

NRC Publications Archive Archives des publications du CNRC

Neural evolution structure generation: high entropy alloys

Tetsassi Feugmo, Conrard Giresse; Ryczko, Kevin; Anand, Abu; Singh, Chandra Veer; Tamblyn, Isaac

This publication could be one of several versions: author's original, accepted manuscript or the publisher's version. / La version de cette publication peut être l'une des suivantes : la version prépublication de l'auteur, la version acceptée du manuscrit ou la version de l'éditeur.

For the publisher's version, please access the DOI link below. / Pour consulter la version de l'éditeur, utilisez le lien DOI ci-dessous.

Publisher's version / Version de l'éditeur:

<https://doi.org/10.1063/5.0049000>

The Journal of Chemical Physics, 155, 4, 2021-07-22

NRC Publications Archive Record / Notice des Archives des publications du CNRC :

<https://nrc-publications.canada.ca/eng/view/object/?id=3b30e425-0723-4c7a-bc47-d313eaa92ca7>

<https://publications-cnrc.canada.ca/fra/voir/objet/?id=3b30e425-0723-4c7a-bc47-d313eaa92ca7>

Access and use of this website and the material on it are subject to the Terms and Conditions set forth at

<https://nrc-publications.canada.ca/eng/copyright>

READ THESE TERMS AND CONDITIONS CAREFULLY BEFORE USING THIS WEBSITE.

L'accès à ce site Web et l'utilisation de son contenu sont assujettis aux conditions présentées dans le site

<https://publications-cnrc.canada.ca/fra/droits>

LISEZ CES CONDITIONS ATTENTIVEMENT AVANT D'UTILISER CE SITE WEB.

Questions? Contact the NRC Publications Archive team at

PublicationsArchive-ArchivesPublications@nrc-cnrc.gc.ca. If you wish to email the authors directly, please see the first page of the publication for their contact information.

Vous avez des questions? Nous pouvons vous aider. Pour communiquer directement avec un auteur, consultez la première page de la revue dans laquelle son article a été publié afin de trouver ses coordonnées. Si vous n'arrivez pas à les repérer, communiquez avec nous à PublicationsArchive-ArchivesPublications@nrc-cnrc.gc.ca.

Neural evolution structure generation: High entropy alloys

Cite as: J. Chem. Phys. **155**, 044102 (2021); <https://doi.org/10.1063/5.0049000>

Submitted: 28 February 2021 . Accepted: 19 May 2021 . Published Online: 22 July 2021

 Conrard Giresse Tetsassi Feugmo,  Kevin Ryczko,  Abu Anand,  Chandra Veer Singh, and  Isaac Tamblyn

COLLECTIONS

Paper published as part of the special topic on [Computational Materials Discovery](#)

 This paper was selected as Featured

 This paper was selected as Scilight



View Online



Export Citation



CrossMark

ARTICLES YOU MAY BE INTERESTED IN

[Evolutionary neural network approach simulates large-scale high-entropy alloys](#)
Scilight **2021**, 301105 (2021); <https://doi.org/10.1063/10.0005590>

[Selfconsistent random phase approximation methods](#)

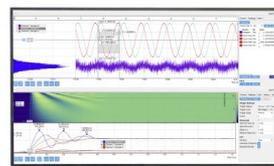
The Journal of Chemical Physics **155**, 040902 (2021); <https://doi.org/10.1063/5.0056565>

[The seven deadly sins: When computing crystal nucleation rates, the devil is in the details](#)

The Journal of Chemical Physics **155**, 040901 (2021); <https://doi.org/10.1063/5.0055248>

Challenge us.

What are your needs for
periodic signal detection?



Zurich
Instruments

Neural evolution structure generation: High entropy alloys



Cite as: J. Chem. Phys. 155, 044102 (2021); doi: 10.1063/5.0049000

Submitted: 28 February 2021 • Accepted: 19 May 2021 •

Published Online: 22 July 2021



View Online



Export Citation



CrossMark

Conrard Giresse Tetsassi Feugmo,^{1,a)} Kevin Ryczko,^{2,3} Abu Anand,⁴ Chandra Veer Singh,^{4,5} and Isaac Tamblyn^{1,2,3,b)}

AFFILIATIONS

¹National Research Council Canada, Ottawa, Ontario K1A 0R6, Canada

²Department of Physics, University of Ottawa, Ottawa, Ontario K1N 6N5, Canada

³Vector Institute for Artificial Intelligence, Toronto, Ontario M5G 1M1, Canada

⁴Department of Materials Science and Engineering, University of Toronto, Toronto, Ontario M5S 3E4, Canada

⁵Department of Mechanical and Industrial Engineering, University of Toronto, Toronto, Ontario M5S 3G8, Canada

Note: This paper is part of the JCP Special Topic on Computational Materials Discovery.

^{a)}Electronic mail: ConrardGiresse.TetsassiFeugmo@nrc-cnrc.gc.ca

^{b)}Author to whom correspondence should be addressed: isaac.tamblyn@nrc.ca

ABSTRACT

We propose a neural evolution structure (NES) generation methodology combining artificial neural networks and evolutionary algorithms to generate high entropy alloy structures. Our inverse design approach is based on pair distribution functions and atomic properties and allows one to train a model on smaller unit cells and then generate a larger cell. With a speed-up factor of ~ 1000 with respect to the special quasi-random structures (SQSs), the NESs dramatically reduce computational costs and time, making possible the generation of very large structures (over 40 000 atoms) in few hours. Additionally, unlike the SQSs, the same model can be used to generate multiple structures with the same fractional composition.

