PNAS RESEARCH ARTICLE | CHEMISTRY

The quantum ensemble variational optimization algorithm: Applications to molecular inverse design

Francesco Calcagno[a,b,1], Delmar G. A. Cabral[c], Ivan Rivalta[a,b,1], and Victor S. Batista[c,d,1]

Affiliations are included on p. 8.

Edited by Laura Gagliardi, The University of Chicago, Chicago, IL; received October 31, 2025; accepted June 18, 2026

Designing molecules with optimized properties remains a fundamental challenge due to the intricate relationship between molecular structure and properties. Traditional computational approaches that address the combinatorial number of possible molecular designs become unfeasible as the molecular size increases, suffering from the so-called "curse of dimensionality" problem. Recent advances in quantum computing hardware present new opportunities to address this problem. Here, we introduce the quantum ensemble variational optimization (QEVO) method for near-term and early fault-tolerant quantum computing platforms. QEVO efficiently maps molecular structures onto an orthonormal basis of binary strings and samples from a superposition state generated by a variational ansatz. The ansatz is iteratively optimized to identify molecular candidates with the desired property. Our numerical simulations demonstrate the potential of QEVO to design drug-like molecules with anticancer properties, operating in combinatorial spaces composed of up to 2^{160} solutions, while employing a shallow quantum circuit that requires only a modest number of qubits. We envision that QEVO could be applied to a wide range of complex problems, offering practical solutions to problems with combinatorial complexity.

molecular design | quantum computing | variational quantum algorithm | drug development

Significance

Molecular inverse design is a classically exponential problem that could greatly benefit from quantum computing. Here, we introduce the quantum ensemble variational optimization (QEVO), a quantum algorithm that encodes molecules into a parameterized quantum state. This state is variationally evolved to minimize the average property of molecules sampled from it. By leveraging quantum superposition, QEVO rapidly identifies molecules with the desired properties. We demonstrate its effectiveness on various design tasks, including anticancer drug design, operating in combinatorial spaces of up to 2^{160} possible solutions. QEVO offers a practical approach to applying quantum algorithms for molecular inverse design.

The computational design of molecules with tailored properties remains a central and long-standing challenge in chemistry (1–8). Unlike traditional approaches that search molecular databases for candidates with desired properties, molecular inverse design aims to directly generate molecular structures that meet suitable properties for specific application (1). This paradigm underlies a broad spectrum of research efforts, spanning from drug discovery to advanced materials design.

The difficulty of inverse design lies in the inherently complex and often nonintuitive relationships between molecular structure and properties. As a result, many current methods rely on "brute-force" strategies, including exhaustive searches and high-throughput screening (9). However, the combinatorial explosion of possible molecular frameworks—the entire chemical space (10)—makes such approaches ineffective in most practical settings.

To date, a unique approach for molecular design remains elusive. As a result, numerous methodologies have been proposed, including physics-based methods (8, 11–13), and computational high-throughput screening strategies (14–16). Recently, machine learning (ML) methods have emerged as valuable tools to accelerate the generation of molecules with predefined properties (17–23). However, ML methods typically require large training datasets, extensive hyperparameters tuning, and do not guarantee a thorough exploration of the chemical space (23).

Quantum computers (QCs) present new opportunities for developing computational methods that surpass the capabilities of classical computational methods (24). The computational utility of QCs extends beyond potential computational speedups, including the development of fundamentally new approaches as demonstrated by prominent quantum algorithms for search (25), factorization (26), optimization (27–29), and quantum chemistry (30–32). Variational Quantum Algorithms (VQAs) employ a hybrid quantum-classical framework to approximate solutions for various computational problems via cost function minimization and have attracted considerable attention due to their practical implementation on noisy intermediate-scale quantum (NISQ) devices (33). In this approach, a parameterized quantum state

$$|\psi(\vec{\theta})\rangle = U(\vec{\theta})|0\rangle \qquad [1]$$

Author contributions: F.C. designed research; F.C. performed research; F.C., D.G.A.C., I.R., and V.S.B. analyzed data; F.C. developed the code; D.G.A.C. contributed to code development; I.R. and V.S.B. supervised the work and provided financial support; and F.C., D.G.A.C., I.R., and V.S.B. wrote the paper.

Competing interest statement: F.C., D.G.A.C., I.R., and V.S.B. are listed as inventors on a patent application ("Articles and methods for quantum ensemble variational optimization").

This article is a PNAS Direct Submission.

[1]To whom correspondence may be addressed. Email: francesco.calcagno@unibo.it, i.rivalta@unibo.it, or victor.batista@yale.edu.

This article contains supporting information online at https://www.pnas.org/lookup/suppl/doi:10.1073/pnas.2531186123/-/DCSupplemental.

Published July 23, 2026.

is prepared by applying a sequence of unitary operations $U(\vec{\theta})$, governed by a set of variational parameters $\vec{\theta}$, to the initial vacuum state $|0\rangle$. Measurements on the resulting state are processed by a classical optimizer, which iteratively updates $\vec{\theta}$ to minimize an objective function, typically the expectation value of a Hamiltonian. This iterative procedure continues until convergence to an optimal solution is achieved (33). The method can also be extended to quantum search tasks, as in the Variational Quantum Search (VQS) approach (34), by encoding the target solution as the ground state of an effective Hamiltonian.

However, many relevant optimization problems—such as predicting molecular solubility, toxicity, or synthetic accessibility—do not naturally lend themselves to expectation value formulations over quantum states. To address these cases, it is necessary to generalize the VQA framework to accommodate broader classes of objective functions beyond expectation values.

Here, we introduce the so-called quantum ensemble variational optimization (QEVO) algorithm, which extends VQAs to cases where the property of interest cannot be computed as an expectation value of a superposition state, thus, explicitly mapped to a cost Hamiltonian or quadratic unconstrained binary optimization (QUBO) formulation (35). QEVO optimizes the variational parameters of a quantum circuit to optimize the ensemble average of the property of interest, after sampling molecules from a distribution defined by the generated coherent superposition state. Contrary to classical methods, QEVO does not require training datasets, broad hyperparameters fine-tuning, and it is highly customizable for solving different combinatorial tasks. We demonstrate the capabilities of QEVO by applying it to the generation of molecules optimized for specific properties, including molecular solubility, drug-likeness, and binding affinity to a protein target relevant to cancer treatment.

The QEVO Algorithm. QEVO maps molecular structures onto an orthonormal basis, $\mathcal{M}$, of binary strings $|m_i\rangle$ (see *Materials and Methods* for details). Each bitstring represents a specific molecule with well-defined property p_i. The bitstrings are sampled by measuring the coherent variational superposition state $|M(\vec{\theta})\rangle = U(\vec{\theta})|0\rangle$ in the computational basis (Fig. 1*A*).

We use a real amplitude ansatz (Fig. 1 *D* and *F*), so the resulting variational state $|M(\vec{\theta})\rangle$ is an alchemical superposition of possible molecules $|m_i\rangle$:

$$|M(\vec{\theta})\rangle = \sum_i c_i(\vec{\theta})|m_i\rangle, \quad \text{where} \quad |m_i\rangle \in \mathcal{M}, \qquad [2]$$

where $|c_i(\vec{\theta})|^2$ determines the probability of sampling molecule i encoded by $|m_i\rangle$. Therefore, QEVO does not optimize the variational parameters with respect to an expectation value as in conventional VQAs (33, 34). Instead, QEVO optimizes $\vec{\theta}$ with respect to the loss function,

$$\mathcal{L}(\vec{\theta}) = |p_M(\vec{\theta}) - p_0| + \epsilon(\vec{\theta}), \qquad [3]$$

defined by the ensemble average,

$$p_M(\vec{\theta}) = \sum_i |c_i(\vec{\theta})|^2 p_i. \qquad [4]$$

Here, p_i is the value of the property of interest for molecule $|m_i\rangle$, while p_0 is the target value. In this context, the term 'ensemble average' is used operationally to refer to the set of properties of molecules sampled from the quantum pure state, not to a quantum mixed state in the density matrix formulation. The ensemble average p_M is thus computed at an arbitrary level of theory after decoding the sampled strings $|m_i\rangle$ into a molecular structure. The regularization term $\epsilon(\vec{\theta}) = \lambda|c_i(\vec{\theta})|^4$ enhances the convergence, as weighted by the hyperparameter λ. According to this approach, QEVO generalizes the quantum variational optimization scheme to problems where the expectation value of the property of interest cannot be computed directly from $|M(\vec{\theta})\rangle$ using an operator. In fact, in the special case where the objective can be evaluated as an expectation value of an observable, QEVO, indeed, reduces to the standard VQA algorithm where a parameterized state is prepared, the observable $\langle O\rangle$ is estimated, and the parameters are optimized to minimize $\langle O\rangle$. However, such implementation is not possible since it does not allow for molecular generation, which is the focus of this work. Upon convergence in QEVO, molecules $|m_i\rangle$ with the desired property can be obtained by measurements of $|M(\vec{\theta})\rangle$ in the computational basis, effectively bridging variational quantum optimization with generative molecular design.

