WHEN GOOD TEAMMATES ARE WEAKER ALONE: HANDOFF REGRET IN SKILL-COMPATIBLE AI

Anonymous authors
Paper under double-blind review

ABSTRACT

A strong agent that shares control with a weaker partner can leave behind states the partner cannot handle, even when each of its own actions is sound. We study this through *handoff regret*, the value lost when a fixed partner rather than the agent takes the next action. The central object is the handed-over loss: the solo losses of the partner's actions, summed over its observation classes with the team's weights. Team loss relative to the solo optimum is exactly the agent's own error plus this loss averaged under the partner's rule; the least any rule on the same observations can pay is the value of the information the partner lacks, and the rest is its decision error. Refinement, as in Blackwell's comparison of experiments, is exactly the order under which the information term never rises in any task, but behaviour identifies the split only up to an interval whose ends respond to different interventions, and a construction at one end shows that the team gain per unit of solo loss can be unbounded. The same accounting yields partner-aware corrections to an existing value oracle with explicit error terms. In tag-team chess with Maia-2 partners and strict alternation, the correction beats an engine given the same search budget head to head (57.2% of points), while the extra search alone gains nothing. In navigation, interventions on the agent's planning, the partner's rule and its information each move the term the decomposition predicts; on human games, estimated handoff regret predicts losing moves better than standard difficulty measures, and adds to them.

> *"I shall reconsider human knowledge by starting from the fact that we can know more than we can tell."* (explained in Appendix J)
> — Michael Polanyi, *The Tacit Dimension* (1966)

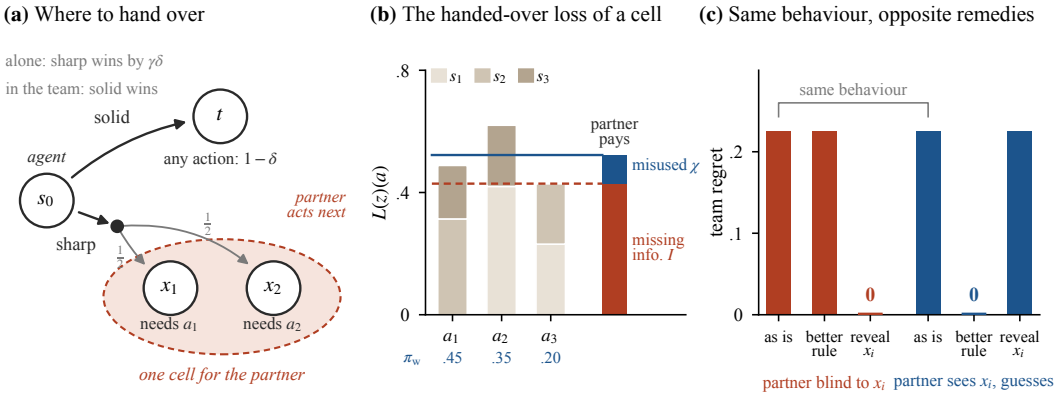


Figure 1: **What a strong agent hands to a weaker partner.** (a) Theorem 8 with $m = 2$: solid leads to a state where any action is fine; sharp leads to x_1, x_2, which need different actions but share a partner cell. (b) The handed-over loss L of a cell: reached states add their solo losses ℓ_s, weighted by the team's occupancy. The partner pays L averaged under its rule; no rule on the cell pays less than $\min_a L(z)(a)$, the missing information I, and the rest is misused information χ (Theorem 5). (c) Team regret of the sharp choice ($p = \frac{1}{2}$, $\gamma = 0.9$, $\delta = 0.01$) for two partners that guess alike at x_1, x_2, one blind to them and one not (revealing x_i comes with a rule that uses it): behaviour cannot tell them apart (Proposition 10), and each remedy helps only one.

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.

1 INTRODUCTION

A strong agent that shares control with a weaker partner can lose in a way it never loses alone. It plays a move that is correct, steers into a position only it can navigate, and then hands the position over. Kleinberg (2025) calls this "the AI sets you up to fail", and tag-team chess (Hamade et al., 2024) is a clean laboratory for it: a superhuman engine and a model of human play alternate moves for the same side without communicating, and the engine's best move is often one whose follow-up the partner cannot find. The same shape appears whenever control passes back to a person: generated code someone else must maintain, an automated manoeuvre that returns the wheel mid-corner. What is handed over is a state, and what matters is what it is worth in the hands of whoever receives it.

Hamade et al. (2024) showed that agents built to be compatible beat a stronger engine as partners while being weaker in solo play, and concluded that compatibility is distinct from strength. This is a puzzle, since folding a fixed partner into the dynamics gives an ordinary MDP whose optimum is a single policy. It also leaves a practical question open. When a handoff goes badly, should we change the agent, the partner's decisions, or what the partner can see?

We answer both by looking at what the partner is handed. *Handoff regret* $\rho(s)$ is the value lost when the partner, rather than the agent, acts at s. The solo values give each action at each state a loss, and a partner that sees only its observation classes faces these losses summed over each class with the team's weights: the *handed-over loss*. Team loss relative to the solo optimum is the agent's own error plus this loss averaged under the partner's rule (Theorem 5). The least any rule on the same observations can pay is an *information* term, the expected value of perfect information inside the classes, and the partner's excess over it is a *decision* term. The three ingredients of this pairing are the three remedies: planning moves the team, a better decision rule changes what the partner does, and more information refines the classes. Refinement is exactly the order under which the information term never rises in any task, yet behaviour fixes the split only up to an interval: a partner that cannot see a distinction and one that ignores it can act alike (Propositions 9 and 10). The construction of Figure 1a shows how the puzzle can arise: its loss is purely informational, and the team value gained per unit of solo value given up is unbounded (Theorem 8); some tasks force the trade-off, not all.

Contributions. (1) A decomposition. An exact split of team loss into own error, missing information and misused information around the handed-over loss (Theorem 5); an order and an identifiability result for the information term (Propositions 9 and 10); floors, a contrast bound, and an aliasing construction with an unbounded solo-for-team exchange rate (Corollary 6, Proposition 7, and Theorem 8). **(2) Partner-aware decoding with error accounting.** Corrections of an existing value oracle that separate oracle error, partner-model error and support truncation (Theorem 12 and Proposition 21), with two negative results that fix how depth and penalty scale may be read (Proposition 19) and a budgeted variant. With full support the correction is the Expector of Hamade et al. (2024), our closest baseline. **(3) Experiments that separate explanations.** In tag-team chess with strict alternation the correction beats an engine given the same search budget head to head ($57.2 \pm 2.3\%$), while extra search alone gains nothing and a partner-free penalty on sharp positions loses (Section 5.1). In navigation, interventions on planning, on the partner's rule and on its information each move the predicted term (Section 5.2). On human games, estimated handoff regret predicts losing moves better than nine standard difficulty features, and adds to them (Section 5.3).

2 RELATED WORK

Skill-compatible agents. Hamade et al. (2024) introduced tag-team and hand-and-brain chess and three compatible agents: a partner-guided tree search, *Expector*, a two-ply expectimax that combines an engine with a model of the junior, and a self-play fine-tune. Expector is our correction with $\lambda = 1$, a two-ply horizon and a renormalised partner support, and we use it as the closest baseline. What we add is an account of what such corrections buy (Theorem 5), what their scale and truncation do (Theorem 12 and Proposition 21), and experiments that separate partner-awareness from compute. Maia and Maia-2 model human move choice at a given rating (McIlroy-Young et al., 2020; Tang et al., 2024), and Zhang et al. (2025) extend human-aligned play to high ratings. In one-shot advice, Bansal et al. (2021) showed that making a model more accurate can lower team performance; handoff regret is the sequential analogue, where what is handed over is a state the partner must keep acting in.

Teaming with fixed or learning partners. Ad hoc teamwork (Stone et al., 2010), zero-shot coordination (Hu et al., 2020; 2021) and human-aware policies in Overcooked (Carroll et al., 2019) learn to play with populations of partners; we correct an existing oracle at decision time for one fixed partner. Learning to defer (Madras et al., 2018; Mozannar & Sontag, 2020; Wilder et al., 2020) decides which single predictions to route to a human; assistance games (Hadfield-Menell et al., 2016) and shared autonomy (Javdani et al., 2015) infer what the human wants, whereas our agent knows the objective and models only what the partner will do. Legibility (Dragan et al., 2013) makes an agent's intent inferable, and physical handovers between people are now a perception benchmark (Gavryushin et al., 2026); handoff regret asks that the state left behind be one the partner can act in.

Information, abstraction and world models. The information term is the classical expected value of perfect information (Raiffa & Schlaifer, 1961; Howard, 1966) evaluated on the partner's observation classes. Bounds on the value lost by planning in a state abstraction (Li et al., 2006; Abel et al., 2016) concern the agent's own abstraction, paid at every step; here the abstraction is the partner's and it is paid only at handoffs, weighted by where the team goes. Vafa et al. (2024) evaluate the world model implicit in a sequence model through Myhill–Nerode distinctions between histories. Our behavioural partition (Proposition 10) is a one-step analogue of such distinctions, and in our experiments the partition is known by construction, not recovered.

3 Handoff regret and where it comes from

3.1 Team control with a fixed partner

Fix a discounted MDP $M = (\mathcal{S}, \mathcal{A}, P, r, \gamma)$ with $\gamma \in (0,1)$, rewards in $[0, R]$ and $H = R/(1-\gamma)$, a partner policy π_{w} and a control rate $p \in (0,1)$. At each step the agent acts with probability p and the partner acts otherwise, and they do not communicate. Write $q_V(s,a) = r(s,a) + \gamma\, \mathbb{E}_{s' \sim P(\cdot|s,a)} V(s')$. If the agent plays π, the team behaves like the mixture $\pi_p = p\,\pi + (1-p)\,\pi_{\mathrm{w}}$, so the team process is an ordinary MDP in which the agent controls a p-fraction of the policy. We write V_p^{π} for its value, V_p^* for the optimal team value, V_1^* for the solo optimum, $q = q_{V_1^*}$, and d_p^{π} for the normalised discounted occupancy of the team process from an initial distribution μ_0. The partner is fixed and does not adapt, the schedule is exogenous, and the agent may query a partner model. Strict alternation, used in chess, replaces the coin by a role phase carried in the state; the statements below then hold with p replaced by a state-dependent control indicator (Appendix C). Proofs are in Appendix B.

Definition 1 (Handoff regret)**.** For a value function V, the handoff regret at s is the value lost when the partner, rather than the agent, picks the next action:

$$\rho_V(s) \;=\; \max_a q_V(s,a) \;-\; \mathbb{E}_{b \sim \pi_{\mathrm{w}}(\cdot|s)}\, q_V(s,b) \;\geq\; 0,$$

and we write $\rho = \rho_{V_1^*}$.

$\rho_V(s)$ combines how often the partner errs with what each error costs, and it is a property of the state and the partner, not of the agent. Two identities organise the rest of the section.

Lemma 2 (Team backup)**.** For $T_p V = p \max_a q_V + (1-p)\mathbb{E}_{\pi_{\mathrm{w}}} q_V$, we have $T_p V = T_1 V - (1-p)\rho_V$ pointwise, and T_p is a monotone γ-contraction.

Proposition 3 (Performance difference)**.** For every agent policy π, the team's loss splits into the agent's own shortfall $g_\pi(s) = \max_a q(s,a) - \mathbb{E}_{a \sim \pi(\cdot|s)} q(s,a)$ where it acts and the handoff regret where the partner acts: with $d = d_p^{\pi}$,

$$\mathbb{E}_{\mu_0}\left[V_1^* - V_p^{\pi}\right] = \frac{p}{1-\gamma}\, \mathbb{E}_{s \sim d}\, g_\pi(s) + \frac{1-p}{1-\gamma}\, \mathbb{E}_{s \sim d}\, \rho(s).$$

The agent can therefore trade: it can accept some error of its own ($g_\pi > 0$) for less handoff regret, either at the next handoff or by steering where the team goes. Proposition 3 does not say why ρ is large where it is large, and so it cannot say who should do what about it.

3.2 THE HANDED-OVER LOSS

A partner acts on what it observes. We represent this by a partition of the state space.

Definition 4 (Information partition)**.** A map $\varphi : \mathcal{S} \to \mathcal{Z}$ is an information partition for π_{w} if $\pi_{\mathrm{w}}(\cdot \mid s)$ depends on s only through the cell $z = \varphi(s)$. We then write $\pi_{\mathrm{w}}(\cdot \mid z)$.

Write $\ell_s(a) = \max_b q(s,b) - q(s,a) \geq 0$ for the solo loss of action a at s, so that $\rho(s) = \mathbb{E}_{b\sim\pi_{\mathrm{w}}(\cdot|s)}\ell_s(b)$. The partner sees only the cell of the state it receives, so what reaches its decision is the loss summed over the cell with the team's own weights. With $d = d_p^\pi$, we call

$$L(z) \;=\; (1-p) \sum_{s\in\varphi^{-1}(z)} d(s)\,\ell_s \;\in\; \mathbb{R}^{\mathcal{A}}_{\geq 0} \tag{1}$$

the *handed-over loss* of cell z (Figure 1b). By Proposition 3 the partner's share of the team regret is L averaged under the partner's rule, the pairing $\frac{1}{1-\gamma}\sum_z \langle \pi_{\mathrm{w}}(\cdot \mid z), L(z)\rangle$. At this d no rule on the same cells can pay less than the smallest coordinate of each $L(z)$, and this splits the pairing in two. For a cell z and a distribution ν on it, let $\bar\ell_\nu = \mathbb{E}_{s\sim\nu}\ell_s$ and

$$I(z,\nu) = \min_a \bar\ell_\nu(a), \qquad \chi(z,\nu) = \mathbb{E}_{b\sim\pi_{\mathrm{w}}(\cdot|z)}\bar\ell_\nu(b) - \min_a \bar\ell_\nu(a).$$

$I = \mathbb{E}_\nu \max_a q - \max_a \mathbb{E}_\nu q$ is the expected value of perfect information inside the cell (Howard, 1966): what one action rule for the whole cell loses against seeing the state. χ is what the partner loses, given its information, against the best such rule. Both are non-negative and $\mathbb{E}_\nu\rho = I + \chi$.

Theorem 5 (Three-term decomposition)**.** Let φ be an information partition for π_{w}. For every agent policy π, with $d = d_p^\pi$ and L as in (1),

$$\mathbb{E}_{\mu_0}\left[V_1^* - V_p^\pi\right] = \frac{1}{1-\gamma}\Big[\underbrace{p\,\mathbb{E}_d\, g_\pi}_{\text{own error}} + \underbrace{\sum_z \min_a L(z)(a)}_{\text{missing information}} + \underbrace{\sum_z \big(\langle \pi_{\mathrm{w}}(\cdot \mid z), L(z)\rangle - \min_a L(z)(a)\big)}_{\text{misused information}}\Big].$$

With $d_{\mathcal{Z}}(z) = d(\varphi^{-1}(z))$ and $\nu_z = d(\cdot \mid z)$, the last two terms equal $\frac{1-p}{1-\gamma}\sum_z d_{\mathcal{Z}}(z)\, I(z,\nu_z)$ and $\frac{1-p}{1-\gamma}\sum_z d_{\mathcal{Z}}(z)\,\chi(z,\nu_z)$.

The information term is the least pairing with L that any rule on the partner's cells can achieve, and the decision term is the partner's excess over it. The three remedies act on the three ingredients of the pairing: planning moves L through d, a better decision rule changes $\pi_{\mathrm{w}}(\cdot \mid z)$ within fixed cells, and more information refines the partition that sums L. For a fixed d refinement can only lower the information term, since the minimum of a sum is at least the sum of the minima (Proposition 9). Section 5.2 tests the three predictions separately.

Three readings of the decomposition are wrong. It is not pointwise: if $\ell_{s_1} = (0,1)$ and $\ell_{s_2} = (1,0)$ share a cell with equal weights, then $I = \frac{1}{2}$, yet a partner that always plays the first action has $\rho(s_1) = 0$. A cell is free to hand over only if both terms vanish: the states the team reaches in it share an optimal action and the partner plays it. And the weights belong to the team process, so the agent moves the information term too, by changing which states of a cell are reached. A floor that no agent can go below therefore needs assumptions.

