Development and Validation of a Structure-Based Computational Method for the Prediction of Protein Specificity Profiles

Olivier Gagnon

A thesis submitted in partial fulfillment of the requirements for the
Master's degree in Chemistry

Department of Chemistry
Faculty of Sciences
University of Ottawa

Abstract

Post-translational modification (PTM) of proteins by enzymes such as methyltransferases, kinases and deacetylases play a crucial role in the regulation of many metabolic pathways. Determining the substrate scope of these enzymes is essential when studying their biological role. However, the combinatorial nature of possible protein substrate sequences makes experimental screening assays intractable. To predict new substrates for proteins, various computational approaches have been developed. Our method relies on crystallographic data and a novel multistate computational protein design algorithm. We previously used our method to successfully predict four new substrates for SMYD2 (Lanouette S & Davey J.A., 2015), doubling the number of known targets for this PTM enzyme that has been difficult to characterize using other methods. This was possible by first extracting a specificity profile of Smyd2 using our algorithm and subsequently screening a peptide library for matching sequences. However, our method did not yield successful results when attempting to reproduce specificity profiles of other proteins (64% accuracy on average). Different protein environments have demonstrated limitations in the methodology and lead us to further develop the algorithm on a more thorough dataset. Using our new optimized method, specificity profile predictions increase by roughly 20% (84% accuracy on average), independent of the structural template used. The algorithm was then used to blindly predict a specificity profile for the methyltransferase Smyd3, an enzyme for which limited data is currently available. A library of 2550 peptides was screened with the predicted profile, yielding 123 matching sequences. We randomly chose 64 for experimental validation (SPOT peptide array) of methylation by Smyd3 and found 45 methylated and 19 non-methylated peptides (70% success rate). Finally, we released to the community a web version of the algorithm, which can be accessed as http://viper.science.uottawa.ca.

Acknowledgements

The realization of this thesis would not have been possible without the help of many. First, I would like to thank my parents for the help they provided throughout these 2 years. A special thanks to Benoit Gagnon, my dad, without whom, I would certainly not have been able to achieve as much. Computational protein design requires many programming skills, which I couldn't have obtain so quickly without him. He taught me most of what I acquired to this date regarding my programming knowledge and I will be forever grateful. All the discussion we had allowed me to develop good programming practices and successfully code a framework for proteins manipulations in JAVA.

I would also like to thank my coworkers Tony St-Jacques, Matthew Eason and James A. Davey for all the helpful discussions in the lab. Their advice and comments were more than appreciated.

Finally, I would like to thank my supervisor, Roberto A. Chica, for the support and the trust he put in me. He let me lead the project my own way, he was open to my ideas and helped me become a better scientist.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

API……Application programming interface

CPD……....Computational protein design

FG……...Functional group

FN……...False negative

FP…….False positive

MOE……...Molecular Operating Environment

MSA……...Multi-state analysis

MSD…….Multi-state design

NMR……..…Nuclear Magnetic Resonance

PPI……...…Protein-peptide interaction

PES…….Potential Energy Surface

REST……...Representational State Transfer

SASA……Solvent accessible surface area

SSD……..Single-state design

TN……True negative

TP……..True positive

Chapter 1. Introduction to Protein Binding and Specificity Profiles

As this thesis relates advances in protein-peptide interaction specificity predictions, this first chapter lays grounds for the significance of the work herein. A brief introduction on how protein-peptide interactions are characterized and their importance in systems biology will be followed by a short overview of current experimental techniques deployed to assess specificity profiles. These methods have inherent limitations, which enlighten our motivation for the presented work.

1.1 Protein Structure

Proteins are long polymers synthesized from 20 different monomers units, known as amino acids (Figure 1.1.1.). Amino acids consist of small organic molecules with an average molecular mass of approximately 100 Da. Amino acids are composed of an amine and a carboxylic acid functional

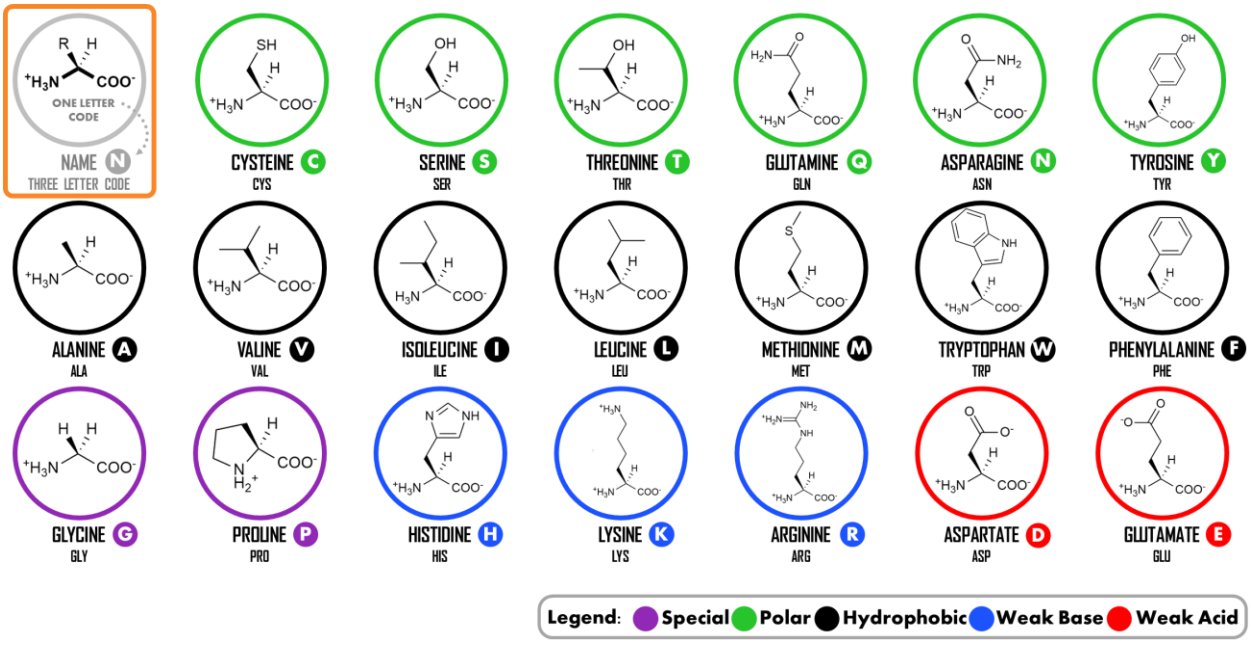


Figure 1.1.1. The 20 canonical amino acids. Each member is identified as by a color, representing the type to which it belongs. Yet, many classifications of amino acids exist, however the specific categorization shown here is used

throughout the thesis. Amino acids differ by their side chain, which all point in the same direction yielding the same L-isomer.

group linked together by a carbon atom (Cα). Those atoms compose the backbone. This proximity of a basic and an acidic group in the backbone leads to a zwitterionic species at a biological pH (7.4). Each amino acid has a different arrangement of atoms attached to their backbone Cα, designated as the side chain. The carbon stereocenter thus generated is levogyre (L) for each amino acid, apart from glycine, which does not have a side chain. Usually, amino acids are clustered in 5 groups based on the properties of their side chain: polar, hydrophobic, acidic, basic and special.

Proteins are polymers of amino acids linked via amide bonds. Their length may vary from few amino acids (peptides) to hundreds. On average, human proteins are made of 375 amino acid residues (L. Brocchieri & S. Karlin, 2005). Protein length can even go above 1000 residues, or as little as 50. Below 50 amino acids, the chain is often no longer designated as a protein, but as a peptide. Proteins, as unfolded amino-acid chains are unstable. Thus, they spontaneously collapse in a folded state. The conformational space available during folding is large, since bonds linking each amino acid are free to rotate in space. Despite the numerous folding pathways possible and conformational space to explore, most proteins repeatedly adopt the same 3-dimensional structure, designated as their native state, a process that can happen in as little as 700 ns (Kubelka J & al., 2006). Considering that folding time frames seem decoupled from the folding space magnitude, it was postulated (Levinthal C., 1969) that proteins do not explore the whole conformational space and that there must be some deterministic mechanism of folding that would allow the protein to converge towards its native state. Therefore, various models were proposed to explain the sequence of events leading to a folded native state. The diffusion-collision model (Karplus, M. & Weaver,

D.L, 1976, 1994) suggest that neighboring amino acids would rearrange in smaller groups via non-bonded interactions, thus generating multiple local three-dimensional structures. Then, local structures would aggregate due to diffusion and collisions, forming the protein's native state. A nucleation-condensation model, on the other hand, suggest that a local rearrangement is slow to initiate, but once a portion of the molecule folds, neighboring amino acids are suddenly constrained into a given conformation which facilitates the folding process and the rearrangement propagates quickly along the polymeric chain, generating a folded native state (Nolting B. & Agard, D. A., 2008). The hydrophobic collapse model (Dill K. A., 1985) propose that proteins first collapse into an unfolded state prior to any higher degree of organization. Since proteins are mainly hydrophobic—more than 74% of the residues are not charged, according to the UniProt database as of April 2013—it is hypothesized to collapse, for the same physical reason that oil droplets aggregate spontaneously in water. Following the collapse, the conformational space is greatly reduced, thus allowing the protein to fold and rearrange into its native state in a reasonable time frame.

These classical models are not so different from one another. They are fundamentally describing the same concept of folding kinetics and energetics, but with different perspectives. This concept is the general principle of energy minimization of a thermodynamic system, which is often represented as an energy landscape, or potential energy surface (PES).