In fact, QEVO formulates inverse molecular design as a variational search problem over a molecular Hilbert space, operating on a quantum state that encodes a superposition of molecules in the search space. Since the sampling procedure arises from direct measurements of $|M(\vec{\theta})|^2$, QEVO can be viewed as a quantum realization of Variational Quantum Monte Carlo (qVQMC) algorithms (36, 37), in which the quantum circuit encoding the sampling distribution is variationally optimized with respect to an ensemble average. At the same time, we note that the bitstring sampling produced by a quantum circuit can, in principle, be emulated by a classical stochastic generator trained to reproduce the same distribution. In this sense, the subsequent evaluation of observables from sampled bitstrings (after decoding) is not uniquely quantum. However, the advantage of the quantum implementation lies in the intrinsic expressivity of the quantum circuit: For an n-qubit state, the underlying amplitude model spans $\mathcal{O}(2^n)$ basis states, and the circuit natively generates samples from this exponentially structured distribution through unitary operations and measurement. Thus, rather than explicitly constructing or storing a high-dimensional probability model, QEVO leverages the native Hilbert-space representation of the quantum device to encode correlations and interference across the molecular search space. In practice, measurements of the parameterized circuit yield candidate molecules distributed according to the squared amplitudes of the quantum state, enabling exploration of combinatorially large chemical spaces without maintaining an explicit $\mathcal{O}(2^n)$ classical representation. Updating the circuit parameters reshapes the sampling landscape by coherently modifying amplitudes over the superposed molecular state.

Results and Discussion

Inverse Design with QEVO. We evaluated QEVO in the context of molecular design targeting specific properties, including molecular solubility, predicted octanol-water partition coefficient (plogP), and drug-likeness (see *Materials and Methods* for details).

As a benchmark study, we first considered a tractable example where the full set of possible molecules could be explicitly characterized. Specifically, we generated all combinations of 6, 7, 8, and 9 ASCII tokens from the SELFIES encoding scheme (38) defining molecules drawn from a dictionary of 2^3 elements (see *Materials and Methods* for details), corresponding to combinatorial spaces of dimensionality 2^{18}, 2^{21}, 2^{24}, and 2^{27}, respectively. Due to degeneracies in the mapping between token

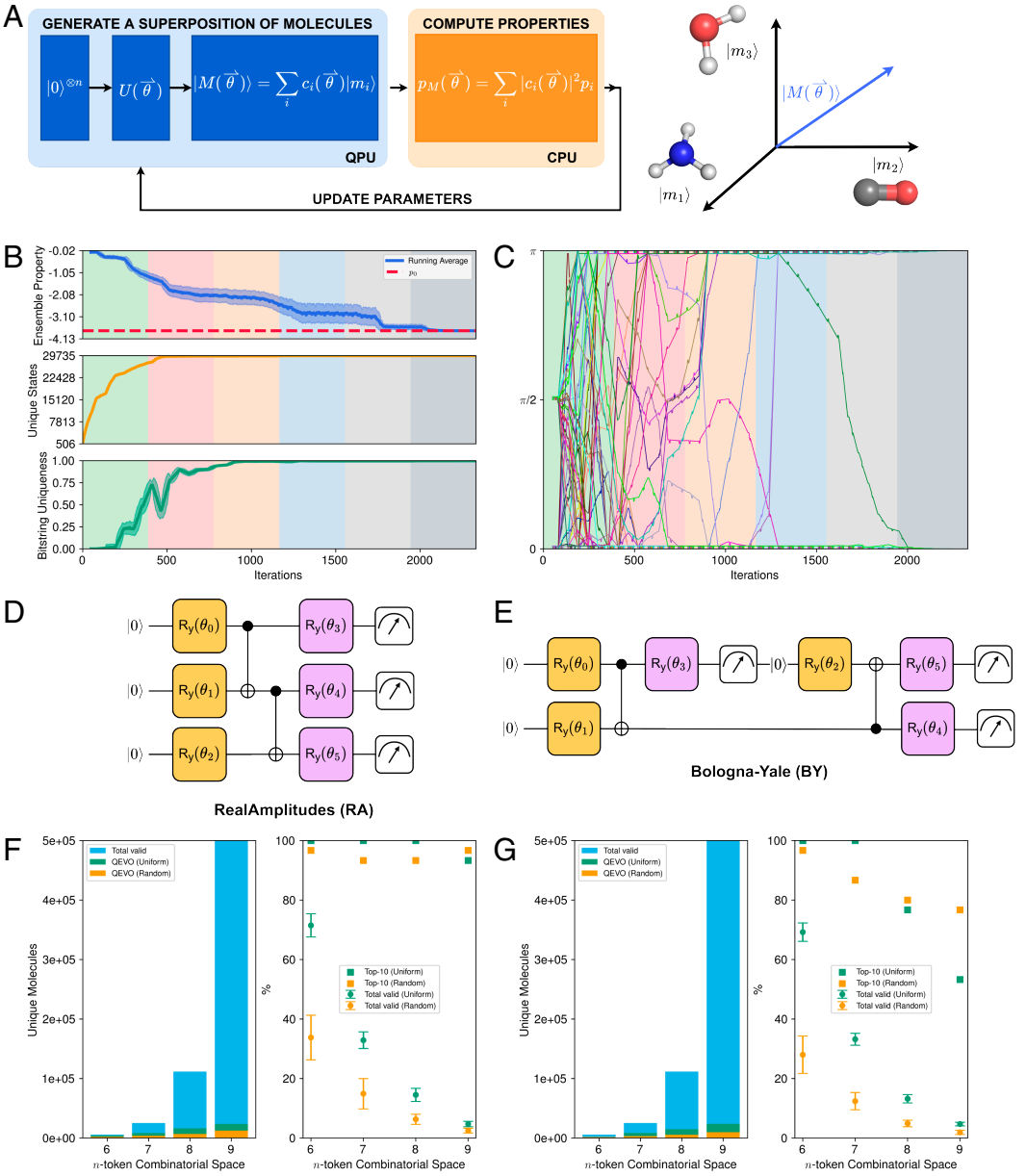

Fig. 1. The quantum ensemble variational optimization (QEVO) algorithm. (*A*) Schematic of the QEVO workflow. Variational parameters of a unitary transformation are optimized to generate quantum states for sampling candidate molecules. A classical computer evaluates the ensemble-averaged property value, p_M, and guides parameter updates. (*B*) Convergence behavior of QEVO during inverse design of molecules with up to 9 ASCII tokens targeting minimization of the plogP value. Shown are the ensemble-averaged property value (blue solid line), the best property value in the full reference space, p_0 (red dashed line), the cumulative number of unique molecules sampled (orange solid line), and the bitstring uniqueness (green solid line). Shaded areas denote running SDs over 50 samples. (*C*) Colored solid lines show the evolution of ansatz parameters over iterations. Panels (*B* and *C*) are segmented into six optimization windows, indicated by different background colors, to illustrate distinct phases during training. The colored iteration window segmentations are equally sized, each spanning one-sixth of the x-axis range. (*D* and *E*) Structure of the RealAmplitudes (RA, *D*) and Bologna-Yale (BY, *E*) ansätze. The BY ansatz is a holographic, qubit-efficient variant that uses iterative measurement and initialization of two qubits. (*F* and *G*) Comparison of QEVO's sampling behavior across different ansätze and initialization schemes. The total number of valid molecules in each *n*-token space is shown (blue histograms), alongside the number of unique molecules sampled using the RA (*F*) or BY (*G*) ansatz with uniform (orange) or random (green) initialization. Circular markers denote the average fraction of unique molecules explored under each scheme, and square markers indicate the success rate of generating at least one top-10-ranked solution. Results are averaged over 30 independent runs for each of the 6-, 7-, 8-, and 9-token combinatorial spaces, with 1,024 binary strings sampled per optimization step.