Corollary 6 (Floors)**.** For every agent policy, $\mathbb{E}_{\mu_0}[V_1^* - V_p^\pi] \geq \frac{1-p}{1-\gamma}\sum_z d_{\mathcal{Z}}(z)\, I(z,\nu_z)$. If there are cells $\mathcal{Z}_0$ and $\alpha, e > 0$ such that every agent policy puts discounted mass at least α on $\mathcal{Z}_0$ and induces $I(z,\nu_z) \geq e$ on each $z \in \mathcal{Z}_0$, then every agent policy has team regret at least $(1-p)\,\alpha\, e/(1-\gamma)$.

How large can the information term be? It is controlled by how much the states of a cell disagree about what to do, not by how different their values are.

Proposition 7 (Contrast bound)**.** With $\operatorname{span}(v) = \max_a v_a - \min_a v_a$,

$$I(z,\nu) \;\leq\; \tfrac{1}{2}\,\mathbb{E}_{s,s'\sim\nu\otimes\nu}\,\operatorname{span}\big(q(s,\cdot) - q(s',\cdot)\big) \;\leq\; \mathbb{E}_{s,s'\sim\nu\otimes\nu}\big\|q(s,\cdot) - q(s',\cdot)\big\|_\infty.$$

The middle term is invariant under $q(s,\cdot) \mapsto q(s,\cdot) + c(s)\mathbf{1}$, as I is, and both inequalities hold with equality for $q(s_1,\cdot) = (1,0)$, $q(s_2,\cdot) = (0,1)$ and ν uniform.

3.3 When the better teammate is the weaker player

Theorem 8 (Aliasing trade-off)**.** Let $m \geq 2$, $p, \gamma \in (0,1)$ and $0 < \delta < (1-p)(1-\frac{1}{m})$. There is an MDP with $m+3$ states, m actions and rewards in $[0,1]$, a partner with an information partition, and two agent policies π^{sharp} and π^{solid} that differ only at the root s_0, such that: (i) π^{sharp} is solo-optimal and $V_1^{\text{sharp}}(s_0) - V_1^{\text{solid}}(s_0) = \gamma\delta$; (ii) the partner's only non-singleton cell has $I = 1 - \frac{1}{m}$ and $\chi = 0$ under every agent policy; (iii) $V_p^{\text{solid}}(s_0) - V_p^{\text{sharp}}(s_0) = p\gamma\big[(1-p)(1-\frac{1}{m}) - \delta\big] > 0$, and π^{solid} is team-optimal. The team value gained per unit of solo value given up is $p(1-p)(1-\frac{1}{m})/\delta - p$, which is unbounded as $\delta \to 0$.

At the root the agent chooses between a safe state worth $1-\delta$ and a chance move into one of m states, each paying 1 for its own action and 0 otherwise (Figure 1a). The partner cannot tell these states apart, so it scores $1/m$ there whatever it does ($\chi = 0$). Alone, the sharp branch is worth $\gamma\delta$ more; in the team the partner acts next with probability $1-p$, the information it lacks is worth $1-\frac{1}{m}$, and the solid branch wins once $\delta < (1-p)(1-\frac{1}{m})$. A partner that sees the state and still guesses gives the same numbers with $I = 0$ and $\chi = 1 - \frac{1}{m}$ (Appendix C), and a better decision rule then removes the loss without new information. Both are existence results: they show which term can drive a trade-off, not that every task has one.

3.4 What information is worth, and what behaviour reveals

The construction and its twin raise two questions: is the information term the right measure of what a partner knows, and can missing information be told from misused information by watching the partner? Write $I_\varphi(d,q) = \sum_z d_{\mathcal{Z}}(z)\, I(z,\nu_z)$ for the information term of a partition φ under a state distribution d and action values q.

Proposition 9 (Information order)**.** Let $|\mathcal{A}| \geq 2$ and let φ, φ' be partitions of $\mathcal{S}$. Then φ' refines φ if and only if $I_{\varphi'}(d,q) \leq I_\varphi(d,q)$ for every distribution d on $\mathcal{S}$ and every $q \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$. For a refinement, the difference is the value of learning the finer cell inside the coarser one.

So refinement is exactly the order under which the information term never rises, whatever the task; this is the deterministic case of Blackwell's comparison of experiments (Blackwell, 1953).

Proposition 10 (What behaviour identifies)**.** Let β be the behavioural partition of π_{w}: s and s' share a cell if and only if $\pi_{\text{w}}(\cdot \mid s) = \pi_{\text{w}}(\cdot \mid s')$. A partition is an information partition for π_{w} if and only if it refines β. Hence, for every agent policy and $d = d_p^\pi$, every information partition φ for π_{w} has $0 \leq I_\varphi(d,q) \leq I_\beta(d,q)$, with the left end attained when the partner sees the state and the right end by β; the decision term moves the other way, as the two sum to $\mathbb{E}_d\rho$.

Behaviour alone thus fixes the split only up to an interval. The two partners of Figure 1c are its ends: they act identically, the first attains I_β and the second sees the state, but they respond to different interventions; Section 5.2, with a partition known by design, applies both.

4 Partner-aware decoding from an existing oracle

The agent's lever on the handed-over loss is where the team goes, and Lemma 2 says how to pull it: the team operator is the solo operator minus a handoff term. A deployed system usually has an approximate solo value (a search engine or a value head) and no team value, so we correct it rather than retrain.

Definition 11 ($\text{HAD}_{\lambda,d}$)**.** Let $\widehat{V}$ be an oracle with $\|\widehat{V} - V_1^*\|_\infty \leq \eta$ and $\widehat{\pi}_{\text{w}}$ a partner model; write $\widehat{\rho}_V = \max_a q_V - \mathbb{E}_{\widehat{\pi}_{\text{w}}} q_V$ and let Π clip to $[0,H]$. Set $V^{(0)} = \Pi\widehat{V}$ and, for $i = 1,\dots,d$,

$$V^{(i)} = \Pi\big(T_1 V^{(i-1)} - \lambda(1-p)\,\widehat{\rho}_{V^{(i-1)}}\big).$$

The agent plays $\pi_{\lambda,d}(s) \in \arg\max_a q_{V^{(d)}}(s,a)$.

Here d counts corrections: $d = 0$ is the uncorrected system, and with an exact oracle $d = 1$ is the explicit penalty $\pi_{\lambda,1}(s) \in \arg\max_a \{q(s,a) - \lambda(1-p)\,\gamma\,\mathbb{E}_{s'\sim P(\cdot|s,a)}\widehat{\rho}(s')\}$, which reads: among

the actions the solo agent values, prefer those after which the partner is less likely to lose value. For $0 \le \lambda(1-p) \le 1$ the corrected operator is the team operator of a partner that acts with probability $\lambda(1-p)$, hence a monotone γ-contraction, and $\lambda = 1$ is the true team operator. Larger λ can be tried but loses these properties. With a full-support partner model and a two-step horizon, HAD$_1$ is the expectimax *Expector* of Hamade et al. (2024); the analysis adds error accounting and a treatment of truncated support.

Theorem 12 (Error accounting). Let $\lambda = 1$, $\sup_s \mathrm{TV}\big(\widehat{\pi}_{\mathrm{w}}(\cdot \mid s), \pi_{\mathrm{w}}(\cdot \mid s)\big) \le \varepsilon$, $\Delta = \sup_{s,\, V \in [0,H]^S} \operatorname{span} q_V(s,\cdot) \le H$ and $A = \|V_1^* - V_p^*\|_\infty$. Then

$$\|V^{(d)} - V_p^*\|_\infty \;\le\; \gamma^d\,(A+\eta) + (1-p)\,\varepsilon\,\Delta\,\frac{1-\gamma^d}{1-\gamma}, \qquad V_p^* - V_p^{\pi_{1,d}} \;\le\; \frac{2p\gamma}{1-\gamma}\,\|V^{(d)} - V_p^*\|_\infty.$$

The value bound separates three errors: the gap between the solo and team optima and the oracle error, both forgotten geometrically, and the partner-model error, which accumulates. The factor p in the policy bound appears because the partner's term cancels when two agent actions are compared.

The bound does not make deeper corrections better, and it does not fix how the best penalty moves as the partner degrades: Proposition 19 (Appendix B) gives a five-state MDP in which one correction hurts and a second undoes the harm, and an exact affine law under which the best scalar penalty can rise or fall. Both are things to measure (Appendix G).

4.1 The chess instance

In tag-team chess (Hamade et al., 2024) each side is a team of a senior (the engine) and a junior (a model of human play) who do not communicate; in our primary setting they move in strict alternation, whereas Hamade et al. (2024) flip a coin. At a senior decision the team's next handoff is two plies away, after the opponent's reply. Algorithm 1 (Appendix D) scores each of the engine's top-K moves a by $\bar{v}(a) - \lambda\, w_{\mathrm{self}}\, \bar{\rho}(a)$, where $\bar{v}$ and $\bar{\rho}$ average the value and the own-partner handoff regret over the opponent's reply. Three details matter.

Role phases. Under strict alternation w_{self} and w_{opp} are indicators read off the role phase, not probabilities: after its senior move a team's junior always moves next ($w_{\mathrm{self}} = 1$), and White's senior always meets the opposing senior and Black's the opposing junior, so every opening is played with both colours.

Truncation. The reply distribution is normalised: the opponent's senior with weight $1 - w_{\mathrm{opp}}$ and the renormalised top-m moves of the opponent's junior with weight w_{opp}. At the handoff state the own-partner risk is $\widehat{\rho}_m = \sum_{b \in B_m} \widehat{\pi}_{\mathrm{w}}(b)\, \ell(b)$ over the partner model's top-m moves with their raw probabilities, $\ell(b) \ge 0$ being the oracle's loss of b. Dropping the tail makes this a lower bound on the model's handoff regret, $\widehat{\rho}_m \le \rho^{\widehat{\pi}_{\mathrm{w}}} \le \widehat{\rho}_m + (1 - w_m)\max_b \ell(b)$ with w_m the retained mass (Proposition 21, Appendix C). It is not a bound on what a real player loses, which also depends on the model and the oracle. Expector renormalises instead, using $\widehat{\rho}_m / w_m$.

Oracle. Values are Stockfish's expected scores from its WDL model. Searches of different depths are not exactly Bellman-consistent (mean inconsistency about 0.01 expected score, Appendix D), so the chess rule is a depth-limited instance of the theory, not a numerical check of Theorem 12.

A budgeted variant. HAD-budget replaces the scale λ by a per-decision solo budget B: among moves within B of the best $\bar{v}$ it takes the one with the smallest $\bar{\rho}$, and leaves the best move only if this lowers $w_{\mathrm{self}}\bar{\rho}$ by more than ζ. With exact values each decision costs at most B alone; in chess the budget is empirical.

5 Experiments

The three settings exercise the three ingredients of the pairing in Theorem 5. In tag-team chess the partner sees the whole board, so its loss is essentially all decision error and the agent's lever is where the team goes: does steering by the handoff term buy what extra search does not? In navigation, where we set the partner's information and can replace its rule, does the decomposition say which intervention helps? On human games, estimating the partner's expected loss at each position, does

Senior rule (focal team)	win share (%)	nodes	s	overrides	partner loss
Raw engine R (reference)	50	18k	0.04	–	–
Matched compute (R with HAD-4's nodes)	50.7 ± 1.9	448k	0.87	–	+1%
OA (opponent-aware, $\lambda = 0$)	50.1 ± 2.3	191k	0.44	–	−9%
HAD-4 (primary: $\lambda = 1$, $m = 4$)	56.9 ± 2.2	366k	1.15	34%	−18%
without opponent mixture	51.7 ± 2.1	308k	0.66	36%	−9%
$\lambda = 0.5$	57.1 ± 2.2	369k	0.85	22%	−16%
$m = 8$	58.9 ± 2.3	528k	1.42	39%	−22%
adaptive support (mass ≥ 0.9)	62.5 ± 2.1	450k	1.11	37%	−24%
Expector ($m = 4$, renormalised)	60.1 ± 2.2	372k	1.16	40%	−23%
HAD-budget, $B = 0.005$ / 0.01 / 0.02	51.6 / 51.6 / 53.9	366k	0.87	8 / 13 / 22%	−11%
Sharpness penalty, no partner model	42.8 ± 2.3	475k	0.99	35%	−9%
head to head, HAD-*4 against*					
matched compute	57.2 ± 2.3				
OA	55.9 ± 2.3				
Expector	50.7 ± 2.0				

Table 1: Tag-team chess, Maia-2 juniors at 1500, strict alternation, 800 held-out opening pairs per row. Win share of the focal team against the reference R (raw engine plus the same junior), with 95% intervals; 50% means no difference. *Nodes* and *s*: engine nodes and seconds per senior decision. *Overrides*: decisions on which the handoff term changes the move chosen by the same rule with $\lambda = 0$. *Partner loss*: the focal junior's realised loss per move relative to the reference junior's in the same games.

it add to ordinary measures of difficulty? Appendix G checks every identity and bound on random MDPs.

5.1 TAG-TEAM CHESS

Protocol. Each team is a senior (Stockfish 17.1, depth 12 at the root and 10 for the evaluations inside Algorithm 1) and a junior (Maia-2 (Tang et al., 2024), an Elo-conditioned model of human move choice, sampled). We use a model of human play rather than a weakened engine because an engine with its search turned down loses like a weak player without playing like one (McIlroy-Young et al., 2020). Teams move in strict alternation, both starting with their senior. The reference team R pairs the raw engine with the same junior. Games start from 1,000 balanced eight-ply openings drawn for this study with a fresh seed and disjoint from all openings used during development; each opening is played once with each colour under common random numbers, and the opening pair is the unit of analysis, with 95% intervals (± 1.96 standard errors over pairs). The primary setting, the primary rule (HAD-4: $\lambda = 1$, $K = 6$ candidates, $m = 4$ partner moves, two opponent-junior replies), the comparisons and the success criterion were fixed before any game on these openings (Appendix D).

Partner-awareness, not compute. Table 1 gives the result. HAD-4 scores $56.9 \pm 2.2\%$ against R. Giving the raw engine HAD-4's node budget, 24 times its usual search, gains nothing as a teammate ($50.7 \pm 1.9\%$), and HAD-4 beats this matched-compute engine head to head ($57.2 \pm 2.3\%$), which was our pre-registered criterion. The matched engine makes fewer mistakes itself but its junior loses as much per move as before; under HAD-4 the junior loses 0.107 instead of 0.131 expected score per move, while the senior's own loss per move, measured the same way, rises from 0.008 to 0.023. The two ingredients of Algorithm 1 are both needed. The opponent model alone does nothing ($50.1 \pm 2.3\%$), and the own-partner term without it is weak ($51.7 \pm 2.1\%$): the handoff regret has to be evaluated at the position the partner will actually receive, which half of the time the opponent's junior has chosen. A control that penalises sharp positions without any partner model, using the spread of the engine's own top five moves, loses ($42.8 \pm 2.3\%$). Avoiding difficulty in general costs more than it saves; what helps is avoiding the positions in which this particular partner goes wrong (Figure 2 shows three such decisions).

Relation to Expector. Performance rises as the risk estimate covers more of the partner model. HAD-4 keeps on average 0.83 of Maia's probability mass with raw probabilities, which by Proposition 21 under-estimates the model's handoff regret; widening the support to $m = 8$ gives 58.9%, renormalising the same four moves as Expector does gives 60.1%, and growing the support until

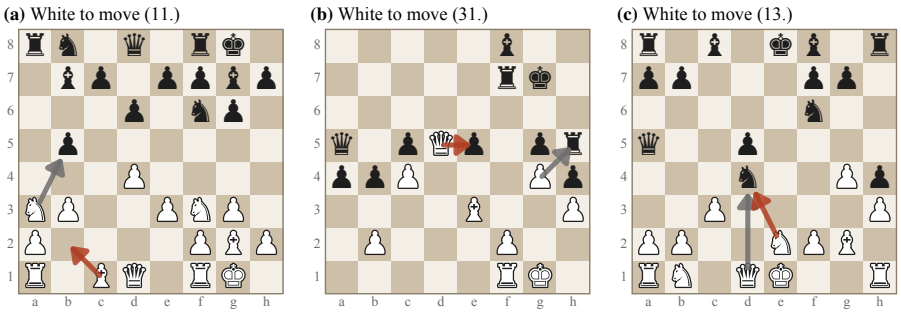


Figure 2: The move chosen alone ($\lambda = 0$, grey) and the team move (HAD-4, red) at three decisions of White's senior, with a Maia-2 partner at 1500. Table 3 in Appendix D gives the values and the selection rule.

