Review | Received 14 April 2025; Accepted 23 May 2025; Published 6 June 2025
https://doi.org/10.55092/aimat20250010

Materials discovery through reinforcement learning: a comprehensive review

Nazir Ahmed[1,2], Muhammad Umar Farooq[1,2] and Fuyi Chen[1,2,*]

[1] School of Materials Science and Engineering, Northwestern Polytechnical University, Xian, China

[2] Solid State Key Laboratory of Solidification Processing, Northwestern Polytechnical University, Xian, China

* Correspondence author(s); E-mail: fuyichen@nwpu.edu.cn

Highlights:

- Application of reinforcement (RL) learning in catalysis and material design is being reviewed.
- RL frameworks are explored for material optimization and nanomaterials design.
- Comparative analysis of reinforcement learning and traditional computational methods.

Abstract: Reinforcement learning (RL) is emerging as a powerful tool in materials science, delivering a paradigm shift in how we find and optimize high-dimensional chemical and structural spaces. Unlike traditional methods, RL agents can learn to explore complex energy landscapes in an adaptive manner, making instant decisions that guide the discovery of novel materials with specific properties. However, the application of RL to materials discovery faces unique challenges, including data scarcity, computationally expensive, and the challenge of designing reward functions that can balance multiple material objectives optimally. In this review, the current challenges and difficulties in applying RL techniques in materials science, and recent advances combining RL with machine learning, generative models, and domain knowledge are emphasized. We also outlined promising future directions, such as transfer learning, hybrid models, and the creation of collaborative, open-access data infrastructures. By addressing these challenges, RL has the potential to transform the discovery and design of functional materials for catalysis, energy storage, and sustainability applications.

Keywords: cluster design; nanomaterials; material design; reinforcement learning; AI-driven cluster generation.

1. Introduction

The unique properties of nanoclusters and nanoalloys provide unparalleled potential for innovation and technological progress based on their size-dependent properties and quantum confinement effects, enabling

Copyright©2025 by the authors. Published by ELSP. This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium provided the original work is properly cited

innovation in fields ranging from clean energy and catalysis to electronics and medicine. Nanomaterials, specifically nanoclusters and nanoalloys, have attracted enormous interest because of their size and composition-dependent properties, which are different from those of bulk materials. These nanoclusters exhibit distinct electronic, optical, magnetic, and chemical properties driven primarily by quantum confinement effects and their very high surface-to-volume ratio [1,2].

However, optimal use of such properties depends on accurate knowledge of the atomic-scale structure, since the performance of nanoclusters is inherently governed by their geometric and energetic configurations. One of the central challenges in this context is the identification of the global minimum (GM) structure, the most thermodynamically stable arrangement of atoms. The GM search is extremely difficult due to the vast number of possible atomic configurations, which increases exponentially with the size and composition of the cluster. Consequently, the potential energy surface (PES) of nanoclusters is often rugged and multidimensional, with numerous local minima, making it particularly challenging for conventional optimization methods to reliably converge to the true GM structure.

Traditional methods such as genetics (GA) and basin hopping (BH) are extremely popular for global minimum searches in nanoclusters. GA has been successfully applied in nanoalloy systems, particularly Au–Ag nanoalloys, to identify stable surface configurations that influence their structural and electronic properties [3,4]. However, GAs often converge slowly, lose diversity, and become trapped in local minima, especially as the complexity of the system increases [5]. Basin hopping has also been utilized in many applications, demonstrating its utility in optimizing atomic arrangements [6,7]. However, it remains less effective in high-dimensional landscapes due to step size sensitivity and inefficiency in more complex systems, indicating the need for a more advanced and scalable approach [8,9].

In recent years, machine learning (ML) and artificial intelligence have revolutionized materials discovery by enabling data-driven approaches to discover new materials more efficiently. Among the various ML techniques, reinforcement learning (RL) has emerged as one of the most promising paradigms for guiding the search for new materials. In nanomaterials discovery, RL is particularly effective in navigating the vast and complex space of potential material configurations, identifying ideal atomic structures, and predicting accompanying properties, such as stability, electron behavior, and mechanical resilience [10,11]. RL is capable of accelerating the discovery of stable materials, and the understanding of their mechanisms substantially by structurally optimizing structural configurations and energy landscapes.

RL has proven to have significant potential to optimize chemical reactions, synthesize new molecules, and predict the properties of materials, and thus makes it a viable tool in chemistry and material science [12]. RL has increasingly been applied in materials science, particularly in recent years. There has been a steady increase in the applications of RL algorithms in the field of material science since their initial adoption in 2019, as illustrated in Figure 2, based on publication data retrieved from the Scopus database [https://www.scopus.com/sources.uri] . This represents a growing awareness of the potential of RL to optimize the material properties, and to improve the computational effectiveness in materials science. The increased usage reflects gains in machine learning techniques, improved computational power, and increased usage of AI-driven methodologies in scientific research.

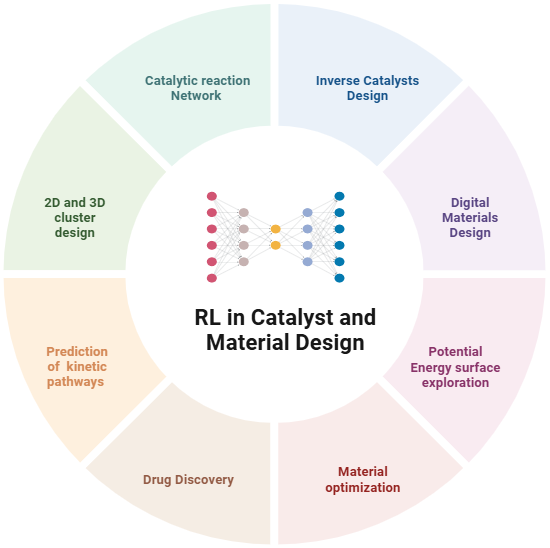

Figure 1. Application of Reinforcement Learning in Catalyst and Material Design.

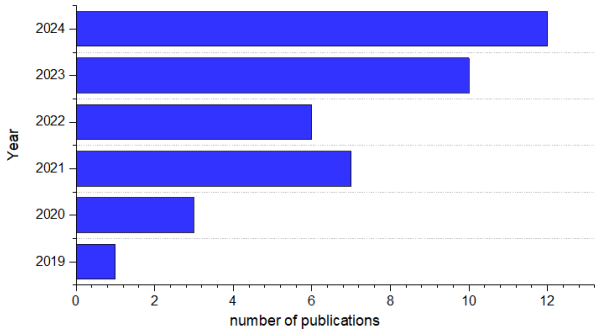

Figure 2. Adoption of reinforcement learning in materials research

2. Theoretical foundations of reinforcement learning frameworks

The RL process is described as a loop generating a sequence of actions, states, and rewards. Based on the Markov Decision Process (MDP), the agent chooses an action a_t in every time step from a set of possible actions, the action space A, based on the current state s_t from the state space S. In response to the selected action, the environment provides a reward r_t for the state-action pair (s_t, a_t), and transitions to the next state according to the transition probability $P(s_{t+1}|s_t, a_t)$ [13]. The agent learns a policy π_t, which is a probabilistic distribution over actions conditioned on the states. This policy maps the state space S to the action space A, and functions as the agent's "brain," directing its decision-making process at each time step [14,15]. The MDP can be mathematically represented as,

$$\text{MDP} = \langle S, A, P(s_{t+1} \mid s_t, a_t), R(s_t, a_t), s_0 \rangle$$