combinations and valid molecular representations, the number of unique, meaningful combinations was reduced to 5,790; 25,218; 111,711; and 504,183, respectively (blue bars in Fig. 1 *F* and *G*). However, it is important to highlight that this reduction does not affect the combinatorial complexity of the problems, which is still equal to the number of total combinations (hereafter called full reference chemical space).

Fig. 1*B* illustrates the performance of QEVO in a representative simulation using the RealAmplitudes (RA) ansatz. QEVO successfully identifies the solution with minimum plogP after 2,332 optimization steps, with 1,024 binary strings sampled per step. Remarkably, convergence is achieved after sampling only 29,735 unique combinations—corresponding to just 6% of all unique molecules in the full chemical space.

Upon initialization, the ansatz produces a uniform superposition, leading to uniform sampling of binary strings. As expected, the cumulative number of unique sampled molecules grows rapidly during the early stages of optimization (Fig. 1*B*). Over time, as the optimization progresses, the bitstring uniqueness increases and the quantum state $|M(\vec{\theta})\rangle$ gradually aligns with the bitstring that minimizes the loss function. In this case, the final solution corresponds to *n*-nonane, which exhibits the lowest loss with a predicted plogP of −3.7569.

QEVO's efficiency arises from the rapid convergence of the ansatz parameters (Fig. 1*C*), which evolve quickly to reduce the loss, despite initially generating poorly scoring candidates. The increasing bitstring uniqueness within the quantum state is reflected in the convergence of the ansatz rotation angles toward either 0 or π, corresponding to states $|0\rangle$ and $|1\rangle$, respectively, for each qubit (Fig. 1*C*).

We compared the performance of QEVO using both the RA ansatz (Fig. 1*D*) and its holographic implementation (39)—hereafter referred to as the Bologna-Yale (BY) ansatz—which operates by iteratively measuring and resetting two qubits (Fig. 1*E*). Both ansätze were tested under two initialization schemes: uniform superpositions and randomly sampled parameters. It is interesting to notice that the choice of selecting real-amplitude ansätze allows for implementing a variational approach with a number of parameters that is only polynomial in the number of qubits, which is particularly suitable for NISQ and near fault-tolerant hardware. Moreover, since QEVO's framework is device-agnostic, we envision that it virtually benefits from advances in error mitigation schemes and quantum error correction codes.

Across all configurations, QEVO demonstrated strong performance, consistently identifying high-quality molecular candidates while sampling only a small fraction of the full solution space (*SI Appendix*). As anticipated, the RA ansatz outperforms the BY variant, primarily due to the trade-off introduced by the latter's qubit-efficient design.

In simulations over the 2^{27}-dimensional chemical space, the RA and BY ansätze, under uniform initialization, yield solutions ranked in the top 10 of the reference set in 93% and 53% of the runs, respectively (*SI Appendix*). When initialized randomly, performance improves: RA achieves a 97% success rate, while BY reaches 77%. Given that the BY ansatz compresses a 27-qubit problem with only 2 qubits, these results are particularly noteworthy, highlighting QEVO's potential resource-adaptive nature and its capacity to operate effectively within the physical and coherence-time constraints of near-term quantum hardware. However, it is important to stress that the compression of the information in BY lacks preserving the original entanglement and correlation as in RA, making BY less expressive than RA while still useful for sampling procedures.

On average, uniform initialization yields better results than random initialization in smaller combinatorial spaces for plogP minimization task, even though uniform initialization requires more extensive exploration (Fig. 1 *F* and *G*). Interestingly, QEVO's performance improves as the size of the reference space increases, and the impact of the initialization scheme diminishes. This trend stems from the fixed number of binary strings sampled per iteration: As the size of the solution space grows, the relative proportion of unique states explored naturally decreases. For example, using the RA ansatz with uniform initialization, QEVO explores on average $72 \pm 4\%$ of unique valid states in the 2^{18} space, compared to only $5 \pm 1\%$ in the 2^{27} space (Fig. 1*F*). With random initialization, these exploration rates drop to $34 \pm 8\%$ and $3 \pm 1\%$, respectively. These trends illustrate QEVO's ability to maintain strong performance while requiring increasingly limited exploration in larger search spaces.

Stairways to New Drugs. We further evaluated QEVO in the context of drug-like molecule design (see *Materials and Methods* for details). When applied to the 2^{27}-dimensional combinatorial space using the RA ansatz with uniform initialization, QEVO explored on average $7 \pm 1\%$ of the unique valid chemical space and successfully identified candidates ranked in the top 10 of the reference set in 90% of simulations (*SI Appendix*, Fig. S1). These results are remarkable given the complexity of the underlying property landscape.

This difficulty is reflected in the nature of the final quantum state: Unlike in the plogP optimization tasks, the final state here never collapses to the computational basis vector encoding the pentanoic acid—the global optimum in this design problem (Fig. 2*B*). In contrast, for plogP minimization the final state aligns to the bitstring value corresponding to the optimal solution in 30% of runs when using the RA ansatz with uniform initialization, indicating a less rugged optimization surface in that case.

As shown in Fig. 2 for a representative simulation, even when QEVO does not converge to the global optimum, it effectively optimizes the quantum state to generate high-quality suboptimal superpositions of molecules. In this simulation, the final superposition reaches a bitstring uniqueness of 99% and explores 7% of all valid solutions (Fig. 2*A*). Despite not collapsing to the optimal state, QEVO samples 60% of the top-ranked solutions throughout the course of optimization (Fig. 2*B*). This behavior is further illustrated by a time-resolved principal component analysis (PCA) performed over the sampled molecules during the simulation (see *Materials and Methods* for details). The PCA projection was computed using principal components derived from the full reference space (Fig. 2 *C*, *Right* panel). The new molecules generated by QEVO were then projected onto this space to visualize the trajectory of the optimization process.

Consistent with uniform initialization, the algorithm initially samples a wide distribution of molecular states during the first sixth of the optimization (green subpanel, Fig. 3 *A* and *B*). As optimization proceeds, the distribution narrows and concentrates around high-reward regions, ultimately approaching the vicinity of the optimal state (orange star, Fig. 2*C*).

To further evaluate QEVO's performance, we compared its results with a classical baseline. Although numerous classical methods for drug discovery have been developed, including data-driven ML approaches (40) and genetics-inspired tools (41), they typically require large datasets for training and extensive hyperparameter tuning (42). This makes a like-for-like comparison with QEVO challenging. We therefore implemented a data-free reinforcement-learning (RL) framework for molecular generation that mirrors QEVO's workflow (*Materials and Methods*). The RL baseline uses a neural network policy with 10,185 trainable parameters, compared with only 54 parameters in the RA or BY ansätze. Despite its larger model size, the RL baseline explores chemical space less effectively, sampling only 0.9% of the total unique valid molecules, and its top-10 molecules substantially overlap with QEVO's outputs (6 out of 10). Finally, the baseline rapidly converges to a local minimum, underscoring the superior quantum-driven exploration achieved by QEVO (*SI Appendix*, Fig. S5).