1.2 Energy Landscapes of Macromolecules

The energy landscape of a protein is a concept that was first suggested in the 1970s, but more generally accepted fifteen years later (Frauenfelder H. et al., 1991). The protein energy landscape is an 'hypersurface' describing a protein's potential energy in relation to the conformation of its

atoms (Figure 1.2.1.). Despite the "hyperdimensionnality" of the protein energy landscape, it is commonly represented as a simplified three-dimensional surface. On that surface, wells represent lower energy conformations, whereas hills correspond to unstable intermediate states. This is highly analogous to the PES as described in the transition state theory (TST) proposed by Eyring in 1935—during chemical reactions reactants must overcome an energy barrier (activation energy, E_a) where molecules adopt a highly unstable transition state, before forming the more stable products. The concept of the energy landscape, for both small and macromolecules, is that the system tries to minimize its global free energy (ΔG, EQ **1.0**) eventually getting trapped in a stable conformation (well). The Gibbs free energy is described by an enthalpic and entropic energy term. The thermodynamic relationship

$$\Delta G = \Delta H - T\Delta S \qquad \text{EQ } \mathbf{1.0}$$

Where ΔG, ΔH and ΔS are changes in free energy, enthalpy and entropy from folded to unfolded states. As proposed by Josiah Willard Gibbs in 1873, ΔG approximates the chemical potential, which is minimized when a system reaches equilibrium. ΔG is independent of any path taken in the energy landscape, and only depends on the initial and final states of the system. Therefore, the Gibbs free energy relates directly to a system's global stability.

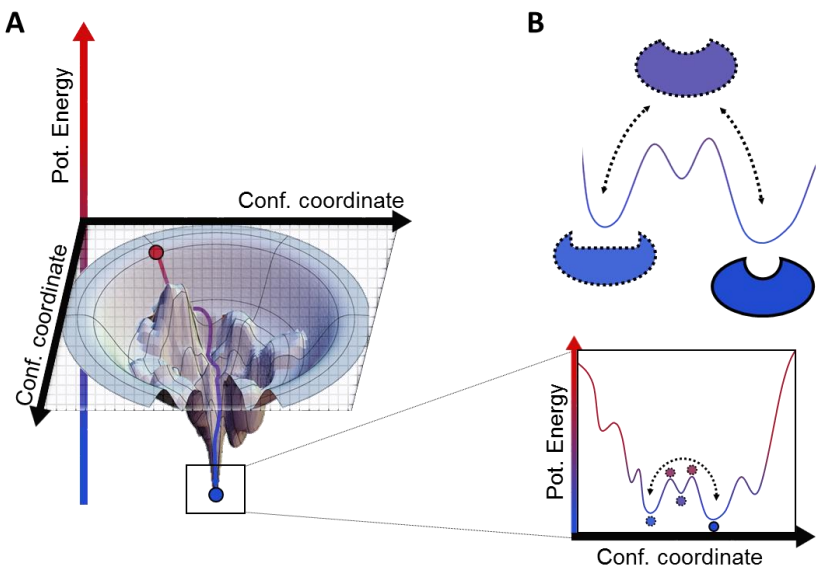


Figure 1.2.1. Protein folding and motion dynamics on the energy landscape. A) The folding funnel model, where the potential energy is represented as a landscape. During folding or binding, proteins are moving down the energy landscape towards a more stable conformation. Energy landscape adapted from Dill & Chan, 1997. B) The dynamic landscape, where protein, in their folded state, can oscillate between various stable states.

As in EQ **1.0**, the Gibbs free energy is composed of an enthalpic (ΔH) and entropic (ΔS) contribution. It consists of non-mechanical work applied to the system summed by the heat supplied to it. As a chemical bond releases heat when formed, protein folding is usually associated with a negative ΔH, as non-covalent bonds form between amino acid residues during folding. The enthalpic contribution is difficult to estimate in macroscopic systems, since ΔH relates to the global internal energy of the whole system, including solvent. Therefore, contribution of bond forming and breaking within solvent, but also within the protein needs to be considered as well for the overall enthalpic contribution to the system.

On the other side, entropy describes the number of microstates a system can adopt. It measures the randomness or disorder of a system. Entropy increases during an irreversible process and does not

change within perfectly reversible processes. It never decreases globally and is the only physical property that is directional. It relates to the loss of conformational flexibility of rotatable bonds upon rigidification (entropy loss), but also to the gain of microstate available to solvent molecules thrown back into the bulk, also known as the hydrophobic effect (entropy gain). This never-ending balance in opposite entropic contributions upon folding makes entropy very complex to evaluate. As solvent is much more abundant than the protein under consideration, its entropic effect is non-negligible, however, very little is known as to how solvent molecules interact with proteins, which makes its estimation challenging. For all the above reasons, the entropic-enthalpic compensations makes it difficult to distinguish between enthalpy-driven versus entropy-driven chemical processes. (Dunitz, 1995; Gilli et al., 1994).

The ruggedness of the protein energy landscape describes the energetically accessible conformational states of a protein. A flat energy landscape contains many local energy minima (shallow wells), each readily accessible to the protein, which dynamically oscillates between these minima since it has enough energy to overcome unstable intermediates. Such a landscape would not solve the Levinthal paradigm, as too many conformations are accessible, slowing down the process of folding, but also by preventing the protein from getting trapped into a specific well (global energy minimum). Since proteins are known to fold quite fast into their native state, the funnel landscape model was proposed, where a deep well would shape landscape funnelling the protein conformational search into a global minimum quickly. According to the diffusion-collision model, the conformational search would be driven by large free energy drops during the initial stages of folding. A long polymer would adopt local secondary structures (α-helices and β-sheets) quickly, due to non-bonding electrostatic interactions lowering overall free energy (ΔG). These local arrangements then fold into a tertiary structure. As opposed to the diffusion-collision model,

the hydrophobic collapse folding model suggest that the protein quickly minimizes its free energy by first collapsing into a “molten globule”, where hydrophobic side chains are found in the core of the protein. The driving force of this folding is mainly the maximization of the solvent entropy, where the polar solvent (water) forces a collapse the hydrophobic residues together. Polar/charged residues tend to stay at the surface, thus stabilizing the interface with water by offering H-bond donors and acceptors (Kauzmann W., 1954, 1959; Lum K. et al., 1999; Stillinger F. H., 1973; R. Zhou R. et al., 2004). This quick collapse is then followed by slow local rearrangements forming isolated secondary structures. Both models represent extreme and opposite views of the initial driving forces of protein folding. The reality, most probably, is a mixture of these models where the energy is dependant on the protein amino acid sequence.

1.3 Protein binding

Proteins fulfill their biological role through interaction with other molecules (Figure 1.3.1), such as DNA and RNA during replication and reparation processes, small organic and inorganic molecules for molecular recognition of enzymatic conversion, proteins and peptides for molecular signaling, ions, gases for cellular transport etc. Binding is thus one of the most fundamental biological concepts. To fully understand these complex processes, deeper understanding of molecular recognition is essential. Basically, it is the mechanism by which a protein and a ligand associates to form a non-covalent protein-ligand complex.

$$P + S_1 \rightleftharpoons PS_1 \qquad \text{EQ } \mathbf{1.1}$$

$$K_{ass} = [PS_1] / [P][S_1] \qquad \text{EQ } \mathbf{1.2}$$

$$\Delta G = -RT \ln K_{ass} \qquad \text{EQ } \mathbf{1.3}$$

$$\alpha = [K_{ass}][P] / \sum_1^i K_{B_i}[B_i] \qquad \text{EQ } \mathbf{1.4}$$

A ligand (S) binds to a protein (EQ 1.1) with affinity directly related to the difference in free energy from bound to unbound states (EQ. 1.2, 1.3). Higher free energy difference (ΔG) translates to a larger association constant (K_{ass}). Another important concept of protein binding is specificity (α), which relates the ratio of K_{ass} multiplied by the target protein P's concentration to the sum of all other association constants , K_{Bi}, multiplied by the concentration of all other interfering binding events with proteins B_i (EQ 1.4, Eaton B.E, Gold L. & Zichi A.D, 1995). A protein is said specific if only a handful of ligands display high affinity upon binding. Low

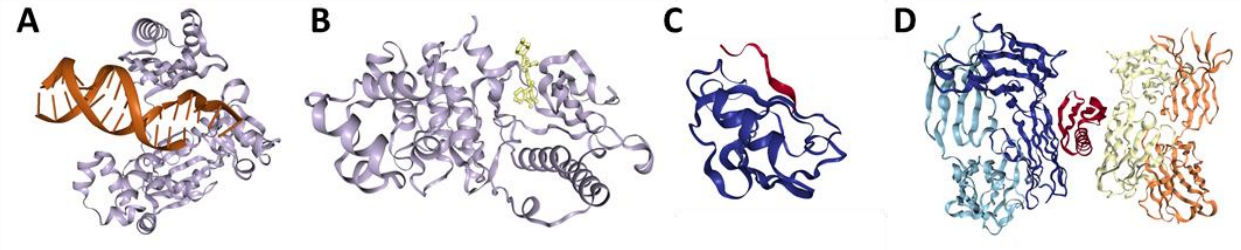


Figure 1.3.1. Various types of substrates binding to proteins. A) DNA, in orange, binds to proteins during the DNA replication process (Polymerase- damaged DNA complex, PDB ID 4Q45). B) Benzodiazepine, in beige, bound to the protein ERK5, a MAP kinase (PDB ID 5BYY). C) C-terminal peptide of the ErbB2 tyrosine kinase receptor, in red, binds to the Erbin PDZ domain (PDB ID 1MFG). D) Bacterial protein (PpL), in red, bound to a human antibody (PDB ID 1HEZ).

specificity affects the binding kinetics as the protein's effective concentration is lowered for a given substrate. In other words, affinity usually refers a change in free energy upon ligand binding, whereas specificity is a ratio of target binding to the sum of off-target binding. To explain the protein-ligand association mechanism and how protein and ligand complexes are more stable from their respective unbound sate, various models have been proposed.

