PHYSICALLY GROUNDED, EXPERIENCE-GUIDED LLM AGENTS FOR DECENTRALIZED GRID CONTROL

Anonymous authors
Paper under double-blind review

ABSTRACT

Power grids are safety-critical infrastructure that must maintain reliable operation under changing demand and increasing shares of variable renewable generation. Topology control can relieve transmission congestion by reconfiguring substation busbars, but decentralized operation requires coordinating interventions across regions with only partial network information. Existing approaches lack a mechanism that combines assessment of the current physical state, coordination of interdependent regional actions, and evidence from past interventions within a single decentralized decision process. We propose GRIDREFINE, a physically grounded, experience-guided framework for decentralized topology control using large language models as regional agents. Each agent uses a graph representation of its local network, physical knowledge, and a memory of past intervention outcomes from offline simulations to select and explain actions. Through counterfactual reasoning, agents consider how differences between past and current grid states may change an intervention's effects, then refine their assessment of promising candidates using feedback from a local power-flow tool. Because these interventions can redistribute power flows across regional boundaries, neighbouring agents coordinate their proposals before execution. We evaluate GRIDREFINE on two power grids of different sizes and include planned line disconnections for maintenance on the larger grid. With two of the three evaluated LLM backbones, GRIDREFINE improves mean grid survival over MARL baseline across all three settings. Code will be released upon publication.

1 INTRODUCTION

The transition to low-carbon electricity requires power grids to accommodate growing shares of renewable generation without compromising reliability. Transmission system operators (TSOs) operate high-voltage networks and must continuously balance generation and demand while keeping power flows within operating limits. As solar and wind supply a larger share of electricity, their weather-dependent output increases variability and uncertainty in grid operation (Marot et al., 2022). Meeting this challenge requires control policies that adapt to changing conditions and coordinate decisions across interconnected regions. Decentralized control offers a way to distribute those decisions while accounting for their effects on the wider grid.

TSOs can relieve transmission congestion by coordinating adjustments to generation or reconfiguring electrical connections within the network. The latter approach, known as topology control, reconfigures connections at substations to redistribute power flows and relieve overloaded transmission lines. Choosing an effective intervention is difficult, as the number of possible configurations grows rapidly, and a change that relieves one line can increase loading elsewhere (Marchesini et al., 2025). As generation and demand evolve, a configuration that is safe now may lead to overloads later. Topology control is therefore a combinatorial and sequential decision problem. Each intervention must mitigate current overloads while preserving reliable operation over time.

Reinforcement learning (RL) provides a way to learn such sequential decisions through interaction with simulated grids. The Learning to Run a Power Network competitions have demonstrated its potential (Marot et al., 2021), and multi-agent approaches distribute decisions among cooperating regional controllers. MARL2Grid-TR (Marchesini et al., 2026) provides a benchmark for studying control under regional observations and strict physical constraints. Because each agent observes only

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.

part of a physically coupled network, an intervention that relieves a local overload can shift stress into a neighbouring region or lose its effect when a neighbour acts at the same time. Multi-agent RL methods such as MAPPO learn implicitly coordinated policies through a shared objective and centralized information during training (Yu et al., 2022). During decentralized execution, however, each agent selects its action from its own observation without knowing which actions its neighbours will take at the same step. Standard MAPPO therefore provides no explicit coordination between regional interventions during grid operation.

Large language model (LLM) agents offer mechanisms for reasoning over environmental feedback and reusing past experience. ReAct combines language-based reasoning with environmental interaction (Yao et al., 2023). In power grid control, ACE uses LLMs to revise actions and evaluates them in simulation to guide RL training (Wan et al., 2025) . Memory-based approaches further allow previous experience to inform subsequent decisions without updating model weights (Shinn et al., 2023; Ouyang et al., 2026). Applying these mechanisms to decentralized topology control requires assessing whether past intervention outcomes remain relevant under current conditions. Changes in injections or topology can alter an action's effect, while neighbouring interventions can change the connections on which it depends. Furthermore, a failed trajectory alone does not establish whether another intervention at an earlier state would have improved survival. The remaining challenge is to combine relevant experience with physical assessment and coordination, enabling agents to evaluate the joint consequences of regional interventions before execution.

We propose GRIDREFINE, a physically grounded, experience-guided framework for decentralized topology control that combines regional LLM reasoning with explicit coordination. Regional LLM agents use physical knowledge and a memory of past intervention outcomes, collected offline in simulation, to reason counterfactually about candidate actions. By comparing past and current grid states, they assess how an intervention's effects may change and refine this assessment using feedback from a local power-flow tool. Nevertheless, a unilateral action on a stressed shared line can conflict with a neighbouring intervention. In such cases, neighbouring agents coordinate over review rounds, in which each agent assesses how the neighbouring proposal affects its own region and reconsiders its action. A joint action is executed only when all agents involved accept it. GRIDREFINE thus couples regional action selection with explicit review of neighbouring interventions under partial network observations. We evaluate GRIDREFINE with LLM backbones of different sizes and reasoning capabilities on two grids, including planned line disconnections on the larger grid to reflect scheduled maintenance in real-world operation. While the smallest evaluated model shows limited effectiveness, larger models with stronger reasoning capabilities outperform the MARL baseline across all three settings, improving mean survival by at least 28.1 percentage points. Further ablations highlight the contribution of regional coordination, while experiments on memory construction suggest that collecting experience around states preceding the agents' own failures can further improve performance without updating model weights.

2 RELATED WORK

Reinforcement learning for grid control. The L2RPN competitions established simulation-based evaluation of learned grid controllers (Marot et al., 2021), and RL2Grid standardizes single-agent tasks under operational constraints (Marchesini et al., 2025). LLMs have also been used to guide grid-control policy learning. ACE combines LLM-generated action refinements evaluated in simulation with reward shaping during RL training, while deploying the learned RL policy for control (Wan et al., 2025). Multi-agent approaches distribute decisions in different ways. In centrally coordinated MARL, regional agents propose actions and a coordinator uses global state information to select the action to execute (de Mol et al., 2025). This approach decomposes the action space while retaining central control. MARL2Grid-TR instead evaluates cooperating regional agents whose actions are executed jointly under partial observations (Marchesini et al., 2026). Coordination in these approaches is either centralized or implicit in learned policies. These approaches therefore lack explicit coordination of simultaneous regional interventions under partial network observations.

Grounded decision-making with LLM agents. An LLM's own reasoning is not evidence about the environment. Stated rationales can misrepresent what determines an answer (Turpin et al., 2023), and without external feedback models correct their reasoning unreliably (Huang et al., 2024). LLM agents therefore ground decisions in observations returned by the environment (Yao et al., 2023), in

tool and simulator calls that supply quantities the model cannot compute itself (Schick et al., 2023; Liu et al., 2023), and in critics and verifiers that revise or reject a proposal before it is acted on (Gou et al., 2024; Kambhampati et al., 2024). When several agents act on one system, their interactions require the same grounding. Cooperating agents exchange observations and intentions under decentralized control (Zhang et al., 2024), submit negotiated plans to a motion planner that checks them for feasibility (Mandi et al., 2024), and control traffic signals from local observations (Lai et al., 2025), more recently bringing neighbouring intersections into the reasoning (Yuan et al., 2026). A remaining gap is reconciling local physical estimates with historical outcomes that may no longer apply under current operating conditions.

Memory and self-improvement in LLM agents. Memory enables LLM agents to retain information across interactions, with existing work addressing its organization, retrieval and updating (Packer et al., 2023; Xu et al., 2025). Beyond storage, using past information requires selecting context relevant to the current query. This has been studied through learned retrieval and joint selection of in-context examples (Rubin et al., 2022; Ye et al., 2023), as well as trajectory retrieval for computer control (Zheng et al., 2024). A complementary research direction examines how agents can transform interaction experience into reusable knowledge without updating model weights. In web environments, this includes constructing reusable workflows and environment-specific knowledge (Wang et al., 2025; Liu et al., 2025). Across reasoning, web and embodied tasks, other approaches distill experience into verbal reflections, reusable insights and context-aware guidelines (Shinn et al., 2023; Zhao et al., 2024; Fu et al., 2024). More recent systems maintain transferable reasoning strategies and evolving playbooks that guide subsequent tasks (Ouyang et al., 2026; Zhang et al., 2026). For decentralized topology control, a remaining question is how to build memory around the states regional controllers encounter and reuse intervention outcomes under changing operating conditions.

3 PRELIMINARIES

We study topology control in the decentralized setting of MARL2Grid-TR (Marchesini et al., 2026), which is built on the Grid2Op simulator. This section first describes the electrical problem and then the information and actions available to each regional agent.

3.1 ELECTRICAL NETWORK AND TOPOLOGY CONTROL

A transmission grid connects generators and loads through substations joined by lines. Each substation has two busbars. Every line end, generator, and load at a substation connects to one of them. Terminals on the same busbar are electrically joined, and terminals on different busbars are not directly connected within the substation. Power flows through the resulting graph according to the injections and the line impedances. Operators can influence the connection graph through topology control. Topology optimization reconfigures grid connectivity to relieve congestion and reduce the risk of equipment failure. The available actions move terminals between the two busbars of a substation or switch a line on or off. Their effects are hard to anticipate. Splitting a busbar redirects power along the routes that remain, and relief on one line can overload another. The number of options also grows quickly, since a substation with n terminals admits up to 2^{n-1} busbar configurations.

The resulting flows must be assessed against each line's thermal limit. We measure the loading of line ℓ at step t as $\rho_{\ell,t} = |I_{\ell,t}|/\bar{I}_\ell$, the ratio of current magnitude to thermal rating, so that $\rho_{\ell,t} > 1$ marks an overload. Protection enforces this limit to safeguard equipment. A line that stays overloaded for more than a few consecutive steps is disconnected, and a line loaded beyond a hard threshold is disconnected at once. Its flow then moves onto the remaining lines, which may overload in turn and lead to cascading failures. Operating rules add further constraints. A switched substation or line can enter a cooldown, a line disconnected by protection can be reconnected only after a delay, and planned maintenance removes lines from service at scheduled times.

A scenario is a time series of generation and demand that covers four weeks at five-minute resolution. At step t, the grid has state s_t and N agents apply a joint action $\mathbf{a}_t = (a_t^1, \ldots, a_t^N)$. The environment evolves according to $s_{t+1} \sim T(\cdot \mid s_t, \mathbf{a}_t)$, where the transition applies the actions, advances generation, demand, cooldowns, and maintenance, solves the AC power flow, and applies protection. An episode ends early when the power flow cannot be solved or the grid can no longer

supply its loads, for example after islanding. The agents share a single objective, which is to keep the grid operating until the end of the scenario.

3.2 DECENTRALIZED CONTROL SETTING

MARL2Grid-TR partitions each grid into N regions with strong internal connectivity and few external links, mirroring how operators divide control areas. Agent i controls the substations of region i. A line whose two ends lie in different regions is a shared line observed by both adjacent agents. The resulting problem is a decentralized partially observable Markov decision process. All agents pursue the shared objective, but each receives only a regional observation from the environment.

Agent i receives the observation $o_t^i = \phi_i(s_t)$, whose features are listed in Table 4 in Appendix A. It covers the busbar assignment of every terminal in region i, the injections of its generators and loads, and measurements on its internal and shared lines. Signed active-power flows on shared lines are measured at the local end. Line loading, status and counters are common to both adjacent agents. Neighbouring busbar configurations are hidden in this local observation.

The action catalog $\mathcal{A}_i$ contains terminal reassignments at one substation of region i and status changes for one internal line. The agent can also leave its connections unchanged. An action is admissible at step t when it satisfies the operating rules, including cooldown constraints.

4 METHOD

We introduce GRIDREFINE, a decentralized LLM framework for topology control (Figure 1). Each agent grounds its decisions in a regional graph representation, physical knowledge of line impedances and power flows, and historical experience. By retrieving grid states and their evaluated interventions, the agent identifies candidate actions and checks their expected effects in the current state (Section 4.1). Since a neighbouring agent action can change the flows affected by a proposed intervention, agents exchange their proposed actions before execution, evaluate their combined effect on shared lines, and act only when the participating agents agree (Section 4.2). Nonetheless, past experience cannot cover every operating state. GRIDREFINE uses offline memory refinement to revisit states preceding failed episodes, simulate alternative interventions, and add the resulting outcomes to regional memory. This expands the evidence available for subsequent decisions using outcomes generated by the agents own control strategy(Section 4.3).

4.1 PHYSICALLY GROUNDED COUNTERFACTUAL REASONING

When loading on an internal or shared line reaches an activation threshold ρ_{act}, an agent compares admissible regional actions with leaving the topology unchanged. The comparison is challenging under partial observability: a reconfiguration that appears to relieve local loading may redistribute power in ways the agent cannot infer from its regional observation alone. GRIDREFINE retrieves relevant cases from a regional memory of past interventions and their outcomes over extended evaluation windows. For retrieval, it represents the observation o_t^i as a regional graph of busbar groups, line connections, injections, power flow directions, line impedances, and thermal headroom. The agent prioritizes memory cases with matching busbar connectivity and line status before ranking by operating similarity, obtaining a shortlist of candidate actions alongside both detailed state-action records and aggregate outcome summaries across past evaluations.

Notwithstanding, observational similarity does not guarantee that an intervention will reproduce its historical outcome. For instance, a busbar reconfiguration that successfully relieves a stressed line by rerouting power through an alternative corridor may cause an overload if that receiving corridor currently lacks thermal headroom. While a favorable historical outcome identifies a plausible relief mechanism, an adverse outcome highlights conditions under which that mechanism transfers risk elsewhere. Changes in topology can also alter the underlying physical operation, causing the same switching action to join connections in one state while separating them in another. The agent therefore evaluates the applicability of each retrieved case based on differences between the past and current states, independently of whether the historical outcome was favorable or adverse.

