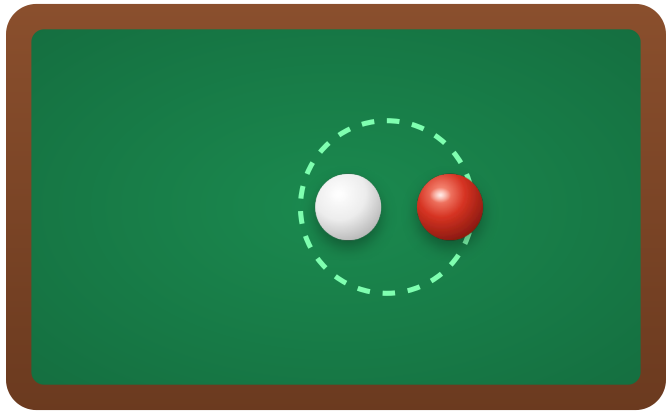
METHOD · REWARD SHAPING

Three dense reward terms



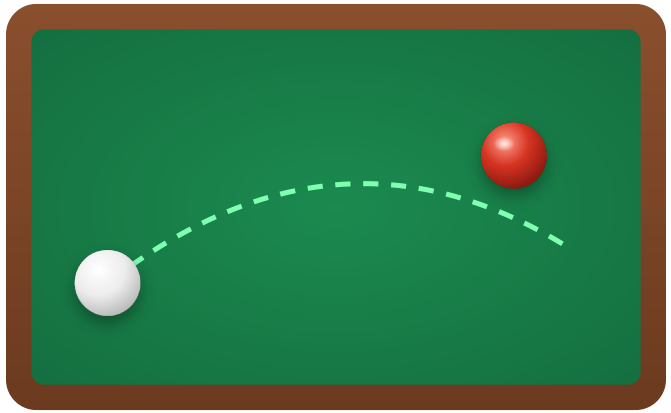
Foul penalty −1

cue ball hits the opponent ball



Gentle shot

leave cue ball near the 2nd red



Near-miss

bonus for almost hitting it

Dense geometric feedback beyond the binary carom score.

RESULT

Reward tinkering doesn't beat structure

TABLE 7 · FOUL PENALTY = SAFETY KNOB

Variant	Mean	Foul/sh	Success/sh
constrain + extra	6.460	11.1%	64.6%
+ foul −1	**5.553**	**6.0%**	**55.5%**

TABLE 8 · DENSE SHAPING = FLAT

Variant	Mean	Foul/sh	Δ base
base (foul1)	5.553	6.0%	—
+ gentle	5.213	3.3%	−0.34
+ near-miss	5.583	5.0%	+0.03
+ both	5.520	5.0%	−0.03

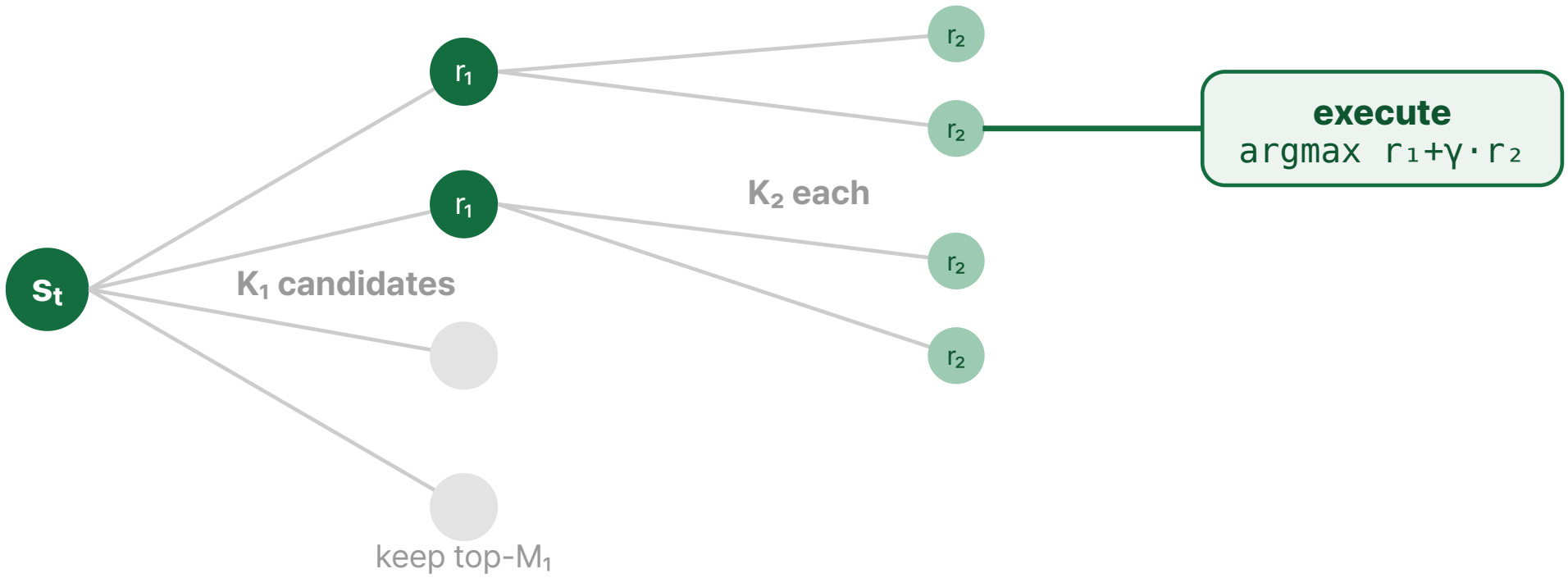
> Foul = score-for-safety trade; shaping stays within seed noise → **structure wins**.

