Communications Physics

https://doi.org/10.1038/s42005-026-02582-y

Article in Press

Accelerating atomic fine structure determination with graph reinforcement learning

Received: 25 September 2025

Accepted: 2 March 2026

Cite this article as: Ding, M., Darvariu, V.-A., Ryabtsev, A.N. *et al.* Accelerating atomic fine structure determination with graph reinforcement learning. *Commun Phys* (2026). https://doi.org/10.1038/s42005-026-02582-y

Milan Ding, Victor-Alexandru Darvariu, Alexander N. Ryabtsev, Nick Hawes & Juliet C. Pickering

Accelerating Atomic Fine Structure Determination with Graph Reinforcement Learning

Milan Ding*
Department of Physics, Imperial College London,
Prince Consort Road, London, SW7 2AZ, United Kingdom

Victor-Alexandru Darvariu
Oxford Robotics Institute, University of Oxford,
17 Parks Road, Oxford, OX1 3PJ, United Kingdom

Alexander N. Ryabtsev
Institute of Spectroscopy, Russian Academy of Sciences,
Troitsk, Moscow 108840, Russia

Nick Hawes
Oxford Robotics Institute, University of Oxford,
17 Parks Road, Oxford, OX1 3PJ, United Kingdom

Juliet C. Pickering†
Department of Physics, Imperial College London,
Prince Consort Road, London, SW7 2AZ, United Kingdom

February 26, 2026

Abstract

Atomic data determined by analysis of observed atomic spectra are essential for plasma diagnostics. For each low-ionisation open d- and f-subshell atomic species, around 10^3 fine structure energy levels can be determined through years of analysis of 10^4 observable spectral lines. We propose a partial automation of this task by casting the analysis procedure as a Markov decision process and solving it by graph reinforcement learning using reward functions partly learned on historical human decisions. In our evaluations on existing spectral line lists and theoretical calculations for Co II, Nd II and Nd III, hundreds of energy levels were identified and determined in hours, agreeing with published values in 95% of cases for Co II and 54-87% for Nd II and Nd III. As the current efficiency in atomic fine structure determination struggles to meet growing atomic data demands, our artificial intelligence approach sets the stage for closing this gap.

*Corresponding author 1: m.ding15@imperial.ac.uk
†Corresponding author 2: j.pickering@imperial.ac.uk

1 Introduction

The determination of atomic fine structure of low-ionisation open d- and f-subshell elements involves extracting energies and total electron angular momenta J of energy levels from observed atomic spectra. This process, commonly referred to as term analysis [1], also assigns term symbols to levels. It is a sequential, complex decision-making task requiring both atomic spectroscopy expertise and extensive human labour (e.g., [2, 3, 4]). The resulting energy levels and transition wavenumbers are fundamental data [5, 6] with applications in the lighting and metal industries, magnetic confinement fusion research [7], nuclear research and medical isotopes production [8], the search for new physics [9, 10], and astronomy [11, 12], including renewed interest in f-subshell species in neutron star merger observations that are transforming our understanding of the origin of heavy elements [13]. However, most levels and transitions for the heavier elements remain unknown [5], and advanced ab initio calculations are accurate only to a few percent [14, 15, 16], insufficient for applications requiring higher spectral resolutions. Term analyses are therefore vital, offering orders-of-magnitude higher accuracies and reliable theoretical constraints.

Term analyses commonly involve Fourier transform (FT) and grating spectroscopy of plasmas under resolving powers up to 10^6 and 10^5, respectively, and dynamic ranges up to 10^4 across the infrared, visible, and UV ranges [17, 18, 19]. A transition between an upper energy level E_u and a lower energy level E_l is observed as a spectral line at wavenumber σ equal to the energy difference, $\sigma = E_u - E_l$. Several 10^4 spectral lines are observable for each open d- or f-subshell element. The primary challenge is determining energy levels from an immense number of observed energy differences, guided by less accurate theoretical predictions [1, 20, 21]. Existing tools support spectral line wavenumber and intensity extraction [22, 23, 24], visualisation for manual energy level determination [25], and energy level optimisation [26]. While spectrum measurement and theoretical calculations for one species can be completed in weeks, the subsequent analyses still require months to years and remain a major bottleneck. Since the advent of FT spectroscopic term analyses in the 1970s [27], its scope has been mainly limited to the iron-group ($23 \leq Z \leq 28$) elements [5].

Early research demonstrated the potential of pattern-recognition methods for partial term analysis procedures [28, 29]. Here, we develop artificial intelligence (AI) techniques to partially automate term analyses by casting the problem as a Markov decision process (MDP) involving graphs, where an agent determines unknown levels (nodes) and lines (edges) over discrete time steps via reinforcement learning (RL). The agent learns to choose valid actions that maximise a reward function partly trained on human preferences from past analyses. We propose a variant of the Deep Q-network (DQN) algorithm [30] called Term Analysis with Graph Deep Q-Network (TAG-DQN), which belongs to the graph RL [31] class of methods for tackling combinatorial decision-making problems over graphs. Key to achieving scalability is the adoption of techniques that have proven successful in this broader literature including action space decompositions, restraining valid actions via domain knowledge, and using graph neural networks (GNNs) as a learning representation [32].

We emulated early to intermediate stages of published term analyses of Co II [2], Nd III [3], and Nd II [4] as initial MDP states for evaluation case studies, using existing FT spectral line lists [2, 3] and theoretical calculations achievable given the levels known in the initial state. Correct energy level determination rates ranged from 54% with ab initio atomic structure calculations to 95% with semi-empirical methods. Up to 10^2 tentative fine structure energy levels can be computed in hours, alleviating months of human labour needed to achieve similar results. These results were enabled by TAG-DQN learning from rapid attempts at energy level determinations and resolving ambiguities by reaching MDP states (level systems) most consistent within experimental and theoretical uncertainties. Performance also compared favourably with baseline search algorithms. Continued application of machine learning in term analyses is expected to enable rapid developments in fundamental atomic data in years that would otherwise take decades, thereby accelerating progress in atomic physics and across the diverse fields supported by atomic physics.

2 Results

We formulate term analysis as an MDP for RL and evaluate TAG-DQN in realistic scenarios. Term analysis is typically carried out for one atomic species at a time using data on its known (empirically determined) and unknown (theoretically predicted) levels and lines, a spectral line list with the wavenumber σ and intensity I information for observed atomic transitions. Figure 1 illustrates an overview of our method, which applies RL

to search in the solution space. The goal is to determine unknown atomic energy levels and to classify spectral lines that best match the theory within uncertainties.

2.1 MDP Definition

An MDP is defined as the tuple $\langle \mathcal{S}, \mathcal{A}, R, \mathcal{T}, \gamma \rangle$ comprising the state space, action space, reward function, transition function, and discount factor [33]. At a discrete time step t, the agent finds itself in state $s_t \in \mathcal{S}$, takes an action $a_t \in \mathcal{A}$, receives a reward $r_t \sim R(s_t, a_t)$, and advances to the next state s_{t+1} determined by the transition function $\mathcal{T}(s_t, a_t, s_{t+1})$. The goal is to maximise the expected cumulative discounted reward

$$\mathbb{E}[G_t] = \mathbb{E}\left[\sum_{k=0}^{H} \gamma^k r_{t+k}\right], \tag{1}$$

where $\gamma \in [0, 1]$ is the discount factor trading off immediate and future rewards. The policy $\pi(a|s)$ is a probability distribution of actions given states determining the agent's behaviour. We also define the state-action value function of a policy π as $Q_\pi(s_t, a_t) = \mathbb{E}_\pi[G_t|s_t, a_t]$; values for particular states and actions are referred to as Q-values. Our MDP is episodic, and t is limited to a finite horizon H for computational feasibility, which is set manually to specify the maximum number of unknown energy levels that can be determined within a single episode, starting from the initial MDP state.

2.2 State Space $\mathcal{S}$

The state is defined by the data incorporated in a term analysis, as illustrated in Fig. 1A. Known spectral lines, known energy levels, and theoretical calculations form a graph $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$ where nodes $v \in \mathcal{V}$ represent levels and edges $e \in \mathcal{E}$ represent lines. Nodes relate to edges through $\sigma = E_u - E_l$, with the ground energy level fixed at zero. The goal of term analysis is to expand the subgraph of known levels by replacing unknown parts of the graph using data from observed spectra, which are typically preprocessed into a line list by fitting $\sim 10^4$ observed spectral line profiles. Each line list entry corresponds to a fine structure transition, mainly characterised by observed wavenumber σ_{obs}, standard wavenumber uncertainty $\delta\sigma_{\text{obs}}$, relative intensity I_{obs}, and signal-to-noise ratio S/N_{obs}. These quantities remain unaltered by the MDP and serve to generate actions. We consider only electric dipole fine structure transitions for the line list and graph edges, as higher order multipole transitions are rarely observed in the laboratory spectra for term analyses.

The graph nodes and edges are described by features. A known line is a theoretical atomic transition (edge) matched to an entry in the line list, and a known level has energy determined via weighted least-squares optimisation of observed known line wavenumbers. The subgraph of known levels is typically connected, so all observed energies E_{obs} are relative to the zero energy ground level. The node features are

$$\mathbf{x}_v = [E_{\text{calc}},\ E_{\text{obs}},\ \text{known},\ \text{selected}], \tag{2}$$

with theoretical energy as E_{calc} and binary flags for "known" and "selected" levels. Unknown E_{obs} values are zero. The "selected" feature indicates the level currently being determined. The edge features are

$$\mathbf{x}_e = [\sigma_{\text{calc}},\ \sigma_{\text{obs}},\ \delta\sigma_{\text{obs}},\ I_{\text{calc}},\ I_{\text{obs}},\ gA_{\text{calc}},\ S/N_{\text{calc}},\ S/N_{\text{obs}},\ \rho,\ \text{known}], \tag{3}$$

representing theoretical and observed wavenumber, observed wavenumber uncertainty, theoretical and observed relative intensities on the same scale, theoretical weighted transition probability, expected and observed S/N, observed local line density ρ (number of lines per cm^{-1}), and a binary "known" flag. Observed quantities are zero if "unknown" and Boltzmann level populations are assumed for I_{calc} estimation. Further details on graph feature design are provided in Methods.