90% of the mass is covered gives $62.5 \pm 2.1\%$ at about the same time per decision. Head to head, HAD-4 and Expector are indistinguishable ($50.7 \pm 2.0\%$), so we do not claim a better decision rule than Expector's, only an account of why both work and what scale and truncation do.

Budget and scale. At $B = 0.01$ the budgeted rule overrides the $\lambda = 0$ choice in 13% of decisions and pays 0.0006 expected score per decision by its own estimate, but gains little (51.6%; $53.9 \pm 2.4\%$ at $B = 0.02$). A smaller scale is more economical: $\lambda = 0.5$ matches $\lambda = 1$ (57.1%) while paying 0.0041 per decision against 0.0138. The gains hold with Maia-2 at 1100 and 1900 ($54.5 \pm 3.1\%$ and $59.8 \pm 3.1\%$) and under a stricter adjudication rule ($59.4 \pm 3.2\%$). Under the coin-flip schedule of Hamade et al. (2024), where a handoff follows a senior move only half the time, HAD-4 gains nothing ($51.2 \pm 2.9\%$) while Expector still gains ($56.2 \pm 2.9\%$), consistent with the under-estimate mattering more when the penalty is halved (Table 2 in Appendix D).

Alone. We then let each team-tuned senior play every move itself against the raw engine, keeping its team assumptions (Figure 4 in Appendix D). The agent that scores 56.9% as a teammate scores $5.9 \pm 1.2\%$ alone; the model of an opposing junior costs most (9.7% for OA), the own-partner term less (18.4%). Told that it now plays alone, it recovers to $35.7 \pm 1.9\%$, the rest being its narrower search (six candidates, shallower replies). So in chess the good teammate is weaker alone mostly because it keeps planning for partners who are no longer there: an avoidable deployment error, not the price of compatibility of Theorem 8.

5.2 NAVIGATION: WHICH INTERVENTION HELPS

Setting. A 12×12 grid with one-way streets; the reward is 1 on reaching the goal. The partner is a small transformer trained to imitate shortest paths from what it observes: its displacement to the goal and its own last $k \in \{0, 1, 2, 3\}$ moves. It cannot see which streets at its current corner are one-way. Its information partition is therefore known exactly, and the partitions are nested in k. The agent sees everything, and the two alternate at random ($p = \frac{1}{2}$). We solve the team process exactly on the common state (goal, position, last three moves) and compute occupancies by linear solves, so every quantity below is exact, and Theorem 5 holds in every run with residual below $4e{-}16$. Training, validation and evaluation goals are disjoint; we use 3 maps with 5 partner seeds each and report means over the 15 runs (intervals in Appendix E).

More information, used less well. On one fixed reference distribution for all k, the information term falls by 38% from $k = 0$ to $k = 3$, as nested partitions require, while the decision term of the learned partner rises by 20% (Figure 3a). Shortest-path accuracy barely moves (75.4% to 77.2%) and misses both.

Each intervention moves its own term. Figure 3b changes each ingredient of the pairing in turn and decomposes the result with Theorem 5. Replacing the partner's rule by the best rule on the same cells (a one-step Bayes rule fitted on training goals) lowers the decision term, leaves the information term unchanged, and closes 14% of the team regret at $k = 0$ and 8% at $k = 3$. Revealing, on top of this rule, the exit pattern of the current corner in 10% of the cells refines the partition and lowers the information term, part of the gain returning as decision error of the refitted rule; choosing the cells by their information term closes 26% in all against 20% for random cells at $k = 0$ (at $k = 3$, 25%

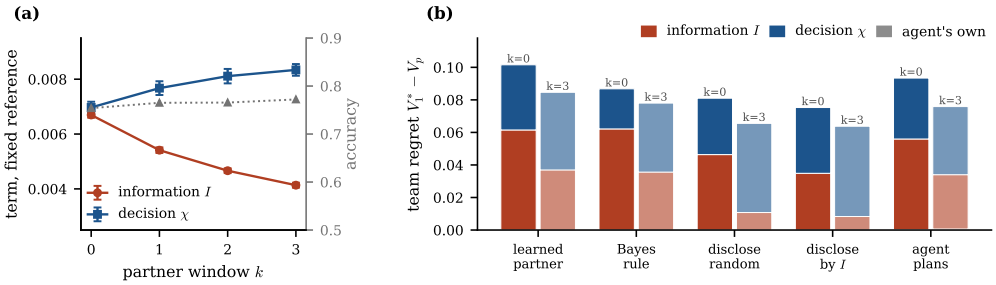


Figure 3: Navigation, 15 runs (3 maps × 5 partner seeds). (a) On one fixed reference distribution, a longer partner window k lowers the information term and raises the decision term; shortest-path accuracy barely changes. (b) Team regret under each intervention, split as in Theorem 5, for $k = 0$ (dark) and $k = 3$ (light). The Bayes rule moves χ, disclosure moves I, and planning by the agent moves both a little; the agent's own term is below 0.001 throughout.

against 23%, within noise), and choosing them by the entropy of the partner's policy does as well (26%). Letting the agent plan with the partner, up to the exact team optimum, moves the occupancy and closes only 8% at $k = 0$ and 10% at $k = 3$, at a negligible own cost. Here most of the loss sits on the partner's side, and the decomposition says which side and which term.

5.3 Human games: does estimated handoff regret add information?

Setting. We sample positions (every third ply from ply 10 to 70) from 1,050 rated Lichess games in three rating bands, with both players in the band and a rapid time control (Lichess, 2025); 17,033 positions in total. For each position, before the player's move, one engine search over all legal moves gives each move's value, and from it the realised loss of the move actually played and nine standard measures of difficulty. The estimate $\widehat{\rho}_6$ weights the engine's move losses by Maia-2's probabilities for the player's rating, over its six most likely moves. All fitting uses one half of the games and all numbers are on the other half (8,605 positions from 537 games), with intervals from a bootstrap over games. The games are from July 2014, inside the period on which Maia-2 was trained; this is a test of what the estimate adds over difficulty features, not of the partner model on unseen data.

Result. Table 6 in Appendix F gives the answer. The estimate alone separates losing moves better (AUROC 0.896) than a logistic model fitted on all nine difficulty features (0.871), and adding it to that model raises the AUROC by 0.018 to 0.025 and the AUPRC by 0.025 to 0.058 (95% intervals). The same engine losses weighted uniformly instead of by the partner model do much worse (0.741), so the gain comes from knowing which moves this kind of player tends to choose. As an expected-loss prediction the estimate is calibrated on average (regression slope of realised loss on $\widehat{\rho}_6$: 0.99; it under-predicts for weaker players and over-predicts at 1800–2000, slopes 1.12, 1.07 and 0.76). The gain holds in every rating band, game phase and value range (Appendix F), and it survives re-scoring a random fifth of the positions with a deeper, separate search, which makes an artefact of the shared oracle unlikely (0.865 to 0.882, gain between 0.009 and 0.026). This is observational evidence about real players' losses, not evidence that people would play better next to a handoff-aware agent.

6 Conclusion

A team's loss relative to the solo optimum has three sources that call for different remedies, and which one dominates is measurable, though telling missing from misused information takes the partner's information or an intervention, not just its behaviour. In tag-team chess the agent's side matters: choosing moves by what the partner will do with them beats spending the same compute on search. In navigation most of the loss sat on the partner's side: a better rule with disclosure recovered far more than planning by the agent. Nothing in Sections 3 and 4 is specific to games: what matters is how much of an action's value survives the handoff, and whether the loss there is missing or misused information.

AI USE STATEMENT

The original research ideas and initial manuscript draft were developed by the authors. AI tools were used to assist with checking mathematical details and the correctness of arguments, filling in routine proof steps, coding assistance, and language polishing. The authors take full responsibility for the content and correctness of the final manuscript.

ETHICS STATEMENT

No human subjects were involved. The human games are public records from the Lichess open database, from which we use only moves, ratings and time controls; no player identifiers enter the analysis. The two-team algebra of Appendix C contains a term that rewards leading an opposing partner into mistakes; we report it because the calculus produces it, and our experiments use it only through the opponent's reply distribution. The solo results of Section 5.1 are a concrete warning that an agent tuned for a team should not be deployed alone without resetting its model of who acts next.

REPRODUCIBILITY STATEMENT

Every number, table and figure is regenerated from raw records by the released scripts. The records include per-game results and per-decision cost logs for all chess runs, the held-out opening book and its generator, the pre-registration note written before the held-out games, the exact navigation solves with their seeds and goal splits, the per-position human-game features, and the numerical checks of every theoretical statement. Proofs are in Appendices B and C; settings, statistics and compute are in Appendices D to F and H. The engine, the partner model and the human games are public (Stockfish 17.1, Maia-2, the Lichess open database).

REFERENCES

David Abel, D. Ellis Hershkowitz, and Michael L. Littman. Near optimal behavior via approximate state abstraction. In *International Conference on Machine Learning (ICML)*, volume 48 of *PMLR*, pp. 2915–2923, 2016.

Ashton Anderson, Jon Kleinberg, and Sendhil Mullainathan. Assessing human error against a benchmark of perfection. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 705–714, 2016.

Gagan Bansal, Besmira Nushi, Ece Kamar, Walter S. Lasecki, Daniel S. Weld, and Eric Horvitz. Beyond accuracy: The role of mental models in human-AI team performance. In *AAAI Conference on Human Computation and Crowdsourcing (HCOMP)*, volume 7, pp. 2–11, 2019.

Gagan Bansal, Besmira Nushi, Ece Kamar, Eric Horvitz, and Daniel S. Weld. Is the most accurate AI the best teammate? optimizing AI for teamwork. In *AAAI Conference on Artificial Intelligence*, volume 35, pp. 11405–11414, 2021.

Nolan Bard, Jakob N. Foerster, Sarath Chandar, Neil Burch, Marc Lanctot, H. Francis Song, Emilio Parisotto, Vincent Dumoulin, Subhodeep Moitra, Edward Hughes, Iain Dunning, Shibl Mourad, Hugo Larochelle, Marc G. Bellemare, and Michael Bowling. The Hanabi challenge: A new frontier for AI research. *Artificial Intelligence*, 280:103216, 2020.

Daniel S. Bernstein, Robert Givan, Neil Immerman, and Shlomo Zilberstein. The complexity of decentralized control of Markov decision processes. *Mathematics of Operations Research*, 27(4): 819–840, 2002.

Dimitri P. Bertsekas and John N. Tsitsiklis. *Neuro-Dynamic Programming*. Athena Scientific, 1996.

David Blackwell. Equivalent comparisons of experiments. *The Annals of Mathematical Statistics*, 24 (2):265–272, 1953.

Murray Campbell, A. Joseph Hoane, Jr., and Feng-hsiung Hsu. Deep Blue. *Artificial Intelligence*, 134(1–2):57–83, 2002.

Micah Carroll, Rohin Shah, Mark K. Ho, Thomas L. Griffiths, Sanjit A. Seshia, Pieter Abbeel, and Anca Dragan. On the utility of learning about humans for human-AI coordination. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Pablo Samuel Castro. Scalable methods for computing state similarity in deterministic Markov decision processes. In *AAAI Conference on Artificial Intelligence*, volume 34, pp. 10069–10076, 2020.

Kate Donahue, Alexandra Chouldechova, and Krishnaram Kenthapadi. Human-algorithm collaboration: Achieving complementarity and avoiding unfairness. In *ACM Conference on Fairness, Accountability, and Transparency (FAccT)*, pp. 1639–1656, 2022.

Anca D. Dragan and Siddhartha S. Srinivasa. Generating legible motion. In *Robotics: Science and Systems (RSS)*, 2013.

Anca D. Dragan, Kenton C. T. Lee, and Siddhartha S. Srinivasa. Legibility and predictability of robot motion. In *ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pp. 301–308, 2013.

Norm Ferns, Prakash Panangaden, and Doina Precup. Metrics for finite Markov decision processes. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, pp. 162–169, 2004.

Niklas Fiekas. python-chess: A chess library for Python. https://python-chess.readthedocs.io, 2025.

Jaime F. Fisac, Monica A. Gates, Jessica B. Hamrick, Chang Liu, Dylan Hadfield-Menell, Malayandi Palaniappan, Dhruv Malik, S. Shankar Sastry, Thomas L. Griffiths, and Anca D. Dragan. Pragmatic-pedagogic value alignment. In *Robotics Research: The 18th International Symposium ISRR (2017), Springer Proceedings in Advanced Robotics, vol. 10*, pp. 49–57, 2020.

Alexey Gavryushin, Dingxi Zhang, Zhao Huang, Alexandros Delitzas, Jiaqi Chen, Ben Ellis, Cedric Zöllner, Manthan Patel, Manuel Kaufmann, Marc Pollefeys, and Xi Wang. CoMind: Understanding collaborative human activity from multiple minds and views. In *European Conference on Computer Vision (ECCV)*, 2026. arXiv:2607.06691.

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

David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2018.

Dylan Hadfield-Menell, Stuart J. Russell, Pieter Abbeel, and Anca Dragan. Cooperative inverse reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

Karim Hamade, Reid McIlroy-Young, Siddhartha Sen, Jon Kleinberg, and Ashton Anderson. Designing skill-compatible AI: Methodologies and frameworks in chess. In *International Conference on Learning Representations (ICLR)*, 2024. arXiv:2405.05066.

Nicholas Hay, Stuart Russell, David Tolpin, and Solomon Eyal Shimony. Selecting computations: Theory and applications. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, 2012.

Mark K. Ho, Michael L. Littman, James MacGlashan, Fiery Cushman, and Joseph L. Austerweil. Showing versus doing: Teaching by demonstration. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

Ronald A. Howard. Information value theory. *IEEE Transactions on Systems Science and Cybernetics*, 2(1):22–26, 1966.

Hengyuan Hu, Adam Lerer, Alex Peysakhovich, and Jakob Foerster. "other-play" for zero-shot coordination. In *International Conference on Machine Learning (ICML)*, volume 119 of *PMLR*, pp. 4399–4410, 2020.

Hengyuan Hu, Adam Lerer, Brandon Cui, Luis Pineda, Noam Brown, and Jakob Foerster. Off-belief learning. In *International Conference on Machine Learning (ICML)*, volume 139 of *PMLR*, pp. 4369–4379, 2021.

Garud N. Iyengar. Robust dynamic programming. *Mathematics of Operations Research*, 30(2): 257–280, 2005.

Shervin Javdani, Siddhartha S. Srinivasa, and J. Andrew Bagnell. Shared autonomy via hindsight optimization. In *Robotics: Science and Systems (RSS)*, 2015.

Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *International Conference on Machine Learning (ICML)*, pp. 267–274, 2002.

Alkis Kalavasis, Anay Mehrotra, and Grigoris Velegkas. On the limits of language generation: Trade-offs between hallucination and mode-collapse. In *ACM Symposium on Theory of Computing (STOC)*, pp. 1732–1743, 2025.

Adam Karvonen. Emergent world models and latent variable estimation in chess-playing language models. In *Conference on Language Modeling (COLM)*, 2024.

Michael Kearns and Satinder Singh. Near-optimal reinforcement learning in polynomial time. *Machine Learning*, 49(2–3):209–232, 2002.

Jon Kleinberg. AI's models of the world, and ours. https://www.youtube.com/watch?v=siu_r8j5-sg, 2025. Theoretically Speaking public lecture, Simons Institute for the Theory of Computing, Berkeley.

Jon Kleinberg and Sendhil Mullainathan. Language generation in the limit. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, pp. 66058–66079, 2024.

Jon Kleinberg and Manish Raghavan. Algorithmic monoculture and social welfare. *Proceedings of the National Academy of Sciences*, 118(22):e2018340118, 2021.