Published under a nonexclusive license by AIP Publishing. <https://doi.org/10.1063/5.0049000>

I. INTRODUCTION

Multicomponent alloy systems, such as High Entropy Alloys (HEAs) and Bulk Metallic Glasses (BMGs), have been in the physical metallurgy research spotlight over the past decade.^{1–3} HEAs are particularly interesting because of their superior structural and functional properties.^{2,4–7} In contrast to the conventional notion of alloying with a principal element (solvent) and alloying elements (solute), HEAs have four or more principal elements in near-equiatomic compositions.^{8–11}

Computational modeling is necessary for targeted and rapid HEA discovery and application.^{12–17} Constructing an appropriate atomic structure is the first step toward reliable predictions of materials properties. This includes predicting thermodynamic, kinetic, electronic, vibrational, and magnetic properties^{18–24} with first-principles method based simulation methodologies, such as Density Functional Theory (DFT). Indeed, DFT modeling of complex, random alloys requires defining a fixed-size cell,^{25–28} which

can introduce non-random periodicity. The inherent local disorder of HEAs makes this a non-trivial task.^{29–31} Special quasi-random structures (SQSs) designed to approximate the radial distribution function of a random^{32,33} system is a quintessential concept to generate realistic random structures when modeling disordered alloys with atomic resolution. Modern SQS generation approaches utilize techniques such as cluster expansions (CEs) in combination with Monte Carlo (MC) algorithms. Several codes are available in the literature, including ICET,³⁴ ATAT MCSQS,^{35,36} and *Supercell*,³⁷ which can generate SQSs for multi-component systems. Although very powerful, these approaches have significant computational overhead. A detailed analysis of the computation time with ICET with the number of atoms is presented in Sec. III A. Along with the computational complexity, present SQS generating techniques require the optimization of multiple parameters, including cluster space cutoffs, number of optimization steps, and simulated annealing temperatures for each system.^{34,35} These create a serious bottleneck in exploring multi-component

alloy systems using first-principles simulations and molecular dynamics.

An alternative is to use machine learning models to achieve the desired property.^{38–42} Recent works have used surrogate models,^{43,44} evolutionary algorithms^{40,42,45–48} (including canonical correlation analysis and cuckoo search), and generative adversarial networks^{39,49} to predict crystal structures and/or optimize the high entropy alloy composition. The inverse design framework that combines artificial neural networks (ANNs) and evolutionary algorithms (EAs) has also had success^{50,51} in generating structures that optimize some objective functions.

The generated structures can be used to collect descriptors such as structural stability, lattice vibrational property, electronic structure, elasticity, and stacking fault energy.

In this work, we build on previous work and present a neural evolution structure (NES) generation algorithm that combines ANNs and EAs to enable the search of HEAs that optimize the

configurational entropy. The NES generation algorithm differs from other approaches such as the cuckoo search⁴⁸ because no DFT calculations are required, and it allows one to train a model on smaller unit cells and then generate a larger cell.

In Sec. II, we outline our methodology including the general workflow of the algorithm, the crystal representation, and the fitness (or objective) functions. In Sec. III, we present our results. This includes a comparison of our algorithm with SQS with respect to performance and timing and analysis of the optimization parameters. In Sec. IV, we summarize and propose future work based on our findings.

II. METHODS

A. General workflow

In this work, we search for HEA structures that minimize the segregation or the maximum-entropy configuration. To do so, we