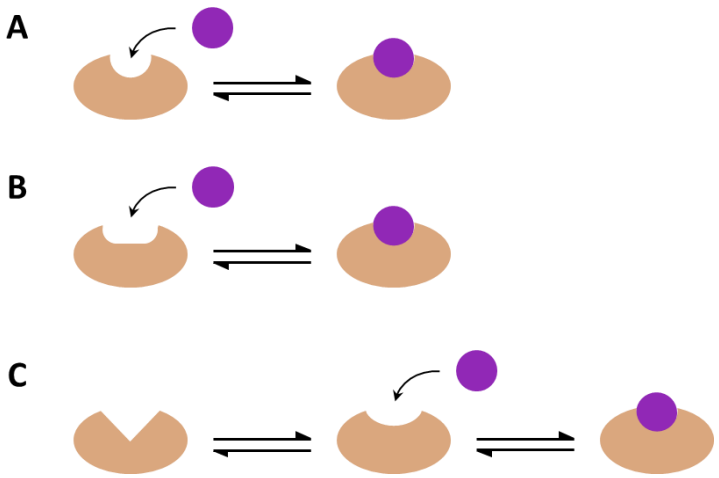


Figure 1.3.2. The proposed binding models. A) Lock and key model, where the protein and peptide both exist in a static shape, both complementary, permitting snug fit binding. B) Induced-fit model, where the protein pocket has a similar shape as its ligand, which upon binding, induces a conformational change, thus enhancing the binding interactions. C) Conformational selection model where the protein is a dynamic entity that exists under various conformation. A pre-existing equilibrium takes place between favorable and unfavorable conformations. The ligand selecting the favorable conformation for binding pull the equilibrium towards that state.

The first model to be proposed, back in 1894 by Fischer, described that two rigid entities could assemble only if both partners have complementary shape, just like a key fit into a specific lock. This lock-and-key model (Fischer E., 1894, Figure 1.3.2.A), where both partners can associate based mainly on shape complementarity is proposed to be driven by a large entropy gain, as both partners are considered individually rigid in the initial state and end state (no significant

conformational degrees of freedom loss upon binding.). Thus, the hydrophobic effect (expelling water molecules from the binding pocket back into the solvent bulk) becomes important in this model.

The most important premise to the lock and key model is that a key with incorrect shape cannot fit into a lock, since both cannot change their shape. However, proteins are known to be dynamic, as described by their energy landscape, and such a model could not explain observed binding of non-complementary protein and ligands. Therefore, an induced-fit model (Koshland D. E. J., 1958, Figure 1.3.2.B) was proposed, where non-bonded stabilizing contributions upon association of the ligand would force the protein to adopt a complementary shape. This conformational switch, initiated by the ligand, is suggested to be enthalpy-driven since the van der Walls and electrostatics contribution must overcome the entropic cost of association. The significant enthalpic barrier needed to induce conformational change explains why this step is the bottleneck of binding.

Proteins are dynamic to a certain extent as described by the ruggedness of their energy landscape (Figure 1.2.1. B). In their native state, the ambient thermal energy provides enough kinetic energy to overcome unstable conformational intermediates and adopt a different shape of similar free energy. Even though protein movements can be drastic, usually, dynamics consist of minor changes in the protein global shape like loop rearrangements, hinge-like motions, compression and decompression of barrel-shaped helices, etc. Dynamical movement can also describe a simple motion like a single or concerted side chain rotation. From this dynamical standpoint, a lock-and-key model would correctly represent a rigid system, whereas the induced-fit model would be more adapted to a dynamic system. However, this model could not explain large movements induced by

a ligand upon binding in various cases where proteins adopt considerably different shapes in the bound state.

To better explain protein binding of highly dynamic systems, a refined model (Frauenfelder H. et al., 1991; Ma B. et al., 1999; Tsai C. J. et al., 1999a) was proposed. Instead of a protein natively trapped into a non-binding conformational state, it is suggested to constantly oscillates between stable favorable and non-favorable binding modes. This movement decrease the entropy cost of binding, since the desired conformation can readily be adopted by the protein. Upon binding, the protein oscillation between states considerably diminishes, and gets trapped into the favorable conformation. Therefore, the conformational state change is not initiated by the ligand, but the thermodynamic equilibrium is pushed towards the favorable binding mode, due to a larger free energy drop in such state. In such a model, the bottleneck step is believed to be the protein`s structural rearrangement towards a more favorable conformation.

Overall, these models explain how proteins and peptides can explore the energy landscape to form a lower free energy complex. Higher affinity complexes would show a larger drop in free energy from unbound to bound states. This is analogous to protein folding, where the most stable fold corresponds to the deepest well in the landscape. In fact, there are no difference conceptually between folding and binding. During folding, amino acid residues rearrange according to enthalpy-entropy compensation, expelling solvent from the interface, creating non-bonding interactions, thus maximizing the free energy drops and rapidly exploring the bottom of the landscape funnel. The process is the same whether the residues are linked in the protein chain or not, as during binding of individual peptides. An energy landscape represents each atom and the energy of the whole system depending on pairwise inter-atomic distances for each atom—the distinction

between a bound and an unbound atom pair is already accounted in their energy contribution to the system. Therefore, in a given atom system described by an energy landscape, there exist states where atoms are close enough to form one single entity (e.g. protein-ligand complex, covalent bonding), and states where sets of atoms are distant enough to form distinct entities (e.g. unbound protein and ligand states, chemical bond breakage). Nonetheless, whether atoms are arranged into single (bound state) or distinct entities (unbound state), the same energy landscape describes both states simultaneously since the atoms considered in the system itself did not change, but their relative distances. Therefore, the same thermodynamic concepts can apply for protein folding and binding.

Overall, understanding of protein binding affinity, to a point where it is possible to determine *a priori* if a ligand can bind to a protein, would require a reconstitution of the energy landscape of the system to virtually explore it to find low energy wells, representing stable complexes. Furthermore, since a protein have a function to fulfill, this means that, from all the possible stable complexes on the energy landscape, only one has the desired state producing the intended effect. Unfortunately, this information must be obtained by complementary methods, since the energy landscape does not consider the whole metabolic pathway in which a protein is involved.

1.4 Investigating Protein-Ligand interactions

Specificity confers proteins their unique biological role allows an organism to develop a complex metabolism and to adapt better to their environment. Mapping a new protein in the network is a challenging task but studying natural binding partners of a protein can help drawing conclusions as to which pathways might be relevant with the target protein. Many peptides and proteins can only be found in certain organelles or area of an organism, or might take part in specific pathways,

which can help relate an enzyme to its biological function. Metabolic pathways are complex and being able to map, link and establish relationships between its constituents is crucial for better design of drugs (Apic G., 2005), metabolic engineering and biological assay development (Tomar N., De R. K., 2013).

Discovering new natural ligands of a protein is not an easy task. The sheer number of possible partners in an organism requires high-throughput synthesis and screening to be able to experimentally validate binding of a molecule. It is necessary to reduce the number of molecules to validate. A first step towards this direction is to understand the rational behind specificity and affinity of a protein towards its ligands as a mean of generalizing the mechanism to ligands. As mentioned in the previous sections, proteins fold into a native 3-dimensional structure as described by their conformational energy landscape. This intrinsic shape and charge distribution allow protein to bind with high affinity to a small subset of ligands such as other proteins, peptides or small molecules in their environment.

X-ray crystallography, nuclear magnetic resonance (NMR), small-angle X-ray scattering are often use for rationalizing protein-peptide interactions (PPI) since those provide visual representations of a system. Crystallography is currently amongst the methods of choice used for PPI study and analysis. From a protein crystal, it is possible to get a grasp at the types of interactions that are made at the interface and the conformation adopted by both partners. Enthalpic contributions can be observed from non-bonded interactions intra and inter chains. Entropic information can also be extracted from crystal structures indirectly by looking at electron density maps or B–factors. Another strength of crystallography consists of the ability to easily identify binding pockets and interfaces where the protein interacts with a ligand.

However, crystal structures do have limitations arising from their acquisition procedure. X-ray crystallography requires low thermal motions of atoms for higher signal to noise ratio. To achieve such results, protein crystals are frozen at -196°C. At that temperature, previously accessible conformational state on the energy landscape are unreachable. Thus, crystallography is generally described as of a picture of the complex frozen in time. Since the conditions for taking this "picture" are harsh and far from typical, it is not possible to tell if the protein crystallized in its native state. Moreover, peptides and proteins are dynamic, as described in the conformational selection binding model. This means that conformation of these polymer changes over time, and consequently, the "pictures" of the protein frozen in time do not inform much on protein dynamics and alternative conformations. Some studies have shown that a more thorough analysis of low electron density maps of crystallized structure might reveal previously hidden information about dynamics (Thompson M.C & al., 2018), but these advances are just emerging and not widely used. Nevertheless, crystal structures are a good start for studying a protein's tertiary structure. This allows for protein-protein interactions (PPI) analysis by identifying binding interfaces and non-bonded contributions, which is essential for rational drug design and protein design, where structure plays a considerable role for understanding binding mechanism. However, crystallography is not viable to study multiple states along a chemical reaction or binding event mainly due to relatively fast kinetics of such events. A workaround commonly use is to crystalize inactivated proteins—where the key residue is mutated—with the substrate, or to synthesize a ligand analogue, which would resemble the products, without being released as fast from the binding pocket. These alternatives offer the possibility to approximate the state of the real protein-ligand by identifying the state of a highly similar protein-ligand complex.

Various other methods for investigating PPI's include Isothermal Titration Calorimetry (ITC) and MicroScale Thermophoresis (MST) can determine association of a ligand and a protein by calculating heat variation. Protein-peptide systems release energy upon association to form a stable complex, and this energy difference can be measured to obtain enthalpy (ΔH) of binding and the association/dissociation constant (K_d). Despite the accuracy of these methods, they are limited by the sheer amount of protein-ligand analysis that can be performed per unit of time. Considering the combinatorial size of the proteome, those methods are not viable for large input datasets. Thus, a common workflow is to perform a pre-screening of the dataset for cheaper and faster qualitative detection of active peptide-ligand pairs.