These applicability assessments are applied systematically across all shortlisted candidates to compare competing relief mechanisms. For each action, the agent identifies the expected topological

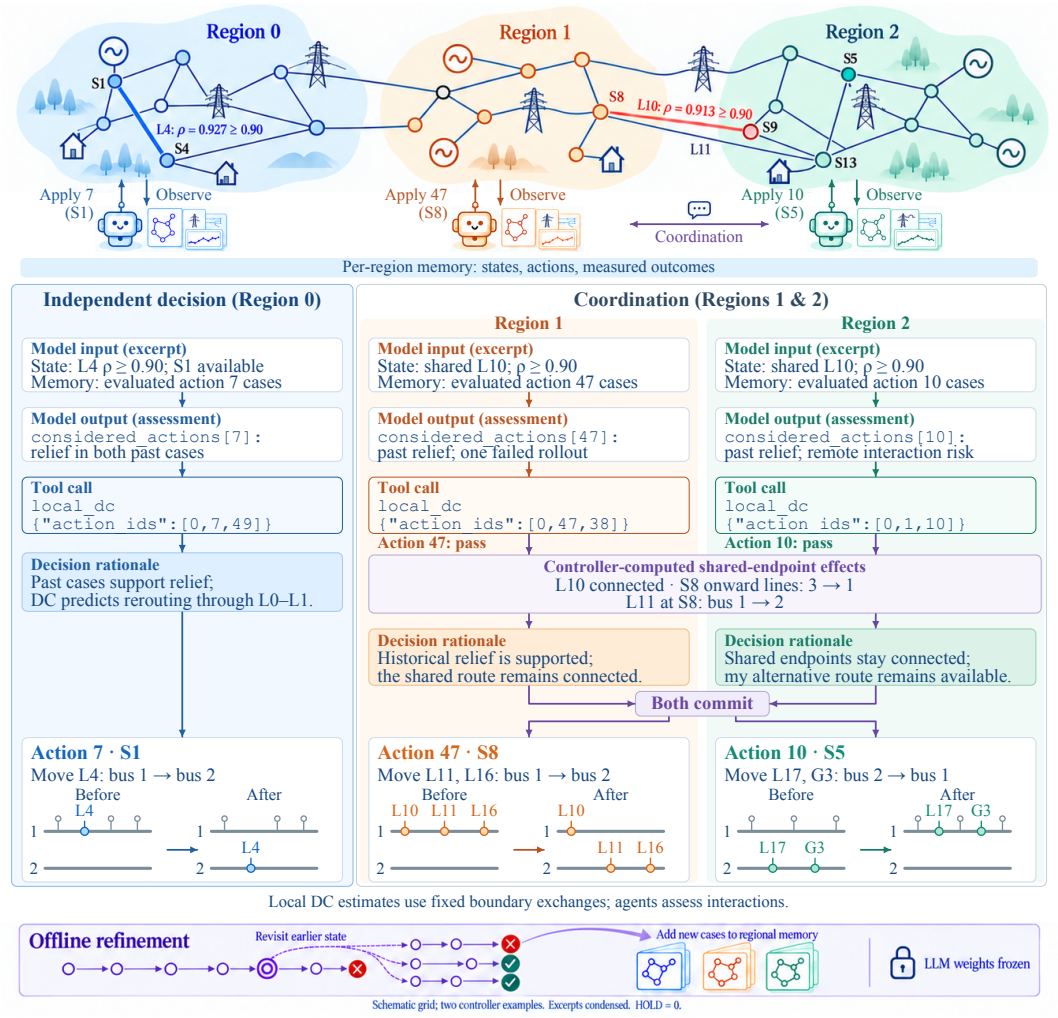


Figure 1: Overview of GRIDREFINE. Regional agents use historical outcomes and local physical calculations to select topology actions, coordinating when neighbouring proposals interact. Two controller examples illustrate independent and coordinated decisions and their busbar reconfigurations. Offline refinement adds counterfactual experience to regional memories.

redistribution and balances its historical support against recorded failures and coverage gaps. Candidate evaluation explicitly balances reducing peak line stress against preserving headroom elsewhere, maintaining supply paths, and sustaining longer grid survival. The comparison prioritizes preventing immediate protection trips and ensuring short-term survival, followed by maximizing supported survival time, minimizing line disconnections, and lowering overall loading. When competing actions yield comparable historical support, the agent penalizes unnecessary switching to preserve future operational flexibility. Figure 2 illustrates how an adverse outcome in a comparable state can support leaving connections unchanged.

Because historical evidence reflects past conditions, the agent grounds its choice in current physics by executing a local DC power-flow calculation on the top-ranked candidates. Holding local injections and boundary exchanges fixed, this calculation estimates the incremental active-power redistribution caused by each reconfiguration. When a historical outcome conflicts with local DC estimates, such as when past cases suggest relief but the DC calculation reveals an overloaded receiving line, the agent must explicitly reconcile the disagreement. The agent then selects the final regional candidate through a structured rationale that cites historical evidence, addresses physical contradictions, and distinguishes past measured outcomes from current DC estimates. If neighboring agents are active, this local selection serves as the proposal for joint review (Section 4.2).

4.2 COMMUNICATION AND COORDINATION

When neighboring agents act simultaneously, an independently plausible regional action may fail if it relies on a shared line that another agent reconfigures. To prevent harmful interference, GRIDREFINE triggers a proposal review whenever a shared line becomes stressed. Paired regions exchange messages across all shared connections, transmitting structural descriptions of proposed endpoint changes, including connection status, busbar assignments, and the number of connected line, load, and generator terminals. Derived directly from local topology changes, these descriptions reveal whether a shared line maintains an onward path into the adjacent region, supplying the information required to evaluate cross-regional relief without exposing private regional states.

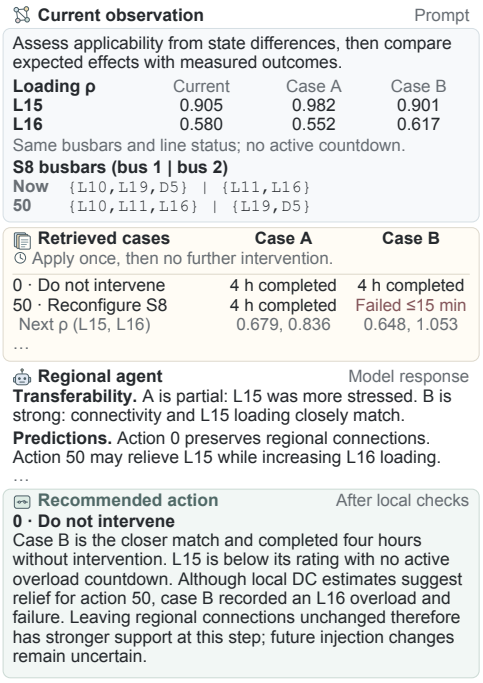


Figure 2: A historical failure under similar conditions supports leaving connections unchanged despite favorable local DC estimates.

During review, each agent combines shared-endpoint descriptions with local DC estimates of its own intervention to assess whether the neighbouring proposal preserves the structural conditions on which its relief mechanism depends. Because agents do not perform joint power-flow simulations, this assessment evaluates structural compatibility alongside regional evidence. An agent can commit to the combined intervention, revise its proposal, or raise an explicit objection identifying a structural conflict on a shared line. An objection leaves the agent's proposal unchanged while prompting the neighbor to reconsider its action, which initiates a renewed review round. To avoid endless coordination loops, execution requires unanimous commitment to a legal joint proposal.

4.3 OFFLINE MEMORY REFINEMENT

The state distribution encountered by an agent depends directly on its control strategy. Consequently, memories constructed solely through offline greedy search on training scenarios may provide limited evidence for the specific operational states reached under the controller's own decisions. GRIDREFINE addresses this coverage gap through offline memory refinement, evaluating counterfactual interventions at states preceding physical grid failures to expand memory around the controller's trajectory. While initial memories built via greedy search supply a baseline of evaluated topology actions, refining memory around failure modes builds targeted comparative evidence for future decision-making without finetuning the model weights.

For trajectories ending in grid failure, we isolate the final continuous period of grid stress, identified by elevated line loading or active overload counters, and replay recorded actions to identify its decision states. We evaluate counterfactual alternatives both along the observed failure trajectory and along a search-driven alternative continuation branching from the onset of stress. Replaying the failure trajectory preserves evaluations at decision states the controller visited before protection removals or prior reconfigurations constrained remaining choices. Concurrently, the alternative search continuation explores how alternative interventions alter subsequent trajectory evolution, collecting decision cases until sustained calm is restored, the grid fails, or the scenario horizon is reached.

The resulting state-action-outcome records are incorporated into regional memories. This self-improvement process iterates across training stages: agents execute decisions using accumulated memory, terminal failures guide counterfactual mining, and updated memories inform subsequent control. Memory updates are restricted strictly to training scenarios, ensuring that validation evaluates performance under fixed empirical knowledge. As the model weights remain frozen, the LLM agent learns entirely through the expansion and refinement of its grounded empirical experience.

5 EVALUATION AND RESULTS

5.1 EXPERIMENTAL SETUP

Environments. We evaluate GRIDREFINE on bus14 and bus36 grid to examine control across different network sizes and coordination demands. Bus14 grid contains 14 substations and 20 transmission lines divided among three control regions, while Bus36 grid contains 36 substations and 59 lines across four regions. We additionally evaluate Bus36 grid with planned line disconnections for maintenance, reflecting operating conditions in which some transmission lines are temporarily unavailable. This setting tests the controller's ability to maintain operation as the available transmission paths change. All methods share the same splits and regional action catalogs(Appendix A).

Baselines and Model backbones. We compare against the following baselines: Do-Nothing takes no corrective action, providing a reference for assessing whether topology interventions improve survival. MAPPO (Yu et al., 2022) is a multi-agent RL method that learns regional control policies through multi-agent reinforcement learning. We select it as our RL baseline based on its strong performance in MARL2Grid-TR, where it outperformed QPLEX (Wang et al., 2021) and fully observable single-agent PPO (Schulman et al., 2017) on Grid14 topology control (Marchesini et al., 2026). The retrieval controller uses the same frozen memory, candidate retrieval procedure, and local DC power-flow tool as GRIDREFINE, but selects actions using a fixed ranking rule based on recorded outcomes and current DC estimates, without an LLM. We compare three model families at different scales to examine how backbone capabilities affect grid control. Gemma 4 E4B-IT (Team et al., 2026) is a dense model with 4.5B effective parameters, selected to assess whether a small model with reasoning and tool-use capabilities can support regional control. Qwen3.8-27B (Qwen Team, 2026) is a dense model with 27B parameters, selected for its strong reported reasoning and instruction-following performance at this scale. DeepSeek-V4-Flash (DeepSeek-AI et al., 2026) combines strong reported agent performance with a mixture-of-experts architecture containing 284B total parameters and 13B active per token. Its inclusion extends the comparison to a larger, sparsely activated backbone, while its thinking modes allow us to assess the benefit of additional reasoning.

Metrics. We report average survival, defined as the percentage of each chronic's full horizon reached, and the number of completed chronics. We report the mean and sample standard deviation of survival across three seed/split runs, together with the mean number of completed chronics.

5.2 RESULTS

GRIDREFINE improves grid survival over MAPPO across both grid sizes and with planned line disconnections (Table 1). With Qwen, survival increases from 52.4% to 80.5% on bus14 grid and from 27.6% to 61.4% on Grid36 with planned line disconnections. MAPPO falls below Do-Nothing in both bus36 grid settings, illustrating the difficulty of selecting and coordinating regional interventions that improve operation. With planned line disconnections, Qwen also increases the mean number of completed chronics from 2.3 to 43.0.

Historical experience alone does not ensure effective control. With the same memory, Qwen raises bus14 grid survival from 37.8% for retrieval alone to 80.5%, with gains in both Grid36 settings. This supports assessing historical outcomes alongside current physical conditions and neighbouring proposals. Gemma, however, reaches 60.2% on bus36 grid, below retrieval and Do-Nothing. Its observed difficulties in following controller instructions suggest that the task requires strong reasoning and reliable instruction following, even with historical evidence and physical tools. Qwen and DeepSeek achieve nearly identical bus36 grid survival without thinking, despite their different sizes and architectures, while DeepSeek performs better on bus 14 and on bus36 grid with planned line disconnections. Thinking raises DeepSeek survival from 78.5% to 84.3% on bus36 grid and from 65.4% to 69.9% with planned line disconnections, but adds little on bus14 grid. These gains require more than twice as many generated tokens per response. The benefit of additional inference therefore depends on the setting, while both backbones outperform MAPPO with thinking disabled.

Under maintenance, the higher variance with planned line disconnections is consistent with the difficulty of controlling a changing network. Lines temporarily taken out of service remove transmission paths, while earlier interventions shape the grid state and the available reconfigurations. Survival can

therefore depend strongly on how successive actions preserve the ability to redirect power as lines become unavailable.

Table 1: Test-set performance over three seed/split runs. Survival is mean ± sample standard deviation; Comp. is the mean number of completed chronics. Δ denotes the survival gain over MAPPO in percentage points, with positive gains highlighted in teal. Bold marks the best reported result.

	Bus14			**Bus36**			**Bus36 + planned disconnections**		
Controller	Survival (%)	Δ (pp)	Comp. (/200)	Survival (%)	Δ (pp)	Comp. (/156)	Survival (%)	Δ (pp)	Comp. (/156)
Baselines									
Do-Nothing	20.3 ± 0.8	-32.1	0.7	67.7 ± 2.4	$+23.0$	93.7	44.0 ± 16.9	$+16.4$	25.3
MAPPO	52.4 ± 6.1	0.0	38.3	44.7 ± 14.3	0.0	49.7	27.6 ± 21.1	0.0	2.3
GRIDREFINE variants									
Retrieval (no LLM)	37.8 ± 1.8	-14.6	23.7	67.5 ± 1.9	$+22.8$	86.3	50.7 ± 13.7	$+23.1$	31.7
Gemma (4.5B)	22.9 ± 2.5	-29.5	2.7	60.2 ± 2.3	$+15.5$	81.3	38.7 ± 16.0	$+11.1$	19.0
Qwen (27B)	80.5 ± 0.5	$+28.1$	128.0	78.4 ± 3.4	$+33.7$	100.0	61.4 ± 12.4	$+33.8$	43.0
DeepSeek (284B)	89.5 ± 0.8	$+37.1$	155.0	78.5 ± 3.5	$+33.8$	99.3	65.4 ± 11.3	$+37.8$	49.7
+ thinking	$\mathbf{89.7} \pm 1.1$	$\mathbf{+37.3}$	**155.7**	$\mathbf{84.3} \pm 5.3$	$\mathbf{+39.6}$	**112.0**	$\mathbf{69.9} \pm 10.6$	$\mathbf{+42.3}$	**61.0**

Memory refinement. Progressive mining improves validation survival over greedy-only memory in all three settings (Figure 3). Both memory construction procedures use the same training scenarios and greedy action-search procedure. They differ in the states selected for evaluation: greedy-only memory collects intervention outcomes along the greedy controller's trajectories, while refinement additionally targets states preceding failures of the memory-guided controller. This focuses memory expansion on situations in which the controller's existing evidence and decisions proved insufficient.

On bus14, the final memory raises survival from 83.10% to 89.42%, while survival on bus36 increases from 81.00% to 84.41% without planned line disconnections and from 59.46% to 62.59% with them. Since both memories use the same training scenarios, these results support refining experience around states encountered by the controller and evaluating alternatives to its failed decisions. This enables experience-driven, in-context adaptation by improving the evidence available for subsequent decisions while keeping the model weights frozen. However, the gains are not necessarily monotonic. Bus14 survival reaches 93.86% in round 4 before decreasing to 89.42% in round 6, still remaining above the greedy-memory baseline.

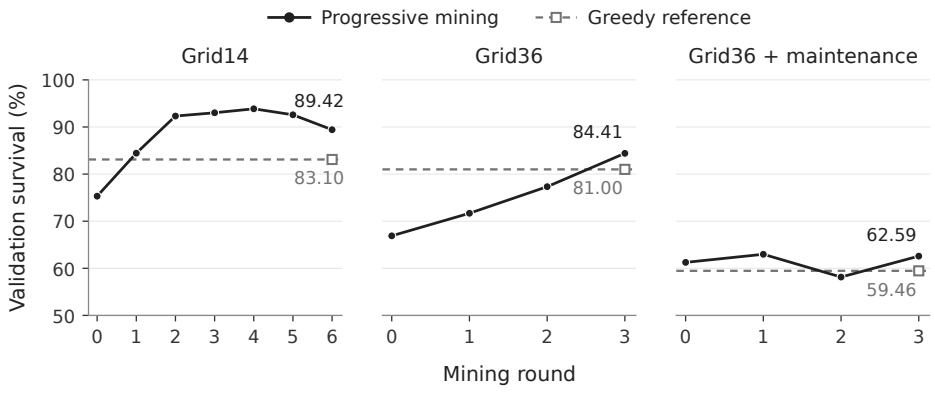


Figure 3: Mean validation survival with Qwen across three environments (seed 0). Progressive mining starts with greedy memory from 100 training chronics. Each subsequent round runs the controller on 100 additional training chronics and expands memory through counterfactual evaluations around its failures. Dashed lines show performance with greedy memory using all training chronics.

Table 2: Qwen validation survival (%) over the same 20 episodes per grid (seed 0, thinking disabled).