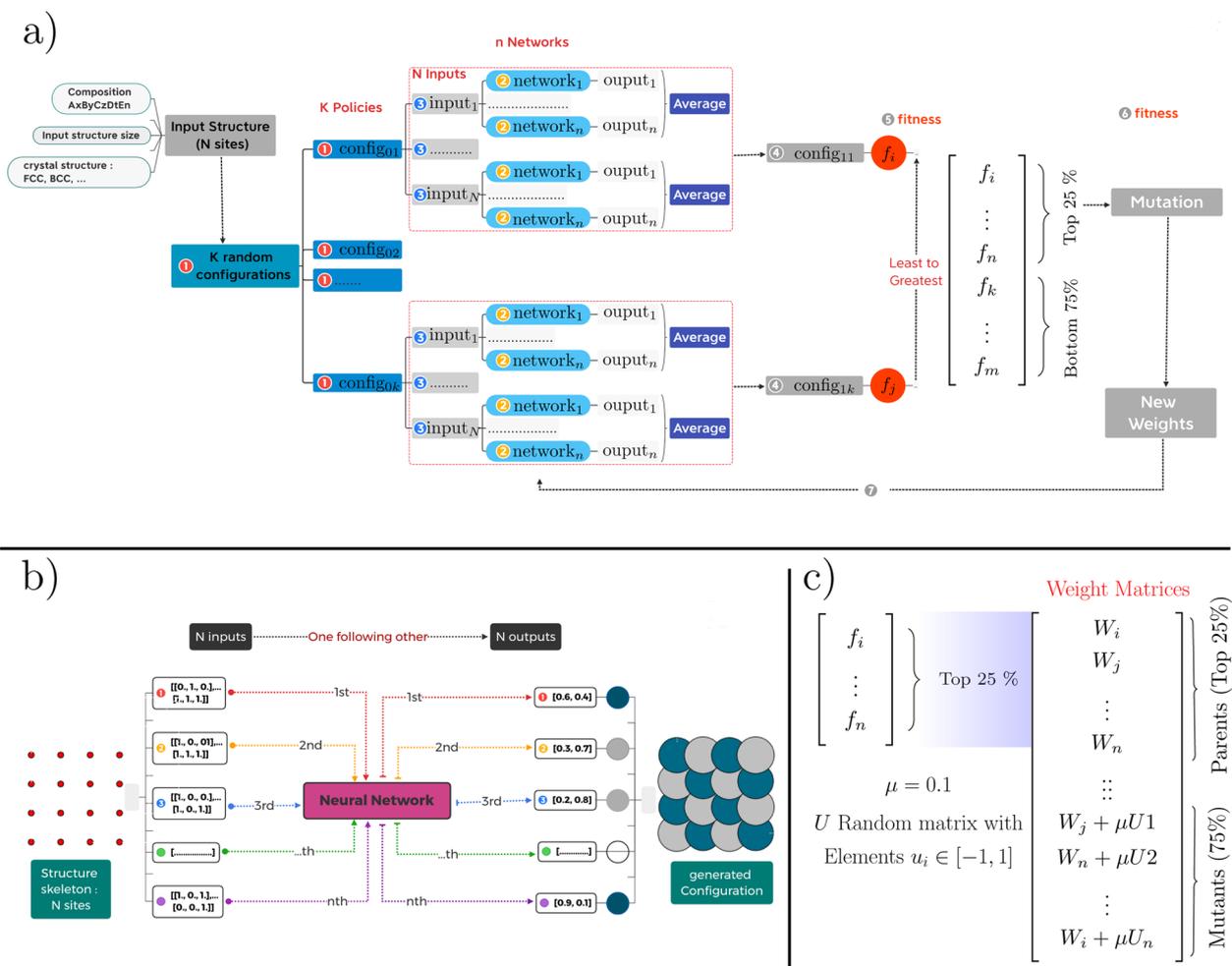


FIG. 1. Sketch of the algorithms required for the NES generation. (a) Key steps of the NES training process. (b) Road-map of the NES generation process. From left to right: an input structure (a template mesh with N sites), a representation of the N input arrays, the ANNs, a representation of the N output vectors, atom-type associated with each output vector, and the generated configuration. (c) Sketch of the mutation step. W_i is a weights matrix, and 25% represent the percentage of the matrix elements mutated.

consider configurations of HEAs that contain M different atomic species over N lattice sites. The workflow is summarized in Fig. 1, which combines ANNs and EAs. This methodology was used to optimize the doping of graphene-based three-terminal devices for valleytronic applications.⁵¹ This workflow can be divided into two processes: the training process [Fig. 1(a)] and the generation process [Fig. 1(b)]. After defining the crystal structure (FCC, BCC, HCP, etc.), the fractional composition ($A_\alpha B_\beta C_\gamma D_\delta E_\eta$), and the size of the supercell, the algorithm is as follows:

1. Place atoms randomly on the lattice following the fractional composition for N structure copies.
2. Initialize M ANN policies for each structure copy. One could have one ANN policy per structure or average over many ANN policies.
3. Generate input arrays (one input array per lattice site) based on the local environment of the lattice sites and feed the vectors into the ANN policies.
4. Based on the outputs of the ANN policies, reassign atoms to sites in the crystal for each structure copy.
5. Calculate the fitness function across each structure copy, sort from least to greatest, and order the associated structures.
6. Select the top 25% best-performing structures and randomly select and mutate the weights of the ANN policy to generate the remaining 75% of the population.
7. Go to step 2.

The training process is repeated until convergence is reached. The input vector is based on the environment of a lattice. The first element is a vector describing the properties of the atom (i.e., the atomic number, valence electrons, etc.) for the selected lattice site. The remainder of the input vector comes from the concatenation of the atomic properties of the neighboring and next-nearest neighboring lattice sites. The input vector, therefore, changes based on the lattice structure. The final input vector is flattened such that it can be passed into the ANN policy. We used the Softmax activation function to convert the ANN output vector into a vector of probabilities of assigning a certain chemical element to a certain lattice site. The index with the highest probability is extracted and matched to the list of elements (“A,” “B,” “C,” “D,” “E,” . . .), and the corresponding element is assigned to the considered site of the training structure (step 6).

Steps 3–6 are iterated over the remaining lattice sites until a new configuration is generated [Fig. 1(b)]. For each structure, M different policies are created, and then for each ANN policy, N configurations are generated and the corresponding fitness functions are computed (step 7). Finally, the average fitness of each policy is evaluated and the averages are sorted from least to greatest.

The top performing 25% ANN policies are kept, and the rest of the population are eliminated. To reproduce the next generation, the ANN policies from the top performers are randomly selected, and the weights are cloned and randomly mutated to generate the remaining 75% [Fig. 1(c)]. New atomic configurations are generated, and the process is repeated. The random mutations consist of adding a random matrix to the parents’ weight [see the following equation]:

$$\begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_l \end{pmatrix}_{new} = \begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_l \end{pmatrix}_{old} + \mu \cdot \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_l \end{pmatrix}, \quad (1)$$

where $\mu = 0.1$ is a small parameter (similar to a learning rate), $u_i \in [-1, 1]$ is a random number, and l is the number of weights. In this work, the number of weights is equal to the number of elements in an input array. All NE calculations in this report were run using a HP Z4 G4 Workstation (Intel Xeon, 16 × 4 GB RAM), whereas the SQS was generated using an 80 core single node (Intel Xeon Gold 6248 CPU @ 2.50 GHz) with 1 TB RAM.

B. Representation of crystal structures

We now describe the generation of the input vectors for each lattice site. The central idea of our approach is to use the pair distribution function (PDF) to characterize a crystal structure. Indeed, for crystals, the number of nearest neighbors and their positions depend on the crystal structure and the lattice parameters, respectively. In the FCC structure, each atom has 12 nearest neighbors (coordination number) at a distance $d = a\sqrt{2}/2$, 6 nearest neighbors at $d = a$, and 24 nearest neighbors at $d = a\sqrt{3}/\sqrt{2}$. In the BCC structure, each atom has 8 nearest neighbors at a distance $d = a\sqrt{3}/2$, 6 nearest neighbors at $d = a$, and 12 nearest neighbors at $d = a\sqrt{2}$.

The number of a -type atoms around an b -type atom is given by

$$N_{ab}(r_{\min}, r_{\max}) = 4\pi c_b \rho_0 \int_{r_{\min}}^{r_{\max}} r^2 g_{ab}(r) dr, \quad (2)$$

where r_{\min} and r_{\max} are the two radii values between which the coordination number is to be calculated and c_b is the fractional composition of b . The partial PDF $g_{ab}(r)$ between types of particles a and b reads

$$g_{ab}(r) = \frac{N}{\rho_0 N_a N_b} \sum_{i=1}^{N_a} \sum_{j=1}^{N_b} \langle \delta(|r_i - r_j| - r) \rangle, \quad (3)$$

where δ is a Dirac δ -function and $\rho_0 = N/V$ is the average density.

