SA-conf: a tool to identify variable regions in a set of related protein using a join analysis of their sequence and local structure defined by a structural alphabet

Leslie Regad[1,2], Jean-Baptiste Chéron[1,2], Caroline Senac[1,2], Triki Dhoha[1,2], Delphine Flatters[1,2], Anne-Claude Camproux[1,2]

1 INSERM, UMRS 973, MTi, Paris, France

2 Univ Paris Diderot, Sorbonne Paris Cité, UMRS 973, MTi, Paris, France}

1- Abstract

SA-conf is a tool dedicated to analysing the structural variability landscape for a protein target of interest. It is based on mining and comparison of all available conformations associated with this target, named multiple target conformations (MTC). This tool produces a joint variability analysis in terms of sequence and local structure of the MTC set based on a multiple sequence alignment (MSA) and the structural alphabet HMM-SA (Hidden Markov Model – Structural Alphabet [1]). SA-conf quantifies the sequence and local structural variability of each MSA positions. These results highlight important regions in terms of variability, which can have a role in the target function independent of the experimental resolution or methods chosen to determine the conformations. By crossing the obtained variability results on the target conformation subsets corresponding to different biological conditions, SA-conf demonstrated efficiency in distinguishing intrinsic from induced-fit flexibility. In conclusion, SA-conf is a relevant tool that facilitates a better understanding of the flexibility associated with a target and offers valuable insight into target interaction mechanisms and functions. The underlying data and SA-conf programs are available at

http://www.mti.univ-paris-diderot.fr/recherche/plateformes/logiciels.

This step-by-step tutorial describes the functionalities and the outputs of SA-conf, see Figure 1.

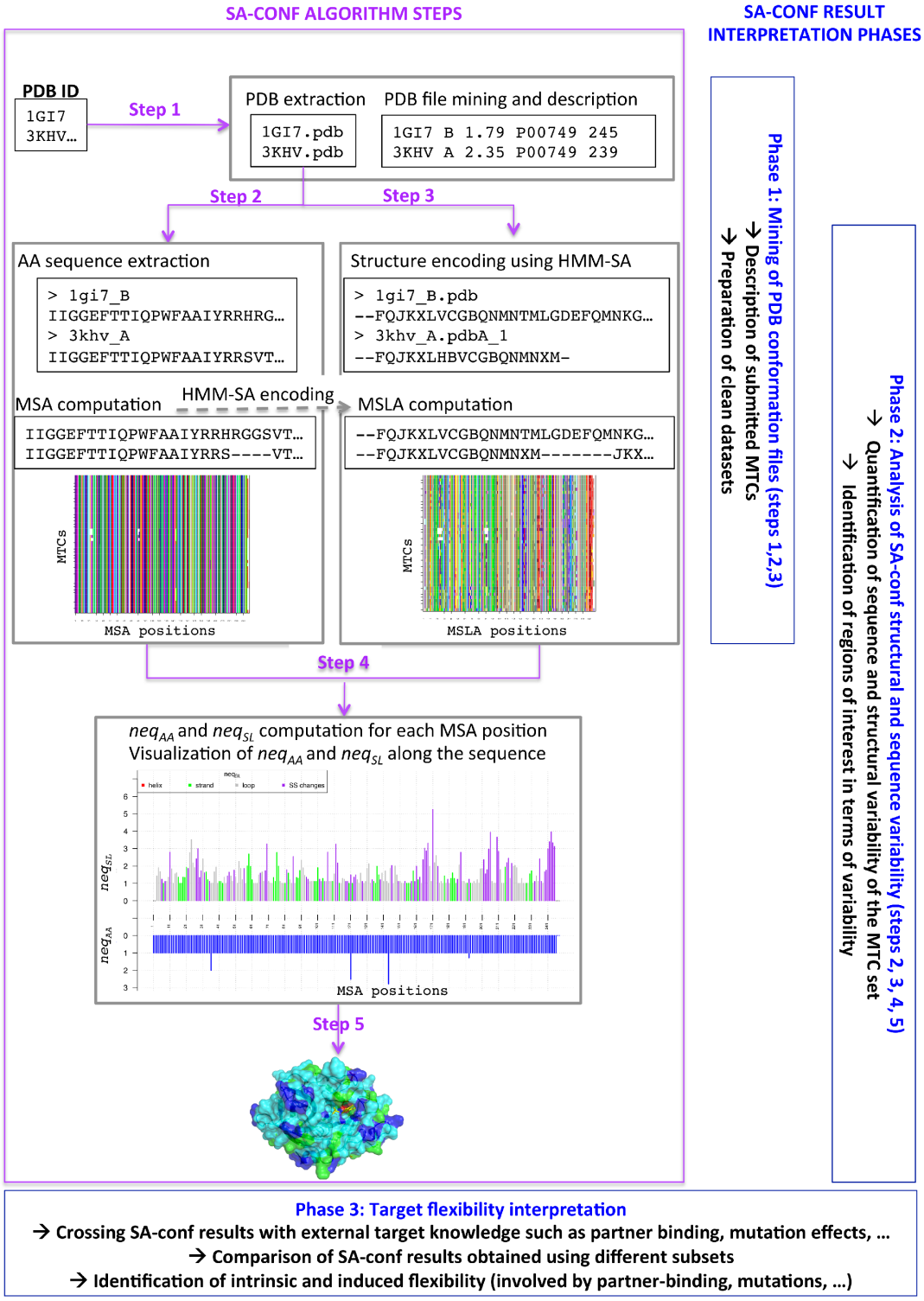


Figure 1: Presentation of the SA-conf algorithm and the protocol of the SA-conf result analysis. This figure is extracted from Regad et al., in submission.

2- Availability and installation

The python code of software SA-conf, along with a set of pre-compiled binaries, is freely available under GNU General Public Licence from http://www.mti.univ-paris-diderot.fr/recherche/plateformes/logiciels.

1. Download the archive file `SA-CONF.tgz` available at http://www.mti.univ-paris-diderot.fr/recherche/plateformes/logiciels.
2. Open a terminal window and move the archive in a directory.
3. Uncompress `SA-CONF.tgz` archive. The directory `SA-CONF` is created and it contains :
    - `saconf.x64` and `saconf.x32:` the two executables for 64 and 32 bits computers.
    - `src` directory that contains the dependency programs and the R script used for the analysis of the MSA.
    - `examples` directory that contains dataset of urokinases to test SA-conf program, see section “available datasets to test SA-conf”.

3- Pre-requires

SA-conf is a freely available python program running on GNU/Linux systems. It requires python[2], Biopython python package [3] a working installation of R [4], PyMOL[5], ClustalW [6] and T-Coffee [7] programs.

4- Available dataset to test SA-conf

In `SA-CONF` directory, the directory `example` contains three dataset available to test SA-conf. Each dataset is included in a directory :

- `uPA_set` directory: contains two datasets :
  - `P00749_set` directory contains the list of the 105 PDB ID corresponding to the P00749 UniProt ID (`list_uniprot.id` file).
  - `uPA_set` directory contains the list of 105 PDB ID chains corresponding to the uPA catalytic domain (`list_S1domain.ids` file); the corresponding PDB files stored in `PDB_chn` directory, the corrected MSA obtained using the 105 chains (`AA_alignment_S1domain_corrected.fasta2` file), text file that contains the target UniProt ID (`list_UniProt_id.ids file`).
- `p53_set` directory contains list of the 78 PDB ID chains corresponding to the p53 DNA binding domain (`identif_DBDdomain1.ids` file); the corresponding PDB files stored in `PDB` directory, the corrected MSA obtained using the 78 chains (`AA_alignment_corrected.fasta2` file) and the list of the UniProt ID of interest (`uniprot.ids` file)
- `2fej_nmr_model` directory contains list of the 36 NMR models extracted from 2FEJ PDB file, which correspond to a p53 DNA binding domain (`list_model_pdb.ids` file); the PDB files stored in `pdb_model_files` directory, the list of the UniProt ID of interest (`filelist_Uinprot_id.id` file). For this dataset, we created at first PDB file for each NMR model by extracting 3D coordinates of each model from 2FEJ PDB file. Then, each

PDB file was named using a code of four numeral from 0001.pdb to 0036.pdb.

5- SA-conf inputs

The simplest way to run SA-conf is by providing a list of protein ID, stored in a text file. To run SA-conf the user uses the following command:

```
$ saconf --IDfile  [option –pdbpath –method –align –uniprot ]

or

$ saconf -i [option -p -m -a -u]
```

Mandatory:

`--Idfile [-i]`: a text file containing the protein ID list. Protein ID are either the PDB *(Protein Data Bank)* code of the protein or complex or a protein chain noted pdbcode_chain, see Figure 2.

a)	1GI7	b)	1GI7_B
	1GJB		1GJB_B
	3KHV		3KHV_A

Figure 2: Two examples of the `input list_S1domain.ids` file that contains protein ID. Protein IDs are either the PDB ID of proteins (a) or the PDB ID of protein chains (b). This file is stored in the `…./SA_CONF/examples/uPA_set/P00749_set` directory

Optional:

`--pdbpath [-p]`: This option specifies a path of a directory that contains standard PDB files of each input protein ID. Each PDB file must be named either PDBID.pdb (e.g., `1GI7.pdb`) or PDBID_chain.pdb (e.g., `1GI7_A.pdb`). See section 7 step 1 to see the run command with this

option.

