# IRA: A shape matching approach for recognition and comparison of generic atomic patterns

Miha Gunde, Nicolas Salles, Anne Hémeryck, Layla Martin-Samos

# IRA: A shape matching approach for recognition and comparison of generic atomic patterns

Miha Gunde,[*,†,‡] Nicolas Salles,[‡] Anne Hémeryck,[†] and Layla Martin-Samos[*,‡]

†*LAAS-CNRS, Université de Toulouse, CNRS, 7 avenue du Colonel Roche, 31031 Toulouse, France*

‡*CNR-IOM, Democritos National Simulation Center, Istituto Officina dei Materiali, c/o SISSA, via Bonomea 265, IT-34136 Trieste, Italy*

E-mail: miha.gunde@gmail.com; marsamos@iom.cnr.it

**Abstract**

We propose a versatile, parameter-less approach for solving the shape matching problem, specifically in the context of atomic structures when atomic assignments are not known a priori. The algorithm Iteratively suggests Rotated atom-centered reference frames and Assignments (Iterative Rotations and Assignments, IRA). The frame for which a permutationally invariant set-set distance, namely the Hausdorff distance, returns minimal value is chosen as the solution of the matching problem. IRA is able to find rigid rotations, reflections, translations, and permutations between structures with different numbers of atoms, for any atomic arrangement and pattern, periodic or not. When distortions are present between the structures, optimal rotation and translation are found by further applying a standard Singular Value Decomposition-based method. To compute the atomic assignments under the one-to-one assignment constraint, we develop our own algorithm, Constrained Shortest Distance Assignments (CShDA). The overall approach is extensively tested on several structures, including distorted structural fragments. Efficiency of the proposed

algorithm is shown as a benchmark comparison against two other shape matching algorithms. We discuss the use of our approach for the identification and comparison of structures and structural fragments through two examples: a replica exchange trajectory of a cyanine molecule, in which we show how our approach could aid the exploration of relevant collective coordinates for clustering the data; and an $SiO_2$ amorphous model, in which we compute distortion scores and compare them with a classical strain-based potential. The source code and benchmark data are available at `https://github.com/mammasmias/IterativeRotationsAssignments`.

# 1 Introduction

Shape matching is the ability to find the transformation that best matches a set of points to another set of points. In the context of atomic structures, shape matching techniques are exploited in a broad variety of applications, ranging from computer-aided drug discovery,[1–3] to global structure optimization approaches, such as genetic-algorithm[4–6] and Basin-hopping Monte-Carlo.[7,8]

Formally, two sets of vector elements are considered congruent or equivalent if they are related by a transformation that preserves distances, i.e. isometric transformation. Such transformations are rigid translations, rigid rotations, reflections, and permutations of indistinguishable vectors. The isometric transformation that fulfills the congruence relation between two structures gives a solution to the shape matching problem. This problem can be addressed from different perspectives. In the following, it is stated as an optimization problem.

If sets $A$ and $B$ represent two atomic structures, e.g. two sets of atomic positions, the congruence relation between them can be written as:

$$P_B B = \mathbf{R}A + \mathbf{t} \tag{1}$$

where $P_B$ is a permutation matrix of atomic indices, $\mathbf{R}$ is a transformation corresponding to either rigid rotation, reflection, or combination of both, and $\mathbf{t}$ is a translation vector.

The problem of finding $P_B$, $\mathbf{R}$, and $\mathbf{t}$ that best matches one structure to another can be reformulated as an optimization problem:

$$\arg\min_{\mathbf{R},\mathbf{t}} \big\{ D(\mathbf{R}A + \mathbf{t}, B) \big\}, \tag{2}$$

in which $D$ is a general distance function between two sets, that is i) variant under $\mathbf{R}$ and $\mathbf{t}$, ii) invariant under permutation $P_B$, and iii) returns value 0 when $\mathbf{R}$ and $\mathbf{t}$ are such that Eq. (1) is satisfied, i.e. when the best match is found. It is important to highlight that $D$ does not rely on an internal structural description (encoding), but rather it directly compares the "raw" state of the two structures, since $\mathbf{R}$ and $\mathbf{t}$ depend on their relative reference frames. When distortions and/or deformations are present, the transformation that minimizes Eq. (2), does not strictly return a 0 distance, but some minimum value. In that case, the relation between $A$ and $B$ is called a near-congruence, and the isometric transformation $\mathbf{R}$ and $\mathbf{t}$ is formally referred to as a near-isometry. This minimum distance value provides a measure of the quality of the congruence, i.e. a measure of the similarity between the structures. Beyond near-isometry, it is not straightforward to assign a meaning to the distance and transformation that is returned from the optimization of Eq. (2). Therefore, a similarity measure obtained from shape matching cannot be thought of only and strictly as a generic similarity metric for arbitrary structures.

A widely used set-set distance function, in particular in computational (bio)-chemistry, is Root-Mean-Square-Deviation ($RMSD$), which is usually defined as:

$$RMSD(A, B) = \sqrt{\frac{1}{N} \sum_{i}^{N} d(a_i, b_i)^2} \tag{3}$$

where $N$ is the number of points, and $d(a_i, b_i)$ denote an Euclidean distance between points

$a_i \in A$ and $b_i \in B$. It can immediately be noted that Eq. (3) depends on the ordering of points $i$ in the two sets, its value depends on the permutation $P_B$. In other words, $RMSD$ depends on atomic assignments, i.e. which atom from one structure is assigned to which atom from the other structure. In addition, if we cast the matching problem as finding a global minimum in the phase space of rotations, reflections, and permutations (neglecting for a moment the translations), the definition of $RMSD$ in Eq. (3) does not guarantee the existence of a single connected path from an arbitrary point to the global minimum. For an example see Fig. 1: a change in the permutation of atoms can lead to a discontinuous

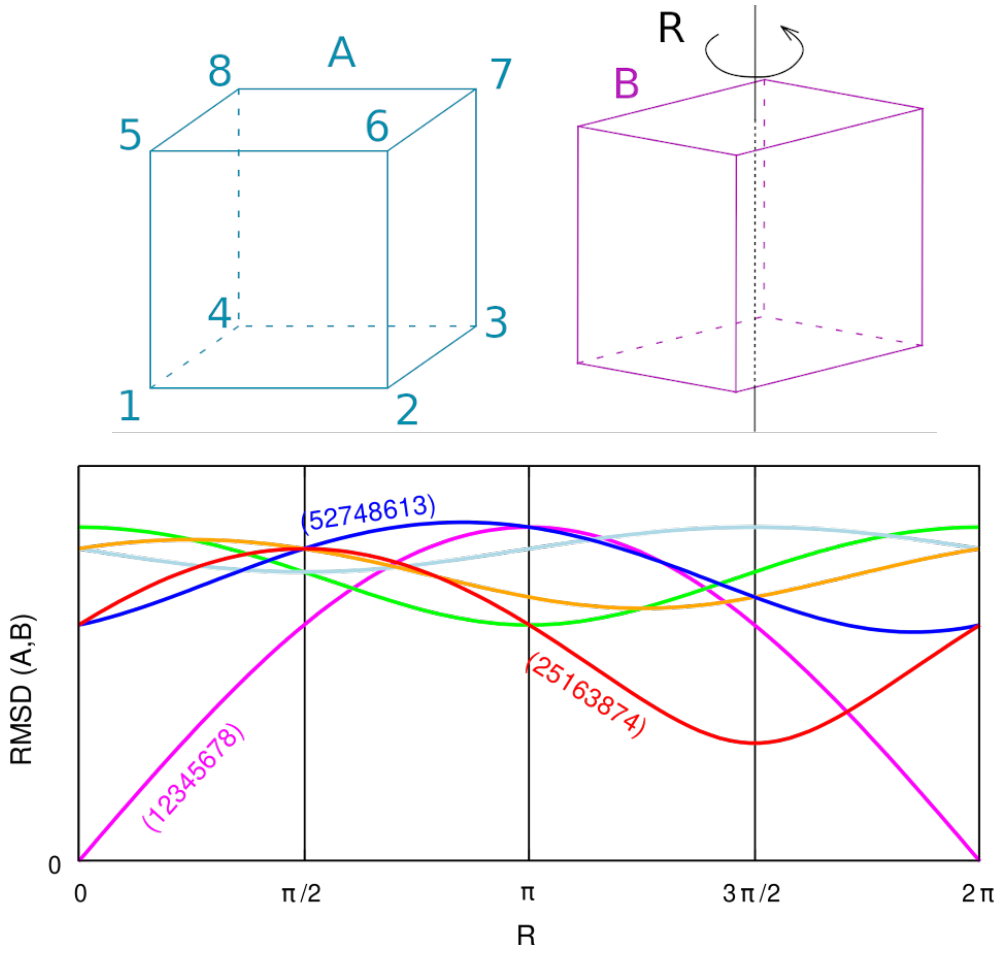



Figure 1: $RMSD$ as a function of rotations $\mathbf{R}$ and permutations between two identical cubes $A$ and $B$, shown above the plot. Cube $A$ is fixed while $B$ is rotated around the $z$-axis only, for simplicity. Each color in the plot represents a different permutation of the rotated cube, some of them are explicitly labelled. Not all permutations are pictured, as there are in total $N_P = 8! = 40320$ possibilities.

jump in $RMSD$ value. For this reason $RMSD$ is not directly suitable for a shape matching problem as formulated by Eq. (2). In Refs. 9 and 10, authors suggested a re-definition of $RMSD$ based on shortest distances, as an attempt to obtain a permutationally invariant quantity. Ref. 11 noted that $RMSD$ draws a picture of similarity in an averaging fashion, and proposed an additional criterion for similarity based on the maximal deviation for any atom of $A$ with respect to that atom in $B$. Despite the Eq. (2) providing stringent criteria for choosing distance functions, in practice there is always some arbitrariness in the choice.

Approaches for finding rotations when the atomic assignments are known and the two structures have the same number of atoms are well established. Generally they rely on symmetrization of a special matrix, or minimization of a cost function. [1] Examples of the two ideas include Lagrange multiplier method,[12] matrix symmetrization,[13,14] decomposition of a matrix into orthonormal and positive semidefinite matrices,[15] Singular Value Decomposition (SVD),[16–18] and quaternion eigensystem problem[19–22] (a review of quaternions can be found in Ref. 23, and more recently in Refs. 24,25). Usually the cost function minimized is the $RMSD$ distance.

Finding the assignments between points of two structures is usually called the Linear Assignment Problem (LAP). The most widely used general-purpose LAP algorithm is the Hungarian algorithm,[26,27] however others exist, see for example Ref. 28. Briefly, it is a mapping from indices of one set to indices of another set, which minimizes a given cost given in the form of a matrix. When applied to atomic structures, an atom represents an index of a point, and the atomic structure represents a set of points. Solving this problem might seem simple, but without the knowledge of any intrinsic relation between the atoms, the complexity increases very quickly as the total number of possible permutations $N_P$ of indistinguishable vectors (atoms) in a structure grows as $N_P = \prod_{k=1}^{m} n_k!$, where $m$ is the total number of different atomic types present, and $n_k$ is the number of atoms of atomic type $k$.

[1] Symmetrization or minimization algorithms for rotations require square matrices. As such, the rotations can not be found if the structures have a different number of atoms, without a pre-processing.

One can also quickly realize that the optimal assignment or mapping of points depends on the relative rotation of the two structures. However, algorithms for finding rotations alone are not able to switch permutations by themselves, while algorithms for finding atomic assignments provide the permutation order that minimizes a distance function at fixed rotation, but are not able to suggest rotations that would further minimize it.