Controller variant	Bus14	Bus36	Bus36 + planned disconnections
LLM only	14.78	54.58	24.95
+ Memory	61.23	67.07	60.02
+ Local DC feedback	68.72	75.07	58.74
+ Coordination (full)	83.10	81.00	59.46
+ Self-improvement (mining)	89.42	84.41	62.59

Ablations. The ablations highlight the importance of coordination for decentralized control (Table 2). Regional agents using the full framework without coordination achieve 68.72% survival on bus14. Coordinating their proposals raises survival to 83.10% and improves bus36 survival from 75.07% to 81.00%. These gains show the value of considering neighbouring interventions when selecting a regional action. A locally promising reconfiguration may rely on connections that another agent changes. Coordination also improves survival with planned line disconnections. Furthermore, adding memory raises bus14 survival from 14.78% to 61.23%. Past successes and failures give agents evidence for selecting interventions in similar grid states. Local DC feedback improves survival on bus 14 and 36 grids, while slightly reducing on bus 36 with planned line disconnections. This suggests that model reasoning about power redistribution cannot reliably replace a physical calculation of an action's effects. Historical outcomes help identify promising interventions, while numerical feedback then helps assess their suitability under current conditions.

6 Discussion and limitations

The results highlight the value of combining past experience with explicit reasoning about alternative actions and their interactions. Historical outcomes help agents assess what an intervention might achieve and where it might fail. Counterfactual reasoning uses this evidence to compare possible changes with leaving the topology unchanged. Coordination extends that comparison across regional boundaries by allowing agents to reconsider proposals that depend on a neighbour's actions. These capabilities address a central difficulty of decentralized control.

A natural next step is to extract more general knowledge from this experience. The current memory stores evaluated interventions in specific grid states. An LLM could compare these records to identify recurring strategies and the conditions under which they succeed or fail. Simulation could then test whether these strategies remain useful across different operating conditions. This would extend memory beyond recalling similar cases towards identifying interventions from a more general understanding of their physical effects.

GridRefine expresses its assessments in plain text and explains its choices through historical evidence and physical estimates. This makes the reasoning behind a recommendation available for review alongside the proposed action. In TSO control rooms, operators could use these explanations to compare alternatives and question assumptions about neighbouring interventions. Future work should study how operators use this information under operational time constraints. Such collaboration could support earlier identification of harmful interactions and timely interventions to help prevent cascading failures. However, memory mining requires repeated LLM inference alongside counterfactual simulations. Additional thinking also increases the cost of control decisions. These costs raise the question of where additional computation is most useful. Selective mining could prioritize failures that expose gaps in the available experience. Consolidating records into reusable strategies could reduce repeated analysis at decision time. Finally, adaptive reasoning budgets could then concentrate additional computation on decisions that remain difficult to resolve.

7 Conclusion

We propose GridRefine, a physically grounded framework for decentralized topology control whose regional LLM agents combine past intervention outcomes, local power-flow estimates, and explicit coordination to select actions. With two of the three evaluated LLM backbones, GridRefine improves mean grid survival over a multi-agent reinforcement learning baseline across two grid sizes, including planned line disconnections on the larger grid. These findings open a pathway towards decision-support systems that combine LLM reasoning with physics-based tools and the expertise of human operators to guide decisions in safety-critical grid control.

ETHICS STATEMENT

All experiments are conducted in simulation using public grid benchmarks. Topology interventions can affect electricity supply beyond the region in which they are applied. Operational use of GRIDREFINE would therefore require independent validation and oversight by qualified operators. Its explanations provide evidence for review but do not establish that an intervention is safe. Evaluating how operators interpret and question these recommendations will be important to avoid misplaced confidence in automated decision support.

REPRODUCIBILITY STATEMENT

All experiments use publicly available Grid2Op benchmarks. Methods share the same data splits and regional action catalogs. The appendix describes memory construction and the controller protocol alongside the experimental settings and computational resources. We will release the source code and experiment configurations upon publication to support reproduction of the reported results.

AI USE STATEMENT

We used generative AI tools to assist with literature review, manuscript drafting, and language revision. These tools also supported checks of consistency between the method description and the implementation. The authors reviewed and revised the text and take responsibility for the final manuscript, including its scientific claims, references, and reported results.

REFERENCES

Barbera de Mol, Davide Barbieri, Jan Viebahn, and Davide Grossi. Centrally coordinated multi-agent reinforcement learning for power grid topology control. In *Proceedings of the 16th ACM International Conference on Future and Sustainable Energy Systems*. Association for Computing Machinery, 2025. doi: 10.1145/3679240.3734602. URL https://arxiv.org/abs/2502.08681v2.

DeepSeek-AI, Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chengyu Hou, Chenhao Xu, Chenze Shao, Chong Ruan, Conner Sun, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Donghao Li, Dongjie Ji, Erhang Li, Fang Wei, Fangyun Lin, Fangzhou Yuan, Feiyu Xia, Fucong Dai, Guangbo Hao, Guanting Chen, Guoai Cao, Guolai Meng, Guowei Li, Han Yu, Han Zhang, Hanwei Xu, Hao Li, Haofen Liang, Haoling Zhang, Haoming Luo, Haoran Wei, Haotian Yuan, Haowei Zhang, Haowen Luo, Haoyu Chen, Haozhe Ji, Hengqing Zhang, Honghui Ding, Hongxuan Tang, Huanqi Cao, Huazuo Gao, Hui Qu, Hui Zeng, J Yang, JQ Zhu, Jia Luo, Jia Song, Jia Yu, Jialiang Huang, Jialu Cai, Jian Liang, Jiangting Zhou, Jiasheng Ye, Jiashi Li, Jiaxin Xu, Jiewen Hu, Jieyu Yang, Jin Chen, Jin Yan, Jingchang Chen, Jingli Zhou, Jingting Xiang, Jingyang Yuan, Jingyuan Cheng, Jingzi Zhou, Jinhua Zhu, Jiping Yu, Joseph Sun, Jun Ran, Junguang Jiang, Junjie Qiu, Junlong Li, Junmin Zheng, Junxiao Song, Kai Dong, Kaige Gao, Kang Guan, Kexing Zhou, Kezhao Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Wang, Leyi Xia, Li Zhang, Liang Zhao, Lihua Guo, Lingxiao Luo, Linwang Ma, Linyan Zhu, Litong Wang, Liyu Cai, Liyue Zhang, Longhao Chen, MS Di, MY Xu, Max Mei, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Mingxu Zhou, Minmin Han, Ning Wang, Panpan Huang, Panpan Wang, Peixin Cong, Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qingyang Li, Qinyu Chen, Qiushi Du, Qiwei Jiang, Rui Tian, Ruifan Xu, Ruijie Lu, Ruiling Xu, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runqian Chen, Runqiu Yin, Runxin Xu, Ruomeng Shen, Ruoyu Zhang, Ruyi Chen, SH Liu, Shanghao Lu, Shangmian Sun, Shangyan Zhou, Shanhuang Chen, Shaofei Cai, Shaoheng Nie, Shaoqing Wu, Shaoyuan Chen, Shengding Hu, Shengyu Liu, Shiqiang Hu, Shirong Ma, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, Shuying Yu, Songyang Zhou, Tao Ni, Tao Yun, Tian Jin, Tian Pei, Tian Ye, Tianle Lin, Tianran Ji, Tianyi Cui, Tianyuan Yue, Tingting Yu, Tun Wang, W Zhang, WL Xiao, Wangding Zeng, Wei An, Weilin Zhao, Wen Liu, Wenfeng Liang, Wenjie Pang, Wenjing Luo, Wenjing Yao, Wenjun Gao, Wenkai Yang, Wenlve Huang,

Wenqing Hou, Wentao Zhang, Wenting Ma, Xi Gao, Xiang He, Xiangwen Wang, Xianzu Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaokang Zhang, Xiaotao Nie, Xiaowen Sun, Xiaoxiang Wang, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xingchen Liu, Xingkai Yu, Xingyou Li, Xinyu Yang, Xinyu Zhang, Xu Chen, Xuanyu Wang, Xuecheng Su, Xueyin Chen, Xuheng Lin, Xuwei Fu, YC Yan, YQ Wang, YW Ma, Yanfeng Luo, Yang Zhang, Yanhong Xu, Yanru Ma, Yanwen Huang, Yao Li, Yao Li, Yao Xu, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Qian, Yi Shao, Yi Yu, Yichao Zhang, Yifan Ding, Yifan Shi, Yijia Wu, Yiliang Xiong, Yiling Ma, Ying He, Ying Tang, Ying Zhou, Yingjia Luo, Yinmin Zhong, Yishi Piao, Yisong Wang, Yixiang Zhang, Yixiao Chen, Yixuan Tan, Yixuan Wei, Yiyang Ma, Yiyuan Liu, Yonglun Yang, Yongqiang Guo, Yongtong Wu, Yu Wu, YuKun Li, Yuan Cheng, Yuan Ou, Yuanfan Xu, Yuanhao Li, Yuduan Wang, Yuehan Yang, Yuer Xu, Yuhan Wu, Yuhao Meng, Yuheng Zou, Yukun Zha, Yunfan Xiong, Yupeng Chen, Yuping Lin, Yuqian Cao, Yuqian Wang, Yushun Zhang, Yuting Yan, Yutong Lin, Yuxian Gu, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuxuan Zhou, Yuyang Zhou, Yuzhen Huang, ZF Wu, Zehao Wang, Zehua Zhao, Zehui Ren, Zekai Zhang, Zhangli Sha, Zhe Fu, Zhe Ju, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zheren Gao, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong Shao, Zhixian Huang, Zhixuan Chen, Zhiyu Wu, Zhizhou Ren, Zhongyu Wu, Zhuoshu Li, Zhuping Zhang, Zian Xu, Zihao Wang, Zihua Qu, Zihui Gu, Zijia Zhu, Zilin Li, Zipeng Zhang, Ziwei Xie, Ziyi Gao, Ziyi Wan, Zizheng Pan, and Zongqing Yao. Deepseek-v4: Towards highly efficient million-token context intelligence, 2026. URL https://arxiv.org/abs/2606.19348.

Yao Fu, Dong-Ki Kim, Jaekyeom Kim, Sungryull Sohn, Lajanugen Logeswaran, Kyunghoon Bae, and Honglak Lee. AutoGuide: Automated generation and selection of context-aware guidelines for large language model agents. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-3811. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/d8efbb5dd415974eb095c3f06bff1f48-Abstract-Conference.html.

Zhibin Gou, Zhihong Shao, Yeyun Gong, Yelong Shen, Yujiu Yang, Nan Duan, and Weizhu Chen. CRITIC: Large language models can self-correct with tool-interactive critiquing. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/fef126561bbf9d4467dbb8d27334b8fe-Abstract-Conference.html.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. Large language models cannot self-correct reasoning yet. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/8b4add8b0aa8749d80a34ca5d941c355-Abstract-Conference.html.

Subbarao Kambhampati, Karthik Valmeekam, Lin Guan, Mudit Verma, Kaya Stechly, Siddhant Bhambri, Lucas Saldyt, and Anil Murthy. Position: LLMs can't plan, but can help planning in LLM-Modulo frameworks. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 22895–22907, 2024. URL https://proceedings.mlr.press/v235/kambhampati24a.html.

Siqi Lai, Zhao Xu, Weijia Zhang, Hao Liu, and Hui Xiong. LLMLight: Large language models as traffic signal control agents. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 2335–2346, 2025. doi: 10.1145/3690624.3709379.

Ruibo Liu, Jason Wei, Shixiang Shane Gu, Te-Yen Wu, Soroush Vosoughi, Claire Cui, Denny Zhou, and Andrew M. Dai. Mind's Eye: Grounded language model reasoning through simulation. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=4rXMRuoJlai.

Yitao Liu, Chenglei Si, Karthik R. Narasimhan, and Shunyu Yao. Contextual experience replay for self-improvement of language agents. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14179–14198. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.acl-long.694. URL https://aclanthology.org/2025.acl-long.694/.

Zhao Mandi, Shreeya Jain, and Shuran Song. RoCo: Dialectic multi-robot collaboration with large language models. In *IEEE International Conference on Robotics and Automation*, 2024. URL https://arxiv.org/abs/2307.04738.

Enrico Marchesini, Benjamin Donnot, Constance Crozier, Ian Dytham, Christian Merz, Lars Schewe, Nico Westerbeck, Cathy Wu, Antoine Marot, and Priya L. Donti. RL2Grid: Benchmarking reinforcement learning in power grid operations. *arXiv preprint arXiv:2503.23101*, 2025. URL https://arxiv.org/abs/2503.23101.

Enrico Marchesini, Eva Boguslawski, Alessandro Leite, Christopher Amato, Matthieu Dussartre, Marc Schoenauer, Benjamin Donnot, and Priya L. Donti. MARL2Grid-TR: A multi-agent RL benchmark in power grid operations. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=mpAMH1OyMO.

Antoine Marot, Benjamin Donnot, Gabriel Dulac-Arnold, Adrian Kelly, Aidan O'Sullivan, Jan Viebahn, Mariette Awad, Isabelle Guyon, Patrick Panciatici, and Camilo Romero. Learning to run a power network challenge: a retrospective analysis. In *Proceedings of the NeurIPS 2020 Competition and Demonstration Track*, volume 133 of *Proceedings of Machine Learning Research*, pp. 112–132, 2021. URL https://proceedings.mlr.press/v133/marot21a.html.

Antoine Marot, Alexandre Rozier, Matthieu Dussartre, Laure Crochepierre, and Benjamin Donnot. Towards an AI assistant for power grid operators. In *HHAI2022: Augmenting Human Intellect*, pp. 79–95. IOS Press, 2022. doi: 10.3233/FAIA220191. URL https://arxiv.org/abs/2012.02026.

Siru Ouyang, Jun Yan, I-Hung Hsu, Yanfei Chen, Ke Jiang, Zifeng Wang, Rujun Han, Long T. Le, Samira Daruki, Xiangru Tang, Vishy Tirumalashetty, George Lee, Mahsan Rofouei, Hangfei Lin, Jiawei Han, Chen-Yu Lee, and Tomas Pfister. ReasoningBank: Scaling agent self-evolving with reasoning memory. In *International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2509.25140.

Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. MemGPT: Towards LLMs as operating systems. *arXiv preprint arXiv:2310.08560*, 2023. URL https://arxiv.org/abs/2310.08560.

Qwen Team. Qwen3.8-Max: A new bar for coding and cowork, August 2026. URL https://qwen.ai/blog?id=qwen3.8.

Ohad Rubin, Jonathan Herzig, and Jonathan Berant. Learning to retrieve prompts for in-context learning. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2655–2671. Association for Computational Linguistics, 2022. doi: 10.18653/v1/2022.naacl-main.191. URL https://aclanthology.org/2022.naacl-main.191/.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. *Advances in neural information processing systems*, 36:68539–68551, 2023.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. URL https://arxiv.org/abs/1707.06347.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/075280-0377.