Each crystal site is represented by an array in which the elements are the atomic properties of the chemical element occupying the site and those of its nearest neighbors (Fig. 2). The number of rows corresponds to the number of nearest neighbors (NNs) plus 1 (NN + 1), and the number of columns is equal to the number of atomic properties describing each chemical element (P_1, P_2, \dots, P_n). The number of input vectors is equal to the number of sites in the crystal structure. The properties of the chemical element occupying the i th site are always stored in the first row of the i th input vector. These atomic properties can be classified into quantitative and qualitative variables. The quantitative variables include the atomic number, the number of valence electrons, the electronegativity, the oxidation state, and the atomic radius. The qualitative variables include the row and the group (metal, transition metal, alkali, and metalloid) in the Periodic Table. They are represented by integer and Boolean numbers, respectively.

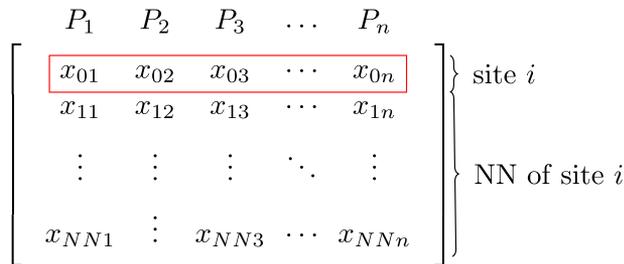


FIG. 2. Sketch of the NES input vector for a single crystal site. The P_i column corresponds to the i -property and NN is the number of nearest neighbors.

C. Fitness functions

Our objective is to reduce the segregation of chemical elements or to maximize the entropy of the configuration. Examples of equiatomic two- and four-component high entropy configurations are presented in Fig. 3 (top row) along with an equiatomic two-component random configuration (bottom row). Characteristics of four-component high entropy configurations have been studied using a $4 \times 4 \times 4$ supercell (64 atoms), and four functions characterizing the disorder in the crystal structures have been derived. For a site occupied by an A-type atom, we note the following:

- The first fitness function minimizes the number of A-type atoms occupying the nearest neighbor site in the first coordination shell N_{aa} . Knowing that the target is 0, the fitness defined as the root-mean-square deviation from 0 is

$$F_{AA}^1 = \sum_a \sqrt{\frac{\sum_{i=1}^{N_a} (N_{aa}^i - 0)^2}{N_a}}. \quad (4)$$

- The second fitness function maximizes the number of A-type atoms occupying the nearest neighbor site in the second coordination shell N_{aa} . If NN_2 is the number of

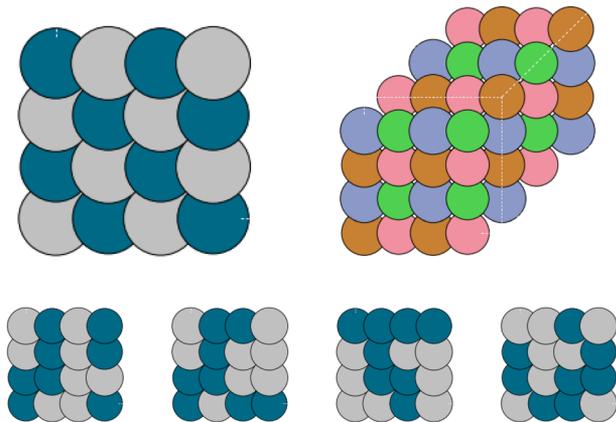


FIG. 3. Example of configurations for equiatomic two- and four-component systems. Top: high entropy configurations. Bottom: random configurations.

nearest neighbors in the second coordination shell, then the fitness function reads

$$F_{AA}^2 = \sum_a \sqrt{\frac{\sum_{i=1}^{N_a} (N_{aa}^i - NN_2)^2}{N_a}}. \quad (5)$$

- The third fitness function equalizes the number of other types of atoms occupying the nearest neighbor site in the first coordination shell N_{ab} and reads

$$F_{AB} = \sum_{a \neq b} \sqrt{\frac{\sum_{i=1}^{N_b} [N_{ab}^i - (c \cdot NN_1)]^2}{N_b}}, \quad (6)$$

where $c = c_a + [c_b/(s-1)]$. c_a and c_b are the target fractional compositions of a and b , respectively, and s is the number of atom-types. NN_1 is the number of nearest neighbors in the first coordination shell.

- The last fitness function checks how the maximum number of each type of atom (N_a, N_b, \dots) deviates from the target composition. These numbers are proportional to the fractional composition

$$F_N = \sqrt{\frac{\sum_a [N_a - (c_a \cdot N)]^2}{s}}. \quad (7)$$

The minimum of the total fitness [Eq. (8)] depends on both the fractional composition and the number of components, and it is not necessarily equal to 0. As an example, for a two-component system, the minimum will never be equal to 0,

$$F = F_{AA}^1 + F_{AA}^2 + F_{AB} + F_N. \quad (8)$$

III. RESULTS AND DISCUSSION

A. NES computation time

All calculations were carried out on equiatomic $\text{Cu}_a\text{Ni}_b\text{Co}_c\text{Cr}_d$ FCC alloy structures. Important aspects of the algorithm are the optimization and generation times, which depend on three parameters: (i) the number of policies optimized simultaneously, (ii) the size of the input structures, and (iii) the number of ANNs included in each policy. The three parameters have been investigated, and the results are presented in Fig. 4.

First, the average training time per generation as a function of the size of the input structure is shown in Fig. 4(a). This figure shows that the training time increases slowly with the size of the input structure (ratio of 1.4).

Second, Fig. 4(b) shows the average training time per generation as a function of the number of policies optimized. One observes a linear increase ($r = 0.99$). In addition, the slope also increases with the number of ANNs reaching 0.04, 0.05, and 0.06 for 1, 5, and 10 ANNs, respectively.