In any RL setup, it is essential to explicitly define the state space, action space, and reward function, as they form the core components of the learning environment. The state space represents the set of all possible environmental states, where each state contains the environment's present state. In the work by Simm *et al* [16], the state space is made up of two parts: a canvas C_t and a "bag" of atoms, as given in equation 1. The canvas refers to the current structure containing all the atoms that have already been placed by the agent up to that point.

$$S_t = \langle C_t, t \rangle \tag{1}$$

$$C_t = C_0 \cup \{(e_i, x_i)\}_{i=0}^{t-1} \qquad t = \{e, m(e)\}$$

Where (e_i, x_i) each corresponds to an atom with chemical element e_i and spatial position x_i, representing all atoms that have been placed up to time $(t-1)$, and m(e) denotes the multiplicity of each chemical element. The second part of the state is the bag t, where the available atoms are stored. For each type of atoms in the bag, there is a number indicating how many of that type of atoms are

left. This helps the agent keep track of atoms that are left to use, and guides its choices as it constructs molecules or materials. Together, the canvas and the bag fully describe the current state of the system at any point in the construction process.

In the study by Raju *et al*, the state space represents the current configuration of the nanocluster using atom-centered symmetric functions (ACSFs) [17], which encode structural features from two- and three-body atomic interactions. The state also includes energies, forces, and binary vectors that indicate conditions such as overlapping atoms, dissociation, local minima, and convergence (e.g., discovery of five distinct local minima). These elements are processed through neural networks to form a state embedding, which is fed into the reinforcement learning agent.

In this framework, action space includes all possible actions that the agent can take. In the study by Raju *et al*, it is defined as a discrete set of two actions: The first action involves selecting one atom at random from the entire cluster. The second action determines how that selected atom will move, choosing between a displacement of +2.0 Å or -2.0 Å. This allows the agent to explore different structural configurations by shifting atoms within the cluster [17].

$$a_t = \left(a_t^1,\, a_t^2\right) \tag{2}$$

Simm *et al*, defines the action space as decisions made by the agent at each step of molecule construction[18]. At each time step, the agent selects one atom and chooses a specific position in 3D space to place that atom. Each action tells the system which atom to place next and exactly where it should go in the structure. This process continues until all atoms have been placed to complete the molecules.

The design of an effective reward function is essential for the performance of an RL model. It is typically more effective when it provides a gradient. This will allow the agent to better understand when it gets closer or farther from the target. In their study, Matignon *et al.* [19] proposed a reward function based on a Gaussian distribution that assigns constant rewards to distant states s from the goal Sg to avoid destabilizing the learned policies. In this formulation, the parameter β controls the reward amplitude, while σ determines the standard deviation, which defines the extent of the region influenced by gradients (Equation 3).

$$R(s,u,s') = \beta\, e^{-\frac{d(s,s_g)^2}{2\sigma}} \tag{3}$$

Spielberg *et al.* [20] proposed a reward function formula, which is specifically designed for process control problems, as they are often set-point tracker-based. When the process output is within the error tolerance of the set-point, the agent receives the maximum reward c. If the output deviates beyond this tolerance, the agent is penalized with a negative reward proportional to the deviation from the set point. The regions of the state space where negative rewards dominate can lead the agent to prioritize reaching a termination point quickly to minimize negative rewards. Nevertheless, negative values can be added to the reward to accelerate the learning phase. Thus, selecting the appropriate negative reward amplitude is crucial to balance effective exploration and efficient learning.

$$R(s,u,s') = \begin{cases} c, & \varepsilon > |s_{g,i} - s_i|, \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

In developing the DRL framework for exploring nanocluster configurations, as proposed by K. Raju *et al.* [17], the reward function is carefully designed to guide the agent effectively toward the identification of multiple local energy minima within PES. The reward design combines both incentives and penalties to shape the agent's behavior during exploration. A penalty of -10 is applied for undesirable actions, such as cluster dissociation, atom overlapping, and revisiting previously visited minima. Conversely, if the agent discovers a new local minimum with energy lower than the initial configuration, it receives a positive reward calculated by:

$$\text{Reward} = \Delta E \cdot 1000 \tag{5}$$

where Δ E is the relative energy with respect to the initial configuration in eV. Specifically, the reward is greater if the new minimum has a lower energy level, thus incentivizing the discovery of energetically favorable configurations. If the agent identifies a new configuration that has a higher energy value than the initial setup, it will receive no reward (*i.e.*, reward = 0). This discourages the exploration of less favorable energy states. This reward structure is carefully designed to balance between penalizing unproductive explorations, and rewarding the discovery of new, energetically favourable configurations, thus guiding the DRL agent in the efficient exploration of nanocluster configuration space.

Table 1. Conceptual comparison between on-policy and off-policy RL methods.

Aspect	**On-policy**	**Off-policy**
Principal data source	Trajectories generated by the current policy π	Experience generated by a behaviour policy μ (which may differ from π)
Sample efficiency	Usually lower (requires fresh interaction)	Usually higher (can reuse old data / replay buffers)
Stability & convergence	Typically more stable; lower risk of divergence	Can be less stable due to distribution shift; requires careful off-policy correction
Exploration–exploitation	Same policy governs both	Can explore with one policy while learning another
Bias–variance trade-off	Lower bias, higher variance	Potentially higher bias from off-policy correction, but lower variance
Typical learning update	On-policy, policy gradient, or advantage-based update	Importance-weighted or value-based off-policy update
Example algorithms	SARSA, REINFORCE, A2C, PPO	Q-learning, DQN, DDPG, TD3, SAC

In RL algorithms are often categorized as either on-policy or off-policy according to how they exploit experience. On-policy methods improve the behavior policy using data collected by that same policy, whereas off-policy methods decouple data collection from learning and can profit from experience gathered by the behavior of the policy. This distinction has important implications for sample efficiency, stability,

and exploration/exploitation trade-offs (Table 1). Moreover, the suitability of a particular algorithm depends on whether the environment's action space is discrete or continuous (Table 2).

Table 2. Representative RL algorithms grouped by learning family, policy paradigm, and native action space support.

Algorithm (Year)	Family	Policy paradigm	Native action space
Q-learning (1992) [21]	Value-based	Off-policy	Discrete
SARSA (1994) [22]	Value-based	On-policy	Discrete
Deep Q-Network (DQN, 2015) [23]	Value-based (deep)	Off-policy	Discrete
REINFORCE (1992) [24]	Policy-gradient	On-policy	Discrete / Continuous
Advantage Actor-Critic (A2C, 2017) [25]	Actor-critic	On-policy	Discrete / Continuous
Proximal Policy Optimisation (PPO, 2017) [26]	Actor-critic	On-policy	Discrete / Continuous
Deep Deterministic Policy Gradient (DDPG, 2016) [27]	Actor-critic	Off-policy	Continuous
Twin Delayed DDPG (TD3, 2018) [28]	Actor-critic	Off-policy	Continuous
Soft Actor-Critic (SAC, 2018) [29]	Actor-critic	Off-policy	Continuous
Trust Region Policy Optimization (TRPO) (SAC, 2018) [30]	Actor-critic	On-policy	Continuous

RL algorithms are typically categorized into several core methodologies, enabling the agent to learn optimal strategies by interacting with the environment [31]. These approaches are commonly classified into value-based, policy-based, and model-based methods. Value-based methods, such as Q-learning and deep Q-learning (DQN), are based on estimating the value function to predict the reward, and the Q-value of each state is learned by a neural network [32,33]. In contrast, policy-based methods directly optimize the policy function that maps states to actions, bypassing the need to estimate value functions [34].