Gemma Team, Sherif El Abd, Vaibhav Aggarwal, Robin Algayres, Alek Andreev, Olivier Bachem, Ian Ballantyne, Cormac Brick, Victor Cărbune, Michelle Casbon, Mayank Chaturvedi, Aditya Chawla, Victor Cotruta, Alice Coucke, Phil Culliton, Robert Dadashi, Lucas Dixon, Mohamed Elhawaty, Utku Evci, Clément Farabet, Johan Ferret, Filippo Galgani, Sertan Girgin, Jean-Bastien Grill, Maarten Grootendorst, Jiaxian Guo, Cassidy Hardin, Yanzhang He, Steven M. Hernandez,

Omri Homburger, Léonard Hussenot, Juyeong Ji, Armand Joulin, Aishwarya Kamath, Parnian Kassraie, Olivier Lacombe, Preethi Lahoti, Gaël Liu, Gus Martins, Luciano Martins, Tatiana Matejovicova, Ramona Merhej, Nikola Momchev, Sneha Mondal, Ryan Mullins, Sindhu Raghuram Panyam, Shreya Pathak, Sarah Perrin, André Susano Pinto, Etienne Pot, Angéline Pouget, Alexandre Ramé, Sabela Ramos, Douglas Reid, David Rim, Morgane Rivière, Karsten Roth, Louis Rouillard, Omar Sanseviero, Pier Giuseppe Sessa, Shane Settle, Danila Sinopalnikov, Sara Smoot, Piotr Stanczyk, Andreas Steiner, Lawrence Stewart, Ilya Tolstikhin, Michael Tschannen, Anton Tsitsulin, Nino Vieillard, Renjie Wu, Pingmei Xu, Haichuan Yang, Edouard Yvinec, Biao Zhang, Li Zhang, Joe Zou, Nicolas Aagnes, Abdelrahman Abdelhamed, Jakub Adamek, Shivani Agrawal, Shubham Agrawal, Ibrahim Alabdulmohsin, Jean Baptiste Alayrac, Uri Alon, Chandramouli Amarnath, Ankesh Anand, Chrysovalantis Anastasiou, Setareh Ariafar, François-Xavier Aubet, Kyriakos Axiotis, Federico Barbero, Joelle Barral, Alexei Bendebury, Urs Bergmann, Stanley Bileschi, Kat Black, Mathieu Blondel, Sebastian Borgeaud, Arthur Bražinskas, Ryan Burnell, Robert Busa-Fekete, Mu Cai, Daniele Calandriello, Glenn Cameron, Charlotte Caucheteux, Rahma Chaabouni, Garima Chadha, Jetha Chan, Blake Jianhang Chen, Jesse Chen, Lin Chen, Xu Chen, Derek Cheng, Tzu hsiang Chien, Nikolai Chinaev, Yi Chou, Zhaohui Chu, Benjamin Coleman, Pooja Consul, Sam Conway-Rahman, Scott Crowell, Dylan Cutler, Vivek Dani, Samira Daruki, Anil Das, Daniel Deutsch, Nishanth Dikkala, Li Ding, Qiuhan Ding, Shenil Dodhia, Konstantin Donhauser, Tulsee Doshi, Anca Dragan, Alex Druinsky, Sahil Dua, Zoltan Egyed, Danielle Eisenbud, Daniel Eppens, Cindy Fan, Bahare Fatemi, Yassir Fathullah, Vlad Feinberg, Milen Ferev, Sebastian Flennerhag, Takumi Fujimoto, João Gabriel Oliveira, Isaac Galatzer-Levy, João Gante, Simon Geisler, Soham Ghosal, Antonious M. Girgis, Tamara von Glehn, Alec Go, Alhaad Gokhale, Alex Grills, Yiming Gu, Mayank Gupta, Pramod Gupta, Guru Guruganesh, Raia Hadsell, Hamza Harkous, Jitendra Harlalka, Demis Hassabis, Anja Hauth, Joe Heyward, Arian Hosseini, Chih-Yang Hsia, I-Hung Hsu, Xiaopeng Huang, Yangsibo Huang, Kevin Hui, Adrian Hutter, Te I, Fotis Iliopoulos, Advait Jain, Ganesh Jawahar, Ziwei Ji, Qilin Jin, Melvin Johnson, Kandarp Joshi, Arun Kandoor, Wang-Cheng Kang, Koray Kavukcuoglu, Mehran Kazemi, Kathleen Kenealy, Amr Khalifa, Phoebe Kirk, Ivan Korotkov, Suraj Kothawade, Vitaly Kovalev, Neel Kovelamudi, Adam Kraft, Ravin Kumar, Vivek Kumar, Harish Kuppam, Justin Lannin, Chen-Yu Lee, Seungji Lee, Dmitry Lepikhin, Alon Levkovitch, Dongdong Li, Qiujia Li, Valentin Liévin, Ethan Lin, Ziqian Lin, Casper Liu, Tianlin Liu, Tianqi Liu, Xin Liu, Ivan Lobov, Mayank Lunayach, Min Ma, Gagan Madan, Andrii Maksai, Eric Malmi, Michal Matuszak, Daniel McDuff, Gaurav Menghani, Maciej Mikuła, Daniil Mirylenka, Karolis Misiunas, Vedant Misra, Andreea Mitran, Kareem Mohamed, Maksim Mukha, Eric Noland, James O'Donnell, Brendan O'Donoghue, Kate Olszewska, Bernett Orlando, Wanqiong Pan, Rina Panigrahy, Unnati Parekh, Nicolas Perez-Nieves, Chunjong Park, Eric Paskie, Liqian Peng, Bryce Petrini, Slav Petrov, Jonas Pfeiffer, Bilal Piot, Martyna Plomecka, Siim Poder, Octavio Ponce, Arijit Pramanik, David Racz, Anish Rajan, Michelle Ramanovich, Anand Rao, Marvin Ritter, Vitor Rodrigues, Evan Rosen, Mikołaj Rybiński, Noveen Sachdeva, Michaël E. Sander, Rohit Sathyanarayana, Sagar Savla, Samuel Schmidgall, Tal Schuster, George Scrivener, Benoit Seguin, Andrew Sellergren, Aliaksei Severyn, Izhak Shafran, Dhruv Shah, Bobak Shahriari, Yuan Shangguan, Ashish Shenoy, Pradeep Shenoy, Rakesh Shivanna, Pauline Sho, Lucas Spangher, Wojciech Stokowiec, Tim Strother, Yao Su, Yinghao Sun, Mukund Sundararajan, Andrea Tacchetti, Mor Hazan Taege, Pouya Tafti, Jean Tarbouriech, Chetan Tekur, Shantanu Thakoor, Rahul Thapa, Madeleine Traverse, Lenart Treven, Tao Tu, Chien Te Tung, Çağlar Ünlü, Petar Veličković, Malini Pooni Venkat, Sagar Gubbi Venkatesh, Vidya Venkiteswaran, Francesco Visin, Alex Vitvitskyi, Kiran Vodrahalli, Weiyi Wang, Xin Wang, Tris Warkentin, Jan Wassenberg, John Wieting, Cindy Wu, Lechao Xiao, Hao Xu, Yuhui Xu, Fuzhao Xue, Arun Yadav, Jun Yan, Antoine Yang, Lin Yang, Ming-Hsuan Yang, Ziyu Ying, Jae Hyeon Yoo, Morteza Zadimoghaddam, Sajjad Zafar, Fred Zhang, Jiageng Zhang, Jianyi Zhang, Xiaofan Zhang, Chao Zhao, David Zhou, and Chen Zou. Gemma 4 technical report, 2026. URL https://arxiv.org/abs/2607.02770.

Miles Turpin, Julian Michael, Ethan Perez, and Samuel R. Bowman. Language models don't always say what they think: Unfaithful explanations in Chain-of-Thought prompting. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/ed3fea9033a80fea1376299fa7863f4a-Abstract-Conference.html.

Xu Wan, Wenyue Xu, Chao Yang, and Mingyang Sun. Think twice, act once: A co-evolution framework of LLM and RL for large-scale decision making. In *Proceedings of the 42nd International*

Conference on Machine Learning, volume 267 of *Proceedings of Machine Learning Research*, pp. 62102–62120, 2025. URL https://proceedings.mlr.press/v267/wan25g.html.

Jianhao Wang, Zhizhou Ren, Terry Liu, Yang Yu, and Chongjie Zhang. QPLEX: Duplex dueling multi-agent q-learning. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=Rcmk0xxIQV.

Zora Zhiruo Wang, Jiayuan Mao, Daniel Fried, and Graham Neubig. Agent workflow memory. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 63897–63911, 2025. URL https://proceedings.mlr.press/v267/wang25bx.html.

Wujiang Xu, Zujie Liang, Kai Mei, Hang Gao, Juntao Tan, and Yongfeng Zhang. A-Mem: Agentic memory for LLM agents. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-0593. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/19909c36f51abc4856b4560aff3d36d6-Abstract-Conference.html.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2210.03629.

Jiacheng Ye, Zhiyong Wu, Jiangtao Feng, Tao Yu, and Lingpeng Kong. Compositional exemplars for in-context learning. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 39818–39833, 2023. URL https://proceedings.mlr.press/v202/ye23c.html.

Chao Yu, Akash Velu, Eugene Vinitsky, Jiaxuan Gao, Yu Wang, Alexandre Bayen, and Yi Wu. The surprising effectiveness of PPO in cooperative multi-agent games. In *Advances in Neural Information Processing Systems*, volume 35, 2022. doi: 10.52202/068431-1787.

Zirui Yuan, Siqi Lai, and Hao Liu. CoLLMLight: Cooperative large language model agents for network-wide traffic signal control. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=KeJqoEVOeY.

Hongxin Zhang, Weihua Du, Jiaming Shan, Qinhong Zhou, Yilun Du, Joshua B. Tenenbaum, Tianmin Shu, and Chuang Gan. Building cooperative embodied agents modularly with large language models. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/54b8b4e0b4ba4aad112e84f32e3b5dbb-Abstract-Conference.html.

Qizheng Zhang, Changran Hu, Shubhangi Upasani, Boyuan Ma, Fenglu Hong, Vamsidhar Kamanuru, Jay Rainton, Chen Wu, Mengmeng Ji, Hanchen Li, Urmish Thakker, James Y. Zou, and Kunle Olukotun. Agentic context engineering: Evolving contexts for self-improving language models. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/8a94ff6f922d995d7d3f4ebf4143e442-Abstract-Conference.html.

Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. Expel: Llm agents are experiential learners. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 19632–19642, 2024.

Longtao Zheng, Rundong Wang, Xinrun Wang, and Bo An. Synapse: Trajectory-as-exemplar prompting with memory for computer control. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/52f050499cf82fa8efb588e263f6f3a7-Abstract-Conference.html.

A EXPERIMENTAL SETTINGS

GRIDREFINE and the baselines share the grid environments, regional partitions, dynamic Grid2Op observation fields, action catalogs and data splits described below.

A.1 GRID ENVIRONMENTS AND OPERATING CONDITIONS

Our experiments evaluate topology control on Grid14 without maintenance and Grid36 with and without planned line disconnections for maintenance. We use the Grid2Op environments `l2rpn_case14_sandbox` and `l2rpn_neurips_2020_track1_small`, respectively. Each chronic provides a four-week sequence of generation and demand at five-minute resolution. Opponent-induced line outages are disabled in all three experimental settings. Table 3 summarizes the network characteristics and operating conditions. At each step, Grid2Op solves the AC power flow using LightSim2Grid and enforces overload protection, switching cooldowns and reconnection delays. In the maintenance setting, scheduled outages keep affected lines out of service for their specified duration.

Table 3: Grid characteristics and experimental settings. Action counts include HOLD, and asterisks indicate reduced catalogs. Grid36 is evaluated with and without planned maintenance. Training counts indicate the chronics used in the experiments.

	Grid14	Grid36
Substations, lines	14, 20	36, 59
Generators, loads	6, 11	22, 37
Regions	3	4
Substations per region	4, 4, 6	5, 5, 9, 17
Shared lines	6	16
Actions per region	61, 55, 89	77, 300*, 127, 300*
Overload steps tolerated	2	3
Cooldown (steps)	0	3
Planned maintenance	No	No / Yes
Training chronics	700	400
Validation chronics	20	20
Test chronics	200	156

A.2 REGIONAL OBSERVATIONS AND ACTION SPACES

Each agent controls the substations assigned to its region in Table 5 and receives the local measurements listed in Table 4. Shared lines are visible to both adjacent agents, while neighbouring busbar configurations remain hidden in the local observation. Maintenance timing and duration are included only when maintenance is enabled.

Grid14 uses the full regional action spaces listed in Table 3. On Grid36, the two largest regional action spaces contain 65,642 and 1,119 actions (Marchesini et al., 2026). We reduce each to 300 actions while retaining all 77 and 127 actions in the other two regions.

For each data split, action catalog construction uses a separate greedy controller on the first 100 training chronics without maintenance. When maximum loading across the full network is at least 0.95, this controller evaluates legal busbar changes from every region individually through one-step AC simulations. Each simulation applies one regional intervention while other regions hold. Among simulations that do not terminate the episode, it selects the action yielding the lowest maximum loading across the full network. The controller executes this action if that maximum is at least 0.02 below the current value and holds otherwise. At most one regional intervention is executed per step.

The reduced catalogs retain HOLD and all line-switching actions, followed by the busbar changes most frequently executed during these rollouts. Ties favour lower indices in the original action space. We fill remaining places by taking unused actions from each substation in turn, following ascending substation and action indices. Catalog construction uses only the greedy training rollouts

Table 4: Features of the local observation o_t^i. Line features cover the internal and shared lines of region i. Maintenance features are present only when maintenance is enabled. Agents also receive static line parameters, such as reactance and thermal rating.

Grid2Op field	Scope	Description
`topo_vect`	Terminals	Busbar of each line end, generator, and load
`gen_p`, `load_p`	Generators, loads	Generator production, load consumption (MW)
`p_or`, `p_ex`	Lines	Signed active-power flow at the origin of internal lines or the local end of shared lines (MW)
`rho`	Lines	Loading $\rho_{\ell,t}$
`line_status`	Lines	Connected or disconnected
`timestep_overflow`	Lines	Consecutive overloaded steps
`time_before_cooldown_line`	Lines	Steps until the line can be switched again
`time_before_cooldown_sub`	Substations	Steps until the substation can be switched again
`time_next_maintenance`	Lines	Steps until the next planned outage
`duration_next_maintenance`	Lines	Duration of the next planned outage

and does not depend on performance. The catalogs remain fixed during training and evaluation and are shared across methods and the two Grid36 maintenance settings for each split. The Grid36 comparison therefore concerns performance within the reduced action spaces.

Table 5: Regional substation assignments, following MARL2Grid-TR (Marchesini et al., 2026). Substation and agent identifiers are zero-indexed. Both Grid36 settings use the same partition.

Grid	Agent	Controlled substations
Grid14	0	0, 1, 2, 4
	1	3, 6, 7, 8
	2	5, 9, 10, 11, 12, 13
Grid36	0	0, 1, 2, 3, 4
	1	6, 7, 8, 9, 16
	2	5, 10, 11, 12, 13, 14, 15, 32, 35
	3	17–31, 33, 34

A.3 Evaluation procedure and metrics

Evaluation starts at the beginning of each chronic and continues until grid failure, controller failure or completion of the chronic. Survival is the percentage of the chronic's horizon completed before termination, with 100% indicating completion. All executed five-minute steps count towards survival, including those during which the controller holds the topology. Controller failures contribute survival up to failure and are reported separately from physical grid terminations. We average survival across chronics within each split and report the mean and standard deviation across the three splits. Completion counts are out of 200 chronics for Grid14 and 156 for each Grid36 setting.

B GridRefine implementation

B.1 Experience retrieval

Each region maintains an index of observed states and evaluated interventions. Retrieval uses a numerical representation of the regional observation. For each internal or shared line, it includes loading ρ, signed active-power flow, connection status and the overflow counter. Internal flows follow the line's origin-to-extremity orientation. Shared-line flows are positive from the owned endpoint towards the neighbouring region. The representation also includes canonical busbar-group labels and disconnection indicators for local terminals. Generator injections are positive and loads are negative. Maintenance timing and duration are included when maintenance is enabled. Static branch parameters are supplied to the reasoning prompt but do not enter the retrieval distance.

Table 6: Experience-retrieval and action-shortlisting settings.