1.5 Specificity Profile

One way of filtering a dataset is to screen against a protein specificity profile using a pattern matching algorithm (Figure 1.5.1.). Such specificity profile represents favorable amino acids at

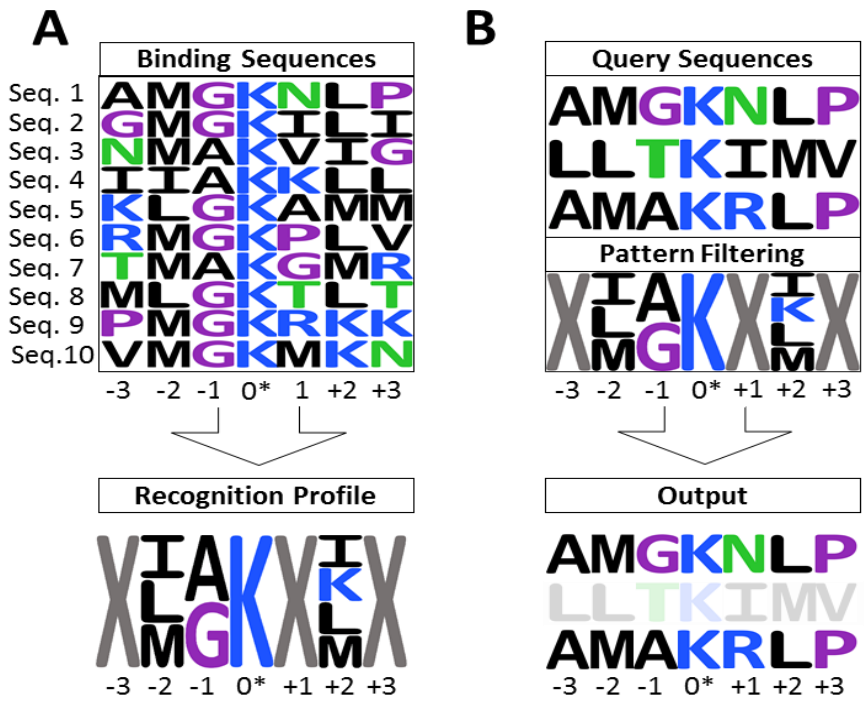


Figure 1.5.1. A typical recognition motif for a protein obtained from known binding sequences used for screening of dataset. A) Combined preferential amino acids at each position flanking a residue of interest (denoted by "0*") are displayed. B) Filtering a dataset's peptide sequences from the previously obtained recognition motif. Each amino acid of a sequence is compared to the motif's allowed amino acids at that position. Is on amino acid along the sequence does not match any of the allowed amino acid, the sequence is discarded (shown in light grey in the output of the filtering). This should yield promising binding sequences and discards non-matching peptides.

each position across a peptide sequence. Non-specific positions are residues that can be permuted to most amino acids without affecting molecular recognition. Such position is denoted by a single "X" in the profile. These motifs are built from prior experimental knowledge of binding peptide sequences which are aligned and compared position-wise (Figure 1.5.1A). Each amino acid

extracted from the same relative position of every binding peptides are merged and represented into a profile. This process is done at each position to yield a compact representation of all binding information into a recognition motif. The information related to the exact sequence of each peptide is lost when merging. They can be retrieved by proper recombination of amino acids, but new peptides sequences can be generated from random combinations as well. Each possible sequence generated is a matching peptide sequence (Figure 1.5.1B) and is inferred to potentially bind to the parent protein.

Gaining binding information towards a protein can be done via permutation peptide arrays (Figure 1.5.2). A peptide array consists of many different peptide sequences chemically bonded to a cellulose surface, laid as a matrix, which will be soaked in a protein buffer solution for affinity and reactivity testing (Volkmer R., Tapia V., Landgraf C., 2012). Single-point mutant libraries (matrix) offer low sequence diversity but are useful to study in isolation the effect of a single amino acid mutation on the affinity with the protein. Combinatorial mutations offer greater sequence coverage but require bigger libraries for relevant data analysis and statistical significance. Nonetheless, such libraries allow for discovering more complex interactions between pairs of amino acids that might be required for binding, which is not possible with single-point mutations. On the other hand, single-point libraries, due to their manageable size (20 amino acids × peptide length) and analysis simplicity, are commonly used for determining a recognition profile.

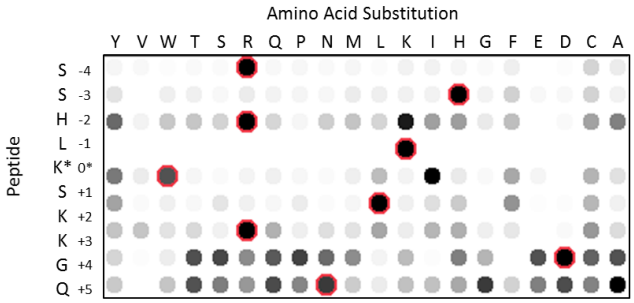


Figure 1.5.2. Typical peptide array. Each position in the peptide (represented as a column) is permuted to every 20 amino acids. The wild–type amino acid is circled in red. Activity response is colored from white to black for low to high response, respectively. For each position (row), any amino acid mutation with a response of at least 50% of wild–type sequence is included in the recognition motif.

In a simple functional assay, wild type amino acids along a peptide sequence are mutated to every other 19 amino acids at each position. Overall, a library of size $20 \times \mathbf{n}$, where **n** is the number of positions in the peptide, is generated and accounts for all single-point mutants of the native peptide. The peptides are then soaked with a protein solution buffer and washed after a period, which vary depending on the system. The array is exposed to reveal peptides that bound to the protein, as spots. The exposition mechanism is specific to the assay, but light exposure and radioactivity counters are the most common revelation methods. In these cases, the spot intensity (shade) can represent radioactive decay (Figure 1.5.2.) or chemiluminescence (Landgraf, C. et al., 2004). The output signal from the array, can be compared to other sequences within the array to rank mutations in terms of protein-peptide binding preference. This ranking provides a qualitative measure of binding and is used to generate a recognition profile for the protein.

Although SPOT peptide arrays are well documented methods (Frank R., 2002), processing of the raw output signal for motif extraction is not standardized since authors use different methods: relative factors (Lanouette S. & Davey J. A., 2015) (EQ 1.5), discrimination factors (Rathert P.,

Dhayalan A., & al., 2008) (EQ 1.6), and wild–type raw output signal cut-off (EQ 1.7, current work).

$$\mathrm{AA_i} = S_i \,/\, (S_i - \overline{\textstyle\sum_{j=1,j\neq i}^{k} S_j}) \qquad \text{EQ } \mathbf{1.5}$$

$$\mathrm{AA_i} = (S_i / \overline{\textstyle\sum_{j=1,j\neq i}^{k} S_j}) - 1 \qquad \text{EQ } \mathbf{1.6}$$

$$\mathrm{AA_i} = S_i / W_i \qquad \text{EQ } \mathbf{1.7}$$

EQ 1.5 and EQ 1.6 are similar methods of calculating a score for an amino acid AA at position i of a peptide, where the score at i corresponds to the ratio of the spot intensity S at i over the average intensity of all other positions j. In EQ 1.7 the score at position *i* is evaluated as the ratio of the spot intensity *S* at *i* over the spot intensity of the wild–type *W* amino acid at position *i*.

When all scores are computed, a threshold is used to discriminate binding from non-binding amino acids. There is no consensus yet regarding how the cut-off should be determined. This mainly stems from the fact that we cannot define a relative value for a binding or non-binding compound. What reaction rate and binding affinity defines a ligand? The actual values are system-dependant, but is there a universal relative value that can be used in every protein-peptide system? In EQ 1.5 & EQ 1.6, the standard deviation from average score is used as a metric for discriminating amino acids. This method has the drawback of possibly eliminating the wild–type amino acid from the recognition motif, depending on the initial distribution of scores. Cut-off based on population's score average, especially when population is low—size of 20 for peptides—depends on the score of the rest of the population since extreme scores have more impact on the average than a larger population would. In EQ 1.7, a percentage of the wild–type amino acid sequence's score is used as a cut-off instead of the average. This way, amino acid scores are independent—a score does not

influence the cut-off outcome of others. On the other hand, this method is influenced by the input wild–type sequence. A poor binding wild–type sequence would yield mutant scores higher than wild–type residue, leading to a tolerant recognition motif, where as a high binding wild–type sequence would yield lower mutant scores, leading to a stringent profile.

1.6 Simulating permutation peptide arrays

Permutation Peptide arrays, as a method for generating a protein's recognition profile, have some limitations. First, the binding sequences derived from a peptide array is often biased towards the wild–type sequence template used since only single-point mutant of that native sequence are tested. Consequently, care must be taken when choosing the initial binding peptide as it may influence the profile output. Generating single-point mutants allows for a fast experiment by limiting the sequence space to explore. A combinatorial peptide array (combinatorial mutations) would require an exponential amount of sequences to be tested ($20^{\mathbf{n}}$, where **n** is the number of positions to evaluate) to evaluate the complete combinatorial space, which is not feasible experimentally. Inspired by this combinatorial concept, some methods such as yeast two-hybrid (Y2H) (Young K. H., 1998) have been developed that allow for broader sequence space exploration, but still do not cover the full combinatorial space.

Another limitation of peptide array is the cost associated with an experiment, in terms of resources, and equipment. Producing quality results is a system dependant iterative process. The binding profile is highly dependent on the reaction conditions, including time of incubation and buffer salts used. Several arrays reproducing the same protein-peptide system have shown different profile results, notably SMYD2-p53 published by both Lanouette S., Davey J. A., 2015 and Kudithipudi

S., 2012 respectively, (Figure 1.6.1 A), and the protein-peptide complex SET8-H4K20 by Biggar K., 2016 and Kudithipudi S. et al. 2012 (Figure 1.6.1 B).