JAK2 Inhibition. While exploring a combinatorial space of 2^{27} molecular candidates is already computationally demanding,

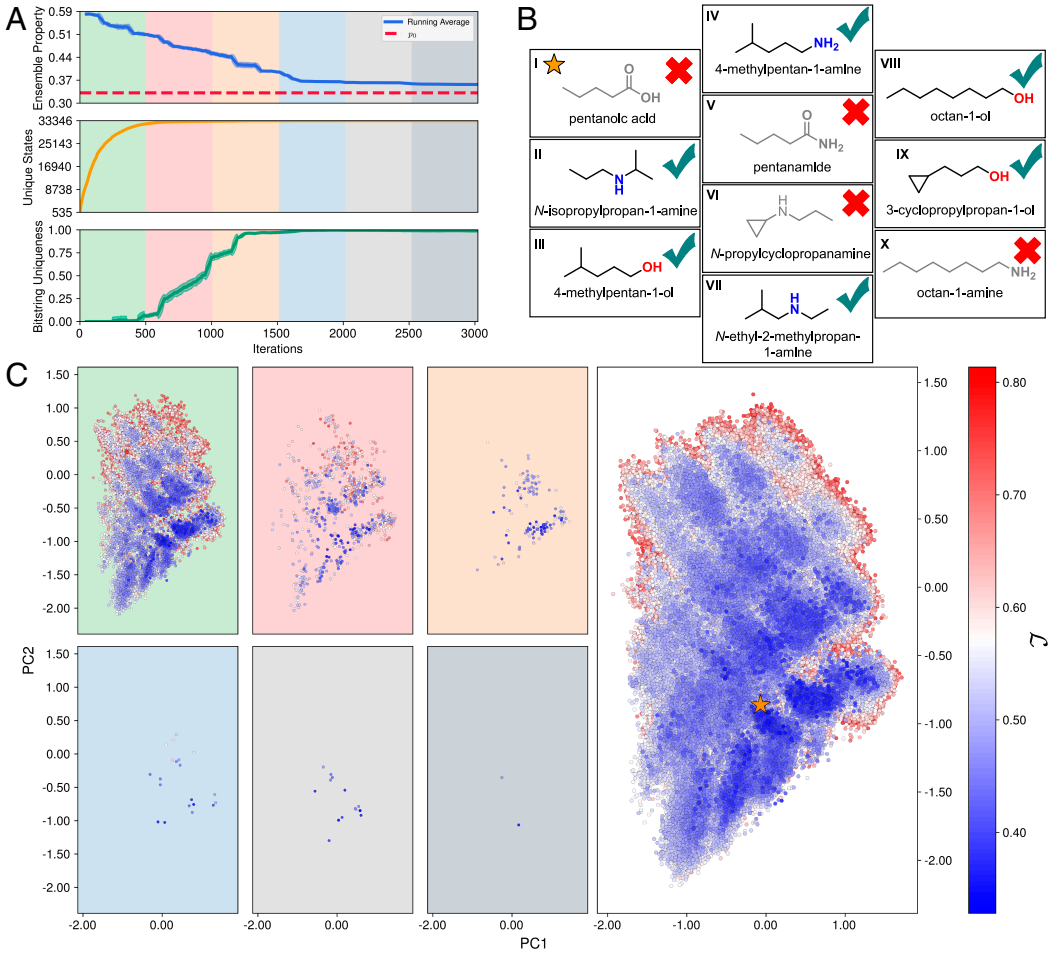

Fig. 2. Drug design with QEVO. Results from a representative QEVO simulation (out of 30 independent runs) for the inverse design of drug-like molecules composed of up to 9 ASCII tokens, using the RealAmplitudes ansatz with uniform initialization. (*A*) Convergence of the running average (over 50 samples) of the ensemble property value (blue solid line) toward the best value p_0 (red dashed line) within the full reference space. Also shown are the cumulative number of unique molecules sampled (orange solid line) and the evolution of the bitstring uniqueness (green solid line). Shaded areas denote running SDs over 50 samples. The optimization process is segmented into six stages, indicated by different background colors. The colored iteration window segmentations are equally sized, each spanning one-sixth of the x-axis range. (*B*) Top-10 molecules from the reference space are listed, and those sampled during the QEVO run reported in panel (*A*) are highlighted with a green tick. (*C*) Projection of newly generated unique molecules onto the principal component space computed from the full reference set (*Right* subpanel). Each point corresponds to a molecule sampled during a given optimization window (as defined in panel *A*) and is colored according to its associated loss function value. The global optimum, identified in the reference space, is marked with an orange star.

it still falls short of the scale required for many real-world drug discovery applications. Approved pharmaceuticals often consist of complex structures containing tens of atoms, which correspond to much longer ASCII strings than the 9-token examples considered thus far. A case in point is ruxolitinib (43), a clinically approved Janus kinase 2 (JAK2) inhibitor (44).

JAK2 mutations are a key driver of myelofibrosis, a rare hematological malignancy marked by the abnormal and excessive proliferation of blood cells (45). Developing selective JAK2 inhibitors remains particularly challenging due to the need to distinguish JAK2 from structurally similar off-target kinases—most notably, lymphocyte-specific protein tyrosine kinase (LCK), which is not implicated in myelofibrosis (46). The structural similarity between these kinases imposes a high selectivity barrier, making JAK2 inhibition an ideal benchmark for evaluating molecular design algorithms. In this context, QEVO offers a compelling testbed, as it must optimize candidate molecules not only for potency against JAK2 but also for selectivity over competing targets within a large and complex chemical space.

Prompted by these findings, we extended QEVO to the inverse design of JAK2 inhibitors using 40-token-long ASCII strings and a vocabulary of 2^4 tokens (see *Materials and Methods* for details), yielding a combinatorial design space of 2^{160} possible solutions, for which verifying the density of valid states is already computationally unfeasible. The choice of 40 tokens reflects the encoded size of ruxolitinib, while allowing additional flexibility to explore chemically diverse candidates. Since QEVO samples a finite number of bitstrings per iteration, it is important to highlight that the algorithm samples a finite number of bitstrings from the 2^{160}-dimensional space at each step, according to the distribution of $|c_i|^2$ (*Materials and Methods*).

We first targeted the generation of drug-like molecules, as described earlier, and selected the top 10,000 candidates generated by QEVO for postprocess molecular docking out of a total of 1,804,016 unique generated molecules. While QEVO successfully generated molecules with strong drug-like properties (QEVO-UNBIASED generation, *SI Appendix*, Fig. S2), their selectivity for JAK2 over LCK remained limited. To assess this, we compared the QEVO-generated molecules to a baseline of 100,000 randomly sampled compounds from the ZINC22 database (47). QEVO's candidates scored worse in terms of binding selectivity: The best-performing molecule from ZINC22