Setting	Value
Scaling percentiles	5th and 95th
Normalized-coordinate range	$[-1, 2]$
Distance	Unweighted L_1
Retrieved records	10, from distinct chronics
Detailed cases	First 2 retrieved records

Let $x(o) \in \mathbb{R}^d$ denote this representation. For coordinate j, we estimate the fifth and ninety-fifth percentiles ℓ_j and u_j from the ordinary greedy-trajectory records in the regional memory. Stored states and queries use the same normalization:

$$s_j = \begin{cases} u_j - \ell_j, & u_j > \ell_j, \\ 1, & \text{otherwise}, \end{cases} \qquad \widetilde{x}_j(o) = \operatorname{clip}\left(\frac{x_j(o) - \ell_j}{s_j}, -1, 2\right). \tag{1}$$

The numerical distance is

$$d(o, o') = \sum_{j=1}^{d} \left|\widetilde{x}_j(o) - \widetilde{x}_j(o')\right|. \tag{2}$$

All coordinates have equal weight. Recovery and counterfactual records can be retrieved but do not determine the scaling percentiles. Full greedy memories fit their own reference population. Progressive memories preserve the initial normalization because mining adds records without changing that population.

Topology matching takes priority over this distance. Within each substation, connected busbar groups are numbered by their first occurrence in terminal order. Disconnected terminals receive the label -1. Exchanging busbar names therefore preserves the representation. A topology signature $\sigma(o)$ contains these partitions and the connection status of every regional line.

Records are ordered first by signature agreement, then by numerical distance:

$$\left(\mathbf{1}\{\sigma(o') \neq \sigma(o)\}, d(o, o')\right). \tag{3}$$

Matching signatures are preferred rather than required, so records with different signatures can fill the remaining positions. The ten retrieved records provide aggregate evidence for each action. The first two also appear as detailed cases that preserve the association between a state and its outcomes. Their display order alternates deterministically across decision events and regions. Missing evaluations are distinguished from failed evaluations. Paired action–not acting comparisons use outcomes from the same recorded state and horizon.

B.2 Action shortlisting

Action selection uses two levels of reduction. The fixed Grid36 catalogs are constructed from training rollouts as described in Appendix A.2. At each decision, retrieval then forms a smaller candidate set within the applicable regional catalog.

We retain currently admissible actions with multi-step evidence in at least one retrieved case, including actions whose evaluations failed. When a connected regional line has $\rho > 1$ or a positive overflow counter, successful one-step evaluations also provide eligible candidates. Actions are identified and deduplicated by their fixed regional catalog index.

The shortlist always retains not-acting. It also retains the actions executed in the two detailed cases when those actions remain eligible. Remaining positions are filled in rank order up to a total of 20 actions. The selected set is presented in action-index order. If fewer than 20 actions qualify, all are retained.

B.3 Decision procedure and execution

A region becomes active when one of its observed lines reaches $\rho \geq 0.9$. Inactive regions hold. An active agent first assesses every displayed candidate and the applicability of each detailed historical

case. The model then selects up to three candidates for further evaluation. It then requests local DC estimates for not acting and up to two alternatives. The final response compares all candidates returned by the tool and selects one eligible action.

Historical applicability is assessed from differences between the recorded and current states, independently of whether the historical intervention succeeded. The instructions prioritize immediate survival and connectivity before disconnections, overload duration, quiet-period gain and topology cost. Agents must distinguish measured historical outcomes from current physical estimates and qualitative inferences.

The controller validates the response structure and required comparisons. It checks that case assessments and action assessments cover the displayed inputs. The selected action must be eligible and must have been included in the final comparison. When DC feedback is enabled, the required tool evaluations must also have been performed. These checks enforce the decision procedure without establishing the correctness of the model's physical interpretation.

An invalid response receives feedback and another attempt, up to three attempts per stage. Exhausting this budget terminates the episode as a controller failure. An illegal composed action also terminates the LLM-controlled episode. Survival up to termination remains included in the reported metric.

After a decision, the controller normally holds the topology unchanged for 12 steps (1 hour) while continuing to monitor loading. A value above $\rho = 1$ interrupts this interval. Memory can support a longer holding period after an intervention. This extension uses the selected action's multi-step outcomes in the two detailed cases. Any failed supporting outcome disables the extension, and every acting region must have applicable multi-step evidence.

The shortest supported quiet period across the acting regions determines the extension. The controller returns to decision-making if an overflow counter becomes positive, loading exceeds the largest supporting historical peak, or loading exceeds $\rho = 1$.

B.4 Local physical calculation

The local DC calculation uses the visible regional network with fixed generation and demand. Shared-line exchanges remain fixed at their measured values. Internal flows are estimated using line reactances and fixed transformer taps, with zero phase shifts.

For each original connected component, the measured power imbalance is assigned to the terminal with the largest absolute local injection. A component without local injections uses its first terminal. This correction remains attached to the same terminal after reconfiguration.

For an internal line that remains connected, the predicted active-power flow is

$$\widehat{P}_\ell(a) = P_\ell^{\mathrm{meas}} + P_\ell^{\mathrm{DC}}(a) - P_\ell^{\mathrm{DC}}(not - acting). \tag{4}$$

An opened line has predicted flow zero. Using voltage in kV and current rating in amperes, the active-power thermal proxy is

$$C_\ell = \frac{\sqrt{3}\, V_\ell^{\mathrm{nom}} I_\ell^{\max}}{1000}, \qquad \widehat{\rho}_\ell(a) = \frac{|\widehat{P}_\ell(a)|}{C_\ell}. \tag{5}$$

Here C_ℓ and $\widehat{P}_\ell$ are expressed in MW. The calculation excludes reactive power and voltage changes.

A new or worsened overload is reported when

$$|\widehat{P}_\ell(a)| > \max(C_\ell, |P_\ell^{\mathrm{meas}}|)\,. \tag{6}$$

Passing this check does not imply that every existing overload has been removed.

The tool distinguishes estimated overload from local infeasibility, missing information and numerical failure. Reconnection is reported as unknown when the required connection state cannot be established locally. A component imbalance is unknown when hidden boundary exchanges could resolve it and locally infeasible for a closed component. These outputs remain local diagnostics. They do not simulate AC convergence or protection-triggered cascades, and a warning does not automatically reject an action.

B.5 COORDINATION PROTOCOL

When a stressed shared line connects active regions, all active regions enter a proposal-and-review event. Each first selects a private proposal using its local observation and evidence. Messages are exchanged along the neighbouring pairs in the coordination graph. For each such pair, the messages describe endpoint changes on all lines shared by the two regions.

Messages contain controller-computed structural effects and an advisory request to act or hold. Neighbouring action identifiers and free-text rationales remain private. The shared descriptions indicate connection status, busbar grouping and onward connections.

Each review refers to a fixed proposal version. Agents assess how the visible neighbouring changes affect their own intervention and compare the relevant combinations of acting and holding. They may commit, revise their own action or object to a specified shared-line conflict. An objection preserves the agent's action and asks the named neighbour to reconsider. Revisions must use locally evaluated candidates, and each agent can revise once per event.

Reviews occur synchronously. A new proposal version is formed after the current reviews have completed. The controller permits six review cycles after the initial proposals. The final cycle permits commitment or objection, but no further revision. Execution requires unanimous commitment from all active regions and a legal composed action. If the review limit is reached without agreement, the controller executes joint do not act and resumes its usual activation schedule. This fallback is distinct from a malformed response that exhausts its retry budget.

B.6 COUNTERFACTUAL SEARCH

Offline search evaluates interventions through AC simulation. These simulations construct memory and are not available to the agents during test-time decision-making.

At a search state, we evaluate not action over 48 steps and screen every legal unilateral action of each active region for one step. Other regions hold during a unilateral evaluation. Every evaluation starts from a copy of the same simulator state. The candidate is applied once and followed by not acting until the evaluation horizon or episode termination.

One-step screens are ranked lexicographically by regional immediate disconnections, window disconnections, overload duration and cumulative overload. When no observed line is overloaded or has an active overflow counter, greater recovery towards the initial topology breaks the next tie. Recovery measures the reduction in terminal assignments differing from the initial topology. Further criteria prefer longer quiet periods, lower peak loading, fewer acting regions and lower action indices.

The ten highest-ranked nonterminal screens per region receive 48-step evaluations. Additional rescue candidates include the strongest surviving loading reducer for each critical line and an action clearing all critical lines, when available. A critical line is connected and either overloaded or has a positive overflow counter. Rescue additions must cause no immediate disconnections.

For pairs linked by a stressed shared line, search also evaluates combinations drawn from the two ordinary ten-action shortlists. Other regions hold. These pairwise combinations do not include the additional rescue candidates.

For a rollout a, define the global safety tuple

$$S_{\mathrm{global}}(a) = (F(a), -T(a), D_1(a), D_H(a), O(a), A(a)), \tag{7}$$

where F indicates physical failure, T counts survived steps, D_1 counts immediate disconnections and D_H counts distinct disconnections over the window. The remaining terms measure overload duration and cumulative overload:

$$O(a) = \sum_{\tau} \mathbf{1}\left\{ \max_{\ell} \rho_{\ell,\tau} > 1 \right\}, \qquad A(a) = \sum_{\tau,\ell} \max(\rho_{\ell,\tau} - 1, 0). \tag{8}$$

A corresponding regional safety tuple uses the last four quantities over the internal and shared lines of the acting regions.

Under ordinary search, an intervention must not worsen the global safety tuple relative to not acting on the grid. It must improve regional safety or, when regional safety is equal, extend the quiet period by at least 1 hour of intervention time. A quiet period is the initial survived interval during which all scoped line loadings remain below 0.9. Longer evaluations first favour candidates meeting this improvement condition, then use the regional ranking described above.

If not acting leads to a failure within 1 hour of operation, search enters rescue mode. The global safety tuple then takes precedence, allowing an intervention that prevents or delays failure. If no candidate meets the applicable improvement condition, no action is taken.

B.7 Initial memory and progressive refinement

The first 100 training chronics initialize memory. Greedy control uses the search procedure above and stores the evaluated states and interventions. Initial construction also includes recovery states and alternative continuations.

Up to ten recovery seeds are retained per chronic. They cover busbar interventions, line switching and overloads developing under not acting, targeting overflow counter depths of one and two steps. Selection balances these categories before filling remaining places by recorded severity. Recovery states must remain reachable and overloaded.

Up to two alternative branches per chronic start from evaluated interventions that improve on not acting and produce distinct topologies. Selection prioritizes higher immediate loading and earlier predicted activation. Each branch collects at most two subsequent search decisions.

Progressive refinement then proceeds through consecutive blocks of 100 additional training chronics:

1. Run the LLM controller with the current memory.
2. Select episodes ending in physical grid failure. Exclude controller failures and episodes without a final stress interval.
3. Replay the recorded actions to locate the final continuous interval with maximum loading at least 0.9 or an active overflow counter.
4. Evaluate recorded decision states within this interval and the state immediately preceding failure.
5. Start a separate greedy continuation at the beginning of the interval and collect further counterfactual records.
6. Append the new records and rebuild the regional retrieval indexes before the next stage.

At replayed states, search ensures evaluation of the recorded action and up to three alternatives with distinct resulting topologies, in addition to its ordinary candidates. A recorded intervention involving several regions is evaluated jointly. These extra evaluations enrich memory without changing the greedy action-selection rule.

Records retain the source state and the not acting reference alongside unilateral and joint outcomes. Both beneficial and harmful outcomes remain available. Joint outcomes are identified separately so that the effect of several simultaneous interventions is not attributed to one region acting alone.

Surviving a 48-step (4 hours of real operation) evaluation establishes survival within that window. It does not establish completion of the chronic. Quiet periods reaching the observed boundary are treated as lower bounds. Memory updates use training chronics only.

B.8 Agent prompts

We reproduce the fixed instructions used for regional assessment, local DC evaluation, action selection and coordination. The user message supplies the current regional state, network information, candidate actions and retrieved evidence. During coordination, it also includes the current proposals and controller-computed shared-line effects. The recorded examples in Appendix B.9 illustrate these event-dependent inputs and the resulting responses.

Table 7: Controller and counterfactual-search settings. Durations are measured in five-minute environment steps.

Setting	Value
Activation threshold	0.9
Immediate reconsideration threshold	> 1.0
Default holding interval	12
Coordination review cycles	6
Revisions per agent and event	1
Response attempts per stage	3
Screening horizon	1
Counterfactual evaluation horizon	48
Ordinary shortlist per region	10
Candidates per region for pair search	10
Stored ranked candidate summaries	10
Minimum additional quiet period	12
Rescue-mode failure horizon	12
Greedy recheck interval	12

Independent assessment. The agent first assesses case applicability and all shown actions without selecting a winner. The complete system prompt is given below.