Paul Knott, Micah Carroll, Sam Devlin, Kamil Ciosek, Katja Hofmann, Anca Dragan, and Rohin Shah. Evaluating the robustness of collaborative agents. In *International Conference on Autonomous Agents and Multiagent Systems (AAMAS), extended abstract*, pp. 1560–1562, 2021.

Kenneth Li, Aspen K. Hopkins, David Bau, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Emergent world representations: Exploring a sequence model trained on a synthetic task. In *International Conference on Learning Representations (ICLR)*, 2023.

Lihong Li, Thomas J. Walsh, and Michael L. Littman. Towards a unified theory of state abstraction for MDPs. In *International Symposium on Artificial Intelligence and Mathematics (ISAIM)*, 2006.

Lichess. The lichess open database. https://database.lichess.org, 2025.

Andrei Lupu, Brandon Cui, Hengyuan Hu, and Jakob Foerster. Trajectory diversity for zero-shot coordination. In *International Conference on Machine Learning (ICML)*, volume 139 of *PMLR*, pp. 7204–7213, 2021.

David Madras, Toniann Pitassi, and Richard Zemel. Predict responsibly: Improving fairness and accuracy by learning to defer. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

Reid McIlroy-Young, Siddhartha Sen, Jon Kleinberg, and Ashton Anderson. Aligning superhuman AI with human behavior: Chess as a model system. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 1677–1687, 2020.

Reid McIlroy-Young, Russell Wang, Siddhartha Sen, Jon Kleinberg, and Ashton Anderson. Learning models of individual behavior in chess. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 1253–1263, 2022.

Reuth Mirsky, Ignacio Carlucho, Arrasy Rahman, Elliot Fosong, William Macke, Mohan Sridharan, Peter Stone, and Stefano V. Albrecht. A survey of ad hoc teamwork research. In *European Conference on Multi-Agent Systems (EUMAS)*, 2022.

Hussein Mozannar and David Sontag. Consistent estimators for learning to defer to an expert. In *International Conference on Machine Learning (ICML)*, volume 119 of *PMLR*, pp. 7076–7087, 2020.

Hussein Mozannar, Hunter Lang, Dennis Wei, Prasanna Sattigeri, Subhro Das, and David Sontag. Who should predict? exact algorithms for learning to defer to humans. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2023.

Neel Nanda, Andrew Lee, and Martin Wattenberg. Emergent linear representations in world models of self-supervised sequence models. In *BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pp. 16–30, 2023.

Frans A. Oliehoek and Christopher Amato. *A Concise Introduction to Decentralized POMDPs*. Springer, 2016.

Valerio Ortenzi, Akansel Cosgun, Tommaso Pardi, Wesley P. Chan, Elizabeth Croft, and Dana Kulić. Object handovers: A review for robotics. *IEEE Transactions on Robotics*, 37(6):1855–1873, 2021.

Michael Polanyi. *The Tacit Dimension*. Doubleday, Garden City, NY, 1966.

Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley Series in Probability and Statistics. Wiley, 1994.

Maithra Raghu, Katy Blumer, Greg Corrado, Jon Kleinberg, Ziad Obermeyer, and Sendhil Mullainathan. The algorithmic automation problem: Prediction, triage, and human effort. *arXiv preprint arXiv:1903.12220*, 2019.

Howard Raiffa and Robert Schlaifer. *Applied Statistical Decision Theory*. Division of Research, Graduate School of Business Administration, Harvard University, Boston, 1961.

Balaraman Ravindran and Andrew G. Barto. SMDP homomorphisms: An algebraic approach to abstraction in semi-Markov decision processes. In *International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 1011–1016, 2003.

Siddharth Reddy, Anca D. Dragan, and Sergey Levine. Shared autonomy via deep reinforcement learning. In *Robotics: Science and Systems (RSS)*, 2018.

Jonathan Richens and Tom Everitt. Robust agents learn causal world models. In *International Conference on Learning Representations (ICLR)*, 2024.

Stuart Russell and Eric Wefald. Principles of metareasoning. *Artificial Intelligence*, 49(1–3):361–395, 1991.

David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, Timothy Lillicrap, Karen Simonyan, and Demis Hassabis. A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science*, 362(6419):1140–1144, 2018.

Mark Steyvers, Heliodoro Tejeda, Gavin Kerrigan, and Padhraic Smyth. Bayesian modeling of human–AI complementarity. *Proceedings of the National Academy of Sciences*, 119(11):e2111547119, 2022.

Peter Stone, Gal A. Kaminka, Sarit Kraus, and Jeffrey S. Rosenschein. Ad hoc autonomous agent teams: Collaboration without pre-coordination. In *AAAI Conference on Artificial Intelligence*, volume 24, pp. 1504–1509, 2010.

Kyle Strabala, Min Kyung Lee, Anca Dragan, Jodi Forlizzi, Siddhartha S. Srinivasa, Maya Cakmak, and Vincenzo Micelli. Toward seamless human-robot handovers. *Journal of Human-Robot Interaction*, 2(1):112–132, 2013.

DJ Strouse, Kevin R. McKee, Matt Botvinick, Edward Hughes, and Richard Everett. Collaborating with humans without human data. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Zhenwei Tang, Difan Jiao, Reid McIlroy-Young, Jon Kleinberg, Siddhartha Sen, and Ashton Anderson. Maia-2: A unified model for human-AI alignment in chess. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, pp. 20919–20944, 2024. arXiv:2409.20553.

Shubham Toshniwal, Sam Wiseman, Karen Livescu, and Kevin Gimpel. Chess as a testbed for language model state tracking. In *AAAI Conference on Artificial Intelligence*, volume 36, pp. 11385–11393, 2022.

Keyon Vafa, Justin Y. Chen, Ashesh Rambachan, Jon Kleinberg, and Sendhil Mullainathan. Evaluating the world model implicit in a generative model. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, pp. 26941–26975, 2024.

Rajeev Verma and Eric Nalisnick. Calibrated learning to defer with one-vs-all classifiers. In *International Conference on Machine Learning (ICML)*, 2022.

Bryan Wilder, Eric Horvitz, and Ece Kamar. Learning to complement humans. In *International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 1526–1533, 2020.

Yiming Zhang, Athul Paul Jacob, Vivian Lai, Daniel Fried, and Daphne Ippolito. Human-aligned chess with a bit of search. In *International Conference on Learning Representations (ICLR)*, 2025. arXiv:2410.03893.

Xiaojin Zhu. Machine teaching: An inverse problem to machine learning and an approach toward optimal education. In *AAAI Conference on Artificial Intelligence*, volume 29, pp. 4083–4087, 2015.

A AUXILIARY RESULTS

This appendix collects the standard facts the proofs use, so that every argument can be checked without leaving the paper. Throughout, $\mathcal{S}$ and $\mathcal{A}$ are finite, $\gamma \in (0,1)$, rewards lie in $[0,R]$, $H = R/(1-\gamma)$, and $\|\cdot\|$ is the supremum norm on $\mathbb{R}^{\mathcal{S}}$. For $V \in \mathbb{R}^{\mathcal{S}}$,

$$q_V(s,a) = r(s,a) + \gamma\, \mathbb{E}_{s' \sim P(\cdot|s,a)} V(s'), \qquad T_1 V(s) = \max_a q_V(s,a),$$

$$T_p V(s) = p\, T_1 V(s) + (1-p)\, \mathbb{E}_{b \sim \pi_{\mathrm{w}}(\cdot|s)} q_V(s,b),$$

and for a policy π (possibly stochastic)

$$T_p^{\pi} V(s) = p\, \mathbb{E}_{a \sim \pi(\cdot|s)} q_V(s,a) + (1-p)\, \mathbb{E}_{b \sim \pi_{\mathrm{w}}(\cdot|s)} q_V(s,b).$$

For a policy σ and a start distribution μ_0, the normalised discounted occupancy is

$$d^{\sigma}_{\mu_0}(s) = (1-\gamma) \sum_{t \geq 0} \gamma^t\, \mathbb{P}_{\mu_0,\sigma}(s_t = s),$$

and V^{σ} is the value of σ.

Lemma 13 (Contraction and monotonicity)**.** Each of T_1, T_p and T_p^{π} is a γ-contraction in $\|\cdot\|$ and monotone, i.e. $V \leq W$ pointwise implies $TV \leq TW$ pointwise (Puterman, 1994, Ch. 6.2). Each has a unique fixed point (V_1^*, V_p^* and V_p^{π} respectively), and for every V and $d \geq 0$

$$\|T^d V - V^{\mathrm{fix}}\| \leq \gamma^d\, \|V - V^{\mathrm{fix}}\|.$$

Each maps $[0,H]^{\mathcal{S}}$ into itself.

Proof. Step 1: one-step values. For all V, W and (s,a),

$$q_V(s,a) - q_W(s,a) = \gamma\, \mathbb{E}_{s' \sim P(\cdot|s,a)} \big[V(s') - W(s')\big],$$

so $|q_V(s,a) - q_W(s,a)| \leq \gamma \|V - W\|$, and $V \leq W$ implies $q_V \leq q_W$.

Step 2: the operators. A maximum over actions and an expectation under a fixed distribution are monotone and satisfy

$$\Big|\max_a f(a) - \max_a g(a)\Big| \leq \max_a |f(a) - g(a)|, \qquad \big|\mathbb{E}_{\mu} f - \mathbb{E}_{\mu} g\big| \leq \max_a |f(a) - g(a)|.$$

Composed with Step 1, the maps $V \mapsto T_1 V$, $V \mapsto \mathbb{E}_{\pi_{\mathrm{w}}} q_V$ and $V \mapsto \mathbb{E}_{\pi} q_V$ are monotone γ-contractions, and so is any convex combination of them, which covers T_p and T_p^{π}.

Step 3: fixed points. Existence and uniqueness follow from Banach's theorem on the complete space $(\mathbb{R}^{\mathcal{S}}, \|\cdot\|)$, and $\|T^d V - V^{\mathrm{fix}}\| = \|T^d V - T^d V^{\mathrm{fix}}\| \leq \gamma^d \|V - V^{\mathrm{fix}}\|$.

Step 4: invariance. If $V \in [0,H]^{\mathcal{S}}$, then

$$0 \leq q_V(s,a) \leq R + \gamma H = H,$$

and each operator returns a maximum or an average of such values. □

Lemma 14 (Performance difference; Kakade & Langford, 2002, Lemma 6.1)**.** For any two policies σ, σ' of the same discounted MDP and any μ_0,

$$\mathbb{E}_{\mu_0}\big[V^{\sigma} - V^{\sigma'}\big] = \frac{1}{1-\gamma}\, \mathbb{E}_{s \sim d^{\sigma}_{\mu_0}}\, \mathbb{E}_{a \sim \sigma(\cdot|s)} \Big[q_{V^{\sigma'}}(s,a) - V^{\sigma'}(s)\Big].$$

Lemma 15 (Perturbed contraction)**.** Let $T, \widehat{T}$ be γ-contractions on a set $\mathcal{V}$ that both map $\mathcal{V}$ into itself, with $\sup_{V \in \mathcal{V}} \|\widehat{T}V - TV\| \leq \beta$, and let $V^{\star} \in \mathcal{V}$ be the fixed point of T. Then for every $V \in \mathcal{V}$ and $d \geq 0$,

$$\|\widehat{T}^{\,d} V - V^{\star}\| \leq \gamma^d\, \|V - V^{\star}\| + \beta\, \frac{1-\gamma^d}{1-\gamma}.$$

Proof. Since $\widehat{T}$ maps $\mathcal{V}$ into itself, $\widehat{T}^{\,d-1}V \in \mathcal{V}$ for every $d \geq 1$. Inserting $T(\widehat{T}^{\,d-1}V)$ and using $TV^\star = V^\star$,

$$\begin{aligned}\|\widehat{T}^{\,d}V - V^\star\| &\leq \big\|\widehat{T}(\widehat{T}^{\,d-1}V) - T(\widehat{T}^{\,d-1}V)\big\| + \big\|T(\widehat{T}^{\,d-1}V) - TV^\star\big\| \\ &\leq \beta + \gamma\,\|\widehat{T}^{\,d-1}V - V^\star\|.\end{aligned}$$

Unrolling the recursion d times gives $\gamma^d\|V-V^\star\|+\beta\sum_{i<d}\gamma^i$, and $\sum_{i<d}\gamma^i = (1-\gamma^d)/(1-\gamma)$. □

Lemma 16 (Total variation and span)**.** For probability vectors P, Q on a finite set and any f,

$$|\mathbb{E}_P f - \mathbb{E}_Q f| \leq \mathrm{TV}(P,Q)\cdot \operatorname{span} f, \qquad \mathrm{TV}(P,Q) = \tfrac{1}{2}\|P-Q\|_1, \quad \operatorname{span} f = \max f - \min f.$$

Proof. Let $c = \frac{1}{2}(\max f + \min f)$ be the midrange of f, so that $\max_b |f(b)-c| = \frac{1}{2}\operatorname{span} f$. Since $\sum_b (P-Q)(b) = 0$,

$$|\mathbb{E}_P f - \mathbb{E}_Q f| = \Big|\sum_b (P-Q)(b)\,\big(f(b)-c\big)\Big| \leq \|P-Q\|_1 \max_b |f(b)-c| = 2\,\mathrm{TV}(P,Q)\cdot\tfrac{1}{2}\operatorname{span} f.$$

□

Lemma 17 (Jensen for the maximum)**.** For a random vector $X \in \mathbb{R}^{\mathcal{A}}$,

$$\mathbb{E}\max_a X_a \geq \max_a \mathbb{E}X_a,$$

with equality if and only if some action attains the maximum almost surely.

Proof. $\max_a X_a \geq X_b$ pointwise for every b; take expectations and then the maximum over b. Equality forces $\mathbb{E}[\max_a X_a - X_{b^*}] = 0$ for a maximiser b^* of $\mathbb{E}X_b$, and the integrand is non-negative. □

Lemma 18 (Refinement)**.** Let φ' refine φ (every cell z of φ is a union of cells z' of φ'), let ν be a distribution on z, $\nu_{z'} = \nu(\cdot \mid z')$, and $\bar{q}_\nu(a) = \mathbb{E}_{s\sim\nu} q(s,a)$ with $q = q_{V_1^*}$. Then

$$I(z,\nu) = \sum_{z'\subseteq z} \nu(z')\, I(z',\nu_{z'}) + \underbrace{\Big[\sum_{z'\subseteq z}\nu(z')\max_a \bar{q}_{\nu_{z'}}(a) - \max_a \bar{q}_\nu(a)\Big]}_{\geq 0\text{: value of learning } z' \text{ inside } z}.$$

In particular, for a fixed state distribution the information term of Theorem 5 can only fall when the partition is refined.

Proof. Step 1: split the cell. Since $\mathbb{E}_\nu \max_a q = \sum_{z'}\nu(z')\mathbb{E}_{\nu_{z'}}\max_a q$, adding and subtracting $\sum_{z'}\nu(z')\max_a \bar{q}_{\nu_{z'}}(a)$ gives

$$I(z,\nu) = \sum_{z'}\nu(z')\Big[\mathbb{E}_{\nu_{z'}}\max_a q - \max_a \bar{q}_{\nu_{z'}}(a)\Big] + \Big[\sum_{z'}\nu(z')\max_a \bar{q}_{\nu_{z'}}(a) - \max_a \bar{q}_\nu(a)\Big],$$

and the first sum is $\sum_{z'}\nu(z')I(z',\nu_{z'})$.

Step 2: the bracket is non-negative. $\bar{q}_\nu = \sum_{z'}\nu(z')\,\bar{q}_{\nu_{z'}}$, so the bracket is $\mathbb{E}\max_a X_a - \max_a \mathbb{E}X_a$ for the random vector $X = \bar{q}_{\nu_{z'}}$ with $z' \sim \nu$, which is non-negative by Lemma 17. □

B Deferred proofs

Proof of Lemma 2. Fix s. Adding and subtracting $(1-p)\max_a q_V(s,a)$,

$$\begin{aligned}T_p V(s) &= p\max_a q_V(s,a) + (1-p)\,\mathbb{E}_{\pi_{\mathrm{w}}} q_V(s,\cdot) \\ &= \max_a q_V(s,a) - (1-p)\Big[\max_a q_V(s,a) - \mathbb{E}_{\pi_{\mathrm{w}}} q_V(s,\cdot)\Big] = T_1 V(s) - (1-p)\,\rho_V(s).\end{aligned}$$

$\rho_V \geq 0$ because $\max_a q_V(s,a) \geq q_V(s,b)$ for every b. Contraction and monotonicity are Lemma 13. □

Proof of Proposition 3. Step 1: the team process is an MDP. At s the action is drawn from $\pi(\cdot \mid s)$ with probability p and from $\pi_{\mathrm{w}}(\cdot \mid s)$ otherwise, independently of the past. The team process is therefore M run under the policy $\pi_p = p\,\pi + (1-p)\,\pi_{\mathrm{w}}$, so $V_p^{\pi} = V^{\pi_p}$ and $d_p^{\pi} = d_{\mu_0}^{\pi_p}$.