`--alignFile [-a]`: This option specifies a text file that contains a multiple sequence alignment (MSA) of input the PDB sequence of each protein ID. This file must be in fasta format (sequence is written on only one line). The aligned sequence ID must be the same as the input protein ID. Sequences included in this file must correspond to the sequence of the solved protein residues, i.e. sequences extracted from the PDB file. During SA-conf process, a step consists in checking the length of AA sequences extracted from the `AAalign_file.fasta file` and from the corresponding PDB file have the same length. If it is false for one protein, SA-conf returns an error message, see section10. See section 7 step 2 to see the run command with this option.

`--method [-m]`: (default = clustalw). This option enables to choose the multiple alignment method. Two choices are possible: clustalw or tcoffee to compute the MSA using ClustalW [6] or T-Coffee [7] algorithm. See 7 step 2 step 3 to see the run command with this option.

`--uniprot [-u]`: This option indicates a text file that contains a list of uniProt ID corresponding to one or several input PDB files. This option allows the creation of a text file containing the correspond between the positions number in the computed MSA, in each PDB file and in UniProt sequences for which the Ids are submitted. This option is incompatible with the `–method` option

6- List of SA-conf outputs

- **AA_alignment.fasta2**: text file with the MSA of all AA sequences in fasta format
- **Correspondence_positions.csv:** text file that contains the correspondence between MSA, PDB positions and the UniProt positions

- **Count_position_type.txt**: text file that provides information on the occurrence of the different position types.
- **dataset_composition.csv**: description of each submitted structures.
- **graph: directory that contains produced graphics**
  - **AA-alignment.pdf:** MSA representation
  - **SL-alignment.pdf** : MSLA representation
  - **Neq_graph.pdf**: Vizualisation of the *neqAA* and *neqSL* value for each MSA position
- **Mutation_res.txt**: Repartition of AA for each mutated positions
- **PDB**: directory with all downloaded PDB files
- **pdb_list_encode.id**: text file containing pdb id for encoded protein (into HMM-SA space)
- **Position_description.csv**: description of each MSA position in terms of AA and SL conservation/variability.
- **R_files_tmp**: files needed for the analysis of MSA and MSLA using R script.
- **script_pymol.pml**: pymol script allowing the visualisation of highlighted aligned positions onto a protein structure
- **SL_alignment.fasta2**: text file with the MSA translated into SL-alignment in fasta format
- **Structural_Variable_position_res.txt**: Repartition of SL for each structurally variable position.

7- Algorithm description and SA-conf outputs

This section presents the different steps of SA-conf program and its outputs. The example of the command to run SA-conf is performed using files located in the directory `…/SA-CONF/examples/uPA_set/uPA_set`. All output files will be created in the directory `.../SA-`

`CONF/examples/uPA_set/uPA_set/saconf_out`. Graphics obtained during the SA-conf process are stored in the directory `graph.`

Step 1- Extraction of PDB information

The PDB file of each input protein ID is extracted from the Protein Data Bank [8]. This step results in the creation of the directory named `PDB`, where all downloaded PDB files are stored. This step is run if the option `–pdbpath` is not used. To skip this step, the user must specifies a directory containing the PDB files of each input protein using the option `–pdbpath`. To do so, the user must use the following command to run SA-conf:

```
$ ../saconf --IDfile list_S1domain.ids --pdbpath PDB_chn
```

Based on a parsing of the submitted PDB files, SA-conf provides a description of *N* PDB files in terms of experimental approach used to solve the structures, the associated resolution for X-ray structures or the number of models for NMR structures, the number of chain(s), their length(s) and their associated UniProt ID, and the name of HETATM, i.e. atomic coordinate records used for atoms presented in HET groups (for atoms within "non-standard"groups), present in the PDB file. This information is stored in a csv file named `dataset_composition.csv`, see Figure 3.

pdb_id	experimental method	resolution	number of models	nbr of chain	chain:length	HETATM:Occ	chn:UniProtId:pos
3KGP	X-RAY DIFFRACTION	2.35	NA	1	A:239	SO4:1/4AZ:1	A:P00749:179-431
4JK5	X-RAY DIFFRACTION	1.55	NA	2	A:245/B:16	CL:2/DSN:1/ZBR:1/P6G:1/NH2:1/SO4:4	A:P00749:179-423
4OS7	X-RAY DIFFRACTION	2.0	NA	2	A:245/B:13	GOL:1/ACT:1/SO4:3/823:1/NH2:1	A:P00749:179-423
4OS6	X-RAY DIFFRACTION	1.75	NA	2	A:245/B:12	81R:1/ACT:1/SO4:2/NH2:1	A:P00749:179-423
4OS5	X-RAY DIFFRACTION	2.26	NA	2	A:245/B:13	81R:1/SO4:2/NH2:1	A:P00749:179-423
4OS4	X-RAY DIFFRACTION	2.0	NA	2	A:245/B:13	GOL:1/CL:1/NH2:1/SO4:2/ACT:1/81R:1	A:P00749:179-423
4OS2	X-RAY DIFFRACTION	1.79	NA	2	A:245/B:12	ACT:2/81S:1/SO4:2/NH2:1	A:P00749:179-423
4OS1	X-RAY DIFFRACTION	2.2	NA	2	A:245/B:13	ACT:1/81S:1/SO4:2/NH2:1	A:P00749:179-423
4MNY	X-RAY DIFFRACTION	1.7	NA	4	A:245/B:245/C:13/D:13	GOL:3/ACT:5/29O:2/SO4:4/NH2:2	A:P00749:179-423/B:P00749:179-423

Figure 3: `dataset_composition.csv` file. This file contains for each submitted PDB ID (column 1) experimental approach used to solve the structures (column 2), the associated resolution for X-ray structures (column 3) or the number of models for NMR structures (column 4) the number of chain(s) and their length(s) (column 5), the name of HETATM present in the PDB file (column 6) and the chain UniProt ID (column 7). HETATM, correspond to atomic coordinate records used for atoms presented in HET groups (for atoms within "non-standard"groups), present in the PDB file.

```
Created output files or directory:

     - dataset_composition.csv

     - PDB (directory)
```

Step 2- Sequence extraction and multiple sequence alignment computation

SA-conf extracts subsequently amino-acid (AA) sequences of each chain contained in all submitted PDB files. Extracted AA sequence corresponds to solved residues. From these extracted AA sequences, SA-conf computes a multiple sequence alignment (MSA) using either ClustalW [6] (by default) or T-Coffee [7] algorithm. The choice of the alignment method is done using the `--method` option:

```
# to choose clustalw method

       $ ../saconf --IDfile list_S1domain.ids--method clustalw

# to choose t-coffee method

      $ ../saconf --IDfile list_S1domain.ids--method tcoffee
```

The obtained MSA is written in fasta format in the text file `AA_alignment.fasta2`, see Figure 4.

The user can submit his own MSA file using the `--align` option:

```
$ ../saconf --IDfile list_S1domain.ids --align AAalign_file.fasta
```

