Spatiotemporal multistage consensus clustering in molecular dynamics studies of large proteins

Journal:	*Molecular BioSystems*
Manuscript ID	MB-ART-12-2015-000879.R1
Article Type:	Paper
Date Submitted by the Author:	29-Feb-2016

Complete List of Authors:	Kenn, Michael; Medizinische Universitat Wien, Center for Medical Statistics, Informatics and Intelligent Systems (CeMSIIS) Ribarics, Reiner; Medizinische Universitat Wien, Center for Medical Statistics, Informatics and Intelligent Systems (CeMSIIS) Ilieva, Nevena; B\'lgarska akademija na naukite, b. Institute of Information and Communication Technologies (IICT) Cibena, Michael; Medizinische Universitat Wien, Center for Medical Statistics, Informatics and Intelligent Systems (CeMSIIS) Karch, Rudolf; Medizinische Universitat Wien, Center for Medical Statistics, Informatics and Intelligent Systems (CeMSIIS) Schreiner, Wolfgang; Medizinische Universitat Wien, Center for Medical Statistics, Informatics and Intelligent Systems (CeMSIIS)

Journal Name

ARTICLE

Spatiotemporal multistage consensus clustering in molecular dynamics studies of large proteins

Received 00th January 20xx,
Accepted 00th January 20xx

DOI: 10.1039/x0xx00000x

www.rsc.org/

Michael Kenn[a], Reiner Ribarics[a], Nevena Ilieva[b], Michael Cibena[a], Rudolf Karch[a], Wolfgang Schreiner[a*]

The aim of this work is to find semi-rigid domains within large proteins as reference structures for fitting molecular dynamics trajectories. We propose an algorithm, multistage consensus clustering, MCC, based on minimum variation of distances between pairs of C_α-atoms as target function. The whole dataset (trajectory) is split into sub-segments. For a given sub-segment, spatial clustering is repeatedly started form different random seeds, and we adopt the specific spatial clustering with minimum target function: The process described so far is stage 1 of MCC. Then, in stage 2, the results of spatial clustering are consolidated, to arrive at domains stable over the whole dataset. We found that MCC is robust regarding the choice of parameters and yields relevant information on functional domains of the Major Histocompatibility Complex (MHC) studied in this paper: The α-helices and β-floor of the protein (MHC) proved to be most flexible and did not contribute to clusters of significant size. Three alleles of the MHC, each in complex with ABCD3 peptide and LC13 T-cell receptor (TCR), yielded different patterns of motion. Those alleles causing immunological allo-reactions showed distinct correlations of motion between parts of the peptide, the binding cleft and the complementary determining regions (CDR)-loops of the TCR. Multistage consensus clustering reflected functional differences between MHC alleles and yields a methodological basis to increase sensitivity of functional analyses of bio-molecules. Due to the generality of approach, MCC is prone to lend itself as a potent tool also for the analysis of other kinds of big data.

Introduction

Background

Rapid increase of computational speed at reasonable costs has empowered molecular dynamics (MD) as a tool for analyzing bio-molecular function at an atomistic level [1-6]. In particular, the term immuno-informatics [7-10i] has been coined by V. Brusic [8] and aims at scrutinizing the mechanisms of immuno-recognition by studying the interplay between loaded peptides, Major Histocompatibility Complex (MHC)-molecules and T-cell-receptors (TCRs) [11-16] in atomistic detail [17;18]. Intricate statistical procedures, aiming at isolating patterns of motion underlying molecular functions [19;20], have to draw on some kind of reference within an overall moving, distorting and flexible biomolecule [21-23].

To these ends, newly produced MD data are usually subjected to a so-called fitting procedure [24]. One configuration (reference frame) is selected (usually the first frame) and each other frame of the remaining trajectory is transformed (shifted and rotated) so as to fit to the reference frame in a least-squares sense [25]. Such a fitting procedure removes absolute movements while retaining relative movements and deformations. Atom coordinates preprocessed in this way lend themselves for further sophisticated analyses [26-28], aiming at extracting patterns of motion actually relevant for the molecular mechanism in question.

Obviously, these results critically rely on the fitting procedure, in particular on the choice of the reference structure. One question is which frame to select. A second question is an appropriate selection of the fitting domain, i.e. those parts of the molecule for which root mean squared displacement (RMSD)-deviations are computed and minimized.

Each atom within the fitting domain clearly influences the results of fitting. Even small flexible domains easily create large RMSD-values, and thereby might dominate the fitting ('the tail wags the dog'). Hence, one tries to exclude very flexible parts of the molecule from fitting and to resort to semi-rigid domains as reference structures.

Finding such structures by spatial clustering has been described previously [29].

In the present work we use the analysis of MD data as a specific example to prove the concept and performance of spatiotemporal clustering, while at the same time extending and generalizing its methodology towards multistep consensus clustering (MCC).

Motivation for the new approach presented here

Shortcomings of spatial clustering are

[a] Section of Biosimulation and Bioinformatics
Center for Medical Statistics, Informatics and Intelligent Systems (CeMSIIS)
Medical University of Vienna, Spitalgasse 23, A-1090 Vienna, Austria.
[b] Institute of Information and Communication Technologies (IICT),
Bulgarian Academy of Sciences, 25A, Acad. G. Bonchev Str., Sofia 1113, Bulgaria
* Correspondence should be addressed to Wolfgang Schreiner.

- the arbitrariness in selecting the number of clusters and
- open questions regarding stability of the results: would another (or elongated) trajectory produce similar clusters?

In this work we solve both issues by proposing the novel two-stage procedure of *spatiotemporal clustering*:

- Spatial clustering is performed on parts of a trajectory.
- Resulting spatial clusters are further subjected to temporal clustering, which reduces them to semi-rigid domains stable over time (cluster consolidation). In this step, a criterion for stability guides the clustering process.

The new procedure of spatiotemporal clustering mends two shortcomings (presetting the number of clusters, lacking information on stability of clusters) of our previous approach by introducing the new concept of 'cluster consolidation' to obtain 'domains'.

Material and Methods

Molecules

MD simulations were performed on two biomolecular complexes, one of these (TCR/peptide-MHC = TCR/pMHC: "molecule 1") was a substructure of the other (TCR/pMHC/CD8-coreceptor = TCR/pMHC/CD8: "molecule 2"). This allows assessing essential features regarding the quality, stability and fidelity of clustering:

1. Performance of spatiotemporal clustering for molecules of different size
2. Check if the clustering within a substructure remains stable if the substructure becomes part of a larger complex

In addition, molecule 1 was investigated in three alleles.

Three alleles of TCR/pMHC complexes

Spatiotemporal clustering was applied to MD-simulations of three molecular systems that have already been described in a previous paper [29]. The molecular complex of HLA-B*4405, ABCD3 peptide and LC13 T-cell receptor [30] is labelled B4405 in this work. Two other complexes, HLA-B*44:02 and HLA-B*44:03 each with ABCD3 peptide and LC13 TCR (B4402 and B4403, respectively) were obtained using B4405 as a homology model and introducing the following mutations:

$$\text{B4405} \xrightarrow{\text{Y116D}} \text{B4402} \xrightarrow{\text{D156L}} \text{B4403} \qquad (1)$$

Elements of secondary structure are located within the complex as shown in Table 1, based on consecutive numbering of Cα atoms.

Structural elements are given in terms of consecutive numbers of C_α atoms, renumbered throughout the whole modelled TCR/pMHC/CD8 complex, as if the complex as a whole were taken from one single PDB-file, see also the following section on modelling. The same data apply to all three complexes (B4405, B4402, B4403).

Table 1: Molecules and their secondary structural elements

		Chain	Type	Length in C_α	C_α index
Molecule 2: TCR/pMHC/CD8	Molecule 1: TCR/pMHC	Chain A	**MHC**	276	1-276
		Chain B	**MHC, β_2 -microglobulin**	99	277-375
		Chain C	**Peptide**	9	376-384
		Chain D	**TCR, α chain**	201	385-585
		Chain E	**TCR, β chain**	241	586-826
		Chain F	**CD8 alpha$_1$**	114	827-940
		Chain G	**CD8 alpha$_2$**	114	941-1054

Secondary structures	Chain	Length in C_α	C_α index
Alpha helix Gα_1	A	25	59-83
Alpha helix Gα_2	A	31	141-170
Domain α_3	A	92	184-275
Beta-sheet	A	52	2-13, 21-29, 30-37, 93-103, 110-118, 124-127

Molecular modelling of TCR/pMHC/CD8 complex

The molecular structures of many co-crystallized TCR/pMHC have been resolved so far (www.pdb.org). Molecular structures of CD8 co-receptors bound to MHC are also available (e.g. PDB ID 1AKJ). However, a TCR/pMHC/CD8 complex has not been co-crystallized so far to our knowledge. To model such a structure on the computer we first localized the binding site of CD8 on MHC in the 1AKJ crystal structure. We defined all C_α atoms of the MHC within the range of 0.8 nm to CD8 to belong to the MHC binding site. In the next step, structures of TCR/pMHC and MHC/CD8 were merged into one file and the MHC binding sites were superimposed so as to minimize RMSD in a least-squares sense. The MHC molecule from the MHC/CD8 complex was deleted and the resulting TCR/pMHC/CD8 used for MD simulation. This procedure was

repeatedly done for TCR/pMHC complexes B4402, B4403 and B4405.

Target function for semi-rigidity

As previously explained [29] we base spatial clustering on the STDDV-matrix S_{ij} holding the standard deviations of distances d_{ij} between atom pairs (i,j) along a given trajectory:

$$S_{ij} = \sqrt{\frac{L}{L-1}\left\langle \left(d_{ij} - \langle d_{ij} \rangle\right)^2 \right\rangle} \qquad (2)$$

where L is the number of frames (configurations) considered and $\langle\ \rangle$ denotes the average over a trajectory (or a part of it). As opposed to absolute atom-coordinates, pair-distances offer the advantage that they are defined relatively within the molecule, rendering fitting unnecessary. Bernhard and Noé [31] have proposed a decomposition of N atoms (c_{im} is the membership of atom i in cluster m) into a given number of k clusters by minimizing the target function

$$q(\mathbf{c}) = \sum_{m=1}^{k}\sum_{i=1}^{N}\sum_{j=1}^{N} c_{im}\, c_{jm}\, S_{ij} = tr(\mathbf{c}^{\mathrm{T}}\mathbf{S}\mathbf{c}) \rightarrow \min \qquad (3)$$

We have previously shown that these class memberships have to be crisp $\{0,1\}$ [29]. Based on that we developed a corresponding algorithm offering maximum computational speed, see the section on benchmarks.