To try to overcome such limitations, some strategies have been proposed and are in use in different communities. For instance, the algorithm Iterative Closest Point (ICP)[29] exploits the idea of self-consistent iteration, where each step combines an assignment procedure and consecutive rotation procedure, until a solution is found. However, ICP might remain trapped in local minima of the transformation space.[30] Local minima are a consequence of structural symmetries, see also Fig. 1. Authors in Ref. 31 suggested an algorithm in which the space of possible rotations and reflections is discretized into a uniform grid of points. For each grid-point $\mathbf{R}$ the optimal atomic assignment $P_B$ is obtained as the optimal assignment of an inter-structure distance matrix with the Hungarian algorithm,[26] which is then used to minimize rotations with SVD.[17] Such strategy is however difficult to optimize, as the number of grid points is not directly related to any property of the system. A slightly different approach has been proposed in Ref. 11, with an atomic-centered grid of approximate rotations, in which the farthest atoms from the center are selected as the basis for aligning the reference frames and to find approximate rotations. The atomic assignments are obtained via finding optimal assignment of the inter-structure distance matrix with the Hungarian algorithm. The authors in Ref. 32 propose an approach for the alignment of molecules based on ideas from image recognition, which relies on filtering methods to obtain atomic assignments. Optimal rotations are later resolved by applying an SVD minimization. Alternatively, finding a rough equivalent reference frame (or Eckart frame[33,34]) through, for example principal axes of inertia, might also provide a good-enough rotation for identifying reasonable assignments, see for instance Refs. 35–37. The principal axes idea is however not suitable for isotropic or compact structures, and crystalline or bulk environments, since

the principal axes might be ambiguously defined due to the symmetry of the structures. Moreover, the computation of principal axes of inertia requires the knowledge of associated weights, *i.e.* atomic masses. A successful Monte-Carlo-based decision scheme for finding the global minimum of $RMSD$[38] has also been reported.

In this work, we present an alternative and versatile, parameter-less approach that solves the general shape matching problem by finding isometries and near-isometries between two (sub-)structures when the assignment is not known a priori, named Iterative Rotations and Assignments (IRA). Isometries and near isometries can be found even in the case of structures with different numbers of atoms and belonging to some periodic lattice. The proposed algorithm iteratively suggests rotated atom-centered reference frames of one structure, to find an approximate rotation in which the matching to the other structure is best. This best match provides the one-to-one atomic assignment, thus the permutation $P_B$. When structural distortions are present between the structures, the optimal rotation $\mathbf{R}$, is later found via SVD.[18] To avoid the ambiguity in the mitigation of improper rotations in SVD and to enable the matching of mirror structures, reflection symmetries are taken into account by also proposing a reflected configuration at each step of the iteration. To assess the matching, our approach exploits a truly permutationally invariant set-set distance function, namely the Hausdorff distance.[39] This distance measure is often exploited in the computer vision community, where the shape matching problem is referred to as *point set registry*. In our implementation, the Hausdorff distance is evaluated after imposing the one-to-one atomic assignment.

We first test the reliability of our proposed matching approach (Sec. 3.1), by applying random rigid transformations and permutations to a range of structures, and then applying the shape matching algorithm to re-find them. Later, the performances are compared to two other algorithms, namely ArbAlign[37] and fastoverlap.[10] In all benchmarks, IRA performs significantly better. To test behavior in near-congruent structures, we apply the algorithm to two short finite-temperature Monte Carlo trajectories (Sec. 3.2). We next apply it to

match and analyse the distortion of cyanine molecule fragments (Sec. 3.3) along a replica-exchange molecular dynamics trajectory from Ref. 40. We also discuss the use of Eq. (2) as a definition of a similarity relation to blindly identify, compare and analyze local structures or fragments. Such sub-structures can be connected or not, and the larger structure to be matched might or not include lattice periodicity.

# 2 Our Approach

Similarly to other matching techniques briefly summarized in the introduction, we address the general matching problem (Eq. (2)) in two parts. The first part iteratively solves the approximate rotation, which makes it possible to compute the correct atomic assignments. The second part uses the atomic assignments to compute the final optimal rotation via standard Singular Value Decomposition (SVD). We develop the approach Iterative Rotations and Assignment (IRA, Sec. 2.1), to obtain the approximate rotation in the first part of our algorithm. To compute the atomic assignments, we develop our own algorithm: Constrained Shortest Distance Assignment (CShDA, Sec. 2.1.1), that solves the Linear Assignment Problem (LAP) under the one-to-one assignment constraint. The flowchart representing the full algorithm is shown on Fig. 2, where the first part of the algorithm is colored in blue, the second part in green, and the final matching solution is colored in red.

## 2.1 Iterative Rotations and Assignment (IRA)

A rigid rotation and translation of a structure by $\mathbf{R}$ and $\mathbf{t}$ is equivalent to rotating and translating its reference frame. As the distance $D$ in Eq. (2) directly compares the "raw" state of the two structures, and $\mathbf{R}$ and $\mathbf{t}$ depend on their relative reference frames, the shape matching problem can be addressed as finding a common approximate reference frame between structures $A$ and $B$. The reference frames that are evaluated in our algorithm are atom centered, with basis vectors chosen as follows.

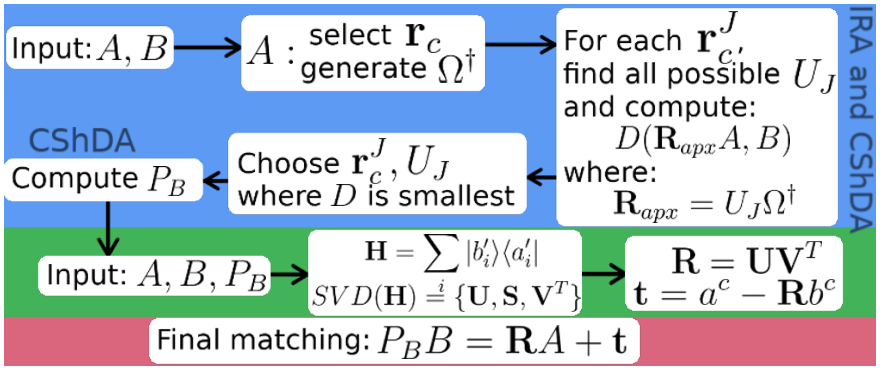



Figure 2: Flowchart of the algorithm. First part of the algorithm colored in blue gives an approximate solution to rotations and translations, and solution to the permutations $P_B$ needed in the second part of the algorithm colored in green, which finds the optimal rotation and translation by utilising the SVD method. Final solution of the matching algorithm is colored in red.

The atom closest to the geometrical center of $A$ is taken as the central atom and reference frame origin of $A$, i.e. all atoms in $A$ are shifted by the former atomic coordinate vector $-\mathbf{r}_c$ (in case of periodic structures, periodic boundary conditions are applied). Two non-colinear atomic coordinate vectors are subsequently chosen and orthonormalized with the standard Gramm-Schmidt procedure such that $\hat{\mathbf{e}}_1$ points to an atom. The last reference-frame basis vector is obtained as vector product of the other two, such obtaining a set of three orthonormal basis vectors $\hat{\mathbf{e}}_1$, $\hat{\mathbf{e}}_2$ and $\hat{\mathbf{e}}_3$. The coordinates of $A$ in the new basis can be obtained as:

$$A_{\{\hat{\mathbf{e}}\}} = \Omega^\dagger(A - \mathbf{r_c}), \tag{4}$$

where $\Omega^\dagger$ is the transformation matrix from original reference frame of $A$ to $A_{\{\hat{\mathbf{e}}\}}$, formed by the vectors $\hat{\mathbf{e}}_i$. To find a similar atom-centered reference frame in $B$, all atoms of the same atomic type as central atom of $A$ are designated candidate central atoms. For each candidate central atom $J$, an ensemble of reference frames, and their mirrors are generated by the same procedure as for $A$. Namely, $\{\hat{\mathbf{e}}_1', \hat{\mathbf{e}}_2', \hat{\mathbf{e}}_3' = \hat{\mathbf{e}}_1' \times \hat{\mathbf{e}}_2'\}$ and their mirror $\{\hat{\mathbf{e}}_1', \hat{\mathbf{e}}_2', \hat{\mathbf{e}}_3' = \hat{\mathbf{e}}_2' \times \hat{\mathbf{e}}_1'\}$. Each candidate central atom $J$ has its atomic vector $\mathbf{r}_c^J$, and an ensemble of

transformation matrices $U_J$, one for each reference frame guess $\{\hat{\mathbf{e}}'\}_J$, such that

$$B_{\{\hat{\mathbf{e}}'\}_J} = U_J^{\dagger}(B - \mathbf{r}_c^J) \tag{5}$$

where $U_J^{\dagger}$ is formed by the vectors $\hat{\mathbf{e}}_i'$.

The LAP (Sec. 2.1.1) is solved for all reference frames and central atoms, and the combination of reference frame and central atom guess $J$ that return the *lowest* set-set distance function $D(A_{\{\hat{\mathbf{e}}\}}, B_{\{\hat{\mathbf{e}}'\}_J})$, defines permutation $P_B$, the approximate rotation matrix $\mathbf{R}_{apx}$, and approximate translation vector $\mathbf{t}_{apx}$:

$$\begin{aligned} \mathbf{R}_{apx} &= U_J\Omega^{\dagger} \\ \mathbf{t}_{apx} &= \mathbf{r}_c^J - \mathbf{R}_{apx}\mathbf{r}_c. \end{aligned} \tag{6}$$

The distance $D$ is evaluated with the help of our CShDA algorithm, and is equal to the Hausdorff distance, see Sec. 2.1.1, and Sec. 2.1.2.

To reduce the number of combinations to be tested in $B$, vectors in $A$ are sorted according to their norm, such that the two atoms taken to generate the basis are as close as possible to the central atom. The largest norm among these two atomic vectors is taken as a cutoff distance, which is multiplied by a factor (1.2 by default) to account for possible distortions, and taken as maximal-norm threshold for possible basis vectors in $B$. The total number of rotations tested $N_R$ thus depends on the compactness (density) and number of nearest neighbors, and goes as $N_R = n_C(n_C - 1)$, where $n_C$ are the number of neighbors. For a highly compact crystal structure the number of atoms $n_C$ in this sphere can be large (e.g. 15-20), while for molecular structures it is usually much lower (e.g. 5-8). The overall order of the procedure is therefore well below $\mathcal{O}(N^3)$, where $N$ is the total number of atoms (see also the Discussion section). In addition, contrary to the uniform grid proposed in Ref. 31, our approach does not require blind and massive checks on the number of grid points and

their completeness in parsing the rotation space/manifold.

When $A$ and $B$ contain the same number of points/atoms, the search over possible central atoms of $B$ is not required. In that case $\mathbf{r}_c$ and $\mathbf{r}_c^J$ is replaced by the coordinates of the geometrical centers of $A$ and $B$, respectively. If any other point that is common to both $A$ and $B$ is known, that particular point can also be used as the center.

If $A$ and (a subset of-)$B$ are exactly congruent, i.e. no atomic deformations, the algorithm would already return the $P_B$, $\mathbf{R}$, and $\mathbf{t}$ that exactly minimize Eq. (2), as $D(\mathbf{R}_{apx}A+\mathbf{t}_{apx}, B) = 0$.

### 2.1.1 LAP algorithm: CShDA

For the shape matching algorithm presented here, we develop our own atomic assignment algorithm based on shortest distances $d_{ij}$, the Constrained Shortest Distance Assignment (CShDA). It gives an assignment, or mapping between two atoms $i \rightarrow j$, such that each atom gets a minimum possible cost, under the constraint that each atom can only have one and only one match, so-called one-to-one assignment. The idea is that the distances from an atom $a_i \in A$ to all atoms $b \in B$ are used as a cost for computing the assignment of atom $a_i$, such that shortest distances are prioritized for each atom $a_i$ locally. To showcase, an atom $a_i$ gets assigned an atom $b_j$ with the shortest distance $d(a_i, b_j)$ among all atoms $b$. However, if during the algorithm an atom $a_i \in A$ is assigned an atom $b_j \in B$ with some distance $d(a_i, b_j)$, and another atom $a_{i'} \in A$ gets assigned the same atom $b_j \in B$ with a distance $d(a_{i'}, b_j) < d(a_i, b_j)$, the atom $a_{i'}$ will be prioritized for this $b_j$, and the atom $a_i$ gets assigned a different atom. Symbolically, CShDA iteratively assigns a single atom $a_i \in A$ to a single atom $b_j \in B$ following:

$$a_i \rightarrow b_j \quad | \quad \min_{b_j \in B} d(a_i, b_j) \quad \forall a_i \in A \tag{7}$$

with the constraint that $b_j$ has not yet been assigned with a distance lower than $d(a_i, b_j)$, where $d$ is the Euclidean distance between the points. When applied to a general set of points, this kind of local assignment is sometimes referred to as bottleneck LAP.[41]

With respect to one of the most widely known general-purpose LAP solvers, the Hungarian algorithm,[26,27] there are two main differences with our proposed CShDA algorithm, explained in the following.