2.3 Action Space $\mathcal{A}$

As illustrated by Fig. 1B, the MDP alternates between two types of actions $\langle a^{(1)}, a^{(2)} \rangle$ in order to maintain feasible MDP branching factors. The pair of actions results in the determination of E_{obs} for one unknown level and matching at least two observed spectral lines to corresponding edges connecting it to known levels.

The first action type $a^{(1)}$ selects an unknown level to determine and flips the "selected" flag of the corresponding node. The action space $\mathcal{A}^{(1)}$ contains only unknown levels with at least two edges to known levels. While energy level determination by a single line is possible for a few levels in human analyses, we exclude these to avoid unfeasibly large $\mathcal{A}^{(1)}$ as such cases are highly situational. If $|\mathcal{A}^{(1)}| = 0$, the MDP terminates. In our case studies, $|\mathcal{A}^{(1)}|$ ranges between 0 and 200.

The second action type $a^{(2)}$ matches at least two lines from the line list to edges between the unknown level selected by $a^{(1)}$ and known levels. To determine the action space $\mathcal{A}^{(2)}$ (possible E_{obs} values), we first gather the $k > 1$ graph edges connecting the selected level to k known levels ($k = 2$ for Fig. 1B). For each edge, candidate matches in the line list are filtered by

$$\sigma_{\text{obs}} \in [\sigma_{\text{calc}} \pm \Delta E] \quad \text{and} \quad I_{\text{obs}} \in [I_{\text{calc}} \pm \Delta I], \tag{4}$$

where ΔE and ΔI are theoretical uncertainties in energy level and line intensity, respectively. The filtered σ_{obs} values of an edge are added to, or subtracted from, the known level energy of that edge, depending on whether the known level is predicted as the lower or upper level. This generates candidate E_{obs} values from each edge. Since $k > 1$, E_{obs} candidates not repeating within a tolerance δE (approximately the maximum experimental wavenumber uncertainty) are neglected. The expected total number of candidates can be estimated by

$$N_{\text{cand}}(k) \sim \frac{\Delta E}{\delta E}\left(\prod_{i=1}^{k}(1 + 2\,\delta E\,\rho_i) - 1 - \sum_{i=1}^{k} 2\,\delta E\,\rho_i\right), \tag{5}$$

where ρ_i is estimated as the number of line list entries filtered by (4) divided by $2\,\Delta E$. This expression is built from considering the probability of a line with ρ_1 lying within $\pm\,\delta E$ of another line with ρ_2, which is $\delta E/\Delta E$, so that the expected number of pairs within tolerance is $(2\Delta E\rho_1)(2\Delta E\rho_2)(\delta E/\Delta E)$. For typical values ($\Delta E = 500$ cm^{-1}, $\delta E = 0.05$ cm^{-1}, $\rho = 0.5$ per cm^{-1}), $N_{\text{cand}}(k = 2) \sim 5$. The base $|\mathcal{A}^{(2)}|$ is $N_{\text{cand}}(k)$ and can reach 10^3 for larger k, ΔE, δE, and ρ. In practice, a minimum E_{obs} repetition count $N_{\text{rp}} > 2$ could be set to suppress large amounts of lower order terms in (5), reducing the action space when more than two spectral lines involving connecting known levels are expected significantly above the spectrum noise level. We raised N_{rp} to 3 or 4 if more than 3 or 4 of the k lines exceed $S/N_{\text{calc}} = 5$, effectively neglecting terms with products of 2 or 3 ρ_i in (5). In our case studies, the median $|\mathcal{A}^{(2)}|$ is below 10.

2.4 Transitions $\mathcal{T}$

The graph structure remains fixed throughout the MDP, but node and edge features are updated by actions, as illustrated by Fig. 1. Transitions are deterministic; a single future state s' can be reached with probability 1 after taking action a in state s. Concretely, each next state s_{t+1} is generated by updating the "known" and "selected" features of the level chosen by $a^{(1)}$, re-optimising E_{obs} for all known levels, and updating $[\sigma_{\text{obs}}, \delta\sigma_{\text{obs}}, I_{\text{obs}}, S/N_{\text{obs}}, \text{known}]$ of the new known edges using corresponding entries in the line list, which are excluded from $\mathcal{A}^{(2)}$ for the remainder of the episode. A no-op action for $a^{(2)}$ is always available, handling levels with no candidates and the forfeit of $a^{(1)}$. When no-op is chosen, the next state reverts to the state prior to $a^{(1)}$, but the time step increments and the level chosen by $a^{(1)}$ is excluded from $\mathcal{A}^{(1)}$ until the episode end.

2.5 Reward Function R

As term analysis is inherently empirical, the reward reflects confidence in determined energy levels, which often have ambiguous values. This confidence arises from the ability of the newly determined level to enable future level determinations and its agreement with theory and observations. The consequences of choosing ambiguous E_{obs} values will be explored by RL. While the action space incorporates theoretical and experimental uncertainties, a non-trivial reward remains necessary to differentiate between term analysis states. As $a^{(1)}$ and $a^{(2)}$ have different semantics, we also differentiate their reward functions.

Term analysis prioritises lines with the highest signal-to-noise ratios (equivalently, minimising uncertainties and entropy [34]), constraining subsequent analysis of weaker lines. Thus for $a^{(1)}$, we define reward $r^{(1)}$ as

$$r^{(1)} = 0.1 \cdot \log_{10}\left(\sum_{i=1}^{k} S/N_{\text{calc},i}\right), \tag{6}$$

where $S/N_{\mathrm{calc},i} \in [2, 10^4]$, the sum is over the k edges between the selected unknown level and known levels, and the 0.1 factor scales reward magnitude for learning stability. For $a^{(2)}$, the reward is

$$r^{(2)} = (D - 1) \cdot r^{(1)}, \tag{7}$$

with $D \in [0, 1]$ representing preference score for a particular $a^{(2)}$. The no-op $a^{(2)}$ has $D = 0$, yielding net zero reward for the action pair; otherwise, the net reward is positive.

Designing D is challenging. In practice, human experts weigh the agreement between observed and theoretical line intensities and the consistency of repeated E_{obs} values within wavenumber uncertainties. However, uncertainties in transition probabilities and errors in spectral line fitting, especially for widespread weak and/or blended lines, complicate this judgement. While these nuances can be verified by spectrum inspection or improving theoretical calculations, they can also be inferred from properties of the k lines from experience.

Thus, instead of hardcoding a fixed functional form for D, we learn it from historical human expert decisions via a form of inverse reinforcement learning [35], approximating part of the human reward function. We use a simple multi-layer perceptron (MLP) to predict D from features of the k lines

$$[\delta\sigma_{\mathrm{obs}},\ \sigma_{\mathrm{diff}},\ \delta\sigma_{\mathrm{obs}} - \sigma_{\mathrm{diff}},\ I_{\mathrm{obs}},\ I_{\mathrm{calc}},\ -|I_{\mathrm{obs}} - I_{\mathrm{calc}}|,\ \rho], \tag{8}$$

where $\sigma_{\mathrm{diff}} < \delta E$ is the smallest wavenumber difference between E_{obs} of this line and E_{obs} of the other $k - 1$ lines. To train the model, we generate expert MDP state transitions $\langle s_t, a_t, r_t, s_{t+1}\rangle$ for supervised learning by stochastically "reversing" the MDP from an intermediate term analysis state (Fig. 1D; see Methods for details).

To evaluate the reward function independently of RL, we considered a normalised ranking metric to account for variable $|\mathcal{A}^{(2)}|$. The fractional rank metric is defined as the rank of the expert action among the predicted D scores, normalised by $|\mathcal{A}^{(2)}|$. A score of 1.0 indicates the expert action was top-ranked and random ranking yields 0.5 in expectation. For our validation dataset, the fractional rank averaged 0.91 under a median $|\mathcal{A}^{(2)}| = 87$ (N_{rp} fixed at 2). Thus, expert actions were frequently ranked among the top predicted D scores.

2.6 Case Study MDP Environments and Key Parameters

We investigated four MDP environments from Co II, Nd III, and Nd II term analyses using FT spectral line lists from published studies [2, 3]. We used atomic structure and spectrum calculations that are only achievable given the known levels of the initial state of each environment to accurately represent the challenges of term analysis. Reducing MDP complexity is key for feasibility and RL performance. We removed graph edges with $S/N_{\mathrm{calc}} < 2$, excluded line list entries already matched to edges from $\mathcal{A}^{(2)}$, fixed initial state known levels in energy level optimisations, limited spectral ranges to target groups of levels, and neglected levels unlikely to be determinable (e.g., very highly excited configurations). We also found that a maximum cap on $|\mathcal{A}^{(2)}|$ aids efficient exploration and memory control. If an $a^{(1)}$ induces an $|\mathcal{A}^{(2)}|$ exceeding this cap, a no-op is enforced.

Environment parameters are summarised in Table 1. For Nd III, the initial graph state includes observed FT spectral lines and 40 levels of the $4f^3(4f, 5d)$ configurations revised from [36], excluding the level at 19,403 cm^{-1} with $J = 3$ as it was concluded erroneous [3], and Cowan code [14, 37] calculations (Hartree-Fock method with relativistic corrections) parameterised using these 40 levels [36]. For Co II, the initial graph state includes 141 levels of $3d^7(3d, 4s, 4p, 5s, 4d)$ from Sugar and Corliss (1985) [38] that were revised [2], and we use Cowan code calculations parameterised using these 141 levels. For both Nd II u and Nd II k, the initial graph state includes the six $4f^4(^5I)6s\ ^6I$ ground term levels, the $4f^4(^5I)6s\ ^4I_{9/2}$ level, five $4f^4(^5I)6p\ ^6K$ levels ($J = 9/2$ to $17/2$), and 11 transitions between them. These were chosen for their relatively high eigenvector purities and line intensities. Nd II u uses only ab initio calculations [15], while Nd II k uses Cowan code calculations parameterised using all known levels, representing revision of known energies [4] using more precise measurements. For efficiency, doubly-excited configurations of Nd III, and Nd II levels predicted above 35,000 cm^{-1}, were excluded from the graphs. Setting the $|\mathcal{A}^{(2)}|$ cap was also necessary for Nd II due to large N_{lin}.

