

Simplifying Generation of Special Quasirandom Structures with ATAT Using Interactive Online Interface - SimplySQS

Miroslav Lebeda^{a,b,c}, Jan Drahokoupil^a, Petr Vlčák^a, Šimon Svoboda^a, Axel van de Walle^d

^a Faculty of Mechanical Engineering, Czech Technical University in Prague, Technická 4, 16607 Prague 6, Czech Republic

^b Faculty of Nuclear Sciences and Physical Engineering, Czech Technical University in Prague, Trojanova 339/13, 12000 Prague 2, Czech Republic

^c FZU – Institute of Physics of the Czech Academy of Sciences, Na Slovance 2, 18200 Prague 8, Czech Republic

^d School of Engineering, Brown University, 182 Hope Street, 02912 Providence, USA

Corresponding author: Miroslav Lebeda, lebedmi2@cvut.cz

Keywords: Special quasirandom structure (SQS), ATAT mcsqs, Random alloys, Graphical user interface (GUI), Streamlit application

Abstract

The special quasirandom structure (SQS) method is a widely used approach for modeling random alloys with periodic boundary conditions. Among available implementations, the Alloy Theoretic Automated Toolkit (ATAT) mcsqs module remains one of the most established tools for SQS generation. However, its command-line operation and input file preparation often present a steep learning curve, particularly for students and researchers without programming experience. To lower these barriers, we have developed *SimplySQS* (simplysqs.com), an online, browser-based interactive interface that automates the preparation, execution, and analysis of ATAT mcsqs outputs. SimplySQS allows users to upload or import structures from databases (MP, AFLOW, COD), configure compositions and supercells via guided forms, automatically generate ATAT input and batch scripts, and visualize results such as convergence behavior, correlation functions, and radial distribution functions. The best SQS outputs can be exported in standard formats (POSCAR, CIF, LMP, XYZ). By simplifying setup and minimizing file-handling errors, *SimplySQS* broadens access to ATAT mcsqs, making SQS generation more approachable for beginners while streamlining workflows for experienced users. The workflow is illustrated on the perovskite series $\text{Pb}_{1-x}\text{Sr}_x\text{TiO}_3$ (PSTO, including PbTiO_3 (PTO) and SrTiO_3 (STO)), where an all-in-one batch script

generated with *SimplySQS* was used to obtain SQSs across the composition range. The resulting structures were subsequently optimized using the MACE MATPES-r²SCAN-0 universal machine-learning interatomic potential (MLIP) to predict evolution of lattice parameters. The MLIP accurately reproduced the cubic-to-tetragonal transition near $x \approx 0.5$, with lattice parameters deviating by less than 1 % in the cubic ($x > 0.5$) and 3 % in the tetragonal ($x < 0.5$) regions.

1. Introduction

Accurate and efficient modeling of disordered alloys is essential in materials science, as many technologically significant materials adopt disordered alloy structures. One simple and transparent approach is to construct a large supercell and randomly assign atomic species to atomic sites. However, due to the finite size of the supercell, this method often results in local atomic arrangements that can deviate significantly from the true random state. The special quasirandom structure (SQS) approach [1] offers a systematic improvement over the random assignments. An SQS is a periodic supercell specifically designed so that its local atomic correlation functions closely approximate those of an ideal random alloy. In doing so, SQS provides an efficient and reasonably accurate model for disordered systems under periodic boundary conditions. Over the years, SQSs have been widely employed in *ab-initio* density functional theory (DFT) or molecular dynamics (MD) studies to investigate properties of various materials [2–10].

Several powerful software applications have been developed for generating SQSs. Among the most widely adopted are the mcsqs package [11] in Alloy Theoretic Automated Toolkit (ATAT) [12], the Integrated Cluster Expansion Toolkit (ICET) [13], and the Sqsgenerator Python package (github.com/dgehringer/sqsgenerator) [14]. ATAT, implemented in C++, is a long-established console-based tool, widely used in the community. ICET, a more recent Python-based framework, combines a Python interface with a C++ back-end, enabling efficient SQS generation and integration with common materials science packages such as Pymatgen [15] and Atomic Simulation Environment (ASE) [16]. The Sqsgenerator package, built with a Python interface and a C++ back-end, provides fast parallel execution via OpenMP and optional MPI. Its main current limitation is that it generates SQSs for a single user-specified supercell at a time (with lattice vectors restricted to simple multiples of the unit cell) and that it is limited to pair and triplet clusters. Among these three tools, we seek to provide a front-end for ATAT mcsqs, because of its large established user base, the extent of its capabilities and its integration with the broader ATAT toolkit.

The ATAT mcsqs significantly advanced the capabilities of researchers working in the field of disordered alloys. However, despite its power, the learning curve associated with it can pose a challenge, particularly for users without a strong programming background or experience with command-line environments. Preparing input structures, defining chemical configurations for atomic sites, specifying supercell parameters or number of atoms, and interpreting or converting output files require a deep understanding of the format settings and certain coding familiarity. However, as visible in the community forum (matsci.org/c/atat/), users often encounter various problems, mostly related to the input files. Many of these problems origin from misunderstandings of the required file formats. Therefore, simplifying the process of input file creation would greatly improve accessibility and adoption.

To address this barrier, we have developed *SimplySQS* (simplysqs.com), an online, interactive browser-based interface for ATAT mcsqs. An advantage of the online platform is that it requires no compilation and can be used on any device with an internet connection and a web browser. For cases where online access may be unavailable, the application can also be compiled and run locally (see *Section 2: Implementation Details*). The primary goal of *SimplySQS* is to streamline the preparation of input files, automate execution scripts, and provide analysis of ATAT mcsqs outputs. In doing so, *SimplySQS* lowers the entry barrier for newcomers while providing convenience for more experienced users. Currently, the key features of the application include:

- **Structure import:** Upload structures in common formats (POSCAR, CIF, LMP, XYZ with lattice) or retrieve them directly from databases such as the Materials Project (MP) [17], Automatic FLOW (AFLOW) [18,19], and Crystallography Open Database (COD) [20].
- **Configurable SQS setup:** Guided forms allow to define supercell sizes (number of atoms), atomic species and concentrations for atomic sites, or cluster's cutoff distances.
- **Automated ATAT input and script generation:** The platform creates all basic required mcsqs files (*'rndstr.in'*, *'sqscell.out'*) and a comprehensive batch script (*'monitor.sh'*) that automates the mcsqs run while providing information about the objective function and calculation time directly into the console (otherwise not previously available with mcsqs).
- **Output log parsing and visualization:** The platform allows users to upload ATAT mcsqs output files and automatically extract from them the convergence of the

objective function, correlation values, or partial radial distribution functions from the best SQS.