LEVER ④ · INFERENCE-TIME SEARCH

The simulator runs in 5ms.
Let it verify.

METHOD · INFERENCE-TIME SEARCH

Propose → verify → look ahead



Policy **proposes**, the simulator **verifies** every candidate (5 ms each) — **no learned reward model**.

RESULT · INFERENCE-TIME SEARCH

Chains far beyond the horizon ★

9392

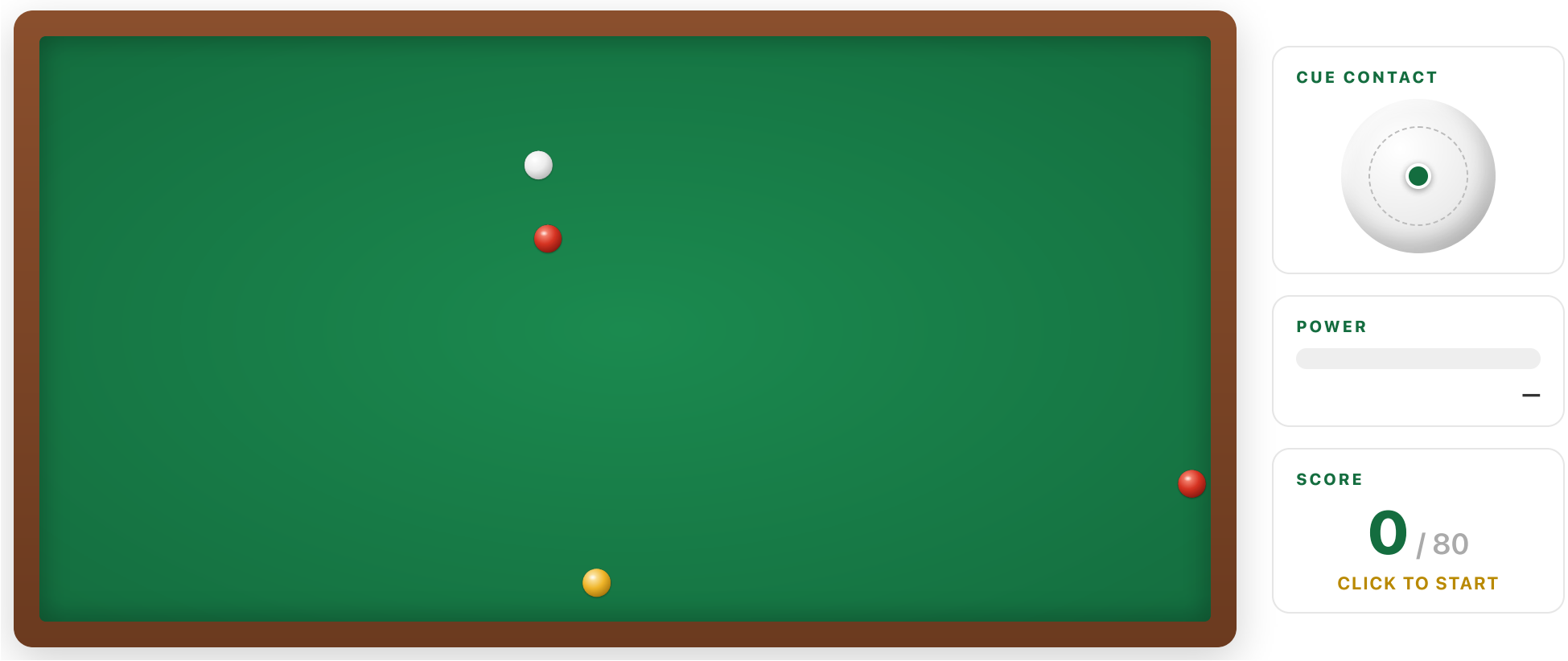
longest single verified chain
vs a 10-shot evaluation horizon

TABLE 9 · FIXED-ENGINE PROPOSER COMPARISON · H=2 VS PUCT

Proposer	P1 · h=2	P1 · PUCT	P3 · h=2 (max)	P3 · PUCT (max)
base	99.80	67.93	1383 (2634)	85.7 (404)
gentle	99.71	79.11	955 (1750)	32.4 (157)
near-miss	**99.83**	**81.85**	**2052 (9392)**	**77.4 (345)**
both	99.77	62.84	1194 (3272)	50.4 (191)

A chain that doesn't end ★

Depth-2 lookahead with the simulator as verifier — 80 consecutive scores, far past the 10-shot horizon.



depth-2 lookahead search · 80-in-a-row — each shot previewed before it plays (click to start)

A SMALL BITTER LESSON

Not solved by a cleverer reward. What scales is compute through a faithful forward model.

- **Learning** gives a prior.
- **Domain knowledge** makes it competent.
- **Simulator verification** extends it.

Thank you!