Clustering algorithms

The problem of minimizing $q(\mathbf{c}) = tr(\mathbf{c}^{\mathrm{T}}\mathbf{S}\mathbf{c})$ is NP-complete [32]. Thus, under the assumption of $P \neq NP$, no algorithm can exist that would be capable of finding the minimum efficiently after a number of operations being bounded by a polynomial in N. To illustrate this, we first consider the case $k = 2$ of two (distinct) clusters C_1 and C_2, and note that

$$q(\mathbf{c}) = \sum_{i,j\in C_1} S_{ij} + \sum_{i,j\in C_2} S_{ij} \qquad (4)$$

Instead of minimizing $q(\mathbf{c})$ one can alternatively maximize

$$\bar{q}(\mathbf{c}) = \sum_{i\in C_1, j\in C_2} S_{ij} \qquad (5)$$

We introduce $x_i = 1$ for $i \in C_1$ and $x_i = -1$ for $i \in C_2$, ($i = 1,2,.....n$) and obtain after some calculations

$$\bar{q}(\mathbf{c}) = \sum_{i\in C_1, j\in C_2} S_{ij} = \frac{1}{2}\mathbf{x}^{T}\mathbf{L}\mathbf{x} \qquad (6)$$

with the Laplacian matrix $\mathbf{L} = \mathbf{D} - \mathbf{S}$ and $\mathbf{D}$ being the diagonal matrix with entries

$$d_{jj} = \sum_i S_{ij} \qquad (7)$$

Finding $\mathrm{argmax}\left(\mathbf{x}^{T}\mathbf{L}\mathbf{x}\right)$ is the well-known weighted max-cut problem [32]. In our case the entries of matrix $\mathbf{S}$ are real numbers, $S_{ij} \in \mathbf{R}$. In the literature, a much simpler and special case of finding the solution $\hat{x} = \underset{x\in\{-1,1\}^n}{\mathrm{argmax}}\,\bar{q}(\mathbf{x})$ for $k = 2$ and a binary matrix, $S_{ij} \in \{0,1\}$, has been described and labelled the 'maximum cut'. Even for this simpler case, the decision problem whether there is a solution better than a given vector $\hat{x}$ is a reduction from the nae3sat (Not–All–Equal–3-SAT) problem, already treated by Cook [33]. Thus, we know that a solution exists - but this is only of theoretical value, since it could take exceedingly long computational time to determine it. Given these characteristics of the simplified, binary case of **S**, finding the solution will be even harder in our case of real-valued target function, $S_{ij} \in \mathbf{R}$ and $k > 2$. As a consequence, the number of C_α atoms within a protein is of a size that requires a highly optimized approach for clustering, as described earlier [29].

Spatial clustering

The easiest method of clustering would be agglomerative hierarchical clustering (AHC), for which numerous methods have been proposed [34;35]. However, AHC only works satisfactorily if shifting one atom from its cluster into another cluster, worsens the target function significantly. However, this is not the case with our MD data (as pre-evaluations revealed): A considerable number of atoms may change clusters without significantly worsening the target function. Hence, as an alternative to AHC we propose a greedy, nested, 3-step algorithm to cluster N atoms into k clusters according to the target function $q(\mathbf{c})$, see (4), as follows:

1. First we choose a preliminary number of clusters, k. In chapter 'Optimizing the number of spatial clusters' we describe how to choose the optimum k.
2. Then we randomly assign each atom to one of the k clusters.
3. Lap of single-atom moves: For each of the N atoms we try if a move to one of the other (k-1) clusters would be beneficial, i.e. decrease the target function (N(k-1) trials). If a move is beneficial, the new cluster assignment of the moved atom is accepted and retained throughout the trials of the remaining atoms.
4. The lap of single atom moves (step 3) is repeated (N(k-1) trials in each repetition) until not a single beneficial move occurs within a whole lap (exhaustive search).
5. Steps 2-4 are repeated until m-times in succession no better solution is found. Please see below on how to select m.
6. Localization of ground state (optional). To obtain the ground state, a variant of the Metropolis algorithm [36] is applied. In the current implementation, a suitable set of atoms, approximately $\sqrt{4N}$, is selected, which have least influences on the target function (when changing clusters individually). This group of atoms is spotted during steps 3 and 4. These atoms are then randomly redistributed among all clusters. Starting from this new cluster assignment, step 6 is repeated until no further improvement is observed over a preselected number of iterations (100 times proved sufficient by far).

We have conducted pilot studies to estimate the success probability and overall performance of the algorithm and render human choice of parameters unnecessary:

a) Above steps 2-5 lead to the global optimum of $q(\mathbf{c})$ in approximately 0.5% ($p = 0.005$) of the random assignments (for 826 atoms used in the pilot study). Thus, when performing the procedure m times, there is a probability of $\varepsilon = (1-p)^m$ to miss the optimum. If we are satisfied with a failure probability of $\varepsilon < 0.01$, 918.7 is a sufficient number of repetitions, and thus choosing $m = 1000$ should be safe.

b) In step 3 only single atom moves are performed and one might speculate if joint moves of several atoms (step 6) could further optimize $q(\mathbf{c})$. We have evaluated this issue and found that moving groups of atoms (instead of single atoms) adds only little benefit and always pertains to atoms without distinct cluster membership (those which are weakly bound to clusters and do not worsen $q(\mathbf{c})$ significantly when changing clusters). Such atoms are, however, exactly those being eliminated from clusters by temporal clustering, see below.

Thus, if we would perform nothing but spatial clustering, the additional step 6 is recommended. However, if spatial clustering is followed by temporal clustering, step 6 can well be omitted (what we actually did).

Benchmarks for spatial clustering

Table 2 gives benchmarks in seconds for two systems with $N = 117$ and $N = 826$ C_α atoms, respectively. We have deployed the clustering tool on the distance variation matrices obtained from MD-trajectories of the complexes B4402, B4403 and B4405.

Table 2: Benchmarks for the spatial clustering algorithm.

Number of clusters *k*	*N* = 117	*N* = 826
4	0.11 ± 0.03	5.2 ± 1.0
5	0.13 ± 0.03	7.0 ± 1.7
6	0.16 ± 0.05	9.0 ± 3.2
7	0.25 ±0.07	10.7 ± 3.2
8	0.25 ± 0.09	12.7 ± 4.5

Execution times in seconds on an Intel Core i7-2600, 3.40 GHz, 4 GB memory. The molecules consist of 8706 atoms (B4402 and B4405) and 8715 atoms (B4403), out of which 826 are C_α–atoms.

Critical review of spatial clustering

Fig. 1 shows the circular-plots (usually used for depicting e.g. genomic rearrangements) for the B4405 trajectory, preselecting two different numbers of clusters (8 and 16).

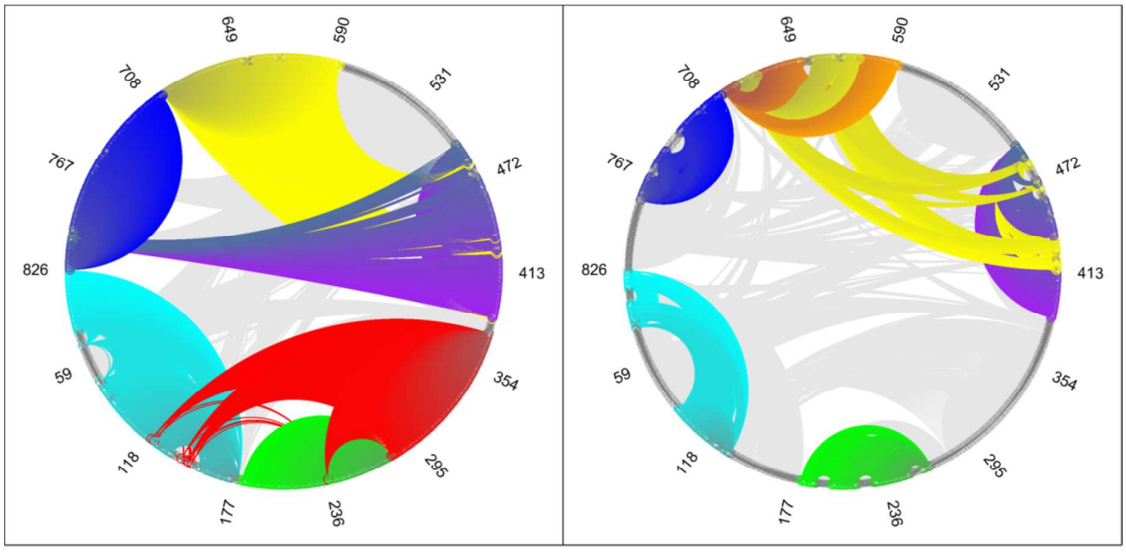

Fig. 1: Spatial clustering for trajectory B4405. Different numbers of clusters have been preset ($k = 8$, 16). Each circular-plot (see the reference to software in the acknowledgements) shows the number of C_α counter-clockwise along the circumference (numbers in steps 1, 59, 118…826). The color of the circumference encodes if the respective C_α is part of the cluster (shown as colored dot) or if it is not part of the cluster (grey dot). Colors are assigned as explained later (section 'Displaying clustering results for different conditions'). Each pair of Cα atoms belonging to the same cluster is connected via a curved line, colored according to the start C_α. Links between pairs of Cαs are shown in grey if both C_αs are members of a smaller cluster (note: only the largest clusters, in this case 6, are considered). C_αs which are shown in grey along the circumference without giving rise to a link to any other C_α do not belong to any cluster. The total number of connections declines with increasing number of clusters, k.

When inspecting Fig. 1 one observes:

1. The magnitudes of pair-distance variations, S_{ij}, are not reflected in the circular-plot: Weak connections between atoms (i.e. large distance variations) and strong connections appear similar for atoms within the same cluster.
2. When presetting different numbers of clusters, *k*, spatial clustering yields different results.
3. Even presetting a moderate number of clusters, the circular-plot appears crammed with connections, see Fig. 1. The reason is that spatial clustering assigns each C_α-atom to exactly one cluster – even if this C_α is not actually part of a rigid domain. Moreover, all atoms within each cluster are mutually connected, as reflected in the resulting membership matrix, $c_{ij} \in \{0,1\}$. One may think that aiming at smaller clusters, e.g. by increasing the number of clusters, k, might avoid some of the above disadvantages. However, we have to bear in mind:
4. The target function, Eq. (4), tends to yield clusters of similar size, thus introducing a possible bias regarding cluster size.
5. If one selects *k* clusters and they result approximately equal in size, the number of atom-atom connections within all *k* clusters declines approximately as $1/k$ for the following reason: Each cluster will contain about N/k atoms, giving rise to $N/k \cdot (N/k-1)/2$ connections within each cluster. All clusters together will then contain

$$\frac{k \cdot N/k \cdot (N/k-1) \cdot 2}{N \cdot (N-1) \cdot 2} = \frac{N-k}{k \cdot (N-1)} \approx \frac{1}{k} \qquad (8)$$

   connections, the approximation $1/k$ being valid for $k \ll N$. This entails a fairly large number of connections, e.g. $1/8 \approx 0.125 = 12.5\%$ of all possible connections for $k = 8$ clusters.
6. We found that increasing *k* to an extent that would sufficiently reduce graph size, renders even the larger clusters so small that they no longer lend themselves as a basis for RMSD-fitting the frames of an MD-trajectory, see the introduction.