- **Direct export of created SQS:** Generated SQSs can be converted into widely used formats (POSCAR for VASP [21,22], CIF, LMP for LAMMPS [23], XYZ) for direct use in DFT or MD simulations.

In this manuscript, we describe the design and implementation of *SimplySQS* and demonstrate its capabilities through two illustrative examples: the preparation of input files for creating an SQS supercell of $\text{Sr}_{0.6}\text{Ba}_{0.4}\text{Ti}_{0.6}\text{Zr}_{0.4}\text{O}_3$ perovskite, and the generation of a complete series of SQSs for $\text{Pb}_{1-x}\text{Sr}_x\text{TiO}_3$ in order to evaluate the evolution of lattice parameters as a function of Sr content using the MACE MATPES-r²SCAN-0 [24] universal machine-learning interatomic potential (MLIP). Finally, we discuss the potential impact of *SimplySQS* on improving accessibility, reproducibility, and educational adoption of the ATAT mcsqs framework for modeling disordered alloys

2. Implementation Description

The *SimplySQS* application is implemented in Python and deployed online (simplysqs.com) as an interactive web interface using the Streamlit framework (streamlit.io). It is currently hosted on the free Streamlit Community Cloud server, which provides direct deployment from the public GitHub repository (github.com/bracerino/atat-sqs-gui). The repository documentation also includes a tutorial for local installation.

Within the application, Pymatgen [15] and ASE [16] are employed for structure import, manipulation, and file format conversion, while Matminer [25] is used to calculate partial radial distribution functions (PRDF). Numerical operations and tabular data management are handled by NumPy [26] and Pandas [27]. Structure visualization is provided through py3Dmol [28], and the graphs are created with Plotly and Matplotlib [29].

The crystal structures from the MP, AFLOW, and COD databases are obtained through their publicly available application programming interfaces (APIs). For MP and AFLOW, data retrieval is carried out using their corresponding Python packages that wrap their APIs, whereas access to COD is facilitated via a RESTful API. An overview of all Python packages grouped by their primary role used in *SimplySQS* is presented in **Tab. 1**.

Web Interface and deployment	Streamlit
------------------------------	-----------

Structure handling	pymatgen, ASE, mp-api, aflow, RESTful API
Data handling	NumPy, Pandas
Structure and data visualization	py3DMol, Plotly, Matplotlib

Tab. 1: Python packages used in the implementation of *SimplySQS*, grouped by their functionality.

3. Application Workflow and Features

SimplySQS is designed to guide users through the entire workflow of preparing input files and analyzing output files for the SQS generated with the ATAT mcsqs module (see its block diagram in **Fig. 1**). The process can be divided into four main stages: (i) structure input, (ii) SQS setup, (iii) input file generation, and (iv) analysis of ATAT outputs. A video tutorial demonstrating the workflow is also available at: implant.fs.cvut.cz/atat-sqs-gui. Below, we illustrate the main application functionality using the example of generating an SQS for the perovskite $(\text{Sr}_{0.6}\text{Ba}_{0.4})(\text{Ti}_{0.6}\text{Zr}_{0.4})\text{O}_3$, starting from the parent SrTiO_3 structure.

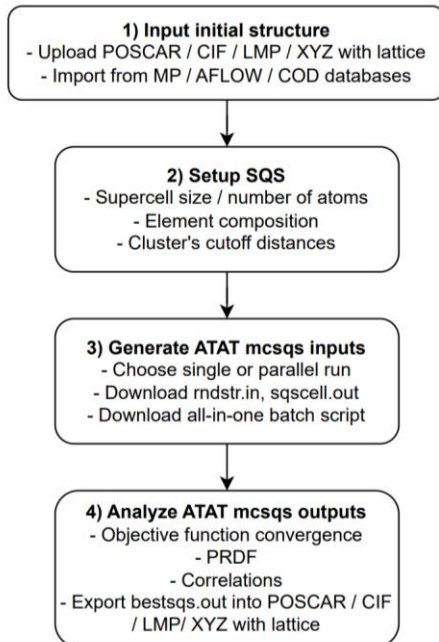


Fig. 1: Block diagram illustrating the main workflow of *SimplySQS* for the ATAT mcsqs module.

3.1. Structure Input

The workflow begins with uploading the initial crystal structure that will serve as the basis for SQS generation. *SimplySQS* accepts several widely used structure file formats, including POSCAR (VASP), CIF, LMP (LAMMPS), and XYZ (with lattice information). In addition to direct file upload, the interface provides integrated search tool for accessing structures from three crystal structure databases: MP, AFLOW, and COD. Users can query these databases directly within the application and import structures without leaving the interface. This retrieval approach is similar to that used in our crystallography-focused application, XRDlicious [30].

As an example, the initial structure of SrTiO_3 uploaded into the *SimplySQS* interface is shown in **Fig. 2**. The application automatically identifies unique Wyckoff positions, which provide the sublattices basis for defining element configurations on them. In our example case, the Wyckoff indices corresponding to Sr and Ti define two sublattices, where Sr will be partially substituted with Ba, while Ti will be partially substituted with Zr to create SQS for $(\text{Sr}_{0.6}\text{Ba}_{0.4})(\text{Ti}_{0.6}\text{Zr}_{0.4})\text{O}_3$ perovskite. Moreover, the application offers option to automatically convert the structure into its primitive cell.