The episode length H balances computational feasibility with the ability of the agent to learn beyond short-term consequences. Additionally, an upper limit on H ensures meaningful analysis, as atomic structure calculations can be improved after determining several dozen levels. For example, H is the smaller of the two limits: the maximum allowed by feasible MDP complexity and the maximum imposed by theoretical uncertainties. We found $H = 128$ effective for Nd III and Co II, with no improvement in performance over $H = 64$ for Nd II u, and $H = 512$ viable for Nd II k due to lower ΔE, which is set by domain knowledge.

The ΔI range is estimated at one order of magnitude to account for uncertainties from theoretical transition probabilities and Boltzmann level populations. ΔI is higher for Co II due to averaging of the line lists and charge-transfer effects on level populations [39], and for Nd II u due to ab initio calculations. The repeating candidate energy level tolerance δE approximately matches the highest $\delta\sigma_{\text{obs}}$.

2.7 TAG-DQN Summary

TAG-DQN is a model-free, value-based method based on the DQN [30], as illustrated in Fig. 1C. The agent estimates Q-values of valid actions, from which a policy π can be derived by selecting the a_t with the largest Q-value. Experience tuples $\langle s, a, r, s' \rangle$ are stored in a replay buffer. Model parameters are adjusted by stochastic gradient descent using losses from experience batches sampled from the buffer, with respect to a periodically updated target network. We chose this algorithm class for its higher sample efficiency compared to policy gradient approaches [40]. Furthermore, as we only aim to find a maximum-reward trajectory, a "greedy" policy π with respect to the estimated state-action values is sufficient. Large state and action spaces require deep neural networks for function approximation; we opt for GNNs as a learning representation, as successfully leveraged by other recent works treating graph combinatorial optimisation problems with RL [31]. We also adopt several widely accepted DQN extensions [41] including duelling [42], double Q-learning [43], multi-step returns [44], and noisy networks for exploration [45]. Implementation and hyperparameter details are in Methods.

2.8 Level Determination Accuracy and Baseline Methods

Several solution methods apply to the term analysis MDP. We also evaluate two baseline agents: a greedy search agent which always chooses the maximum reward action, and a standard Monte-Carlo tree search (MCTS) agent [46] with exploration by upper confidence bound for trees (UCT) [47]. Due to large state and action spaces, these discrete search methods operate with a limited horizon and may choose myopic actions. We evaluate TAG-DQN and MCTS across 25 different random seeds and report average performance and 95% confidence intervals in all tables and figures, unless otherwise stated. Greedy search is deterministic. MCTS hyperparameters are tuned as detailed in Methods. Performance is assessed using the following metrics:

- R_{max}: Maximum cumulative reward acquired in a MDP episode with length H.
- N_{c}: Number of levels determined in the R_{max} episode with E_{obs} within $\pm\delta E$ of any previously published energy level E_{known}.
- Accuracy (Acc.): Ratio between N_{c} and the total number of levels determined in the R_{max} episode, which is $N_{\text{c}}/\text{Acc.} \leq \frac{H}{2}$.

We note that N_{c} and Acc. are estimates since these include incorrectly determined levels within the $\pm\delta E$ tolerance and exclude levels unknown in the literature, but we expect these to be in the minority.

Main results for the four case studies are in Table 2. Generally, the highest reward episode corresponded to the highest N_{c}. For Nd II k, more accurate calculations for known levels allowed a reliable secondary metric involving level labels such as J, term symbol, and eigenvector composition, distinguished by level index in the calculations. This is given in the final row of Table 2, where N_{c} is instead the number of determined levels with E_{obs} agreeing with accepted values and human-assigned level indices. Level determination accuracy (Acc.) is concluded to be dependent on the accuracy of theoretical calculations and the alignment between reward and N_{c}. Performance was best in the Co II environment, as expected given its simpler atomic structure and strong influence in reward learning (see Methods).

Greedy search consistently underperformed compared to RL agents. Notably, MCTS achieved higher R_{max} than TAG-DQN, yet TAG-DQN reached higher upper bounds of N_{c}, significantly in the Nd II cases. We interpret this as the ability of TAG-DQN to choose level identities that are more likely to remain consistent with observations and theory throughout the episode, whereas MCTS rollouts were shallow and favoured short-term rewards. This yields higher trajectory rewards that nevertheless have lower accuracy, an interpretation supported by the significantly larger number of correctly labelled levels for TAG-DQN in Nd II k. Incorrect initial level labels are also common for humans, though likely less frequent.

Learning curves and analyses for TAG-DQN in each case study are shown in Fig. 2. From Fig. 2A and B, reward alignment with N_c is evident. Levels of different electron configurations were also determined, as shown in Fig. 2C. The Boltzmann level population and $\Delta I \sim 1$ provides a reasonable basis for line intensity filtering for $\mathcal{A}^{(2)}$, even for Co II where line intensities were averaged over two plasma sources, one exhibiting charge-transfer population enhancements for 4d levels [2, 39]. However, lines with observed intensities deviating significantly from the Boltzmann assumption were excluded from $\mathcal{A}^{(2)}$, as shown in Fig. 2C.

2.9 Ablation Studies

Standard DQN extensions for TAG-DQN were individually disabled (ablated) in the Nd III environment to assess their impacts. Figure 3 shows that all extensions except double Q-learning and prioritised experience replay (PER) improved performance. Double Q-learning was inconclusive despite being recommended in the RL literature. Bias toward high loss subsets of replays from PER was found to be harmful across all case studies, possibly because the buffer size and number of steps required for convergence were around one or two orders of magnitude smaller than values used in environments for which PER was shown beneficial [48]. Duelling is critical for TAG-DQN, likely because of the large action spaces, often hundreds in size per step.

3 Discussion

The proposed usage of TAG-DQN (Fig. 1D) is to first prepare a term analysis state as the initial MDP state, train the reward function using this and/or other term analysis states, train the DQN agent with RL using the reward function, and then use results from the maximum reward $R_{\max}$ episode achieved during RL. If resources permit, running RL with multiple seeds and taking results from the highest reward seed is desirable in maximising the number of correctly determined energy levels N_c. Further improvements may also be achievable with hyperparameter tuning (see Methods).

By design, the final MDP state of the $R_{\max}$ episode after H time steps becomes a new initial state for determining additional levels, though human intervention is expected. This stage allows spectrum inspection, exclusion of poorly measured lines from energy level optimisation, pruning incorrectly determined levels, refining semi-empirical calculations and level labels, and retraining the reward function. For example, Nd II k is the first revision of Nd II energy levels using FT spectra, we include the new levels and lines determined by TAG-DQN from the $R_{\max}$ episode in the online repository. However, these Nd II data should only be treated as tentative (a poor quality MDP initial state) as they currently lack human validation, which is a part of our ongoing Nd II term analysis project.

We used δE to determine $\mathcal{A}^{(2)}$ and the reward function to address influence from spectral lines with outlier wavenumbers (e.g., unknown line blending or poor fitting of the spectral line profiles). However, shifts to determined energy levels from including such outliers in the energy level optimisation remain, as shown in Figure 4, where a small fraction of energy levels significantly deviate from their accepted values but remain within δE. These energy deviations are negligible for most applications at spectral resolving powers $\ll 10^6$. Priority improvements could incorporate raw spectrum and line profile analysis in the MDP to consider additional term analysis factors such as isotope shifts, hyperfine structure, and spectra comparisons. Information on the wavefunction, such as leading configuration label, electron probability density, and nearby level density within its parity and J value may also aid decisions on line profiles and intensities. As our reward function is trained only using data from the Co II MDP (see Methods), a robust reward function trained using a large variety of line lists and MDPs would also be a logical improvement.

Our case studies and methods span diverse term analysis scenarios but omit the most challenging situations, such as single-line level determinations or joining disconnected known-level subgraphs, including term analyses starting with no known levels (applicable to very few low-ionisation species). In these cases, MDP complexity increases significantly, likely requiring careful MDP redesign involving the aforementioned improvements. Our approach is extendable to other experimental methods (e.g., grating spectroscopy) with minor environment adjustments such as modifying δE, ΔI, and the reward function. Lastly, evaluating against published, known levels was far more feasible in our scope compared to carrying out several new term analyses of unknown levels for evaluation. Thus, we anticipate future applications of our methods to better demonstrate their true potential.

Automated fine structure energy level determination is now possible, assisting experts in analyses requiring months to years of effort with traditional techniques. Despite the reduced accuracy compared to humans and dependence on atomic structure calculations, we expect our methods to be incorporated as a key tool for enhancing the completeness and reliability of atomic databases, assuming strict adherence to human evaluation. Beyond atomic spectroscopy, the present work is testament to the potential of graph reinforcement learning [31] and AI in general [49] to assist scientific discovery, particularly when co-developed by scientists and AI researchers. We hope that this work will encourage AI researchers to consider scientific problems as valuable testbeds for developing AI techniques, while also showing scientists how RL can potentially be harnessed for their data- and labour-intensive tasks.

4 Methods

4.1 Level Energy Optimisation

Observed energies $E_{\rm obs}$ of N known levels are determined from M known lines by weighted least-squares minimisation [26]

$$E_{\rm obs} = \underset{E_n}{\rm argmin} \left[\sum_{m=0}^{M-1} w_m \left(\sum_{n=0}^{N-1} S_{mn} E_n - \sigma_m \right)^2 \right], \tag{9}$$

where $w_m = (\delta\sigma_m)^{-2}$, σ_m and $\delta\sigma_m$ are the $\sigma_{\rm obs}$ and $\delta\sigma_{\rm obs}$ of line m, and S_{mn} defines the relationship

$$\sum_{n=0}^{N-1} S_{mn} E_n - \sigma_m = E_u - E_l - \sigma_m, \tag{10}$$

with E_u and E_l as the upper and lower energy levels of line m. The ground energy level E_0 is fixed at zero.

4.2 Graph Feature Design

Level parity and total angular momentum J are distinguished by the unique node indices but excluded from node features, as their selection rules [14] are already encoded by the graph structure, and transition probabilities are edge features. Configuration labels, term labels, and eigenvector compositions are also omitted, since they change with improved semi-empirical atomic structure calculations and are less meaningful under high level densities.