Finally, the average time per generation as a function of the size of the input structure is shown in Fig. 4(c). It increases slowly with

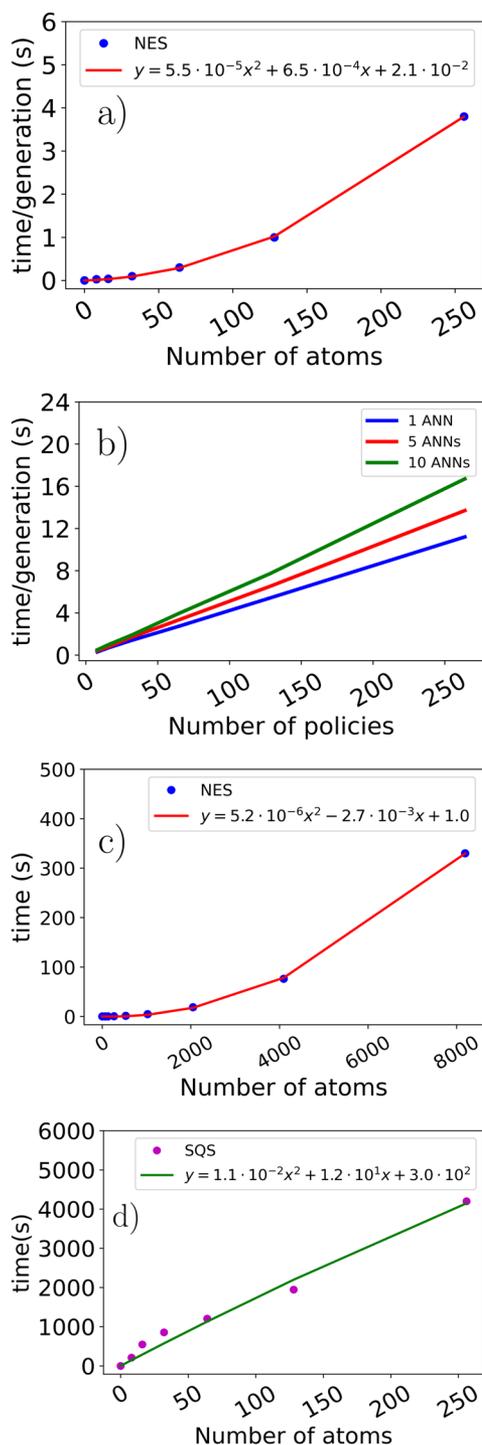


FIG. 4. NES computation time. (a) Average training time vs the size of input structure. Calculations were carried out using one policy made with one ANN. (b) Average training time per generation vs the number of policies. Calculations were carried out using a structure of 64 atoms. (c) Average NES generation time as a function of the size of the system. (d) ICET-SQS generation time as a function of the size of the system.

the size of the structures, going from a few tens of seconds (up to 256 atoms) to a few hundreds of seconds around 8000 atoms (ratio of 0.3).

Moreover, by comparing Figs. 4(a)–4(d), we observe that the SQSs scale up linearly [Eq. (9)] with time, whereas the NESs follow a x^2 polynomial behavior (but with a very low value of a and b) [Eq. (10)]. Knowing that the number of steps toward convergence increases with the increase in the number of atoms, and taking into account that the NE is trained on small clusters, it will always require fewer steps. Additionally, NESs can be sped up with multiprocessing. Taking into account all the previous remarks, we can derive a

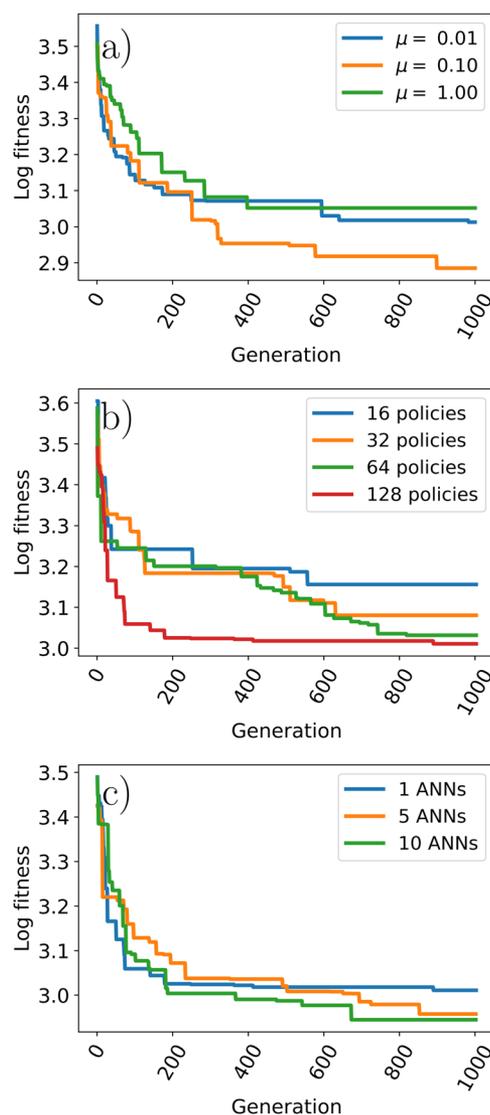


FIG. 5. NES training curves. (a) Optimization of the scaling factor μ . (b) Optimization of the number of policies trained simultaneously. (c) Optimization of the number of ANNs considered in each policy. Calculations were carried out on the input structure of 64 atoms and for an equiatomic $\text{Cu}_x\text{Ni}_y\text{Co}_z\text{Cr}_z$.