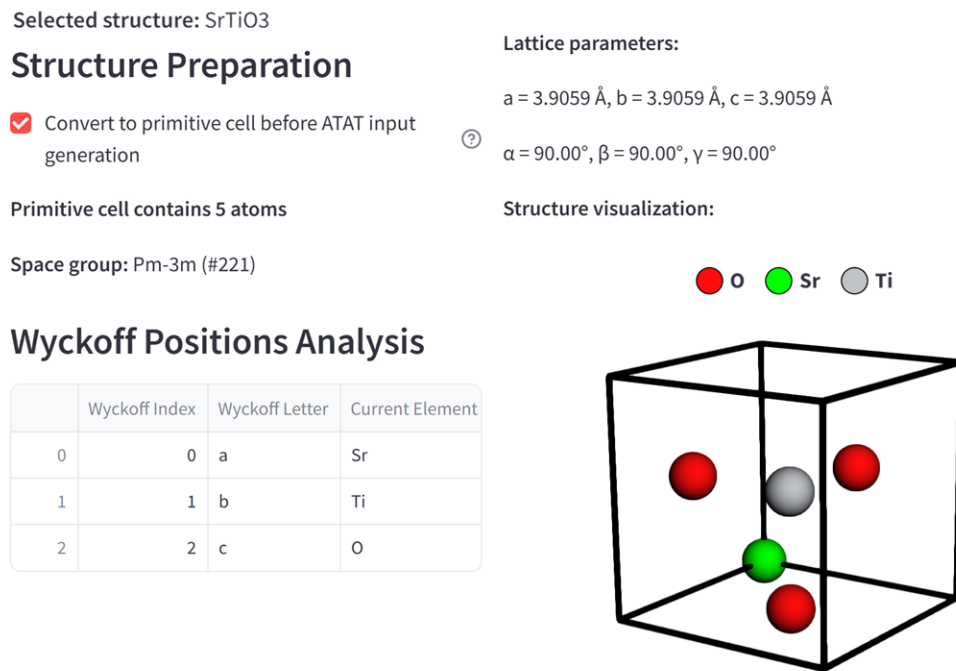


Fig. 2: Uploaded initial SrTiO_3 perovskite crystal structure within the *SimplySQS* interface.

3.2. SQS Setup and Input File Generation

Once the base structure is imported, users can configure the target disordered alloy through guided forms. Two modes are available:

- **Global concentration mode:** specify overall species concentrations that apply to all atomic sites.
- **Sublattice mode:** define compositions independently for each crystallographic sublattice.

Supercell dimensions can be set either by specifying exact size or by constraining the number of atoms without fixing supercell shape. Once the supercell is set, the available concentrations for each sublattice are automatically calculated within the interface. For the perovskite example, the sublattice mode was chosen, with $3 \times 3 \times 3$ supercell of SrTiO_3 , containing 135 atoms. This allows to create Ba and Zr concentrations close to 40 % - $(\text{Sr}_{0.59}\text{Ba}_{0.41})(\text{Ti}_{0.59}\text{Zr}_{0.41})\text{O}_3$. **Fig. 3** illustrates the interface for the configuration of sublattice A, corresponding to the initial element of Sr.

Elements for sublattice A: ? Set concentrations for sublattice A:

Sr x Ba x x v

Sr fraction: 0.592593

0.000000 1.000000

Ba: 0.407407 (automatic)

☒ Total fraction = 1.000000

Resulting atom counts:

- Sr: 16.0 atoms
- Ba: 11.0 atoms

Fig. 3: Definition of element types and concentrations for the A-site sublattice within the *SimplySQS* interface.

Furthermore, users can choose whether they wish to run the mcsqs in either single or parallel mode. In parallel mode, independent searches for SQSs are launched with different initial seeds. For example, specifying 5 parallel runs will generate 5 SQS candidates. Additionally, users can adjust the cut-off distances for cluster interaction parameters (for pairs and optionally also for triplets and quadruplets). To simplify this step, *SimplySQS* can calculate nearest-neighbor distances (up to the 6th shell, normalized to the largest lattice

parameter) for the chosen structure. For instance, as indicated in **Fig. 4**, including pairs up to the 6th nearest neighbor in the Sr-Ba sublattices corresponds to a pair cutoff of ~ 2.0 Å (normalized unit cell).

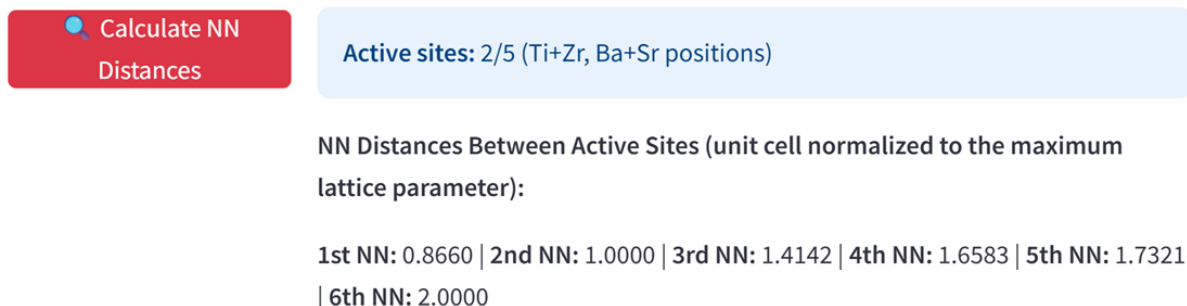


Fig. 4: Illustration of calculated nearest neighbor distances within the Sr and Ba sublattices from the *SimplySQS* interface.