The popular policy-based RL algorithms are proximal policy optimization (PPO) [27,35], trust region policy optimization (TRPO) [36], deep deterministic policy gradients (DDPG) [26] and soft-actor-critic (SAC) [37]. These methods typically use a neural network to determine the policy, which is updated with gradients calculated using a variety of methods. The combination of both value-based and policy-based

methods, such as the actor-critic method, is being used to leverage their respective strengths [25].

Model-based reinforcement learning is an approach that uses a model of the environment to make decisions. This is in contrast to model-free methods, in which policies or value functions are learned by directly interacting with the environment without explicitly modeling the dynamics of the environment. It typically involves learning a dynamic model from data obtained through interactions, and then using it for planning or policy optimization [38]. This approach is especially useful when interactions in the real world are expensive or restricted, since learning is possibly effective with simulated experience. Key algorithms in model-based reinforcement learning (MBRL) include Dyna, model-based policy optimization (MBPO), model-based offline policy optimization (MOPO), and probabilistic inference for learning control (PILCO), each employing different modeling techniques to improve policy optimization in various fields [39,40].

A deep reinforcement learning (DRL) model architecture combines RL with deep neural networks (DNN) to deal with complex environments (Figure 3) [41–43]. Based on the algorithms used, such as DQN, proximal policy optimization (PPO), or actor-critic algorithm, the architecture varies significantly. These frameworks consist of three layers: an input layer, hidden layers, and an output layer, as displayed in Figure 4. The input layer processes the environment's state. The hidden layers are responsible for extracting features from the input data, and the output layer differs based on the type of DRL algorithm. For example, the DQN output layer approximates the Q-value of all possible actions and regulates the agent to perform in a way such that the expected reward is maximized [44]. PPO, however, provides output of action probability and utilizes the clipping method for stable policy updates, as well as preventing abrupt policy shifting [45].

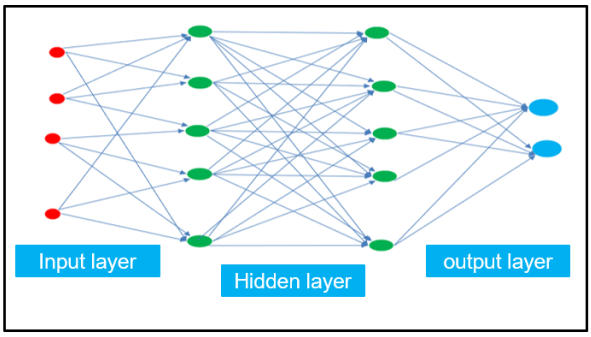

Figure 3. Deep Neural Network

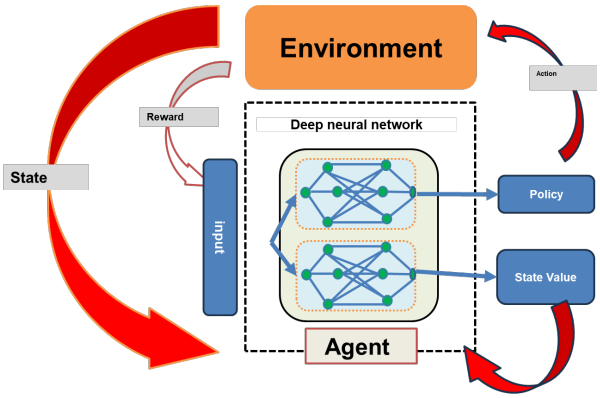

Figure 4. Deep Reinforcement Learning Framework

Training of the DRL agent is an iterative process in which the agent interacts with an environment to gain the maximum cumulative reward. The algorithm begins with a randomly initialized network, and an exploration policy, in which the action is chosen randomly with probability based on learned values [46]. The policy balances between exploring new actions and exploiting learned actions. As the agent moves around in the environment, it gains experiences, which are the current state, the action taken, the reward received, and the next state. These experiences are normally buffered in a replay buffer such that the agent learns from a variety of experiences. The agent updates its model by minimizing a loss function, which improves the model's predictions such that they better align with target values. The reward function plays the central role of guiding the agent's learning in DRL. It should be specified in relation to the

task objectives, providing information that encourages appropriate behavior. Rewards could be sparse or dense: sparse rewards give feedback at task completion, and learning is more difficult, while dense rewards provide feedback at every step, and learning is faster. Reward shaping can also nudge the agent towards the objective by providing the intermediate rewards, while penalty terms discourage undesirable actions [47].

3. RL in material discovery and property prediction

The use of RL for materials discovery is a paradigm shift in computational materials science that offers a very effective framework for automating the search of vast and complex chemical and structural spaces. A unique DRL framework as shown in the Figure 5, consiting of states, reward, and actor-critic network, is specifically designed to explore the PES of nanoclusters to find the GM configurations along with other low-energy states [17]. This study demonstrates the effectiveness of the DRL framework in managing various types of nanoclusters, including mono- and multimetallic compositions, and its proficiency in navigating intricate energy landscapes.

An offline RL method was used to enhance the synthesis of 2D quantum materials like MoS_2 by chemical vapor deposition. By utilizing available molecular dynamics simulation data, a generative model was employed to predict and adjust crucial synthesis parameters, e.g., temperature and gas concentrations [48]. This approach optimizes the material properties and results in improved knowledge of the synthesis process, producing a more effective and scalable solution than traditional trial-and-error material design techniques. Offline RL has also been applied to the design of crystalline materials with target properties, e.g., specific band gaps. Both stability and electronic properties are optimized by this approach at reduced computational cost. It shows the ability of offline RL to generate valid crystals with target properties, offering a promising pathway for large-scale and efficient materials discovery [18].