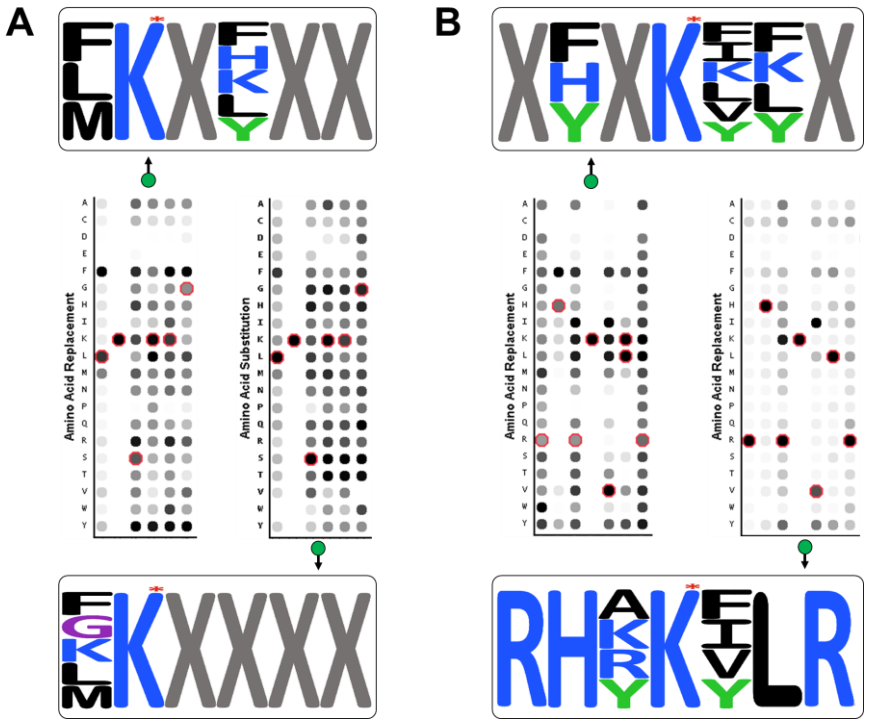


Figure 1.6.1. Peptide arrays can yield variable results. Activity response is colored from white to black for low to high response, respectively. Additionally, wild–type amino acid is circled in red. A) Peptide array for SMYD2 – p53 produced by two different groups (on the left, Lanouette S, Davey J. A., 2015, on the right, Kudithipudi S., 2012). Both motifs extracted show different profiles for position +2 flanking the methylated lysine 370 (indicated by a red star). B) Peptide array for SET8 – H4K20 produce by two groups (on the left, Bigger K. unpublished results, 2016, on the right, Kudithipudi S. and al. 2012). Then again, motifs extracted show significant difference at position -3, -1 and +3 flanking lysine 20 (indicated by a red star).

Such variability in the experimental results raises a legitimate question that gave rise to this thesis. Is the investment in time and money worth the variable output from permutation peptide arrays in terms of specificity profiles? Undoubtedly, specificity profiles are successful at identifying new substrates of novel proteins (Reineke U, Sabat R, 2009 & Wu C, Li S. S., 2009 & Zhang Y. et al.,

2010), and for that reason, we propose a fast and low cost alternative method to obtain specificity profiles for pattern filtering of sequences databases. This can be achieved with a computational workflow which evaluates protein-ligand binding for generating recognition motif. Computational methods offer cheap and quick alternative to costly and tedious experimental peptide arrays. On the other hand, computational simulations of protein-peptide interactions come with its sheer amount of technical difficulties. Conformational space of proteins is so vast that it is inconceivable to simulate protein movement and binding conformational changes in appreciable time frames. Additionally, calculating physical force between protein, peptides and water is very challenging given to the tremendous amount of degrees of freedom (DOF) the system has. In addition, previously small to negligible entropy contributions now become prevalent in polymer chemistry. Moreover, calculating energy contribution from various forces for binding interaction prediction is one thing, but predicting biological response and/or activity requires knowledge of the mechanism of action.

Up to this date, scientists are still struggling to understand and properly model PPI. So far, algorithms are optimized for specific protein domains and systems and lack generalisability. Currently, we do not have the extensive data required to properly train computational algorithms and lack the knowledge to understand all the energy subtleties impacting protein conformation, stability and reactivity. Nonetheless, the next chapter demonstrates how, with current knowledge of PPI's and polymer chemistry, it was possible to successfully simulate protein-peptide binding in order to generate specificity profiles.

Chapter 2. A structure-based computational method for simulating peptide arrays

As explained in chapter 1, a common methodology for discovering new peptide ligands of a protein is to perform a permutation peptide array of every single point mutant of an input peptide to gain insight on mutations that are tolerated by the protein. This valuable information can be harvested to build specificity profiles and infer new potential binding sequences. However, this experimental method is costly, time consuming and has proven to yield variable results. To overcome these limitations, we propose in this chapter a predictive structure-based computational approach for building protein specificity profiles. This method achieved comparable accuracy to peptide arrays but at no cost and a considerably reduced time frame.

This chapter describes how, from a protein-peptide crystal complex, it is possible to extract mutations that are tolerated at the interface by performing a computational mutational analysis. We then explain how crystallization data such as B–factors, interaction maps and solvent exposure are combined for identifying tolerant positions across the peptide sequence. We then demonstrate how augmenting stability calculations with crystal properties generated accurate specificity profiles across 6 protein-peptide complexes. Finally, our computational methodology was blindly applied on SMYD3-VEGFR1/MEKK2 to successfully identify new binding partners at a 70% success rate.

2.1 Hypothesis and prior assumptions

Structure-based computational approaches are a subset of computational algorithms that rely on spatial representation of a problem for finding a set of solution. By starting from a 3D model of a protein bound to a peptide, molecular mechanics force-field calculations can be used to evaluate

interaction energies within the system. Force-field calculations are used to compare interaction energies of each optimized single point mutant of the peptide. Then, using a threshold computed from EQ 1.7 to filter the computed stability scores yields the mutations tolerated at each peptide position in the protein-peptide interface.

2.1 PHOENIX: Evaluating the fitness of a peptide sequence

Fitness, from an algorithmic point of view, is a calculated value defining how well a solution represents a model. A fitness function describes how this fitness value is computed. The fitter the solution, the better it represents the model and vice versa. Each term of the function is called a descriptor. Applied to protein binding problems, two distinct sets of descriptors are commonly used in a fitness function: statistical and physical descriptors. Statistical descriptors rely on extensive prior knowledge of a system extracted from diverse databases. Such descriptors include statistical energy functions, secondary structure propensities from primary sequence, protein family classification, etc. On the other hand, a physical descriptor does not require extensive experimental data, however, it assumes reasonable understanding of the physical laws ruling the underlying system. Those include, but are not limited to, Gibbs free energy, enthalpy and entropy. The ability to correctly model protein binding using only physical descriptors would imply a global understanding of the mechanisms of conformational transitions, motions and folding processes in the context of a solvated protein system. This is a priori a challenging multivariate problem, far from being solved. Nonetheless, decent simplified physical models approximating those systems have been developed and have yielded fruitful results (Lanouette S. & Davey J. A., 2015).

2.1.1 Scoring function

In this thesis, sequence scoring relies on a physics-based model (Mayo S. L. & al., 1990). The fitness function is composed of three non-covalent terms for describing pairwise atom interactions (A_i-A_j) within the system: a Van der Waals energy term (E_{VdW}) [Lennard-Jones, 1924] (EQ. 2.1), an electrostatic term (Gasteiger J., Marsili M., 1980) (EQ. 2.2), and an H-bonding term (Dahiyat B. I., & Mayo S. L., 1997) (EQ. 2.3). The van der Waals energy curve follows a 12-6 Lennard-Jones potential, where the maximal well depth is found at the geometric mean *r* of the Van der Waals radii of each atom R_i and R_j forming the pair. The well magnitude D is calculated from the geometric mean of the well depth parameter of each atom respectively (D_i and D_j). To account for the imperfection of the energy calculation, E_{VdW} can be dampened by modulating the α constant to reduce or increase the effective radii of the atom pair. A typical value for α is 0.9, thus allowing atoms to be closer to one another before entering the highly exponential (x^{12}) energy penalty part of the Lennard-Jones curve. This is often needed to account for discretization of atomic positions used by the search algorithm to save computation time.

$$E_{VdW}(A_i, A_j) = \sqrt{D_i D_j}\left[\left(\frac{\alpha\sqrt{2(R_i R_j)}}{r}\right)^{12} - 2\left(\frac{\alpha\sqrt{2(R_i R_j)}}{r}\right)^{6}\right] \qquad \text{(EQ. 2.1)}$$

The second energy term used is the electrostatic term. This term (E_{ele}) calculates energy between point charges of the atom pair (q_i and q_j), depending on their inter-atomic distance (r). The energy of these point charges in proteins is corrected for the dielectric constant (relative permittivity) of that environment. Studies have shown that a dielectric constant (ε) between 8 and 40 (Boas F. E. & Harbury P. B., 2007) represents adequately the protein environment. This deviates considerably from the water dielectric constant of 80.2 at 20°C (Archer D. G. & Wang P., 1990) mainly because the protein core is shielded from solvent.

$$E_{ele} = \frac{q_i q_j}{\varepsilon r} \qquad \text{(EQ. 2.2)}$$

The third energy term used is the Hydrogen bonding term E_{hbd} (EQ. 2.3). This term accounts for minor electrostatic effects between an electron rich heavy atom (donor) and an electron deficient hydrogen bonded to a more electronegative atom (acceptor). Thus, this term computes the energy contribution of an H-bond to the system depending on the ratio of the ideal Van der Waals distance R_0 of the acceptor hydrogen and the heavy atom donor over the actual distance (R) separating them, represented by a 12-10 Lennard-Jones potential with a well depth (D_0) of 8 kcal/mol. This potential is modulated by a geometrical function $F(\theta, \phi, \varphi)$ which depends on the angle formed by the atoms (θ) and the hybridization state of the donor (ϕ) and acceptor (φ) heavy atoms (Dahiyat B. I., and Mayo S. L., 1997).

$$E_{Hbd} = D_0 \left[5 \left(\frac{R_0}{R}\right)^{12} - 6 \left(\frac{R_0}{R}\right)^{10} \right] F(\theta, \phi, \varphi) \qquad \text{(EQ. 2.3)}$$

In addition, the energy calculation includes a solvation penalty term E_{solv} (EQ 2.4), which models the solvent contribution to protein stability. The solvent is not explicitly modelled as water molecules, but instead treated as continuous medium (implicit). The solvent contribution is calculated for each residue, based on three terms: a stabilizing term for buried surface area of non-polar amino acids ($A_{np,b}$), a penalizing term proportional to exposed surface area of non-polar amino acids ($A_{np,e}$) and a penalizing term for surface area of buried polar amino. The terms can be modulated by changing $\kappa, \sigma_{np}\ and\ \sigma_p$ which have default values of 1.6, 0.026 and 1 respectively.

$$E_{solv} = -(\kappa + 1)\sigma_{np} A_{np,b} + \kappa \sigma_{np} A_{np,e} + \sigma_p A_{p,b} \qquad \text{(EQ. 2.4)}$$

Implicit solvation was chosen for its significantly lower computation time, and implementation complexity. Accurate modeling of water molecules, due to their numerous degrees of freedom (e.g., rotation or translation) requires extensive sampling and a computation framework supporting movement of solvent molecules during side-chain optimization to prevent solvent-protein collision and clash. Water molecules are easily displaceable by side chain when forcing conformational side chain rearrangement.