Step 2: performance difference. Apply Lemma 14 with $\sigma = \pi_p$ and σ' a solo-optimal policy, for which $V^{\sigma'} = V_1^*$ and $q_{V^{\sigma'}} = q$:

$$\mathbb{E}_{\mu_0}\left[V_1^* - V_p^{\pi}\right] = \frac{1}{1-\gamma}\,\mathbb{E}_{s\sim d_p^{\pi}}\,\mathbb{E}_{a\sim\pi_p(\cdot|s)}\left[V_1^*(s) - q(s,a)\right].$$

Step 3: split along the mixture. Using $V_1^*(s) = \max_a q(s,a)$,

$$\mathbb{E}_{a\sim\pi_p(\cdot|s)}\left[V_1^*(s) - q(s,a)\right] = p\,\underbrace{\left[\max_a q(s,a) - \mathbb{E}_{a\sim\pi} q(s,a)\right]}_{g_\pi(s)} + (1-p)\,\underbrace{\left[\max_a q(s,a) - \mathbb{E}_{b\sim\pi_{\mathrm{w}}} q(s,b)\right]}_{\rho(s)}.$$

□

Proof of Theorem 5. Step 1: group the partner term by cells. With $d = d_p^{\pi}$,

$$\mathbb{E}_{s\sim d}\,\rho(s) = \sum_{z:\,d_{\mathcal{Z}}(z)>0} d_{\mathcal{Z}}(z)\,\mathbb{E}_{s\sim\nu_z}\rho(s).$$

Fix a cell z with $d_{\mathcal{Z}}(z) > 0$ and write $\nu = \nu_z$.

Step 2: use measurability. Because $\pi_{\mathrm{w}}(\cdot \mid s) = \pi_{\mathrm{w}}(\cdot \mid z)$ for every $s \in z$ and the action set is finite,

$$\mathbb{E}_{s\sim\nu}\,\mathbb{E}_{b\sim\pi_{\mathrm{w}}(\cdot|s)} q(s,b) = \sum_b \pi_{\mathrm{w}}(b \mid z)\,\mathbb{E}_{s\sim\nu} q(s,b) = \sum_b \pi_{\mathrm{w}}(b \mid z)\,\bar{q}_\nu(b).$$

Step 3: add and subtract the best single action.

$$\begin{aligned}\mathbb{E}_\nu \rho &= \mathbb{E}_\nu \max_a q(s,a) - \sum_b \pi_{\mathrm{w}}(b \mid z)\,\bar{q}_\nu(b) \\ &= \underbrace{\left[\mathbb{E}_\nu \max_a q(s,a) - \max_a \bar{q}_\nu(a)\right]}_{I(z,\nu)} + \underbrace{\left[\max_a \bar{q}_\nu(a) - \sum_b \pi_{\mathrm{w}}(b \mid z)\,\bar{q}_\nu(b)\right]}_{\chi(z,\nu)}.\end{aligned}$$

Step 4: signs. $I \geq 0$ by Lemma 17 applied to $X_a = q(s,a)$ with $s \sim \nu$. $\chi \geq 0$ because an average of the values $\bar{q}_\nu(b)$ does not exceed their maximum, and $\chi = 0$ if and only if $\pi_{\mathrm{w}}(\cdot \mid z)$ is supported on $\arg\max_a \bar{q}_\nu(a)$.

Step 5: combine. Substituting Steps 1 and 3 into Proposition 3 gives the per-cell form of the theorem.

Step 6: the handed-over form. Since $\max_a q(s,a) - q(s,b) = \ell_s(b)$, the bracket defining $I(z,\nu)$ in Step 3 equals $\min_b \mathbb{E}_\nu \ell_s(b) = \min_b \bar{\ell}_\nu(b)$, and $\chi(z,\nu) = \mathbb{E}_{b\sim\pi_{\mathrm{w}}(\cdot|z)}\bar{\ell}_\nu(b) - \min_b \bar{\ell}_\nu(b)$. With $L(z) = (1-p)\,d_{\mathcal{Z}}(z)\,\bar{\ell}_{\nu_z}$ this gives $(1-p)\,d_{\mathcal{Z}}(z)\,I(z,\nu_z) = \min_a L(z)(a)$ and $(1-p)\,d_{\mathcal{Z}}(z)\,\chi(z,\nu_z) = \langle \pi_{\mathrm{w}}(\cdot \mid z), L(z)\rangle - \min_a L(z)(a)$, the form stated in Theorem 5. □

Proof of Corollary 6. The first claim drops the two non-negative terms g_π and χ from Theorem 5. For the second,

$$\sum_z d_{\mathcal{Z}}(z)\,I(z,\nu_z) \;\geq\; \sum_{z\in\mathcal{Z}_0} d_{\mathcal{Z}}(z)\,I(z,\nu_z) \;\geq\; e\sum_{z\in\mathcal{Z}_0} d_{\mathcal{Z}}(z) \;\geq\; e\,\alpha.$$

□

The second part of Corollary 6 is not vacuous. In the construction of Theorem 8 the partner itself enters the confusable cell from the root, and the within-cell distribution is uniform whatever the agent does, so every agent policy satisfies

$$d_{\mathcal{Z}}(z_x) \;\geq\; \alpha = (1-p)\,\gamma\,(1-\gamma), \qquad I(z_x,\nu_{z_x}) = e = 1 - \tfrac{1}{m}.$$

The assumption cannot be replaced by "$I(z,\nu) \geq e$ for every ν supported on z": a point mass has $I = 0$.

Proof of Proposition 7. Write $q_s = q(s,\cdot)$ and $D_{ss'} = q_s - q_{s'}$, and fix a selection $a^*(s) \in \arg\max_a q(s,a)$.

Step 1: a coupling bound. For every s' the action $a^*(s')$ is admissible in $\max_a \bar{q}_\nu(a)$, so $\max_a \bar{q}_\nu(a) \geq \mathbb{E}_{s\sim\nu} q_s(a^*(s'))$ and

$$I(z,\nu) \leq \mathbb{E}_{s\sim\nu}\big[q_s(a^*(s)) - q_s(a^*(s'))\big] \qquad \text{for every } s'.$$

Averaging over $s' \sim \nu$,

$$I(z,\nu) \leq G := \mathbb{E}_{(s,s')\sim\nu\otimes\nu}\big[q_s(a^*(s)) - q_s(a^*(s'))\big].$$

Step 2: symmetrise. The product measure is invariant under $(s,s') \mapsto (s',s)$, so also $G = \mathbb{E}\big[q_{s'}(a^*(s')) - q_{s'}(a^*(s))\big]$. Adding the two expressions,

$$2G = \mathbb{E}_{(s,s')}\Big[D_{ss'}\big(a^*(s)\big) - D_{ss'}\big(a^*(s')\big)\Big].$$

Step 3: bound by the span. The difference of two coordinates of a vector is at most its span, and the span is at most twice the sup-norm, so

$$I(z,\nu) \leq G \leq \tfrac{1}{2}\,\mathbb{E}_{(s,s')} \operatorname{span} D_{ss'} \leq \mathbb{E}_{(s,s')}\|D_{ss'}\|_\infty.$$

Step 4: invariance. Replacing q_s by $q_s + c(s)\mathbf{1}$ changes $D_{ss'}$ by $(c(s) - c(s'))\mathbf{1}$, which leaves its span unchanged. It changes $\mathbb{E}_\nu \max_a q$ and $\max_a \bar{q}_\nu$ by the same amount $\mathbb{E}_\nu c$, which leaves I unchanged.

Step 5: tightness. For $q_{s_1} = (1,0)$, $q_{s_2} = (0,1)$ and ν uniform, $I = 1 - \frac{1}{2} = \frac{1}{2}$. The two off-diagonal pairs have $D = (1,-1)$ and $D = (-1,1)$, with span 2 and sup-norm 1, each with probability $\frac{1}{4}$; the diagonal pairs have $D = 0$. Both upper bounds therefore equal $\frac{1}{2}$. □

Proof of Proposition 9. Write $L_d(z) = \sum_{s\in z} d(s)\,\ell_s$, so that $I_\varphi(d,q) = \sum_z \min_a L_d(z)(a)$ by the definition of I; cells with $d_{\mathcal{Z}}(z) = 0$ contribute 0.

Step 1: refinement lowers the information term. If φ' refines φ, every cell z of φ is a disjoint union of cells z' of φ', and $L_d(z) = \sum_{z'\subseteq z} L_d(z')$. A minimum of a sum is at least the sum of the minima, so

$$\min_a L_d(z)(a) \;\geq\; \sum_{z'\subseteq z} \min_a L_d(z')(a),$$

and summing over z gives $I_{\varphi'}(d,q) \leq I_\varphi(d,q)$. In cell z the difference is $d_{\mathcal{Z}}(z)$ times the bracket of Lemma 18, the value of learning z' inside z.

Step 2: the converse. Suppose φ' does not refine φ. Then some cell of φ' contains two states s_1, s_2 that lie in different cells of φ. Let d put mass $\frac{1}{2}$ on each of them, pick actions $a_1 \neq a_2$, and let $q(s_1,a_1) = q(s_2,a_2) = 1$ and $q = 0$ otherwise. Then $\ell_{s_1}(a_1) = \ell_{s_2}(a_2) = 0$ and every other coordinate of ℓ_{s_1} and ℓ_{s_2} equals 1. Under φ' the two states share a cell whose summed loss $\frac{1}{2}(\ell_{s_1} + \ell_{s_2})$ has every coordinate at least $\frac{1}{2}$, so $I_{\varphi'}(d,q) = \frac{1}{2}$. Under φ each of them is the only state with positive mass in its cell, so every cell has minimum 0 and $I_\varphi(d,q) = 0 < I_{\varphi'}(d,q)$. □

Proof of Proposition 10. Step 1: information partitions are the refinements of β. If π_{w} is φ-measurable and $\varphi(s) = \varphi(s')$, then $\pi_{\mathrm{w}}(\cdot \mid s) = \pi_{\mathrm{w}}(\cdot \mid s')$, so s and s' share a cell of β; every cell of φ therefore lies inside a cell of β. Conversely, if φ refines β, then π_{w} is constant on every cell of φ. In particular β and the identity partition are information partitions.

Step 2: the interval. The occupancy $d = d_p^\pi$ depends on π and on the partner's behaviour, not on the partition used to describe the partner. For an information partition φ, Step 1 and Step 1 of the proof of Proposition 9 give $I_\varphi(d,q) \leq I_\beta(d,q)$, and $I_\varphi \geq 0$ always. Every cell of the identity partition is a single state s, with $\min_a \ell_s(a) = 0$, so its information term is 0. Finally $\sum_z d_{\mathcal{Z}}(z)\big[I(z,\nu_z) + \chi(z,\nu_z)\big] = \mathbb{E}_d\rho$ for every information partition, by the cellwise identity $\mathbb{E}_\nu\rho = I + \chi$, and the right side does not depend on φ. □

Proof of Theorem 8. Step 1: the instance. States are s_0 (root), t (safe), $x_1, \dots, x_m$ (confusable) and an absorbing terminal state $\perp$ with reward 0; actions are $\{1, \dots, m\}$. At s_0, action 1 leads to each x_i with probability $\frac{1}{m}$ and every other action leads to t, all with reward 0. At t every action gives reward $1-\delta$ and leads to $\perp$. At x_i action i gives reward 1, every other action gives reward 0, and all lead to $\perp$. Rewards lie in $[0,1]$. The partner's partition puts $x_1, \dots, x_m$ in one cell z_x and every other state in a cell of its own; the partner plays action 1 at s_0 and an arbitrary distribution $\pi_{\mathrm{w}}(\cdot \mid z_x)$ on the cell (for example uniform, as in Figure 1c). π^{sharp} plays 1 at s_0, π^{solid} plays 2 at s_0, and both play i at x_i.

Step 2: solo values. $V_1^*(x_i) = 1$ and $V_1^*(t) = 1-\delta$, so

$$q(s_0, 1) = \gamma, \qquad q(s_0, a) = \gamma(1-\delta) \quad (a \neq 1).$$

Hence π^{sharp} is solo-optimal and $V_1^{\mathrm{sharp}}(s_0) - V_1^{\mathrm{solid}}(s_0) = \gamma\delta$.

Step 3: the cell. From $\mu_0 = \delta_{s_0}$ the states x_i can only be reached at time 1, each with probability $\frac{1}{m}$ times the probability that action 1 is taken at s_0; the latter is at least $1-p > 0$ because the partner takes it. So under every agent policy ν_{z_x} is uniform. Since $q(x_i, a) = \mathbb{1}[a = i]$,

$$\bar{q}_\nu(a) = \tfrac{1}{m} \text{ for every } a, \qquad I(z_x, \nu) = 1 - \tfrac{1}{m}, \qquad \chi(z_x, \nu) = \tfrac{1}{m} - \sum_b \pi_{\mathrm{w}}(b \mid z_x)\, \tfrac{1}{m} = 0,$$

for every partner rule on the cell.

Step 4: team values. Under either policy the agent plays i at x_i, so

$$V_p(x_i) = p + (1-p)\, \pi_{\mathrm{w}}(i \mid z_x), \qquad \frac{1}{m}\sum_i V_p(x_i) = p + \frac{1-p}{m}, \qquad V_p(t) = 1-\delta.$$

At s_0 the partner's move is the same under both policies, so only the agent's branch differs:

$$V_p^{\mathrm{solid}}(s_0) - V_p^{\mathrm{sharp}}(s_0) = p\left[\gamma(1-\delta) - \gamma\Big(p + \frac{1-p}{m}\Big)\right] = p\gamma\Big[(1-p)\Big(1 - \frac{1}{m}\Big) - \delta\Big],$$

which is positive under the assumption on δ.

Step 5: team optimality. $V_p^*(x_i) = p + (1-p)\pi_{\mathrm{w}}(i \mid z_x)$ is attained by playing i at x_i. At s_0 the agent compares

$$q_{V_p^*}(s_0, 1) = \gamma\Big(p + \frac{1-p}{m}\Big) \qquad \text{with} \qquad q_{V_p^*}(s_0, a) = \gamma(1-\delta) \quad (a \neq 1),$$

and the latter is larger, so π^{solid} is greedy with respect to V_p^* and hence team-optimal.

Step 6: exchange rate. Dividing the team gap by the solo gap,

$$\frac{V_p^{\mathrm{solid}}(s_0) - V_p^{\mathrm{sharp}}(s_0)}{V_1^{\mathrm{sharp}}(s_0) - V_1^{\mathrm{solid}}(s_0)} = \frac{p(1-p)(1-\frac{1}{m})}{\delta} - p. \qquad \square$$

The penalty form of $d = 1$ and the mixture reading of λ. With an exact oracle, $V^{(0)} = V_1^*$ and $T_1 V_1^* = V_1^*$, so with $\widehat{\rho} = \widehat{\rho}_{V_1^*}$

$$V^{(1)} = \Pi\big(V_1^* - \lambda(1-p)\,\widehat{\rho}\big).$$

For $0 \le \lambda(1-p) \le 1$ the clip is inactive, because

$$0 \le \mathbb{E}_{\widehat{\pi}_{\mathrm{w}}} q = V_1^* - \widehat{\rho} \le V_1^* - \lambda(1-p)\,\widehat{\rho} \le V_1^* \le H,$$

and then

$$q_{V^{(1)}}(s,a) = r(s,a) + \gamma\, \mathbb{E}_{s'}\big[V_1^*(s') - \lambda(1-p)\widehat{\rho}(s')\big] = q(s,a) - \lambda(1-p)\,\gamma\, \mathbb{E}_{s' \sim P(\cdot|s,a)}\widehat{\rho}(s'),$$

which is the penalty in Section 4. For a general V,

$$T_1 V - \lambda(1-p)\,\widehat{\rho}_V = \big(1 - \lambda(1-p)\big) \max_a q_V + \lambda(1-p)\, \mathbb{E}_{\widehat{\pi}_{\mathrm{w}}} q_V,$$

which for $\lambda(1-p) \in [0,1]$ is the team operator with control rate $p' = 1 - \lambda(1-p)$ and partner $\widehat{\pi}_{\mathrm{w}}$. By Lemma 13 it is a monotone γ-contraction that maps $[0,H]^{\mathcal{S}}$ into itself, so after the first step the clip Π is never active.