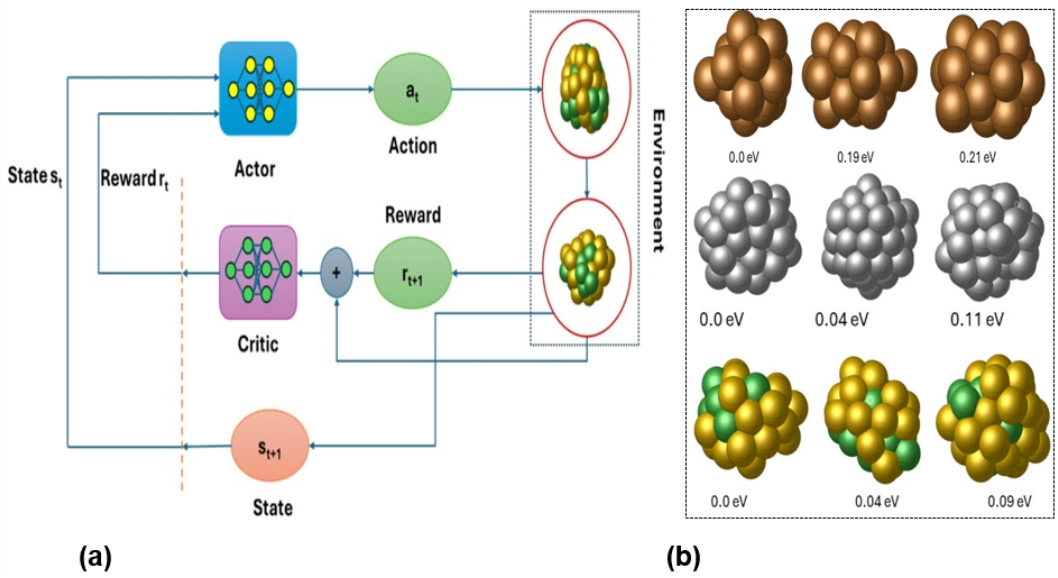

Figure 5. (a) Schematic representation of the DRL framework proposed by Raju *et al.* (b) GM and low-energy configurations generated by using the deep reinforcement learning technique. [17]

Building on the need for more scalable and adaptable RL models, Banik *et al.* [49] presented an RL framework as illustrated in Figure 6. The CASTING (Continuous Action Space Tree Search for

Inverse Design) framework is used to avoid the limitations of the conventional RL methods for continuous materials search spaces. Through the combination of a Monte Carlo Tree Search (MCTS) algorithm and modified policies and sampling methods, CASTING enables effective exploration of complex potential energy landscapes. Extended to systems from elemental metals like silver (Ag) to covalent materials like carbon (C) and multicomponent compounds like graphene, boron nitride, and correlated oxide. The scheme showed high accuracy, rapid convergence, and excellent scalability, representing a great leap in data-driven inverse material design. The convergence of the MCTS optimizer for the sampling of gold nanoclusters of different sizes and their global minima obtained by MCTS for each case is illustrated in Figure 7.

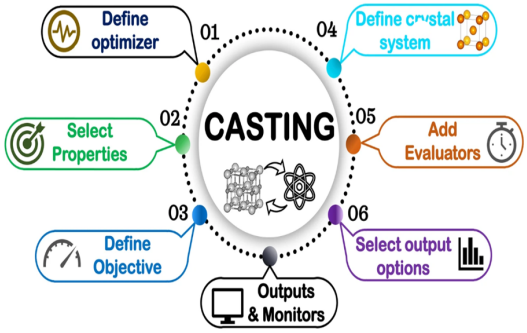

Figure 6. Schematic depiction of the workflow of the CASTING framework for performing inverse design [49].

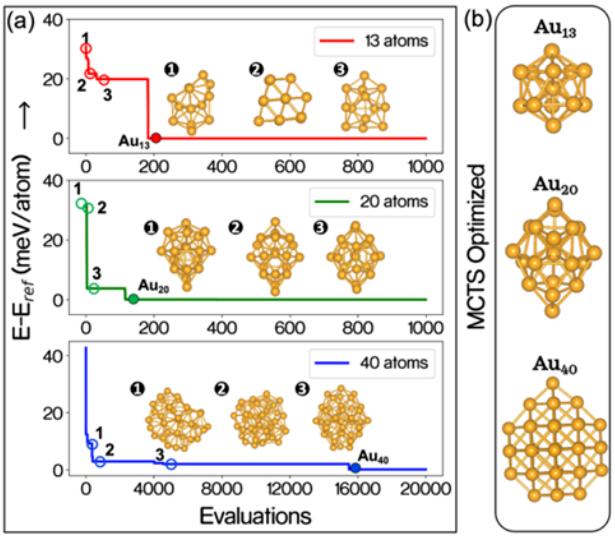

Figure 7. (a) The convergence of MCTS optimizer for the sampling of gold nanoclusters of different sizes, and (b) shows the global minima obtained by MCTS for Au_{13}, Au_{20} , Au_{40} nanoclusters [49].

Complementing these developments, Modee *et al.* [50] proposed a novel actor-critic architecture, that generates low-lying isomers of gallium metal clusters at a fraction of the computational cost in comparison to conventional methods. Their approach emphasizes the generation of low-energy 3D metal cluster structures depends on the efficiency of the search algorithm and the accuracy of the description of the interatomic interaction. Their RL-based search algorithm uses a previously developed deep learning-enabled topological interaction (DART) model [51], as a reward function to describe interatomic interactions to validate predicted structures. Using the DART model as a reward function incentivizes the

RL model to generate low-energy structures, and helps generate valid structures. They demonstrate the advantages of their approach over conventional methods for scanning local minima on the PES (Figure 8). Furthermore, Pan *et al.* [52] explore the use of RL for the inverse design of inorganic materials, focusing on discovering materials with specific properties. Simm *et al.* [16] developed an RL framework guided by quantum mechanics, enabling the design of molecules with optimized properties. Their approach uses quantum-chemical methods, allowing for efficient exploration of molecular spaces to identify novel catalyst candidates with enhanced properties.

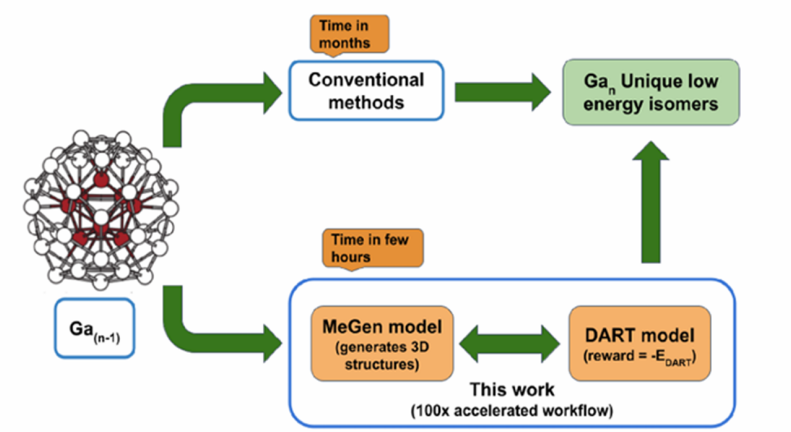

Figure 8. Workflow to generate GS/low-energy gallium clusters using reinforcement learning. [50].

4. RL in property prediction and optimization

In a broader context, RL advances material discovery by efficiently identifying materials with extreme properties. When combined with big data techniques, it enables precise control over nanomaterial structures, properties, and synthesis methods [53]. RL not only guides the selection of molecular fragments, but also aids in the design of new molecules with optimized properties [54]. The workflow of material prediction and optimization through RL is demonstrated in Figure 9.

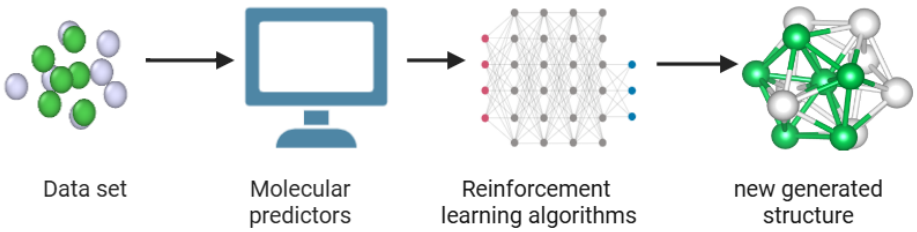

Figure 9. Workflow of material prediction and optimization through artificial intelligence.