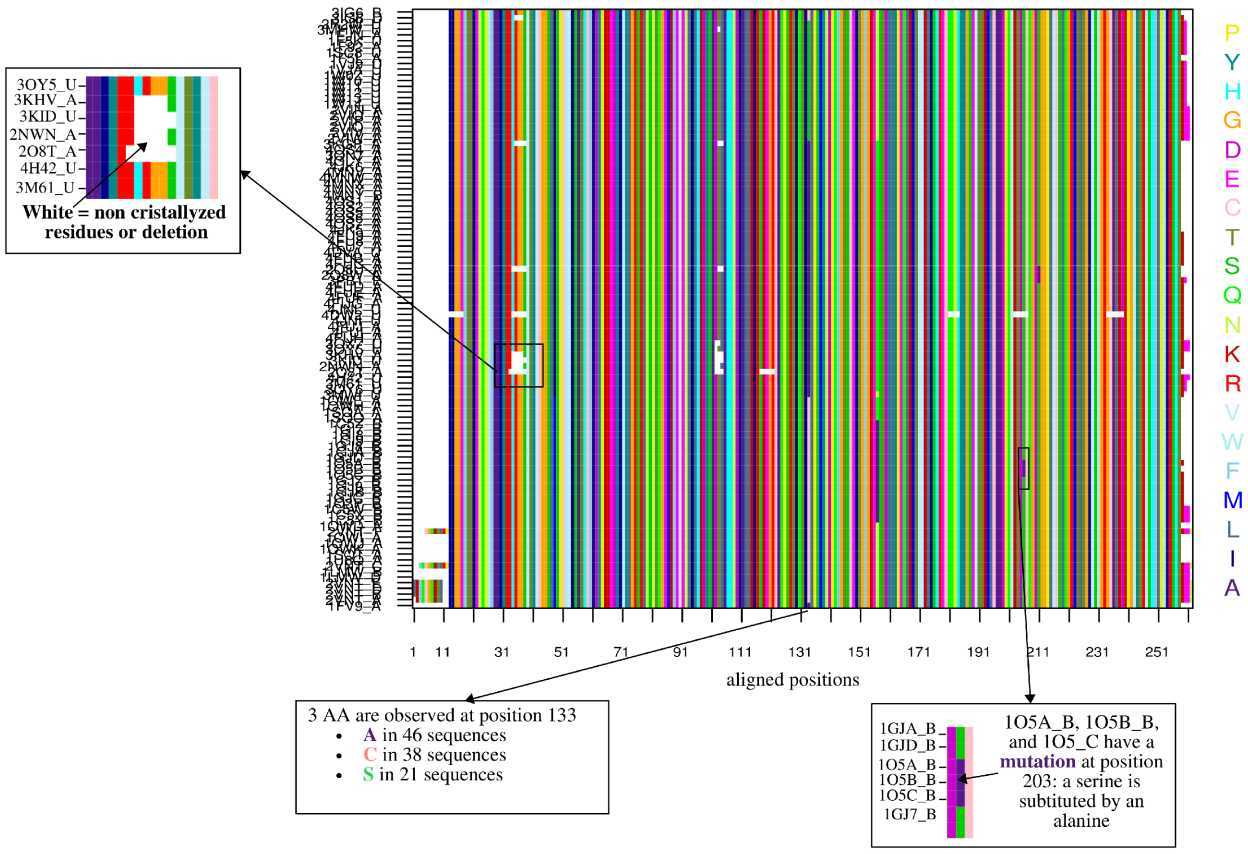


Figure 4: `AA-alignment.pdf` output file which presents the MSA (SA-conf Step 2 output) obtained using the set of 184 human uPA catalytic domains. A total of 184 aligned AA sequences are presented in rows, and the 261 MSA positions are shown in columns. Each position is coloured *according to the 20 AAs types.*

Protein sequences included in the submitted `AAalign_file.fasta` file must be the AA sequences of the solved residues, i.e. sequences extracted from the PDB file. During this step, a procedure ensures that AA sequences extracted from the `AAalign_file.fasta` file have the same length

than sequences extracted from PDB files. If it is false for one protein, SA-conf returns an error message, see section 10.

At the end of this step, SA-conf produces a graphic, named `AA-alignment.pdf` that presents the MSA with the *C* aligned AA sequences in rows and *p* MSA positions in columns, see Figure 4. Each position is coloured according to the 20 AA types. If the PDB ID list contains more than 50 ID, in addition SA-conf produces several graphics corresponding to a subpart of the `AA-alignment.pdf` graphic.

SA-conf also produces a table, named `Correspondence_positions.csv`, that contains the correspondence between MSA position numbers, position numbers in all PDB files and position numbers in UniProt sequences submitted UniProt ID, see Figure 5. The correspondence with UniProt sequences is proposed only if UniProt ID were submitted with `–uniprot` option using following command :

```
$ ../saconf --IDfile list_S1domain.ids --uniprot list_UniProt_id.ids
```

It is important to note that if the user submit a MSA it is impossible to use this option.

Aligned	P00749	3IG6_D	3IG6_B	2R2W_U	3MHW_U	1EJN_A	1F5K_U	1F92_A	1SC8_U	1F5L_A	1VJ9_U	1VJA_U	1W0Z_U	1W10_U	1W11_U	1W12_U	1W13_U	1W14_U	2VIN_A	2VIO_A	2VIP_A	2VIQ_A	2VIV_A	2VIW_A	3KGP_A	4OS4_A
1	169	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
2	170	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
3	171	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
4	172	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
5	173	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
6	174	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
7	175	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
8	176	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
9	177	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
10	178	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
11	179	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16
12	180	17	17	17	17	17	17	17	17	17	17	17	17	17	17	17	17	17	17	17	17	17	17	17	17	17
13	181	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18
14	182	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19
15	183	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20
16	184	21	21	21	21	21	21	21	21	21	21	21	21	21	21	21	21	21	21	21	21	21	21	21	21	21
17	185	22	22	22	22	22	22	22	22	22	22	22	22	22	22	22	22	22	22	22	22	22	22	22	22	22
18	186	23	23	23	23	23	23	23	23	23	23	23	23	23	23	23	23	23	23	23	23	23	23	23	23	23
19	187	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24
20	188	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25
21	189	26	26	26	26	26	26	26	26	26	26	26	26	26	26	26	26	26	26	26	26	26	26	26	26	26
22	190	27	27	27	27	27	27	27	27	27	27	27	27	27	27	27	27	27	27	27	27	27	27	27	27	27
23	191	28	28	28	28	28	28	28	28	28	28	28	28	28	28	28	28	28	28	28	28	28	28	28	28	28
24	192	29	29	29	29	29	29	29	29	29	29	29	29	29	29	29	29	29	29	29	29	29	29	29	29	29
25	193	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30
26	194	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31	31
27	195	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32	32
28	196	33	33	33	33	33	33	33	33	33	33	33	33	33	33	33	33	33	33	33	33	33	33	33	33	33
29	197	34	34	34	34	34	34	34	34	34	34	34	34	34	34	34	34	34	34	34	34	34	34	34	34	34
30	198	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35
31	199	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36	36
32	200	36S	37	37	37	37	37	37	37	37	37	37	37	37	37	37	37	37	37	37	37	37	37	37	37	37
33	201	-	37A	37A	37A	37A	37A	37A	37A	37A	37A	37A	37A	37A	37A	37A	37A	37A	37A	37A	37A	37A	37A	37A	-	37A
34	202	-	37B	37B	37B	37B	37B	37B	37B	37B	37B	37B	37B	37B	37B	37B	37B	37B	37B	37B	37B	37B	37B	37B	-	37B
35	203	-	37C	37C	37C	37C	37C	37C	37C	37C	37C	37C	37C	37C	37C	37C	37C	37C	37C	37C	37C	37C	37C	37C	-	37C
36	204	38D	37D	37D	37D	37D	37D	37D	37D	37D	37D	37D	37D	37D	37D	37D	37D	37D	37D	37D	37D	37D	37D	37D	-	37D
37	205	38	38	38	38	38	38	38	38	38	38	38	38	38	38	38	38	38	38	38	38	38	38	38	38	38
38	206	39	39	39	39	39	39	39	39	39	39	39	39	39	39	39	39	39	39	39	39	39	39	39	39	39

Figure 5: `Correspondence_positions.csv` file. This file contains the correspondence between MSA position numbers (first column) and the intital PDB position numbers in *C* chains (from the third column) and the associated UniProt numbers (second column) if a UniProt ID list is submitted.

```
Created output files:

      - AA_alignment.fasta2

      - AA-alignment.pdf

      - Correspondence_positions.csv
```

Step 3- Local structure and multiple SL (structural letter) alignment computation

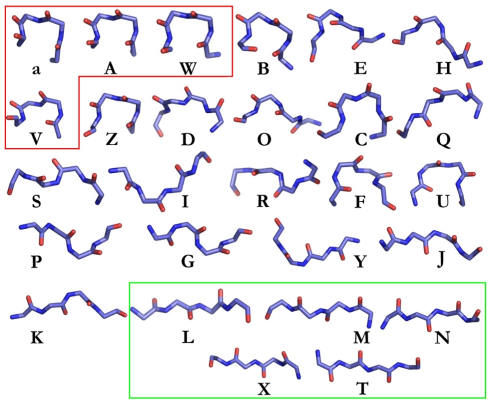
The next step of SA-conf consists in the extraction of the residue's local structures of all *C* protein chains. This step is based on the structural alphabet HMM-SA (Hidden Markov Model – Structural Alphabet [1]).

HMM-SA is a collection of 27 structural prototypes of four residues, named SLs, and labeled by letters {a, A-Z}, see Figure 6. It was established with hidden Markov model[1, 9] producing a

classification of protein four-residue fragments according to their geometric and their succession in structures. Using its 27 SLs, HMM-SA offers a very precise description of protein structures, particularly of protein loops [10]. A comparison between the 27 SLs and the 3 secondary structures (SS) shows that {A,a,V,W}-SLs are specific to α-helices, and {L,M,N,T,X}-SLs are specific to β-strand [1, 11]. The 18 remaining are devoted to loop description. α-helix-SLs and β-strand-SLs have an average RMSD of 0.16 ± 0.01Å and 0.69 ± 0.19Å, respectively, that shows that these two SL classes group 4-residue fragments with the same geometry, see Figure 6. Loop-SLs are more variable: they present an average RMSD of 1.35 ± 0.45Å (SLs G} and C have the maximal RMSD: 2.22 Å).