Upper level Boltzmann populations fitted using known lines are extrapolated to estimate $I_{\rm calc}$ on the same scale as $I_{\rm obs}$. The $S/N_{\rm calc}$ is derived from $I_{\rm calc}$ and the estimated ratio between $S/N_{\rm obs}$ and $I_{\rm obs}$ as a function of wavenumber. The line density ρ is the number of lines in the line list within $\sigma_{\rm calc} \pm \Delta E$ with $S/N_{\rm obs}$ within one order of magnitude of $S/N_{\rm calc}$, divided by $2\,\Delta E$. To avoid underestimation for weak lines, $S/N_{\rm calc}$ is clipped to ≥ 10 when computing ρ. All features relating to ρ, S/N, and $gA_{\rm calc}$ are on the log scale for stable learning.

4.3 Reward Learning

The MLP model for predicting D embeds features of each new known line, aggregates the embeddings by summation to reflect higher confidence with more lines, and outputs $D \in [0, 1]$ via a sigmoid activation function. Explicit feature engineering (notably including feature differences) improved generalisation to unseen states, compensating for MLP simplicity and limited training data. The intensities $I_{\rm obs}$ and $I_{\rm calc}$ are normalised by their respective maximum values within each set of k lines, with 10% noise added to $I_{\rm calc}$ to account for calculations for known transitions being generally more accurate. Without intensity normalisation, the mean validation fractional rank was higher but led to lower $N_{\rm c}$ during RL, likely because weaker lines were underrepresented in the training dataset and unknown lines are typically weaker.

We collected 115 expert $a^{(2)}$ MDP state transitions from the Co II environment as the training dataset and 23 from Nd III as the validation dataset, by stochastically reversing the MDP from environment initial states. Each reverse step converts a known level and its lines to unknown, continuing until no cycles remain on the known-level subgraph. To invert the prioritisation of levels with the highest $S/N_{\rm obs}$ lines, known level

removal probabilities were assigned via a softmax over $-2r^{(1)}$, excluding levels whose removal would disconnect the known-level subgraph. This random reversal avoids reliance on unavailable human MDP trajectories and is valid since level determinations have no strict order. Levels outside cycles on the known-level subgraph are also removed before MDP reversal as they cannot appear in $\mathcal{A}^{(1)}$. After removal, the level is marked "selected" in the prior state, with $\mathcal{A}^{(2)}$ computed at $N_{\text{rp}} = 2$, which includes an expert action for reward learning. The next known level removal proceeds after reversing $a^{(1)}$ by flipping the "selected" feature.

In each of the $115 + 23$ training samples, only the expert-chosen $a^{(2)}$ was labelled positive. The MLP model for D contains 105 parameters, trained with Adam [50] to minimise weighted cross-entropy over 32 epochs. The model from the epoch with the lowest validation loss was selected and then used for RL in all four environments of Table 1.

4.4 TAG-DQN Learning Architecture

The TAG-DQN agent consists of one GNN with parameters θ_1 and four MLPs $(V_{\theta_2}, A_{\theta_3}, V_{\theta_4}, V_{\theta_5})$. We denote the union of all parameters by $\theta = \cup_i\{\theta_i\}$ and the parameterised model as Q_θ. Parameters θ are optimised by Adam [50] using a loss computed from a sampled experience batch once every N_{SPT} steps (with SPT being steps per train).

The graph state s, defined by nodes $\mathcal{V}$ and edges $\mathcal{E}$, is processed by multi-head graph attention layers [51], producing node embedding vectors $\text{GNN}(s) = \{\mathbf{h}_v\}_{v\in\mathcal{V}}$. Node and edge features $\mathbf{x}_v$ (2) and $\mathbf{x}_e$ (3), are used directly if continuous or one-hot encoded if discrete. An exponential linear unit (ELU) activation [52] follows each GNN layer except for the final one. The number of heads N_{head} and hidden layer dimension H_{GNN} are the same for all layers, and outputs from each head are concatenated (into dimension $N_{\text{head}} \times H_{\text{GNN}}$). The graph state embedding vector is obtained through mean aggregation

$$\mathbf{s}_{\text{agg}} = \frac{1}{|\mathcal{V}|}\sum_v \mathbf{h}_v. \tag{11}$$

The number of time steps remaining in the episode divided by the horizon H is concatenated with $\mathbf{s}_{\text{agg}}$ (extending its dimension by 1) and then used as input for MLP value estimators.

Separate state value MLPs V_{θ_2} and V_{θ_5} are used in duelling for $a^{(1)}$ and $a^{(2)}$ due to significantly different semantics of the two action types. For $a^{(1)}$, the node embedding vector of each valid level is concatenated with $\mathbf{s}_{\text{agg}}$ for representations (s, a) for the advantage MLP estimator A_{θ_3}. The Q-value for actions of type $a^{(1)}$ is estimated as

$$Q(s,a) = V_{\theta_2}(\mathbf{s}_{\text{agg}}) + A_{\theta_3}(s,a) - \frac{1}{|\mathcal{A}^{(1)}|}\sum_{a'\in\mathcal{A}^{(1)}} A_{\theta_3}(s,a'). \tag{12}$$

For $a^{(2)}$, we explicitly compute each next state s', embed s' via the GNN, mean aggregate this embedding, and then concatenate it with normalised steps remaining to form $\mathbf{s}'_{\text{agg}}$ for advantage A estimation

$$A(s,a) = V_{\theta_4}(\mathbf{s}'_{\text{agg}}) - V_{\theta_5}(\mathbf{s}_{\text{agg}}), \tag{13}$$

where V_{θ_4} is used to estimate the value of the next state s', and the Q-value is estimated in duelling with the same formula as (12). This advantage estimation is possible as the transition function $\mathcal{T}$ (see the Transitions $\mathcal{T}$ subsection of the Results section) is fully deterministic, allowing direct evaluation of s' after applying $a^{(2)}$ to s. All MLPs have two hidden layers of size H_{MLP} with ReLU activations and a scalar output as $|\mathcal{A}^{(1)}|$ and $|\mathcal{A}^{(2)}|$ vary depending on s.

Parameters $\bar{\theta}$ of the target network $Q_{\bar{\theta}}$ are soft-updated after each update of the online network parameters θ [53]

$$\bar{\theta} = (1-\tau)\bar{\theta} + \tau\theta \tag{14}$$

for $\tau \in [0, 1]$. Using n-step returns [44] and double DQN [43], the loss from a replay buffer sample $\langle s_t, a_t, r_t, s_{t+n}\rangle$ is

$$L = [\text{TD}^{[n]}]^2 = [r_t^{[n]} + \gamma_t^n Q_{\bar{\theta}}(s_{t+n}, \underset{a_{t+n}}{\text{argmax}}\, Q_\theta(s_{t+n}, a_{t+n})) - Q_\theta(s_t, a_t)]^2, \tag{15}$$

where

$$r_t^{[n]} = \sum_{k=0}^{n-1} \gamma_t^k r_{t+k}. \tag{16}$$

For exploration, all MLP output layers are noisy layers [45]. Deterministic weights of noisy layers are initialised under a uniform distribution between $\pm(H_{\text{MLP}})^{-0.5}$, weights multiplying noise are initialised at $\sigma_0(H_{\text{MLP}})^{-0.5}$, where $\sigma_0 \approx 0.5$ is a hyperparameter.

4.5 Ablation Protocol

To disable double Q-learning, the target network $Q_{\bar{\theta}}$ was used in (15) for both action selection and value estimation [43]. For ablating duelling, state value estimators V_{θ_2}, V_{θ_5} and advantage calculations were removed, and A_{θ_3} and V_{θ_4} were used directly to estimate $Q(s, a)$. Noisy-network exploration was replaced by standard ϵ-greedy exploration [33, Chapter 2] with ϵ decaying from 1 to 0.1 at a rate of 0.99 per episode. Setting $n = 1$ disables multi-step returns.

4.6 Hyperparameter Tuning

We tuned TAG-DQN hyperparameters in the Nd III environment via grid search with the validation objective of maximising reward. Each set of hyperparameters was evaluated under 5 random seeds after training for 512 episodes ($\sim$ 66K steps). Table 3 summarises the search space and final settings. Due to high dimensionality, only a subset of all possible combinations indicated in Table 3 were investigated. TAG-DQN results from Table 2 were obtained with the final set of hyperparameters in Table 3, except for Nd II k, where $n = 1$ was used as it yielded significant improvements over $n = 2$.

The MCTS [46, 47] exploration and rollout depth parameters were optimised at 0.4 and 4 by grid search over the ranges [0.025, 0.1, 0.4, 0.8, 1.2] and [4, 8, 16], respectively, with 8 seeds per parameter pair. Random action sampling of vanilla MCTS is inefficient in large action spaces where only one action is correct, and low optimal rollout depth is expected as deeper rollouts introduce noise in the returns. For fair comparison between MCTS and DQN, the number of trials per timestep for MCTS is set at 512, and hence the two algorithms encounter the same number of MDP transitions.

4.7 Implementation and Runtime Details

The TAG-DQN agent is built using the PyTorch [54] and PyTorch Geometric [55] libraries. Convergence of TAG-DQN is reached within 24 hours of training. For MCTS, we re-purpose an implementation originally designed for another graph combinatorial optimisation problem (causal structure discovery) [56].

Experiments were carried out using the Imperial College high performance computing facility. Each task was allocated 8 CPU cores, 128 GB memory, and a 24 hour runtime. The full set of experiments presented in this paper, including hyperparameter tuning, would require about 10^5 hours ($\approx$ 11 years) of hypothetical single-core CPU time. Development was carried out on a personal computer with 32 GB RAM under reduced N_{buffer} and GNN complexity, achieving similar single-seed runtime as reported above. TAG-DQN can therefore be run on commodity hardware, but comprehensive experiments that include hyperparameter tuning and multi-seed runs require higher cost infrastructure.

Code Availability

All code used to generate results in this paper is made available on Zenodo (https://doi.org/10.5281/zenodo.18452552) and at the GitHub repository.

Data Availability

All data used to generate results in this paper are made available on Zenodo (https://doi.org/10.5281/zenodo.18452552) and at the GitHub repository.

References

[1] S. Johansson. Term analysis of a complex spectrum. *Physica Scripta*, T65:7–14, 1996.