Hence, increasing *k* cannot provide a remedy since it is in fact not a valid substitute for getting rid of exceedingly flexible atoms, being the major source of the above shortcomings. Instead, to mend shortcomings of spatial clustering, we drew from another, quite obvious fact: Different clusters are obtained for time-wise successive parts of a trajectory. How do they relate to the clusters obtained for the trajectory as a whole? Answering these questions provided the key for developing temporal (time-wise) clustering, see the following chapter.

Temporal clustering

Spatial clustering, as described above, yields solutions optimized for a given MD trajectory; in case of Fig. 1 it was the whole trajectory for B4405. However, if spatial clustering is applied to parts of a trajectory (in the following called segments), different clusters may result for each segment. This time-wise variability of clustering may be considered a drawback regarding the reliability of clustering, but it offers at the same time an opportunity to improve clustering quality. With this aim in mind we devised an algorithm capable of excluding the most flexible atoms, which otherwise might preclude any clustering stable over time.

Given the trajectory of a molecule with *N* atoms we search for domains with maximum spatiotemporal stability. The trajectory (comprising, e.g., *L* = 4000 frames) is divided into *F* temporal segments of *F'* frames each (typical numbers are *F* = 100 and *F'* = *L*/*F* = 40). For each segment we compute the STDDV matrix, see (2), with *i,j* = 1,..., *N*, yielding STDDV-matrices $\mathbf{S}^{(f)}$, with *f* = 1,..., *F*.

Let $\mathbf{A} = \{A_1, A_2, ..., A_N\}$ be the set of atoms in the molecule considered for the stability analysis. Based on the STDDV matrix $S^{(f)}$ of each segment, the spatial clustering algorithm (described above) generates *k* disjunct clusters. Thus $C_k^{(f)} = \{A_{k1}, A_{k2}, ...\}$ with $k = 1,..., K$ and $\mathbf{A} = \bigcup_{k=1}^{K} C_k^{(f)}$ resp. $C_i^{(f)} \cap C_j^{(f)} = \varnothing$ for $i \neq j$.

We now compare the *F* results of clustering and focus on an arbitrary pair of atoms, A_i and A_j. We define the dissimilarity between these two atoms by

$$\Delta_{ij} = \frac{1}{F} \sum_{f=1}^{F} \Delta_{ij}^{(f)} \qquad (9)$$

with $\Delta_{ij}^{(f)} = 0$ if A_i and A_j belong to the same cluster $C_k^{(f)}$ in segment *f* and $\Delta_{ij}^{(f)} = 1$ otherwise. The resulting dissimilarity matrix $\mathbf{\Delta} = (\Delta_{ij})$ therefore consists of integers between 0 and *F*. We define a threshold Δ_{th} and construct an adjacency matrix $\mathbf{\Delta}'$ of the graph *G*(*V*, *E*) by setting $\Delta'_{ij} = 0$ if $\Delta_{ij} \geq \Delta_{th}$ and $\Delta'_{ij} = 1$ otherwise. By definition, $\Delta'_{ii} = 0$. The set *V* of vertices of *G*(*V*, *E*) corresponds to the *N* atoms, and the set *E* of edges is represented by pairs of atoms with low dissimilarity.

The threshold Δ_{th} has to be chosen wisely. It should be large enough to retain sufficient information about correlations between atoms – but still small enough to distinguish between groups of atoms. The choice of Δ_{th} can be guided by posing an upper limit on the size of the graph (number of edges) as compared to the complete graph $|E(G)| \leq p \cdot N(N-1)/2$, with the percentage p being selected typically between 0.0 and 0.07. Fig. 2 shows how different thresholds Δ_{th} change the shapes and relative sizes of graphs obtained for N = 826 atoms, k = 7 clusters, L = 4000 MD-frames, F = 100 trajectory segments, F' = 40 frames within each segment, with N(N-1)/2 = 340.725 being the complete graph size.

The dependence of graph size on p will play a central role in selecting the optimum number of clusters, *k*, for spatial clustering, see also Fig. 4.

Journal Name

ARTICLE

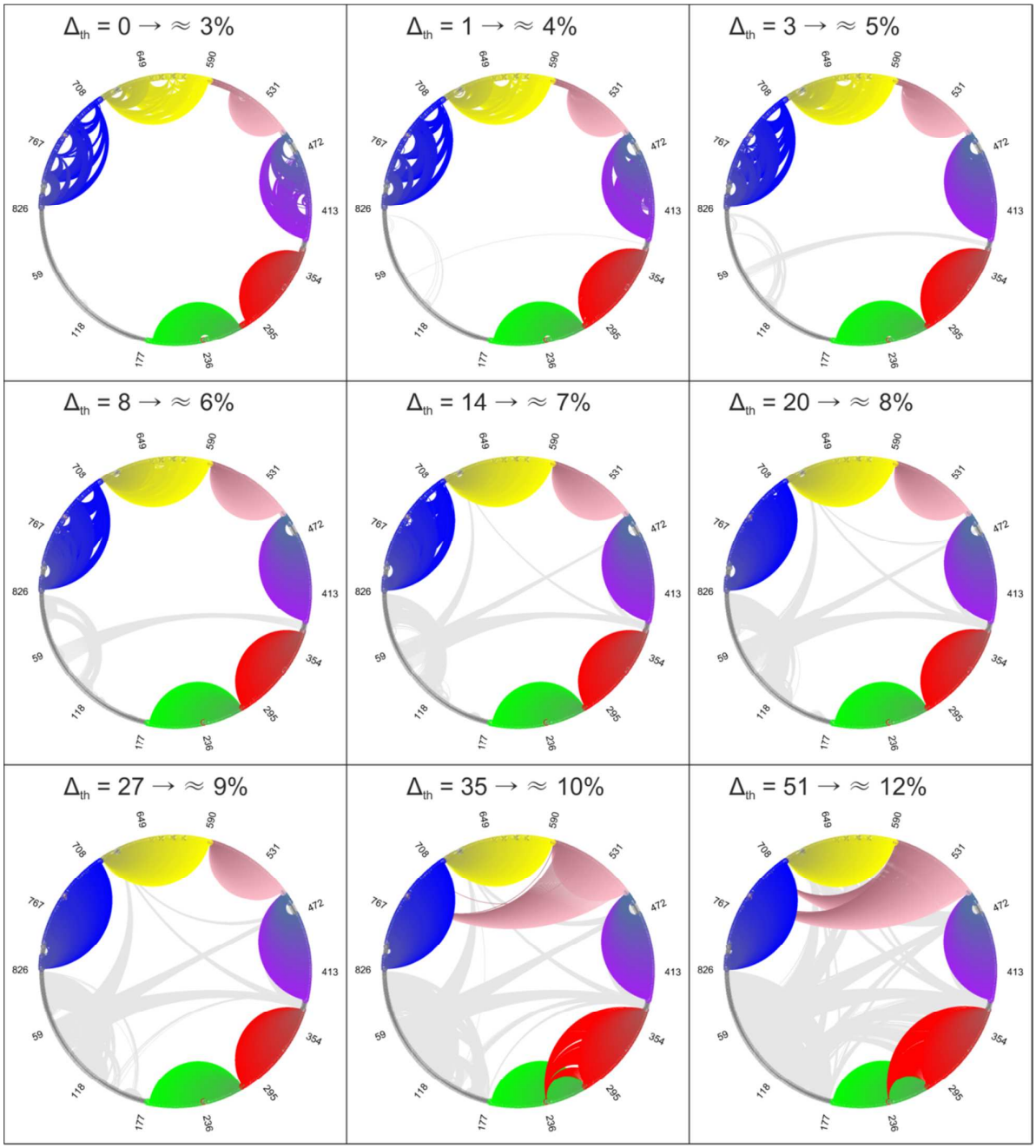

Fig. 2: Time-wise consolidating spatial clusters for different thresholds Δ_{th}. Spatial clustering was performed for $k = 7$ clusters. Despite different thresholds Δ_{th} for temporal cluster consolidation, almost the same large domains result. Incidentally, using $k = 8$ spatial clusters yields almost identical results, since these are finally condensed to 7 domains. Parameters: $N = 826$ atoms, $L = 4000$ MD-frames, $F = 100$ trajectory segments, $F' = 40$ frames within each segment. The size of the complete graph would be N(N-1)/2 = 340.725 edges.

Journal Name

ARTICLE

The essential step is now to cluster the graph $G(V,E)$, which was done via two established clustering methods:

- single-link clustering
- complete-link clustering

A detailed comparison was summarized by Jain [37]. We note that single-link clustering is equivalent to finding sub-graphs of maximal order (i.e. maximum number of vertices) and is solvable in computing time of polynomial order of the size of the graph. As opposed to this, complete-link clustering means iteratively looking for cliques of maximal order (cliques are complete subgraphs, all vertices connected). It is well known, however, that the cliques problem is computationally exceedingly demanding (NP-complete [32]) and may, for instance, be tackled with the Bron-Kerbosch algorithm [38].

For this reason we performed single-link clustering first and inspected the results. To our surprise, the majority of atoms found in clusters was not only single linked but additionally fulfilled the much more stringent criterion of complete linkage, even if some dissimilarity (threshold $\Delta_{th} > 0$) was allowed. Even more, allowing for zero tolerance ($\Delta_{th} = 0$), results for single-link and complete-link proved to be identical. This renders (the computationally demanding) complete-link clustering unnecessary, and we end up with the methodological finding that single-link clustering suffices for our MD data if we go for really rigid clusters (zero tolerance) over time.

For temporal clustering the total number of clusters results from clustering itself rather than being preset (as was necessary with spatial clustering). In fact, temporal clustering yields clusters of different sizes – a few large ones and more and more small ones. We consider only the larger clusters above a certain (case-dependent) size and call them 'domains' within the molecule. Each domain may be colored in a circular-plot, to visualize areas of coherent motion, see the figures in the results-section.

Those many smaller clusters (down to clusters of single atoms) are assigned to a joint pool of 'mobile groups' and shown in (the no-color) grey in the circular-plots.