Firstly, the criteria for the assignment of two atoms differ. The Hungarian algorithm assigns indices such that the total sum of the cost is minimized, where the cost of assignment is the distance between two points. In CShDA, each assignment cost is minimized separately, under the one-to-one constraint, where the assignment cost is the distance between points. The CShDA algorithm tends to concentrate the maximum deviations on a small number of atoms, contrary to the Hungarian algorithm that favours smaller deviations, but spread over several atoms. Practically, it means that the Hungarian prefers globally "distorted" solutions over rigid single mismatches, see Fig. 3.

The second difference is that the Hungarian algorithm requires two structures to have equal number of atoms, as the cost of assignment is computed from an all-to-all distance matrix, which needs to be square. While it is true that any square matrix can be made to be non-square by the addition of ghost rows or columns at specific indices, this is not trivial since it is not known a priori which should these indices be. Our proposed CShDA algorithm does not have such a constraint. The only requirement for CShDA is that the number of atoms $n_A$ in structure $A$ is $n_A \leq n_B$, where $n_B$ is the number of atoms in structure $B$ (this point is also addressed in the Discussion). In the case when the two sets contain a different number of atoms, there will be some points of $B$ that are left unassigned. We enforce that the permutation $P_B$ of set $B$ will in this case be such that the points of $A$ will be assigned to the first $n_A$ points of $P_B B$. The unassigned points of $B$ will be permuted to the end of the set.

### 2.1.2 The set-set distance function

A distance function that fulfils the requirements for solving the shape matching problem as formulated by Eq. (2) is the Hausdorff distance function.

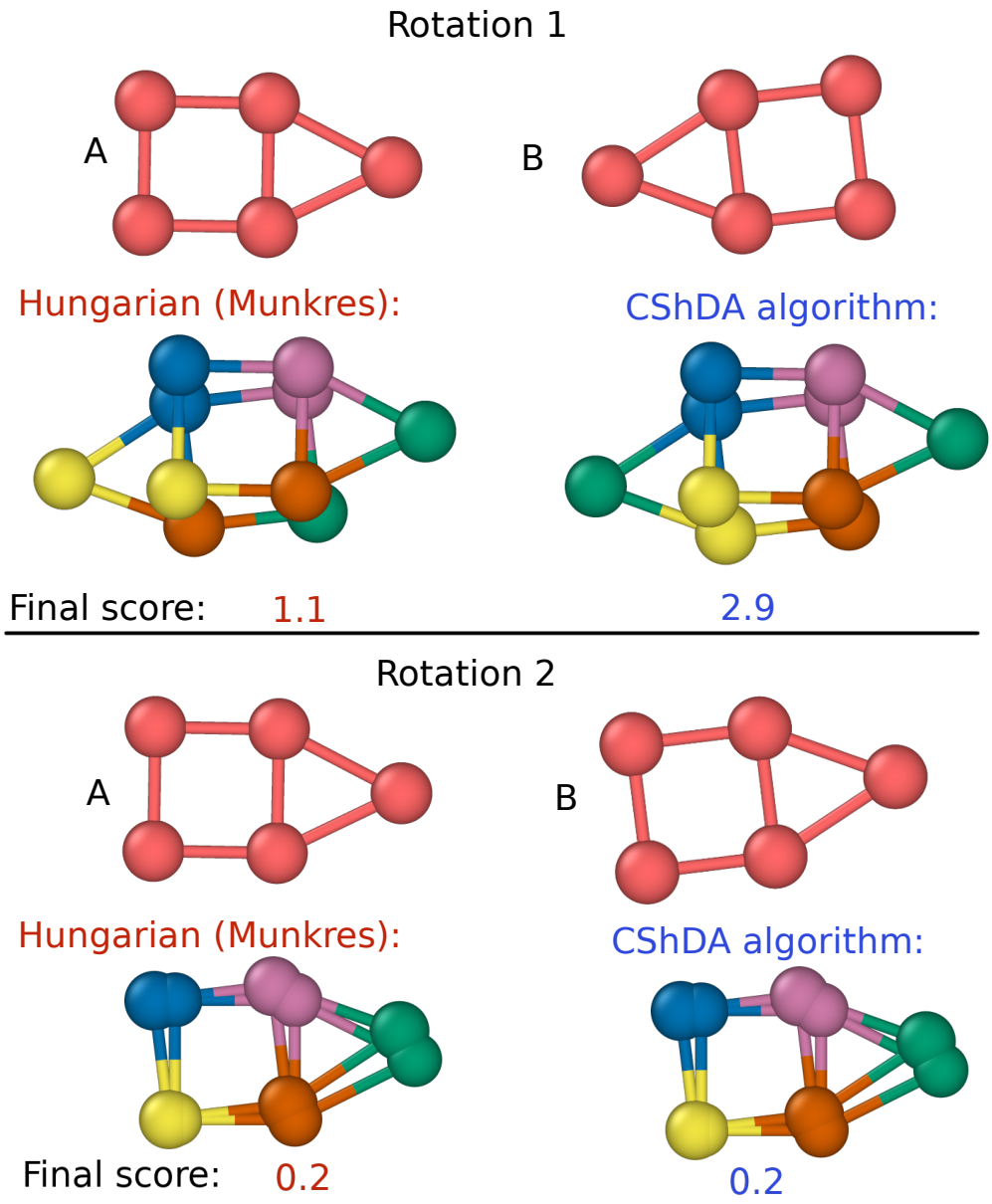



Figure 3: A schematic of the assignment problem, solved for structures $A$ and $B$ in two rotated states. On the left the assignment by the Hungarian algorithm following the algorithm proposed by Munkres,[27] and on the right by our CShDA algorithm. The colors show final assignments of atoms, e.g. a blue atom is assigned to a blue atom, yellow atom to yellow, etc. The final scores are computed as $max(d(a_i, b_i))$. The first rotated state could represent a particular intermediate step within the iterative rotations procedure (IRA).

The Hausdorff distance $d_H(A, B)$ between two structures $A$ and $B$ is formally defined as

$$d_H(A, B) = \max(h(A, B), h(B, A)) \tag{8}$$

where

$$h(A,B) = \max_{a \in A} \min_{b \in B} d(a,b) \tag{9}$$

where $d(a,b)$ is an Euclidean distance between points $a \in A$ and $b \in B$. The value of $h(A,B)$ is the largest value among the smallest distances from points in $A$ to points in $B$.

One can realize that our LAP algorithm corresponds to the *min* part of the Hausdorff distance in Eq. (9), with the additional constraint of one-to-one assignment. The evaluation of Eq. (9) is then the maximal distance $d(a_i, b_i)$ among all points $i$, where the order of atoms $b_i$ follows the assignment provided by the LAP algorithm.

## 2.2 Final Optimal Rotation

In the case in which the two systems are not equivalent, i.e. in the case of near-congruence, after finding the atomic assignments by our IRA algorithm, the optimal rotations are found via an SVD-based algorithm as follows.

Point sets $A$ and $B$ are shifted to their geometrical centers, obtaining $A' = \{a'_i = a_i - a^c\}$ and $B' = \{b'_i = b_i - b^c\}$ where $a^c$ and $b^c$ are the vectors of geometric centers of $A$ and $B$ respectively. A 3x3 matrix $\mathbf{H}$ is constructed from $n_A$ points which are common to $A'$ and $B'$ (to enable the decomposition for sets with different number of atoms).

$$\mathbf{H} = \sum_{i}^{n_A} |b'_i\rangle\langle a'_i|, \tag{10}$$

with $a'_i$ and $b'_i$ the vector points of $A'$ and $B'$, and $|..\rangle\langle..|$ denoting outer vector product. The SVD returns three matrices, $\mathbf{U}$, $\mathbf{S}$, and $\mathbf{V}$, such that $SVD(\mathbf{H}) = \mathbf{USV}^T$, where $\mathbf{U}$ and $\mathbf{V}$ are orthonormal matrices corresponding to rotations, and $\mathbf{S}$ contains the singular values on its diagonal. The rotation matrix $\mathbf{R}$ is then found as

$$\mathbf{R} = \mathbf{UV}^T, \tag{11}$$

and if $det(\mathbf{R}) = -1$, then $\mathbf{R}$ is multiplied by $diag(1, 1, -1)$. The translation vector $\mathbf{t}$ is found as

$$\mathbf{t} = a^c - \mathbf{R}b^c. \tag{12}$$

Rotation $\mathbf{R}$ and translation $\mathbf{t}$ found in this way, are such that the $RMSD(A, B)$ is minimized (details on SVD can be found in Ref. 18).

It is commonly believed that SVD-based algorithms are not particularly suited for matching purposes, due to the ability of SVD to find non-proper rotations,[23] i.e. rotation matrices with negative determinant. Such improper rotations correspond to reflections (sometimes also addressed as pseudorotations[42]), i.e. inversions, or mirroring over some axis, which changes the chirality of a vector set (which is not always desired, e.g.[32]). It has been suggested[18] to mitigate this issue by multiplying the rotation matrix by $diag(1, 1, -1)$, thus forcing a positive determinant. This strategy might however result in a completely wrong rotation, as the matrix $\mathbf{H}$ depends on the order of points (see Eq. (10)).

As our IRA approach (see Sec. 2.1) suggests permutations corresponding to both rotations and reflections, it is always able to rigorously keep track of what has been suggested, and properly enforce the final rotation matrix to have $det(\mathbf{R}) = 1$ (corresponding to rotation), or by multiplying it by $diag(1, 1, -1)$ to obtain $det(\mathbf{R}) = -1$ (corresponding to reflection). Thus consistently providing a correct rotation or reflection matrix.

# 3 Results

## 3.1 Exact congruence and equal number of atoms between sets

The reliability of the algorithm has been first checked by attempting to find the matching between a structure $A$ and a randomized version of that same structure $B$. The randomized structure $B$ is obtained by randomly permuting, translating by random vector (with norm in the interval (0,10]), rotating by a random angle (in the interval (0,2$\pi$]) along a random

rotation axis, and randomly mirroring the structure $A$. The structures $A$ used for this test are from the Cambridge Cluster Database,[43] more specifically we have used the TIP4P water clusters with $n = 2$ to $n = 21$ molecules of water in the cluster, and the Lennard-Jones (LJ) clusters of sizes $n = 3$ to $n = 150$ and from $n = 310$ to $n = 1000$ atoms, from the same database.[43] We have also used an amorphous Si structure with $n = 64$ atoms. Some sample structures are shown in Fig. 4. The test is done 10000 times for each of the water cluster structures, 100 times for each LJ structure, and 10000 times for Si structure. The final matching is evaluated by computing distances $h(A, B)$ and $RMSD(A, B)$ after the matching, they have in all cases both been below the floating point precision value (*i.e.,* zero). Which implies that with our approach, the correct transformation has always been found without mistake. The TIP4P test has also been performed by authors in Ref. 11. Their algorithm has failed for $n = 10$ once, for $n = 11$ once, and for $n = 13$ once. The same authors reported testing on five amino acids with the same procedure, however the structures of the amino acids claimed to be included in Supporting Information of Ref. 11 have not been found.

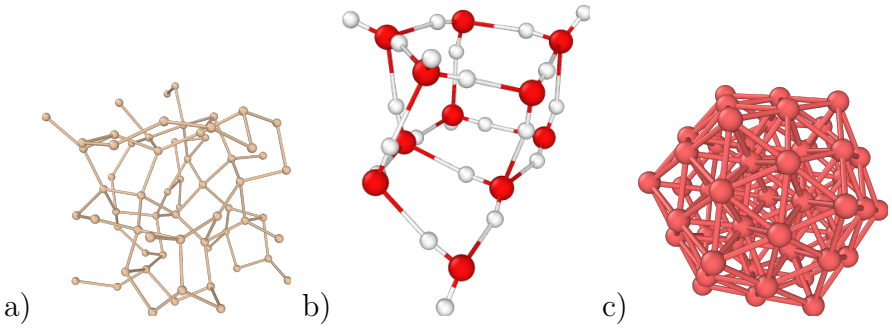



Figure 4: Some sample structures used to test the reliability of the overall algorithm: a) amorphous bulk silicon, b) $n = 11$ TIP4P water cluster, and c) $n = 52$ Lennard-Jones cluster.