Proof of Theorem 12. Let $\mathcal{V} = [0, H]^{\mathcal{S}}$ and let $\widehat{T}$ be the team operator with partner $\widehat{\pi}_{\mathrm{w}}$,

$$\widehat{T}V = T_1 V - (1-p)\,\widehat{\rho}_V = p \max_a q_V + (1-p)\,\mathbb{E}_{\widehat{\pi}_{\mathrm{w}}} q_V.$$

By the paragraph above (with $\lambda = 1$), $V^{(0)} = \Pi\widehat{V} \in \mathcal{V}$ and $V^{(d)} = \widehat{T}^{\,d} V^{(0)}$, and T_p and $\widehat{T}$ are γ-contractions mapping $\mathcal{V}$ into itself.

Step 1: operator error. For $V \in \mathcal{V}$ and every s, by Lemma 16,

$$|\widehat{T}V(s) - T_p V(s)| = (1-p)\left|\mathbb{E}_{\widehat{\pi}_{\mathrm{w}}} q_V(s,\cdot) - \mathbb{E}_{\pi_{\mathrm{w}}} q_V(s,\cdot)\right| \le (1-p)\,\varepsilon\ \operatorname{span} q_V(s,\cdot) \le (1-p)\,\varepsilon\,\Delta.$$

Since $q_V(s,\cdot) \in [0, H]$ for $V \in \mathcal{V}$, $\Delta \le H$.

Step 2: starting error. Clipping is non-expansive towards points of $\mathcal{V}$ and $V_1^* \in \mathcal{V}$, so

$$\|V^{(0)} - V_p^*\| \le \|\Pi\widehat{V} - V_1^*\| + \|V_1^* - V_p^*\| \le \eta + A.$$

Step 3: value bound. Lemma 15 with $T = T_p$, $V^\star = V_p^*$ and $\beta = (1-p)\varepsilon\Delta$ gives the first bound of the theorem.

Step 4: policy bound. Write $V = V^{(d)}$, $V^\star = V_p^*$ and $\pi = \pi_{1,d}$, so that $q_V(s, \pi(s)) = \max_a q_V(s,a)$. Then

$$V^\star - V_p^\pi = \underbrace{\left(T_p V^\star - T_p^\pi V^\star\right)}_{\text{(i)}} + \underbrace{\left(T_p^\pi V^\star - T_p^\pi V_p^\pi\right)}_{\text{(ii)}}.$$

In (i) the partner's terms cancel, and

$$\begin{aligned}
\text{(i)} &= p\Big[\max_a q_{V^\star}(s,a) - q_{V^\star}(s,\pi(s))\Big] \\
&\le p\Big(\Big[\max_a q_{V^\star}(s,a) - \max_a q_V(s,a)\Big] + \Big[q_V(s,\pi(s)) - q_{V^\star}(s,\pi(s))\Big]\Big) \\
&\le 2p\gamma\,\|V - V^\star\|.
\end{aligned}$$

In (ii) the rewards cancel, leaving $\gamma P_{\pi_p}(V^\star - V_p^\pi)$, where P_{π_p} is the transition matrix of the team process. Since $V^\star \ge V_p^\pi$, taking norms,

$$\|V^\star - V_p^\pi\| \le 2p\gamma\,\|V - V^\star\| + \gamma\,\|V^\star - V_p^\pi\|, \qquad \text{so} \qquad \|V^\star - V_p^\pi\| \le \frac{2p\gamma}{1-\gamma}\,\|V - V^\star\|. \ \square$$

Proposition 19 (Two things the bound does not give)**.** (a) With an exact oracle and an exact partner model, $V_p^* \le V^{(d+1)} \le V^{(d)}$, yet there is a five-state MDP in which the team values of $\pi_{1,0}$, $\pi_{1,1}$ and $\pi_{1,2}$ are 0.72, 0.5625 and 0.72. (b) If the partner degrades to $\pi_{\mathrm{w}}^\theta = (1-\theta)\pi_{\mathrm{w}} + \theta u$, then $\rho_V^\theta = (1-\theta)\rho_V^{\pi_{\mathrm{w}}} + \theta\rho_V^u$ exactly, but the scalar multiple of $\rho_V^{\pi_{\mathrm{w}}}$ that best represents ρ_V^θ can rise or fall with θ: with two actions, a partner that errs with probability 0.1 and u uniform, $\rho^\theta = (1+4\theta)\rho^{\pi_{\mathrm{w}}}$.

Proof of Proposition 19. Step 1: the value iterates decrease. Since $\rho_V \ge 0$, $T_p V \le T_1 V$ for every V. Hence $V^{(1)} = T_p V_1^* \le T_1 V_1^* = V^{(0)}$, and by monotonicity of T_p,

$$V^{(d)} \le V^{(d-1)} \quad \Longrightarrow \quad V^{(d+1)} = T_p V^{(d)} \le T_p V^{(d-1)} = V^{(d)}.$$

Step 2: they stay above the team value. $V_p^* = T_p V_p^* \le T_1 V_p^*$, and iterating the monotone T_1 gives $V_p^* \le V_1^*$. Therefore

$$V^{(d)} = T_p^d V_1^* \ \ge\ T_p^d V_p^* = V_p^*.$$

Step 3: the counterexample to monotone policies. Let $\gamma = 0.9$, $p = \frac{1}{2}$, states $s_0, x, y, z, \perp$ ($\perp$ absorbing with reward 0) and actions $\{0, 1\}$:

state	action 0	action 1	partner plays
s_0	reward 0, go to x	reward 0, go to y	0
x	reward 1, end	reward 0.6, end	1
y	reward 0, go to z	reward 0, go to z	0
z	reward 1, end	reward 0, end	1

The partner loses 0.4 at x and 1 at z, so the exact team values below the root are $V_p(x) = \frac{1}{2} \cdot 1 + \frac{1}{2} \cdot 0.6 = 0.8$, $V_p(z) = 0.5$ and $V_p(y) = 0.9 \cdot 0.5 = 0.45$. Going to x is worth $0.9 \cdot 0.8 = 0.72$ to the team, since the partner also goes to x at the root; going to y is worth $\frac{1}{2} \cdot 0.9 \cdot 0.45 + \frac{1}{2} \cdot 0.9 \cdot 0.8 = 0.5625$. The corrected values and the greedy root choices are

d	$V^{(d)}$ on (s_0, x, y, z)	$q_{V^{(d)}}(s_0, 0)$	$q_{V^{(d)}}(s_0, 1)$	root choice of $\pi_{1,d}$	team value of $\pi_{1,d}$
0	(0.9, 1, 0.9, 1)	0.9	0.81	x	0.72
1	(0.9, 0.8, 0.9, 0.5)	0.72	0.81	y	0.5625
2	(0.765, 0.8, 0.45, 0.5)	0.72	0.405	x	0.72

One correction lowers the value of x but not yet that of y, because the partner's loss on the x branch is one step from the root and its loss on the y branch two; the greedy policy switches to y, and a second correction switches it back.

Step 4: degradation. For every s, by linearity of the expectation in the policy,

$$\mathbb{E}_{\pi_{\mathrm{w}}^{\theta}} q_V(s, \cdot) = (1-\theta)\, \mathbb{E}_{\pi_{\mathrm{w}}} q_V(s, \cdot) + \theta\, \mathbb{E}_u q_V(s, \cdot), \qquad \max_a q_V = (1-\theta) \max_a q_V + \theta \max_a q_V,$$

and subtracting the first identity from the second gives $\rho_V^{\theta} = (1-\theta)\rho_V^{\pi_{\mathrm{w}}} + \theta \rho_V^u$.

Step 5: the scalar can rise. With two actions whose values differ by $g(s) \geq 0$, a partner that plays the worse action with probability 0.1 and u uniform give $\rho^{\pi_{\mathrm{w}}} = 0.1\, g$ and $\rho^u = 0.5\, g$, so

$$\rho^{\theta} = (0.1 + 0.4\,\theta)\, g = (1 + 4\theta)\, \rho^{\pi_{\mathrm{w}}},$$

and the exact scalar rises with θ. For a falling example, take any V with $\langle \rho_V^u, \rho_V^{\pi_{\mathrm{w}}} \rangle_w < \|\rho_V^{\pi_{\mathrm{w}}}\|_w^2$ and apply Proposition 22. □

C Extensions of the theory

State-dependent control and strict alternation. Let $c : \mathcal{S} \to [0, 1]$ give the probability that the agent controls s; strict alternation with role phases stored in the state has $c(s) \in \{0, 1\}$. The team operator is

$$T_c V(s) = c(s) \max_a q_V(s, a) + \big(1 - c(s)\big)\, \mathbb{E}_{\pi_{\mathrm{w}}} q_V(s, \cdot) = T_1 V(s) - \big(1 - c(s)\big)\, \rho_V(s),$$

and the proofs of Proposition 3 and Theorem 5 go through with the weights p and $1 - p$ moved inside the expectation:

$$\mathbb{E}_{\mu_0}[V_1^* - V_c^{\pi}] = \frac{1}{1-\gamma}\, \mathbb{E}_{s \sim d}\big[c(s)\, g_{\pi}(s)\big] + \frac{1}{1-\gamma} \sum_z \tilde{d}_{\mathcal{Z}}(z) \big[I(z, \tilde{\nu}_z) + \chi(z, \tilde{\nu}_z)\big],$$

where $\tilde{d}(s) = d(s)(1 - c(s))$ is the occupancy of the states at which the partner acts and $\tilde{\nu}_z = \tilde{d}(\cdot \mid z)$. The information and decision terms are thus evaluated exactly where handoffs happen. For the two-team games of Section 4.1 the opposing team's policies are part of the environment, and the own-partner risk in Algorithm 1 is ρ evaluated at the states reached after the opponent's reply, the states with $c = 0$.

Two-team zero-sum games. In a turn-taking zero-sum game in which our team controls its moves with rate p and partner π_{w} and the opposing team with rate p' and partner π'_{w}, define at their states

$$\rho_V^{\mathrm{them}}(s) = \mathbb{E}_{\pi'_{\mathrm{w}}} q_V(s, \cdot) - \min_a q_V(s, a) \;\geq\; 0.$$

The same algebra gives the operator

$$(TV)(s) = \begin{cases} \max_a q_V(s,a) - (1-p)\,\rho_V^{\mathrm{us}}(s) & \text{at our states,} \\ \min_a q_V(s,a) + (1-p')\,\rho_V^{\mathrm{them}}(s) & \text{at theirs,} \end{cases}$$

a γ-contraction whose two correction terms are the *helping* and *tricking* effects of Hamade et al. (2024). The opposing handoff must be priced once: through the reply distribution in Algorithm 1 or through an explicit bonus, not both.

Proposition 20 (Approximate measurability)**.** Suppose the partner is not exactly measurable but, on a cell z, $\mathrm{TV}(\pi_{\mathrm{w}}(\cdot \mid s), \bar{\pi}_z) \le \varepsilon_z(s)$ for a cell-level policy $\bar{\pi}_z$. Then for every distribution ν on z,

$$\mathbb{E}_\nu \rho \;\ge\; I(z,\nu) + \bar{\chi}(z,\nu) - \mathbb{E}_{s\sim\nu}\big[\varepsilon_z(s)\ \mathrm{span}\, q(s,\cdot)\big] \;\ge\; I(z,\nu) - \mathbb{E}_{s\sim\nu}\big[\varepsilon_z(s)\ \mathrm{span}\, q(s,\cdot)\big],$$

where $\bar{\chi}$ is χ computed for $\bar{\pi}_z$.

Proof. Step 1: replace the partner by its cell representative.

$$\mathbb{E}_\nu \rho = \underbrace{\mathbb{E}_\nu \max_a q - \mathbb{E}_\nu \mathbb{E}_{\bar{\pi}_z} q}_{= I(z,\nu) + \bar{\chi}(z,\nu)} + \mathbb{E}_{s\sim\nu}\Big[\mathbb{E}_{\bar{\pi}_z} q(s,\cdot) - \mathbb{E}_{\pi_{\mathrm{w}}(\cdot\mid s)} q(s,\cdot)\Big].$$

The first part is $I + \bar{\chi}$ by Steps 2–3 of the proof of Theorem 5, since $\bar{\pi}_z$ is measurable.

Step 2: bound the remainder. By Lemma 16, the bracket is at least $-\varepsilon_z(s)\ \mathrm{span}\, q(s,\cdot)$ for every s. The second inequality drops $\bar{\chi} \ge 0$. □

This is what licenses clustering a learned partner's behaviour into approximate cells: the floor degrades by the within-cell behavioural spread times the action-value spread, not by the size of the cell. In our navigation experiment the partition is known exactly (Appendix E), so the proposition is not needed there.

Proposition 21 (What the chess estimator bounds)**.** At a handoff state let $\ell_b = \max_a q(a) - q(b)$ be the true losses and $\rho = \sum_b \pi_{\mathrm{w}}(b)\ell_b$ the true handoff regret. Suppose the oracle satisfies $|\tilde{q}(b) - q(b)| \le \eta_Q$ for every legal b, the model satisfies $\mathrm{TV}(\widehat{\pi}_{\mathrm{w}}, \pi_{\mathrm{w}}) \le \varepsilon$, and $\mathrm{span}\, q \le D$. Let $\tilde{\ell}_b = \max_a \tilde{q}(a) - \tilde{q}(b)$, let B_m be the model's top-m moves with mass w_m, and $\widehat{\rho}_m = \sum_{b\in B_m} \widehat{\pi}_{\mathrm{w}}(b)\tilde{\ell}_b$. Then, with $e = 2\eta_Q + \varepsilon D$,

$$\max\{0, \widehat{\rho}_m - e\} \;\le\; \rho \;\le\; \widehat{\rho}_m + (1-w_m)\max_b \tilde{\ell}_b + e.$$

Without oracle and model error ($e = 0$), $\widehat{\rho}_m$ is a lower bound on the model's handoff regret, and the renormalised $\widehat{\rho}_m / w_m$ used by Expector lies between $\widehat{\rho}_m$ and $\widehat{\rho}_m + (1-w_m)\max_{b\in B_m} \tilde{\ell}_b$: it values the tail at the retained average loss.

Proof. Step 1: oracle error. $|\tilde{\ell}_b - \ell_b| \le 2\eta_Q$ for every b, so

$$\Big|\sum_b \widehat{\pi}_{\mathrm{w}}(b)\,\tilde{\ell}_b - \sum_b \widehat{\pi}_{\mathrm{w}}(b)\,\ell_b\Big| \le 2\eta_Q.$$

Step 2: model error. By Lemma 16 applied to $f = \ell$,

$$\Big|\sum_b \widehat{\pi}_{\mathrm{w}}(b)\,\ell_b - \rho\Big| \le \varepsilon\ \mathrm{span}\,\ell = \varepsilon\ \mathrm{span}\, q \le \varepsilon D.$$

Step 3: truncation. Every dropped term is non-negative and at most $\max_b \tilde{\ell}_b$, so

$$\widehat{\rho}_m \le \sum_b \widehat{\pi}_{\mathrm{w}}(b)\,\tilde{\ell}_b \le \widehat{\rho}_m + (1-w_m)\max_b \tilde{\ell}_b.$$

Chaining Steps 1–3 gives the two inequalities of the proposition. Finally, $\widehat{\rho}_m / w_m$ is a weighted average of the retained losses, so it lies between 0 and $\max_{b\in B_m} \tilde{\ell}_b$, and $\widehat{\rho}_m/w_m - \widehat{\rho}_m = (1-w_m)\,\widehat{\rho}_m/w_m$ gives the stated range. □

In chess, the differences between Stockfish searches of different depths are not a valid η_Q and the average error of Maia-2 on a test set is not a valid ε; Proposition 21 says which quantities would have to be controlled for the estimate to become a guarantee, and Section 5.3 checks empirically how well the estimate predicts real players.