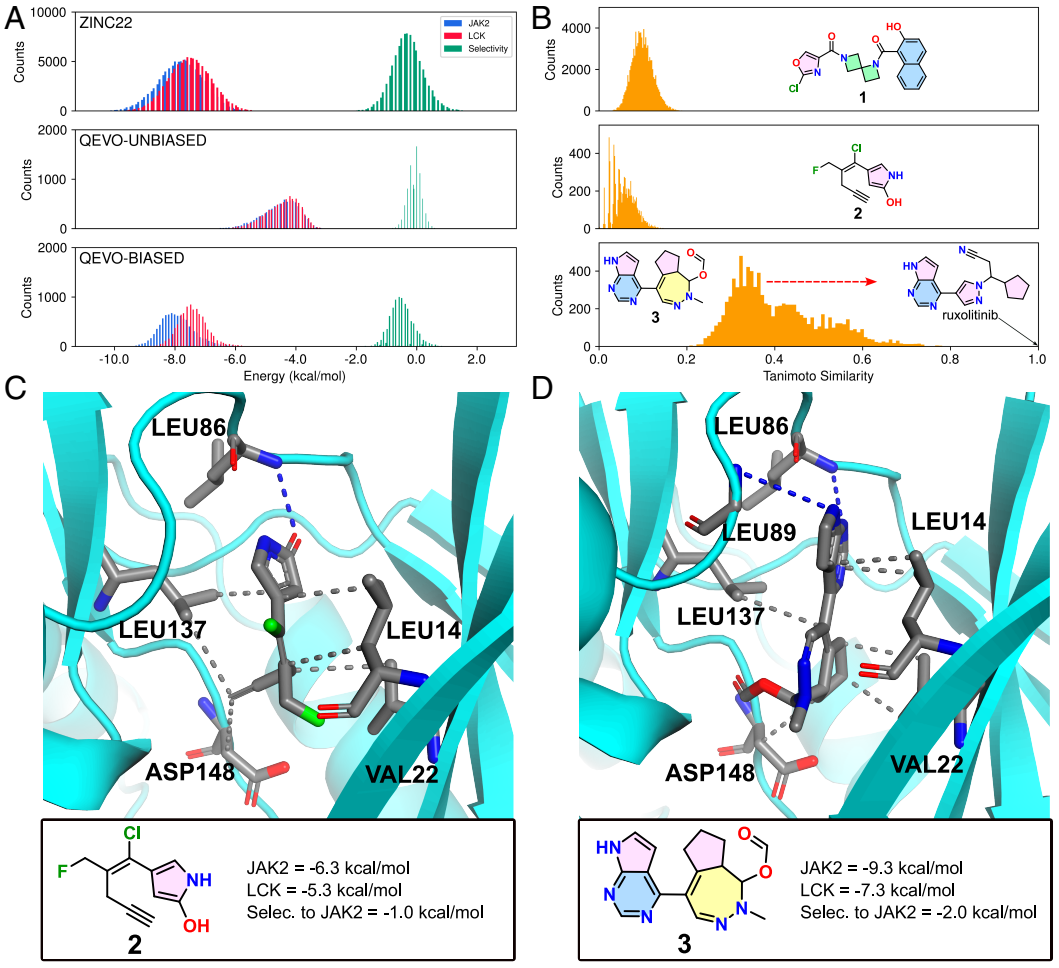

Fig. 3. Design of a selective JAK2 inhibitor. (*A*) Distributions of docking energies for candidate molecules against JAK2 (blue), LCK (crimson), and the binding energy difference favoring JAK2 over LCK (green). Results are shown for 100,000 randomly sampled compounds from the ZINC22 database, and the top 10,000 candidates generated by QEVO under unbiased and similarity-biased optimization schemes. (*B*) Distributions of chemical similarity to the reference inhibitor (ruxolitinib) across the three candidate sets in panel (*A*) are reported. In the *Insets*, the fittest candidates are reported for ZINC and the two QEVO simulations. (*C* and *D*) Docking poses of the top-scoring QEVO-generated molecules from the unbiased (*C*) and similarity-biased (*D*) design runs. Hydrogen bonds and hydrophobic interactions are indicated by blue and gray dashed lines, respectively.

exhibited a binding energy difference between JAK2 and LCK of −2.8 kcal/mol, while the top QEVO-derived compound showed only −1.0 kcal/mol (species **1** and **2**, respectively, Fig. 3*B*).

Interestingly, species **2**, designed by QEVO, acts as a type I kinase inhibitor, similar to ruxolitinib (44), forming hydrogen bonds with the hinge region via residue LEU86 (Fig. 3*C*) (48). In contrast, species **1** from ZINC22 is a type V inhibitor, interacting with multiple binding sites on JAK2 (*SI Appendix*, Fig. S6).

Relying solely on docking scores may not be sufficient for optimal drug design. Although ruxolitinib exhibits modest selectivity for JAK2—with a differential binding energy of only 0.6 kcal/mol as determined by our molecular docking studies—its FDA approval underscores its clinical relevance and provides valuable structural insights warranting further exploration. To this end, we evaluated the chemical similarity of each candidate molecule to ruxolitinib (see *Materials and Methods* for details). Neither molecules from the ZINC22 database nor those generated by QEVO approached close similarity to the reference compound (Fig. 3*B*). Consequently, we performed an additional QEVO experiment in which molecules were positively rewarded based on their similarity to ruxolitinib. To enhance convergence, the ansatz parameters were initialized so that the quantum state $|M(\vec{\theta})\rangle$ aligned with the encoding of ruxolitinib—effectively starting the optimization within a known, suboptimal subspace.

Docking of the top 10,000 QEVO-generated candidates (out of 3,566,364) from this biased run confirmed improved selectivity: Their binding affinities to JAK2 and LCK more closely resembled those of molecules in ZINC22, and their average similarity to ruxolitinib increased (QEVO-BIASED generation, Fig. 3 *A* and *B*). Specifically, the separation between the median affinities for JAK2 and LCK was 0.3 kcal/mol for ZINC22, 0.1 kcal/mol for unbiased QEVO, and increased to 0.5 kcal/mol in the biased QEVO simulation (Fig. 3*A*). The top candidate from the biased QEVO run, species **3** (Fig. 3*B* and *SI Appendix*, Fig. S3), exhibits a 2.0 kcal/mol stronger binding to JAK2 over LCK and shares 31% similarity with ruxolitinib. Like species **2**, species **3** forms hydrogen bonds with the hinge region via LEU86 and LEU89 residues (Fig. 3*D*), alongside hydrophobic interactions at multiple protein sites, indicative of a Type I kinase inhibitor. To further stress-test QEVO's capabilities, we performed again the QEVO-BIASED simulation but replacing in the loss function 2-dimensional descriptor values with the energy binding selectivity for those candidates featuring suboptimal 2-dimensional properties (*Materials and Methods*). Results showed that QEVO generated 10,964,103 unique valid

molecules, out of which 3,317 were evaluated using a docking scorer (*SI Appendix*, Fig. S8). Interestingly, the top-scored molecule features a 1.8 kcal/mol binding energy to JAK2 larger than LCK (*SI Appendix*, Fig. S4), highlighting that QEVO can effectively incorporate explicit 3-dimensional properties within its routine.

These findings highlight the capabilities of QEVO for molecular design, enabling rapid identification of promising anticancer candidates without reliance on training of neural networks. Furthermore, by adjusting ansatz initialization strategies, QEVO offers flexible control over chemical space exploration, making it a versatile tool for diverse inverse design challenges.

Conclusions

Inverse design in chemistry—and combinatorial optimization more broadly—remains a significant challenge due to the exponential size of the solution spaces, which precludes exhaustive enumeration and characterization. In this work, we introduced QEVO, a hardware-agnostic and derivative-free variational quantum algorithm that optimizes ensemble averages computed from molecules sampled from a distribution function defined by measurements of a quantum ansatz in the computational basis. We demonstrated QEVO as applied to molecular design. Our simulations highlight its effectiveness to generate high-quality molecular candidates.

We find that QEVO operates efficiently using either constant-depth or constant-qubit RealAmplitudes ansätze, requiring modest quantum resources. This makes it well-suited not only for medium-size problems, i.e. below 100 qubits, as they are currently accessible on several existing platforms, but also for timely, resource-constrained applications. This is the case of the design of novel anticancer drugs, for which a large yet still accessible number of qubits (160) is feasible on current devices.

Overall, QEVO represents a promising algorithm that leverages quantum superposition directly, circumventing the need for explicit Hamiltonian embedding. This approach opens a computational paradigm for tackling large-scale combinatorial problems.