To benchmark IRA with respect to other shape matching approaches, we have performed the same kind of tests against ArbAlign[37] and fastoverlap[10] algorithms. The testing procedure is identical to the previous paragraph, but done on the following datasets. From Ref. 37: datasets of Ne clusters with $n = \{10, 50, 100, 150, 200, 300, 500, 1000\}$ atoms, water clus-

ters with $n = 2$ to $n = 21$ and $n = \{25, 40, 60\}$ water molecules, FGG peptides with $n = 37$ atoms and 4 atomic types, and S1-MA-W1 hydrates with $n = 17$ atoms and 5 atomic types. From Ref. 44: Al clusters with $n = 63$ to $n = 160$ in steps of 1, and $n = 160$ to $n = 310$ in steps of 5 or 10. From Ref. 45: GaN clusters with $n = 12$ to $n = 96$ in steps of 2 or 4. From Ref. 46: Au26 clusters with $n = 26$ atoms and a varying number of atoms of a different type. From Ref. 43: Lennard-Jones clusters with $n = 5$ to $n = 150$ and $n = 310$ to $n = 520$. Each structure from each dataset is tried with 50 random initial transformations, and the final matching is marked as failure if the final distance $RMSD(A, B)$ is greater than threshold 0.001. The results of this test are reported in Table 1, containing the information on the total number of failures for each dataset. The values of final $RMSD$ distances, for each dataset where failures have occurred, are given in the Supporting Information, in Figs. S2-S7.

The algorithm ArbAlign[37] relies on principal axes of inertia as initial guess for rotations, uses the Hungarian[26] algorithm for the LAP, and minimizes rotations with SVD.[17] It considers 48 pre-defined symmetry operations applied in the reference frame of the principal axes. The algorithm fastoverlap[10] is based on kernel correlation. It uses Fourier transform to find maximum correlation between density representations of two structures.

Table 1: Results of the efficiency test of the three algorithms. Each dataset is referred to by its name, $N_s$ is the number of different structures in each dataset. Each structure from each dataset was tested with 50 random initial transformations. The tabulated values are in the form $m/n$, where $m$ is the total number of failures, and $n$ is the number of structures in which the failures occur. Values marked with *: the structures in this dataset include several atomic types, which fastoverlap cannot distinguish.

| Dataset | $N_s$ | ArbAlign[37] | fastoverlap[10] | IRA |
|---|---|---|---|---|
| Al[44] | 93 | 0/0 | 613/34 | 0/0 |
| Au26[46] | 6 | 186/4 | *0/0 | 0/0 |
| FGG[37] | 15 | 0/0 | *0/0 | 0/0 |
| GaN[45] | 31 | 50/1 | *294/14 | 0/0 |
| LJ[43] | 357 | 45/1 | 1177/113 | 0/0 |
| Neon[37] | 16 | 100/2 | 82/8 | 0/0 |
| S1MAW1[37] | 20 | 0/0 | *0/0 | 0/0 |
| water[37] | 70 | 0/0 | *217/11 | 0/0 |

From the results of our benchmark test in Table 1, we can conclude the following. The

algorithm ArbAlign[37] has problems to find the correct rigid transformation in structures where the principal axes of inertia are ambiguous, as anticipated in our introduction. This is very clear from the Au26 dataset from Ref. 46, which includes cylindrical shape structures, where only the principal axis along the cylinder is well defined. We note that since each structure was tried 50 times, the result of 186 failures in 4 structures (see Table 1) indicates that on average, there were 46 failed attempts out of 50 trials per structure.

On the other hand, the algorithm fastoverlap[10] shows a higher overall rate of mismatches, but with broadly dispersed failures. Interestingly, the final values of distance from fastoverlap show clustering around several distinct values for each structure (see Figs. S2-S7 in the Supporting Information), which might be the signature of trapping on some local minima.

Our proposed IRA algorithm shows a success rate of 100% across all of the structures tested. We can say with high confidence that it is fully reliable at finding any rigid transformation between two congruent structures.

## 3.2 Near congruence and equal number of atoms between sets

To test the performance under conditions of near congruence, i.e. the structures present some deformations - we perform a short NVT-ensemble Monte Carlo (MC) simulation for a LJ-20 cluster from the Cambridge Database[43] at two different temperatures. The specific temperatures used are $T = 0.02$ and $T = 0.3$ in the reduced units. These two values have been chosen as corresponding to "low" and "a bit higher", and are only used to induce some atomic vibration.

We take the equilibrium configuration of the cluster as reference structure $A$. At each step of the MC simulation, the current structure is taken as $B$, and the distance $RMSD_{ini} = RMSD(A, B)$ is calculated. During the MC, the structure undergoes some distortion, translation, and rotation, but not permutation of atoms. We can readily apply the SVD method to obtain rotation that minimizes $RMSD(A, B)$ at current step, store this $RMSD$ value as $RMSD_{ref}$. Then apply random rotation, reflection, translation, and permutation to

structure $B$, and run our shape matching algorithm on it, to obtain $B'$ aligned to $A$, and calculate distance $RMSD_{fin} = RMSD(A, B')$. The distance $RMSD_{fin}$ should be equal to $RMSD_{ref}$ if our algorithm has successfully found the right transformation. The results are shown on Fig. 5. The difference $RMSD_{ref} - RMSD_{fin}$ on every step is on the order of floating point precision error (*i.e.,* zero), confirming the ability of the presented approach to find the correct matching transformation efficiently.

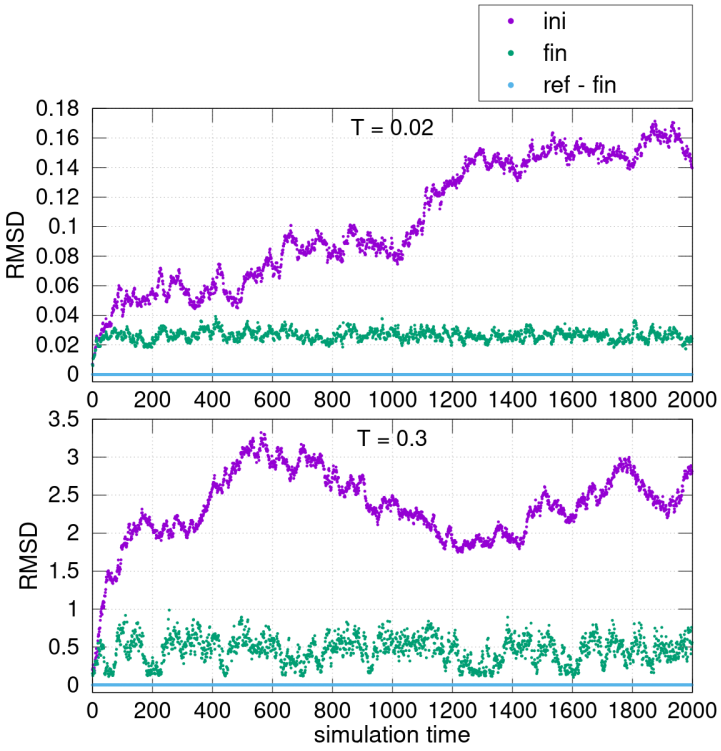



Figure 5: Plot of $RMSD_{ini}$, $RMSD_{fin}$, and the difference $RMSD_{ref} - RMSD_{fin}$ for temperatures (top) $T = 0.02$, and (bottom) $T = 0.3$.

The non-zero value of $RMSD_{fin}$, provides with a measure of the congruence between the structures.

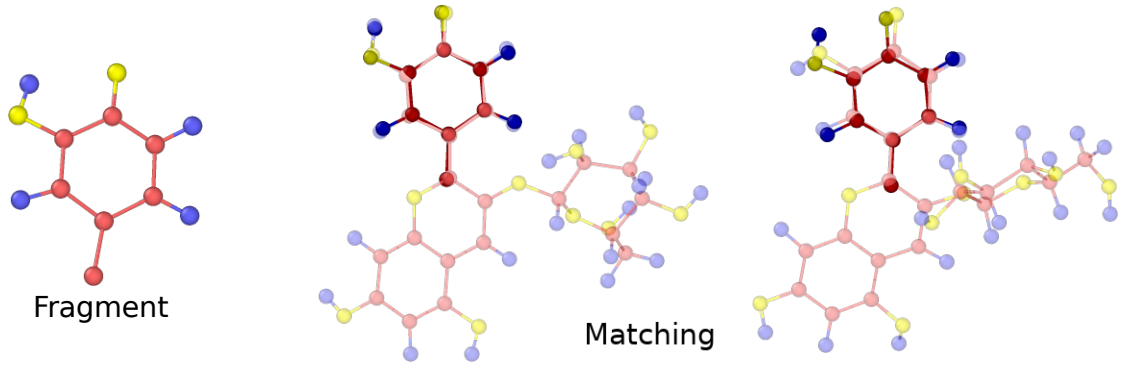



Figure 6: The fragment to be matched, and two instances of the final matching of the molecule, the atoms of the fragment are shown with a darker shade for better distinction. Red, blue and yellow atoms correspond to Carbon, Hydrogen and Oxygen atoms respectively, the same color code is used in the following.

### 3.3 Near congruence and different number of atoms between the sets

In order to show the ability and performances of our approach in finding the correct transformation and atomic assignment that best matches the structural fragments to a larger structure, we use a trajectory of replica-exchange molecular dynamics simulation of the cyanine molecule (data provided by authors of Ref. 40). We select two kinds of fragments, a connected one shown in Fig. 6, and a non-connected one shown in Fig. 7.

During the trajectory, the atoms move and distort the molecule, but they do not permute. Thanks to this, we can apply a similar test for reliability as in the previous section. We choose a fixed reference fragment $A$, and compute the optimal rotation of molecule $B$ using SVD, giving $RMSD_{ref} = RMSD(A, B)$. Then we randomly rotate, reflect, translate, and permute structure $B$, and run our shape matching algorithm on it, to obtain $B'$ aligned to fragment $A$, and calculate $RMSD_{fin} = RMSD(A, B')$. The distances $RMSD_{ref}$ and $RMSD_{fin}$ should be equal if the right transformation has successfully been found. The sum in all $RMSD$ calculations in this case goes up to number $n_A$ of atoms in fragment $A$.

The result when structure $A$ is the connected fragment from Fig. 6, is that out of the eighty thousand configurations in the trajectory, there are 313 instances of the difference

$RMSD_{ref} - RMSD_{fin}$ being above the floating point precision value. These instances represent structures where the algorithm has mismatched the fragment. Some of the reasons for this behaviour are explored in the discussions section (Sec. 4). However a deep analysis of the particular instances is beyond the scope of the current paper.

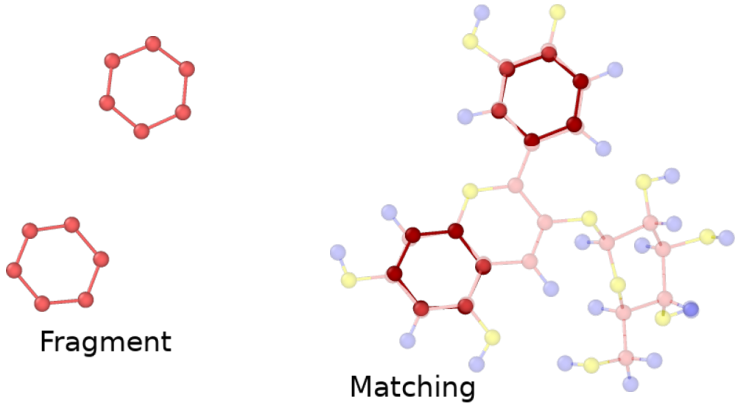



Figure 7: A disconnected fragment, and matching of a molecule.

Tracking the number of mismatches when structure $A$ is the non-connected fragment from Fig. 7 is not straightforward, since the two hexagons do not move rigidly. As a consequence, $RMSD_{ref}$ as defined previously is ambiguous.