This approach enables search in vast chemical spaces and rapid discovery of high-performance materials. Zhang *et al.* [55] recently created a DRL algorithm to forecast molecular properties in drug design and material discovery. The strongest advantage of the approach is that it predicts the structure of a molecule from historical data, enabling faster and more efficient discovery without the requirement of new experimental samples. Moreover, a multi-objective RL approach is used in catalyst design to optimize the balance of several performance metrics [56].

The geometry optimization of nanoclusters via DRL is proposed by Mubeen *et al.* [57], uses an actor-critic architecture to navigate efficiently PES, and identify the most stable geometries of Ag_{15} nanocluster. By bypassing traditional optimization methods, such as GA, the DRL system achieves significant improvements in computational efficiency and precision. Anderson *et al.*'s [58], study of gallium-metal clusters shows the power of model-based RL for streamlining material synthesis. Their approach used a transition model that forecasts the effect of different synthesis conditions on the formation of stable metal clusters, which offers important insights into the optimization of nanoalloys. Meldgaard *et al.* [59], present the Atomistic Structure Learning Algorithm (ASLA), an RL-driven method for predicting reconstructed crystalline surface structures. This method effectively reconstructs complex structures using transfer learning, significantly cutting down computational costs compared to traditional approaches, demonstrating the scalability of RL in structure prediction. Additionally, the autonomous optimization of molecular geometry has been markedly improved in both efficiency and accuracy through the application of multi-agent RL [60]. Elsborg *et al.* [61], developed an actor-critic RL method utilizing equivariant graph representations to identify low-energy nanoparticle structures. Their approach effectively discovers stable configurations for mono- and bimetallic clusters. While successful in finding known stable configurations, the study also highlights challenges such as the agent's limited generalization ability, suggesting the need for further improvements to achieve broader applicability in nanoparticle design.

The catalytic reaction mechanisms over various processes are studied via a DRL framework [62] coupled with first-principles calculations. In this approach, the optimal reaction channels are identified in a self-controlled manner and uncover reduced energy barriers, showcasing the potential of AI-assisted methods to accelerate the identification of catalytic reactions, and improve catalyst efficiency [62]. Tian *et al.* [63], explore the catalytic reaction mechanism of ammonia synthesis on the Fe surface via the high throughput deep reinforcement learning (HDRL) framework, which integrates DRL and first-principles calculations for the autonomous exploration of reaction pathways. Both Langmuir-Hinshelwood (LH) and Eley-Rideal (ER) mechanisms for hydrogen migration have been successfully discovered in their research, and a pathway with a lower energy barrier compared to the nudged elastic band (NEB) method. Figure 10 displays the process of the DRL approach applied to catalyst optimization, demonstrating the integration of AI techniques in accelerating catalytic design, and Figure 11 shows the inference process for molecular generation, highlighting both the generation pathway and the corresponding evolution of molecular properties.

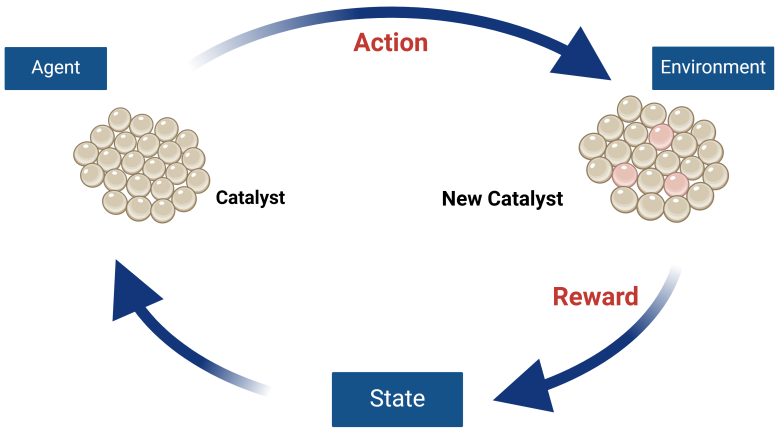

Figure 10. Deep Reinforcement Learning for Catalyst Optimization.

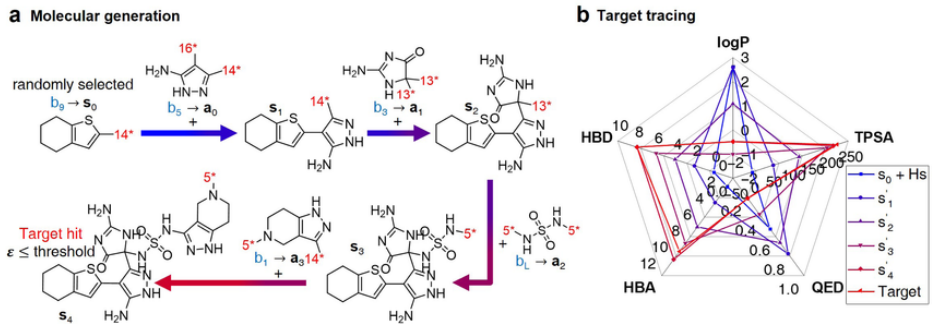

Figure 11. Inference process for molecular generation. (a) An example of a molecular generation process, (b) shows the property changes for generated molecules [54].

The AI-driven approach by Yoon *et al.* [64], improves the design process by identifying metastable surface phases and accelerating the discovery of optimal catalytic systems. On a ternary $Ni_3Pd_3Au_2$ alloy, their CatGym DRL framework outperforms traditional methods like minima hopping to efficiently explore disparate surface compositions and predict reconstruction pathways with greater accuracy. Chang *et al.* [65], demonstrated that integrating chemical information into RL enhances catalyst design by predicting optimal atomic arrangements, and identifying metastable states that improve catalytic activity. Their approach incorporates a graph-based topological method, accelerates the discovery of novel catalysts by improving reaction efficiency and selectivity in catalytic processes. The study by Mills *et al.* [66] explores the use of RL to effectively navigate PES in molecular systems. Through the application of RL to complex PES landscapes, it demonstrated the capability of RL to optimize molecular geometries and computational efficiency in molecular simulations, opening up a new avenue for the investigation and design of catalytic systems. The MolOpt framework developed by Modee *et al.* [67] shows promise for advancing catalyst design by enhancing the optimization of molecular structures for catalytic applications. It efficiently optimized molecules like propane and octane, outperforming traditional methods such as MDMin and FIRE. The DRL method is proposed [68] to automate the diagnostic process of nanocatalysts by analyzing data from techniques like X-ray absorption spectroscopy (XAS). This approach enhances the understanding of nanocatalyst behavior, leading to improved design and optimization for catalytic applications.