Displaying clustering results for different conditions

Clusters resulting from unsupervised procedures (such as spatiotemporal clustering) are labelled automatically, e.g. according to decreasing size. This is fine for considering one single run of clustering. However, when comparing results from several runs of clustering, labelling needs special attention in order to arrive at results which are nicely comparable. For example, when displaying and coloring clusters within molecular 3D-plots or circular-plots (see below), we expect that clusters with equal labels appear (in each plots of a series) around the same 'locations', regarding 3D coordinates or residue number, respectively. This is not guaranteed with labeling according to size (see histograms in Fig. 5, second row). Given two clusters almost equal in size, either one may result larger in a certain condition. As a consequence both may interchange labels (and colors) from condition to condition.

We devised a procedure to keep labelling consistent. Given a set of clustering results, say all those shown in circular-plots within this work. First we choose the number of clusters to label and color (in our case $l = 6$) in each plot. Next, in each plot we label (color) the l largest clusters according to decreasing size, and assign the no-color (grey) to all other clusters. Then we let the researcher pick one plot of his like and take this as a reference (in our case $k = 9$, with permuted selection of frames). Then, in all other plots we re-label the l largest clusters as follows: We determine that cluster of the reference with maximum overlap (e.g. according to residue numbers within the circular-plot), and re-label it according to the reference. This makes labels (colors) appear at similar locations in all plots (including the reference), allowing for easy, intuitive comparisons. Re-labelling tidies circular-plots, bar charts as well as 3D-visualisations at the same time, see Fig. 5.

Note the special case when one of the l largest clusters in one of the plots does not coincide with one of the l largest clusters in the reference. In this case additional labels (2.a, 2.b, etc. and respective colors) have to be introduced. In Fig. 3, this occurred in the plots for $k = 6$ (hist and perm), $k = 9$ (perm) and $k = 10$ (hist).

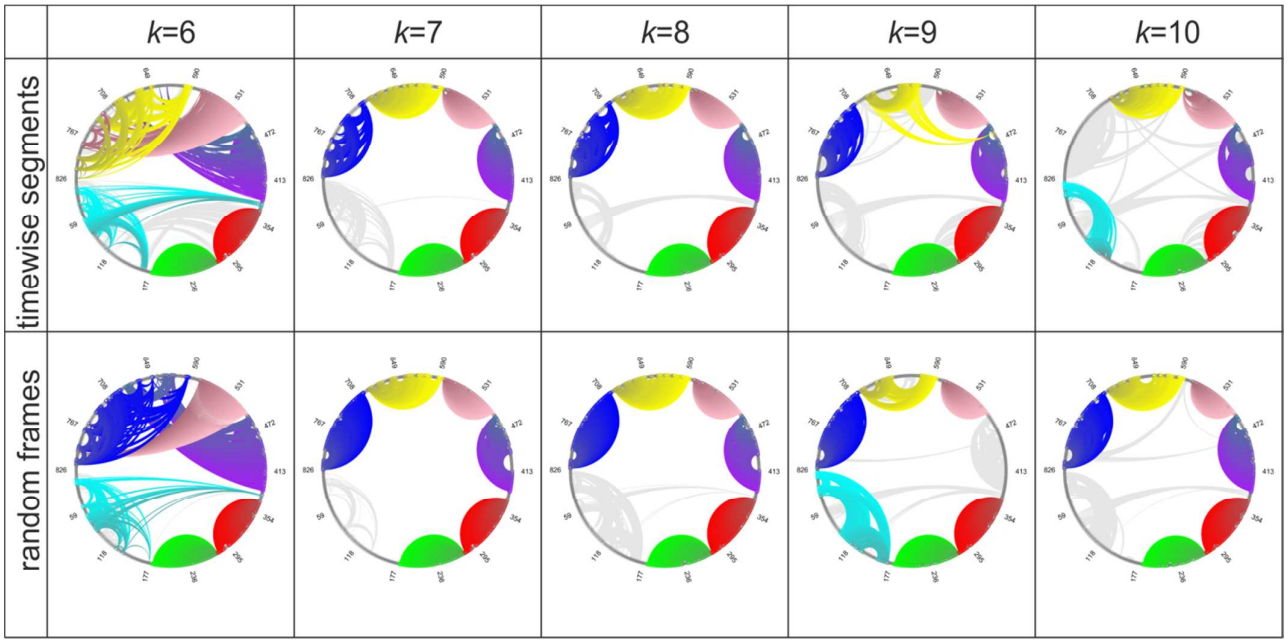

Fig. 3: Adjacency matrix depends on the number of spatial clusters but is insensitive to mode of sampling. Sampling from time-wise segments (upper row) and from random permutation of MD-frames (bottom row) shows almost equal results. Selecting 7 or 8 spatial clusters yields most consistent results for both, time-wise segments and random sampling of frame-segments. Threshold was set to 6%. Note that clusters have been colored according to section 'Displaying clustering results for different conditions'.

Results

Dependence of TCR/pMHC on parameter settings

The algorithm described above contains several free parameters. Below, we compare the results for various parameter choices and study the stability of this newly proposed spatiotemporal clustering procedure.

Number of spatial clusters

The first free parameter is the number of clusters, *k*, for spatial clustering. The columns in Fig. 3 show results for $6 \le k \le 10$ for the B4405 trajectory (consult Fig. 7 for secondary structure and its labelling). One cluster of atoms (labelled α_3, shown in light green) is identically grouped, regardless of *k*. Moreover, it is completely self-contained (solid), i.e. all atoms within the cluster seem tightly connected, and not a single connection lurks outside. Six other clusters, nearly as stable and self-contained as α_3, emerge — almost identical — for $k = 7$ and $k = 8$. On the contrary, the remaining parts of 'Chain A (including β-sheet)' (see Table 1), including both alpha-helices Gα_1 and Gα_2, appear poorly self-contained regardless of *k*, indicating significant internal motion. In addition, distinct connections point towards the 'peptide', indicating joint motion.

Random sampling of MD-frames

As an alternative to blocking $F' = 40$ successive frames into 100 timewise consecutive segments, we drew groups of 40 frames at random out of the whole trajectory and considered them as a random segment. We drew 1000 such random segments (much more than those 100 timewise consecutive ones) and clustered each of them (first spatially, then temporally). Selection of threshold, computation of adjacency matrix $\mathbf{\Delta}'$ and clustering of the graph $G(V, E)$ was performed following the same protocol applied for consecutive segments. Expectedly, due to the much higher number of segments (1000 instead of 100), many graphs of intermediate sizes are found, and the elements of the dissimilarity matrix $\mathbf{\Delta} = (\Delta_{ij})$ show many more different values (as compared to blocking into 40 successive segments).

The results of random selection (bottom row in Fig. 3) and 100 timewise segments are very similar. This suggests that our primary choice of 100 timewise segments fully suffices and can be adopted. We have also tested different numbers of permutations, which confirmed the results obtained for 1000 samples.

Different thresholds

Selecting the threshold Δ_{th} changes the adjacency matrix and the sizes of graphs in relation to that of the full graph, see Fig. 4. These results can be used to first select a relative size of the graph, say 6% (0.06) and then adapt the threshold Δ_{th} accordingly (so as to induce a ratio $p \approx 6$ %).

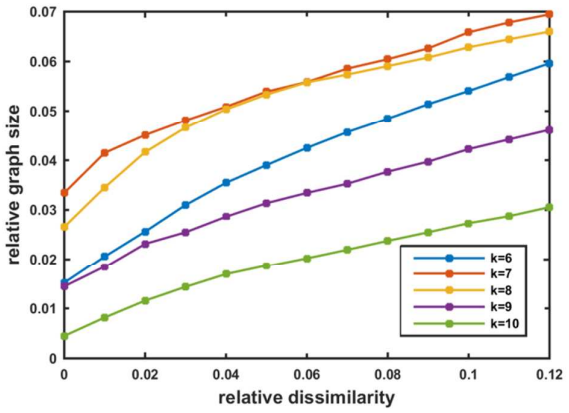

Fig. 4: Relative size of graph depends on threshold Δ_{th} for dissimilarity within groups. Increasing the threshold, Δ_{th}, for absolute dissimilarity allows two atoms to belong more often to different spatial clusters in some of the trajectory segments while still counting for the very same domain consolidated by temporal clustering. Increasing Δ_{th} means increasing relative dissimilarity of clusters (x-axis). As a consequence graph size (= number of edges, y-axis) also increases. Results refer to the following parameters: N = 826 atoms, k = 6,7..10 clusters, L = 4000 MD-frames, F = 100 trajectory segments, F' = 40 frames within each segment, N(N-1)/2 = 340.725 being the complete graph size. A separate analysis was performed for each of the preset number of spatial clusters, k = 6, 7,…10, see legend. Generally, the density of the adjacency matrix increases as the criterion for dissimilarity is relaxed (increasing trend of all curves). Increasing the number of clusters k the series of curves passes through a maximum (highest density) for k =7 or 8, marking the optimum choice for k.

Optimizing the number of spatial clusters

Spatial clustering on its own does not offer a criterion for the optimum choice for the number of clusters, k. Temporal clustering, however, provides a sound criterion as follows.

Small k during spatial clustering induces larger clusters and increases the number of atoms belonging only vaguely to their assigned cluster: they easily swap clusters, see section 'Critical review of spatial clustering'. With larger k (i.e. choosing more spatial clusters), the number of swapping atoms declines and the adjacency matrix becomes increasingly populated (becomes more dense), and curves in Fig. 4 are shifted upwards. For $k \geq 9$, however, this trend is reversed: Many small clusters increasingly fluctuate between segments of the trajectory, thus diminishing the sets of atoms constantly within the same clusters (curves in Fig. 4 are shifted downwards).

As a result, temporal clustering provides us with a sound estimate to select 7 or 8 spatial clusters as an optimum, see Fig. 4.

Different alleles of TCR/pMHC

Clustering as described above was carried out on MD trajectories for each of the three different TCR – peptide – MHC (TCR/pMHC) complexes B4405, B4402 and B4403, using the parameters k = 7 clusters, a threshold yielding $p \approx 6$ % and 100 time-wise segments of 40 frames each. Resulting circular-plots look very similar, see Fig. 5, first row. In order to scrutinize clusters in more detail we also present bar-charts, for each atom giving the size of the cluster it belongs to, cf. Fig. 5, second row. The spatial location of these clusters within the molecular complex are shown in Fig. 5 third row.