# 4 Discussion

In the IRA part of the algorithm (Sec. 2.1), the evaluation of Hausdorff distance $h(A, B)$ is compliant with the one-to-one matching constraint of the CShDA, and strictly corresponds to distance function $D$ in Eq. (2). Due to the relatively low number of atoms in the atomic structure matching, the usage and implementation of the Hausdorff distance needs some attention. The expression for $h(A, B)$ in Eq. (9) is only commutative when $A$ and $B$ contain the same number of points, which is the reason the expression for Hausdorff distance is generally written in the form of Eq. (8), which penalizes the situation where some points are present in one structure but not in the other. Fig. 8 schematically shows the shortest distances between points of set $A$(triangles) and points of set $B$(circles) as arrows, where

the largest among them is colored in red and represents the value of $h(A, B)$, and $h(B, A)$ respectively. As described in Sec. 2.1.1, the assignment of atoms is done under the one-to-one constraint, which poses a problem for the situation of $h(B, A)$ on the right side of Fig. 8, where $B$ contains more atoms than $A$, since two atoms of $B$ get assigned to the same atom of $A$. A mitigation for avoiding this problem is to systematically impose that the number of atoms $n_A \leq n_B$, which is the situation of $h(A, B)$ on the left side of Fig. 8. This imposition also opens up the possibility of matching fragments. However, the fragment as a whole needs to be a substructure of the larger structure, i.e. our proposed algorithm is not finding the largest common subset of both the structures.

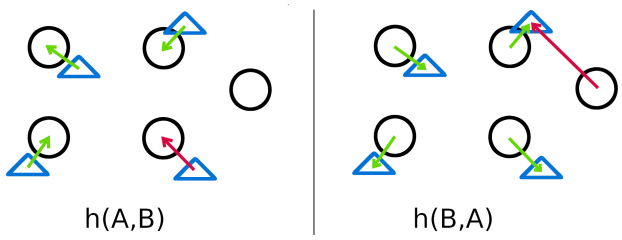



Figure 8: Schematic representation of the difference between $h(A, B)$ on the left, and $h(B, A)$ on the right, when $A$ and $B$ contain different number of points. Set $A$ is represented by triangles, set $B$ by circles. Arrows show the minimum distances between points in green, and the maximum value in red, $h(A, B)$ and $h(B, A)$ respectively.

As the value of $h$ only takes the maximal distance in Eq. (9), it only contains information about one specific atom/point. This particularity can be advantageous in cases of low distortions between the structures, where the value of $h$ is low, meaning that all atoms are within a low-distance $h$ of the reference positions. Larger distortions lead to higher $h$ value, which can hide the behavior of any specific atom. A high $h$ can be due to single atom distortion, and any information on other atoms is completely obscured. This property of Hausdorff distance is often described as high susceptibility to noise. It opens the possibility of a situation in our algorithm, where a "wrong" assignment gives a transformation $U_J^\dagger$ whose distance $D(A_{\{\hat{\mathbf{e}}\}}, B_{\{\hat{\mathbf{e}}'\}_J})$ is lower than the distance $D$ when the transformation is given by the "correct" assignment, which then leads to a wrong final assignment and transformation. Replacing the $h$ with a sum of minimal distances, which should capture a more "collective"

behaviour of the atoms, has not shown any significant changes in the performances with the highly distorted cyanine molecule tests (Sec. 3.3). The mismatches still happen at large set-set distance values. The choice of a particular set-set distance function is therefore not crucial, as long as the distance complies with the permutational invariance, and translational and rotational variance, imposed by Eq. (2). The "mismatches" are rather due to attempting to match structures that are far from congruence. Which raises the general question for any structure similarity approach, how meaningful can it be to attempt matching such structures, and how could the results be interpreted? On the specific and known case of the cyanine we were able to assess that there were mismatches, but for huge data sets for which the parsing is generally blind, the meaning of large distances and their interpretation should be of concern.

It is possible to reduce the number of mismatches by assuming some prior knowledge on the system. The first step of our IRA algorithm (Sec. 2.1) selects a central atom in structure $A$ by the criteria of closeness to the geometrical center of $A$. The second step is to select a basis $\{\hat{\mathbf{e}}\}$ for a reference frame in $A$, which is based on positions of atoms around the central atom. Then the structure $B$ is searched for the equivalent basis $\{\hat{\mathbf{e}}'\}_J$. When large distortions are present in structure $B$, there is no guarantee that the basis found in $B$ is equivalent to the basis found in $A$, or that it even exists. If we assume that there still exist local environments in the two structures that are congruent to each other, then the central atom of $A$ could be chosen as the atom for which its local environment is the most similar to any local environment in $B$. Choosing the central atom in $A$ according to that criterion in the case of cyanine for instance, reduces the number of mismatches by an order of magnitude (313 originally, 30 with this choice).

As already mentioned in Sec. 2.1, the total number of rotations tested $N_R$ is greatly dependent on the structure. In this respect, the Al dataset, along with LJ and Ne datasets from the benchmark test in Sec. 3.1, represent worst-case scenarios for IRA as all atoms are of the same atomic type, and the structures are close-packed, which yields the highest

number of reference frames to be tested. This number is related to the structure surrounding the origin point, as mentioned in Sec. 2.1. For example, in the Al dataset,[44] the number of rotations tested for each member structure varies on the range [2, 154], without any apparent rule (see also Fig. S8 in Supporting Information). In that example, there is a single origin point, which is set to the geometrical center of the structures. A higher number of rotations needs to be tested when the geometrical center coincides with an atomic position. In that case, a larger number of atoms is included in the radial cutoff region, which defines the possible reference frames. Conversely, when the geometrical center falls in between atoms, the number of atoms in the region is lower, and thus less reference frames have to be tested. In the case of matching structures with different number of atoms, the origin point is set by the central atom in structure $A$. In that case, each possible central atom of structure $B$ gets tested with a number of rotations that depends on the local environment of that atom. In any case, the number of rotations tested is not explicitly related to the total number of atoms $N$, but related to the density of atoms in the region around the origin point, and the number of possible origin points. When prior knowledge of the origin point in the form of a known central atom is assumed, as discussed in the previous paragraph, the number of rotations tested is given only by the local environment surrounding that specific atom. The overall performance thus depends on the specific atomic structure, and any prior knowledge influencing the choice of the origin point.

In situations when we know that the two structures being matched are sufficiently similar, the multiplication factor 1.2, used for the cutoff can be reduced, but the value should in any case remain above 1.0. This effectively reduces the search space of rotations, and the algorithm can be faster as a result. When matching structures with different number of atoms, making a computational effort to reduce the number of candidate central atoms, as previously mentioned, can also be very beneficial for the speed of the algorithm, as it reduces the set of possibilities. In situations where the equality of two structures is being tested with a certain known threshold for equality, heuristic approaches can be used on top of the logic

of the IRA and CShDA algorithms, to exit certain loops as soon as certain criteria are met. This method has the potential to speed up the algorithm considerably, however at the cost of generality. Because of the non straight-forward relationship between the speed of IRA algorithm and the atomic structure, a discussion about scalability would hardly be useful. As point of reference for the timing, our fortran implementation of IRA as described in this work, running on a single core of a standard laptop: matching the LJ $n = 100$ cluster[43] with a randomized version of itself takes about 0.02 seconds with 40 rotations tested, and 0.15 seconds for the LJ $n = 400$ cluster with 12 rotations tested. However these numbers cannot be generalized at all.

Similarly, when matching structures with different number of atoms, the best-case and worst-case scenarios in terms of overall speed of execution, would be the following. Best-case would be matching a fragment of a low-density structure, to a slightly larger structure with a small number of possible central atoms, meaning the central atom of $A$ has an atomic type that is not very present in structure $B$ (as is the case for example for some organic compounds). The worst-case scenario would be matching a fragment of a high-density structure, to a much larger structure with many possible central atoms (as for example in close-packed bulk structures).

Once the transformation that best matches one structure to the other is found, the corresponding set-set distance value becomes a similarity measure or a distortion score: a similarity measure that is not an arbitrary choice, but that arises from a minimization. As our approach is also able to match fragments (connected or not), including a lattice periodicity, it can provide with a similarity measure for any part of any structure.

Exploited in (semi)-blind fragment exploration, our approach could aid in revealing the most important collective coordinates, which ultimately cluster the data set along the relevant collective axes. For example, Fig. 9 and Fig. 10 show two sample histograms of $RMSD$ for the final matching of the eighty thousand trajectory steps of the cyanine example (see Sec. 3.3) with respect to two sample reference fragments. The cases of mismatching are

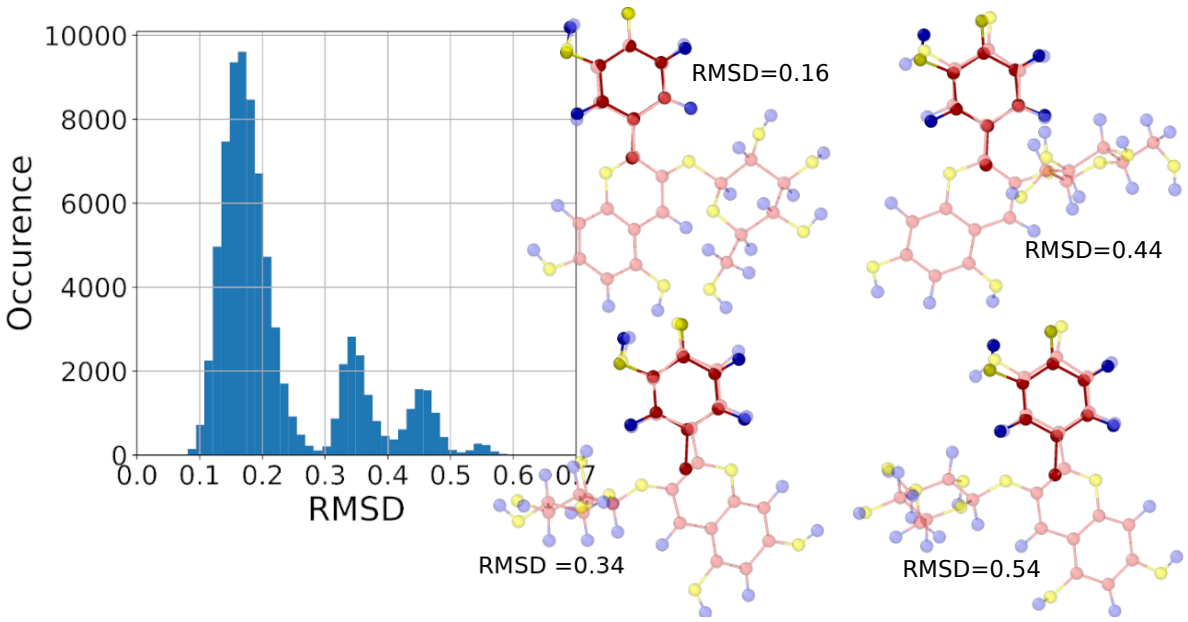



Figure 9: Histogram of $RMSD$ values of the final matching for 80 thousand trajectory steps. We clearly see four peaks, representing four clusters of structures in the MD trajectory, the typical member structure corresponding to each peak is shown. The viewing angle is such that the reference fragment, shown in darker colors, is kept fixed on all images.

excluded from these plots. In Fig. 9, four peaks can be identified, representing the grouping of structures in the MD trajectory into four clusters. From the representative fragments belonging to each cluster, we can notice that there is a H-atom (blue) that rotates around an O-atom (yellow), and that the rest of the molecule that is attached through the bottom C-atom (red) of the fragment is roughly oriented in two main directions. Indeed the original paper with the cyanine molecule[40] reports the dihedral angle going through the bottom C-atom as one of the relevant axes which clusters the whole data set into two main groups.

In the context of amorphous or disordered structures, it can also enable the characterization and analysis of local disorder at different scales, i.e. as a function of the number of neighbors included in the fragment and accounted during matching. Fig. 11, shows the Hausdorff and $RMSD$ distance color map for $SiO_4$ tetrahedra in silica. In this example, IRA was used to find the matching between an ideal $SiO_4$ tetrahedron and the whole silica crystal, centered on each of the Si atoms. The O atoms are shown in blue, and Si atoms are colored by the value of chosen distance function. The color map is compared to the values

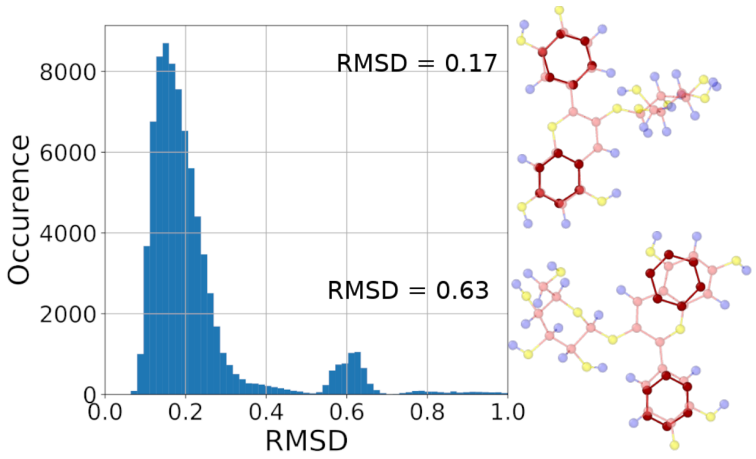



Figure 10: Histogram of $RMSD$ values of the final matching of a disconnected fragment. Two peaks can be identified, corresponding to the grouping of the structures into two clusters. A representative structure from each cluster is shown.

obtained through the Keating potential,[47] which is a strain-based potential, where a low value corresponds to Si atoms with local environments closely resembling a tetrahedron (low strain), and higher values otherwise (higher strain).