HMM-SA presentation

A	0.15																										
V	0.19	0.15																									
W	0.18	0.15	0.17																								
Z	0.34	0.31	0.27	0.28																							
B	0.29	0.27	0.29	0.29	0.41																						
C	0.58	0.57	0.51	0.51	0.42	0.68																					
D	2.14	2.15	2.1	2.09	1.98	2.14	1.77																				
E	0.72	0.72	0.71	0.7	0.73	0.53	0.93	2.05																			
O	1.39	1.39	1.38	1.37	1.37	1.18	1.51	1.85	0.75																		
S	1.76	1.76	1.73	1.74	1.69	1.57	1.75	1.63	1.15	0.64																	
R	2.21	2.22	2.16	2.18	2.08	2.09	1.98	0.93	1.81	1.4	1.03																
Q	1.22	1.21	1.17	1.19	1.11	1.05	1.16	1.81	0.72	0.69	0.74	1.39															
I	1.08	1.07	1	1.03	0.88	1.04	0.75	1.5	1	1.31	1.35	1.5	0.83														
F	1.42	1.43	1.39	1.37	1.3	1.43	1.18	1.26	1.42	1.59	1.67	1.54	1.43	1.19													
U	0.84	0.83	0.79	0.75	0.7	0.92	0.54	1.6	1.09	1.59	1.84	1.93	1.3	0.9	1.08												
P	2.3	2.31	2.28	2.25	2.19	2.19	2.07	0.81	1.91	1.48	1.41	0.92	1.76	1.76	1.45	1.85											
H	2.04	2.04	2	2	1.88	2	1.69	0.63	1.86	1.74	1.75	1.06	1.72	1.45	1.21	1.45	0.61										
G	2.19	2.19	2.18	2.17	2.17	2	2.22	1.25	1.59	0.95	0.85	0.98	1.37	1.91	1.71	2.15	0.76	1.2									
Y	1.81	1.82	1.76	1.74	1.63	1.8	1.42	0.82	1.67	1.72	1.69	1.15	1.48	1.15	1.12	1.21	0.9	0.53	1.5								
J	1.7	1.7	1.64	1.63	1.52	1.61	1.38	1.32	1.39	1.35	1.26	1.24	1.07	0.99	1.3	1.28	1.25	1.07	1.48	0.83							
K	2.15	2.15	2.09	2.08	1.96	2.09	1.78	0.76	1.9	1.76	1.54	0.85	1.58	1.37	1.32	1.6	0.66	0.55	1.18	0.57	0.9						
L	2.4	2.4	2.35	2.34	2.23	2.33	2.07	0.91	2.07	1.67	1.4	0.78	1.7	1.63	1.52	1.9	0.58	0.74	0.92	0.87	1.08	0.5					
N	2.54	2.55	2.51	2.52	2.43	2.38	2.34	1.15	2.01	1.4	1.04	0.67	1.56	1.83	1.76	2.26	0.81	1.14	0.72	1.29	1.33	0.88	0.6				
M	2.57	2.58	2.54	2.53	2.45	2.42	2.35	1.18	2.05	1.46	1.22	0.87	1.74	1.92	1.72	2.18	0.65	1.01	0.69	1.19	1.31	0.8	0.52	0.46			
T	2.56	2.57	2.54	2.54	2.48	2.38	2.47	1.42	1.97	1.34	1.01	0.91	1.59	1.99	1.9	2.38	0.93	1.33	0.7	1.49	1.43	1.1	0.77	0.44	0.47		
X	2.3	2.31	2.27	2.27	2.2	2.12	2.19	1.54	1.72	1.15	0.73	0.89	1.22	1.68	1.85	2.19	1.21	1.6	0.9	1.6	1.28	1.28	1.03	0.58	0.81	0.53	
	a	A	V	W	Z	B	C	D	E	O	S	R	Q	I	F	U	P	H	G	Y	J	K	L	N	M	T



Cα RMSD between the 27 SLs

Figure 6: Presentation of the 27 SLs of HMM-SA: their geometry (figure on the right) and their Cα-RMSD (Figure on the leftà. SLs are ranked according to their increasing stretches. On right figure, SLs specific to α-helices are highlighted in red and those specific to β-strands are highlighted in green. Other SLs describe loop conformations.

HMM-SA simplifies a 3D protein structure of *k* residues into a 1D sequence of *k-3* SLs, where each SL describes the geometry of a four-residue fragments, see Figure 7. Each SL provides information on the local structure of each residue and its neignbours. This simplification takes into account the structural similarity of four-residue fragments with the 27 SLs. It is achieved by a dynamic programming algorithm based on Markovian process to obtain maximum a posteriori encoding using the Viterbi algorithm. For each 4-residue fragments, the SL is assigned to the third residue of the corresponding 4-residue fragments. As the SL is assigned to the third fragment residue, the first,

second and last positions of the complete sequence have not assigned SL and is indicated by a '-', see Figure 7. For certain 4-residue fragments the encoding can be impossible because distances between non successive α-carbons are aberrant or residues are missing, ... In this case, the SL-sequence is split into several SL-sequences, see SL-sequences of `3KHV_A` chain in Figure 7.

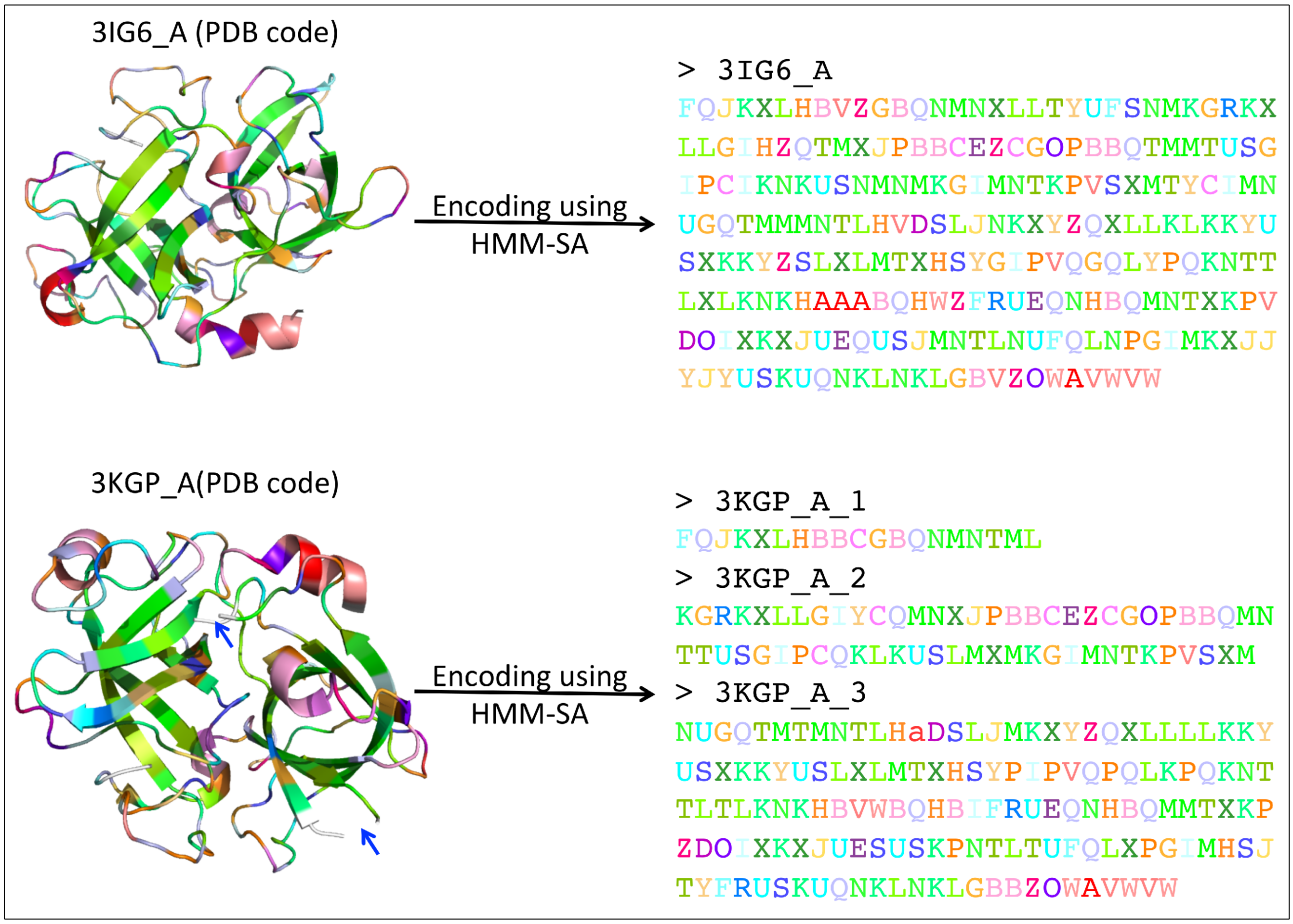


Figure 7: Encoding of 3IG6_A and 3KGP_A protein chains into a SL sequences using HMM-SA tool. Protein are coloured according to their SLs: α-helix-SLs and β-strand-SLs are presented in red and green, respectively. Other colours present loop-SLs.