Once the parameters are set, the platform automatically generates required ATAT input files (*'rndstr.in'*, *'sqscell.out'*) along with an all-in-one batch script (*'monitor.sh'*). This script streamlines execution by:

1. Automatically creating input files with the specified parameters.
2. Launching the mcsqs search with the chosen cluster settings.
3. Printing real-time progress to the console, including elapsed runtime and best objective function values (supporting both single and parallel runs) as well as creating a new file which contains time-dependent progress of objective function (*'mcsqs_progress.csv'* for a single run or *'mcsqs_parallel_progress.csv'* for a parallel run).
4. Allowing users to set a maximum runtime for the mcsqs search and, upon completion or manual termination, automatically converting the final bestsqscell.out into POSCAR, ensuring direct compatibility with VASP and other simulation packages.

An additional advantage of the all-in-one script is that it facilitates reproducibility: the same script can be easily shared and rerun on different computers, ensuring that the SQS search process can be reliably reproduced by other researchers.

The content of *'rndstr.in'*, *'sqscell.out'*, and the first command which generates the information about clusters for the set example of $(\text{Sr}_{0.59}\text{Ba}_{0.41})(\text{Ti}_{0.59}\text{Zr}_{0.41})\text{O}_3$ is shown in **Fig. 5**.

Generated Files

 rndstr.in

```
1.000000 1.000000 1.000000 90.00 90.00 90.00
1 0 0
0 1 0
0 0 1
0.000000 0.000000 0.000000 Ba=0.407407,Sr=0.592593
0.500000 0.500000 0.500000 Ti=0.592593,Zr=0.407407
0.000000 0.500000 0.500000 O=1.000000
0.500000 0.000000 0.500000 O=1.000000
0.500000 0.500000 0.000000 O=1.000000
```

 Download rndstr.in

 sqscell.out

```
1
3 0 0
0 3 0
0 0 3
```

 Download sqscell.out

ATAT Commands to Run

```
# Step 1: Generate cluster information, -noe: not including empty cluster, -nop: not including point cluster(s)
corrdump -l=rndstr.in -ro -noe -nop -clus -2=2.1 -3=2.0
```

Fig. 5: Content of ‘rndstr.in’, ‘sqscell.out’ input files, and the first command to run that generates the cluster information for ATAT mcsqs search for SQS of $(\text{Sr}_{0.59}\text{Ba}_{0.41})(\text{Ti}_{0.59}\text{Zr}_{0.41})\text{O}_3$.

To further enhance usability, *SimplySQS* includes a binary concentration sweep mode. When users wish to generate SQSs across concentration range of two elements (e.g., $\text{Sr}_{1-x}\text{Ba}_x\text{TiO}_3$, where $0 \leq x \leq 1$), the program automatically detects that the binary option has been selected and user can use it to generate dedicated all-in-one batch script. This script automatically performs SQS searches for each concentration, creating a separate folder for every case and running the search for the user-defined total runtime (see **Fig. 6** for interface parameters). Once a search is complete, the best structure (‘bestsqs.out’) is automatically converted into POSCAR format, eliminating the need for manual conversion from mcsqs to formats compatible with external simulation packages. Users can also customize parallelization settings: (i) the number of independent mcsqs runs per concentration and (ii) the number of concentrations processed simultaneously. A video tutorial demonstrating this feature is available also at: implant.fs.cvut.cz/atat-sqs-gui.

Available Ba concentrations for sublattice A: 26

Select Ba concentrations for SQS generation:

0.037037 x	0.074074 x	0.111111 x	0.148148 x
0.185185 x	0.222222 x	0.259259 x	0.296296 x
0.333333 x	0.37037 x	0.407407 x	0.444444 x
0.481481 x	0.518519 x	0.555556 x	0.592593 x
0.62963 x	0.666667 x	0.703704 x	0.740741 x
0.777778 x	0.814815 x	0.851852 x	0.888889 x
0.925926 x	0.962963 x		

Choose MCSQS mode: ?

☒ Supercell Mode
☐ Atom Count Mode

Uses -rc flag (predefined supercell)

Time per concentration (minutes) ?
 30.00 - +

Maximum parallel jobs ?
 4 - +

Parallel runs per concentration ?
 5 - +

Progress update interval (seconds) ?
 10 - +

Fig. 6: Interface of the binary concentration sweep mode showing the configurable parameters: selected mcsqs mode, total search time, number of parallel runs per concentration, number of concentrations executed simultaneously, and the update interval for printing progress information to the console.

3.3 Analysis of ATAT mcsqs Outputs

The *SimplySQS* interface also facilitates post-processing and evaluation of results generated by ATAT mcsqs. In addition to simplifying the preparation of input files, it provides tools for converting output structures into standard formats and for analyzing the quality of the generated SQS.

Optimized SQS (*'bestsqs.out'*) can be imported into the interface and converted into widely used formats (**Fig. 7**), including POSCAR for VASP, LMP for LAMMPS, CIF, extended XYZ (with lattice information). The best multiple SQSs obtained from parallel run can be also uploaded and batch-converted simultaneously. Moreover, partial radial distribution functions from the best uploaded SQS can be directly calculated and visualized within the interface.