5. Comparative study: DRL vs. conventional optimization approaches

A comparative study of DRL and traditional optimization techniques shows significant differences in their performance metrics, including computational cost, convergence rate, and success rate in identifying global minima. DRL has shown significant capability in dynamically optimizing intricate systems in real time; for instance, a study proved its application in industrial catalytic processes, where it achieved results comparable to mathematical optimization standards at faster inference times [69]. Regarding computational costs, the DRL models prove cost-efficient due to converging faster than standard methods [70]. RL frameworks have been used effectively to optimize the microstructural material properties, like silica aerogels with outstanding fast convergence in low-complexity and high-complexity cases [71]. Traditional optimization algorithms like GA and BH are typically in the form of pre-defined fitness functions, and become trapped in local minima, leading to worse success rates of finding global optima compared to iterative learning by RL [72]. While traditional methods may make it easier to interpret and require less initial computation, they lack the flexibility that DRL allows in dynamically changing environments [73]. Overall, while RL provides flexibility and efficiency advantages in solving hard optimization problems, traditional methods are also

usable on specific applications where established protocols can be employed optimally. The choice between these methods ultimately hinges on the specific requirements of the material design problem at hand.

Experiments on the performance of RL, GA, and BH on optimization problems provide clear findings on the differences between RL and traditional search-based algorithms. It is the extreme ruggedness, fast convergence, and nature of a global minima-based approach attained with utmost flexibility that differentiates RL [74]. Such properties render RL particularly suitable for complex and dynamic environments, where fast feedback and being adaptable are critical [75]. For example, GA has high invariance, which makes it robust and enables it to be widely used for optimization problems, although GA also comparatively has slow convergence, especially in high-dimensional solutions. Moreover, GA can struggle with reliably finding global minima, particularly on more complex or non-convex optimization surfaces. In contrast, BH sometimes has speed advantages over GA, but not always, and offers poorer robustness than GA, along with a poorer chance of finding global minima in tougher optimization cases. Furthermore, BH is limited in its adaptability [76], which may hinder its performance in dynamic or evolving problem spaces. These synthesized findings suggest that while RL represents a powerful and versatile optimization tool, particularly for dynamic and complex problems, GA and BH may remain suitable for simpler tasks or when computational resources are constrained. Despite these generalizations, a direct comparison of these algorithms on the same dataset, along with quantitative metrics, is a significant challenge. Such comparative studies are necessary to rigorously benchmark these optimization methods and contrast their relative strengths in diverse real-world applications. A summary of these findings can be found in Table 3.

Table 3. Comparison of different optimization methods based on various metrics.

Metric	RL	GA	BH
Robustness	High	High	Moderate
Speed/Convergence	High	Moderate	Moderate
Global Minimum Identification	High	Moderate	Moderate
Adaptability	High	Low	Low

6. Current challenges and future perspectives

The RL has surfaced as a viable approach in material discovery that could automate and accelerate the design of materials with desired properties. Various issues that have to be tackled to gain full advantage from its potential, including data scarcity and quality issues, reward function design complexity, and generalization and validation issues [52]. RL holds a lot of promise for future breakthroughs. It can be further developed with hybrid model-based approaches, physics-informed rewards, and meta learning to improve sample efficiency and generalization. For instance, RL-based combinatorial chemistry has been shown to be successful for the discovery of molecules with extreme properties, outperforming probability distribution-learning models in generating chemically valid molecules.

Future directions include integrating RL with foundation multi-modal models that can process different data modalities, such as spectroscopic and crystallographic data, understanding material behavior

in an integrated manner [77]. This could lead to novel domains such as catalysis, drug discovery, and energy storage because it would enable the discovery of materials with unprecedented properties. Furthermore, modular RL structures like MANDREL facilitate flexible experimentation and design via mixing different chemical representations and RL strategies to accelerate the discovery of new molecular species [78]. Overall, resolving such challenges as well as the embracing of such future trends could make RL a groundbreaking tool in materials science to propel advancements in sustainability, medicine, and innovative manufacturing

7. Conclusion

The integration of reinforcement learning (RL) in materials discovery is a significant advance in the quest for efficient, sustainable technologies. By leveraging the capabilities of RL, researchers can tune complex systems, enhance the performance of catalytic materials, and accelerate the identification of new compounds with desirable properties. The ability of RL to handle big data and anticipate optimal configurations allows for more efficient probing of chemical spaces, reducing the time and cost of traditional experimental methods. Application of RL systems, as described in many studies demonstrates efficiency in improving the activity of catalysts through self-guided search and optimization of reaction paths. Specifically, the progress of RL has enabled the application of more sophisticated strategies in simulating complex environments, thereby enabling a more comprehensive understanding and control over material properties.

Besides, the continuous development of RL approaches, such as model-based methods, and multi-objective optimization, provides great opportunities for solving intricate problems in material science. These RL methods have the potential to generate new avenues for innovation in materials and catalyst design, with enhanced energy sustainability and environmental protection. The intersection of RL with materials discovery not only accelerates the research, but also holds the key to the development of next-generation materials that can keep up with the demands of a changing world in a high-speed manner. The ongoing pursuit and evolution of these AI-based methods are important in unshackling their full capabilities in both scientific inquiry and industrial applications.

Author's contribution

Conceptualization, A.N., F.U.M. and C.F.; formal analysis, A.N., C.F. and F.U.M.; data curation, A.N.and F.U.M.; Methodology, A.N.and F.U.M.; visualization, A.N.and F.U.M.; validation, A.N.and F.U.M.; writing—original draft, A.N.and F.U.M.; resources, C.F.; supervision, C.F.; writing—review and editing, C.F. All authors have read and agreed to the published version of the manuscript.

Conflicts of interests

There is no conflict of interest.

References

[1] Ramalingam G, Kathirgamanathan P, Ravi G, Elangovan T, kumar BA, *et al.* Quantum confinement effect of 2D nanomaterials. In *Quantum Dots*, Rijeka: IntechOpen, 2020,.

[2] Park H, Shin DJ, Yu J. Categorization of quantum dots, clusters, nanoclusters, and nanodots. *J. Chem. Educ.* 2021, 98(3):703–709.

[3] Chen F, Johnston RL. Charge transfer driven surface segregation of gold atoms in 13-atom Au–Ag nanoalloys and its relevance to their structural, optical and electronic properties. *Acta Mater.* 2008, 56(10):2374–2380.

[4] Chen F, Johnston RL. Energetic, Electronic, and Thermal Effects on Structural Properties of AgAu Nanoalloys. *ACS Nano* 2008, 2(1):165–175.

[5] Zhang J, Dolg M. ABCluster: the artificial bee colony algorithm for cluster global optimization. *Phys. Chem. Chem. Phys.* 2015, 17:24173–24181.

[6] Gehrke R, Reuter K. Assessing the efficiency of first-principles basin-hopping sampling. *Phys. Rev. B* 2009, 79:085412.

[7] Banerjee A, Jasrasaria D, Niblett SP, Wales DJ. Crystal Structure Prediction for Benzene Using Basin-Hopping Global Optimization. *J. Phys. Chem. A* 2021, 125(17):3776–3784. PMID: 33881850.

[8] Wales DJ, Doye JPK. Global optimization by basin-hopping and the lowest energy structures of lennard-jones clusters containing up to 110 atoms. *J. Phys. Chem. A* 1997, 101(28):5111–5116.