In our previous studies, we have shown that HMM-SA is pertinent to precisely describe protein structures particularly loop conformation that allowed studying and classifying loop conformations [12,13], extracting structural and functional motifs from protein loops [14,15], characterizing protein-protein interactions [16] and quantifying deformation upon protein-protein interaction [17].

Local structure and multiple SL alignment computation

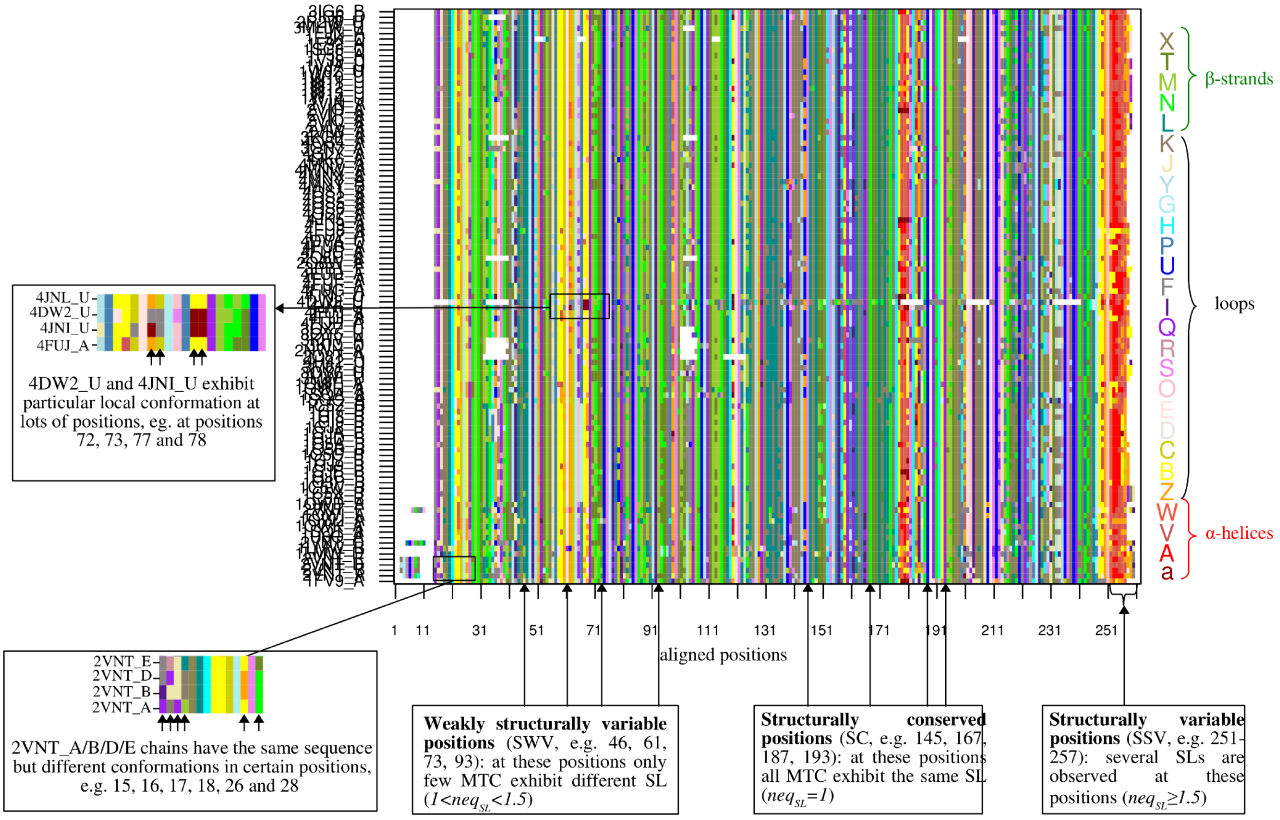


Figure 8: `SL-alignment.pdf` file which presents the MSLA (SA-conf Step 3 output) computed using the 184 uPA catalytic domains. In the MSLA, the 184 aligned SL sequences are shown in rows, and the 261 *MSA* positions are shown in columns and are coloured according to the 27 SLs. The colors of the 27 SLs indicate the secondary structure that each SL describes. [`a`, `A`, `V`, `W`]-SLs that are primarily found in the α-helix are coloured in red, and [`L`, `M`, `N`, `T`, `X`]-SLs that are primarily found in the β-strand are coloured in green.

First, SA-conf encodes each MTC chain into a SL-sequence. To compare efficiently the AA and structural variability of the MTC set and to overcome the difficulty to obtain multiple structural alignments of different length-conformers, SA-conf performs the local structure comparison of MTC chains using the computed MSA. To this aim SA-conf translates the MSA into a multiple SL alignment (MSLA) by replacing each MSA residue by its corresponding SL. When a gap `(``-") is

seen in the AA-alignment, a ``-" symbol is put in the MSLA. At the end of this stage, SA-conf produces a text file, named `SL_alignment.fasta2`, which contains the SL-alignment. When a residue has not a assigned SL, a ``-" symbol is put in the SL-alignment, such as the two first and last positions because the SL is assigned for the third residue of the 4-Cα fragment.

SA-conf also produces a graphic, named `SL-alignment.pdf`, that presents the MSLA where *C* aligned SL series of chains are in rows and *p MSLA* positions in columns, coloured according to the 27 SLs, see Figure 8.

```
Created output files:

     - SL_alignment.fasta2

     - SL-alignment.pdf
```

Step 4- MSA and MSLA analysis in terms of conserved or variable positions and regions

From computed MSA and MSLA, SA-conf determines the conservation and variability of the MTC sets *C* conformers by computed four indexes by positions :

- the number of AA observed by positions,
- the number of SL observed by positions,
- the exponential of Shannon entropy of the AA repartition at each position, noted neq_{AA},
- the exponential of Shannon entropy of the SL repartition at each position, noted neq_{SL}.

Neq_{AA} (resp. neq_{SL}) quantifies the number of different AA (resp. SL) observed at a position by taking into account both the number of different AA (resp. SL) observed in one position and also about their frequency, see equation 1. Neq_{AA} and neq_{SL} are computed in a position only if less than 50% of conformers have a gap in this position.

$$\text{neq}_{AA}(i) = e^{-\sum_{p=1}^{20} freq(aa_p^i) \times \ln freq(aa_p^i)}$$
$$\text{neq}_{SL}(i) = e^{-\sum_{p=1}^{27} freq(sl_p^i) \times \ln freq(sl_p^i)} \quad \text{Equation 1}$$

where $freq(aa_p{}^i)$ and $freq(sl_p{}^i)$ are the frequencies of the Aas aa_p and the structural letters sl_p, respectively observed at the MSA positions.

The neq_{AA} values vary between 1 and 20 Aas, and the neq_{SL} values vary between 1 and 27 Sls. A position with a neq_{AA} (resp. neq_{SL}) of 1 indicates that all *C* chains exhibit the same Aas at this position (resp. one SL). A neq_{AA} (resp. neq_{SL}) value close to 20 (resp. 27) indicates that 20 Aas (resp. 27 SLs) are equivalently observed on *C* chains, indicating a highly variable sequence position (resp. extremely structurally variable position). The *Neq* index offers information on the number of different AAs (or SLs) observed at one position and on their frequency. For instance, a *neq* of more than 1 but less than 2 indicates that more than one AA (resp. SL) is observed but with one AA predominantly observed, whereas a *neq* greater than 2 indicates at least two different AAs (resp. SLs) are predominantly observed at the considered position. Thus, the *neq* parameters can differentiate several types of position *i*:

- *neq(i)* = 1: Strictly conserved position *i*. The *C* chains exhibit only one AA (resp. one SL) at the position *i in the* MSA or MSLA.
- 1 < *neq(i)* < 1.5: Weakly variable position *i*. The *C* chains exhibit more than one AA (resp. one SL) at the considered position *i*, but one AA (resp. SL) is predominantly observed. This position exhibits certain rare AA (or SL) changes between the *C* chains.
- *1.5* ≥ *neq*(i): Rather variable position *i* Different AAs (resp. SLs) are observed corresponding to a variable position in terms of sequence (or local structure). The higher the value of *neq*, the greater the position will be. For example, a *neq* ≥ 3 indicates that more than 3 AAs (or three SLs) are equivalently observed and highlights a “highly variable” sequence (resp. structurally variable) position. A *neq* ≥ 5 is associated with a “strongly variable” sequence (resp. structurally variable) position.