This final fitness function is applied to the structural model to evaluate and optimize each single point mutations performed on the wild–type sequence (Figure 2.1.1 A). It is done in 2 steps. First, the fitness function is applied to evaluate the energy of the side chain with the rest of the structural model that is not being optimized (one-body energy evaluation). Lastly, it is applied to the side chain in the context of the other side-chain conformations of the model that are to be optimized (two-body energy evaluation). The sum of these energy evaluation corresponds to the final fitness value for a structure. Since we are only interested in simulating single-point mutations, we constrained the system to the interface between the peptide and the protein to limit the number of pairwise interactions to compute.

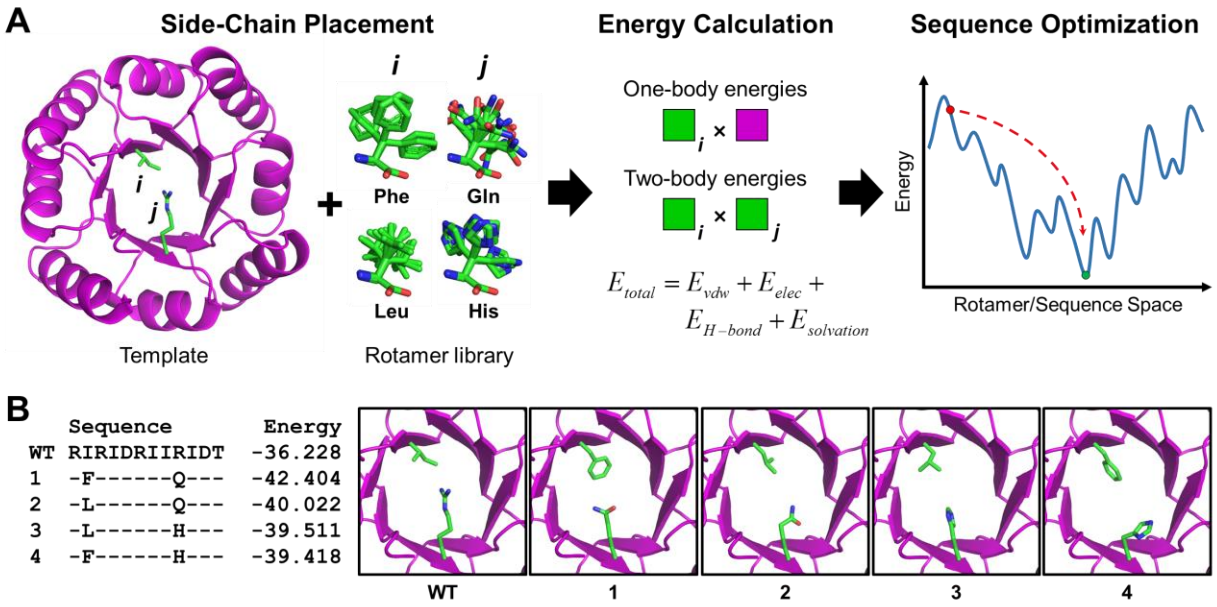


Figure 2.1.1. Computational protein design (CPD)'s mutational analysis workflow. A) An energy matrix is built by evaluating the contribution in energy off each user-defined side chain individually (green) in the context of the template (pink)—one body energy. Then a second energy matrix is generated, summarizing the contribution in energy of all pair-wise interactions up to an arbitrary distance involved with the user-defined positions. Then a sequence optimization process takes place to refine an input structural model. When a combination is tested, its energy is quickly retrieved in the energy matrices and summed up to obtain a global fitness score. This allows for quickly finding a low energy structure despite the vast sequence space. Figure is adapted from St-Jacques A. D., Gagnon O. & Chica R. A., 2018.

2.1.2 Search algorithms

Full enumeration of the solution space is often too large for evaluating with the fitness function each single possibility in a decent amount of computational time. Consequently, a brute force solution for finding the fittest solution amongst all possibilities (Deterministic approach) is not viable. Evaluating the smallest subset of solutions, while preserving reasonable odds at finding the most fit solution is desired (Heuristic approach), as it drastically improves computational efficiency.

Following this idea, a search algorithm is required to find the most stable overall conformation upon mutating a residue since the number of side chain configurations to evaluate can rapidly become intractable. For example, if each rotatable bond of an amino acid side chain would be allowed to adopt continuously each torsion angle, an infinite number of configurations would have to be evaluated. A common way to get around this is to consider only the most stable rotamer configurations of amino acid side chains. The Dunbrack backbone independent rotamer library (Dunbrack R. L. & Cohen F. E., 1997) is the rotamer library used in this work. This library contains the most stable rotamers of each amino acid extracted from crystallographic data, independently of the secondary structure and the phi-psi of the residue. This considerably reduces the combinatorial space and speeds up the calculation process. When considering many side-chain rotamers in a design, the combinatorial explosion often requires the use of an efficient search algorithm. To efficiently converge to a reliable solution in a reasonable time-frame, various stochastic algorithms have been developed. In this thesis, the Way-FASTER (Allen B. B. & Mayo S. L., 2006) search algorithm is used for convergence to a reliable solution.

Way-FASTER is a stochastic search algorithm consisting of 3 steps. The first step is a combination of a Monte Carlo (MC) (Metropolis N. & al., 1953) search and simulated-annealing (SA) (Kirkpatrick S. & al., 1983) and its goal is to stochastically identify the lowest energy combination of rotamers for the initial structure to be refined in the optimization process. The second step is called iBR (Figure 2.1.2 A), for “iterative branch relaxation” where each position is optimized independently and then combined into an intermediate solution. The process is performed again, until the output structure does not yield a lower energy than the input structure. When all positions have been optimized, the model is updated and the iBR process is repeated until no lower energy configuration are found. Then a stochastic version of the iBR, named ciBR (conditional iterative

branch relaxation) takes over, which basically consist of the same steps as iBR, but each new configuration is rejected at a 20% rate. At convergence of ciBR, single or double perturbation relaxation (sPR, dPR) is initiated (Figure 2.1.2 B). As opposed to iBR, one or two rotamers are held fixed, while the rest of the sequence is optimized. Way-FASTER then returns the "FMEC", or Faster Minimal Energy Conformation, and a list of all other structures evaluated, ranked based on their computed energy.

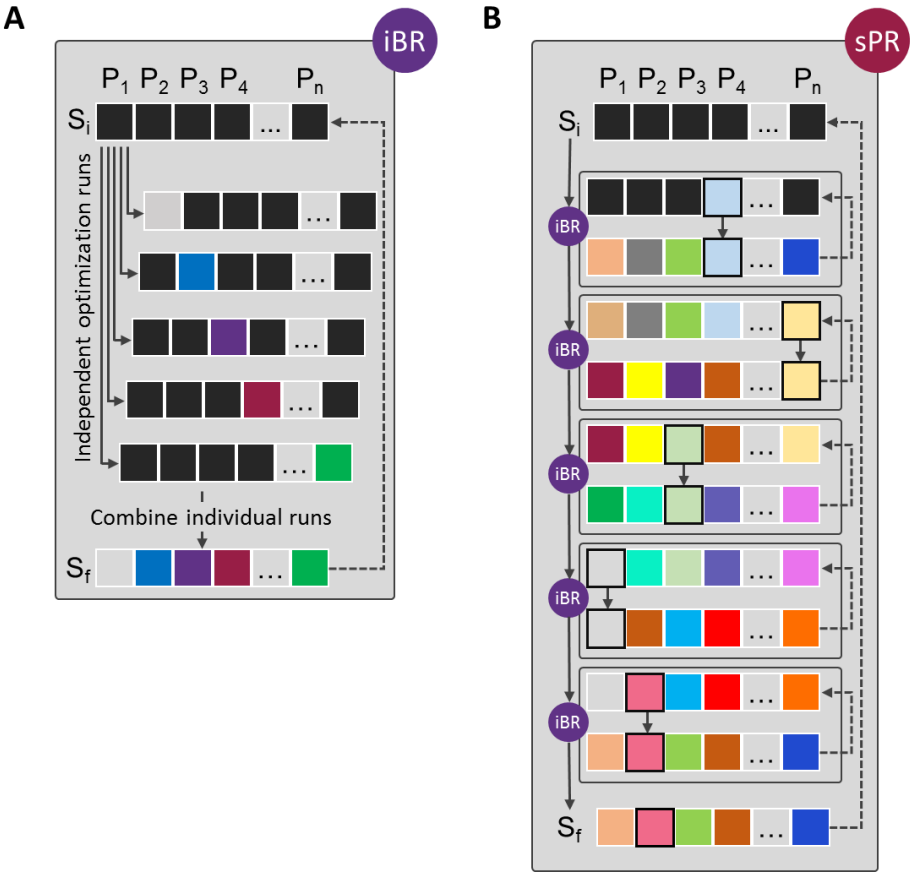


Figure 2.1.2. FASTER implementation in PHOENIX. The two modules iBR and sPR demonstrate how FASTER optimizes rotamers of a given initial structure with sequence S_i towards a FMEC (FASTER minimum energy conformation) of sequence S_f. A) During the first phase of iterative batch relaxation (iBR), the algorithm performs a loop iteration over each position P_x targeted for optimization. During an iteration, the position is optimized to the lowest energy rotamer from the library of rotamers specified by the user. Then, the structure is set back to its initial

state and the next position P_{x+1} is optimized. Each box represents an amino acid at P_x. Boxes a colored according to their amino acid identity. Boxes colored the same from one stage to the other indicated by an elbow arrow represent the same rotamer pose. The loop terminates when all positions where optimized independently and then a new structure is generated by combining all the optimized solutions from each iteration. From this new structure, the iBR module is done a again, until the new structure generated is equivalent to the initial structure (convergence). At this point, the optimized structure goes through the sPR module for further optimization. B) sPR consists in looping over all targeted position P_x of a sequence S_i in a random order. The first randomly chosen position is changed to every other rotamer sequentially. Between each perturbation, the remaining positions are optimized with iBR (A). The structure obtained at this stage replaces the initial perturbed structure if its computed energy is lower. For clarity, only one cycle is shown per position P_x, but the dashed arrow indicates that the cycle is repeated for all amino acids. Outlines boxes illustrates that the position is not optimized from one stage to another. When rotamers have been exhausted for one position, the same workflow is used on the next randomly chosen position. The process ends when a cycle of sPR on all positions did not yield any improvements in energy of the initial structure.