Materials and Methods

The Mapping Scheme. Molecules are encoded using the SELFIES representation (38), which expresses each molecule as a sequence of discrete tokens t_i. To interface this representation with a quantum system, SELFIES strings are bijectively mapped to quantum states. Each token t_i is assigned a binary string $|q_i\rangle$, represented in the computational basis $\mathcal{C} = \{|0\rangle, |1\rangle\}$. To encode a vocabulary of 2^n unique SELFIES tokens, n qubits are required:

$$|q_i\rangle = \bigotimes_{j=1}^{n} |c_j\rangle, \quad \text{where} \quad |c_j\rangle \in \mathcal{C}. \tag{5}$$

Given the binary-encoded SELFIES tokens $|q_i\rangle \in \mathcal{Q}$, where $\mathcal{Q}$ is the token Hilbert space, a molecule m_i composed of up to k tokens is mapped to a unique quantum state $|m_i\rangle$ by concatenating the binary encodings of its constituent tokens:

$$|m_i\rangle = \bigotimes_{j=1}^{k} |q_j\rangle, \quad \text{where} \quad |q_j\rangle \in \mathcal{Q}. \tag{6}$$

Since both $\mathcal{C}$ and $\mathcal{Q}$ are orthonormal bases, the resulting molecular space $\mathcal{M}$, composed of all $|m_i\rangle$, also forms an orthonormal basis. When constructed using all possible token combinations of length up to k from a fixed vocabulary, $\mathcal{M}$ spans the entire relevant chemical design space.

Bitstring Uniqueness. The bitstring uniqueness $B(\vec{\theta})$ quantifies the diversity of computational basis states sampled from the coherent quantum state. It is defined as

$$B(\vec{\theta}) = 1 - \frac{|\text{set}(|M\rangle)|}{x_e}, \tag{7}$$

where x_e is the total number of samples (i.e., the number of times the quantum state is measured), and $|\text{set}(|M\rangle)|$ denotes the number of unique bistrings observed.

A uniqueness value close to 1 indicates that a small number of distinct bitstrings is sampled –suggesting convergence toward a specific molecular candidate–while values near 0 indicate a broad, highly diverse superposition of bitstrings.

Ansatz Implementation. We implemented two variational ansätze: the RealAmplitudes (RA) ansatz and the Bologna-Yale (BY) ansatz.

The RA ansatz applies a linearly entangled unitary transformation $U(\vec{\theta})$ composed of real amplitude rotations to the vacuum state. The circuit features alternating layers of single-qubit parameterized rotations and linear entangling gates. This structure ensures constant circuit depth and a number of trainable parameters $\vec{\theta}$ that scales linearly with the number of qubits.

The BY ansatz is a holographic implementation (39) of the RA circuit. Although multiple holographic implementations are possible, in the present work we chose to use only two physical qubits, which are repurposed by iterative measurement and reinitialization. As a result, the BY ansatz maintains a constant number of qubits and circuit depth, while the number of variational parameters $\vec{\theta}$ scales linearly with the size of the target mapped state. This trade-off allows the BY ansatz to achieve sampling schemes that are similar to the RA ansatz but using hardware-constrained systems. However, the qubit-compression employed in BY does not preserve the same level of entanglement and correlation present in RA, introducing an intrinsic limitation in the expressiveness of this ansatz.

Initialization Mode. To ensure a representative initial sampling of the solution space, the variational quantum circuit $U(\vec{\theta})$ is initialized in a uniform superposition over all possible basis states. Applying a Hadamard gate to each qubit,

$$H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle), \tag{8}$$

we obtain the state of the n-qubit register:

$$H^{\otimes n}|0\rangle^{\otimes n} = \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle, \tag{9}$$

as the hyperdiagonal of the molecular Hilbert space $\mathcal{M}$. This uniform initialization ensures that the quantum state $|M(\vec{\theta})\rangle$ initially contains equal probability amplitude over all possible molecular configurations encoded in the computational basis.

The RA and BY ansätze consist of two layers of parameterized $R_y(\theta)$ rotations separated by one (RA) or two (BY) entangling gates. Since $R_y(\frac{\pi}{2})$ acts identically to a Hadamard gate for the input state $|0\rangle$, we initialize the ansatz to approximate this behavior. Specifically, the angles of the R_y gates preceding the entangling operations are set to 0:

$$R_y(0)|0\rangle = |0\rangle, \tag{10}$$

ensuring that all control qubits in subsequent entangling gates (e.g., CNOT) remain in the $|0\rangle$ state

$$\text{CNOT}_{i,j}|00\rangle = |00\rangle. \tag{11}$$

Following entanglement, the second set of R_y gates is initialized to $\frac{\pi}{2}$, yielding

$$R_y\left(\frac{\pi}{2}\right)|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle), \tag{12}$$

which completes the transformation of the vacuum state into a uniform superposition. Thus, $U(\vec{\theta})|0\rangle^{\otimes n} = H^{\otimes n}|0\rangle^{\otimes n}$, ensures that the initial variational state spans the entire solution space for unbiased initial exploration of the chemical space.

Loss Functions. Here, we define the loss functions used for quantum optimization. These functions are evaluated for each sampled molecule m_i, according to chemical properties, or structural similarity to a reference molecular structure.

plogP loss. The first objective focuses on the optimization of the negative logarithm of the octanol-water partition coefficient (plogP) (49), which is widely used to assess molecular solubility and permeability. The loss function is defined as

$$\mathcal{L}(m_i) = \begin{cases} \text{plogP}(m_i) & \text{if } m_i \text{ is a valid molecule,} \\ 1.0 & \text{otherwise,} \end{cases} \quad [13]$$

where a penalty of 1.0 is applied to invalid molecular encodings.

Drug design loss. For the design of drug-like molecules, we define a composite loss function that integrates three terms:

- $a(m_i)$: A modified Quantitative Estimate of Drug-likeliness (QED)-based term penalizing low drug-likeness and large ring sizes (50),
- $b(m_i)$: A normalized synthetic accessibility score (SAS) (51),
- $c(m_i)$: A dissimilarity score based on 1 minus the Tanimoto similarity to a reference molecule m_{ref} (52).

These components are defined as follows:

$$a(m_i) = \begin{cases} 1.0 - \text{QED}(m_i) & \text{if } m_i \text{ has <=7-atom rings,} \\ 1.2 - \text{QED}(m_i) & \text{otherwise,} \end{cases}, \quad [14]$$

$$b(m_i) = \frac{\text{SAS}(m_i)}{10.0}, \quad [15]$$

$$c(m_i) = 1.0 - \text{S}(m_i, m_{\text{ref}}), \quad [16]$$

with $a \in [0.0, 1.2]$, $b \in [0.1, 1.0]$, and $c \in [0.0, 1.0]$. The final weighted loss is given by

$$\mathcal{L}(m_i) = \begin{cases} \dfrac{\alpha \cdot a(m_i) + \beta \cdot b(m_i) + \gamma \cdot c(m_i)}{\alpha + \beta + \gamma} & \text{if } m_i \text{ is valid,} \\ 1.0 & \text{otherwise,} \end{cases} \quad [17]$$

where α, β, γ are user-defined weights that control the relative importance of each objective.

All molecular properties (QED, SAS, plogP) and similarity scores are computed using the RDKit Python package (53), after converting SELFIES representations into SMILES strings (54–56).

When also docking values are included explicitly in the loss function, this is modified as follows:

$$\mathcal{L}'(m_i) = \begin{cases} \Delta G(JAK2 - LCK) & \text{if} \mathcal{L}(m_i) \leq 0.35, \\ \mathcal{L}(m_i) & \text{Eqn.17, otherwise} \end{cases} \quad [18]$$

Docking scores are evaluated as reported below.

Simulation Details. Quantum simulations were conducted in the Qiskit framework (v1.3.1) using the AerSimulator backend (qiskit-aer v0.15.1) with the matrix_product_state simulation method (57).

Token-based simulations (6–9 tokens). For simulations involving molecular encodings of 6 to 9 SELFIES tokens, we used the token-to-qubit mapping defined in *SI Appendix*, Table S1. Each optimization iteration sampled 1,024 computational basis states. The variational parameters were optimized using the Implicit Filtering (ImFil) algorithm from Qiskit Algorithms (v0.3.1). Drug-like molecule generation was guided by the weighted loss function described in this see *SI Appendix* file, with coefficients set to $\alpha = 2$, $\beta = 1$, and $\gamma = 0$.