To finish, SA-conf quantifies the structural deformation using secondary structures (SS) information by determining the number of SS observed at each position. The SS conformation for each residue was determined using the association of the 27 SLs with the three SS [1, 18]. Residues exhibiting

the {A, a, V, W}-SLs (resp. {L, M, N, T, X}-SLs) are assigned to a α-helix (resp. β-strand) conformation, while other residues are assigned to loop-conformation. This index allows differentiating two different positions:

- **conserved positions in terms of SS**: positions in which all MTC exhibit the same SS,
- **variable positions in terms of SS**: positions in which SS changes occur. All MTC do not exhibit the same SS at these positions.

Both neq_{SL} index and number of SS observed at a position are used to identify structurally variable positions. However, neq_{SL} can detect structural changes not captured by the classical SS information because it is based on the precise HMM-SA [12]. These particular changes are mainly occurring within loop conformations.

Examining the *neq* diversity index along the *p* MSA positions allows for extracting regions of interest composed *l* successive positions with similar *neq* values: conserved or variable structural regions. These important regions can have a role in target function.

During this step SA-conf produces several output files:

- `Count_position_type.txt` file. The first part of this file (“Occurrences of different type of positions”) gives the occurrence of each position types in terms of AA and SL, see Figure 7. From these positions remarkable regions can be identified. The second part of this file (“Distribution of position types along the MSA”) is created to facilitate their identification localization on the MSA and MSLA. It provide the starting position of each region type: structurally conserved positions, weakly structurally variable positions and strongly structurally positions, see Figure 7.

```
********* The MSA has 261 aligned positions *********

************************************************
*** Occurrences of different type of positions ***
************************************************

* 246 positions with a computed neq_AA value: average neq_AA = 1.02 (+/-0.17)
    -AA conserved positions :  240  (=  97.6 % of positions)
    -AA weakly variable positions :  3  (=  1.2 % of positions)
    -AA strongly variable positions :  3  (=  1.2 % of positions)

* 243 positions with a computed neq_SL value: average neq_SL = 1.67 (+/-0.75)
    -SL conserved positions :  22  (=  9.1 % of positions)
    -SL weakly variable positions :  115  (=  47.3 % of positions)
    -SL strongly variable positions :  106  (=  43.6 % of positions)


************************************************
*** Distribution of position types along the MSA ***
************************************************

* 22 SL structural conserved in the MSLA
    - 10 SL structural conserved positions are isolated
        start positions: 25 ; 48 ; 57 ; 64 ; 69 ; 92 ; 145 ; 167 ; 187 ; 193
    - 6 SL structural conserved positions are grouped in regions of  2 positions
        start positions: 29 ; 75 ; 84 ; 206 ; 225 ; 238

* 115 SL weakly structural variable in the MSLA
    - 12 SL weakly structural variable positions are isolated
        start positions: 28 ; 41 ; 49 ; 58 ; 61 ; 83 ; 93 ; 159 ; 172 ; 182 ; 194 ; 230
    - 14 SL weakly structural variable positions are grouped in regions of  2 positions
        start positions: 31 ; 46 ; 73 ; 86 ; 90 ; 99 ; 120 ; 134 ; 137 ; 155 ; 188 ; 191 ;
    - 10 SL weakly structural variable positions are grouped in regions of  3 positions
        start positions: 78 ; 95 ; 140 ; 146 ; 151 ; 168 ; 174 ; 196 ; 219 ; 235
    - 3 SL weakly structural variable positions are grouped in regions of  4 positions
        start positions: 52 ; 208 ; 244
    - 4 SL weakly structural variable positions are grouped in regions of  5 positions
        start positions: 18 ; 114 ; 162 ; 201
    - 1 SL weakly structural variable positions are grouped in regions of  6 positions
        start positions: 107
    - 1 SL weakly structural variable positions are grouped in regions of  7 positions
        start positions: 126

* 106 SL strongly structural variable in the MSLA
    - 15 SL strongly structural variable positions are isolated
        start positions: 56 ; 77 ; 94 ; 98 ; 113 ; 119 ; 133 ; 136 ; 139 ; 154 ; 171 ; 173
    - 13 SL strongly structural variable positions are grouped in regions of  2 positions
        start positions: 23 ; 26 ; 50 ; 59 ; 62 ; 81 ; 88 ; 143 ; 149 ; 157 ; 160 ; 199 ; 2
    - 4 SL strongly structural variable positions are grouped in regions of  3 positions
        start positions: 15 ; 70 ; 222 ; 227
    - 5 SL strongly structural variable positions are grouped in regions of  4 positions
        start positions: 42 ; 65 ; 122 ; 183 ; 231
    - 1 SL strongly structural variable positions are grouped in regions of  5 positions
        start positions: 177
    - 1 SL strongly structural variable positions are grouped in regions of  6 positions
        start positions: 101
    - 2 SL strongly structural variable positions are grouped in regions of  7 positions
        start positions: 212 ; 251
    - 1 SL strongly structural variable positions are grouped in regions of  8 positions
        start positions: 33
```

Figure 7: `Count_position_type.txt` file. The first part of this file provide information on the occurrence of each position types: conserved, weakly variable and strongly variable positions in terms of AA and SL. The second part of this file precises the starting position for each region types: structurally conserved positions, weakly structurally variable positions and strongly structurally positions. Each region is ranked according its size.

- `Position_description.csv` file, see Figure 8. In this file, SA-conf stores all variability

indexes.

aligned.pos	P00749	pdb.3IG6_D.pos	nbr.AA	nbr.SL	neqAA	neqSL	SS	signif
16	184	21	1	4	1	1.41	s-l	
17	185	22	1	2	1	1.06	s	
18	186	23	1	3	1	1.11	s-l	
19	187	24	1	3	1	1.11	l	
20	188	25	1	2	1	1.34	l	
21	189	26	1	3	1	2.78	h-l	*
22	190	27	1	2	1	1.83	l	/
23	191	28	1	1	1	1	l	
24	192	29	1	3	1	1.64	l	/
25	193	30	1	2	1	1.98	l	/
26	194	31	1	2	1	1.24	s	
27	195	32	1	1	1	1	s	

Figure 8: `Position_description.csv` file that stores in all these sequence and structure variability indexes for each MSA position. For a MSA position "`aligned.poS`" corresponds to its number of the position in the MSA, "`nbr.AA`" and "`nbr.SL`" correspond its number of AA and SL observed at here, "`neqAA`" and "`neqSL`" correspond to its neq_{AA} and neq_{SL} values, "`SS`" corresponds to its SS-change status, "`signif`" summarizes previous information. Concerning the SS status of a position, "`h`" is assigned when all MTCs have an α-helix conformation, "`s`" is assigned when all MCs have an β-strand conformation, "`l`" is assigned when all MTCs have a loop conformation, "`h-l`" is assigned when MTCs have either a loop or α-helix conformation, "`s-l`" is assigned when MTCs have either a loop or β-strand conformation at this position, "`h-s-l`" is assigned when MTCs have eigher a loop, α-helix or β-strand conformation at this position. Concerning the "`signif`" status of a position "!" is assigned mutations occur in this position, "/" is assigned if the position is structurally variable (neq_{SL} > 1.5) and "" is assigned if at this position both structural variability (neq_{SL} > 1.5) and SS changes are observed*

- `Neq_graph.pdf` graphic file, see Figure 9. This graphic presents neq_{AA} and neq_{SL} values for each position, see Figure X. Neq_{AA} yields the localization onto the AA sequence of strictly conserved position, weakly and strongly mutated positions and neq_{SL} yields the localization of the sequence the strictly conserved, weakly and variable positions in terms of local

structure.

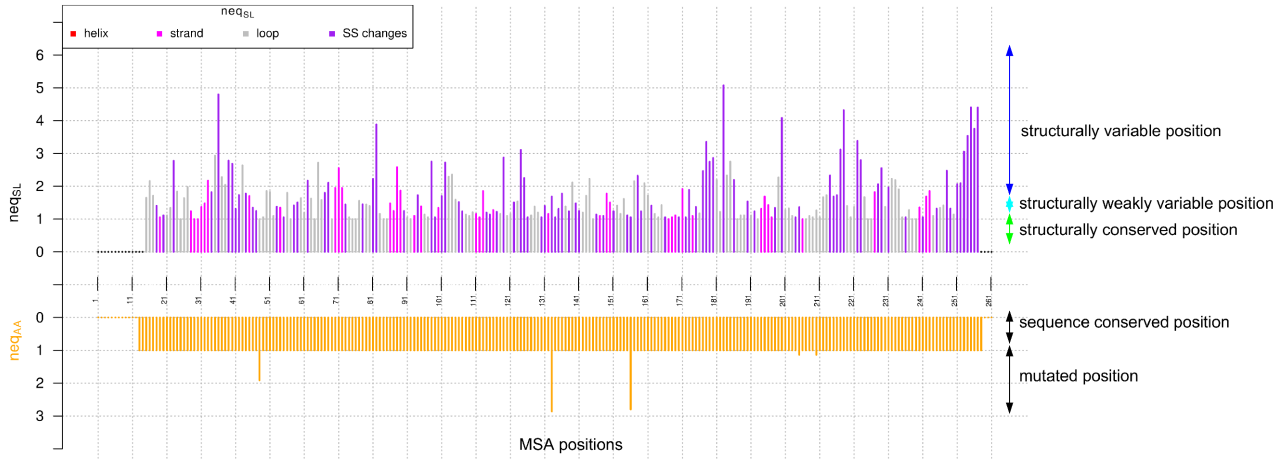


Figure 9: `Neq_graph.pdf`. This graph presents the neqAA (bottom graph) and neqSL (top graph) values along the 261 MSA positions. Bars presenting neqSL values are coloured according to their SS status: red presents the positions in which all chains have an α-helix conformation, magenta presents the positions in which all chains have an β-strand conformation, gray presents the positions in which all chains have a loop conformation, and purple presents the positions where SS changes occur.