A decision-error companion to Theorem 8. Keep the instance of Theorem 8 but give the partner a finer partition in which every state is its own cell, and let it play uniformly at the x_i. Nothing about the values changes: the solo gap is $\gamma\delta$, the team gap is $p\gamma[(1-p)(1-\frac{1}{m})-\delta]$, and π^{solid} is team-optimal. But now

$$I = 0 \text{ at every state}, \qquad \chi(x_i) = \max_a q(x_i, a) - \tfrac{1}{m}\sum_a q(x_i, a) = 1 - \tfrac{1}{m} :$$

the partner sees which state it is in and still guesses. The two instances have the same trade-off and opposite decompositions, and they call for opposite remedies. In the aliasing instance a better decision rule on the same information cannot help and only more information (or avoiding the cell) can; in this one, a partner that plays i at x_i removes the loss with no new information. Theorem 5 names the difference; the partition, or an intervention, reveals it. With the uniform cell rule the two partners act identically, and the behavioural partition differs from the aliasing partition only by states whose loss vectors vanish (t and $\perp$), so the first partner attains I_β and the second, which sees the state, has $I = 0$: they are the two ends of Proposition 10.

Proposition 22 (Least-squares reading of a scalar tilt)**.** Let $\langle f, g\rangle_w = \sum_s w(s)f(s)g(s)$ for a weight w on states. The least-squares coefficient of ρ_V^θ on $\rho_V^{\pi_w}$ is

$$\lambda_{\text{LS}}(\theta) = (1-\theta) + \theta\,\frac{\langle \rho_V^u, \rho_V^{\pi_w}\rangle_w}{\|\rho_V^{\pi_w}\|_w^2},$$

which is affine in θ and decreases exactly when $\langle \rho_V^u, \rho_V^{\pi_w}\rangle_w < \|\rho_V^{\pi_w}\|_w^2$.

Proof. Project both sides of the identity $\rho_V^\theta = (1-\theta)\rho_V^{\pi_w} + \theta\rho_V^u$ (Proposition 19(b)) onto $\rho_V^{\pi_w}$:

$$\lambda_{\text{LS}}(\theta) = \frac{\langle \rho_V^\theta, \rho_V^{\pi_w}\rangle_w}{\|\rho_V^{\pi_w}\|_w^2} = (1-\theta) + \theta\,\frac{\langle \rho_V^u, \rho_V^{\pi_w}\rangle_w}{\|\rho_V^{\pi_w}\|_w^2}. \qquad \square$$

The condition compares the size of the reference policy's regret with the modelled partner's where the latter is large; it is not a statement about alignment alone, and it can fail, as the two-action example in Proposition 19 shows. Whether the best λ rises or falls with degradation is therefore an empirical question for each task.

D CHESS: PROTOCOL, AUDIT AND FURTHER RESULTS

Algorithm 1 Handoff-aware decoding at one senior decision (tag-team chess)

Require: position s; oracle V; partner models $\widehat{\pi}_{\text{w}}^{\text{us}}, \widehat{\pi}_{\text{w}}^{\text{them}}$; K, m; role indicators $w_{\text{self}}, w_{\text{opp}}$; λ

$\mathcal{C} \leftarrow$ the oracle's top-K moves at s

for $a \in \mathcal{C}$ **do**

  $\mathcal{R} \leftarrow \{(\text{oracle reply},\ 1 - w_{\text{opp}})\} \cup \{(b,\ w_{\text{opp}}\,\widehat{\pi}_{\text{w}}^{\text{them}}(b)/w_m^{\text{them}})\}_{b \in B_m^{\text{them}}}$ ▷ sums to one

  $\bar{v}(a) \leftarrow \sum_{(c,w)\in\mathcal{R}} w\,V(s\,a\,c)$; $\bar{\rho}(a) \leftarrow \sum_{(c,w)\in\mathcal{R}} w\,\widehat{\rho}_m(s\,a\,c)$ ▷ raw $\widehat{\pi}_{\text{w}}^{\text{us}}$ weights

return $\arg\max_{a\in\mathcal{C}}\ \bar{v}(a) - \lambda\, w_{\text{self}}\,\bar{\rho}(a)$

Held-out openings and pre-registration. The opening book used for every chess result in this paper was generated for this study: 1,000 positions reached by eight plies sampled from Maia-2 at 1900 with a new random seed, kept when the engine's expected score lies in $[0.40, 0.60]$ and discarded if the position (ignoring move counters) occurs in the book used during development. Before any game on these openings was played we fixed, in a dated note released with the code, the primary setting (Maia-2 at 1500, strict alternation, openings 0–799), the primary rule (HAD-4 with $\lambda = 1$, $K = 6$, $m = 4$, two opponent-junior replies), the comparisons against R (OA, Expector, matched compute,

the sharpness control, HAD-4 without the opponent mixture), the three head-to-head comparisons, the list of secondary and exploratory runs, and the success criterion (HAD-*4 beats the matched-compute engine head to head with a* 95% *interval excluding* 50%*; otherwise we report that the gain cannot be distinguished from spending the same compute on search*). The matched budget, 447,955 nodes per decision, was HAD-4's mean in a pilot on development openings; on the held-out games HAD-4 used $366k$ nodes per decision on average, so the matched engine searched 22% more than the rule it is compared with. No rule, setting or threshold was changed after held-out games were played.

Protocol details. Stockfish 17.1 with one thread and a 64 MB hash table per process, driven through python-chess (Fiekas, 2025); values are expected scores from Stockfish's WDL model. Juniors are Maia-2 (rapid checkpoint), conditioned on their own and the opponent's rating and sampled. Junior moves and coin flips draw from random streams keyed by (seed, colour, ply), so two games from the same opening consume randomness identically even after they diverge. Games are adjudicated when one side's expected score stays beyond 0.5 ± 0.45 for six consecutive plies, or at 240 plies (stricter rule: 0.49 for twelve plies). The own-partner support uses the model's top-m moves with raw probabilities; the opponent's reply distribution gives weight $1 - w_{\text{opp}}$ to the opponent senior's best reply and w_{opp} to the opponent junior's two most likely moves, renormalised, so that the weights sum to one. Under strict alternation $w_{\text{self}} = 1$ and $w_{\text{opp}} \in \{0, 1\}$ by role phase; under the coin-flip schedule both are $\frac{1}{2}$.

Implementation audit. Before the held-out runs we checked the implementation against the items a reader would need to trust it. By construction, the reply distribution is normalised (its weights are divided by their sum), and the node count of a multi-line search is counted once rather than once per line (a bug in our first implementation, fixed before these runs). Numerically: terminal positions are valued correctly (1 for a mate delivered by White, $\frac{1}{2}$ for stalemate); under strict alternation, after its own senior move White's senior always meets the opposing senior and Black's always meets the opposing junior, as stated in Section 4.1; in 60 cases (15 positions, $m \in \{1, 2, 4, 8\}$) the truncated regret is non-decreasing in m, never exceeds the full-support value, and Expector's estimate equals $\widehat{\rho}_m / w_m$; and with the full partner support HAD-4 and Expector choose the same move in 12 of 12 test positions (at reduced depth). *Oracle consistency.* A position and its colour-mirrored twin should receive complementary values; with fresh hash tables the discrepancy averages 0.014 expected score at depth 12 (maximum 0.042) and 0.016 at depth 10 (maximum 0.074), and repeating a search with a warm hash table changes the value by 0.012 on average. In one of 40 tactical positions at depth 10 with a shared hash table the two searches differed by 0.33. Differences of this size between searches are why we treat the chess rule as an approximation to the theory rather than an instance of it.

Setting	OA	HAD-4	Expector	matched	budget 0.01
Maia-2 @1100, alternation	48.4 ± 3.0	54.5 ± 3.1	55.8 ± 3.1	51.1 ± 2.9	50.3 ± 3.1
Maia-2 @1500, alternation	50.1 ± 2.3	56.9 ± 2.2	60.1 ± 2.2	50.7 ± 1.9	51.6 ± 2.2
Maia-2 @1900, alternation	52.9 ± 3.2	59.8 ± 3.1	60.4 ± 3.2	52.9 ± 3.0	53.6 ± 3.3
Maia-2 @1500, coin flip	51.1 ± 2.8	51.2 ± 2.9	56.2 ± 2.9	–	–
@1500, strict adjudication	–	59.4 ± 3.2	–	–	–

Table 2: Win share (%) against R with 95% intervals across partner ratings, schedules and adjudication rules. Rows at 1100 and 1900 use 400 opening pairs, the 1500 row repeats Table 1 (800 pairs). Under the coin-flip schedule the handoff after a senior move happens half the time.

Cost per decision. HAD-4 takes 1.15 s per senior decision on one core (90th percentile 1.54 s) against 0.035 s for the raw engine, and about 46 engine calls: six candidates, three replies each, and a four-move partner support at each resulting position. Adaptive support uses 5.0 partner moves on average and retains 0.91 of the model's mass, against 0.83 for $m = 4$.

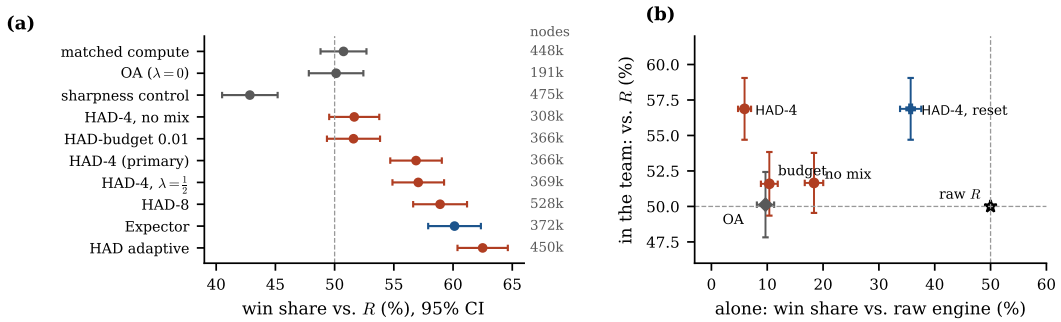


Figure 4: Tag-team chess at 1500. (a) Win share against R with 95% intervals and engine nodes per decision; the matched-compute engine searches more than every partner-aware rule. (b) The same configuration in the team (vertical) and playing every move alone against the raw engine (horizontal), keeping its team assumptions; the reset agent is HAD-4 told that no partner will move.

Alone. In the solo runs the focal senior makes every move and the reference is the raw engine making every move; each opening is played with both colours. A senior that keeps its team assumptions uses $w_{\text{self}} = 1$, the value at every senior decision under alternation, and $w_{\text{opp}} = \frac{1}{2}$, the average over the two role phases; the reset agent uses $w_{\text{self}} = w_{\text{opp}} = 0$, which removes both the handoff term and the model of the opposing junior and leaves a two-ply search over six candidates with depth-10 replies. Results against the raw engine (400 opening pairs each): OA $9.7 \pm 1.5\%$, HAD-4 $5.9 \pm 1.2\%$, HAD-4 without the opponent mixture $18.4 \pm 1.7\%$, HAD-budget with $B = 0.01$ $10.4 \pm 1.5\%$, reset HAD-4 $35.7 \pm 1.9\%$. The budgeted rule is not much better alone than HAD-4 because its candidate values still include the opposing junior; its own solo budget applies to the handoff term only.

Three decisions on the board. Figure 2 and Table 3 show what the handoff term changes at a single decision. The positions come from games of the reference team against itself, started from the first 20 held-out openings under strict alternation; at every senior decision of White we ran HAD-4 with the settings of Section 5.1. White's senior always meets the opposing senior, so each candidate hands the partner one position. Of 341 decisions, 81 were overrides of the $\lambda = 0$ choice, the move the same rule plays without the handoff term, as if playing alone. We show the three with the largest reduction in estimated handoff regret among those whose two-ply solo cost is at most 0.03 and whose value lies between 0.15 and 0.85, a rule fixed before looking at any position (`code/examples_boards.py`). In each case the move chosen alone keeps slightly more value but hands over a position whose most likely continuation loses heavily, while the team move gives up at most 0.02 and hands over a position in which the partner's likely moves are safe. In (b) the team move is a check that leaves the capture of the rook to the partner, which then finds it with probability 0.96.

		move	reply	$\bar{v}$	$\widehat{\rho}$	partner's likely move	prob.	loss
(a)	alone	Nxb5	Ba6	.472	.413	a4	.75	.47
(a)	team	Bb2	Qd7	.459	.073	Nc2	.37	.05
(b)	alone	gxh5	Qc7	.514	.265	Bxg5	.95	.28
(b)	team	Qxe5+	Kg8	.494	.023	gxh5	.96	.00
(c)	alone	Qxd4	Qc7	.508	.213	Bxd5	.42	.49
(c)	team	Nxd4	Qa6	.507	.008	Qe2+	.25	.00

Table 3: The two moves of Figure 2: the opposing senior's reply, the two-ply value $\bar{v}$ and the estimated handoff regret $\widehat{\rho}$ of the position handed over (expected score, White's view), and the partner's most likely move there, with its probability and its loss.

E NAVIGATION: DESIGN AND FULL RESULTS

Design. The grid is 12×12 with 30% of directed edges closed (no node is left with fewer than two exits) and $\gamma = 0.92$; a blocked move leaves the position unchanged but is recorded in the move history. The reward is 1 on entering the goal, after which the process terminates. The common state is

(goal, position, last three moves), with a padding symbol before the first move; a partner with window k sees the displacement to the goal and the last k moves, so its cells are (displacement, last k moves) and the partitions are nested in k. For each map, goals reachable from at least half of the grid are split at random into 72 training, 24 validation and 24 evaluation goals. The partner is a two-layer pre-norm transformer ($d_{\text{model}} = 64$, four heads) trained by cross-entropy for six epochs on shortest paths (random tie-breaking) from random starts to training goals; the same trajectories, truncated to the last k moves, train every k. The team process starts at a uniformly random position that can reach an evaluation goal, with an empty history; it is solved exactly by sparse linear algebra on $18{,}000$ states per goal, and discounted occupancies are computed by the transposed solve. We use 3 maps and 5 partner seeds per map.

Reference distribution and interventions. The fixed reference distribution of Figure 3a is the occupancy of a team in which the agent plays solo-optimally and the partner follows the shortest-path expert with probability 0.7 and acts uniformly otherwise; it depends on neither k nor the learned partner. The Bayes rule plays, in each cell, the action with the highest average value under the reference occupancy of the *training* goals, falling back to the learned partner in cells unseen in training. Disclosure reveals, in a chosen set of cells, which of the four exits of the current corner are open, splitting each chosen cell into up to 16 sub-cells in which the Bayes rule is refitted; the other cells keep the Bayes rule, so the comparison with the Bayes row isolates the information added. Cells are chosen by their information term under the training reference distribution, by the value of the revealed feature (the drop in information term it would bring), by the entropy of the learned partner weighted by reference mass, or at random with probability proportional to reference mass (three draws). The agent's interventions use the learned partner as an exact model: $\text{HAD}_{1,d}$ from the exact solo value, and the exact team optimum.

	$k=0$	$k=1$	$k=2$	$k=3$
information term I (fixed reference)	0.0067	0.0054	0.0047	0.0041
decision term χ, learned partner	0.0070	0.0077	0.0081	0.0083
decision term χ, Bayes rule	0.0019	0.0030	0.0039	0.0048
shortest-path accuracy	0.754	0.765	0.766	0.772
team regret, learned partner	0.1015	0.0883	0.0861	0.0847

Table 4: Navigation, one fixed reference distribution for every k, means over 15 runs. The information term falls with k as nested partitions require; the learned partner's decision term rises; the Bayes rule's decision term on evaluation goals is not zero because it is fitted on training goals.