Journal Name

ARTICLE

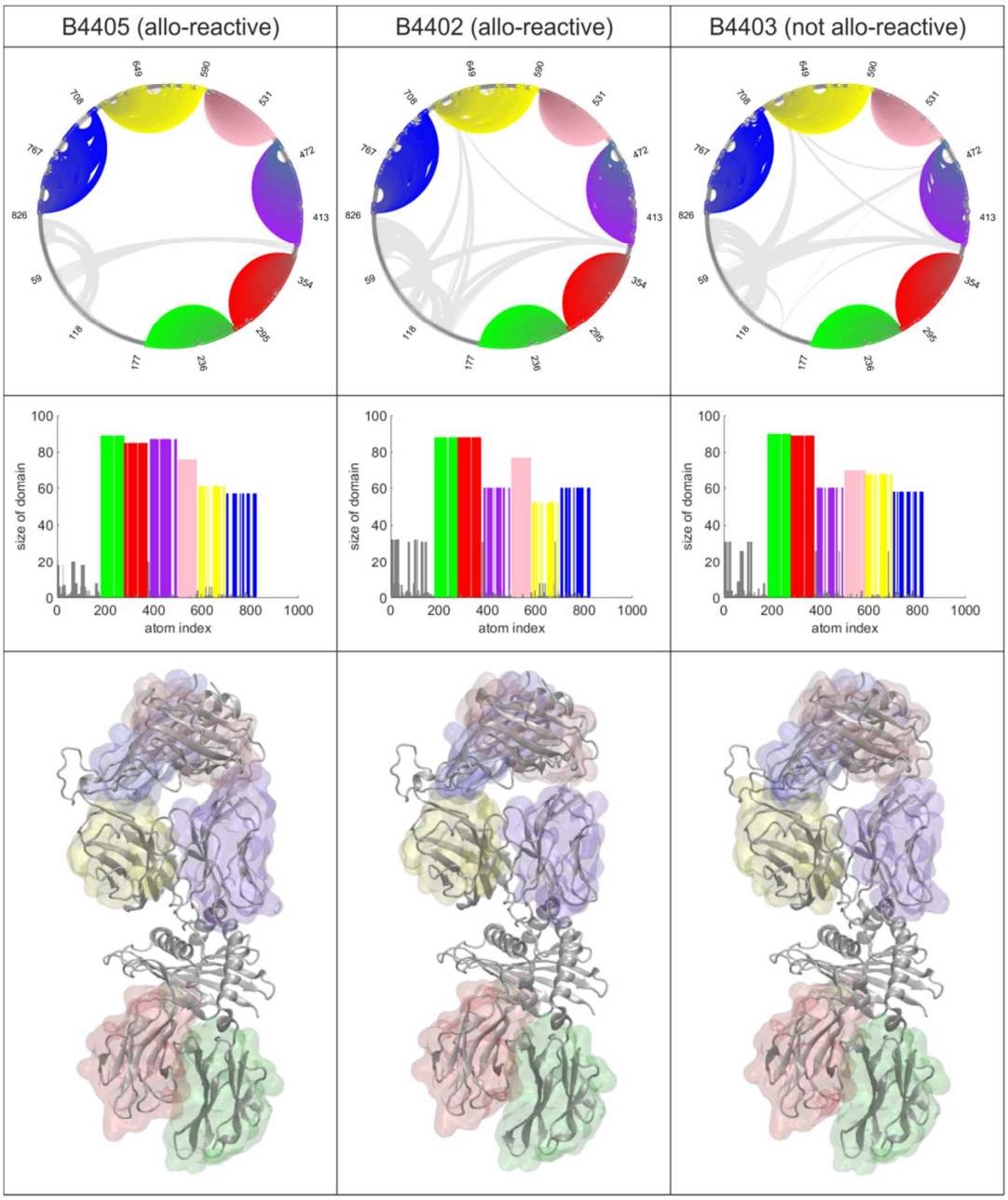

Fig. 5: Spatiotemporal clustering for three MHC-alleles B4405 (left), B4402 (middle) and B4403 (right). Spatiotemporal clustering recognized the membrane-facing domain of the MHC, the β_2-microglobulin and the whole TCR as semi-rigid. All semi-rigid domains comprise beta-sheets that are inherently stable protein secondary structures due to their geometry allowing nearly optimal hydrogen bonding. Interestingly, the MHC beta-floor and alpha-helices at the protein-protein interface as well as CDR3 are not found in the larger clusters and thus are more flexible. The six largest domains (1 to 6) are shown in color, all other (smaller) domains are shown in grey. Preceeding spatial clustering performed with $k = 7$. Row 1: Domains mapped on circular-plots. Row 2: Bar charts showing sizes of domains and their positions within the protein chain, according to Cα-index. Row 3: Domains mapped on a 3D display (VMD) of the molecule.

Journal Name

ARTICLE

Domains considered representative for all three alleles of TCR/pMHC

As a main result we observe that similar clusters result for all three alleles, see Fig. 5. In the following, these are described in more detail with reference to the circular-plot and 3D visualization in Fig. 7.

MHC occurs as an α-chain that is composed of three domains, α_1, α_2, and α_3. The α_1-domain rests upon the β_2-microglobulin (chain B) and, together with the α_2-domain, forms the antigen-presenting interface to the TCR. The α_3-domain resides below α_2 and anchors the MHC molecule to the cell membrane.

Most interesting MHC α_1 and α_2 domains ($C_{\alpha 1}$ … $C_{\alpha 182}$) turn out to be highly flexible in our MD-simulations. Temporal clustering captured this feature by relegating these atoms into several small clusters, all of them shown in the same color (grey) in Fig. 5. Of note, this region contains anchor sites and is thus naturally linked to the peptide ($C_{\alpha 376}$ … $C_{\alpha 384}$). This functional feature is clearly captured by spatiotemporal clustering and evidently shown in the circular-plots.

In contrast, the α_3-domain of the MHC ($C_{\alpha 183}$ … $C_{\alpha 276}$) together with the β_2-microglobulin (chain B, $C_{\alpha 277}$ … $C_{\alpha 375}$) turned out as the most rigid parts of the molecule. This holds for all three alleles.

Atomic mobility: Within and external of domains

In order to quantify the difference in atomic mobility between inside- and outside of domains, we computed the frequency distributions of the elements S_{ij} of the STDDV-matrix, see Fig. 6.

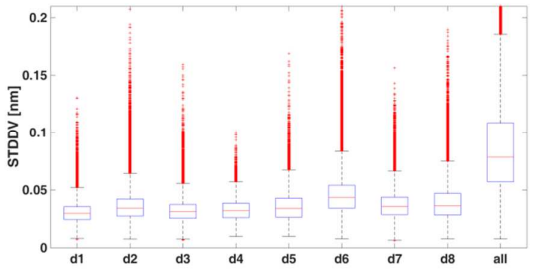

Fig. 6: Frequency distribution of standard deviations of pair distances. The semi-rigid domains identified by clustering have been labelled d1 to d8, cf. Fig. 9 for their location within the protein. Atomic distance variation internal domains is significantly lower than overall in the protein (box 'all'). Although C_α-atoms directly neighbouring in the protein backbone (C_α- C_α-atom-binding) have been excluded from the analysis, minima are very close to zero in each domain, indicating that other pairs of C_α-atoms also stay at very constant distances. Naturally, as larger domains harbour pairs with larger distances, which in turn also render larger variations (see the maxima).

Discussion and conclusions

Clustering to obtain semi-rigid domains within large bio-molecules such as TCR/pMHC complexes, is identified as an important pre-requisite for the numerical analysis of antigen cognition and signal transduction at an atomic level [39].

Survey of new methods introduced

For spatial clustering, as described in the literature [40], we have developed and implemented very fast new algorithms, see sections 'Spatial clustering' and 'Benchmarks for spatial clustering'. However, spatial clustering by the virtue of its basic principles suffers from drawbacks as outlined in section 'Critical review of spatial clustering'. Therefore we have extended spatial clustering by a subsequent step of 'temporal clustering', in general called 'consolidation', which — by its design — gets rid of highly flexible atoms: they end up in negligibly small clusters. The consolidation provided by temporal clustering not only resolves those shortcomings of spatial clustering but also yields additional information on interactions between different domains of the TCR/pMHC complex, which is highly valuable for understanding its function on a bio-molecular level.

Our new methods were evaluated for optimum performance regarding the choice of parameters (sections 'Number of spatial clusters' and 'Different thresholds'.) as well as statistical robustness (section Random sampling of MD-frames).

The convergence of spatial clustering was carefully evaluated in a pilot study as described above. The reliability of clustering is in fact secured by a two-layer procedure: 100 matrices, each one derived from a different time-interval, were spatially clustered and each clustering started from 1000 independent random assignments of atoms. In temporal clustering, above results are compared, and only compatible parts (threshold Δ_{th}) retained. If results of spatial clustering were in any respect random, clusters would by no means be compatible over time and by no means would they survive the temporal consolidation step.

Temporal clustering not only consolidated spatial clustering but also provided additional information on it, by evaluating its stability over time. As a first by-product, we have obtained a mathematical estimate for the optimum number of spatial clusters (section 'Optimizing the number of spatial clusters'), which could hitherto be selected by educated guess only. A second by-product of temporal clustering is the possibility to reduce artifacts due to time-wise autocorrelation otherwise affecting evaluations of MD trajectories: Selecting time frames

at random (from the whole trajectory) yielded similar results as compared to sets of time-wise successive frames. We conclude that clustering results obtained for more restricted subdomains conform to those pertaining over larger portions of configurational space.

Two methods of temporal clustering, single-link and complete-link, yielded very similar results, but different thresholds apply and have to be chosen with care.

Lower atomic motility inside domains as compared to outside domains, has been verified via respective frequency distributions (box-plots, Fig. 6).

The stability of spatiotemporal clustering, even in the presence of a majority of volatile atoms has been verified within the MHC complex, comprising two highly volatile alpha-helices and a beta-sheet.

Finally, we have introduced circular-plots, generally used in genome-sequence analysis, to delineate domains of joint motion along the peptide chains of the TCR/pMHC complex. This visualization technique, in conjunction with 3D molecular displays (VMD) [41], proved extremely helpful for interpreting results, see section 'Regions of joint motion' and Fig. 7.