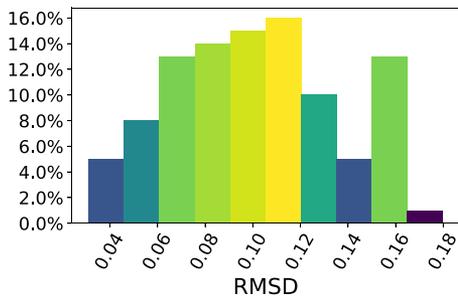


FIG. 6. Histogram of the root-mean-square deviations (RMSDs) from the target fractional composition. The RMSD values were computed with Eq. (11) derived from Eq. (7).

speed-up factor of ~ 1000 by comparing the x^2 coefficients in the following equations:

$$\text{SQSs: } y = 1.1 \cdot 10^{-2} x^2 + 1.2 \cdot 10^1 x + 3.0 \cdot 10^2, \quad (9)$$

$$\text{NESs: } y = 5.5 \cdot 10^{-5} x^2 + 6.5 \cdot 10^{-4} x + 2.1 \cdot 10^{-2}. \quad (10)$$

B. Convergence of NESs

In addition to the computation time, we now analyze the convergence of our algorithm. We show the results of optimizations with different parameters in Fig. 5. These parameters include μ [Fig. 5(a)], the number of policies trained [Fig. 5(b)], and the number of ANNs considered in each policy [Fig. 5(c)]. First, we investigated three values of μ (0.01, 0.1, and 1) and found that the best convergence is reached with 0.1. Second, the increase in the number of policies considered accelerates the convergence. Finally, in Fig. 5(c), we see that increasing the number of ANNs per policy does not improve the convergence rate but improves the quality of the solution.

However, it is worth noting that the minimum never reaches 0 for all three figures. For an equiatomic four-component system, the total fitness of the maximum-entropy configuration is equal to 0. The NES training process does not always converge to this maximum-entropy configuration, thus introducing imperfections that can be characterized by evaluating the deviation from the target

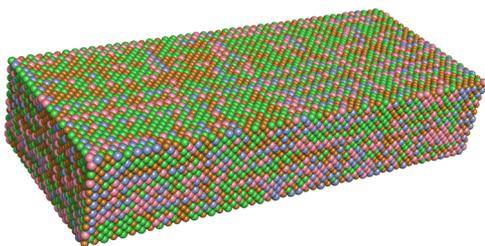


FIG. 7. Example of the $\text{Cu}_{0.234}\text{Ni}_{0.320}\text{Co}_{0.226}\text{Cr}_{0.220}$ NES structure (40 000 atoms). The model was trained on an eight-atom input-structure, and the generation was completed in 328 min.

composition. One hundred equiatomic $\text{Cu}_\alpha\text{Ni}_\beta\text{Co}_\gamma\text{Cr}_\zeta$ structures with 256 atoms were generated using the same NES model, and the root-mean-square deviation (RMSD) from the target fractional composition was computed as

$$\text{RMSD} = \sqrt{\frac{\sum_a [c'_a - c_a]^2}{s}}, \quad (11)$$

where c_a and c'_a are the target fractional composition of the a-type atom and the fractional composition of the a-type in the NESs and

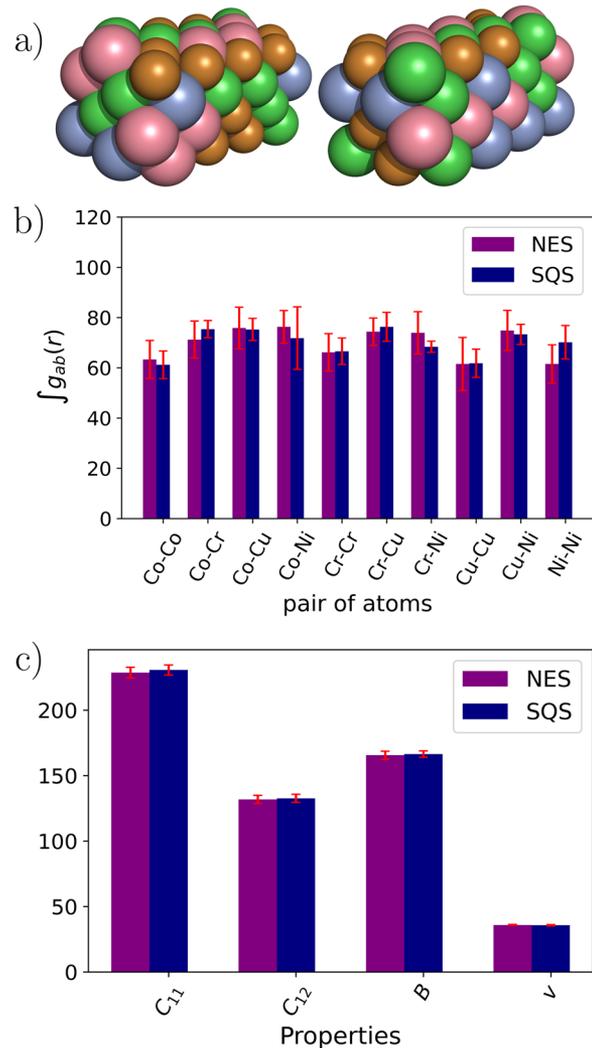


FIG. 8. NES vs SQS. (a) Selected 64-atom equiatomic $\text{Cu}_\alpha\text{Ni}_\beta\text{Co}_\gamma\text{Cr}_\zeta$ NES structures. (b) Comparison of partial PDFs [Eq. (2)]. Each bar corresponds to the average over ten structures of the area under the first peak of $g_{ab}(r)$. (c) Selected computed properties. C_{11} and C_{12} are the elastic constants (GPa), B is the bulk modulus (GPa), and ν is Poisson's ratio ($\times 100$). Properties were computed using the classical molecular dynamics.

s is the number of atom-types. The result is plotted in Fig. 6. The RMSD values vary from 0.03 to 0.18, and fractional composition associated with the minimum value is $\text{Cu}_{0.234}\text{Ni}_{0.258}\text{Co}_{0.258}\text{Cr}_{0.250}$. The true value in this example is $\text{Cu}_{0.25}\text{Ni}_{0.25}\text{Co}_{0.25}\text{Cr}_{0.25}$. To this end, multiple structures should be optimized in parallel and one should select the structure with the highest score (low fitness function).