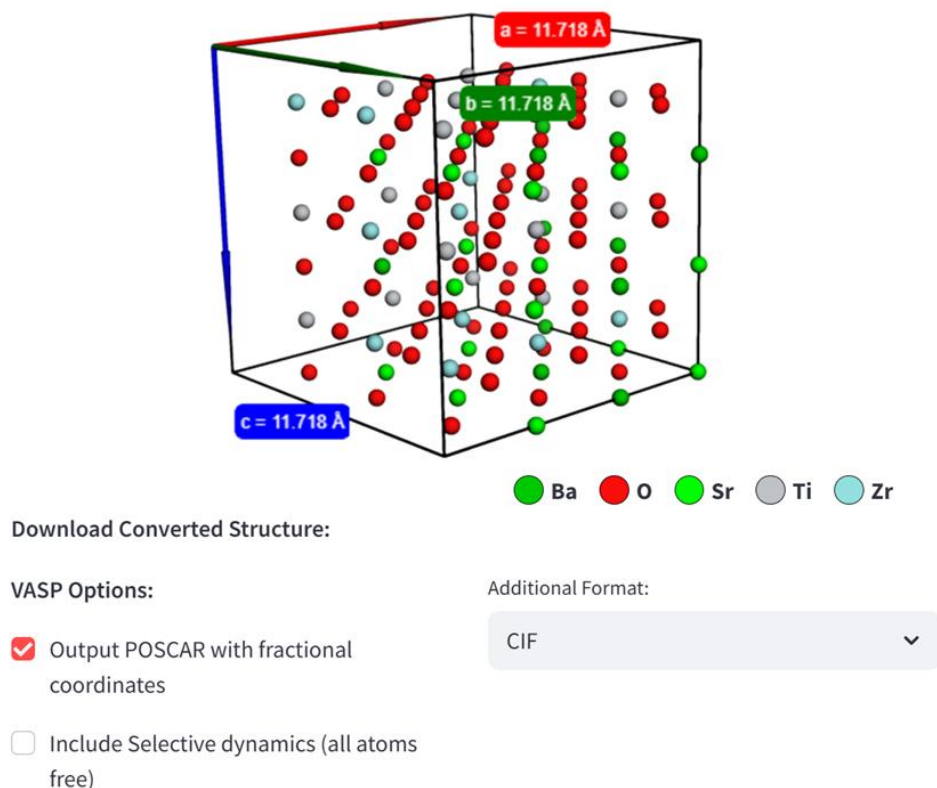


Fig. 7: Illustration of the best SQS for the $(\text{Sr}_{0.59}\text{Ba}_{0.41})(\text{Ti}_{0.59}\text{Zr}_{0.41})\text{O}_3$ uploaded into the *SimplySQS* interface, showing options to download the structure in different formats.

The platform also supports monitoring of the mcsqs search process. By uploading the main log file (*mcsqs.log*) or the progress files generated by the all-in-one batch script (*mcsqs_progress.csv* for single runs and *mcsqs_parallel_progress.csv* for parallel executions), users can visualize the convergence of the objective function with respect to search steps or elapsed time (**Fig. 8**) and immediately see which run yielded the best and which yielded the worst objective function. Moreover, *SimplySQS* can process the *bestcorr.out* file, which reports the cluster correlations of the final SQS. These are compared against the target values of an ideal random alloy, enabling users to evaluate how closely the generated structure reproduces the desired random alloy.

Parallel MCSQS Optimization Progress

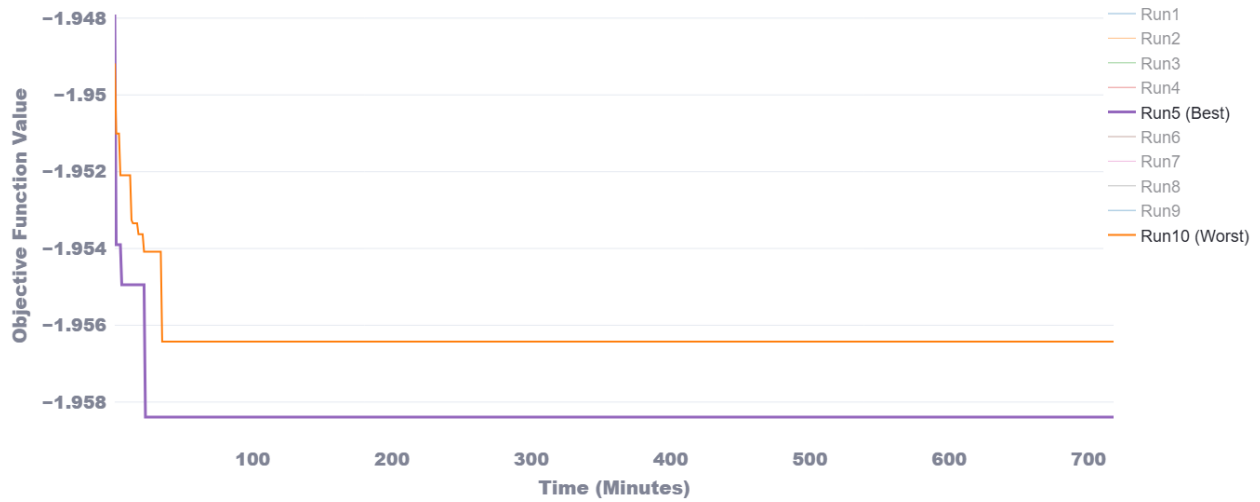


Fig. 8: Time-dependent convergence of the SQS search for $(\text{Sr}_{0.59}\text{Ba}_{0.41})(\text{Ti}_{0.59}\text{Zr}_{0.41})\text{O}_3$ using the ATAT mcsqs parallel run, highlighting the runs with the best and worst objective functions.

4. Example – $\text{Pb}_{1-x}\text{Sr}_x\text{TiO}_3$ Lattice Parameters with MACE $r^2\text{SCAN}$ Universal MLIP

4.1 SQS Settings

To illustrate the effectiveness of *SimplySQS*, a series of SQSs was generated for the perovskite alloy system $\text{Pb}_{1-x}\text{Sr}_x\text{TiO}_3$ (PSTO, including PbTiO_3 (PTO) and SrTiO_3 (STO)) covering the full composition range ($0 < x < 1$). Each SQS comprised a 320-atom supercell, constructed as a $4 \times 4 \times 4$ expansion of the initial PbTiO_3 unit cell. The SQS generation was performed using a single all-in-one batch script created via the binary concentration sweep mode of *SimplySQS*. The used script and generated SQS files are available on GitHub at: github.com/bracerino/atat-sqs-gui/tree/main/Example. Each concentration was searched for 4 hours, with five independent parallel runs per composition and 5 parallelly running concentrations. A pair-cluster cut-off distance of 1.5 Å and a triplet-cluster cut-off of 1.2 Å (in normalized unit cell units) were used. Notably, for all concentrations, the objective function exhibited no change in the last three hours of the search.