[2] J. C. Pickering, A. J. J. Raassen, P. H. M. Uylings, and S. Johansson. The spectrum and term analysis of Co II. *The Astrophysical Journal Supplement Series*, 117(1):261, 1998.

[3] M. Ding et al. Spectrum and energy levels of the low-lying configurations of Nd III. *Astronomy & Astrophysics*, 692:A33, 2024.

[4] J. Blaise, J. F. Wyart, M. T. Djerad, and Z. B. Ahmed. Revised interpretation of the spectrum of singly-ionised neodymium (Nd II). *Physica Scripta*, 29:119, 1984.

[5] A. Kramida, Y. Ralchenko, and J. Reader. NIST Atomic Spectra Database (version 5.12). National Institute of Standards and Technology, Gaithersburg, MD, 2024. [Accessed September 2025].

[6] A. Kramida. Legacy of Charlotte Moore Sitterly in the internet age. *Proceedings of the International Astronomical Union*, 18(S371):12–40, 2022.

[7] A. Müller. Fusion-related ionization and recombination data for tungsten ions in low to moderately high charge states. *Atoms*, 3(2):120–161, 2015.

[8] B. A. Marsh. Resonance ionization laser ion sources for on-line isotope separators. *Review of Scientific Instruments*, 85(2):02B923, 2014.

[9] M. S. Safronova, D. Budker, D. DeMille, D. F. Jackson Kimball, A. Derevianko, and C. W. Clark. Search for new physics with atoms and molecules. *Reviews of Modern Physics*, 90:025008, Jun 2018.

[10] J. Hu, J. K. Webb, T. R. Ayres, M. B. Bainbridge, J. D. Barrow, M. A. Barstow, J. C. Berengut, R. F. Carswell, V. Dumont, V. Dzuba, et al. Measuring the fine-structure constant on a white dwarf surface; a detailed analysis of Fe V absorption in G191-B2B. *Monthly Notices of the Royal Astronomical Society*, 500(1):1466–1475, 2021.

[11] J. Atkins and P. V. Baranov. Nailing fingerprints in the stars. *Nature*, 503:437, 2013.

[12] U. Heiter et al. Atomic data for the Gaia-ESO Survey. *Astronomy & Astrophysics*, 645:A106, 2021.

[13] J. J. Cowan et al. Origin of the heaviest elements: The rapid neutron-capture process. *Reviews of Modern Physics*, 93:015002, 2021.

[14] R. D. Cowan. *The Theory of Atomic Structure and Spectra*. University of California Press, Berkeley, CA, 1981.

[15] G. Gaigalas, D. Kato, P. Rynkun, L. Radžiūtė, and M. Tanaka. Extended calculations of energy levels and transition rates of Nd II-IV ions for application to neutron star mergers. *The Astrophysical Journal Supplement Series*, 240:29, 2019.

[16] A. Kramida. Assessing uncertainties of theoretical atomic transition probabilities with Monte Carlo random trials. *Atoms*, 2:86–122, 2014.

[17] F. Concepcion, C. P. Clear, M. Ding, and J. C. Pickering. The laboratory astrophysics programme at Imperial College London. *The European Physical Journal D*, 77:104, 2023.

[18] J. Reader. Atomic spectroscopy at NIST: 2001. In *Proc. Harnessing Light: Optical Science and Metrology at NIST, USA*, 2001.

[19] W.-Ü. L. Tchang-Brillet and V. I. Azarov. Recent laboratory studies of multiply charged ion spectra using high resolution VUV spectrographs. *Physica Scripta*, T100:104–113, 2002.

[20] V. I. Azarov. Formal approach to the solution of the complex-spectra identification problem. I. Theory. *Physica Scripta*, 44:528, 1991.

[21] V. I. Azarov. Formal approach to the solution of the complex-spectra identification problem. II. Implementation. *Physica Scripta*, 48:656, 1993.

[22] L. Engström. GFit, a computer program to determine peak positions and intensities in experimental spectra. Lund Reports in Atomic Physics; Vol. LRAP-232, Atomic Physics, Department of Physics, Lund University, 1998. [Accessed August 2025].

[23] G. Nave, U. Griesmann, J. Brault, and M. Abrams. Xgremlin: Interferograms and spectra from Fourier transform spectrometers analysis. Astrophysics Source Code Library, record ascl:1551.004, 2015. [Accessed August 2025].

[24] M. Ding, S. Z. J. Lim, X. Yu, C. P. Clear, and J. C. Pickering. A neural network approach for line detection in complex atomic emission spectra measured by high-resolution Fourier transform spectroscopy. *Machine Learning: Science and Technology*, 6:035008, 2025.

[25] V. I. Azarov, A. Kramida, and M. Y. Vokhmenstev. IDEN2 - a program for visual identification of spectral lines and energy levels in optical spectra of atoms and simple molecules. *Computer Physics Communications*, 225:149–153, 2018.

[26] A. Kramida. The program LOPT for least-squares optimization of energy levels. *Computer Physics Communications*, 182:419–434, 2011.

[27] J. Connes et al. Spectroscopie de Fourier avec transformation d'un million de points. *Nouvelle Revue d'Optique Appliquée*, 1(1):3, 1970.

[28] K. L. Peterson and M. L. Parsons. Spectral classification using pattern-recognition techniques. I. Feasibility with hydrogen as a model system. *Physical Review A*, 17(1):261, 1978.

[29] K. L. Peterson, D. L. Anderson, and M. L. Parsons. Spectral classification using pattern-recognition techniques. II. Application to curium energy levels. *Physical Review A*, 17(1):270, 1978.

[30] V. Mnih et al. Human-level control through deep reinforcement learning. *Nature*, 518:529–533, 2015.

[31] V.-A. Darvariu, S. Hailes, and M. Musolesi. Graph reinforcement learning for combinatorial optimization: A survey and unifying perspective. *Transactions on Machine Learning Research*, 2024.

[32] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini. The graph neural network model. *IEEE Transactions on Neural Networks*, 20:61–80, 2009.

[33] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 2018.

[34] D. S. Sivia and J. Skilling. *Data analysis: a Bayesian tutorial*. Oxford University Press, 2nd edition, 2006.

[35] A. Y. Ng and S. J. Russell. Algorithms for Inverse Reinforcement Learning. In *Proc. International Conference on Machine Learning*, 2000.

[36] T. Ryabchikova, A. Ryabtsev, O. Kochukhov, and S. Bagnulo. Rare-earth elements in the atmosphere of the magnetic chemically peculiar star HD 144897. *Astronomy & Astrophysics*, 456:329–338, 2006.

[37] A. Kramida. A suite of atomic structure codes originally developed by R. D. Cowan adapted for Windows-based personal computers. National Institute of Standards and Technology, 2021. [Accessed August 2025].

[38] J. Sugar and C. Corliss. *Atomic energy levels of the iron-period elements: potassium through nickel*. American Chemical Society, Washington, DC, 1985. PB-86-165446/XAB/.

[39] S. Johansson and U. Litzén. Possibilities of obtaining laser action from singly ionised iron group elements through charge transfer in hollow cathode lasers. *Journal of Physics B: Atomic and Molecular Physics*, 13:L253, 1980.

[40] R. S. Sutton, D. McAllester, S. Singh, and Y. Mansour. Policy gradient methods for reinforcement learning with function approximation. In *Proc. Conference on Neural Information Processing Systems*, 1999.

[41] M. Hessel et al. Rainbow: Combining Improvements in Deep Reinforcement Learning. In *Proc. AAAI Conference on Artificial Intelligence*, 2018.

[42] Z. Wang. Dueling network architectures for deep reinforcement learning. In *Proc. International Conference on Machine Learning*, 2016.

[43] H. van Hasselt, A. Guez, and D. Silver. Deep reinforcement learning with double Q-learning. In *Proc. AAAI Conference on Artificial Intelligence*, 2016.

[44] R. S. Sutton. Learning to predict by the methods of temporal differences. *Machine Learning*, 3:9–44, 1988.

[45] M. Fortunato. Noisy networks for exploration. In *Proc. International Conference on Learning Representations*, 2018.

[46] C. Browne et al. A survey of Monte Carlo tree search methods. *IEEE Transactions on Computational Intelligence and AI in Games*, 4(1), 2012.

[47] L. Kocsis and C. Szepesvári. Bandit based monte-carlo planning. In *Proc. European Conference on Machine Learning*, 2006.

[48] T. Schaul, J. Quan, I. Antonoglou, and D. Silver. Prioritized experience replay. In *Proc. International Conference on Learning Representations*, 2016.

[49] H. Wang et al. Scientific discovery in the age of artificial intelligence. *Nature*, 620(7972):47–60, 2023.

[50] D. P. Kingma and J. Ba. Adam: a method for stochastic optimization. In *Proc. International Conference on Learning Representations*, 2018.

[51] S. Brody, U. Alon, and E. Yahav. How attentive are graph attention networks? In *Proc. International Conference on Learning Representations*, 2022.

[52] D.-A. Clevert, T. Unterthiner, and S. Hochreiter. Fast and accurate deep network learning by exponential linear units (ELUs). In *Proc. International Conference on Learning Representations*, 2016.

[53] T. P. Lillicrap et al. Continuous control with deep reinforcement learning. In *Proc. International Conference on Learning Representations*, 2016.

[54] A. Paszke et al. PyTorch: an imperative style, high-performance deep learning library. In *Proc. Conference on Neural Information Processing Systems*, 2019.

[55] M. Fey and J. E. Lenssen. Fast graph representation learning with PyTorch geometric. In *Proc. International Conference on Learning Representations*, 2019.

[56] V.-A. Darvariu, S. Hailes, and M. Musolesi. Tree search in DAG space with model-based reinforcement learning for causal discovery. In *Proc. of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 2025.

Acknowledgements

M.D. and J.C.P. acknowledge support from the Science and Technology Facilities Council (STFC) of the UK under grant numbers ST/N000939/1, ST/S000372/1, ST/W000989/1, and UKRI1188, and The Bequest of Prof. Edward Steers. V.-A.D. and N.H. acknowledge support from the Natural Environment Research Council (NERC) Twinning Capability for the Natural Environment (TWINE) Programme NE/Z503381/1, the Engineering and Physical Sciences Research Council (EPSRC) From Sensing to Collaboration Programme Grant EP/V000748/1, and the Innovate UK AutoInspect Grant 1004416. A.N.R is grateful to the late Dr J.-F. Wyart for help in Nd II calculations and to the support from research project FFUU-2025-0005 of the Institute of Spectroscopy of the Russian Academy of Sciences.