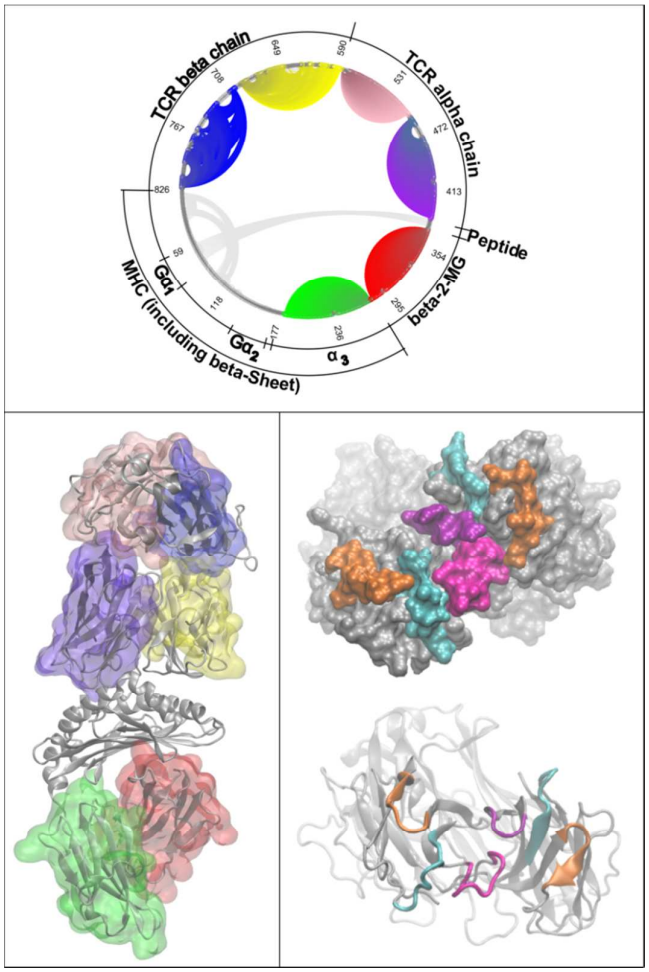

Fig. 7: Joint Motion analysis of domains shown in circular-plot and VMD. Upper panel: Circular-plot of spatiotemporal clustering for B4405 (k = 7, Δth = 8) with specific regions (CDR1, CDR2,…) and elements of secondary structure (peptide, β_2-microglobulin, α_3) annotated.

Lower left: Domains of B4405 in 3D, colored according to circular-plot above.

Lower right: Regions important for immunorecognition and domains with joint motion: TCR alpha chain (gray, left domain), TCR beta chain (gray, right domain), CDR1 alpha and beta chain (cyan), CDR2 alpha and beta chain (orange). Regions with joint motion in alpha chain are shown in magenta and partly coincide with CDR3 alpha. Regions with joint motion in beta chain are shown in purple and partly coincide with CDR3 beta.

Semi-rigid domains within TCR/pMHC complexes

Comparison of the results for the 3 alleles has identified several semi-rigid domains already discussed in section 'Different alleles'.

It is remarkable that no information whatsoever about secondary structure was plugged into the clustering algorithm beforehand. Spatial clustering, as usually done up to now, yielded results rather hard to interpret and utilize, cf. Fig. 1. Condensing spatial clusters by temporal clustering, however, resulted in domains conforming in many respects to protein chains and elements of secondary structure.

Each of the semi-rigid domains found – or all of them taken together – lend themselves as reference structures for fitting the trajectories in coming MD-studies of TCR/pMHC complexes. Generally, in choosing reference structures, the regions with internal motion should be avoided. However, at the same time the reference domain should be located as close as possible to the region whose motion is to be analyzed.

Accordingly, the following conclusions/recommendations drawn from this work may serve as examples:

1. To investigate the flexible region of chain A ($G\alpha_1$, $G\alpha_2$, and β-floor) one could restrain the fitting region to the stable

cluster α_3 of chain A (green cluster in Fig. 7) together with chain B (red cluster).

2. The clusters within the TCR chain D (violet, pink) and chain E (yellow, blue) exhibit small parts spared from clusters, see Fig. 7. To investigate details of motion of such a small domain, one might use the surrounding cluster (in which the small domain is embedded) as a reference providing optimum contrast. Of note, some of these spared parts are in fact the CDR-loops, supposed to play an important role in the immune-recognition process, see also 'Regions of joint motion'.

Regions of joint motion within TCR/pMHC complexes

Temporal clustering not only consolidates spatial clusters to several larger semi-rigid domains. It also creates a considerable number of smaller clusters, which we originally considered an irrelevant by-product. However, some of them turned out to carry important information by reflecting immunological functionalities. Inspecting Fig. 5 and Fig. 7 for the small clusters (all shown in grey) running across the circle, one may read off:

1. Clusters (shown in grey) connecting the peptide and the binding cleft are present in all alleles. This is trivial, since the peptide is by definition attached to the binding cleft and hence moving concordantly (little relative motion between cleft and peptide, hence same cluster). This is truely captured by spatiotemporal clustering.
2. Parts of the α-helices and β-floor ($C_{\alpha 1}$ … $C_{\alpha 117}$, groups 7, 8, …) are linked to CDR3 of the TCR beta-chain in B4402, $C_{\alpha 680}$ … $C_{\alpha 684}$, see Fig. 5. CDR3s are supposed to be crucial players in immune-recognition [42;43], seem to be moved by parts of the α-helices for this allele. This could explain why B4402 is allo-reactive: The immunoreaction is triggered by a (non-matching) MHC rather than an immunogenic peptide.
3. Also the peptide is linked to TCR-CDR3 in B4402. At this point we cannot decipher whether there is a separate role for the peptide regarding immunogenicity. Note that the LC13 TCR allo-reacts with HLA-B*44:02/ABCD3 complex. The ABCD3 peptide is a self-peptide, which is not immunogenic in other antigen-binding clefts (e.g. HLA-B*44:03), but adopts a conformation similar to viral peptides and stimulates an allo-reaction when the LC13 TCR is ligated to the HLA-B*44:05/ABCD3 or HLA-B*44:02/ABCD3 complexes [30]. Being presented within the cleft of a mutant allele might, however, transform the motility of the peptide so as to boost the allo-reaction.
4. In B4403, parts of antigen binding cleft are linked to a highly flexible short section within chain D ($C_{\alpha 473}$ … $C_{\alpha 481}$), which is also part of the CDR3 region and has – similarly - been spared from the violet cluster, see Fig. 5. Interestingly, this joint motion does not give rise to an immune reaction.
5. From the flexible region ($C_{\alpha 1}$ … $C_{\alpha 117}$, groups 7, 8, …) numerous links lurk out in each complex (B4402 and B4403) but not in B4405.

All in all, even the mini-clusters emerging from spatiotemporal clustering seem to capture elements of immunological functionality.

Clustering for a larger molecule: TCR/pMHC in complex with CD8

As outlined in section 'Molecules' we have applied spatiotemporal clustering also to a second and even larger bio-molecular complex.

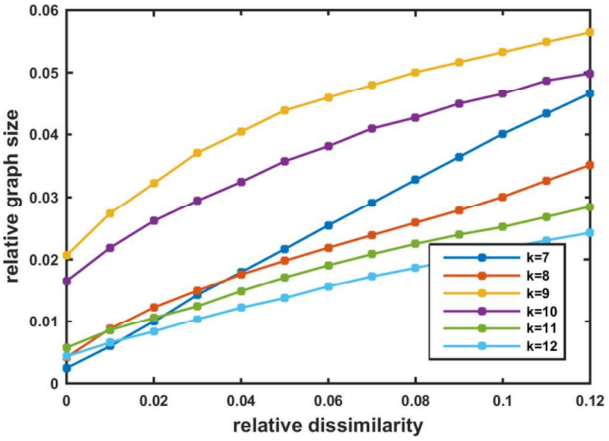

Fig. 8: Finding the optimum number of clusters from the relative size of graph as it depends on the threshold Δ_{th} for dissimilarity within groups. A separate analysis was performed for each preset number of spatial clusters, k = 7, 8,…12. The optimum is obtained for k =9. Note that the intercepts of curves with the vertical axes correspond to relative graph sizes for zero tolerance of dissimilarity.

Our previous complex, TCR/pMHC was enlarged by adding a CD8 coreceptor. Doing this, three important questions arise: (i) will spatiotemporal clustering automatically yield a larger number of clusters as the optimum choice? (ii) will the semi-rigid domains found within TCR/pMHC (alone) remain unchanged with CD8 being attached? (iii) Which semi-rigid domains emerge within CD8?

The first question is answered by Fig. 8: The optimum temporal stability of graphs is obtained for k = 9, i.e. adding CD8 yields two more clusters than optimum for TCR/pMHC alone (without CD8).

The resulting domains including annotations are shown in Fig. 9. Explicit results for domains in terms of C_α-indices are referenced in the 'Supporting information'.

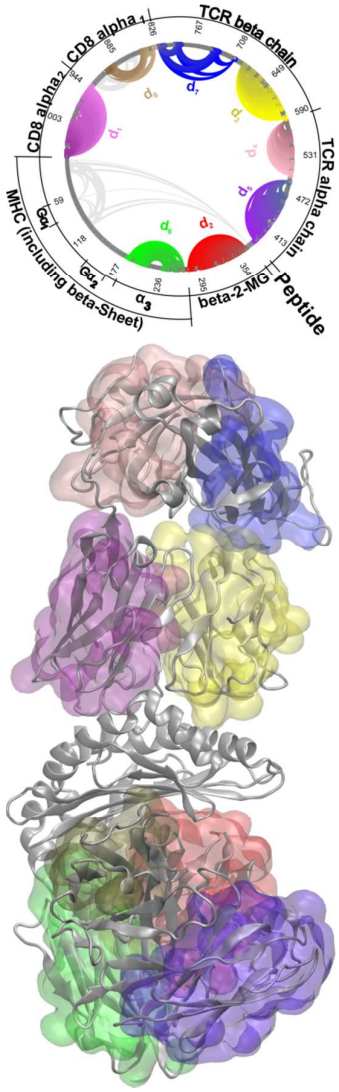

Fig. 9 Semirigid domains for TCR/pMHC/CD8. The circular-plot shows molecular parts of CD8 inserted after those of the TCR, see the annotation and also Table 1. Spatial Clustering was performed for $k = 9$ and termporal clustering with zero tolerance. The largest 8 clusters were considered semi-rigid domains (SRDs) and thus colored. The lower part shows a 3D representation (VMD), with domains colored with respect to the circular-plot above.

The second question can be answered by comparing the C_α-ranges of corresponding domains, see Fig. 10, left column (panels A, C). Also in the presence of CD8, MHC $\alpha1$ and $\alpha2$ domains ($C_{\alpha1}$ … $C_{\alpha182}$) are most flexible and via their anchor sites linked to the peptide ($C_{\alpha376}$ … $C_{\alpha384}$), see the grey cluster. The $\alpha3$-domain of the MHC ($C_{\alpha183}$ … $C_{\alpha276}$) together with the β_2-microglobulin (chain B, $C_{\alpha277}$ … $C_{\alpha375}$) turned out as the most rigid parts of the molecule, also in the presence of CD8. The two chains forming the TCR, both located more remote from CD8, remain virtually unaffected by the presence of CD8. All in all, the TCR/pMHC complex accomodates identical semi-rigid domains in the presence of CD8 and in the absence of CD8. We may conclude that the addition of CD8 (with a considerable number of 228 C_α atoms) to the whole system does not change clustering within TCR/pMHC. The two domains emerging in addition, exclusively reside within CD8.