4.2 Geometry Optimization Settings

For each concentration, all five generated SQSs were employed for geometry optimization to obtain statistical changes in lattice parameters. Optimizations were performed with the foundation MACE MATPES-r2SCAN-0 [24] universal MLIP. This foundation model was used because it was trained on DFT data computed with the r^2 SCAN exchange–correlation functional [31], which generally yields more accurate lattice parameters and energetics than conventional GGA-PBE [32]. Here, its predictions are directly compared with reference experimental lattice parameters of the PSTO series, thus also assessing the model’s quantitative accuracy for this system.

In each geometry optimization, both atomic positions and lattice vectors were relaxed while fixing cell angles and enforcing $a = b$ to avoid minor unintended deviations between a and b arising from finite supercell size effects. The starting lattice parameters for each concentration were taken from the previously optimized lower-Pb concentration, i.e., the sequence began with SrTiO_3 (initialized at the experimental cubic lattice constant of 3.905 Å [33]), and each subsequent composition used the optimized lattice of the prior step as its initial guess. A strict force convergence criterion of $0.002 \text{ eV} \cdot \text{\AA}^{-1}$ was applied, and the BFGS algorithm was used for the optimization in ASE. All optimizations converged in no more than 300 steps.

4.3 Results

The optimized lattice parameters show a clear cubic-to-tetragonal phase transformation around $x \approx 0.5$, consistent with experimental observations (**Fig. 9**) [34]. For $x < 0.5$, all structures remained cubic ($a = c$), and the deviation between simulated and experimental lattice parameters was under 0.3 %, indicating high precision of the MACE r^2 SCAN potential in this region. For $x > 0.5$, a tetragonal distortion emerged: the simulated a parameter decreases (and becomes lower than experiment, max deviation of 0.9 %) whereas the c parameter increases (and becomes slightly higher than experiment, maximum deviation of 3.4 %) with increasing Pb content. We attribute this opposite trend to the increasing contribution of Pb–O bonding, which may not be fully captured by the universal potential’s training DFT set.

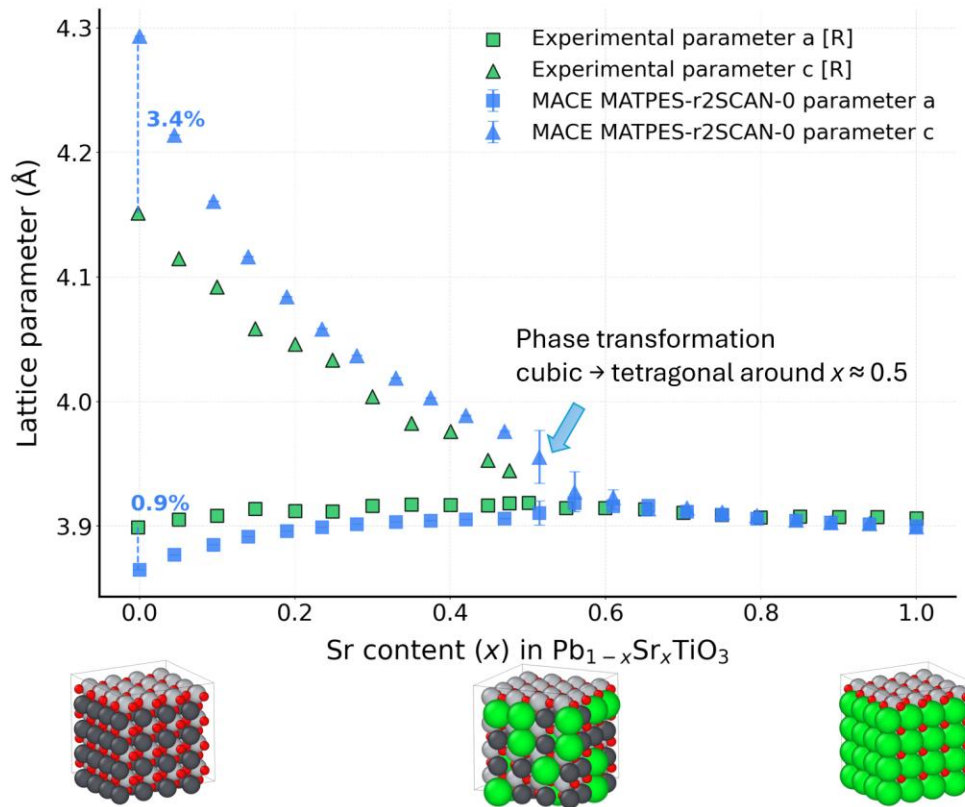


Fig. 9: The lattice parameters of $\text{Pb}_{1-x}\text{Sr}_x\text{TiO}_3$ as a function of Sr content calculated by universal MLIP MACE MATPES-r2SCAN from the generated SQSs, compared to the experimental reference ([R] = [34]).

The standard deviation across the five SQSs at each composition was negligible except near the transition region ($x \approx 0.5$), where $\sigma(a) = 0.01597 \text{ \AA}$ and $\sigma(c) = 0.02114 \text{ \AA}$, before decreasing again at other concentrations. Slight residual differences between calculated a and c in the cubic region were observed and are attributed to the finite-size of the 320-atom supercell, but these are minor (less than $0.001 \text{ \AA}$) and do not alter the compositional trends.

5. Impact on User Adoption and Education

The presented interactive interface for preparing ATAT mcsqs input files significantly lowers the entry barrier for generating SQS with ATAT mcsqs, particularly for users with limited programming or command-line experience. Traditionally, the use of ATAT has required manual preparation of input files, familiarity with file formats, and often trial-and-error troubleshooting, all of which can discourage new users or students entering the field. By automating these steps through a graphical and intuitive interface, generating the necessary input files for SQS becomes far more straightforward.