C. FCC multicomponent alloys

NES generation was applied to build equiatomic $\text{Cu}_\alpha\text{Ni}_\beta\text{Co}_\gamma\text{Cr}_\zeta$ FCC alloy structures, and the performance was then evaluated by generating a structure with 40 000 atoms (Fig. 7). The model was trained with a $2 \times 2 \times 2$ cell (eight atoms), and the generation was completed in 328 min. In addition, a number of 64-atom structures were generated and selected properties were compared to ICET-SQS³⁴ (Fig. 8). Figure 8(a) shows selected structures used for the comparison.

The partial PDFs of the structures are compared in Fig. 8(b). Each bar corresponds to the average of the area under the first peak of $g_{ab}(r)$ (coordination numbers). The standard deviations are plotted in red. The purple bars represent NES, and the blue bars represent SQS. The analysis of the chart shows that NES is almost equivalent to SQS. Indeed, for each pair, the values of $g_{ab}(r)$ for NES are almost always within one standard deviation of the SQS values.

Second, the elastic constants, the bulk modulus, and Poisson's ratio were also investigated using classical molecular dynamics. These simulations were carried out using the LAMMPS molecular dynamics simulator,⁵² and an Embedded Atom Method (EAM) potential was used to define the inter-atomic interactions.⁵³ ICET-SQS and NES of equiatomic $\text{Cu}_\alpha\text{Ni}_\beta\text{Co}_\gamma\text{Cr}_\zeta$ were systematically deformed, and the change in virial stress tensor was used to calculate the elastic constants. Each deformed structure was energetically minimized using the conjugate gradient algorithm⁵⁴ before performing the stress calculations. All simulations were performed at 0 K. The bar chart of these properties is plotted in Fig. 8(b). All of the calculated values from the NES structure are within one standard deviation of the SQS method.

IV. CONCLUSIONS

We introduce and utilize a neural evolution structure (NES) generation methodology combining artificial neural networks (ANNs) and evolutionary algorithms (EAs) to generate High Entropy Alloys (HEAs). Our inverse design approach based on pair distribution functions and atomic properties dramatically reduces computational cost, allowing for the generation of very large structures with over 40 000 atoms in few hours. The computation time is reduced with a speed-up factor of about 1000 with respect to the SQSs. Unlike the SQSs, the same model can be used to generate multiple structures with the same fractional composition. A number of NE structures have been used to compute selected properties, such as the elastic constants, the bulk modulus, and Poisson's ratio, and the results are similar to those of structures generated with SQSs.

ACKNOWLEDGMENTS

K.R., C.V.S., and I.T. acknowledge support from the NSERC. Work at NRC was carried out under the auspices of the AI4D and MCF Programs.

DATA AVAILABILITY

Data sharing is not applicable to this article as no new data were created or analyzed in this study.

REFERENCES

- 1 J. J. Kruzic, *Adv. Eng. Mater.* **18**, 1308 (2016).
- 2 M.-H. Tsai and J.-W. Yeh, *Mater. Res. Lett.* **2**, 107 (2014).
- 3 J.-W. Yeh, "Overview of high-entropy alloys," in *High-Entropy Alloys: Fundamentals and Applications*, edited by M. C. Gao, J.-W. Yeh, P. K. Liaw, and Y. Zhang (Springer International Publishing, Cham, 2016), pp. 1–19.
- 4 S. Martin, K. Dennis, C. Zlotea, and J. Ulf, *Sci. Rep.* **6**, 36770 (2016).
- 5 A. Amiri and R. Shahbazian-Yassar, *J. Mater. Chem. A* **9**, 782 (2021).
- 6 C. Nyby, X. Guo, J. E. Saal, S.-C. Chien, A. Y. Gerard, H. Ke, T. Li, P. Lu, C. Oberdorfer, S. Sahu, S. Li, C. D. Taylor, W. Windl, J. R. Scully, and G. S. Frankel, *Sci. Data* **8**, 58 (2021).
- 7 E. J. Pickering, A. W. Carruthers, P. J. Barron, S. C. Middleburgh, D. E. J. Armstrong, and A. S. Gandy, *Entropy* **23**, 98 (2021).
- 8 Y. F. Ye, Q. Wang, J. Lu, C. T. Liu, and Y. Yang, *Mater. Today* **19**, 349 (2016).
- 9 Y. Zhou, D. Zhou, X. Jin, L. Zhang, X. Du, and B. Li, *Sci. Rep.* **8**, 1236 (2018).
- 10 Y. Ikeda, B. Grabowski, and F. Körmann, *Mater. Charact.* **147**, 464 (2019).
- 11 E. P. George, D. Raabe, and R. O. Ritchie, *Nat. Rev. Mater.* **4**, 515 (2019).
- 12 S. Wang, *Entropy* **15**, 5536 (2013).
- 13 Z. Li, F. Körmann, B. Grabowski, J. Neugebauer, and D. Raabe, *Acta Mater.* **136**, 262 (2017).
- 14 L. Rogal, P. Bobrowski, F. Körmann, S. Divinski, F. Stein, and B. Grabowski, *Sci. Rep.* **7**, 2209 (2017).
- 15 Y. Lederer, C. Toher, K. S. Vecchio, and S. Curtarolo, *Acta Mater.* **159**, 364 (2018).
- 16 J. Hu, J. Zhang, J. Zhang, H. Xiao, L. Xie, H. Shen, P. Li, G. Sun, and X. Zu, *Inorg. Chem.* **60**, 1388 (2021).
- 17 J. D. Strother and C. Z. Hargather, *Data Brief* **34**, 106670 (2021).
- 18 D. Ma, B. Grabowski, F. Körmann, J. Neugebauer, and D. Raabe, *Acta Mater.* **100**, 90 (2015).
- 19 F. Körmann, D. Ma, D. D. Belyea, M. S. Lucas, C. W. Miller, B. Grabowski, and M. H. F. Sluiter, *App. Phys. Lett.* **107**, 142404 (2015).
- 20 M. Troppenz, S. Rigamonti, and C. Draxl, *Chem. Mater.* **29**, 2414–2424 (2017).
- 21 D. Chattaraj and C. Majumder, *J. Alloys Compd.* **732**, 160 (2018).
- 22 D. Chen, S. Zhao, J. Sun, P. Tai, Y. Sheng, Y. Zhao, G. Yeli, W. Lin, S. Liu, W. Kai *et al.*, *J. Nucl. Mater.* **526**, 151747 (2019).
- 23 N. Zhao, Y. F. Zhu, and Q. Jiang, *J. Mater. Chem. C* **6**, 2854 (2018).
- 24 M. Rostami, M. Afkani, M. R. Torkamani, and F. Kanjouri, *Mater. Chem. Phys.* **248**, 122923 (2020).
- 25 M. C. Gao and D. E. Alman, *Entropy* **15**, 4504 (2013).
- 26 D. Wang, L. Liu, W. Huang, and H. L. Zhuang, *J. Appl. Phys.* **126**, 225703 (2019).
- 27 B. Yin and W. A. Curtin, *npj Comput. Mater.* **5**, 14 (2019).
- 28 J. Hu, J. Zhang, H. Xiao, L. Xie, H. Shen, P. Li, J. Zhang, H. Gong, and X. Zu, *Inorg. Chem.* **59**, 9774 (2020).
- 29 M. C. Gao, C. Niu, C. Jiang, and D. L. Irving, *High-Entropy Alloys* (Springer, 2016), pp. 333–368.
- 30 M. C. Gao, P. Gao, J. A. Hawk, L. Ouyang, D. E. Alman, and M. Widom, *J. Mater. Res.* **32**, 3627 (2017).
- 31 X. L. Ren, P. H. Shi, W. W. Zhang, X. Y. Wu, Q. Xu, and Y. X. Wang, *Acta Mater.* **180**, 189 (2019).