Finally, because of the ability of our approach to match non-connected fragments, it can be also exploited to compute time correlation functions based on fragments taken at two different times.

# 5 Conclusion

In this work, we have presented an alternative, parameter-less shape matching approach that allows to find isometric transformations (rigid rotation, reflection, translation, and permutation/atomic assignment) between congruent and near-congruent structures that do not necessarily have the same number of atoms, and that can be part of a periodic lattice. The *best match* transformation coincides with a minimum of the set-set distance, which has value zero in case of exact congruence between the structures. As such, the set-set distance can be interpreted as a measure of similarity, thus enabling the use of our approach for comparing and recognizing atomic structures. The CShDA algorithm, the LAP solver we developed, is

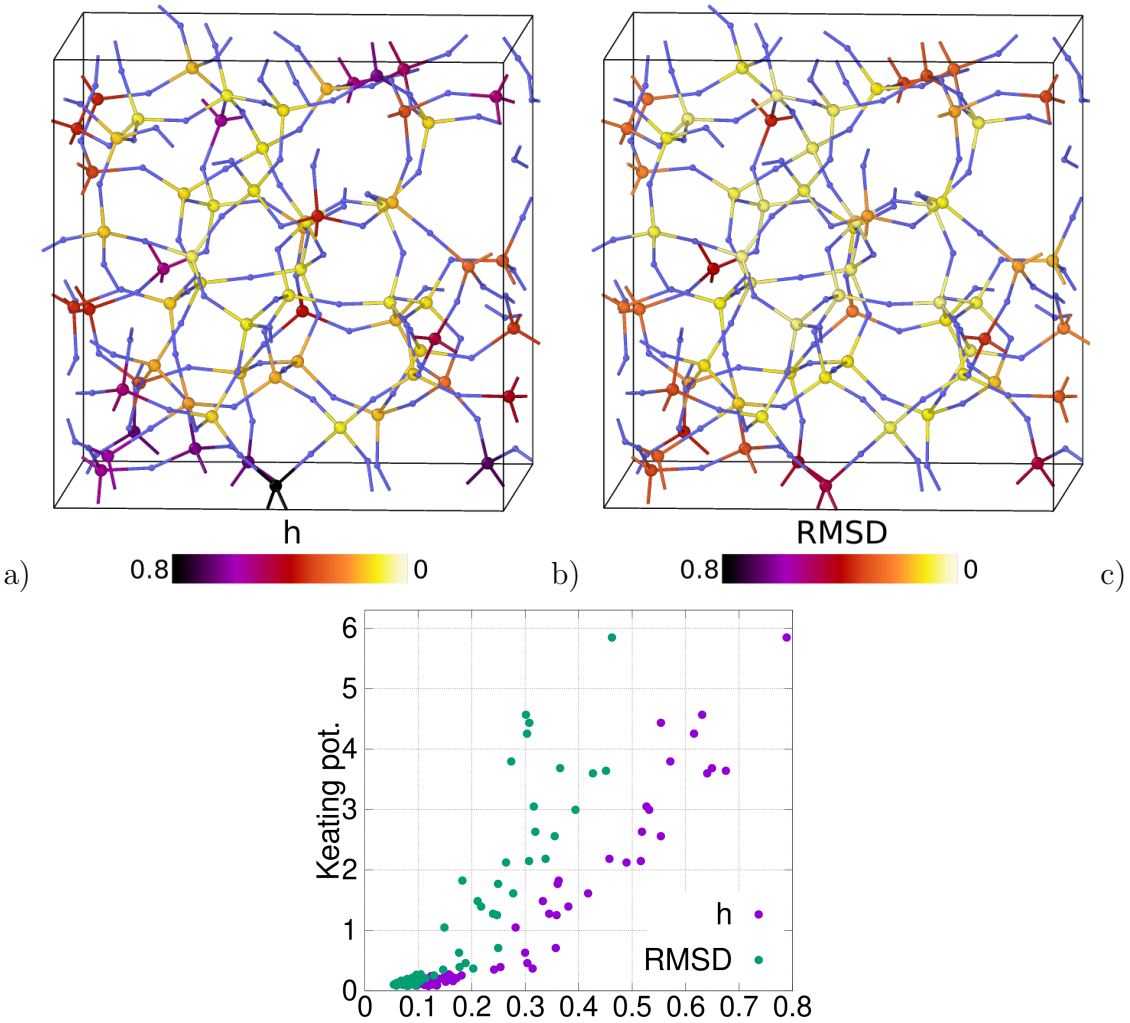



Figure 11: Color map of distortions in a 192 atoms silica model, as obtained through a) $h(A, B)$, b) $RMSD(A, B)$, and c) correlation with respect to Keating potential.[47]

able to compute atomic assignments for structures with non-equal number of atoms. This is exploited in the IRA algorithm, and enables the resolution of the shape matching problem for structural fragments. Among the performed tests, the reliability of the algorithm is 100% in the case of exact congruence of structures (Sec. 3.1), while the performances might drop slightly for larger deformations (99.6% in the cyanine case Sec. 3.3). When available, prior knowledge of the structures can be exploited to reduce the number of mismatches. In the context of finding correlations and identifying collective behaviours, our approach could aid in revealing the most important collective axes, either in space or time.

## Data and source code availability

IRA is released under double licensing, GPL v3 and Apache v2. The source code and data used for testing and benchmarking is available at `https://github.com/mammasmias/IterativeRotationsAssignments`. For cyanine trajectory please contact authors in Ref.[40]

## Acknowledgement

The authors are active members of the Multiscale And Multi-Model Approach for MaterialS In Applied Science consortium (MAMMASMIAS consortium), and acknowledge the efforts of the consortium in fostering scientific collaboration. This work was partially funded from the European Union's Horizon 2020 research and innovation program under grant agreement No. 871813 MUNDFAB, and by the European Union's Horizon 2020 research and innovation program under grant agreement No 899285 MAGNELIQ. All images of atomic structures in this article were generated with ovito[48] software.

**ASSOCIATED CONTENT:** Supporting Information is Available free of charges. The supporting Information contains detailed figures of the benchmark test results as well as a detailed analysis on the rotations that are required to find the approximate rotation transformation.

## References

(1) Leach, A. R.; Gillet, V. J.; Lewis, R. A.; Taylor, R. Three-Dimensional Pharmacophore Methods in Drug Discovery. J. Med. Chem. **2010**, 53, 539–558.

(2) Giangreco, I.; Cosgrove, D. A.; Packer, M. J. An Extensive and Diverse Set of Molecular Overlays for the Validation of Pharmacophore Programs. J. Chem. Inf. Model. **2013**, 53, 852–866.

(3) Brown, B. P.; Mendenhall, J.; Meiler, J. BCL::MolAlign: Three-Dimensional Small Molecule Alignment for Pharmacophore Mapping. J. Chem. Inf. Model. **2019**, 59, 689–701.

(4) Schönborn, S. E.; Goedecker, S.; Roy, S.; Oganov, A. R. The Performance of Minima Hopping and Evolutionary Algorithms for Cluster Structure Prediction. J. Chem. Phys. **2009**, 130, 144108.

(5) Sierka, M. Synergy Between Theory and Experiment in Structure Resolution of Low-Dimensional Oxides. Prog. Surf. Sci. **2010**, 85, 398–434.

(6) Weal, G. R.; McIntyre, S. M.; Garden, A. L. Development of a Structural Comparison Method to Promote Exploration of the Potential Energy Surface in the Global Optimization of Nanoclusters. J. Chem. Inf. Model. **2021**, 61, 1732–1744.

(7) Ferrando, R.; Fortunelli, A.; Johnston, R. L. Searching for the Optimum Structures of Alloy Nanoclusters. Phys. Chem. Chem. Phys. **2008**, 10, 640–649.

(8) Yang, S.; Day, G. M. Exploration and Optimization in Crystal Structure Prediction: Combining Basin Hopping with Quasi-Random Sampling. J. Chem. Theory Comput. **2021**, 17, 1988–1999.

(9) Trott, O.; Olson, A. J. AutoDock Vina: Improving the Speed and Accuracy of Docking With a New Scoring Function, Efficient Optimization, and Multithreading. J. Comput. Chem. **2010**, 31, 455–461.

(10) Griffiths, M.; Niblett, S. P.; Wales, D. J. Optimal Alignment of Structures for Finite and Periodic Systems. J. Chem. Theory Comput. **2017**, 13, 4914–4931, PMID: 28841314.

(11) Helmich, B.; Sierka, M. Similarity Recognition of Molecular Structures by Optimal Atomic Matching and Rotational Superposition. J. Comput. Chem. **2012**, 33, 134–140.

(12) Green, B. F. The Orthogonal Approximation of an Oblique Structure in Factor Analysis. Psychometrika **1952**, 17, 429–440.

(13) Strauss, H. L.; Pickett, H. M. Conformational Structure, Energy, and Inversion Rates of Cyclohexane and Some Related Oxanes. J. Am. Chem. Soc. **1970**, 92, 7281–7290.

(14) Fábri, C.; Mátyus, E.; Császár, A. G. Numerically Constructed Internal-Coordinate Hamiltonian With Eckart Embedding and Its Application for the Inversion Tunneling of Ammonia. Spectrochim. Acta, Part A **2014**, 119, 84 – 89, Frontiers in molecular vibrational calculations and computational spectroscopy.

(15) Horn, B. K. P.; Hilden, H. M.; Negahdaripour, S. Closed-Form Solution of Absolute Orientation Using Orthonormal Matrices. J. Opt. Soc. Am. A **1988**, 5, 1127–1135.

(16) Cliff, N. Orthogonal Rotation to Congruence. Psychometrika **1966**, 31, 33–42.

(17) Kabsch, W. A Solution for the Best Rotation To Relate Two Sets of Vectors. Acta Cryst. A **1976**, 32, 922–923.

(18) Arun, K. S.; Huang, T. S.; Blostein, S. D. Least-Squares Fitting of Two 3-D Point Sets. IEEE Transactions on Pattern Analysis and Machine Intelligence **1987**, PAMI-9, 698–700.

(19) Horn, B. K. P. Closed-Form Solution of Absolute Orientation Using Unit Quaternions. J. Opt. Soc. Am. A **1987**, 4, 629–642.

(20) Kearsley, S. K. On the Orthogonal Transformation Used for Structural Comparisons. Acta Cryst. A **1989**, 45, 208–210.

(21) Kneller, G. R. Superposition of Molecular Structures using Quaternions. Mol. Simul. **1991**, 7, 113–119.

(22) Krasnoshchekov, S. V.; Isayeva, E. V.; Stepanov, N. F. Determination of the Eckart Molecule-Fixed Frame by Use of the Apparatus of Quaternion Algebra. J. Chem. Phys. **2014**, 140, 154104.

(23) Flower, D. R. Rotational Superposition: A Review of Methods. J. Mol. Graph. Model. **1999**, 17, 238–244.

(24) Coutsias, E. A.; Wester, M. J. RMSD and Symmetry. J. Comput. Chem. **2019**, 40, 1496–1508.

(25) Hanson, A. J. The Quaternion-Based Spatial-Coordinate and Orientation-Frame Alignment Problems. Acta Cryst. A **2020**, 76, 432–457.

(26) Kuhn, H. W. The Hungarian Method for the Assignment Problem. Naval Research Logistics Quarterly **1955**, 2, 83–97.

(27) Munkres, J. Algorithms for the Assignment and Transportation Problems. J. Soc. Ind. Appl. Math. **1957**, 5, 32–38.