From an adoption perspective, this simplification is expected to:

- Broaden the user base of ATAT by making it accessible to researchers outside computational materials science, such as experimentalists or educators.
- Reduce the dependency on community support forums by minimizing errors caused by file formatting or parameter misinterpretation in the input files.
- Make it easier to continue with simulations by allowing direct conversion of the best SQS into widely used formats, such as POSCAR (for VASP) or LAMMPS input files. This lets users move directly from SQS generation to DFT or MD simulations without extra file preparation.
- Improve the workflow for experienced users.

From an educational perspective, the interface also provides a valuable teaching tool. Instructors can integrate it into classroom demonstrations or computational labs to introduce students to the concept of random alloys and SQS generation without requiring prior coding skills. The guided workflows and integrated visualizations of structural data, correlation convergence, and radial distribution functions help students build intuition about the underlying theory.

Moreover, since *SimplySQS* is available online, it eliminates the need for local compilation and can be accessed from any device with an internet connection and a web browser.

6. Conclusions

In this work, we have introduced *SimplySQS* (simplysqs.com), an online, interactive browser-based application designed to simplify the generation and analysis of special quasirandom structures (SQS) using ATAT mcsqs module. The interface guides users through the entire workflow, from structure import (either directly with structure files such as CIF or POSCAR or via retrieval from MP, AFLOW, and COD databases) and supercell setup to automated preparation of input files and execution scripts. It further enables direct analysis of ATAT output files, including visualization of objective function convergence over time, correlation values, calculation of partial radial distribution functions, and offers direct conversion of the best SQS into common simulation formats. An additional feature is the all-in-one batch script, which automates the mcsqs search, monitors its progress, and automatically exports the best SQS in POSCAR format upon completion. Since all parameters are contained in a single, shareable file, this script enhances reproducibility and allows other researchers to reliably repeat the SQS search.

By integrating these steps into a single platform, *SimplySQS* improves the user-friendliness of ATAT mcsqs and minimizes errors caused by manual file preparation. For these reasons, the platform lowers the entry barrier for students and researchers with limited coding experience, while also providing workflow improvement and reproducibility benefits for more advanced users. Moreover, being a fully online browser-based application, *SimplySQS* requires no compilation, making it accessible from any device with an internet connection.

The usefulness of *SimplySQS* was illustrated on the perovskite series $\text{Pb}_{1-x}\text{Sr}_x\text{TiO}_3$ (PSTO, including PbTiO_3 and SrTiO_3), where a set of SQSs across the full composition range was generated using an all-in-one batch script. The resulting structures were subsequently optimized with the MACE MATPES- r^2 SCAN-0 universal MLIP to determine the evolution of lattice parameters with increasing Sr content. The optimized PSTO structures accurately reproduced the experimentally observed cubic-to-tetragonal phase transformation near $x \approx 0.5$. In the cubic concentration region ($x > 0.5$), the simulated lattice parameters showed excellent agreement with experimental values, within a maximum of 0.3 % deviation. In the tetragonal region ($x < 0.5$), the lattice parameter a decreased slightly more (maximum deviation of 0.9 %), and parameter c increased more (maximum deviation of 3.4 %) than the experimental values, suggesting that the r^2 SCAN-based universal MLIP may have insufficient underlying DFT training data for Pb-rich compositions.

7. Acknowledgments

This work was supported by the Grant Agency of the Czech Technical University in Prague [grant No. SGS24/121/OHK2/3T/12].

8. References

- [1] A. Zunger, S.-H. Wei, L. G. Ferreira, and J. E. Bernard, Special quasirandom structures, *Phys Rev Lett* **65**, 353 (1990).
- [2] D. Vizoso and C. Deo, Determination of vacancy formation energies in binary UZr alloys using special quasirandom structure methods, *Front Mater* **8**, 692660 (2021).
- [3] J. Hu, X. Zhang, C. Hu, J. Sun, X. Wang, H.-Q. Lin, and G. Li, Correlated flat bands and quantum spin liquid state in a cluster Mott insulator, *Commun Phys* **6**, 172 (2023).

- [4] T. Wang, W. Li, C. Ni, and A. Janotti, Band gap and band offset of Ga_2O_3 and $(\text{Al}_x\text{Ga}_{1-x})_2\text{O}_3$ alloys, *Phys Rev Appl* **10**, 011003 (2018).
- [5] A. M. Ritzmann, A. B. Muñoz-García, M. Pavone, J. A. Keith, and E. A. Carter, Ab initio DFT+ U analysis of oxygen vacancy formation and migration in $\text{La}_{1-x}\text{Sr}_x\text{FeO}_{3-\delta}$ ($x=0, 0.25, 0.50$), *Chemistry of Materials* **25**, 3011 (2013).
- [6] W. Hu, M. Pan, Z. Huang, H. Yang, H. Wu, L. Pan, Z. Cao, and Y. Zhao, The Stability and Elasticity in Ta-Ti-V Medium-Entropy Alloys Using First-Principles Calculations, *Front Mater* **8**, 651435 (2021).
- [7] K. Li, X. Hu, R. Guo, W. Jiang, L. Zeng, L. Li, P. Yu, K. Wang, C. Zhang, and S. Guo, Superconductivity in the medium-entropy alloy TiVNbTa with a body-centered cubic structure, *The Journal of Physical Chemistry C* **127**, 16211 (2023).
- [8] J. Sun, S. Mukhopadhyay, A. Subedi, T. Siegrist, and D. J. Singh, Transport properties of cubic crystalline $\text{Ge}_2\text{Sb}_2\text{Te}_5$: A potential low-temperature thermoelectric material, *Appl Phys Lett* **106**, (2015).
- [9] J. Han, J. Park, M. Kim, S. Lee, J. M. Heo, Y. H. Jin, Y. Chae, J. Han, J. Wang, and S. Seok, Ultrahigh Conductive MXene Films for Broadband Electromagnetic Interference Shielding, *Advanced Materials* 2502443 (2025).
- [10] P. P. O. Borges, R. O. Ritchie, and M. Asta, Chemical trends favoring interstitial cluster formation in bcc high-entropy alloys from first-principles calculations, *Acta Mater* 121091 (2025).
- [11] A. 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, Efficient stochastic generation of special quasirandom structures, *Calphad* **42**, 13 (2013).
- [12] A. Van De Walle, M. Asta, and G. Ceder, The alloy theoretic automated toolkit: A user guide, *Calphad* **26**, 539 (2002).
- [13] M. Ångqvist, W. A. Muñoz, J. M. Rahm, E. Fransson, C. Durniak, P. Rozyczko, T. H. Rod, and P. Erhart, ICET—a Python library for constructing and sampling alloy cluster expansions, *Adv Theory Simul* **2**, 1900015 (2019).
- [14] D. Gehringer, M. Friák, and D. Holec, Models of configurationally-complex alloys made simple, *Comput Phys Commun* **286**, 108664 (2023).
- [15] S. P. Ong, W. D. Richards, A. Jain, G. Hautier, M. Kocher, S. Cholia, D. Gunter, V. L. Chevrier, K. A. Persson, and G. Ceder, Python Materials Genomics (pymatgen): A