	$k=0$				$k=3$			
Intervention	gap	own	I	χ	gap	own	I	χ
learned partner	0.1015	0.0000	0.0615	0.0400	0.0847	0.0000	0.0369	0.0477
Bayes rule on the same cells	0.0868	0.0000	0.0621	0.0246	0.0780	0.0000	0.0356	0.0424
disclose 2%, random	0.0854	0.0000	0.0589	0.0265	0.0723	0.0000	0.0260	0.0463
disclose 2%, by I	0.0815	0.0000	0.0527	0.0287	0.0670	0.0000	0.0190	0.0479
disclose 2%, by entropy	0.0815	0.0000	0.0528	0.0287	0.0661	0.0000	0.0182	0.0479
disclose 5%, random	0.0836	0.0000	0.0537	0.0299	0.0682	0.0000	0.0177	0.0505
disclose 5%, by I	0.0798	0.0000	0.0441	0.0357	0.0648	0.0000	0.0127	0.0521
disclose 5%, by entropy	0.0798	0.0000	0.0438	0.0360	0.0631	0.0000	0.0111	0.0520
disclose 10%, random	0.0810	0.0000	0.0464	0.0347	0.0656	0.0000	0.0108	0.0548
disclose 10%, by I	0.0753	0.0000	0.0349	0.0404	0.0638	0.0000	0.0083	0.0555
disclose 10%, by value of feature	0.0762	0.0000	0.0356	0.0405	0.0639	0.0000	0.0085	0.0554
disclose 10%, by entropy	0.0755	0.0000	0.0342	0.0413	0.0627	0.0000	0.0071	0.0556
agent: $\text{HAD}_{1,1}$	0.0961	0.0000	0.0584	0.0377	0.0790	0.0003	0.0344	0.0443
agent: $\text{HAD}_{1,3}$	0.0943	0.0000	0.0566	0.0378	0.0772	0.0008	0.0336	0.0428
agent: team optimum	0.0934	0.0002	0.0558	0.0374	0.0759	0.0006	0.0334	0.0419

Table 5: Navigation, team regret $V_1^* - V_p$ on evaluation goals and its three terms (Theorem 5), means over 15 runs; the three terms add up to the regret to within $4e{-}16$ in every run. Disclosure budgets are fractions of the cells seen in training.

Predictor of a loss > 0.10	AUROC	AUPRC	Spearman
engine value of the position	0.552	0.177	0.04
volatility $v(1-v)$	0.705	0.262	0.48
spread of move values	0.601	0.196	0.15
best minus second-best move	0.778	0.340	0.43
number of near-best moves	0.787	0.351	0.42
expected loss of a uniform move	0.741	0.290	0.32
entropy of the partner model	0.627	0.239	0.28
partner-model probability of the best move	0.600	0.210	0.29
all eight, plus legal-move count (logistic)	0.871	0.576	–
$\widehat{\rho}_6$ alone, no fitting	**0.896**	0.595	0.61
all nine and $\widehat{\rho}_6$ (logistic)	**0.892**	**0.617**	–
gain over all nine, 95% interval	[0.018, 0.025]	[0.025, 0.058]	

Table 6: Predicting whether a human move loses more than 0.10 expected score (base rate 18%), before the move is seen; test half of the games. Single features are oriented on the fit half. The estimate alone beats a logistic model on all nine difficulty features, and adds to it.

F HUMAN GAMES: DATA, FEATURES AND FULL RESULTS

Data. Rated games from the Lichess open database (Lichess, 2025) for July 2014, read in file order. A game is kept when both players are inside one of the bands $[1000, 1250)$, $[1400, 1600)$ or $[1800, 2000)$, its estimated duration (base time plus 40 times the increment) lies in $[480, 1500)$ seconds, the game has at least 24 plies and it did not end on time; the first 350 games per band are used. Positions are sampled at every third ply from ply 10 to 70. Maia-2 was trained on Lichess games from 2013 to 2023, so these games are inside its training period: the analysis tests what the estimate adds over difficulty features, and a month after Maia-2's training data would be needed to test the partner model on unseen games.

Features and targets. One Stockfish search at depth 10 with as many principal variations as legal moves gives the value of every legal move for the side to move (legal moves missing from the output are evaluated separately). From these values: the value of the position (best move), the spread (best minus worst), the gap between the best and second-best move, the number of moves within 0.02 of the best, and the expected loss of a uniformly random legal move; from Maia-2 at the players' ratings: the entropy of its move distribution and the probability it gives the engine's best move; and the number of legal moves. The estimate $\widehat{\rho}_6$ sums Maia-2's raw probabilities times the engine loss over its six most likely moves; ρ over the full support is also recorded. The realised loss is the engine loss of the move played, from the same search; a random fifth of positions, chosen by a hash of game and ply, is re-scored with separate depth-16 searches of the position and of the position after the move. The event of interest is a loss above 0.10 expected score.

Protocol. Games are split in halves by a hash of their identifier. On the first half we orient each single feature (the sign of its rank correlation with the loss), fit a logistic model on the nine features (standardised, $C = 10$) with and without $\widehat{\rho}_6$, and a ridge regression for the expected loss. All reported numbers are on the second half. Intervals are percentiles of $1{,}000$ bootstrap resamples of test games (300 for the strata), so that positions from one game stay together.

Subset	test positions	nine features	$+\widehat{\rho}_6$	gain, 95% interval
rating 1000–1250	2950	0.897	0.911	[0.010, 0.019]
rating 1400–1600	2559	0.870	0.893	[0.018, 0.030]
rating 1800–2000	3096	0.846	0.875	[0.022, 0.037]
engine value < 0.3	2606	0.976	0.979	[0.001, 0.005]
engine value in $[0.3, 0.7]$	2543	0.787	0.817	[0.021, 0.039]
engine value > 0.7	3456	0.879	0.893	[0.010, 0.019]
ply < 30	3721	0.821	0.856	[0.027, 0.043]
ply 30–49	3089	0.894	0.906	[0.009, 0.016]
ply ≥ 50	1795	0.905	0.922	[0.012, 0.023]
deeper oracle (depth 16)	1702	0.865	0.882	[0.009, 0.026]

Table 7: Human games: AUROC of the nine-feature logistic model with and without $\widehat{\rho}_6$, by rating band, engine value and game phase, and with losses re-scored by a deeper separate search. The gain is positive in every subset; it is smallest where the side to move is already losing, where most moves lose little.

decile of $\widehat{\rho}_6$	1	2	3	4	5	6	7	8	9	10
1000–1250: mean $\widehat{\rho}_6$	0.000	0.002	0.016	0.044	0.083	0.147	0.327			
mean loss	0.001	0.013	0.034	0.089	0.126	0.197	0.381			
1400–1600: mean $\widehat{\rho}_6$	0.000	0.001	0.008	0.024	0.047	0.079	0.132	0.290		
mean loss	0.003	0.002	0.020	0.036	0.060	0.104	0.175	0.310		
1800–2000: mean $\widehat{\rho}_6$	0.000	0.002	0.009	0.020	0.035	0.057	0.090	0.145	0.291	
mean loss	0.000	0.002	0.013	0.027	0.047	0.060	0.085	0.123	0.226	

Table 8: Calibration of $\widehat{\rho}_6$ as a prediction of the expected loss, test half, by decile of $\widehat{\rho}_6$ within each band.

The certificate is an expected loss over a truncated partner model, so it is not a bound on what any single player loses; it predicts the average loss well in the middle band and under-predicts or over-predicts at the extremes (Table 8). By Proposition 21, under-prediction can come from the dropped tail (the six moves keep on average 0.92 of Maia-2's mass), from the model and from the oracle; the data do not separate the three.

G TABULAR CHECKS

Every statement in Sections 3 and 4 and Appendix C has a numerical check in the released script. On random MDPs with rewards in $[0, 1]$: Theorem 5 holds on 400 instances with residual below $3e{-}13$; the contrast bound of Proposition 7 is never violated on 5,000 random cells, both its sides are invariant to per-state offsets while the sup-norm bound is not, and its example is tight; both bounds of Theorem 12 hold on 300 instances with random oracle and partner-model errors; Proposition 20 holds on 3,000 cells; the instances of Theorem 8, its companion and Proposition 19 reproduce exactly, and the aliasing identities hold on 300 random parameter draws. For Theorem 5 and Propositions 9 and 10 a separate script (`tests_theory_handed.py`) checks, on 3,000 random cells and partitions, that the handed-over form of the information term equals the EVPI form (largest difference 2×10^{-16}), that refinement never raises it, that the two-state task of the proof separates every non-nested pair of partitions it draws, and that every refinement of the behavioural partition has an information term between 0 and that of β. With an exact model, the value iterates of Proposition 19(a) never increase, but the greedy policies lose team value from one correction to the next in $6 of 300$ random MDPs, so the counterexample is not isolated.

Depth and scale on random team MDPs. We repeat the sweep of $\text{HAD}_{\lambda,d}$ on 120 random MDPs (40 states, 5 actions, four successors per state-action, rewards uniform in $[0, 1]$, $\gamma = 0.9$, $p = \frac{1}{2}$); the partner plays a softmax (temperature 0.3) of cell-averaged action values on a random partition, so its decision term is positive; 118 instances have a non-negligible solo-to-team gap. With an exact partner model, one correction closes 86% of the gap to the team optimum, two 97% and three 99%. The theory's scale $\lambda = 1$ is within 1% of the best λ in 76% of instances at $d = 1$ and 97% at $d = 3$. Table 9 shows what partner-model error does: the deepest correction falls more than 1% below the best depth in 0 instances with an exact model and 37 at $\varepsilon = 0.4$, and $\lambda = 1$ is less often near-best. The greedy policies of an exact model are not monotone in d in $19 of 118$ instances, but the drops are

small. These are observations on one distribution of MDPs, not consequences of Theorem 12; the degradation sweep, in which the partner is mixed with a uniform policy while the agent keeps the nominal model, gives a rank correlation between the degradation level and the best scalar λ of -0.29, which Proposition 19(b) says can have either sign.

partner-model error ε	0	0.05	0.1	0.2	0.4
deepest correction $> 1\%$ below the best depth	0/118	5/118	10/118	17/118	37/118
$\lambda = 1$ within 1% of the best λ, $d = 1$	76%	77%	71%	66%	58%
$\lambda = 1$ within 1% of the best λ, $d = 3$	97%	97%	96%	81%	69%

Table 9: Random team MDPs, $\text{HAD}_{\lambda,d}$ with a partner model mixed with a uniform policy at weight ε. First row: instances in which $d = 15$ corrections end more than 1% of the instance's range below the best $d \in \{0, 1, 2, 3, 5, 8, 15\}$ ($\lambda = 1$). Other rows: fraction of instances in which $\lambda = 1$ is within 1% of the best $\lambda \in \{0, 0.25, \ldots, 3\}$.

H Compute

The chess runs used CPU nodes of a university cluster (16 cores and 32 GB per job, one Stockfish thread per worker); the 800-pair primary comparisons take between a few minutes and half an hour per 400 pairs, and all revision runs together took roughly 150 core-hours, dominated by the Stockfish searches inside the partner-aware rules and the solo runs. The navigation runs (8 cores per map and seed) take a few minutes each. The human-game features take about 25 minutes on an 8-core laptop, and the tabular sweeps and theory checks a few minutes. Maia-2 inference is a negligible share of every budget.

I Additional related work

Human–AI complementarity. That the most accurate model need not be the best teammate has been documented in advice-giving settings (Bansal et al., 2021; 2019), modelled Bayesianly (Steyvers et al., 2022), and analysed for fairness (Donahue et al., 2022); Wilder et al. (2020) train a model explicitly to complement a fixed human. The *learning to defer* line asks which single predictions to route to an expert (Madras et al., 2018; Raghu et al., 2019; Mozannar & Sontag, 2020; Verma & Nalisnick, 2022; Mozannar et al., 2023). All of these are one-shot: a prediction is either made or deferred, and the cost of a wrong routing is a misclassification. Handoff regret is the sequential analogue, where what is handed over is a *state* that the partner must continue to act in, and Theorem 5 accumulates the cost along the trajectory.

Cooperative multi-agent settings. When both agents optimise, the problem is a decentralised MDP (Bernstein et al., 2002; Oliehoek & Amato, 2016) and the difficulty becomes convention selection. Ad hoc teamwork (Stone et al., 2010; Mirsky et al., 2022), Hanabi (Bard et al., 2020), other-play and off-belief learning (Hu et al., 2020; 2021), trajectory diversity (Lupu et al., 2021), fictitious co-play (Strouse et al., 2021) and robustness evaluation in Overcooked (Carroll et al., 2019; Knott et al., 2021) all learn a policy that is good against a population of partners. We hold the partner fixed and non-adaptive, which is what lets the team backup be a single contraction (Lemma 2); the price is that a partner who learns from the agent is outside the model (Section 3).

Assistance, legibility and physical handover. Assistance games and pragmatic–pedagogic alignment (Hadfield-Menell et al., 2016; Fisac et al., 2020), shared autonomy (Javdani et al., 2015; Reddy et al., 2018) and legible motion (Dragan et al., 2013; Dragan & Srinivasa, 2013) concern an agent that infers or signals *intent*; teaching by demonstration (Ho et al., 2016) and machine teaching (Zhu, 2015) concern changing the partner. In robotics the handover of a physical object is its own literature (Strabala et al., 2013; Ortenzi et al., 2021), and Gavryushin et al. (2026) make collaborative handover prediction a perception benchmark. Our quantity is decision-theoretic rather than perceptual: it prices the state handed over, not the act of handing it.

World models and state abstraction. Probing sequence models for board state (Toshniwal et al., 2022; Li et al., 2023; Nanda et al., 2023; Karvonen, 2024) and the Myhill–Nerode framing of Vafa et al. (2024) ask what state a model has; Richens & Everitt (2024) show robust agents must learn

causal world models; Ha & Schmidhuber (2018) learn them explicitly. The abstraction literature (Givan et al., 2003; Ravindran & Barto, 2003; Li et al., 2006; Ferns et al., 2004; Castro, 2020; Abel et al., 2016) bounds the value lost by an agent that plans in a quotient. Our information term (Theorem 5) is the same kind of object placed on the *other* side of the table: it is the partner's partition, and the team pays for it only at handoffs.

Value of information and robustness to a wrong model. Expected value of perfect information is classical (Raiffa & Schlaifer, 1961; Howard, 1966) and underlies metareasoning (Russell & Wefald, 1991; Hay et al., 2012); the information term of Theorem 5 is EVPI evaluated on the partner's cell. The partner-model error term of Theorem 12 is a simulation-lemma argument (Kearns & Singh, 2002) for a perturbed operator, and its bound is of the form studied in robust dynamic programming (Iyengar, 2005); the policy-improvement step is standard (Bertsekas & Tsitsiklis, 1996; Puterman, 1994).

Chess as a model system. Superhuman play (Campbell et al., 2002; Silver et al., 2018) coexists with a line of work modelling *human* play at its own level (Anderson et al., 2016; McIlroy-Young et al., 2020; 2022; Tang et al., 2024; Zhang et al., 2025), which is what makes tag-team chess (Hamade et al., 2024) a controlled instance of the handoff problem. The talk that motivated this paper (Kleinberg, 2025) also raises algorithmic monoculture (Kleinberg & Raghavan, 2021) and generation in the limit (Kleinberg & Mullainathan, 2024; Kalavasis et al., 2025); we have not pursued the connection between a monoculture of partner models and the degradation identity of Proposition 19, which seems worth doing.

J On the epigraph

Polanyi's sentence opens his account of tacit knowledge, the skills and recognitions a person exercises but cannot put into words (Polanyi, 1966). It describes our setting twice. First, the agent and its partner do not communicate, so whatever the agent knows reaches the partner only through the state it hands over: a strong agent knows more than it can tell. The information term of Theorem 5 measures this surplus in the partner's cells, the value of what the agent can see and the partner cannot, and disclosure, the intervention that lowers it in Section 5.2, is a way of telling. Second, Proposition 10 turns the sentence around: from a partner's behaviour one cannot tell how much it knows. A partner that lacks a distinction and one that has it but does not act on it behave alike, and only an intervention separates them. Our chess partners are models of human play, whose skill is largely of the kind Polanyi described; the handed-over loss asks not what such a partner could say about a position but what it will do with the position it receives.