- `Mutation_res.txt` file. This file allow for facilitating the analysis of mutated positions, It informs on the AA and SL distribution for each mutated positions by precising the different observed AAs and SLs and their occurrences, see Figure 10.

```
Mutation  1  - aligned position:  48
*** Repartition of AA in this mutated position ***
     I :  68
     M :  37
*** Repartition of SL in this mutated position ***
    I(68)      G:68
    M(37)      G:37
------------------------------------------------------------
Mutation  2  - aligned position:  102
*** Repartition of AA in this mutated position ***
     D :  99
     - :  5
     T :  1
*** Repartition of SL in this mutated position ***
    D(99)      T:68 ; M:11 ; N:7 ; X:7 ; -:3 ; G:2 ; O:1
    -(5)      0
    T(1)      -:1
------------------------------------------------------------
```

Figure 10: First lines of the Mutation_res.txt file. For each mutated position, it is indicated the different observed AA and SLs and their occurrences.

- `Structural_Variable_position_res.txt` file. This file allows for facilitating the

analysis of structurally variable positions. It informs on the SL distribution for each structural variable positions by precising the different observed SLs and their occurrences, see Figure 11.

```
Structural variable position n° 1 - aligned position: 11 neqSL= 2.85
*** Repartition of SL in this structural variable position ***
     K :  12
     L :  21
     - :  9
     R :  2
     Y :  3
------------------------------------------------------------
Structural variable position n° 2 - aligned position: 12 neqSL= 1.12
*** Repartition of SL in this structural variable position ***
     K :  42
     - :  4
     Y :  1
------------------------------------------------------------
```

Figure 21: First lines of the Structural_Variable_position_res.txt file. For each structural variable position, it is indicated the neq_{SL} value and the different SLs and their occurrences.

```
Created output files:

     - Count_position_type.txt

     - Position_description.csv

     - Neq_graph.pdf

     - Mutation_res.txt

     - Structural_Variable_position_res.txt
```

Step 5: Localization of the significant positions into a protein structure

SA-conf produces a PyMOL script, named `script_pymol.pml` to localize mutated, structurally conserved and variable positions on the protein structure. To perform that, SA-conf chooses the first protein of the MSA file, `./saconf_out/AA_alignment.fasta2`.

To run this generated PyMOL script, open a terminal and move in the directory where directory `saconf-out` was created and type the following command:

```
$pymol saconf_out/script_pymol.pml
```

This script open a PyMOL session where the target is presented and colored according to the different position type: structurally conserved position, weakly structurally variable positions, structurally variable positions, see Figure 12.

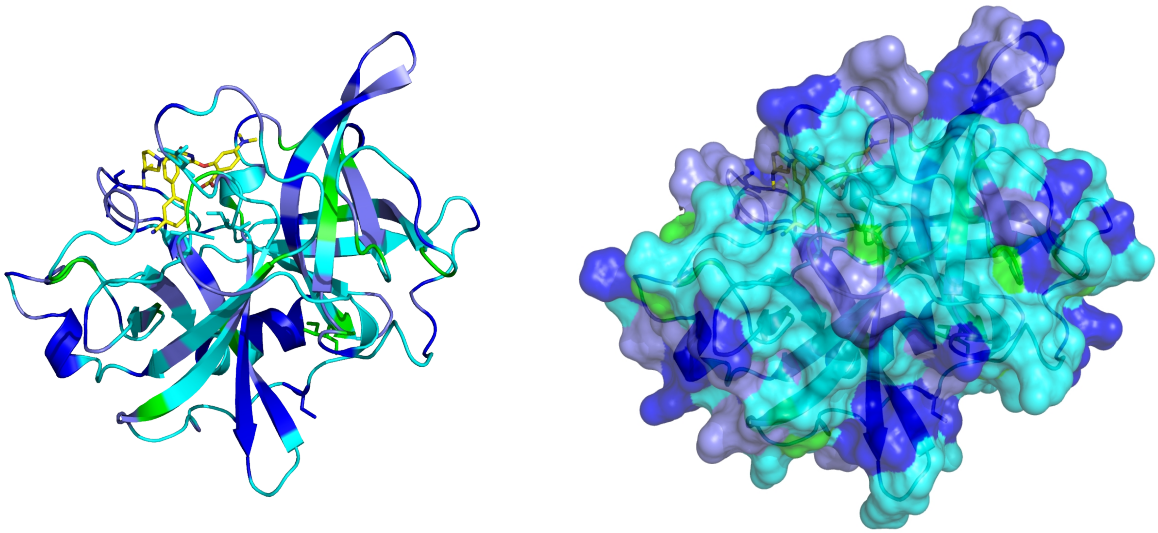


Figure 12: Graphical representation of the human uPA domain complexed with a chemical inhibitor (PDB code 3IG6_B). The protein is displayed as a cartoon (left figure) and as a surface (right figures) and is coloured *according to the type of positions: structurally conserved positions are* coloured *in green, weakly structurally variable positions are* coloured *in cyan, structurally variable positions where SL changes do not imply SS changes are* coloured *in "blue marine", and structurally variable positions where SL changes imply SS changes are* coloured *in dark blue. Residues located at mutated positions are displayed as sticks. The ligand 438 (hetatom code) is displayed in stick form with its C atoms* coloured *yellow. These figures were generated using PyMol and the* `script_pymol.pml` *generated during the SA-conf Step 5 output.*

8- Analysis protocol of the SA-conf results

The SA-conf protocol analysis operates via the five steps presented in the Materials and Methods section and in Figure 1 and can be grouped into three main phases.

Phase 1 results in an overview of protein PDB files corresponding to an ensemble of MTCs

characterized by specific experimental conditions, sequences and local structure contents. Phase 2 is applied on a selected MTC set to simultaneously compare the sequence and local structure variability associated with this set. Quantification of the sequence/structure variability is performed for each position in the aligned MTC to identify the sequence and structure variable regions of interest. In phase 3, the detected variable regions are analysed in terms of flexibility information and are then crossed to biological contexts and conditions (such as protein/protein and protein/ligand interactions) using different MTC subsets based on user expertise to create a better understanding of the target flexibility.

Phase 1: Mining of PDB conformation files

Mining of available MTCs associated with a given target is performed by combining the first three steps of SA-conf (see Materials and Methods section and Figure 1a). The first step produces a description of the submitted MTCs in terms of the experimental method used to solve each PDB file, the oligomerization state, the list of non-AA residues, and noted hetatoms complexed with each structure. The output of step 1 summarizes which PDB entries are solved using NMR or X-ray crystallography experimental methods and which correspond to a monomer or a homo- or hetero-oligomer. In addition, apo forms of the target (not complexed with a ligand) from holo forms can be identified by considering information on the presence of a HETATM. An MSA of the submitted MTCs is proposed in the step 2 output. Protein chains with incomplete sequences (including deletions, insertions, missing or mutated positions) or that are isolated from different organisms are directly identified in the MSA visualization. Step 3 deduces a multiple structural letter alignment (MSLA) of MTCs from the MSA. Protein chains with particular global or local 3D conformations can be directly detected in this MSLA visualization. The results from these three steps can be combined to distinguish conformations with putative important roles in biological mechanisms from unreliable conformations, e.g., due to experimental resolution limitations. Finally, phase 1 can aid the user in preparing a clean MTC set (and eventually different subsets of interest) by detecting and removing unreliable protein chains (including missing residues), retaining the conformation corresponding to one chain in the case of homo-oligomers, or selecting MTCs corresponding to protein chains of interest (i.e., with high-resolution X-ray structures with particular mutations or interactions).