Large-scale simulations (40 tokens). For higher-dimensional simulations (i.e., involving 40-token encodings), the mapping scheme in *SI Appendix*, Table S2 was employed. Each iteration involved sampling 10,240 computational basis states. Optimization was performed using the Simultaneous Perturbation Stochastic Approximation (SPSA) algorithm with 50 resampling evaluations per step, also from Qiskit Algorithms. In these simulations, the loss function included a similarity constraint to ruxolitinib, using coefficients $\alpha = 2$, $\beta = 1$, and $\gamma = 2$, where γ weights the Tanimoto similarity to the reference molecule.

Postprocessing. Principal Component Analysis (PCA) was performed using the scikit-learn Python package (58), with molecules encoded as 1,024-bit MorganFingerprints (radius = 3), as implemented in RDKit (53).

All molecular docking calculations were carried out using the Dockstring Python package (59), which provides a fully deterministic pipeline for docking molecules. Docking poses were analyzed with the Protein–Ligand Interaction Profiler (PLIP) web server (60), and visualizations were rendered using PyMOL (61).

Reinforcement Learning Baseline. Baseline calculations were performed using the stable baselines3 (v2.7.1) Python package (62) with proximal policy optimization (PPO) (63), as implemented in the library. We used the default MlpPolicy architecture and default hyperparameters. Training consisted of 3,000 epoch updates; each epoch comprised 1,024 episodes of 9 steps. We developed a custom environment to generate SELFIES sequentially, one token at a time. Intermediate states received zero reward ($r = 0$) while the terminal state received a reward $r = 1 - \mathcal{L}$, where $\mathcal{L}$ is the loss defined in Eq. **17**. This sign convention converts the minimization objective used in QEVO into a maximization objective, as required for RL. The action space was the set of allowable tokens that can be appended to construct the final SELFIES string. Policy optimization used PPO with batch_size=64 and n_epochs=3.

Large Language Model Usage. Grammar and stylistic improvements were made using ChatGPT (OpenAI, GPT-o4, 2024). Prompts were limited to requests for language refinement (e.g., "Improve clarity and conciseness of the following passage while retaining its technical meaning"). No scientific content, data analysis, or interpretation was generated by the tool.

Data, Materials, and Software Availability. Dataset containing raw simulations data have been deposited in AMSActa (https://doi.org/10.6092/unibo/amsacta/8972)(64). Some study data are available: Full and detailed description of the code, including its pseudocode, is provided to allow the full reproducibility of results. However, the source Python code has not been disclosed for copyright reasons.

ACKNOWLEDGMENTS. F.C. acknowledges the "Ing. Luciano Toso Montanari" Foundation for financially supporting his secondment at Yale University (U.S.A.) in the group of Professor V.S.B., and partial support from the NSF Engines Development Award: Advancing Quantum Technologies (CT) under Award Number 2302908. V.S.B. acknowledges support from the NSF Center for Quantum Dynamics on Modular Quantum Devices under grant number 2124511.

Author affiliations: [a]Department of Industrial Chemistry, Alma Mater Studiorum University of Bologna, Bologna 40129, Italy; [b]Center for Chemical Catalysis - C3, Alma Mater Studiorum University of Bologna, Bologna 40129, Italy; [c]Department of Chemistry, Yale University, New Haven, CT 06520; and [d]Yale Quantum Institute, Yale University, New Haven, CT 06511

1. J. G. Freeze, H. R. Kelly, V. S. Batista, Search for catalysts by inverse design: Artificial intelligence, mountain climbers, and alchemists. *Chem. Rev.* **119**, 6595–6612 (2019).
2. A. Aspuru-Guzik, R. Lindh, M. Reiher, The matter simulation (r)evolution. *ACS Cent. Sci.* **4**, 144–152 (2018).
3. A. M. Mroz, V. Posligua, A. Tarzia, E. H. Wolpert, K. E. Jelfs, Into the unknown: How computation can help explore uncharted material space. *J. Am. Chem. Soc.* **144**, 18730–18743 (2022).
4. C. Poree, F. Schoenebeck, A holy grail in chemistry: Computational catalyst design: Feasible or fiction? *Acc. Chem. Res.* **50**, 605–608 (2017).
5. M. Wang, X. Hu, D. N. Beratan, W. Yang, Designing molecules by optimizing potentials. *J. Am. Chem. Soc.* **128**, 3228–3232 (2006).
6. D. Xiao, W. Yang, D. N. Beratan, Inverse molecular design in a tight-binding framework. *J. Chem. Phys.* **129**, 044106 (2008).