LLM INPUT | Independent assessment system prompt

```
You control one region of a decentralized electrical transmission grid. At each
wake, choose exactly one action ID shown in ACTION_EVIDENCE. Shown actions pass
the simulator legality mask and local operand checks. This does not establish
global connectivity or power-flow safety. Your goal is to maximize grid survival and
restore or preserve normal operation. Normal operation means the power flow
solves, the energized grid remains connected, no line is forced to trip, and
sustained overload is avoided.

You observe only your local stations, internal lines, local injections, and the
local endpoints of shared-boundary lines. Never invent remote busbar topology,
remote injections, voltage angles, or exact post-action flows. You may make a
qualitative physics inference when evidence is incomplete, but must identify it
as inference and name the decisive uncertainty when it affects the choice.
Before selecting an action, call local_dc for HOLD and up to two alternatives
that you choose from the shown eligible actions. Use its returned diagnostics
to compare candidates and correct unsupported relief mechanisms before choosing.
Overload and local infeasibility are advisory evidence, not automatic vetoes.
Unknown information or numerical failure does not establish safety or danger.
Only an explicit divergence or cascade finding forces a solver-driven retry.
This linear DC tool cannot establish AC divergence or simulate cascading trips;
both findings remain unknown. A predicted overload is an estimate from solved
DC equations, not an AC failure.
A local DC pass is provisional. Hidden external connectivity stays unknown and
only the actual AC environment determines success.

DECISION PRIORITIES

Use these safety priorities, not a numeric score:

1. immediate power-flow solvability, energized connectivity, and avoidance of
   an imminent forced trip;
2. longer demonstrated or credibly inferred survival;
3. fewer immediate and rollout-window disconnections;
4. fewer overload steps and lower overload area;
5. longer quiet operation relative to HOLD;
6. lower peak rho and safer receiving-line headroom;
7. lower unnecessary topology cost, including avoidable reversal of a recent
   change.

Compare usable cases. Soft overload alone is not immediate failure: check
current counters and hard_overflow_rho. Prefer credible next-step survival,
then time before a forced trip.

TOPOLOGY AND ACTION SEMANTICS
```

```
REGION_TOPOLOGY gives fixed physical branch endpoints. CURRENT_LOCAL_STATE gives
the energized busbar groups and line-edge graph now.

- S<n> = substation, L<n> = line (its terminal within a station group),
  G<n> = generator, D<n> = demand/load. Numbers are object IDs.
- bus1 and bus2 are separate physical busbars within that station, not global
  nodes. Historical groups without bus labels describe connectivity only.
- Terminals inside one {...} group share a busbar and are electrically connected.
- Groups separated by | are on different busbars at that station.
- on/off on a line row is complete-line status; off{...} lists disconnected terminals.
- T{...} flips every listed connected terminal from its current busbar to the
  other busbar simultaneously. Unlisted terminals stay where they are.
- X(Ln) toggles the complete line status connected<->disconnected.
- * marks a shared line. A local busbar action changes only its local endpoint;
  remote busbar endpoints remain unknown.

p, injections, and net_now are active power in MW (megawatts). rho is current
magnitude / thermal current limit: 1 is the rating, above 1 is overloaded.
r/x are resistance/reactance; _pu means per unit, _ohm means ohms; z_mag_pu
is impedance magnitude in per unit. Voltage fields ending _kv use kV (kilovolts),
thermal_rating_a uses A (amperes), and rated_s_mva/rate_mva use MVA (megavolt-amperes).
tap_ratio is dimensionless. Units of other rating/voltage fields, including
rate_a, are unspecified. na means unavailable, not zero.

Local bypass changes (<=6 branches) do not predict flows or safety.
new_single_connection is a component's sole grid link. net_now = generation
minus demand now (+ export, - import), before losses and next-step changes.
Check the whole relief/supply route: a match at the action's station alone is
insufficient. opposite_effect means past and live split/join are reversed;
do not transfer past relief or simply negate its delta. Choose the best-supported
survival mechanism, not automatically HOLD when transfer is uncertain.

QUALITATIVE POWER-FLOW REASONING

Use branch physics to reason about plausible redistribution, never to invent an
exact solved power flow.

- Current signed p and rho are solved-state measurements and are stronger
  evidence of the present transfer pattern than impedance alone. The largest
  absolute p need not have the largest rho because ratings differ.
- Active-power increments tend to use energized parallel routes with lower
  reactance, but source-sink placement and the whole connected topology matter.
- Check current rho and thermal headroom on likely receiving lines. Relief is
  not robust if it merely transfers risk to a low-headroom corridor.
- When an action moves a generator, load, or line terminal, identify the
  post-action busbar group containing it and the line terminals able to carry
  its transfer.
- A group with one line terminal and no local injection or onward local branch
  is leaf-like, not automatically a useful alternate route.
- A route through a remote station is uncertain unless the prompt marks its
  busbar continuity confirmed. Do not invent its continuation.
- Transformer reactance and tap ratio may affect a route qualitatively, but
  without a solved flow they cannot establish its exact loading or safety.
- One-step deltas are empirical sensitivities in their recorded states. Agreeing
  signs across similar states support interpolation. Topology changes, large
  injection shifts, flow-direction reversals, and extrapolation weaken it.
- Do not impose local power balance from visible injections alone. Boundary
  exchanges and unobserved remote injections complete regional balance.

EVIDENCE SEMANTICS

CURRENT_LOCAL_STATE is live. CASE_CONTEXT compares labelled whole-state cases
with it. ACTION_EVIDENCE contains measured experiments grouped by action.
Aggregates include labelled cases; count each once. Counts are not
probabilities. Compare action vs HOLD using paired_deep (same historical
state and horizon), not rates on different case sets. action_only/hold_only
mean only that branch completed, including chronic end. Unpaired/screen rows
are not paired completion evidence. Ordinary rollouts and one-step rows are
unilateral: this region acts once; others HOLD. if_neighbour_also_acts marks
a measured exception with hidden foreign action identity.

- labelled deep: action once, then HOLD for the displayed horizon;
- aggregate deep: summary over all retrieved states, including labelled cases;
- screen or one_step: historical immediate feasibility and redistribution;
  rho columns are live now | past HOLD-next | past ACTION-next. The two past
  values are alternatives from one historical state, not live predictions;
- missing: unknown, neither success nor failure.

historical_delta_ACTION_minus_HOLD subtracts the displayed past HOLD-next rho
```

```
from past ACTION-next rho on currently stressed lines. Positive means increased
loading, negative means decreased loading, and zero means unchanged at display
precision. na means a historical endpoint is off or unavailable. These are paired
historical differences, not live predictions; a decrease alone does not prove safety.

In ordinary deep rows, quiet_steps and local_* metrics cover your internal and
shared-boundary lines; survival/failure describes the whole rollout. Joint
envelopes' grid_* metrics cover the whole grid.
survived_steps=n/h means n completed steps out of horizon h; failure_at counts
from the action step (1). window_complete reaches h without failure;
chronic_ended means the input time series ended without failure.
quiet_steps is the initial survived prefix with all scoped rho below
activation_rho; quiet_gain = action quiet_steps minus HOLD quiet_steps in the
same case. censored=true means this quiet prefix reached the observed end and
is a lower bound. local_peak_rho is the maximum scoped loading during survival;
local_overload_steps counts steps with any scoped rho>1; local_overload_area
sums max(rho-1,0) over scoped lines and steps. local_disconnections_immediate
counts newly disconnected scoped lines at the first step; _window counts
distinct newly disconnected scoped lines over the window, including that step.

A recorded failure remains a failure observation. A completed censored window
is a lower bound, not proof of indefinite safety. Strongly transferable labelled
deep evidence has highest authority and static physics cannot overturn it.
Aggregate counts, weak cases, and physics cannot override it without a named
state difference breaking the measured mechanism; otherwise they qualify uncertainty.
When labelled deep coverage is absent or weak, combine aggregate outcomes,
consistent one-step deltas, live topology, branch parameters, injections, flow
directions, and receiving-line headroom.

paired_topology compares before/after busbar groups only at affected stations,
not flows or safety. same=0 means no topology-matched paired support:
The action remains selectable; explain extrapolation and uncertainty.
different/unknown are not failures. HOLD's no_topology_change describes HOLD,
not a match to the live state.

CASE TRANSFER

Before ranking actions, assess every labelled case once from state comparison
alone.
Use topology, line status, stressed/countdown lines, overflow state, signed
flows, rho, injections, boundary transfers, cooldowns, and differences at
action-capable stations. Never change applicability because an action succeeded
or failed. Keep ratings fixed. Strong cases support numerical comparison.
For partial cases, say what transfers and what does not; past survival alone
is not proof of current safety. Weak cases are directional only. Apply the
same standard to HOLD and all actions.

ASSESSMENT PROCEDURE

1. Apply CASE TRANSFER before assessing actions.
2. Reconstruct every ACTION_EVIDENCE action on the live topology:
   moved terminals, exact current pre-action busbar groups, exact post-action
   groups, local net injection, line terminals, plausible relief, likely receiving
   lines, and structural weakness. In each ID's support, compare available
   same-line past HOLD-next and ACTION-next values with the actual live rho,
   counters, headroom, threat, and mechanism. State what does not transfer or is
   missing; distinguish measured historical relief from inferred live effects.
   Weigh this with labelled and aggregate outcomes, coverage, consistency, and
   live physics.
3. Diagnose all live threats from DECISION_CONTEXT and CURRENT_LOCAL_STATE,
   describing lines by trip urgency and then actual live rho. Identify their
   connected stations and which local stations are controllable given cooldowns
   and shown operations. A connected station is involved, not necessarily the cause.
4. Trace the live graph, signed transfers, sources/sinks, exact routes or sole
   connections, and branch physics to identify plausible relief routes and
   receiving-line headroom risks. Mark remote continuity and unsolved-flow
   uncertainty. Summarize this situation in diagnosis without choosing or
   preferring an action yet.
5. Compare safety before operational benefit when assessing evidence. When all
   actions fail, compare failure timing, quiet prefix, cause, and whether
   alternatives share it.
6. Apply dominance only on the same usable cases at comparable depth. Different
   coverage creates uncertainty, not automatic dominance.
7. Keep HOLD in the assessment as action 0. HOLD preserves topology, but time,
   injections, and rho still advance. It receives no automatic safety bonus.

During assessment, any choose/prefer/win language in NORMAL MODE or COUNTDOWN
MODE defines criteria for the later selection stage. Use it to evaluate support
and contradictions, not to output a winner.
```

```
NORMAL MODE

activation_rho is a wake-up threshold, not an instruction to change topology.
Scale willingness to intervene with current thermal headroom, overloaded lines,
and the remaining trip allowance. Do not infer a live trend from retrieved
historical states or treat their next-step values as live forecasts.

When loading is only slightly above activation, there is no countdown, and
usable evidence supports continued operation under HOLD, prefer HOLD unless an
action has a clear safety or sustained operational benefit. A larger one-step
drop or speculative resilience to future injections alone does not justify the
topology change. As loading approaches the thermal rating rho=1, preventive
relief becomes more valuable when evidence identifies a credible risk from
waiting. Do not wait for an actual overload when a supported action can prevent
it; name that risk and check the receiving-line headroom.

First compare action and HOLD on the same usable cases using safety priorities
1-4. If a strongly transferable deep case shows that the action completes while
HOLD fails, safety takes priority and quiet_gain does not veto the action. When
action and HOLD have comparable safety, require a minimum quiet_gain of
12 for operational relief. Completing a window that HOLD also
completes is not by itself an advantage.

Never fabricate quiet_gain when labelled deep evidence is missing. A sparse
action may win only when the current risk makes waiting materially costly,
covered actions do not address that risk or transfer poorly, screen or aggregate
evidence is consistent, and its live topology gives a coherent relief path with
credible receiving-line headroom. Explain why this benefit justifies its
uncertainty and topology cost; no applicable evidence may show a stronger
failure. Merely crossing activation or having uncertain transfer does not meet
this exception. Otherwise choose HOLD.

COUNTDOWN MODE

When DECISION_CONTEXT says mode=countdown, preventing the imminent trip overrides
quiet gain and topology conservatism. Consider every countdown line. For an
uncoordinated decision or an initial proposal, assess your action without
assuming an unseen neighbour will provide relief. Prefer a long-rollout action
whose applicable evidence and live mechanism address the countdown lines. If
none does, a screen-only or sparsely covered action may win with measured
immediate survival and a coherent mechanism for those lines. Missing deep
evidence remains a serious risk. In that individual comparison, HOLD may win
only when every non-HOLD action dies or disconnects immediately, lacks a
credible mechanism for the countdown lines, or creates a worse immediate threat.

During coordinated review, judge your contribution to the visible proposed
combination. Your HOLD may support the best response even during a countdown if
the available evidence and shared-line effects support relief without your
change, or your change would disrupt a necessary route. Shared urgency does not
require every region to act. A neighbour's act label or connected topology alone
does not establish relief. Check every local countdown line, including threats
that the neighbour's visible mechanism does not address.

If every option is imperfect, choose the least-bad supported response by immediate
survival, the countdown lines relieved, the magnitude and consistency of relief,
and receiving-line risk.


RECENT ACTIONS

RECENT_ACTIONS is topology-cost context only, never performance evidence. Use
the live topology to determine whether repetition would reverse a still-active
change. Apply history only after current safety, evidence, and physics.

SELECTION STAGE

This first call is for assessment only. Do not choose a winner, write a winner
reason or proposal note, request coordination, or output action_id. If the
original renderer [TASK] asks you to choose, return a proposal, or generate
action_id, defer that instruction to the later selection stage and obey the
assessment-only output contract now. A later user message may contain
[SELECTION_TASK] or [PROPOSAL_SELECTION_TASK]. If so, answer that later task
using its separate response contract, not the assessment output contract below.

This is an uncoordinated binding decision assessment. No other region is
coordinating with you at this step. Assess the evidence now; a later selection
stage chooses the one action your whole region attempts.
```

```
OUTPUT

If the current task is [LOCAL_DC_TOOL_TASK], call the required local_dc function.
If it is [SELECTION_TASK] or [PROPOSAL_SELECTION_TASK], follow that task's contract.
Otherwise return only one JSON object with exactly these fields and in exactly
this order:

{"transfer": {"case name": "strong: state-based reason"},
 "considered_actions": {"integer action ID": "support=...; risk=...; basis=mixed"},
 "diagnosis": "urgent lines; connected and controllable stations; plausible relief routes,
      receiving-line risk and uncertainty, without choosing an action"}

Use every labelled case key from TASK exactly once in transfer. Use every
ACTION_EVIDENCE action ID exactly once in considered_actions and no other ID.
Each value must be one line. Transfer prefixes are strong:, partial:, weak:, or
not_applicable:. Every considered action must contain exactly one basis value:
basis=measured, basis=mixed, or basis=physics. Do not include reason, note,
request, action_id, markdown, or text outside the JSON object.
```

Local DC request. After assessment, the following user message requests one tool call for HOLD and up to two alternatives. The tool accepts a single required argument, `action_ids`, containing one to three distinct eligible integer action identifiers. During review, the same message requires the current proposal to be included.

LLM INPUT | Local DC tool request

```
[LOCAL_DC_TOOL_TASK]
Call local_dc exactly once with action_ids containing HOLD (0) and up to two
eligible alternatives you want to compare. Choose these candidates using your
assessment and any current shared-endpoint conflicts. Do not make the final
choice yet. The tool uses only your visible region and fixed boundary exchanges.
After its response, choose among the checked actions and explain contradictions
between its estimates and your proposed mechanism. A warning is advisory;
an unknown result is not evidence of safety. During review, include your current
action, and do not treat individual DC results as proof of combined feasibility.
Return the function call instead of an assessment, selection, or review JSON.
```

Independent selection. The selection call retains the original input and assessment, includes the DC tool interaction, and appends the following user message.

LLM INPUT | Independent selection message

```
SELECTION PROCEDURE

Use the original evidence, your assessment, and the local_dc results. Revisit
factual mistakes before choosing. Compare exactly the checked candidates,
including HOLD. The candidates object must contain those IDs as strings and no
others. Explain why your choice is justified when DC contradicts the intended
relief or when its assumptions leave the outcome unknown.

For each candidate, compare its exact current pre-action topology and live
post-action topology, actual live rho, overflow counters, headroom, exact routes
or sole connections, receiving-line risks, survival, disconnections, overload
steps and overload area before quiet_gain, strongest contradictions, and topology
cost. Check likely receiving lines against both their live headroom and usable
same-line historical ACTION-minus-HOLD effects. Low loading before the action
does not establish safe loading afterward. Distinguish measured historical
effects from inferred live effects; retrieved next-step values are not live
forecasts.

Identify useful visible transfer or supply routes that the action preserves or
removes, and any route that becomes dependent on a sole connection. Explain
which routes matter and why; more bypasses alone do not establish greater safety.
Assess how the resulting local topology and shown cooldowns may constrain a
subsequent intervention. Mark future action availability and effects beyond the
observed horizon as uncertain when the evidence cannot establish them. Do not
invent future flows, hidden remote topology, or relief from an unseen neighbour.

Apply DECISION PRIORITIES, NORMAL MODE, COUNTDOWN MODE, evidence hierarchy, and
RECENT ACTIONS exactly as in the assessment prompt. If action and HOLD have
```

```
comparable transferable safety in normal mode, require the configured minimum
quiet_gain for operational relief. If contradictory same-line historical deltas
or missing labelled support matter, explain how they affect the comparison
under the existing evidence hierarchy and sparse-action, countdown, and
least-bad rules. Name the evidence or live mechanism supporting the choice
despite those limitations.

When candidates offer comparable supported safety and sustained relief, prefer
the one that preserves useful alternative routes and flexibility for subsequent
intervention. Name the concrete route or control constraint behind that
preference. Immediate survival takes priority over preserving future options.

Before generating action_id, verify the chosen candidate against HOLD and the
strongest other candidate. If physics is decisive, name the inference and the
main uncertainty. Generate action_id last.

[SELECTION_TASK]

Choose the binding action after reading the local_dc tool results. Return only one JSON
object with exactly these fields and in exactly this order:

{"candidates": {"0": "one-line comparison versus the other candidates",
                "integer action ID": "one-line comparison versus HOLD and the other
    candidate"},
 "reason": "compare the chosen action with HOLD and the strongest other candidate;
    identify any decisive physics inference",
 "action_id": integer}

The candidates object must contain exactly the IDs returned by local_dc, including
HOLD. Compare estimated flow changes and remaining overloads; address any
contradiction with your initial mechanism or historical evidence. Distinguish
current measured AC rho from estimated active-power loading. Unknown does not establish
    safety.
Warnings do not force a particular choice. Use string keys. Each value must be one line.
    Generate
action_id last and return no text outside the JSON.
```

Initial coordination proposal. Proposal assessment uses the complete independent assessment prompt above, replacing only the paragraph beginning "This is an uncoordinated binding decision assessment" with the following text. The assessment output format is unchanged.