(28) Jonker, R.; Volgenant, A. A Shortest Augmenting Path Algorithm for Dense and Sparse Linear Assignment Problems. Computing **1987**, 38, 325–340.

(29) Besl, P. J.; McKay, N. D. A Method for Registration of 3-D Shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence **1992**, 14, 239–256.

(30) Pottmann, H.; Huang, Q.-X.; Yang, Y.-L.; Hu, S.-M. Geometry and Convergence Analysis of Algorithms for Registration of 3D Shapes. Int. J. Comput. Vision **2006**, 67, 277–296.

(31) Blatov, I. A.; Kitaeva, E. V.; Shevchenko, A. P.; Blatov, V. A. A Universal Algorithm for Finding the Shortest Distance Between Systems of Points. Acta Cryst. A **2019**, 75, 827–832.

(32) Richmond, N. J.; Willett, P.; Clark, R. D. Alignment of Three-Dimensional Molecules Using an Image Recognition Algorithm. J. Mol. Graphics Modell. **2004**, 23, 199–209.

(33) Eckart, C. Some Studies Concerning Rotating Axes and Polyatomic Molecules. Phys. Rev. **1935**, 47, 552–558.

(34) Louck, J. D.; Galbraith, H. W. Eckart Vectors, Eckart Frames, and Polyatomic Molecules. Rev. Mod. Phys. **1976**, 48, 69–106.

(35) Allen, W. J.; Rizzo, R. C. Implementation of the Hungarian Algorithm to Account for Ligand Symmetry and Similarity in Structure-Based Design. J. Chem. Inf. Model. **2014**, 54, 518–529, PMID: 24410429.

(36) Wagner, A.; Himmel, H.-J. aRMSD: A Comprehensive Tool for Structural Analysis. J. Chem. Inf. Model. **2017**, 57, 428–438, PMID: 28191844.

(37) Temelso, B.; Mabey, J. M.; Kubota, T.; Appiah-Padi, N.; Shields, G. C. ArbAlign: A Tool for Optimal Alignment of Arbitrarily Ordered Isomers Using the Kuhn–Munkres Algorithm. J. Chem. Inf. Model. **2017**, 57, 1045–1054, PMID: 28398732.

(38) Sadeghi, A.; Ghasemi, S. A.; Schaefer, B.; Mohr, S.; Lill, M. A.; Goedecker, S. Metrics for Measuring Distances in Configuration Spaces. J. Chem. Phys. **2013**, 139, 184118.

(39) Eiter, T.; Mannila, H. Distance Measures for Point Sets and Their Computation. Acta Inf. **1997**, 34, 109–133.

(40) Rusishvili, M.; Grisanti, L.; Laporte, S.; Micciarelli, M.; Rosa, M.; Robbins, R. J.; Collins, T.; Magistrato, A.; Baroni, S. Unraveling the Molecular Mechanisms of Color Expression in Anthocyanins. Phys. Chem. Chem. Phys. **2019**, 21, 8757–8766.

(41) Burkard, R.; Dell'Amico, M.; Martello, S. Assignment Problems; SIAM e-books; Society for Industrial and Applied Mathematics (SIAM, 3600 Market Street, Floor 6, Philadelphia, PA 19104), 2009.

(42) Dymarsky, A. Y.; Kudin, K. N. Computation of the Pseudorotation Matrix To Satisfy the Eckart Axis Conditions. J. Chem. Phys. **2005**, 122, 124103.

(43) Wales, D. J.; Doye, J. P. K.; Dullweber, A.; Hodges, M. P.; Calvo, F. Y. N. F.; Hernández-Rojas, J.; Middleton, T. F. The Cambridge Cluster Database. `https://www-wales.ch.cam.ac.uk/CCD.html`.

(44) Shao, X.; Wu, X.; Cai, W. Growth Pattern of Truncated Octahedra in AlN ($N \leq 310$) Clusters. J. Phys. Chem. A **2010**, 114, 29–36, PMID: 20014801.

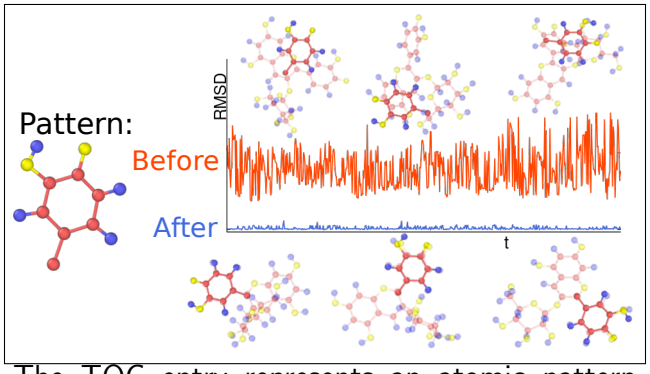
(45) Brena, B.; Ojamäe, L. Surface Effects and Quantum Confinement in Nanosized GaN Clusters: Theoretical Predictions. J. Phys. Chem. C **2008**, 112, 13516–13523.

(46) Liu, Q.; Xu, C.; Wu, X.; Cheng, L. Electronic Shells of a Tubular Au26 Cluster: A Cage–Cage Superatomic Molecule Based on Spherical Aromaticity. Nanoscale **2019**, 11, 13227–13232.

(47) Lee, S.; Bondi, R. J.; Hwang, G. S. Atomistic Structural Description of the Si(001)/a-SiO2 Interface: The Influence of Different Keating-Like Potential Parameters. J. Appl. Phys. **2011**, 109, 113519.

(48) Stukowski, A. Visualization and Analysis of Atomistic Simulation Data With OVITO-the Open Visualization Tool. Modell. Simul. Mater. Sci. Eng. **2010**, 18.

# Graphical TOC Entry




The TOC entry represents an atomic pattern to be matched to some output trajectory of a molecule containing this atomic pattern, in particular a replica exchange molecular dynamics of cyanine. However, the atomic assignments have been randomly changed. Thus the pattern cannot be easily superposed to the molecule as atomic assignments are unknown. This is depicted with the three images above the plot, the RMSD measured in this state is large, depicted by the orange line "Before". After applying our shape matching algorithm, the atomic pattern is found within the molecule in each step of the trajectory. This is depicted by the three images under the plot, the RMSD measured in this state reaches a minimum value, depicted by the blue "After" line in the plot.

# Supporting Information

# IRA: A shape matching approach for recognition and comparison of generic atomic patterns

Miha Gunde,*,†,‡ Nicolas Salles,‡ Anne Hémeryck,† and Layla Martin-Samos*,‡

†*LAAS-CNRS, Université de Toulouse, CNRS, 7 avenue du Colonel Roche, 31031 Toulouse, France*

‡*CNR-IOM, Democritos National Simulation Center, Istituto Officina dei Materiali, c/o SISSA, via Bonomea 265, IT-34136 Trieste, Italy*

E-mail: miha.gunde@gmail.com; marsamos@iom.cnr.it

## Results of the benchmark test

The Fig. S1 shows representative structures from each dataset included in the benchmark test of Sec. 3.1. The collection of structures included in the benchmark test forms a diverse set of general shapes. More details about these structures can be found in their respective original works.[1–5]

A final transformation having $RMSD(A, B) > 0.001$ is considered a mismatch. Failures are reported for each software in Figs. S2-S7. The horizontal axis on these plots gives the name of the particular structure where a failure has occurred, the vertical axis is the number of current trial, the color of a point gives the final value $RMSD$, and the shape of a point is related to the particular software which returned the failure.

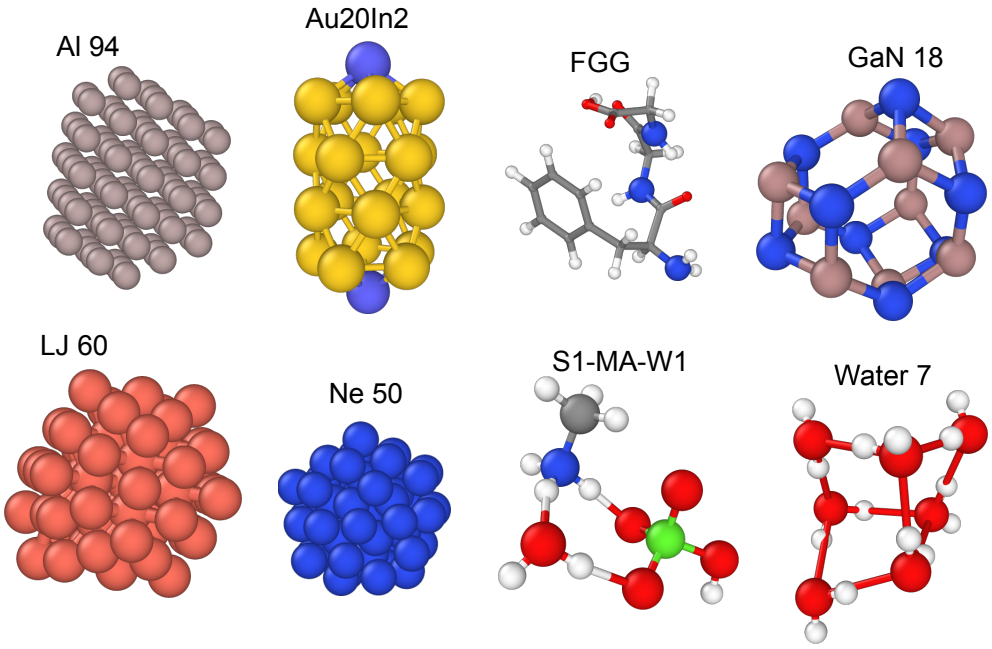



Figure S1: Representative structures from each dataset used in the benchmark test of Sec. 3.1. Note the diversity of general shape in the structures.

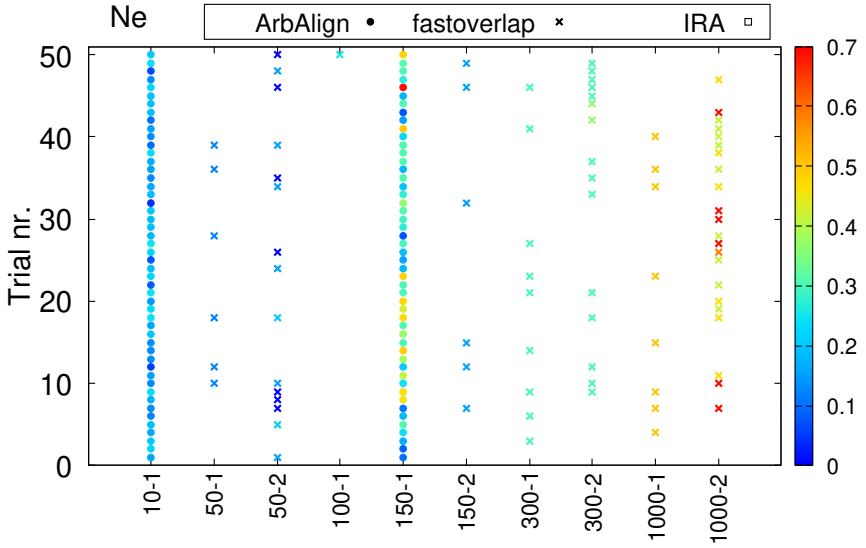



Figure S2: Values of final $RMSD$ for structures from the Ne dataset. Only failures are reported. Structure name on horizontal axis, trial number on vertical, final $RMSD$ value in color. Failures in this dataset: 100 failures in 2 structures by ArbAlign; 82 failures in 8 structures by fastoverlap; 0 failures by IRA.

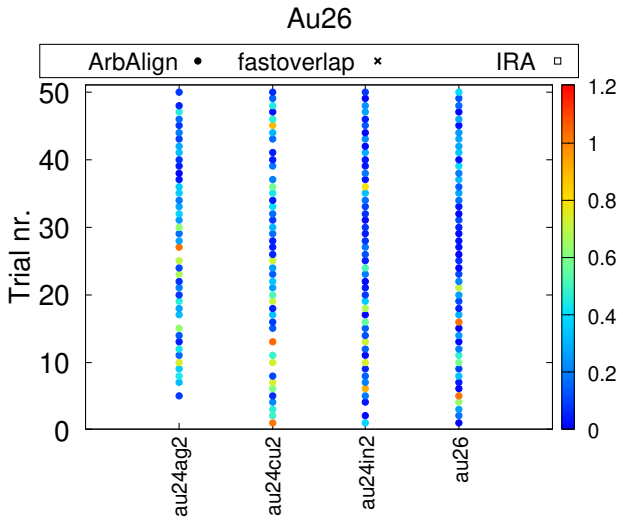



Figure S3: Values of final $RMSD$ for structures from the Au26 dataset. Only failures are reported. Structure name on horizontal axis, trial number on vertical, final $RMSD$ value in color. Failures in this dataset: 186 failures in 4 structures by ArbAlign; 0 failures by fastoverlap; 0 failures by IRA.