Up to this point, we have a defined fitness function based on physical terms to so score a given 3D system and a search algorithm for finding the fittest sequence. This allows us to find the optimal conformation for each single point mutant of the wild–type peptide and rank them accordingly. Since energy scores computed by PHOENIX are not $\Delta G_{binding}$ but an arbitrary interaction energy values, it is only possible to extract useful information when comparing two energy values for the same system in slightly different conformations. A lower energy score means a more stable and representative system. More stable peptides sequences are more likely to represent actual bound states and thus, are potential substrates for the protein.

It is important to note that during optimization, only the side chain of amino acids is optimized, either by mutation, or rotations. We assume that a single point mutation of a 3D model does not change significantly the conformation of both entities, and thus, the initial backbone conformation is valid for all single point mutants evaluated and optimized. This allows for much more efficient calculations as the search algorithm converges faster due to the limited number of degrees of freedom (DOF). Since the backbone is kept immobile, its intrinsic stability does not vary from one

mutant to another, and thus relative energy values are not impacted. For that reason, bonded terms are not included in the fitness function (bond torsion, angles and length within the structural model are ignored).

2.1.3 Fixed backbone and discrete rotamer threading approach

However, this fixed backbone approach may not be adequate for certain systems. Since the fitness function used is highly sensitive regarding atomic coordinates (EQ 2.1), the same rotamer threaded on two slightly shifted backbones could score dissimilarly, up to a point where the same mutation could be favorable on a backbone and unfavorable on the other. This is more likely to lead to an increased number of false negatives (Choi E. J., & al., 2009) since proteins, being natively dynamic, would slightly shift to accommodate the mutation. Since we assume that a single mutation does not alter the conformation of the protein-peptide interface, the dynamic rearrangement of the protein should be minor and retain the overall shape. Therefore, running the same mutational analysis on a set of slightly shifted backbones could improve the false negative rate by providing for each mutation more chances to score favorably. The score for a mutation *a* state *i* is given by EQ 2.5:

$$\mathrm{S_i} = e^{(-E_i/kT)} \qquad \text{EQ } \mathbf{2.5}$$

$$\mathrm{E} = \frac{\sum_{i=1}^{n}(E_i S_i)}{\sum_{i=1}^{n} S_i} \qquad \text{EQ } \mathbf{2.6}$$

where k is the Boltzmann constant (1.9872×10^{-3} kcal · mol^{-1} K^{-1}), T is the temperature in Kelvin, and E_i corresponds to the energy computed for the mutation on backbone i. The total energy E of a mutation corresponds to the Boltzmann-weighted score average of the backbone ensemble scores

at 300 K (EQ 2.6). Using an ensemble of structures when running a mutational analysis is referred to as multistate analysis (MSA) if the optimization runs are done independently, and as multistate design (MSD) if done simultaneously. The MSA method requires a minimum 1 CPU, whereas MSD requires as many CPUs as structures in the ensemble. This can be problematic for big ensembles. The difference between both methods is mainly technical and scores should be identical when the system is limited to a few residues, such as a single-point mutational analysis.

Various methods exist for generating an ensemble, including Molecular Dynamics, BackRub motion (Davis I. W. & al., 2006), and PertMin (Davey J. A. & Chica R. A, 2014). PertMin generates ensemble by perturbing all atomic coordinates by 0.001Å at a 50% rate and then minimizing the resulting structure by a defined number of iterations. The ensembles generated from PertMin shows low root mean square deviation (RMSD) from the input structure, as opposed to ensembles generated from molecular dynamics (Davey J. A. and Chica R. A., 2014). From the initial assumption that conformations of the entities in a crystal structures represents a highly populated state most probably linked to the protein`s biological function, having an ensemble of structure that closely resemble this initial state is beneficial.

2.2 First attempts at predicting specificity profiles

A previous attempt at computationally reproducing permutation peptide arrays was published by Lanouette *et al.* in 2015. The main goal of that research was to show that specificity profiles extracted from an experimental peptide array can be accurately reproduced using a computational MSD algorithm, and therefore, uncover new binding partners derived from the recognition motif predicted.

The protein used as a test case is Smyd2, a methyltransferase, for which a permutation array of a p53 decapeptide was available (Figure 2.2.1). The array showed high stringency at positions

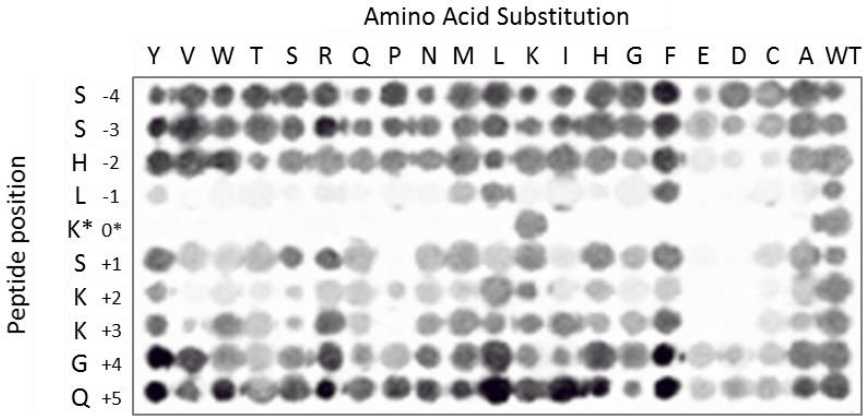


Figure 2.2.1. Experimental peptide array of SMYD2 – p53 (Lanouette S & Davey J.A, 2015). Positions –4 to –2 and +3 to +5 are tolerant to most amino acid mutations. Therefore, these positions are not part of the specificity profile of SMYD2.

–1 and +2 and moderate stringency at +1. Based on spot intensity, a recognition motif was extracted (Figure 2.2.3 A), showing the most active mutations at each stringent position. Most active mutants are defined by computing a relative methylation factor for each mutation (EQ 1.6).

For each position, if the RMF is above 1 or 2 standard deviation, then the mutation is included in the recognition motif. To simulate this experimental data, the MSD computational algorithm was applied, implemented as described in the previous section. It is important to understand that this MSD algorithm does not evaluate directly binding affinity of each single-point mutant with the protein. In fact, the energy contribution of the free form (unbound) of the mutant is never

computed, nor the entropy related to conformational freedom and solvent, and therefore, $\Delta G_{binding}$ cannot be assessed. This algorithm reflects the relative stability of a mutant at the interface of the protein with the native binding conformation preserved. Backbone is not relaxed during energy evaluation to accommodate the new sequence. This method is based on the idea that the binding conformation of the crystallized protein-peptide is favourable and therefore, no alteration to this conformation is desired.

It is important to note here, that the p53 peptide crystallized in the binding pocket of SMYD2 has a length of 6 residues. Relative to the methylated lysine residue found at position 0, the peptide extends from positions –1 to +4. In the example, the recognition motif of SMYD2 spans from positions –1 to +2. Hence, two positions were not included in the predicted motif. By analyzing the structural ensemble built for the procedure, the two terminal positions (+3, +4) showed higher pairwise root mean square deviation (pwRMSD) than that of any other position (Figure 2.2.2.). The rationale is that a flexible residue does not contribute as much to stabilize the peptide as much as tightly bound residues. Therefore, such residue must not be part of the recognition profile. Based on this hypothesis, residue +3 and +4 showed high pwRMSD in the ensemble of structures, and thus, were eliminated from the specificity profile of SMYD2. Fortunately, these results were corroborated by the experimental peptide array, where positions +3 and +4 tolerated most amino acid mutations (Figure 2.2.1.).

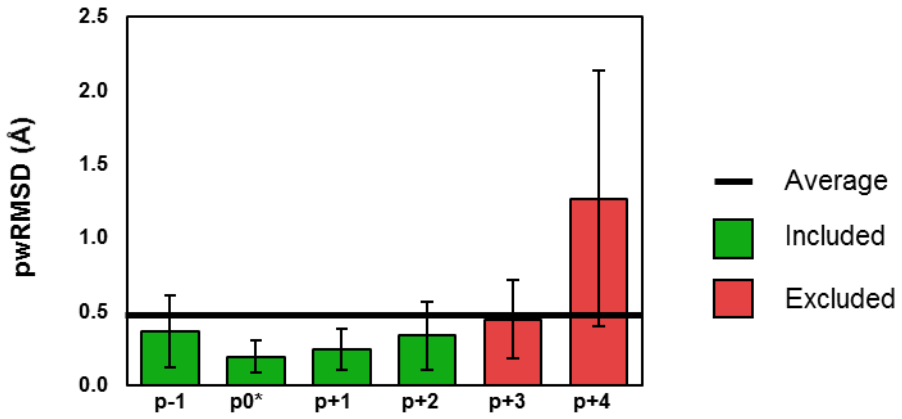
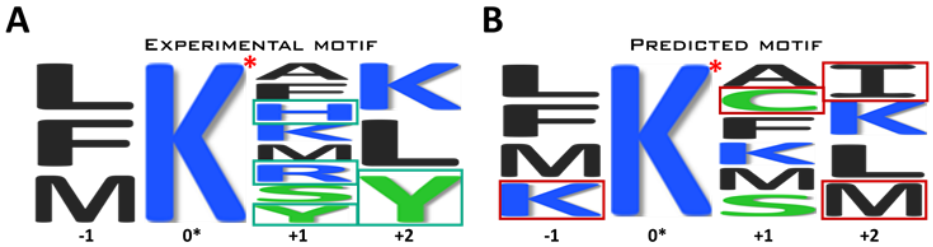


Figure 2.2.2. pwRMSD per position of the p53 peptide bound to SMYD2 (PDB ID: 3S7F) in the PertMin ensemble containing 180 perturb structures used for the energy calculation. Positions 373 (p+3) and 374 (p+4) show higher deviation that the rest of the peptide positions and are therefore excluded from the recognition profile of SMYD2. Those positions are hypothesized to be tolerant, thus unimportant for specificity due to higher pwRMSD than any other p53 positions.