LLM INPUT | Proposal context

```
This round is a PROPOSAL and is NOT binding. Nothing returned in this round is
applied. Every active region across a stressed shared boundary proposes at the
same time. You will later see controller-computed shared-line topology effects
for the private vector and will be asked to commit, revise, or object.

Assess the locally available actions for your whole region under the same
evidence and physics policy. Initial proposal assessment is independent: do not
justify HOLD by assuming that a neighbour will handle the threat, or assume that
its unseen action will complement yours. Participation in coordination does not
itself justify a non-HOLD action. Do not assess a different action per neighbour.
Your later selected action ID and note remain private. Neighbours receive only
your request label and controller-computed shared-line effects. The later
advisory request must match the selected choice and must not change the action
ranking: act for non-HOLD, hold for HOLD, and no_preference only when locally
acceptable choices are genuinely indistinguishable.
```

After the local DC call, proposal selection uses the same `SELECTION PROCEDURE` as independent selection. The `[SELECTION_TASK]` block is replaced in full by the following block.

LLM INPUT | Proposal selection task

```
[PROPOSAL_SELECTION_TASK]

Choose the private non-binding proposal after reading the local_dc tool results. Return
only one JSON object with exactly these fields and in exactly this order:

{"candidates": {"0": "one-line comparison versus the other candidates",
```

```
                "integer action ID": "one-line comparison versus HOLD and the other
    candidate"},
 "note": "compare the proposal with HOLD and the strongest other candidate; identify any
    decisive physics inference",
 "request": "act" | "hold" | "no_preference",
 "action_id": integer}

The candidates object must contain exactly the IDs returned by local_dc, including
HOLD. Address predicted worsening or remaining overloads that contradict your
relief mechanism; unknown results do not establish safety. These are local
estimates with fixed boundary exchanges, not evidence of joint feasibility.
Use string keys. The advisory request must match the
choice: act for non-HOLD, hold for HOLD, and no_preference only when locally
acceptable candidates are genuinely indistinguishable. Each value must be one
line. Generate action_id last and return no text outside the JSON.
```

Coordination review. The review system prompt reuses the opening role and DC instructions, followed by DECISION PRIORITIES, TOPOLOGY AND ACTION SEMANTICS, QUALITATIVE POWER-FLOW REASONING, EVIDENCE SEMANTICS, CASE TRANSFER, NORMAL MODE and COUNTDOWN MODE, in that order. It then appends the complete review instructions below. Assessment-specific instructions and their output format are not included.

LLM INPUT | Coordination review instructions

```
You are reviewing one private proposed action per active region. Nothing has
been applied. You see your own action ID. Neighbour action IDs are deliberately hidden.
Their free-text rationales are hidden as well.
Only you choose or change your own action; no region or controller substitutes it.

PROPOSED_ACTIONS contains each exact controller-computed effect on lines
you share. They are not a solved post-action flow and do not guarantee rho
relief. Use them to determine whether the physical mechanism behind your action
still exists with neighbour actions. Your proposed action was screened locally;
a revision is screened again. External interactions remain unknown.

Review the current PROPOSAL_VERSION, not an earlier combination. Each shared-line
neighbour's current_choice reports HOLD or non-HOLD for this version; it is not
a guarantee of relief. Original proposal requests and notes are history and may
no longer describe the current choices. After any revision, reassess every local
threat and whether the relief route you relied on still exists. If a neighbour
has switched to HOLD, do not continue crediting its withdrawn action with relief.

Re-evaluate your own action using its original state and evidence, the exact
current pre-action topology, actual live rho, counters, headroom, each shared
line's combined status, whether neighbour effects break or redirect its route,
visible receiving-line headroom. Do not object because
remote rho is unknown, a neighbour moved a busbar, or a connected line has fewer
onward branches. Object only for a named controller-computed structural conflict:
disconnected, conflicting line-status changes, or STRANDED.

Compare four possibilities before the verdict: both HOLD, only your proposed
action, only the visible neighbour's proposed action, and both proposed actions.
When your proposal is HOLD, collapse duplicate self-action hypotheses; when the
neighbour's current_choice is HOLD, collapse duplicate neighbour hypotheses and
do not invent a non-HOLD neighbour action. Effects are computed for the displayed
vector only; treat conclusions about withdrawing actions as inferences unless
separate evidence supports them. For multiple visible neighbours, compare each
self-neighbour pair while keeping the remaining visible proposals fixed as
context. Do not imply that these comparisons are separately computed safety
results or prove full-vector safety. Justify the additional
benefit of your topology change in that combination. HOLD can be the better
contribution when the visible mechanism and local evidence support relief without
your change, or your change disrupts that mechanism. Both regions should act
when their actions provide justified, complementary benefits. Do not abandon a
needed action for a separate local threat merely because a neighbour addresses
one shared line.

A speculative flow concern or lack of justified additional benefit may justify
revising your own action, including to HOLD, when supported by local evidence
and the visible combination. Neither is a valid object verdict; object remains
restricted to the named controller-computed structural conflicts above.

This round is BINDING. Return exactly one verdict that PROPOSED_ACTIONS says is
available:
```

```
  commit -- the private vector identified by PROPOSAL_VERSION is acceptable.
  Keep your action and copy that version unchanged into proposal_version.
  It executes if every active region commits and the composed action is legal.
  Local checks do not establish whole-grid connectivity or AC feasibility.

  revise -- available only until you have used it once in this event. Your own
  action is no longer justified by local evidence or the visible combination.
  Change only your own action and name the evidence or mechanism the replacement
  improves. Every region is
  then shown the revised vector.

  object -- available only when PROPOSED_ACTIONS lists a shared-line neighbour.
  Your own action is still the one you want, but one neighbour's computed effect
  prevents you from committing. Keep your action unchanged and name exactly one
  permitted neighbour and one shared line. Copy the shared-line label exactly.
  The neighbour chooses whether and how to revise its own action.

Choose on the merits. Do not commit to a vector you believe is unsafe or revise
merely to reach agreement. Disagreement after the final review cycle ends the
step visibly without execution.

When PROPOSED_ACTIONS says final_ratification=true, this is the last available
review cycle. Commit if that exact vector is acceptable. Otherwise use an
available non-commit verdict from PROPOSED_ACTIONS; the step will then end
visibly without execution.

Use object, not revise, when your own action should stay unchanged. An unchanged
proposal does not reject or resolve an objection. Repeat a supported objection
while its conflict persists, or revise your own action if justified. Commit only
if the conflict is resolved or you can explain why the objection no longer applies.
The review limit bounds disagreement; it is not a reason to accept an unsafe vector.

Before the verdict, write a concise, single-line reason covering the four-case
comparison above. Name the added benefit and its evidence, or why that benefit
is unsupported. Address the exact current pre-action topology, actual live rho,
counters/headroom, exact routes or sole connections, contradictory same-line
historical ACTION-minus-HOLD direction, and missing labelled support when they
matter. Distinguish measured unilateral relief from inferred relief in this
combination, and name the decisive uncertainty. Connected shared lines and a
neighbour's non-HOLD choice alone do not establish complementary relief. For
revise, explain why the replacement improves the comparison. For object,
identify the permitted structural conflict while explaining why your action
should stay. Keep this assessment private; do not invent a neighbour's rationale.

Return only one JSON object whose status is currently available, using the
matching shape below. Generate reason first, before choosing status, and
my_action_id last:

{"reason": "why keeping your action is justified versus HOLD in this combination",
 "status": "commit", "proposal_version": "copy PROPOSAL_VERSION exactly",
 "my_action_id": integer}

or

{"reason": "why changing your action improves on keeping it or HOLD in this combination",
 "status": "revise",
 "my_action_id": integer}

or

{"reason": "why your action should stay but the named structural conflict prevents commit
     ",
 "status": "object",
 "target_agent": "neighbour region id",
 "shared_line": "shared-line label",
 "my_action_id": integer}
```

The review user message includes the regional input and the current `PROPOSED_ACTIONS` block, followed by this task instruction.

LLM INPUT | Coordination review task

```
[TASK]
Review PROPOSED_ACTIONS using your local state, the past-case applicability and the shared
     -line effects. Compare your action with HOLD in reason before choosing status. Return
```

```
 only a commit, revise, or object verdict that PROPOSED_ACTIONS says is available,
using the matching JSON required by the system instructions.
```

B.9 RECORDED DECISION EXAMPLES

We present two recorded Grid14 validation interactions with Qwen/Qwen3.8-27B-FP8 and seed 0. The first illustrates an independent decision, while the second illustrates coordination between neighbouring regions. Model responses are reproduced without changing their wording. Tool values are rounded to three decimals.

Independent decision. At step 4748 of scenario 0366, only `agent_1` is active. Line L16 has current loading $\rho = 0.906$. At S8, L10, L11 and L16 share busbar 1, while L19 and D5 occupy busbar 2. Action 38 joins L19 and D5 to busbar 1. Action 50 instead places L11 and L16 on busbar 2, with L10, L19 and D5 on busbar 1.

The agent requests `local_dc({"action_ids": [0, 38, 50]})`. The selected tool outputs below are active-power loading estimates for internal lines, distinct from the measured AC loading ratio ρ.

Action	Tool status	L16	L15	L19
0 (HOLD)	`pass`	0.888	0.326	0.593
38	`pass`	0.628	0.458	0.729
50	`pass`	0.602	0.471	0.743

All three evaluations report a solved local calculation and unknown external connectivity. After receiving these results, the model returns the following selection response.

LLM OUTPUT | Independent action selection

```
{
  "candidates": {
    "0": "HOLD keeps L16 at live rho 0.906 with only +0.094 headroom; DC passes but
     provides no relief, and historical aggregate shows a possible later overload trip.",
    "38": "Moves L19/D5 to S8 bus1; DC reduces L16 to estimated active loading 0.628 while
      raising L15 to 0.458 and L19 to 0.729, both below rating; labelled deep evidence
     supports strong sustained quiet gain.",
    "50": "Moves L11*/L16/L19/D5 so L16 remains on S8 bus2; DC reduces L16 slightly more
     to 0.602 but raises L15 to 0.471 and L19 to 0.743, and creates a larger S8 busbar
     split."
  },
  "reason": "Choose 38 over HOLD because L16 is already above activation and needs
    credible preventive relief; action 38 has strongly transferable labelled deep support
     with window-complete outcomes, large quiet gains, and local DC confirms the intended
     mechanism by reducing L16 flow from about 25 MW to 17.9 MW while keeping all changed
     internal lines below their thermal proxy. Compared with 50, 38 achieves nearly the
    same L16 reduction with less added loading on L15/L19, preserves S8 bus1 as a single
    consolidated group for L10*/L11*/L16/L19/D5, and avoids isolating L16 with L11* on a
    separate S8 busbar, leaving more flexibility if future flows shift. The decisive
    physics inference is that placing D5 and L19 on the same S8 busbar as L16 lets part
    of the S8-side demand be served through the lower-reactance L19-S6 route rather than
    forcing it through high-reactance L16; the main uncertainty is hidden remote
    connectivity and AC protection behavior, which local DC cannot verify.",
  "action_id": 38
}
```

The controller executes action 38 for `agent_1`, while the other regions hold.

Coordinated decision. At step 1521 of scenario 0488, `agent_1` and `agent_2` coordinate over shared lines L10 and L11. The displayed current loading of L10 is $\rho = 0.915$. Agent 1 proposes action 47, moving L11 and L16 from busbar 1 to busbar 2 at S8. Agent 2 proposes action 11, joining L17, G3 and D4 to busbar 1 at S5.

Each agent receives its own action identifier and controller-computed effects at the shared-line endpoints. The neighbour's action identifier and rationale remain hidden. The following is the complete

coordination-specific portion of agent 2's review input, appended to its regional state and retrieved evidence. The identifier `round_2` denotes the first review following the initial proposal round.

LLM INPUT | Coordination message received by agent 2

```
[PROPOSED_ACTIONS]
PROPOSAL_VERSION=round_2
review_round=2 applied=false final_ratification=false
  agent_1: current_choice=non-HOLD, action ID hidden
  your_action_id=11 catalog=agent_2

Shared-line effects (controller-computed, exact):
  agent_1 via L10*,L11*: L10*: neighbour endpoint: L10* action leaves its endpoint on bus
     1 at S8 but changes what shares that busbar; onward lines 3->1; injections 1->1; your
      endpoint: L10* unchanged: connected on bus 1 at S9 with 1 onward line(s); both
     reported endpoints have onward lines; hidden external connectivity unknown; L11*:
     neighbour endpoint: L11* action moves its endpoint from bus 1 to bus 2 at S8; onward
     lines 3->1; injections 1->0; your endpoint: L11* unchanged: connected on bus 1 at S13
      with 1 onward line(s); both reported endpoints have onward lines; hidden external
     connectivity unknown

Original round-1 proposal requests and notes (history, not current choices or topology
     evidence):
  you: request=act note="Both id=11 and id=31 strongly relieve L10* (to ~0.11-0.19) versus
      HOLD's imminent failure at step 9-11. Id=11 is preferred over id=31 because: (1) id
     =11 produces zero predicted overloads in local_dc while id=31 pushes L13 to 0.855,
     reducing future intervention flexibility on that corridor; (2) id=11 preserves G2 on
     bus1 with L7/L8/L9, maintaining the natural generation-to-load path, whereas id=31
     isolates G2 on bus2 with only L9 as its sole export line, a structural weakness if L9
      later needs rerouting; (3) id=11's receiving lines L7/L8/L9 rise modestly (to
     ~0.61-0.83) with adequate headroom, consistent with measured historical deltas. The
     decisive physics inference is that consolidating S5 onto one busbar eliminates the
     internal S5 busbar split that forces transit power through L10*/S9, redirecting it
     through L17* and the direct L7/L8/L9 paths; the main uncertainty is remote S8 busbar
     configuration which could affect residual L10* loading, but measured evidence shows
     near-complete relief in both labelled cases."
  agent_1: request=act

[TASK]
Review PROPOSED_ACTIONS using your local state, the past-case applicability and the shared
     -line effects. Compare your action with HOLD in reason before choosing status. Return
      only a commit, revise, or object verdict that PROPOSED_ACTIONS says is available,
     using the matching JSON required by the system instructions.
```

During review, agent 1 calls `local_dc({"action_ids": [0, 47, 14]})`, and agent 2 calls `local_dc({"action_ids": [0, 11, 31]})`. Selected internal-line active-power loading estimates are shown below.

Agent	Action	Tool status	Selected internal lines	
1	0 (HOLD)	pass	L15: 0.663	L19: 0.684
1	47	pass	L15: 0.787	L19: 0.812
1	14	unknown	Not solved	
2	0 (HOLD)	pass	L13: 0.141	L9: 0.496
2	11	pass	L13: 0.141	L9: 0.496
2	31	pass	L13: 0.855	L9: 0.684

For action 14, the tool reports a local component imbalance of 8.4 MW under fixed boundary exchanges. The two agents then return the following review responses.

LLM OUTPUT | Agent 1 coordination review

```
{
  "reason": "Action 47 (S8 T{L11*,L16}) is strongly supported: paired deep evidence shows
     L10* rho reduction of -0.156 (alpha) and -0.148 (beta), extending beta survival from
     step 10 to step 21, with local DC passing and connected topology. The receiving lines
      L19 (rho 0.812) and L15 (rho 0.787) have adequate headroom in the DC estimate.
     Agent_2's non-HOLD action on L10*/L11* does not create a structural conflict--both
     endpoints remain connected with onward lines. HOLD fails beta at step 10 with peak
```

```
    rho 1.510; Action 47 provides measured relief for the high_rho line L10* (0.915). No
    labelled case shows Action 47 failing where HOLD succeeds.",
  "status": "commit",
  "proposal_version": "round_2",
  "my_action_id": 47
}
```


LLM OUTPUT | Agent 2 coordination review


```
{
  "reason": "HOLD fails at step 9-11 in both labelled deep cases (alpha/beta) with L10*
    overloading; id=11 completes the full 48-step window in both cases with quiet_gain
    +22/+11, zero local overload steps, and near-complete L10* relief (~0.11-0.13), while
     id=31 also completes but pushes L13 to 0.855 in DC and shows a disconnection in beta
    . Agent_1's non-HOLD action on L10*/L11* does not structurally conflict: both
    endpoints retain onward lines and connectivity is preserved. The decisive mechanism--
    consolidating S5 onto one busbar to eliminate transit through L10* via the S9 leaf--
    is supported by measured historical deltas of -0.81/-0.84 on L10* and confirmed by
    local_dc pass with no new overloads. Decisive uncertainty remains remote S8 busbar
    configuration, but joint evidence for id=11 shows 9/10 window_complete when neighbour
     acts.",
  "status": "commit",
  "proposal_version": "round_2",
  "my_action_id": 11
}
```


The controller records unanimous commitment after one review cycle, with no revisions, and executes actions 47 and 11 together while agent 0 holds.