Phase 2: Analysis of SA-conf structural and sequence variability

In phase 2, a deeper target sequence and structural variability analysis is performed by SA-conf using one selected MTC set of interest. The outputs of steps 2 to 5 (see Materials and Methods) allow identification and quantification of the target variability in terms of both sequence and structure. Combining of the outputs from steps 2 and 3 allows for direct comparison of the sequence and local structural variability of MSA positions. For more detailed analyses, the step 4 output offers quantification of the sequence and structural variability for each MSA position using neq_{AA} and $neq_{SL,}$ respectively (see Figure 1D). Visualization of these two criteria along the MSA positions allows identification and localization of mutated positions associated with structural variations and vice versa. For instance, $neq_{SL} = 1$ indicates a strictly conserved position in terms of structural variability, whereas $neq_{SL} \geq 1.5$ indicates a rather variable position. Most variable positions or regions can be extracted by considering positions with a neq_{SL} of interest, i.e., greater than certain fixed thresholds related to the variability of the considered MTC set. The number of successive MSA positions l with neq_{SL} values of interest indicates the length of the variable region. The output of step 5 creates a visualization of the variable positions of interest in one 3D structure conformation and, thus, an overview of the solvent accessibility of these regions (see Figure 1D).

Phase 3: Target flexibility interpretation

In this phase, the SA-conf analysis of the structural and sequence variability of MTC subsets obtained under different biological conditions or contexts based on user knowledge can offer selected clues for interpreting the target flexibility. SA-conf can highlight the flexible regions explained by the presence of AA mutations, partner binding, induced-fit effect or intrinsic target flexibility.

For the AA effect, the outputs of step 4 allow for backbone deformation analysis of the function of mutations by highlighting the mutated positions (corresponding to side chain changes) involved in a backbone deformation. If one position is variable in SL but not in AA, the observed backbone deformation is not implied by the change in side chains. The observed local structural change can be imputed to the intrinsic flexibility of the target, partner interaction or experimental condition variation. In contrast, if one position is variable in AA but not in SL, the side chain change likely has no effect on the backbone conformation of the corresponding residue and its direct neighbours (an SL is a four-residue fragment). However, it is possible that this mutation involves an “indirect” backbone deformation, i.e., a residue not in the direct neighbourhood but close in 3D space and in contact with the side chain.

On the topic of intrinsic flexibility, the SA-conf results using an apo MTC set (of a wild-type target in apo forms) with an identical sequence allow for identification of the structural variation involved via intrinsic flexibility or certain experimental condition variations (pH, space group, etc.). One approach to evaluate the structural variation due only to intrinsic flexibility is to analyse an ensemble of NMR models of a target. Indeed, the NMR model set corresponds to different conformations that satisfy the experimental restraints in solution under the same experimental conditions. The conformational variability derived from these models is known to greatly contribute to valuable insights on the understanding of flexibility [Hirst et al., 2014].

For induced flexibility, matching of SA-conf structural variable regions obtained using a holo MTC set to binding regions extracted by the user can aid in the localization of regions involved in protein function and/or interactions with partner binding (i.e., with small molecules, proteins, and nucleic acids). In the case of protein or nucleic partners, comparison of SA-conf results might aid in identifying structural deformations involved in protein/protein or protein/nucleic acid binding. In the case of ligand partners, comparison of SA-conf results on different holo MTCs (i.e., binding to different ligands) might help identify structural deformations induced by ligand binding and ligand diversity. Finally, comparison of SA-conf variability results obtained using apo versus holo MTC subsets can offer information that distinguishes structural flexibility induced by partner binding from that involved in the target's intrinsic flexibility.

9- Illustrations

Two illustrations of SA-conf result analysis are presented in the manuscrit: Regad L, Chéron JB, Senac JB, Triki D, Flatters D, Camproux AC. SA-conf tool: Comprehensive mining of multiple target conformations for insight on target flexibility. PloS Comp Biol, in submission

10- Error messages encountered during SA-conf run

- error: argument `-i/--IDfile` is required: no text file containing the PDB ID of input proteins is specified.
- [ERROR] PDB list and alignment file length mismatch. This error occurs when a file with

a MSA is specified using the option `--align`. This error message is returned when the the number of protein input is not the same in input file with the protein ID (`list_S1domain.ids`) and in the AA-alignment file (`AAalignment_file.fasta`).

- [ERROR] Rscript not found. With this error, the R script is not run and the pymol script is not created. To solve this problem, user can run himself the R script and a python scripts available in the directory `.../saconf/src`. This step requires the copy and paste of all files contained in the `R_files_tmp` directory into the `saconf_out` directory.

  To run these two scripts follow the following steps:

- open the `.../saconf/src/USER_script.R` file.
- change the path corresponding to the variable `folder_out` (first line) by replacing it by the variable out submitted using the option -o or by default `saconf out`.
  1. open the .../saconf/src/USER gen pml.py file.
  2. change the path corresponding to the variable `folder_out` (fourth line) by replacing it by the variable out submitted using the option `-o` or by default `saconf out`.
  3. move to the directory where the directory `saconf_out` (or directory name submitted using the option -o) is created
  4. use the following command to run `script.R` script.

     ```
     $ R CMD BATCH .../saconf/src/USER_script.R
     ```

  5. use the following command to run `USER_gen_pml.py` script.

     ```
     $ python .../saconf/src/USER gen pml.py
     ```

- [ERROR] sequence mismatch, pdb_id. This error occurs when a file with a MSA is specified using the option `--align`. This error message returns the ID protein for which the AA sequences are not mismatched, its sequences extracted from the alignment and from

the PDB file and the correspondence between each AA in the Fasta and PDB sequences. This error means that for at least one protein, the sequence included in the MSA file is not identical to its sequence extracted from the PDB file.

Références

[1] Camproux, A.C., Gautier, R., Tuffery, P.: A hidden markov model derivated structural alphabet for proteins. J Mol Biol 339, 561–605 (2004)

[2] Python Software Foundation. Python Language Reference, version 2.7. Available at http://www.python.org

[3]Cock PJA, Antao T, Chang JT, Chapman BA, Cox CJ, Dalke A, Friedberg I, Hamelryck T, Kauff F, Wilczynski B and de Hoon MJL. Biopython: freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics* 2009;25:1422-1423.

[4] R Core Team. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria, 2012. ISBN 3-900051-07-0.

[5] The PyMOL Molecular Graphics System, Version 1.8 Schrödinger, LLC.

[6] Larkin MA, Blackshields G, Brown NP, Chenna R, McGettigan PA, McWilliam H, Valentin F, Wallace IM, Wilm A, Lopez R, Thompson JD, Gibson TJ and Higgins DG. ClustalW and ClustalX version 2. Bioinformatics 23(21): 2947-2948. (2007) doi:10.1093/bioinformatics/btm404

[7] Notredame, Higgins, Heringa. T-Coffee: A novel method for multiple sequence alignments , JMB, 302 (205-217) 2000

[8] Berman, H.M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T.N., Weissig, H., Shindyalov, I.N., Bourne, P.E.: The protein data bank. Nucl Acids Res 28, 235–242 (2000)

[9] Camproux AC, Tuffery P, Chevrolat JP, Boisvieux JF, Hazout S. Hidden Markov model approach for identifying the modular framework of the protein backbone. Protein Eng. **12(12):1063-73 (1999).**

[10] Regad, L., Guyon, F., Maupetit, J., Tuffery, P., Camproux, A.C. A hidden markov model applied to the protein 3d structure analysis. CSDA 52, 3198–3207 (2008)

[11] A.C. Camproux, P. Tuffery, L. Buffat, C. Andrea, J.F. Boisvieux, S. Hazout. Analyzing patterns between regular secondary structures using short structural building blocks defined by a hidden Markov model Theor Chem Acc (1999) 101:33-40

[12] Regad, L.; Martin, J.; Camproux, A. C. Identification of non random motifs in loops using a structural alphabet Proceeding in IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology Location: Toronto, CANADA Date: SEP 28-29, 2006

[13] Regad L, Martin J, Nuel G, Camproux AC. Mining protein loops using a structural alphabet and statistical exceptionality. BMC Bioinformatics 2010 11:75

[14] Regad L, Martin J, Camproux AC.Dissecting protein loops with a statistical scalpel suggests a functional implication of some structural motifs. BMC Bioinformatics 2011 12:247

[15] Regad L, Saladin A, Maupetit J, Geneix C, Camproux AC. SA-Mot: a web server for the identification of motifs of interest extracted from protein loops. NAR 2011 9:W203-9

[16] Martin J, Regad L, Etchebest C, Camproux AC.Taking advantage of local structure descriptors to analyze inter-residue contacts in protein structures and protein/protein complexes. Proteins 2008 73: 672-689

[17] Martin J, Regad L, Lecornet H, Camproux AC. Structural deformation upon protein-protein interaction: a structural alphabet approach.BMC Structural Biology 2008 8:12

[18] Camproux AC, Tuffery P, Buffat L, Andrea C, Boisvieux JF, Hazout S. Analyzing patterns between regular secondary structures using short structural building blocks defined by a hidden Markov model Theor Chem Acc (1999) 101:33-40