Author contributions

M.D. – planning, method design, execution, Co II calculations, manuscript preparation.
V.-A.D. – planning, method design, manuscript preparation.
A.N.R. – Nd II-III calculations and manuscript review.
N.H. – resource management, manuscript review.
J.C.P. – resource management, progress and manuscript review.

Competing interests

The authors declare that they have no competing interests.

List of Figures

1 Illustration of the MDP environment and TAG-DQN 16
2 Learning results of TAG-DQN in the four case studies of Table 1 17
3 Learning curves of TAG-DQN (black) and its variants in the Nd III environment 18
4 Difference between energy levels determined from a single seed and their known (accepted) values for each species 19

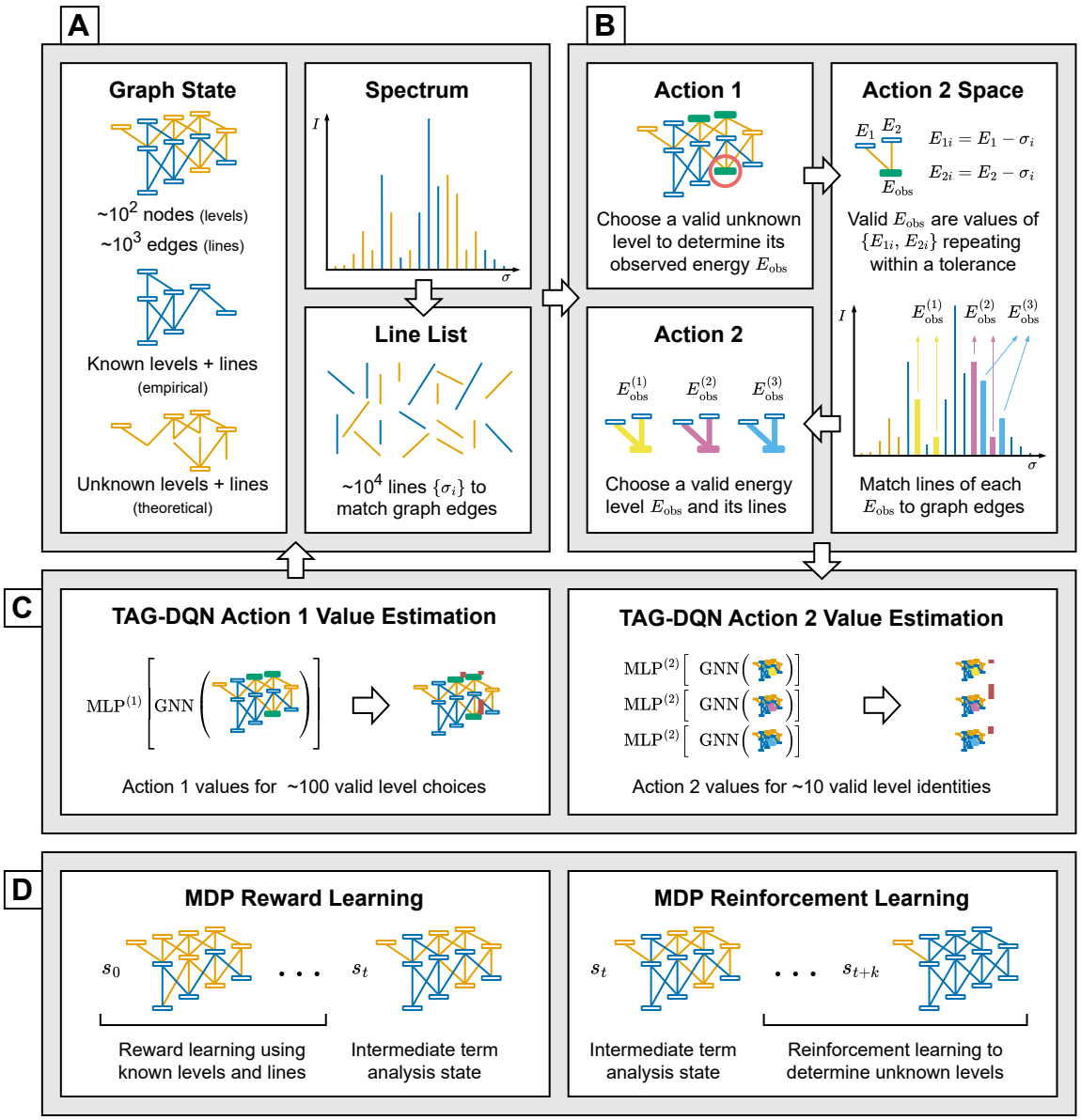


Figure 1: **Illustration of the MDP environment and TAG-DQN. A** The term analysis state is represented as a graph with node and edge features, alongside the spectral line list. **B** Actions alternate between two regimes; each pair of actions leads to the determination of the observed energy $E_{\rm obs}$ for one level by matching at least two lines from the line list to unknown edges in the graph. **C** TAG-DQN employs a GNN to embed graph representations, which are inputs for multilayer perceptrons (MLPs) estimating Q-values for each action (vertical bars); the highest Q-value action advances the MDP to the next state. **D** Given a term analysis state s_t, the MDP trajectory leading to s_t involving known levels is used for reward learning, while RL with the learned reward function guides the discovery of unknown levels in future states s_{t+k}.

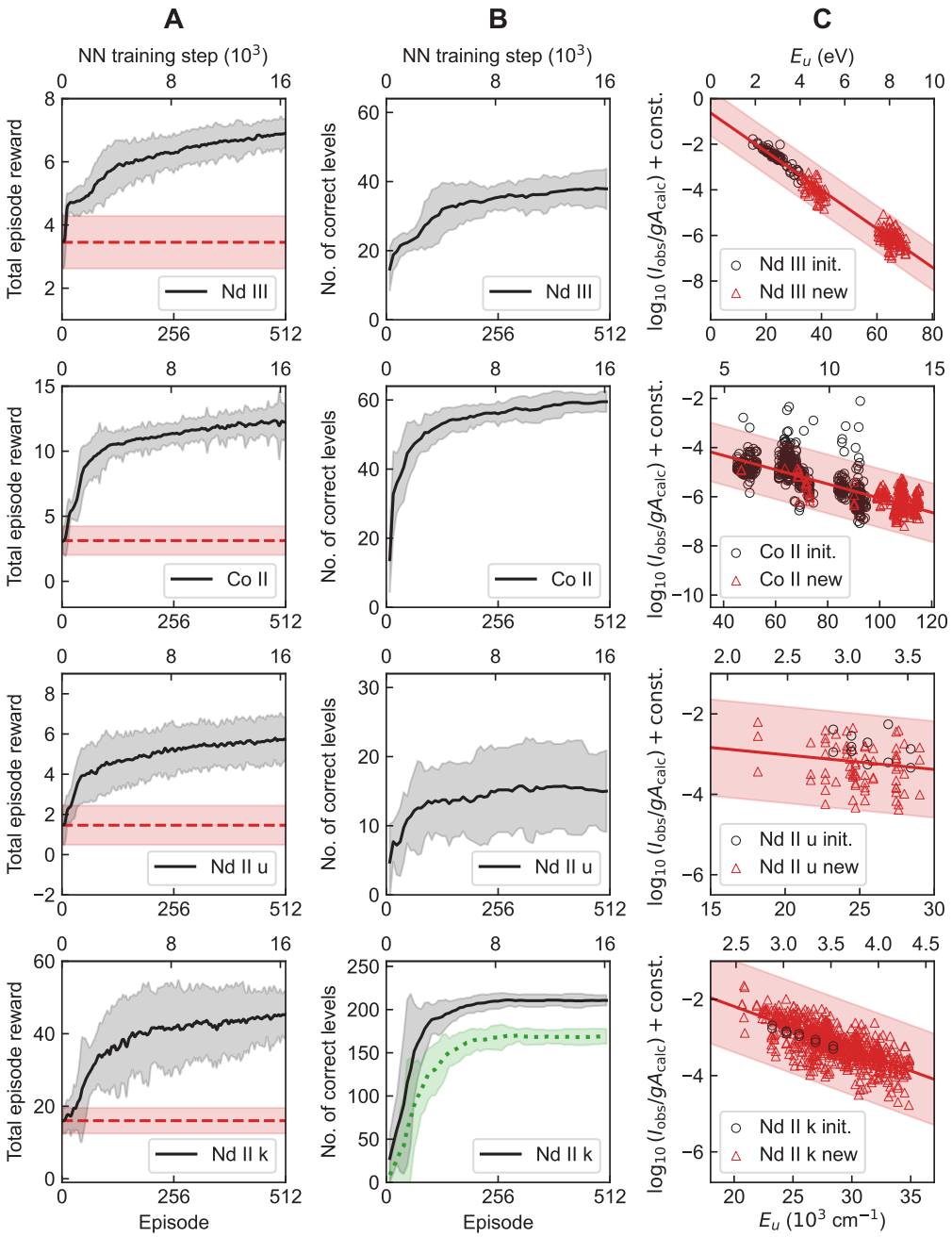


Figure 2: **Learning results of TAG-DQN agent in the four case studies of Table 1. A** Average learning curves, i.e., mean cumulative rewards obtained as a function of neural network (NN) training steps. The dashed horizontal lines show reward obtained during the pre-training episodes and correspond to rewards obtained by choosing actions in the MDP uniformly at random. The shaded regions show ±2 standard deviations across 25 random seeds. **B** Average N_c of the final state of the most recent maximum reward episode. The dotted curve for Nd II k shows the number of correct levels that also match human chosen level labels, and the y-axes limits are $\frac{H}{2}$. The shaded regions show ±2 standard deviations across 25 random seeds. **C** Boltzmann plot (log of relative level population against upper energy level E_u) using known lines from the final state of the maximum reward episode of a single seed; circles show initial state known lines, triangles show new known lines from RL, and the shaded region of the linear fit shows $\pm\Delta I$.