- robust, open-source python library for materials analysis, *Comput Mater Sci* **68**, 314 (2013).
- [16] A. H. Larsen, J. J. Mortensen, J. Blomqvist, I. E. Castelli, R. Christensen, M. Dułak, J. Friis, M. N. Groves, B. Hammer, and C. Hargus, The atomic simulation environment—a Python library for working with atoms, *Journal of Physics: Condensed Matter* **29**, 273002 (2017).
- [17] A. Jain, S. P. Ong, G. Hautier, W. Chen, W. D. Richards, S. Dacek, S. Cholia, D. Gunter, D. Skinner, and G. Ceder, Commentary: The Materials Project: A materials genome approach to accelerating materials innovation, *APL Mater* **1**, (2013).
- [18] S. Curtarolo, W. Setyawan, G. L. W. Hart, M. Jahnatek, R. V. Chepulskii, R. H. Taylor, S. Wang, J. Xue, K. Yang, and O. Levy, AFLOW: An automatic framework for high-throughput materials discovery, *Comput Mater Sci* **58**, 218 (2012).
- [19] M. Esters, C. Oses, S. Divilov, H. Eckert, R. Friedrich, D. Hicks, M. J. Mehl, F. Rose, A. Smolyanyuk, and A. Calzolari, aflow. org: A web ecosystem of databases, software and tools, *Comput Mater Sci* **216**, 111808 (2023).
- [20] S. Gražulis, D. Chateigner, R. T. Downs, A. F. T. Yokochi, M. Quirós, L. Lutterotti, E. Manakova, J. Butkus, P. Moeck, and A. Le Bail, Crystallography Open Database—an open-access collection of crystal structures, *Applied Crystallography* **42**, 726 (2009).
- [21] J. Hafner, Ab-initio simulations of materials using VASP: Density-functional theory and beyond, *J Comput Chem* **29**, 2044 (2008).
- [22] G. Kresse and J. Furthmüller, Efficient iterative schemes for ab initio total-energy calculations using a plane-wave basis set, *Phys Rev B* **54**, 11169 (1996).
- [23] A. P. Thompson, H. M. Aktulga, R. Berger, D. S. Bolintineanu, W. M. Brown, P. S. Crozier, P. J. In't Veld, A. Kohlmeyer, S. G. Moore, and T. D. Nguyen, LAMMPS—a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales, *Comput Phys Commun* **271**, 108171 (2022).
- [24] I. Batatia, P. Benner, Y. Chiang, A. M. Elena, D. P. Kovács, J. Riebesell, X. R. Advincula, M. Asta, M. Avaylon, and W. J. Baldwin, A foundation model for atomistic materials chemistry, *ArXiv Preprint ArXiv:2401.00096* (2023).
- [25] L. Ward, A. Dunn, A. Faghaninia, N. E. R. Zimmermann, S. Bajaj, Q. Wang, J. Montoya, J. Chen, K. Bystrom, and M. Dylla, Matminer: An open source toolkit for materials data mining, *Comput Mater Sci* **152**, 60 (2018).

- [26] C. R. Harris, K. J. Millman, S. J. Van Der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, and N. J. Smith, Array programming with NumPy, *Nature* **585**, 357 (2020).
- [27] W. McKinney, pandas: a foundational Python library for data analysis and statistics, *Python for High Performance and Scientific Computing* **14**, 1 (2011).
- [28] N. Rego and D. Koes, 3Dmol. js: molecular visualization with WebGL, *Bioinformatics* **31**, 1322 (2015).
- [29] J. D. Hunter, Matplotlib: A 2D graphics environment, *Comput Sci Eng* **9**, 90 (2007).
- [30] M. Lebeda, J. Drahokoupil, P. Veřtát, Š. Svoboda, V. Smola, U. Ahmed, and P. Vlčák, XRDlicious: an interactive web-based platform for online calculation of diffraction patterns and radial distribution functions from crystal structures, *Applied Crystallography* **58**, (2025).
- [31] J. W. Furness, A. D. Kaplan, J. Ning, J. P. Perdew, and J. Sun, Accurate and numerically efficient r2SCAN meta-generalized gradient approximation, *J Phys Chem Lett* **11**, 8208 (2020).
- [32] J. P. Perdew, K. Burke, and M. Ernzerhof, Generalized gradient approximation made simple, *Phys Rev Lett* **77**, 3865 (1996).
- [33] V. M. Longo, M. das G. S. Costa, A. Z. Simoes, I. L. V. Rosa, C. O. P. Santos, J. Andres, E. Longo, and J. A. Varela, On the photoluminescence behavior of samarium-doped strontium titanate nanostructures under UV light. A structural and electronic understanding, *Physical Chemistry Chemical Physics* **12**, 7566 (2010).
- [34] X. Xing, J. Chen, J. Deng, and G. Liu, Solid solution $\text{Pb}_{1-x}\text{Sr}_x\text{TiO}_3$ and its thermal expansion, *J Alloys Compd* **360**, 286 (2003).