[9] Johnston RL. Evolving better nanoparticles: genetic algorithms for optimising cluster geometries. *Dalton Trans.* 2003, pp. 4193–4207.

[10] Nørskov JK, Bligaard T, Rossmeisl J, Christensen CH. Towards the computational design of solid catalysts. *Nat. Chem.* 2009, 1(1):37–46.

[11] Usman M, Chen F. Generation and optimization of gold nanoclusters via reinforcement learning. *The European Physical Journal D* 2025 79(5):58. 10.1140/epjd/s10053-025-01006-w.

[12] Zhou Z, Li X, Zare RN. Optimizing chemical reactions with deep reinforcement learning. *ACS Cent. Sci.* 2017 3(12):1337–1344.

[13] Cobbe KW, Hilton J, Klimov O, Schulman J. Phasic policy gradient. In *International Conference on Machine Learning*, PMLR, 2021 pp. 2020–2027.

[14] Wang X, Yang Z, Chen G, Liu Y. A Reinforcement Learning Method of Solving Markov Decision Processes: An Adaptive Exploration Model Based on Temporal Difference Error. *Electronics* 2023 12(19):4176.

[15] Khan QW. Exploring Markov Decision Processes: A Comprehensive Survey of Optimization Applications and Techniques. *Igmin. Res.* 2024, 2(7):508–517.

[16] Simm G, Pinsler R, Hernández-Lobato JM. Reinforcement learning for molecular design guided by quantum mechanics. In *International Conference on Machine Learning*, PMLR, 2020 pp. 8959–8969.

[17] Raju RK. Exploring nanocluster rotential energy surfaces via deep reinforcement learning: strategies for global minimum search. *J. Phys. Chem. A* 2024, 128(42):9122–9134.

[18] Rajak P, Krishnamoorthy A, Mishra A, Kalia R, Nakano A, *et al.* Autonomous reinforcement learning agent for chemical vapor deposition synthesis of quantum materials. *npj Comput. Mater.*

2021, 7(1):108.

[19] Matignon L, Laurent GJ, Le Fort-Piat N. Reward Function and Initial Values: Better Choices for Accelerated Goal-Directed Reinforcement Learning. In *Artificial Neural Networks – ICANN 2006*, Kollias SD, Stafylopatis A, Duch W, Oja E, eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 840–849.

[20] Spielberg P S, Gopaluni B, Loewen P. Deep reinforcement learning approaches for process control. 2017, pp. 201–206.

[21] Watkins CJCH, Dayan P. Q-learning. *Mach. Learn.* 1992, 8(3):279–292.

[22] Rummery GA, Niranjan M. On Line Q learning using connectionist systems. 1994, Available: https://www.repository.cam.ac.uk/handle/1810/271475(accessed on 13 April 2025).

[23] Mnih V, Kavukcuoglu K, Silver D, Rusu AA, Veness J, *et al.* Human-level control through deep reinforcement learning. *Nature* 2015, 518(7540):529–533.

[24] Williams RJ. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Mach. Learn.* 1992, 8(3–4):229–256.

[25] Mnih V, Badia AP, Mirza M, Graves A, Lillicrap T, *et al.* Asynchronous methods for deep reinforcement learning. In *International conference on machine learning*, PmLR, 2016, pp. 1928–1937.

[26] Lillicrap TP, Hunt JJ, Pritzel A, Heess N, Erez T, *et al.* Continuous control with deep reinforcement learning. *arXiv* 2015, arXiv:1509.02971.

[27] Akkaya I, Andrychowicz M, Chociej M, Litwin M, McGrew B, *et al.* Solving rubik's cube with a robot hand. *arXiv* 2019, arXiv:1910.07113.

[28] Fujimoto S, van Hoof H, Meger D. Addressing function approximation error in actor-critic methods. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, PMLR, 2018. pp. 1582–1591.

[29] Haarnoja T, Zhou A, Abbeel P, Levine S. Soft Actor-Critic: Off-Policy maximum entropy deep reinforcement learning with a stochastic actor. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, PMLR, 2018. pp. 1861–1870.

[30] Schulman J, Levine S, Moritz P, Jordan MI, Abbeel P. Trust Region Policy Optimization. *arXiv* 2015, arXiv:1502.05477.

[31] Sutton RS, Barto AG, *et al. Reinforcement learning: An introduction*. 2nd ed. London:The MIT Press, 1998.

[32] Mnih V, Kavukcuoglu K, Silver D, Graves A, Antonoglou I, *et al.* Playing atari with deep reinforcement learning. *arXiv* 2013, arXiv:1312.5602.

[33] Mnih V, Kavukcuoglu K, Silver D, Rusu AA, Veness J, *et al.* Human-level control through deep reinforcement learning. *nature* 2015, 518(7540):529–533.

[34] Engstrom L, Ilyas A, Santurkar S, Tsipras D, Janoos F, *et al.* Implementation matters in deep policy gradients: A case study on ppo and trpo. *arXiv* 2020, arXiv:2005.12729.

[35] Schulman J, Wolski F, Dhariwal P, Radford A, Klimov O. Proximal policy optimization algorithms. *arXiv* 2017, arXiv:1707.06347.

[36] Engstrom L, Ilyas A, Santurkar S, Tsipras D, Janoos F, *et al.* Implementation matters in deep policy gradients: A case study on ppo and trpo. *arXiv* 2020, arXiv:2005.12729.

[37] Haarnoja T, Zhou A, Abbeel P, Levine S. Soft actor-critic: Off-policy maximum entropy deep

reinforcement learning with a stochastic actor. In *International conference on machine learning*, Pmlr, 2018 pp. 1861–1870.

[38] Werbos PJ. Reinforcement learning and approximate dynamic programming (RLADP)—foundations, common misconceptions, and the challenges ahead. In *Reinforcement learning and approximate dynamic programming for feedback control*, Wiley Online Library, 2012, pp. 1–30.

[39] Luo F, Xu T, Lai H, Chen Xh, Zhang W, *et al.* A survey on model-based reinforcement learning. *Sci. China Inf. Sci.* 2024, 67(2):121101.

[40] Yu T, Thomas G, Yu L, Ermon S, Zou JY, *et al.* Mopo: model-based offline policy optimization. *Adv. Neural Inf. Process. Syst.* 2020, 33:14129–14142.

[41] Rao PV, B V, Manjeet M, Kumar A, Mittal M, *et al.* Deep reinforcement learning: bridging the gap with neural networks. *Int. J. Intell. Syst. Appl. Eng.* 2024, 12(15s):576–586.

[42] Van Hasselt H, Guez A, Silver D. Deep reinforcement learning with double q-learning. *Proc. AAAI Conf. Artif. Intell.* 2016, 30(1).

[43] Hafiz AM, Hassaballah M, Alqahtani A, Alsubai S, Hameed MA. Reinforcement learning with an ensemble of binary action deep Q-Networks. *Comput. Syst. Sci. Eng.* 2023, 46(3).

[44] Terven J. Deep reinforcement learning: a chronological overview and methods. *AI* 2025, 6(3).

[45] de la Fuente N, Guerra DAV. A comparative study of deep reinforcement learning models: DQN vs PPO vs A2C. *J. Mach. Learn. Res.* 2024 .