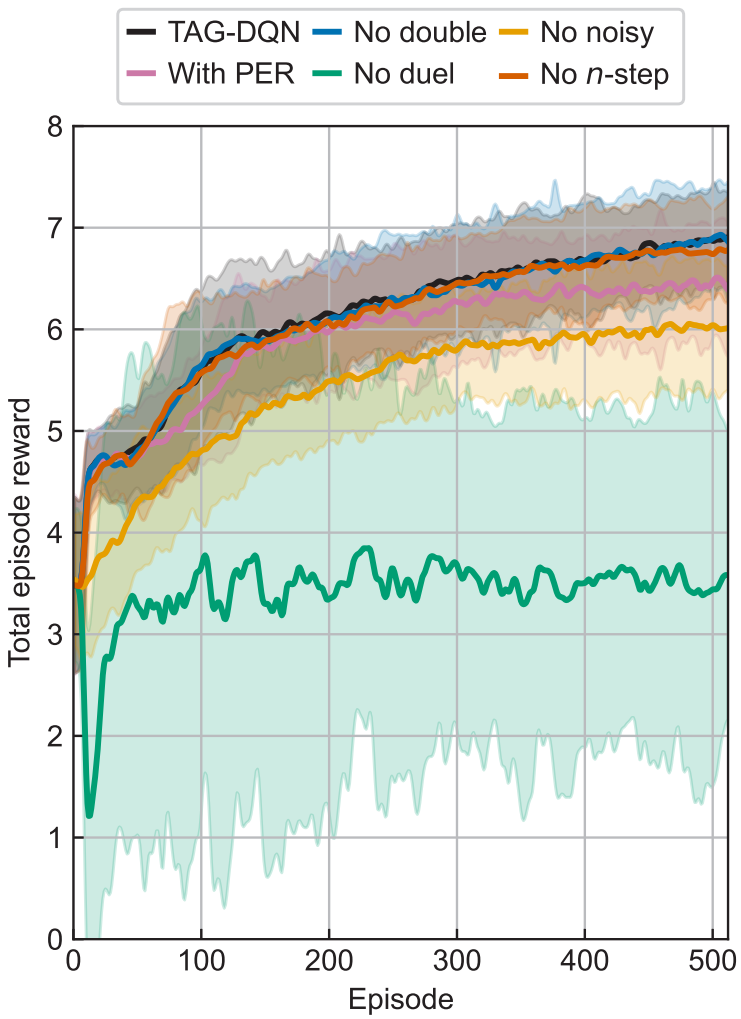


Figure 3: **Learning curves of TAG-DQN (black) and its variants in the Nd III environment.** Average total episode reward during training of TAG-DQN and its designs with or without particular deep Q-network extensions: without double Q-learning (blue), without noisy networks for exploration (yellow), with prioritised experience replay (PER, pink), without duelling deep Q-network (green), and without multi(n)-step return (orange). Shaded regions show ±2 standard deviations across 16 seeds. The duelling extension and noisy networks for exploration are key for TAG-DQN, while using PER reduced performance slightly.

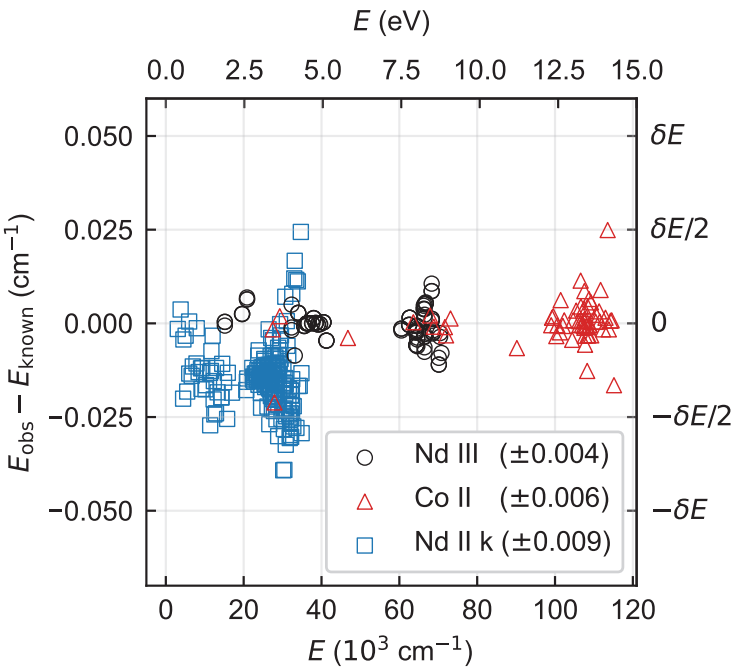


Figure 4: **Difference between energy levels determined from a single seed and their known (accepted) values for each species**. Only levels contributing to N_c are shown. Energy differences for Nd III, Co II, and Nd II k are shown in circles, triangles, and squares, respectively. The root-mean-square energy differences are given in parentheses in the legend. Typical energy level uncertainty by FT spectroscopy is of order 0.001 cm^{-1}. The offset and higher spread of Nd II k energies are expected and within uncertainties as their known values were derived by lower-precision grating spectroscopy [4].

List of Tables

1 Key term analysis MDP environment parameters 21
2 Results and comparisons with benchmark agents 21
3 TAG-DQN parameter ranges and final hyperparameters 21

Table 1: Key term analysis MDP environment parameters.

Case[a]	Range[b] (10^3 cm^{-1})	N_{lin}[b] (10^3)	$(\lvert\mathcal{V}\rvert, \lvert\mathcal{E}\rvert)$[c]	H[d]	ΔE[e] (cm^{-1})	ΔI	[f] δE[g] (cm^{-1})	$\lvert\mathcal{A}^{(1)}\rvert$ range[h]	$\lvert\mathcal{A}^{(2)}\rvert$ median[h]	$\lvert\mathcal{A}^{(2)}\rvert$ max[h]
Nd III	$30-55$	3	(600, 1000)	128	1500	1.0	0.05	$10-60$	6	256
Co II	$34-83$	4	(900, 4000)	128	1500	1.2	0.05	$90-150$	3	256
Nd II u	$10-55$	22	(700, 1800)	64	3000	1.2	0.05	$60-200$	5	32
Nd II k	$10-55$	22	(500, 2200)	512	250	1.0	0.05	$0-200$	3	32

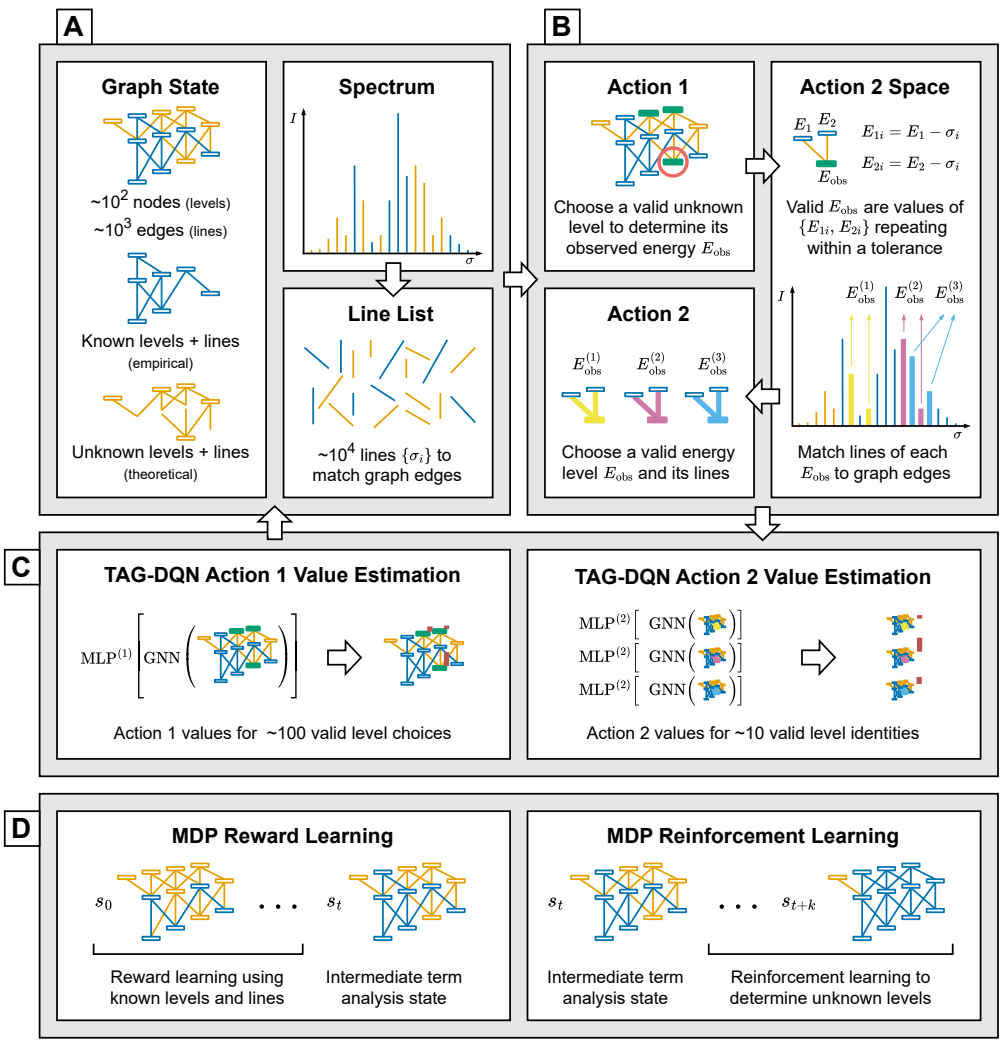
[a] The suffixes u and k denote different theoretical methods, see text.
[b] Line list spectral range, within which N_{lin} lines were used for $\mathcal{A}^{(2)}$ determination.
[c] Graph size in terms of (number of nodes, number of edges).
[d] Horizon, the maximum number of steps in the MDP.
[e] Theoretical wavenumber uncertainty tolerance.
[f] Theoretical and experimental line intensity matching tolerance (order-of-magnitude).
[g] Experimental wavenumber uncertainty tolerance.
[h] Action space sizes for the two action types, range and median were measured over the history of TAG-DQN training, not from uniform random MDP state transitions, the maximum of action space 2 is a setting.