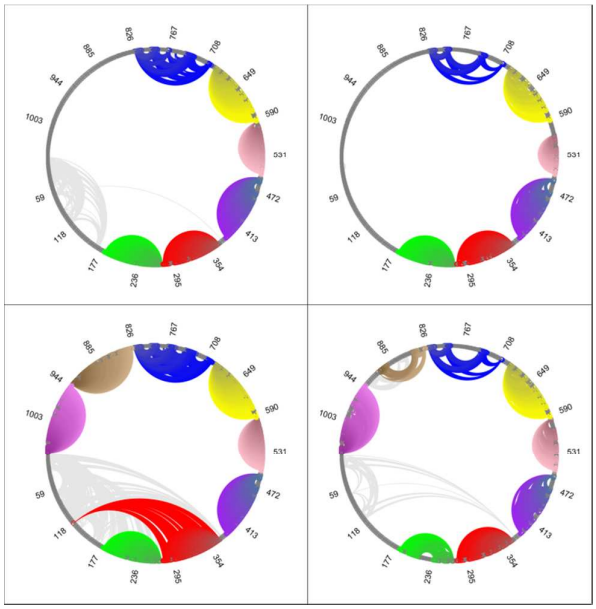

Fig. 10: Spatiotemporal clustering of TCR/pMHC alone and in complex with CD8.

Upper Row (A, B): Spatial clustering of B4405 performed for $k = 7$ clusters. Clustering is displayed on a circular-plot providing space also for accomodating CD8.

Lower Row (C, D): Spatial clustering of B4405+CD8 performed best for $k = 9$ clusters. CD8 clusters are displayed in circular-plots in their numeration corresponding to the upper row.

Panel A: B4405 clustered with $k = 7$, $\Delta_{th} = 7$ (relative dissimilarity 0.07), leading to a relative graph-size of approximately 0.06.

Panel B: B4405 clustered with $k = 7$ and $\Delta_{th} = 0$, (relative dissimilarity 0.00) leading to a relative graph-size of 0.0335.

Panel C: B4405 + CD8 clustered with $k = 9$, $\Delta_{th} = 14$ (relative dissimilarity 0.14), leading to a relative graph-size of approximately 0.06.

Panel D: B4405 + CD8 clustered with $k = 9$ and $\Delta_{th} = 0$, leading to a relative graph-size of 0.0228.

It is remarkable that spatiotemporal clustering, although unsupervised, perfectly separates additional domains (within CD8) from previously recognized ones (within TCR/pMHC).
The answer to the third question (domains internal of CD8) can be directly read off from Fig. 10 , lower row (panels C, D). The two chains of secondary structure (within CD8) are separated into two corresponding semi-rigid domains extending from $C_{\alpha827}$ - $C_{\alpha940}$ and from $C_{\alpha941}$ - $C_{\alpha1054}$. Each of them houses groups of C_α ("exclaves") which do not belong to the semi-rigid domain. For relaxed dissimilarity threshold (larger Δ_{th}), exclaves are small (Fig. 10, Panels A, C). More stringent temporal clustering increases exclaves and renders semi-rigid domains more sparse (Fig. 10, Panels B, D). A detailed example is presented in Fig. 11, showing the membership of C_αs in one of the semi-rigid domains within CD8 for $\Delta_{th} = 0$.

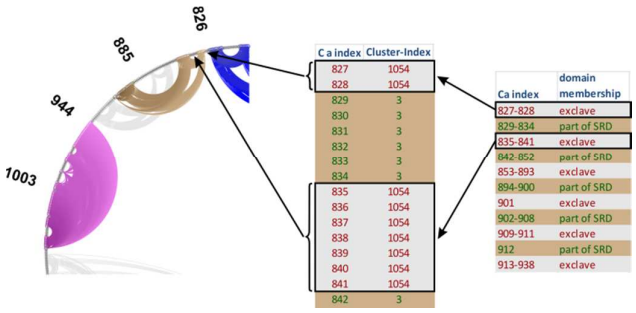

Fig. 11: Details of semi-rigid domain within CD8 α_1.

Detail of semi rigid domains (SRD) shown in Fig. 10 panel D. A small part of the first SRD (shown in brown) within CD8 has been annotated regarding C_α being part of the SRD as opposed to C_α belonging to (mobile) exclaves. In the circular-plot, the color of the circumference encodes if the respective C_α is part of the SRD (shown as brown dot) or if it is not part of the SRD (grey dot). Links between pairs of C_αs are shown in brown if both C_αs are members of the SRD. Links between pairs of C_αs are shown in grey if both C_αs are members of a smaller cluster, which is not large enough to be considered an SRD (note: only the largest clusters, in this case 8, are considered to be SRDs). C_αs which are shown in grey along the circumference without giving rise to a link to any other C_α, do not belong to any cluster.

Temporal clustering was first performed with $\Delta_{th} = 14$, yielding a relative dissimilarity of 0.06. For a large threshold ($\Delta_{th} = 14$), see Fig. 10, panel (C), we observe:

- Besides those large clusters within B4408, already well known from clustering B4405 alone, the addition of CD8 shows numerous C_α within the β_2-microglobulin being linked to one single $C_{\alpha 120}$ within the MHC (shown in red), see the S1 Table.
- Moreover, numerous C_α cluster into a very dense domain within the TCR β chain (shown in blue).

For zero tolerance $\Delta_{th} = 0$ (no dissimilarity tolerated in temporal clustering), see Fig. 10, lower row, right panel (D). Relative graph size shrinks to 0.027. As connections become more sparse, one observes:

- Connections between β2-microglobulin to $C_{\alpha 120}$ vanish.
- Only part of the peptide (anchor residue $C_{\alpha 376}$) is classified as being linked to the beta floor of the MHC.
- Within CD8 a substructure emerges: some pairs of C_α are tightly coupled, while others in-between seem to be more flexible.

Applicability and Scope

Clustering for a molecule with large flexible parts

In the above examples, the new method of spatiotemporal clustering has been applied to molecules with the majority of atoms belonging to rigid domains and only a minority to flexible parts. In order to verify the applicability of spatiotemporal clustering also in a molecule with large flexible parts we considered both chains (A and B) of the MHC (see Table 1 and Fig. 5) as the entire complex, within which domains were to be allocated. In this setting, we know that about half of the atoms, i.e. a significant proportion, belong to flexible regions (alpha-helices G1 und G2 and the beta-floor). The question was if they would again be recognized as flexible or if rigid domains would spuriously be identified? To check this, we first applied the optimality criterion for the number of spatial clusters by inspecting the relative graph size (in this case only for zero tolerance), yielding $|E(G)| = 0.087 \leq p \cdot N(N-1)/2$ for $k = 3$ and lower values for $k > 3$. Given the fact that 2 is the lowest possible number of rigid domains we adopted $k = 3$ as smallest – and in this case optimum – number of spatial clusters. During temporal clustering, only two domains survive the consolidation process. The result is shown in Fig. 12: Of note that the sets of atoms are almost identical to those found previously, when clustering the huge molecular complex (including peptide, TCR and CD8), compare Fig. 12 with Fig. 7 and Fig. 9.

Systems of different size

We have tested spatiotemporal clustering on three systems of different size (MHC: 375 atoms, pMHC+TCR: 826 atoms, pMHC+TCT+CD8: 1054 atoms). Considering sub-parts of one and the same large complex allowed us to prove that our procedure retrieved rigid domains consistently, no matter if and how many other domains or even volatile regions were present.

Conclusions for domain finding, fitting and analysis of molecular function

Finding semi-rigid domains for trajectory fitting and the actual analysis of motion for regions of interest are two distinct steps in scrutinizing bio-molecular function numerically. They interdepend, however. Without appropriate fitting, nothing will be found regarding function or even signal transduction.
On the other hand, given appropriate fitting, a plethora of statistical tools is available for the actual analysis, e.g., normal mode analysis (NMA) and principal component analysis (PCA). Appropriate fitting is hence a necessary precondition but by no means a guarantee to numerically distill functional elements out of bulk molecular motion.
In the present work we have introduced a new, two step algorithm with significant capabilities:

- Automatic detection of the optimum number of clusters.
- Atoms not lending themselves as members of semi-rigid domains ("mobile groups") can be excluded by setting an appropriate threshold for dissimilarity tolerance.

Dissimilarity tolerance for temporal clustering can be selected as appropriate for the specific research goal: When aiming at larger clusters, which may at the same time be less rigid, a larger tolerance (threshold $\Delta_{th} = 1,2,\ldots$) should be selected. Examples are soft links (Van der Waals) between parts of the molecule belonging to different domains of secondary structure, nevertheless inducing some concordant movements. However, if the goal is to arrive at very rigid, although smaller clusters, as desirable, e.g., for RMSD-fitting domains, zero tolerance is appropriate (threshold $\Delta_{th} = 0$). In this case we found that single-link clustering yields the same results as complete-link clustering for our MD-data.
It is interesting that similar relative graph-sizes not necessarily lead to identical connections being displayed, when comparing molecules of different size (such as TCR/pMHC and

TCR/pMHD+CD8): While a relative graph-size of 0.06 suppresses the links between domains of TCR for TCR/pMHC, these become apparent for TCR/pMHC/CD8.

In developing spatiotemporal clustering we hope to contribute some advance to the pipeline of numerical immunology, by finding appropriate references for fitting. Although unsupervised, the method proved to be stable when applied to different molecules, yielding concordant results for corresponding parts. This provides a consistency check of the method. In addition, spatiotemporal clustering revealed features which are by no means obvious and eyeballing would not be able to tell:

- A surprising result is that none of the helices ($G\alpha_1$ and $G\alpha_2$) did emerge as rigid domains. As a consequence they do not lend themselves as fitting domains but should rather themselves be targeted by advanced statistical analyses in future attempts to numerically decipher the molecular triggers of immunogenicity.
- Some elements of secondary structure, such as parts of the TCR-beta chain (shown in blue in Fig. 9) as well as CD8 $alpha_1$ (brown in Fig. 9) did not appear as solid clusters but rather contain considerable exclaves. This feature became apparent by applying zero tolerance in temporal clustering. Detailed results, such as given in the supplementary material, can serve as a definition of domains for RMSD-fitting.

These perspectives indicate good chances for spatiotemporal clustering to provide a valuable basis for computational biologists in deciphering functionally relevant elements of macro-molecular motions.

During the elaboration of spatiotemporal clustering on the specific example of MD trajectory data, it became apparent that it is in fact a special case of a far more general group of procedures that may be devised, multistep consensus clustering. Broader investigations will have to reveal their potentials and applicability.

Acknowledgements

The authors gratefully acknowledge preparation of the manuscript by Susanne Rom. MD was run on the IBM-BlueGene/P at Bulgarian NCSA. Partial support by Bulgarian Science Fund and Austrian Academic Exchange Programme under Grants DNTS-A 01-2/2013 and WTA-BG 06/2013 is acknowledged.

The code is written in MATLAB, also using the package circularGraph (http://www.mathworks.com/matlabcentral/fileexchange/48576-circulargraph) and can be downloaded from https://snowball1108@bitbucket.org/BioSimVienna/multistage-clustering.git.