[46] Ivanov S. Reinforcement learning from scratch: deep Q networks 2018 Available: https://towardsdatascience.com/reinforcement-learning-from-scratch-deep-q-networks-0a8d33ce165b (accessed on 26 February 2025).

[47] Li Y, Hu X, Zhuang Y, Gao Z, Zhang P, *et al.* Deep reinforcement learning (DRL): another perspective for unsupervised wireless localization. *IEEE Internet Things J.* 2020, 7(7):6279–6287.

[48] Tarasov D, Mbou Sob UA, Arbesú M, Siboni N, Boyer S, *et al.* Offline RL for generative design of protein binders. *bioRxiv* 2023, pp. 2023–11.

[49] Banik S, Loefller T, Manna S, Chan H, Srinivasan S, *et al.* A continuous action space tree search for INverse desiGn (CASTING) framework for materials discovery. *npj Comput. Mater.* 2023, 9(1):177.

[50] Modee R, Verma A, Joshi K, Priyakumar UD. MeGen-generation of gallium metal clusters using reinforcement learning. *Mach. Learn.: Sci. Technol.* 2023, 4(2):025032.

[51] Modee R, Agarwal S, Verma A, Joshi K, Priyakumar UD. DART: deep learning enabled topological interaction model for energy prediction of metal clusters and its application in identifying unique low energy isomers. *Phys. Chem. Chem. Phys.* 2021, 23(38):21995–22003.

[52] Pan E, Karpovich C, Olivetti E. Deep reinforcement learning for inverse inorganic materials design. *arXiv* 2022, arXiv:2210.11931.

[53] Suvarna M, Pérez-Ramírez J. Embracing data science in catalysis research. *Nat. Catal.* 2024, pp. 1–12.

[54] Kim H, Choi H, Kang D, Lee WB, Na J. Materials discovery with extreme properties via reinforcement learning-guided combinatorial chemistry. *Chem. Sci.* 2024, 15(21):7908–7925.

[55] Yang Rx, McCandler CA, Andriuc O, Siron M, Woods-Robinson R, *et al.* Big data in a nano world: a review on computational, data-driven design of nanomaterials structures, properties, and synthesis. *ACS nano* 2022, 16(12):19873–19891.

[56] Govindarajan P, Miret S, Rector-Brooks J, Phielipp M, Rajendran J, *et al.* Learning conditional

policies for crystal design using offline reinforcement learning. *Digit. Discov.* 2024, 3(4):769–785.

[57] Mubeen M, Chen F, Rehman K. Optimization of silver nanocluster geometries: a ddeep reinforcement learning approach to identifying the most stable configurations in Ag15 cluster. *J. Chem. Environ.* 2025, pp. 1–17.

[58] Lacombe R, Hendren L, El-Awady K. AdsorbRL: deep Multi-Objective reinforcement learning for inverse catalysts design. In *AI for Accelerated Materials Design - NeurIPS 2023 Workshop*. 2023 .

[59] Meldgaard SA, Mortensen HL, Jørgensen MS, Hammer B. Structure prediction of surface reconstructions by deep reinforcement learning. *J. Phys.: Condens. Matter* 2020, 32(40):404005.

[60] Modee R, Mehta S, Laghuvarapu S, Priyakumar UD. MolOpt: autonomous molecular geometry optimization using multiagent reinforcement learning. *J. Phys. Chem. B* 2023, 127(48):10295–10303.

[61] Elsborg J, Bhowmik A. Equivariant graph-representation-based actor–critic reinforcement learning for nanoparticle design. *J. Chem. Inf. Model.* 2023, 63(12):3731–3741.

[62] Lan T, An Q. Discovering catalytic reaction networks using deep reinforcement learning from first-principles. *J. Am. Chem. Soc.* 2021, 143(40):16804–16812.

[63] Lan T, Wang H, An Q. Enabling high throughput deep reinforcement learning with first principles to investigate catalytic reaction mechanisms. *Nat. Commun.* 2024, 15(1):6281.

[64] Yoon J, Cao Z, Raju RK, Wang Y, Burnley R, *et al.* Deep reinforcement learning for predicting kinetic pathways to surface reconstruction in a ternary alloy. *Mach. Learn.: Sci. Technol.* 2021, 2(4):045018.

[65] Chang Y, Li Y. Integrating chemical information into reinforcement learning for enhanced molecular geometry optimization. *J. Chem. Theory Comput.* 2023, 19(23):8598–8609.

[66] Mills AW, Goings JJ, Beck D, Yang C, Li X. Exploring potential energy surfaces using reinforcement machine learning. *J. Chem. Inf. Model.* 2022, 62(13):3169–3179.

[67] Modee R, Mehta S, Laghuvarapu S, Priyakumar UD. MolOpt: autonomous molecular geometry optimization using multiagent reinforcement learning. *J. Phys. Chem. B* 2023, 127(48):10295–10303.

[68] Kartashov OO, Polyanichenko DS, Savvas IK, Beliavsky GI, Butakova MA. Artificial intelligence approach to palladium nanocatalysts diagnostics automation. In *International Conference on Intelligent Information Technologies for Industry*, St. Petersburg, Russia, 25–30 September 2023, pp. 45–54.

[69] Singh N, Stolte J, Li B, Jaso S, Michler C. Real-time optimization of industrial processes using deep reinforcement learning. *BNAIC/BeNeLearn* 2022 .

[70] Arce D, Solano J, Beltrán C. A comparison study between traditional and deep-reinforcement-learning-based algorithms for indoor autonomous navigation in dynamic scenarios. *Sensors* 2023, 23(24).

[71] Pandit P, Abdusalamov R, Itskov M, *et al.* Deep reinforcement learning for microstructural optimisation of silica aerogels. *Sci. Rep.* 2024, 14(1):1511.

[72] Zhou Z, Li X, Zare RN. Optimizing Chemical Reactions with Deep Reinforcement Learning. *ACS Cent. Sci.* 2017, 3(12):1337–1344.

[73] Chen Y. Active learning and reinforcement learning for autonomous catalyst design in CO2 hydrogenation. *Int. J. Mater. Sci. Technol. Stud.* 2024, 2(2):65–75.

[74] Raju RK. Exploring nanocluster potential energy surfaces via deep reinforcement learning: strategies for global minimum search. *J. Phys. Chem. A* 2024, 128(42):9122–9134.

[75] Cao Y, Lien S, Liang Y, Niyato D. Multi-Tier deep reinforcement learning for Non-Terrestrial

networks. *IEEE Wirel. Commun.* 2024, pp:1–8.

[76] Bauer MN, Probert MIJ, Panosetti C. Systematic comparison of genetic algorithm and basin hopping approaches to the global optimization of Si(111) surface reconstructions. *J. Phys. Chem. A* 2022, 126(19):3043–3056.

[77] Pyzer-Knapp EO, Manica M, Staar P, Morin L, Ruch P, *et al.* Foundation models for materials discovery–current state and future directions. *npj Comput. Mater.* 2025, 11(1):61.

[78] Fare C, Holt GK, Chiazor L, Smyrnakis M, Tracey R, *et al.* MANDREL: modular reinforcement learning pipelines for material discovery. In *Proceedings of the AAAI Conference on Artificial Intelligence*. Vancouver, Canada, 26-27 February 2024, pp. 23787–23789.