- ³²A. Zunger, S.-H. Wei, L. G. Ferreira, and J. E. Bernard, *Phys. Rev. Lett.* **65**, 353 (1990).
- ³³S.-H. Wei, L. G. Ferreira, J. E. Bernard, and A. Zunger, *Phys. Rev. B* **42**, 9622 (1990).
- ³⁴M. Ångqvist, W. A. Muñoz, J. M. Rahm, E. Fransson, C. Durniak, P. Rozyczko, T. H. Rod, and P. Erhart, *Adv. Theory Simul.* **2**, 1900015 (2019).
- ³⁵A. van De Walle, P. Tiwary, M. De Jong, D. L. Olmsted, M. Asta, A. Dick, D. Shin, Y. Wang, L.-Q. Chen, and Z.-K. Liu, *Calphad* **42**, 13 (2013).
- ³⁶A. van de Walle, M. Asta, and G. Ceder, *Calphad* **26**, 539–553 (2002).
- ³⁷K. Okhotnikov, T. Charpentier, and S. Cadars, *J. Cheminf.* **8**, 17 (2016).
- ³⁸C. Wen, Y. Zhang, C. Wang, D. Xue, Y. Bai, S. Antonov, L. Dai, T. Lookman, and Y. Su, *Acta Mater.* **170**, 109 (2019).
- ³⁹C. T. Chen and G. X. Gu, *Adv. Sci.* **7**, 1902607 (2020).
- ⁴⁰J. M. Rickman, H. M. Chan, M. P. Harmer, J. A. Smeltzer, C. J. Marvel, A. Roy, and G. Balasubramanian, *Nat. Commun.* **10**, 2618 (2019).
- ⁴¹K. Kaufmann, D. Maryanovsky, W. M. Mellor, C. Zhu, A. S. Rosengarten, T. J. Harrington, C. Oses, C. Toher, S. Curtarolo, and K. S. Vecchio, *npj Comput. Mater.* **6**, 42 (2020).
- ⁴²J. M. Rickman, G. Balasubramanian, C. J. Marvel, H. M. Chan, and M.-T. Burton, *J. Appl. Phys.* **128**, 221101 (2020).
- ⁴³K. Ryan, J. Lengyel, and M. Shatruk, *J. Am. Chem. Soc.* **140**, 10158 (2018).
- ⁴⁴H. Liang, V. Stanev, A. G. Kusne, and I. Takeuchi, *Phys. Rev. Mater.* **4**, 123802 (2020).
- ⁴⁵A. O. Lyakhov, A. R. Oganov, and M. Valle, “Crystal structure prediction using evolutionary approach,” in *Modern Methods of Crystal Structure Prediction* (John Wiley & Sons, Ltd., 2010), Chap. 7, pp. 147–180.
- ⁴⁶*Computational Materials Discovery*, edited by A. R. Oganov, G. Saleh, and A. G. Kvashnin (The Royal Society of Chemistry, 2019), Chap. 2, pp. 15–65.
- ⁴⁷E. V. Podryabinkin, E. V. Tikhonov, A. V. Shapeev, and A. R. Oganov, *Phys. Rev. B* **99**, 064114 (2019).
- ⁴⁸R. Singh, A. Sharma, P. Singh, G. Balasubramanian, and D. D. Johnson, *Nat. Comput. Sci.* **1**, 54 (2021).
- ⁴⁹S. Kim, J. Noh, G. H. Gu, A. Aspuru-Guzik, and Y. Jung, *ACS Cent. Sci.* **6**, 1412 (2020).
- ⁵⁰A. Zunger, *Nat. Rev. Chem.* **2**, 0121 (2018).
- ⁵¹K. Ryczko, P. Darancet, and I. Tamblin, *J. Phys. Chem. C* **124**, 26117 (2020).
- ⁵²S. Plimpton, *J. Comput. Phys.* **117**, 1 (1995).
- ⁵³D. Farkas and A. Caro, *J. Mater. Res.* **33**, 3218 (2018).
- ⁵⁴E. Polak and G. Ribiere, *ESAIM: Math. Modell. Numer. Anal.* **3**, 35 (1969).