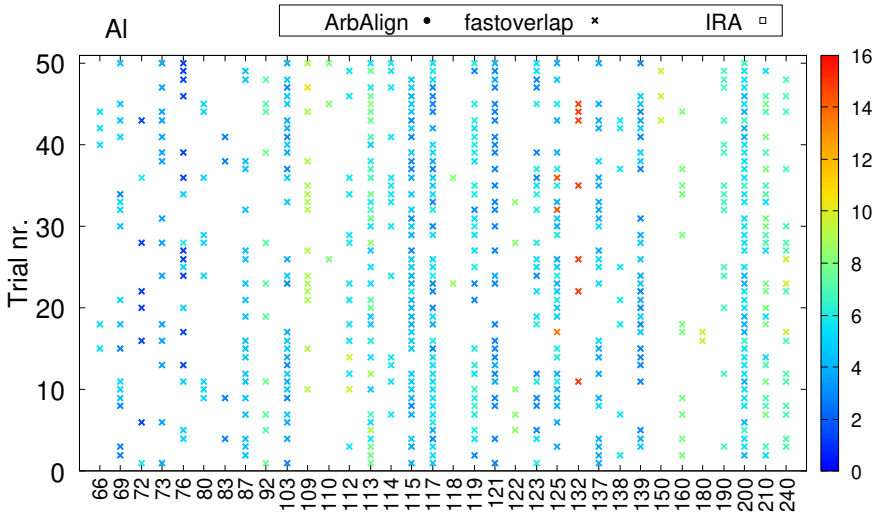



Figure S4: Values of final $RMSD$ for structures from the Al dataset. Only failures are reported. Structure name on horizontal axis, trial number on vertical, final $RMSD$ value in color. Failures in this dataset: 0 failures by ArbAlign; 613 failures in 34 structures by fastoverlap; 0 failures by IRA.

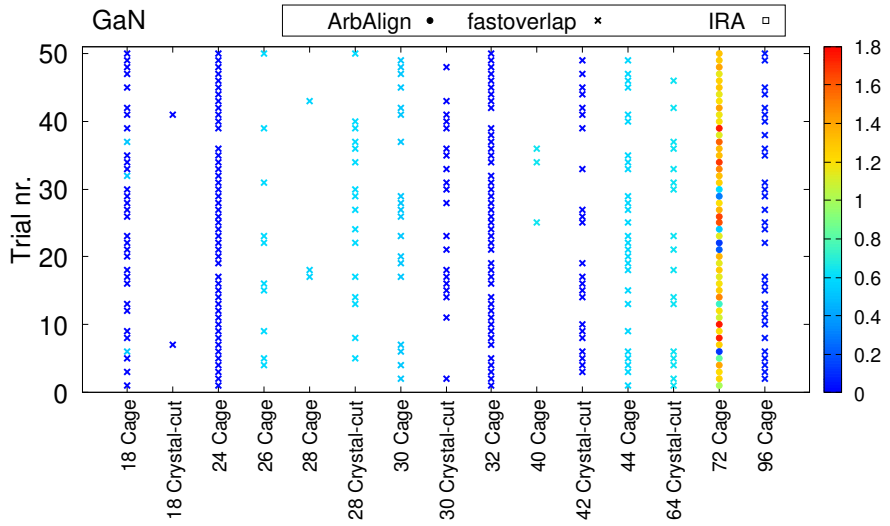



Figure S5: Values of final $RMSD$ for structures from the GaN dataset. Only failures are reported. Structure name on horizontal axis, trial number on vertical, final $RMSD$ value in color. Failures in this dataset: 50 failures in 1 structure by ArbAlign; 294 failures in 14 structures by fastoverlap; 0 failures by IRA.

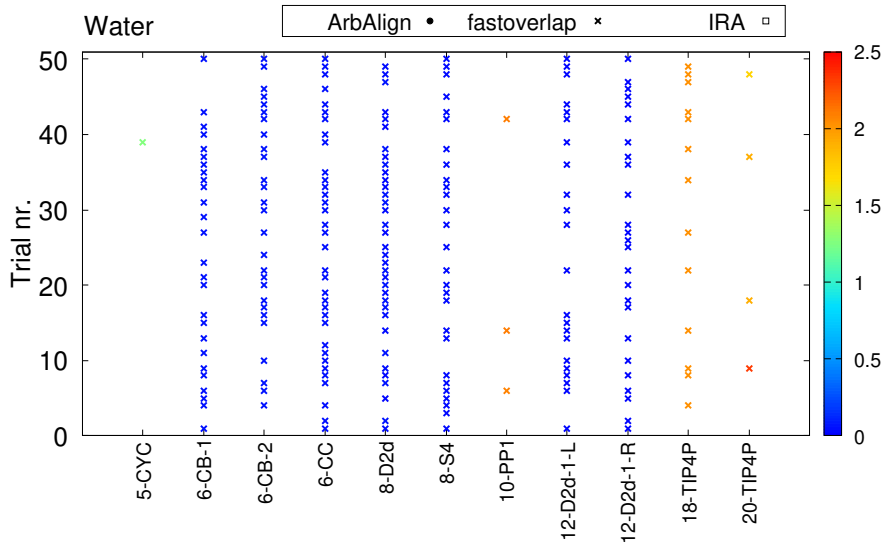



Figure S6: Values of final $RMSD$ for structures from the water dataset. Only failures are reported. Structure name on horizontal axis, trial number on vertical, final $RMSD$ value in color. Failures in this dataset: 0 failures by ArbAlign; 217 failures in 11 structures by fastoverlap; 0 failures by IRA.

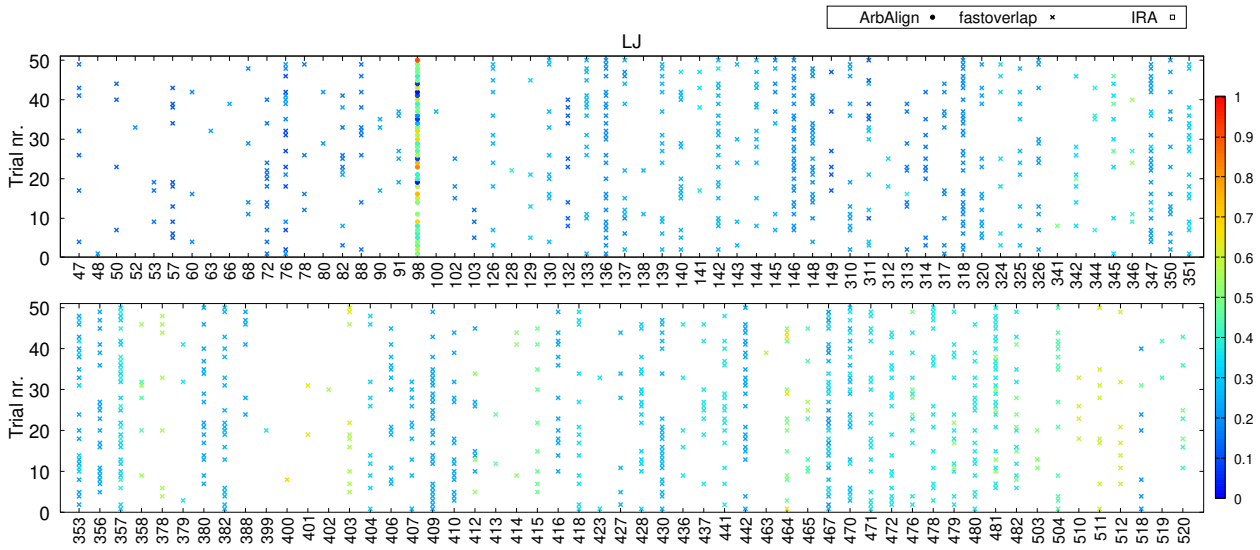



Figure S7: Values of final $RMSD$ for structures from the LJ dataset. Only failures are reported. Structure name on horizontal axis, trial number on vertical, final $RMSD$ value in color. Failures in this dataset: 45 failures in 1 structure by ArbAlign; 1177 failures in 113 structures by fastoverlap; 0 failures by IRA.

# Number of rotations tested

Fig. S8 shows the number of rotations tested for all structures in the Al dataset, versus the total number of atoms in the structure. As it can be seen, the number of rotations tested is on the range [2, 154] and there is no apparent rule. The number of tested reference frames is related to the structure surrounding the origin point as mentioned in Sec. 2.1, which in the case of non-equal number of atoms is a central atom, and in the case of equal number of atoms is the geometrical center (or any known common point). The higher number of tested rotations occurs when the geometrical center of the structure coincides with an atomic position. In that case, the distance to nearest atoms is the highest. A large number of atoms is therefore included in the radial cutoff region, such increasing the number of possible reference frames to be tested. When the geometrical center falls in between atoms, the distance to nearest neighbors is shorter (lower number of atoms), and thus less reference frames have to be tested.

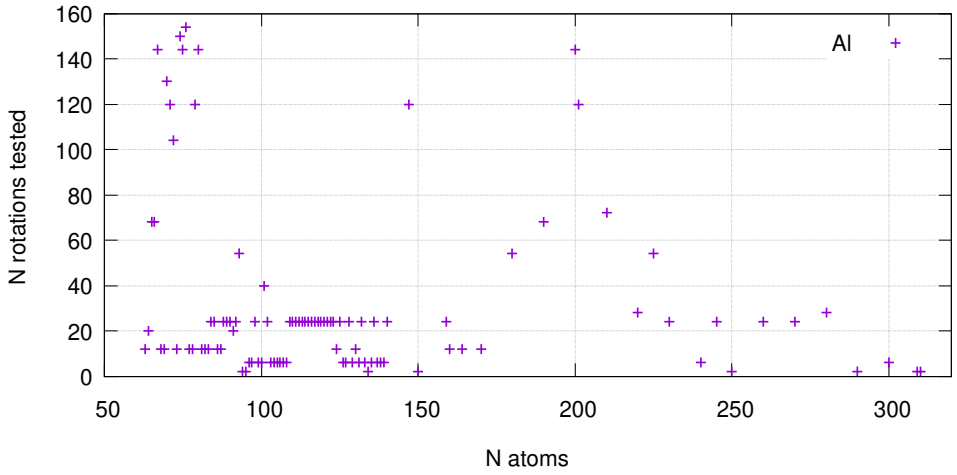



Figure S8: Number of rotations tested versus the number of atoms, for structures in the Al dataset.[1]

# References

(1) Shao, X.; Wu, X.; Cai, W. Growth Pattern of Truncated Octahedra in AlN ($N \leq 310$) Clusters. J. Phys. Chem. A **2010**, 114, 29–36, PMID: 20014801.

(2) Brena, B.; Ojamäe, L. Surface Effects and Quantum Confinement in Nanosized GaN Clusters: Theoretical Predictions. J. Phys. Chem. C **2008**, 112, 13516–13523.

(3) Liu, Q.; Xu, C.; Wu, X.; Cheng, L. Electronic Shells of a Tubular Au26 Cluster: A Cage–Cage Superatomic Molecule Based on Spherical Aromaticity. Nanoscale **2019**, 11, 13227–13232.

(4) Wales, D. J.; Doye, J. P. K.; Dullweber, A.; Hodges, M. P.; Calvo, F. Y. N. F.; Hernández-Rojas, J.; Middleton, T. F. The Cambridge Cluster Database. `https://www-wales.ch.cam.ac.uk/CCD.html`.

(5) Temelso, B.; Mabey, J. M.; Kubota, T.; Appiah-Padi, N.; Shields, G. C. ArbAlign: A Tool for Optimal Alignment of Arbitrarily Ordered Isomers Using the Kuhn–Munkres Algorithm. J. Chem. Inf. Model. **2017**, 57, 1045–1054, PMID: 28398732.