Reiner Ribarics is an employee and stockholder of Gilead Sciences.

The authors declare that there is no conflict of interests regarding the publication of this paper.

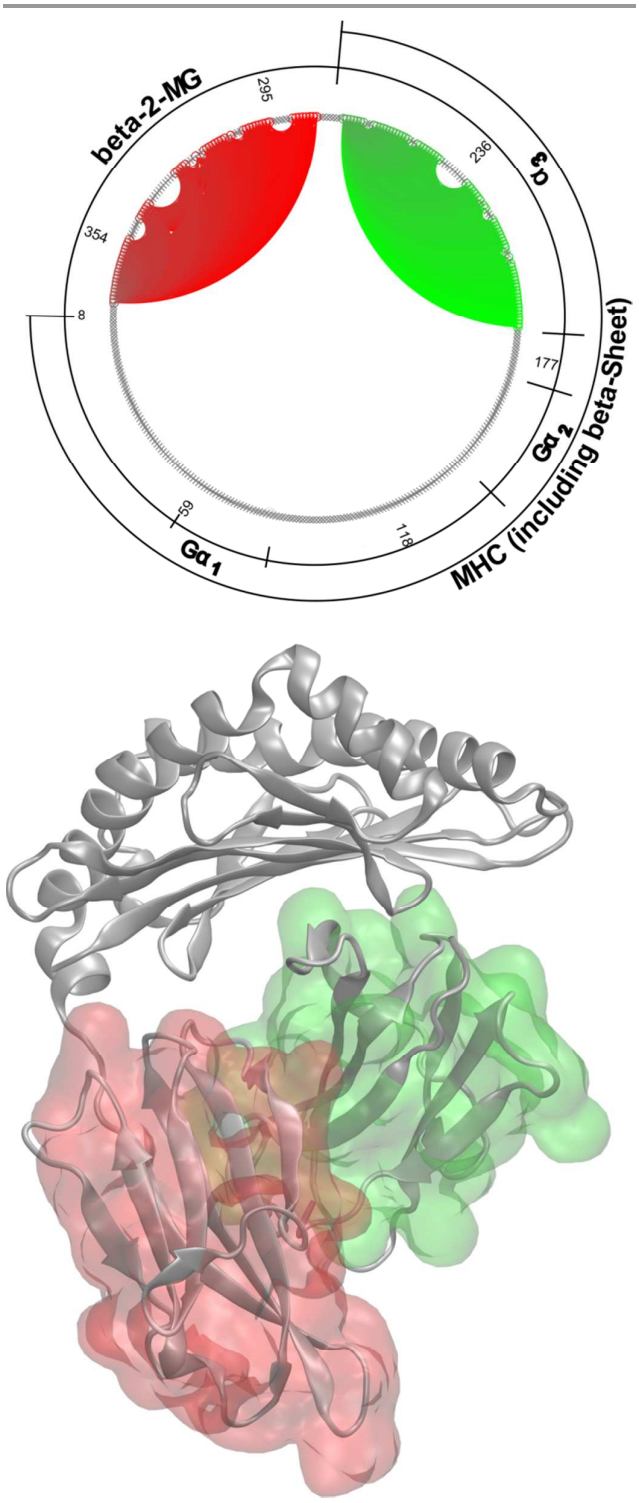

Fig. 12: Spatiotemporal clustering of MHC protein. Even if a large portion of mobile atoms is present (183 mobile atoms out of 375 in total), spatiotemporal clustering does not create spurious rigid domains. It rather retrieves the very same rigid domains (β_2-microglobilin (red) and α_3) as were found for the large complex (including the TCR and CD8), in which mobile atoms represent a much smaller fraction.

Notes and References

1. G. Bao, *Journal of the Mechanics and Physics of Solids*, 2002**, 50**, 2237.
2. R. Lavery, A. Lebrun, J. F. Allemand, D. Bensimon, and V. Croquette, *Journal of Physics Condensed Matter*, 2002**, 14**, R383.
3. D. Gordon, R. Chen, and S. H. Chung, *Physiol Rev.*, 2013**, 93**, 767.
4. Y. Cui, *J.Pharm.Sci.*, 2011**, 100**, 2000.
5. S. A. Adcock and J. A. McCammon, *Chemical Reviews*, 2006**, 106**, 1589.
6. B. Roux, *The Journal of General Physiology*, 2010**, 135**, 547.
7. L. Moise and A. S. De Groot, *Nat.Biotechnol.*, 2006**, 24**, 791.
8. V. Brusic and N. Petrovsky, *Novartis.Found.Symp.*, 2003**, 254**, 3.
9. D. R. Flower, I. K. Macdonald, K. Ramakrishnan, M. N. Davies, and I. A. Doytchinova, *Immunome Res.*, 2010**, 6 Suppl 2**, S1.
10. J. C. Tong and E. C. Ren, *Drug Discov.Today*, 2009**, 14**, 684.
11. F. Pappalardo, F. Martinez, I, M. Pennisi, A. Palazon, I. Melero, and S. Motta, *PLoS.ONE.*, 2011**, 6**, e26523.
12. M. P. Lefranc, J. Robinson, S. G. E. Marsh, M. M. Schuler, M. D. Nastke, S. Stevanovic, M. Bhasin, S. Lata, G. P. Raghava, S. Saha, S. Srivastava, M. K. Singh, G. C. Varshney, P. Guan, I. A. Doytchinova, D. R. Flower, P. Kangueane, M. K. Sakharkar, P. A. Reche, E. L. Reinherz, Y. Zhao, M. H. Sung, R. Simon, C. K. Hattotuwagama, T. H. Lin, D. S. DeLuca, R. Blasczyk, P. Dönnes, W. Liu, J. Wan, X. Meng, T. Li, S. Ranganathan, J. C. Tong, R. R. Mallios, L. Huang, N. Murugan, Y. Dai, D. A. Winkler, and F. R. Burden, *Immunoinformatics: Predicting Immunogenicity In Silico*, Humana Press, Totowa, New Jersey, 2007, pp. 1.
13. F. Ehrenmann, Q. Kaas, and M. P. Lefranc, *Nucleic Acids Res.*, 2010**, 38**, D301.
14. M. Ferber, V. Zoete, and O. Michielin, *PLoS.ONE.*, 2012**, 7**, e51943.
15. B. Knapp, U. Omasits, W. Schreiner, and M. M. Epstein, *PLoS ONE*, 2010**, 5**, e11653.
16. U. Omasits, B. Knapp, M. Neumann, O. Steinhauser, H. Stockinger, R. Kobler, and W. Schreiner, *Molecular Simulation*, 2008**, 34**, 781.
17. D. R. Flower, K. Phadwal, I. K. Macdonald, P. V. Coveney, M. N. Davies, and S. Wan, *Immunome Res.*, 2010**, 6 Suppl 2**, S4.
18. H. Zhang, P. Wang, N. Papangelopoulos, Y. Xu, A. Sette, P. E. Bourne, O. Lund, J. Ponomarenko, M. Nielsen, and B. Peters, *PLoS ONE.*, 2010**, 5**, e9272.
19. M. Karplus and J. A. McCammon, *Nat.Struct.Biol*, 2002**, 9**, 646.
20. H. J. Berendsen and S. Hayward, *Curr.Opin.Struct.Biol*, 2000**, 10**, 165.
21. A. Amadei, M. A. Ceruso, and N. A. Di, *Proteins: Structure, Function, and Bioinformatics*, 1999**, 36**, 419.
22. D. Amaratunga, J. Cabrera, and Y. S. Lee, *Journal of Computational Biology*, 2015**, 22**, 54.
23. I. Bahar and A. J. Rader, *Curr.Opin.Struct.Biol*, 2005**, 15**, 586.
24. V. Gapsys and B. L. de Groot, *Biophys.J.*, 2013**, 104**, 196.
25. A. E. Garcia, *Phys.Rev Lett.*, 1992**, 68**, 2696.
26. A. Amadei, A. B. M. Linssen, and H. J. Berendsen, *Proteins: Structure, Function, and Bioinformatics*, 1993**, 17**, 412.
27. M. A. Balsera, W. Wriggers, Y. Oono, and K. Schulten, *Journal of Physical Chemistry*, 1996**, 100**, 2567.
28. S. Hayward and B. L. de Groot, *Methods Mol.Biol.*, 2008**, 443**, 89.
29. M. Kenn, R. Ribarics, N. Ilieva, and W. Schreiner, *Biomed Res.Int.*, 2014**, 2014**, 731325.
30. W. A. Macdonald, Z. Chen, S. Gras, J. K. Archbold, F. E. Tynan, C. S. Clements, M. Bharadwaj, L. Kjer-Nielsen, P. M. Saunders, M. C. Wilce, F. Crawford, B. Stadinsky, D. Jackson, A. G. Brooks, A. W. Purcell, J. W. Kappler, S. R. Burrows, J. Rossjohn, and J. McCluskey, *Immunity.*, 2009**, 31**, 897.
31. S. Bernhard and F. Noé, *PLoS ONE.*, 2010**, 5**, e10491.
32. R. M. Karp, in *Complexity of Computer Computations*, ed.R. Miller, J. Thatcher, J. Bohlinger, Springer US, 1972, pp. 85-103.

33. S. A. Cook, *The complexity of theorem-proving procedures*, 1971.

34. L. Kaufman and P. J. Rousseeuw, *Finding Groups in Data: An Introduction to Cluster Analysis*, John Wiley, New York, 1 ed., 1990.

35. S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, in *Numerical Recipes: The Art of Scientific Computing*, Cambrige University Press, New York, 3 edn., 2007, chapter 16, pp. 701-744.

36. R. Y. Rubinstein, *Simulation and the Monte Carlo Method*, Wiley, New York, 1981, pp. 1.

37. A. K. Jain, M. N. Murty, and P. J. Flynn, *ACM Computing Surveys*, 1999**, 31**, 264.

38. C. Bron and J. Kerbosch, *Communications of the ACM*, 1971**, 16**, 575.

39. A. K. Dunker and V. N. Uversky, *Nat.Chem.Biol.*, 2008**, 4**, 229.

40. S. Xu, S. Zou, and L. Wang, *Journal of Computational Biology*, 2015**, 22(5)**, 436.

41. J. Hsin, A. Arkhipov, Y. Yin, J. E. Stone, and K. Schulten, *Curr.Protoc.Bioinformatics*, 2008**, SUPPL. 24**, 5.7.1.

42. C. A. Janeway, K. Murphy, P. Travers, and M. Walport, *Immunology*, Garland Science, New York, London, Seventh Edition ed., 2009.

43. C. A. Janeway, Jr., *Annu.Rev.Immunol.*, 1992**, 10**, 645.