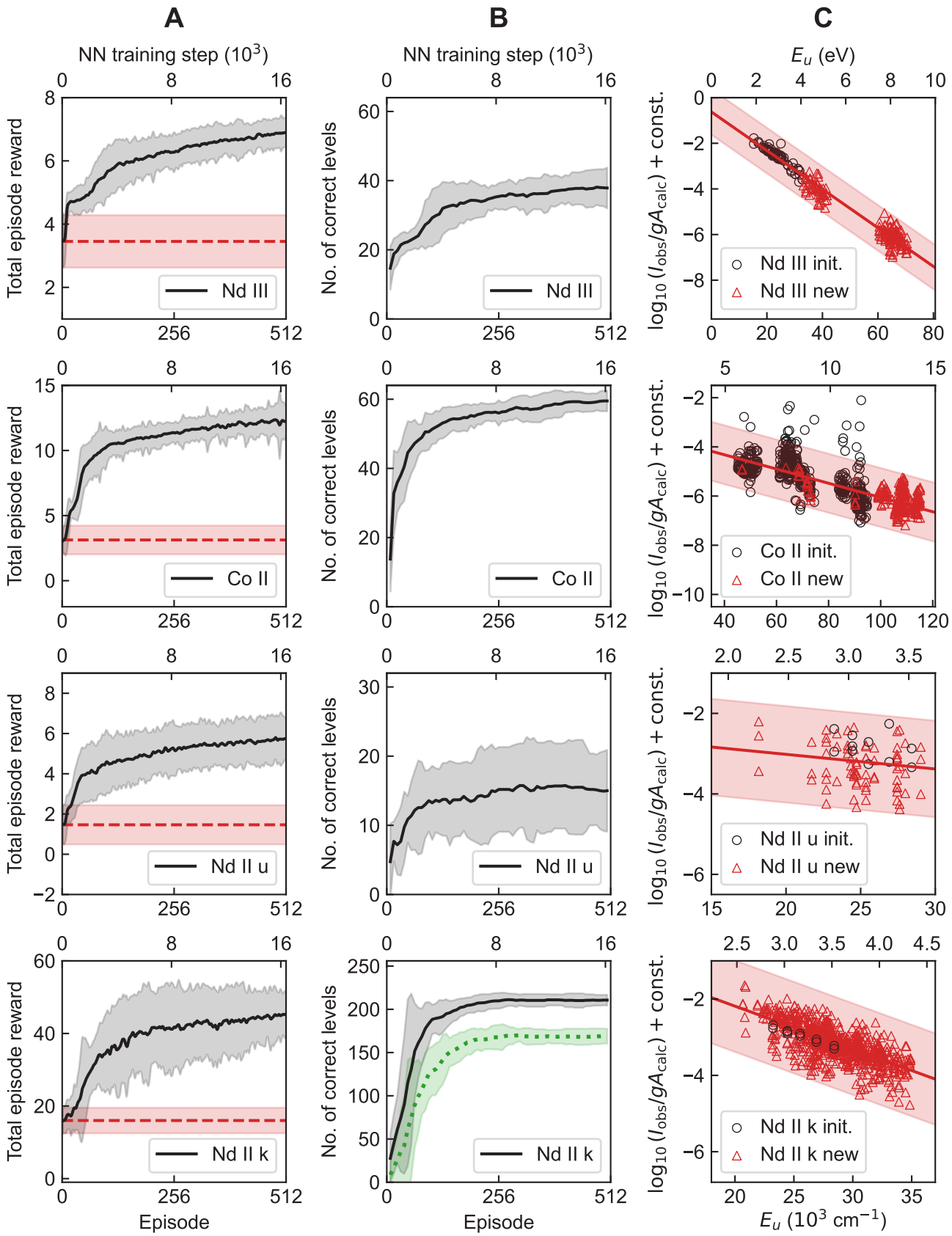
Table 2: Results and comparisons with benchmark agents. TAG-DQN matches MCTS performance as judged by downstream metrics in 2/5 cases and outperforms it in 3/5 cases, while Greedy search is worse overall.

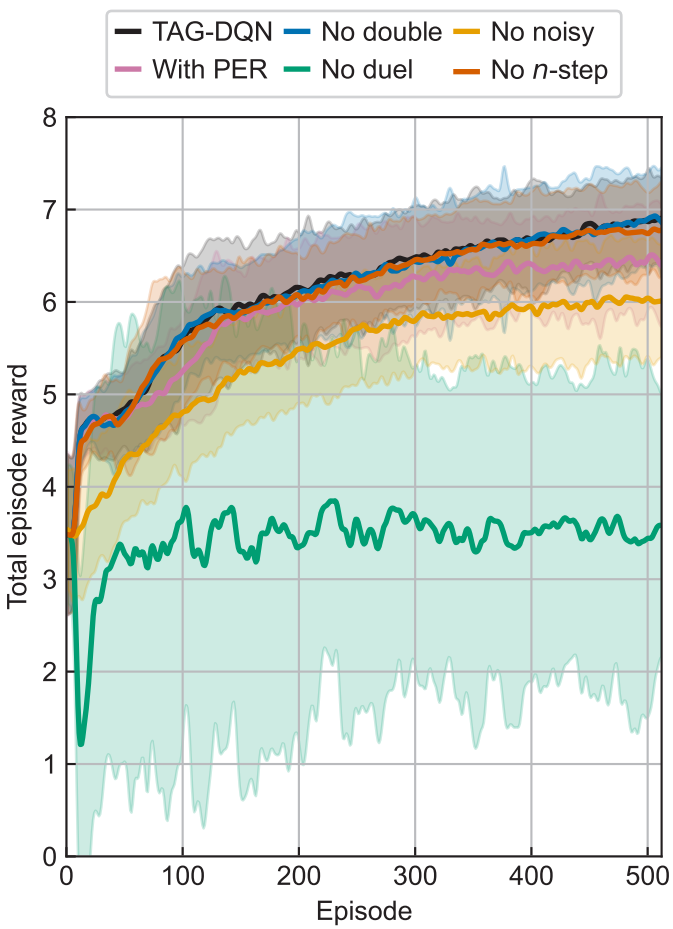
Case	Greedy search			MCTS			TAG-DQN		
	$R_{\max}$	N_{c}	Acc.	$R_{\max}$	N_{c}	Acc.	$R_{\max}$	N_{c}	Acc.
Nd III	7.3	28	0.52	$8.5_{\pm 0.9}$	$37_{\pm 4}$	$0.58_{\pm 0.06}$	$7.2_{\pm 0.3}$	$37_{\pm 5}$	$0.59_{\pm 0.08}$
Co II	9.9	41	0.84	$14.9_{\pm 0.7}$	$61_{\pm 2}$	$0.97_{\pm 0.03}$	$13.2_{\pm 1.1}$	$60_{\pm 3}$	$0.95_{\pm 0.02}$
Nd II u	6.6	8	0.38	$8.4_{\pm 0.1}$	$9_{\pm 6}$	$0.30_{\pm 0.18}$	$6.8_{\pm 0.7}$	$15_{\pm 6}$	$0.54_{\pm 0.21}$
Nd II k	51.3	184	0.79	$53.7_{\pm 2.1}$	$185_{\pm 9}$	$0.77_{\pm 0.02}$	$48.8_{\pm 0.8}$	$210_{\pm 7}$	$0.87_{\pm 0.03}$
Nd II k corr. label	-	132	0.57	-	$130_{\pm 11}$	$0.54_{\pm 0.04}$	-	$169_{\pm 10}$	$0.69_{\pm 0.05}$

Table 3: TAG-DQN parameter ranges and final hyperparameters.

Parameter	Search	Final
No. of GNN layers	[1, 2, 3, 4]	3
H_{GNN}	[16, 32, 64]	32
N_{head}	[1, 2, 4, 8]	4
H_{MLP}	[8, 16, 32, 128, 256]	32
Adam learning rate ($\times 10^{-4}$)	[50, 10, 5, 1]	10
Replay batch size	[4, 8, 16, 32]	16
Soft target update rate τ	[0.05, 0.01, 0.005, 0.001]	0.001
Steps per train N_{SPT}	[2, 4, 16]	$H/32$
Noisy nets σ_0	[0.1, 0.5]	0.5
Replay buffer capacity N_{buffer}	10K steps	10K steps
Min. history to start learning	8 episodes	8 episodes
Multi-step returns n	[1, 2, 3, 4]	2
Discount factor γ	0.99	0.99


A
Graph State
~10^2 nodes (levels)
~10^3 edges (lines)
Known levels + lines
(empirical)
Unknown levels + lines
(theoretical)
Spectrum
I
σ
Line List
~10^4 lines {σ_i} to
match graph edges
B
Action 1
Choose a valid unknown
level to determine its
observed energy E_obs
Action 2
Choose a valid energy
level E_obs and its lines
Action 2 Space
E_1 E_2
E_1i = E_1 − σ_i
E_2i = E_2 − σ_i
E_obs
Valid E_obs are values of
{E_1i, E_2i} repeating
within a tolerance
Match lines of each
E_obs to graph edges
C
TAG-DQN Action 1 Value Estimation
MLP^(1) [GNN ( )]
Action 1 values for ~100 valid level choices
TAG-DQN Action 2 Value Estimation
MLP^(2) [GNN ( )]
Action 2 values for ~10 valid level identities
D
MDP Reward Learning
s_0
s_t
Reward learning using
known levels and lines
Intermediate term
analysis state
MDP Reinforcement Learning
s_t
s_t+k
Intermediate term
analysis state
Reinforcement learning to
determine unknown levels


A
B
C
NN training step (10^3)
NN training step (10^3)
E_u (eV)
Total episode reward
No. of correct levels
$\log_{10}(I_{obs}/gA_{calc}) + \text{const.}$
Nd III
Nd III init.
Nd III new
Co II
Co II init.
Co II new
Nd II u
Nd II u init.
Nd II u new
Nd II k
Nd II k init.
Nd II k new
Episode
Episode
E_u (10^3 cm^{-1})


TAG-DQN
No double
No noisy
With PER
No duel
No n-step
Total episode reward
Episode
0
1
2
3
4
5
6
7
8
0
100
200
300
400
500