7. F. De Vleeschouwer, W. Yang, D. N. Beratan, P. Geerlings, F. De Proft, Inverse design of molecules with optimal reactivity properties: Acidity of 2-naphthol derivatives. *Phys. Chem. Chem. Phys.* **14**, 16002–16013 (2012).
8. D. Xiao, L. A. Martini, R. C. Snoeberger III, R. H. Crabtree, V. S. Batista, Inverse design and synthesis of acac-coumarin anchors for robust TiO_2 sensitization. *J. Am. Chem. Soc.* **133**, 9014–9022 (2011).
9. L. Medrano Sandonas *et al*., "Freedom of design" in chemical compound space: towards rational in silico design of molecules with targeted quantum-mechanical properties. *Chem. Sci.* **14**, 10702–10717 (2023).
10. P. Kirkpatrick, C. Ellis, Chemical space. *Nature* **432**, 823–824 (2004).
11. A. M. Chang, B. Rudshteyn, I. Warnke, V. S. Batista, Inverse design of a catalyst for aqueous CO/CO_2 conversion informed by the Ni^{II}-iminothiolate complex. *Inorg. Chem.* **57**, 15474–15480 (2018).
12. T. Weymuth, M. Reiher, Gradient-driven molecule construction: An inverse approach applied to the design of small-molecule fixating catalysts. *Int. J. Quantum Chem.* **114**, 838–850 (2014a).
13. T. Weymuth, M. Reiher, Inverse quantum chemistry: Concepts and strategies for rational compound design. *Int. J. Quantum Chem.* **114**, 823–837 (2014b).
14. E. Martis, R. Radhakrishnan, R. Badve, High-throughput screening: the hits and leads of drug discovery-an overview. *J. Appl. Pharm. Sci.* **1**, 2–10 (2011).
15. D. C. Fara *et al*., Integration of virtual and physical screening. *Drug Discov. Today Technol.* **3**, 377–385 (2006).
16. L. M. Mayr, P. Fuerst, The future of high-throughput screening. *SLAS Discov.* **13**, 443–448 (2008).
17. K. T. Butler, D. W. Davies, H. Cartwright, O. Isayev, A. Walsh, Machine learning for molecular and materials science. *Nature* **559**, 547–555 (2018).
18. B. Sanchez-Lengeling, A. Aspuru-Guzik, Inverse molecular design using machine learning: Generative models for matter engineering. *Science* **361**, 360–365 (2018).
19. D. Schwalbe-Koda, R. Gómez-Bombarelli, *Generative Models for Automatic Chemical Design* (Springer International Publishing, 2020), pp. 445–467.
20. C. Bilodeau, W. Jin, T. Jaakkola, R. Barzilay, K. F. Jensen, Generative models for molecular discovery: Recent advances and challenges. *WIREs Comput. Mol. Sci.* **12**, e1608 (2022).
21. D. M. Anstine, O. Isayev, Generative models as an emerging paradigm in the chemical sciences. *J. Am. Chem. Soc.* **145**, 8736–8750 (2023).
22. B. Sridharan *et al*., Deep reinforcement learning in chemistry: A review. *J. Comput. Chem.* **45**, 1886–1898 (2024).
23. F. Calcagno *et al*., Quantum chemistry-driven molecular inverse design of stable isomers with data-free reinforcement learning. *J. Chem. Theory Comput.* **22**, 3373–3382 (2026).
24. Y. Cao *et al*., Quantum chemistry in the age of quantum computing. *Chem. Rev.* **119**, 10856–10915 (2019).
25. L. K. Grover, A fast quantum mechanical algorithm for database search. arXiv [Preprint] (1996). https://arxiv.org/abs/quant-ph/9605043 (Accessed 6 July 2026).
26. P. W. Shor, "Polynomial time algorithms for discrete logarithms and factoring on a quantum computer" in *Algorithmic Number Theory*, L. M. Adleman, M. D. Huang, Eds. (Springer, Berlin Heidelberg, Berlin, Heidelberg, 1994), pp. 289–289.
27. P. K. Barkoutsos, G. Nannicini, A. Robert, I. Tavernelli, S. Woerner, Improving variational quantum optimization using CVaR. *Quantum* **4**, 256 (2020).
28. A. Gilliam, S. Woerner, C. Gonciulea, Grover adaptive search for constrained polynomial binary optimization. *Quantum* **5**, 428 (2021).
29. E. Farhi, J. Goldstone, S. Gutmann, A quantum approximate optimization algorithm. arXiv [Preprint] (2014). https://doi.org/10.48550/arXiv.1411.4028 (Accessed 6 July 2026).
30. A. Peruzzo *et al*., A variational eigenvalue solver on a photonic quantum processor. *Nat. Commun.* **5**, 4213 (2014).
31. T. H. Kyaw *et al*., Boosting quantum amplitude exponentially in variational quantum algorithms. *Quantum Sci. Technol.* **9**, 01LT01 (2023).
32. K. Kanno *et al*., Quantum-selected configuration interaction: Classical diagonalization of hamiltonians in subspaces selected by quantum computers. arXiv [Preprint] (2023). https://doi.org/10.1103/dmn4-snfx (Accessed 6 July 2026).
33. M. Cerezo *et al*., Variational quantum algorithms. *Nat. Rev. Phys.* **3**, 625–644 (2021).
34. J. Zhan, Variational quantum search with shallow depth for unstructured database search. arXiv [Preprint] (2022). https://doi.org/10.48550/arXiv.2212.09505 (Accessed 6 July 2026).
35. A. P. Punnen, The quadratic unconstrained binary optimization problem. *Springer Int. Publ.* **10**, 978–983 (2022).
36. M. B. Mansky *et al*., "Sampling problems on a quantum computer" in *2023 IEEE International Conference on Quantum Computing and Engineering (QCE)* (Institute of Electrical and Electronics Engineers, 2023), vol. 01, pp. 485–495.
37. G. Mazzola, Quantum computing for chemistry and physics applications from a Monte Carlo perspective. *J. Chem. Phys.* **160**, 010901 (2024).
38. M. Krenn, F. Häse, A. Nigam, P. Friederich, A. Aspuru-Guzik, Self-referencing embedded strings (selfies): A 100% robust molecular string representation. *Mach. Learn. Sci. Technol.* **1**, 045024 (2020).
39. N. Lyu, P. Bergold, M. B. Soley, C. Wang, V. S. Batista, Holographic gaussian boson sampling with matrix product states on 3DcQED processors. *J. Chem. Theory Comput.* **20**, 6402–6413 (2024).
40. S. Dara, S. Dhamercherla, S. S. Jadav, C. M. Babu, M. J. Ahsan, Machine learning in drug discovery: A review. *Artif. Intell. Rev.* **55**, 1947–1999 (2022).
41. J. O. Spiegel, J. D. Durrant, Autogrow4: An open-source genetic algorithm for de novo drug design and lead optimization. *J. Cheminf.* **12**, 25 (2020).
42. B. L. Greenstein, D. C. Elsey, G. R. Hutchison, Determining best practices for using genetic algorithms in molecular discovery. *J. Chem. Phys.* **159**, 091501 (2023).
43. A. Quintás-Cardama *et al*., Preclinical characterization of the selective jak1/2 inhibitor incb018424: Therapeutic implications for the treatment of myeloproliferative neoplasms. *Blood* **115**, 3109–3117 (2010).
44. R. R. Davis *et al*., Structural insights into jak2 inhibition by ruxolitinib, fedratinib, and derivatives thereof. *J. Med. Chem.* **64**, 2228–2241 (2021).
45. R. L. Levine, A. Pardanani, A. Tefferi, D. G. Gilliland, Role of jak2 in the pathogenesis and therapy of myeloproliferative disorders. *Nat. Rev. Cancer* **7**, 673–683 (2007).
46. T. Zhou *et al*., Specificity and mechanism-of-action of the jak2 tyrosine kinase inhibitors ruxolitinib and sar302503 (tg101348). *Leukemia* **28**, 404–407 (2014).
47. B. I. Tingle *et al*., Zinc-22- a free multi-billion-scale database of tangible compounds for ligand discovery. *J. Chem. Inf. Comput. Sci.* **63**, 1166–1176 (2023).
48. R. Roskoski, Classification of small molecule protein kinase inhibitors based upon the structures of their drug-enzyme complexes. *Pharmacol. Res.* **103**, 26–48 (2016).
49. S. A. Wildman, G. M. Crippen, Prediction of physicochemical parameters by atomic contributions. *J. Chem. Inf. Comput. Sci.* **39**, 868–873 (1999).
50. G. R. Bickerton, G. V. Paolini, J. Besnard, S. Muresan, A. L. Hopkins, Quantifying the chemical beauty of drugs. *Nat. Chem.* **4**, 90–98 (2012).
51. P. Ertl, A. Schuffenhauer, Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions. *J. Cheminf.* **1**, 1–11 (2009).
52. D. Bajusz, A. Rácz, K. Héberger, Why is Tanimoto index an appropriate choice for fingerprint-based similarity calculations? *J. Cheminf.* **7**, 1–13 (2015).
53. G. Landrum, Rdkit documentation. *Release* **1**, 4 (2013).
54. D. Weininger, SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *J. Chem. Inf. Comput. Sci.* **28**, 31–36 (1988).
55. D. Weininger, A. Weininger, J. L. Weininger, SMILES. 2. Algorithm for generation of unique SMILES notation. *J. Chem. Inf. Comput. Sci.* **29**, 97–101 (1989).
56. D. Weininger, SMILES. 3. DEPICT. Graphical depiction of chemical structures. *J. Chem. Inf. Comput. Sci.* **30**, 237–243 (1990).
57. A. Javadi-Abhari *et al*., Quantum computing with qiskit. arXiv [Preprint] (2024). https://doi.org/10.48550/arXiv.2405.08810 (Accessed 6 July 2026).
58. F. Pedregosa *et al*., Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).
59. M. García-Ortegón *et al*., Dockstring: Easy molecular docking yields better benchmarks for ligand design. *J. Chem. Inf. Comput. Sci.* **62**, 3486–3502 (2022).
60. M. F. Adasme *et al*., Plip 2021: Expanding the scope of the protein-ligand interaction profiler to DNA and RNA. *Nucleic Acids Res.* **49**, W530–W534 (2021).
61. W. L. DeLano *et al*., An open-source molecular graphics tool. *CCP4 Newsl. Protein Crystallogr.* **40**, 82–92 (2002).
62. A. Raffin *et al*., Stable-baselines3: Reliable reinforcement learning implementations. *J. Mach. Learn. Res.* **22**, 1–8 (2021).
63. J. Schulman, F. Wolski, P. Dhariwal, A. Radford, O. Klimov, Proximal policy optimization algorithms. arXiv [Preprint] (2017). https://doi.org/10.48550/arXiv.1707.06347 (Accessed 6 July 2026).
64. F. Calcagno, D. G. A. Cabral, I. Rivalta, V. S. Batista, Dataset of The quantum ensemble variational optimization algorithm: Applications to molecular inverse design v2 (2026). AMS Acta. https://amsacta.unibo.it/id/eprint/8972/ [Dataset]. Deposited 6 July 2026.