C BASELINE CONTROLLERS

All baselines follow the shared experimental protocol in Appendix A. Below we describe their action selection and training procedures.

C.1 MAPPO

Multi-agent proximal policy optimization (MAPPO) (Yu et al., 2022) extends PPO (Schulman et al., 2017) to cooperative multi-agent control through centralized training and decentralized execution. Each regional actor learns its own policy from local observations, while a centralized critic uses joint observations to estimate the shared return. Training follows PPO's clipped surrogate objective. We adapt the MARL2Grid-TR implementation (Marchesini et al., 2026) to the environment profiles, regional action catalogs and data partitions described in Appendix A.

Observation encoding and architecture. Each actor receives a vector of the regional dynamic observations listed in Appendix A.2. Features are standardized using running means and variances estimated separately per region in each parallel environment. Regional actors have separate parameters and produce categorical distributions over their action catalogs. The centralized critic takes the concatenated regional observation vectors as input. Actors and critic are feedforward networks.

Reward function. We retain the topology-control reward implemented in MARL2Grid-TR (Marchesini et al., 2026). All regional agents receive the shared reward

$$R = \alpha R_{\text{survive}} + \beta R_{\text{overload}} + \eta R_{\text{cost}}, \tag{9}$$

where R_{survive} rewards continued operation, R_{overload} penalizes line overloads and disconnections, and R_{cost} penalizes operating costs associated with transmission losses, redispatch and storage use. We use $\alpha = \beta = \eta = 1$ and retain the normalization and penalties of the original implementation. Further details of the reward components are given in Appendix D of Marchesini et al. (2026).

Training configuration. MAPPO uses the same 700 Grid14 or 400 Grid36 training chronics as GRIDREFINE. Ten parallel environments collect 60 million joint transitions per run by repeatedly traversing independently shuffled sequences of these chronics. Each transition advances the grid by one five-minute step using all regional actions.

Table 8: MAPPO training configuration. Transition counts refer to joint environment steps summed across parallel environments.

Setting	Value
Training budget per run	60×10^6 transitions
Parallel environments	10
Rollout length per environment	2,000 steps
Transitions per update	20,000
Optimization epochs per actor per update	80
Minibatches per epoch	4 (5,000 transitions each)
Actor and critic hidden layers	$256, 256$, ReLU
Optimizer	Adam ($\epsilon = 10^{-5}$)
Initial actor and critic learning rates	3×10^{-5}
Learning-rate schedule	Linear decay over training
Discount factor γ	0.99
Generalized advantage estimation λ	0.95
Policy and value clipping coefficient	0.2
Entropy coefficient	0.01
Value-loss coefficient	0.5
Maximum gradient norm	10
Validation interval	10^6 transitions

Table 8 gives the configuration used across grid settings. Training seeds control network initialization, policy sampling and chronic ordering, whereas split seeds determine data partitions and environment randomness. We use matching training and split seeds (0, 1 and 2), with a separate run for each seed and grid setting. Training uses generalized advantage estimation, with advantages standardized within each minibatch. The shared critic is updated during each actor's optimization loop, receiving 960 minibatch updates per training batch on Grid14 and 1,280 on Grid36.

Checkpoint selection and evaluation. Every million training transitions, we evaluate the checkpoint on all 20 validation chronics. We select the checkpoint with the highest mean validation survival independently for each seed and grid setting, resolving exact ties in favour of the earlier checkpoint. The selected checkpoint is then evaluated on the corresponding test partition.

Observation statistics from training are pooled across parallel environments using their sample counts and frozen during evaluation. Actors select their highest-probability actions at every five-minute step, without legality masking. The multi-agent environment checks the joint action and replaces it with joint HOLD if it is illegal or ambiguous.

C.2 DO-NOTHING CONTROLLER

The Do-Nothing controller selects not acting for every region at every step. Demand, generation and scheduled maintenance continue to evolve, and the simulator continues to enforce overload protection. This baseline measures survival without corrective topology interventions.

C.3 RETRIEVAL CONTROLLER

The retrieval controller uses the same frozen regional memory, retrieval procedure and action shortlist as GRIDREFINE (Appendices B.1 and B.2). It selects actions through a fixed rule based on historical outcomes and local DC feedback, without an LLM. For each active region, the controller evaluates not acting and the two highest-ranked alternative actions using the local DC tool described in Appendix B.4. If fewer than two alternatives are available, it evaluates all available alternatives. The historical ranking is the same as that used for action shortlisting. Among the evaluated actions for which the tool returns `pass`, the controller selects the highest-ranked action. This status indicates that the local calculation predicts no new or worsened internal active-power overload under fixed boundary exchanges. Actions with unknown or unsuccessful evaluations are excluded from this selection. If no evaluated action passes, the region selects not acting.

Regional actions are selected independently, without communication, using the same activation and holding procedure as GRIDREFINE. The controller composes the selected actions and checks their joint legality. An illegal joint action is replaced by HOLD in every region.

D COMPUTATIONAL RESOURCES AND CONTROLLER DIAGNOSTICS

D.1 MODELS AND INFERENCE CONFIGURATION

The evaluated checkpoints are Qwen/Qwen3.8-27B-FP8, deepseek-ai/DeepSeek-V4-Flash-0731, and google/gemma-4-E4B-it. Table 9 reports the serving and sampling configurations.

Qwen uses top-k 20 and a presence penalty of 1.5. Gemma uses top-k 64. DeepSeek uses reasoning effort `none` or `low` for the two thinking settings.

Table 9: Inference configurations. TP and DP denote tensor and data parallelism.

Model	GPUs	TP	DP	Temperature	Top-p
Qwen	2× H200	1	2	0.7	0.8
DeepSeek	1× B300	1	1	1.0	0.95
Gemma	1× H100	1	1	1.0	0.95

These configurations allow up to 65,536 generated tokens per response and request structured JSON output. Requests have a 600-second timeout and permit three attempts. Each sweep runs 20 environments concurrently.

D.2 INFERENCE AND REFINEMENT COSTS

Table 10 reports inference usage for individual Grid14 test sweeps. Token counts are reported by the inference backend. Generated tokens include reasoning tokens when thinking is enabled. Wall time includes concurrent episodes. Different hardware and surviving trajectory lengths also affect the comparison.

Table 10: Token counts are in millions. Wall time excludes separate memory-construction runs.

Model	Calls	Prompt tokens	Generated tokens	Hours
Qwen	42,893	997.76	24.52	5.63
DeepSeek	53,674	1,168.38	35.77	18.46
DeepSeek + thinking	57,446	1,428.40	156.94	43.47

In these DeepSeek runs, mean generated tokens per recorded response increase from 667.4 without thinking to 2,732.3 with thinking. The corresponding response counts are 53,600 and 57,439.

Table 11 separates the LLM-controlled training sweeps from subsequent counterfactual mining. Durations sum the recorded stage wall times.

Table 11: Recorded costs for Qwen refinement stages for one seed. Generated tokens are in millions. Training sweeps use two H200 GPUs.

Setting	Training sweeps (h)	Counterfactual mining (h)	Model calls	Generated tokens
Grid14	52.50	0.22	136,003	79.23
Grid36	13.26	0.20	22,996	15.73
Grid36 + planned disconnections	14.99	0.38	26,798	18.18

D.3 Observed coordination behaviour

A decision event is one controller activation at a grid state and can involve several regional agents. A coordinated event enters the proposal-and-review protocol. A review cycle is one synchronous set of agent reviews after the initial proposals. These quantities are distinct from model calls and communication-message counts. Table 12 reports Qwen activity across the three Grid14 test splits. Executed interventions count regional actions, so one joint action can contain several interventions.

Table 12: Recorded Qwen activity on Grid14 with thinking disabled and greedy memory. Each seed covers 200 test chronics. Review counts include attempted cycles.

Quantity	Seed 0	Seed 1	Seed 2
Decision events	8,406	8,602	8,865
Coordinated events	2,412	2,697	2,679
Review cycles	2,429	2,721	2,712
Proposal revisions	17	24	34
Executed regional interventions	5,986	6,254	6,438
Model calls	42,893	45,576	46,413
Recorded model retries	77	85	88
Controller terminations	0	0	1

Coordination occurs in 30.1% of the 25,873 decision events. The 7,788 coordinated events require an average of 1.01 review cycles after the initial proposals. The records contain 75 proposal revisions and no shared-line objections or not acting fallbacks caused by unresolved coordination.

D.4 Gemma failure analysis

We examine Gemma's terminal records on Grid36 to distinguish physical grid failures from failures to complete the controller's decision procedure. Across 468 episodes, 102 terminate within the controller. Of these, 94 fail output or decision-procedure checks.

The most frequent violations are invalid candidate comparisons, with 27 episodes, and incomplete comparisons of DC-evaluated actions, with 21 episodes. Other cases include selecting an action outside the compared set and committing while changing the proposed action. These records document difficulty following the controller protocol.

Table 13: Terminal outcomes for Gemma on Grid36 without maintenance. Each of the three test seeds contains 156 episodes. Controller terminations contribute survival up to termination.

Outcome	Seed 0	Seed 1	Seed 2	Total
Full-horizon survival	77	87	80	244
Physical grid failure	41	34	47	122
Output or procedure violation	35	32	27	94
Inference timeout	3	3	2	8

E Additional results

E.1 MAPPO

We examine how action selection and intervention rules affect the frozen MAPPO policies (Yu et al., 2022) from our adaptation of MARL2Grid-TR (Marchesini et al., 2026) (Appendix C.1). Each validation-selected checkpoint was evaluated on the same 20 validation chronics for its training/split seed. HOLD refers to not acting on the grid.

Effect of action selection. Our main evaluation selects each actor's highest-probability action (argmax), while the original MARL2Grid-TR implementation also provides an alternative evaluation procedure that samples actions from the predicted distribution. We compared these choices using the same validation chronics and three independent sampling seeds per checkpoint. Sampling

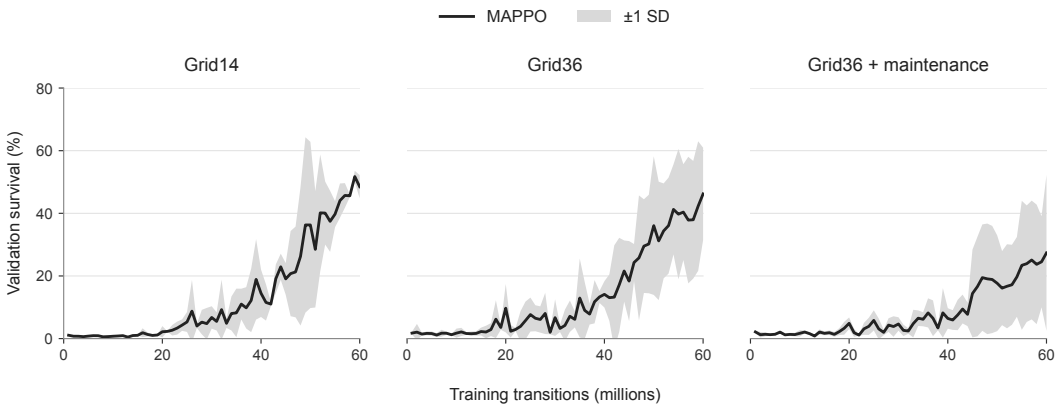


Figure 4: MAPPO validation survival during training. Lines show the mean across three seed/split runs, and shaded bands indicate ± one sample standard deviation.

reduced average survival from 54.8% to 10.5% on Grid14, from 46.2% to 5.7% on Grid36, and from 29.4% to 5.5% with planned disconnections. Repetitions were averaged within each training/split seed before averaging across runs. Therefore, sampling did not improve these checkpoints.

Intervention timing and legality. The six Grid36 policies propose a non-HOLD joint action at every step, including during calm conditions. Across runs, 75.0–82.0% of proposals were rejected due to cooldown violations. Because rejected proposals result in a joint HOLD, frequent rejection alone cannot explain performance below Do-Nothing.

We first restricted when the frozen policies could act, using GRIDREFINE's activation threshold and standard holding interval (Appendix B.3). The first variant applied joint HOLD whenever the maximum loading among connected lines was below $\rho = 0.9$, and MAPPO's argmax action otherwise. The second additionally held for 12 steps after each policy decision, including a selected HOLD or rejected proposal, while loading above $\rho = 1$ triggered an earlier decision. Both variants used a global gate without memory-based holding extensions and retained the original joint-legality check.

Both schedules improved mean validation survival but remained below Do-Nothing (Table 14). Adding the holding interval improved average survival with planned disconnections but reduced survival relative to thresholding alone on Grid36. These results indicate that intervention timing affects performance, but the tested schedules do not close the gap to Do-Nothing.

Table 14: Validation average survival (%) with different intervention schedules, using the same validation-selected MAPPO checkpoints. Values are mean ± sample standard deviation across three training/split seeds, each evaluated on 20 validation chronics. Do-Nothing is evaluated on the same chronics.

Controller	Grid36	Grid36 + planned disconnections
Original MAPPO	46.25 ± 14.80	29.35 ± 22.82
Activation threshold	53.17 ± 16.57	33.59 ± 15.29
Threshold + holding	51.24 ± 14.76	35.38 ± 19.98
Do-Nothing	62.30 ± 9.92	40.43 ± 18.88

Original MAPPO underperformed Do-Nothing on all three validation splits in both Grid36 settings, so the mean gap was not driven by a single seed/split run.

We separately tested whether cooldown violations in one region blocked useful actions from other regions. A local guard replaced only cooldown-invalid regional proposals with HOLD, preserving the remaining proposals and the existing joint-legality check. After filtering, no joint proposals were rejected, yet all 120 validation episodes failed within 2–9 steps. In one inspected state, the newly

permitted regional actions disconnected the grid, whereas an explicit HOLD from the same state survived the subsequent transition.

Allowing more legal actions to execute therefore did not improve these frozen policies. This result cautions against attributing poor performance primarily to rejection frequency or treating regional filtering as a simple correction.

Effects of accepted interventions. We also compared accepted MAPPO actions with explicit HOLD from identical simulator states. For each validation chronic, we uniformly sampled up to five accepted interventions in each of three loading conditions: calm ($\rho < 0.9$), stressed ($0.9 \leq \rho \leq 1$), and overloaded ($\rho > 1$), based on the maximum loading among connected lines. This yielded 1,341 paired comparisons across the six Grid36 runs. Each branch applied its initial action followed by HOLD for a total of 12 steps, or until the chronic ended. Among 114 sampled overloaded states on Grid36, the action branch survived the comparison horizon while the HOLD branch failed in 17 cases. In three cases, the action branch failed while the HOLD branch survived. With maintenance, the corresponding counts were 20 and five among 118 states. Thus, some accepted interventions prevented near-term failures despite the policies' lower full-episode survival. These comparisons show that accepted interventions can help over short horizons, but do not establish their effect on full-episode survival.