Overall, for each position included in the recognition motif (p–1 to p+2), excluding the target lysine at p0, a list of 19 single point mutants (proline mutation was excluded in all cases, since it requires different backbone phi-psi torsions than any other amino acids) was outputted by the PHOENIX-MSD algorithm as described in the previous section. Single-point mutations were clustered, position-wise, based on their fitness score. Three clusters were generated using the K-means clustering method (Kanungo T. & al., 2002). It is an iterative method that allows to cluster numerical data based on their Euclidian distance from a centroid. Each cluster centroid is optimized during several cycles until convergence. The cluster containing the wild–type amino acid for that position on peptide is used as a cut-off and is referred to as the “reference” cluster. Any clusters with a centroid with a lower energy score than the reference cluster is merged with the reference cluster, and any cluster whose centroid has a higher energy score value is rejected. Every mutation comprising the merged reference cluster are then included in the recognition motif

for that position. The process is repeated for each rank-ordered list of amino-acids to complete the recognition motif.

The complete motif was then compared to the motif extracted from the peptide array extracted motif, which matched at 86% (Figure 2.2.3). In other words, 52 of the 60 mutations were correctly inserted or discarded from the specificity profile of SMYD2.



C

BINNING STATISTICS

True		False		Accuracy
Positive	Negative	Positive	Negative	(%)
10	42	4	4	86

Figure 2.2.3. Comparison between experimental and predicted motif of Smyd2. A) Experimental array derived from SPOT peptide arrays. B) MSD predicted motif. Green boxes correspond to false negatives. Red boxes represent false positives C) The predicted motif reproduces the experimental motif with an accuracy of 86%, where only 8 mutations are incorrectly included or excluded from the predicted motif.

This MSD algorithm seems promising for reproducing peptide array results. In fact, it was not only able to reproduce a specificity profile; the authors have used the predicted profile to identify 4 new substrates of SMYD2.

2.2.1 Current limitations of the method and main criticism

The proof of concept described in section 2.3 has various issues if it were to be scaled on a broader range protein-peptide complexes to uncover unknown specificity profiles (blind predictions). Firstly, the K-means clustering method used to determine tolerated amino acid mutations heavily depends on the energy distribution computed for a position. If the energy distribution is flat or uniform, then, clustering the energies based on centroids is meaningless. On the other hand, if the energy distribution is better represented by 4, or even 5 clusters, then, K-means procedure needs to take it into account and would require programmatic modifications. These issues complexify the methodology and for both reasons, K-means clustering might not be the most adequate algorithm to discriminate between tolerated and non-tolerated mutations.

Secondly, it is common for a protein to present areas in the binding pocket which tolerate many amino acid mutations. Such positions are non-specific and are identified in specificity profiles as an ‘‘X’’. The algorithm presented excluded tolerant positions in the p53 peptide sequence by evaluating pwRMSD deviation in the structural ensemble. Two positions (p+3 and p+4) showed higher pwRMSD deviations than the rest of the peptide (Figure 2.2.2) and were excluded from the specificity profile of Smyd2. The main concern here is that the cut-off used for pruning positions is arbitrary. The pwRMSD of the rejected p+3 position of p53 is higher (0.44 Å) than the accepted p–1 position (0.37 Å), but both under the average pwRMSD (0.48 Å) of the whole peptide. This method of discriminating between stringent and tolerant positions would need to be further benchmarked to prove its reliability on a broader dataset.

Finally, the computational method described was tested on a single protein-peptide complex, displaying mainly hydrophobic interactions (Figure 2.2.4a), a relatively short hexamer peptide exempts of proline and deeply buried within the binding pocket of SMYD2 (Figure 2.2.4b).

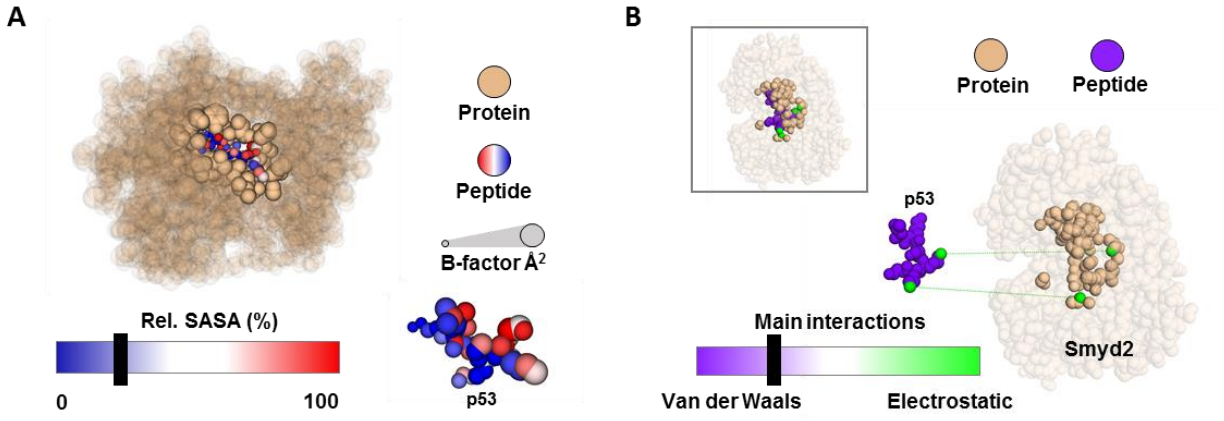


Figure 2.2.4. p53 peptide at the interface of Smyd2. A) The p53 peptide is deeply buried in the protein core, showing exposed surface area of only 26% relative to the free peptide. The B–factor is high for each atom, as shown by the sphere radius. The scale is from 0 – 100. B) The interfacial forces stabilizing the peptide are mainly hydrophobic. Over the 6-mer peptide, only 2 interactions picked-up are H-bonds/electrostatics (green).

Altogether, the Smyd2-p53 complex falls within the limitations of our computational software. First, the software only supports an implicit solvation model, which approximates solvent as a continuous medium. This is often an erroneous approximation at protein interfaces, where solvent distribution is not uniform. In the case of Smyd2-p53, an explicit solvation model should not be needed since water molecules are less likely to play a significant role at stabilizing a deeply buried peptide (26% exposure, relative to its unbound form). Secondly, the algorithm used for rotamer threading fixes the protein and peptide backbones in place, which negatively impacts directional forces (H-bonds and electrostatic interactions) compared to non-directional ones (hydrophobic or

Van der Waals interactions). Since the Smyd2-p53 interface is dominated by hydrophobic forces and shows only two directional side-chain interactions (Figure 2.2.4. B), a fixed backbone approximation should not be an issue. Finally, fixed rotamer threading algorithms such as done in PHOENIX cannot reliably thread rotamers on a proline backbone due to specific phi and psi angles forced by the 5-membered ring of this amino acid. Fortunately, the p53 peptide sequence does not contain prolines.

Overall, without a larger dataset to test the robustness of our computational procedure, there is no guarantee to be successful at predicting and simulating peptide arrays for proteins displaying properties at the limit of what our computational software can test. For example, protein interfaces dominated by electrostatic interactions, highly solvent-exposed peptides, or non-enzymatic proteins might not be suitable for the MSD procedure developed. Therefore, in the following sections we report accuracy of this method on a broader dataset, consisting of 5 protein-peptide complexes, each of those having their own peculiarities.

2.3. Applicability of the method on various complexes

To test the robustness of the method devised in Lanouette S. & Davey J. A., we chose 5 other protein-peptide complexes bearing different interfaces, and for which experimental peptide array data was published. The complexes added to the previous benchmark dataset include Smyd2 – p53 (PDB ID 3TG5, Lanouette S. & Davey J. A., 2015), Atxr5 – H3.1K27 (PDB ID 4O30, Bergamin E., 2017), Set8 – H4K20 (PDB ID 2BQZ, Kudithipudi S. & al. 2012), Gads – SPL76 (PDB ID 2D0N, Seet B. T. & al., 2007) and Erbin – ErbB2 (PDB ID 1MFG, Wiedemann U. & al., 2004). Our dataset consists of 3 methyltransferases (Smyd2, Set8 and Atrx5) as well as SH3c and PDZ recognition domains (Gads and Erbin, respectively). Altogether, they cover a total of 51 individual

peptide positions that will be compared to their respective experimental permutation arrays to assess predictability of the algorithm. Despite the limited size of the dataset, the 51 residue positions across the peptides display a wide range of biochemical environments and properties including relative exposure ratio, B–factors, atomic resolution, dominant interfacial forces, binding modes, amino-acid types and chemical modifications. Variability in these biochemical characteristics is key to properly assess our computational algorithm's robustness and strengths but also to highlight its weaknesses and potential pitfalls. They are also complementary and different to the initial complex (Smyd2 – p53, PDB ID 3S7F) studied and thus will bring further diversity for benchmarking.

Some interesting features of the dataset chosen includes high solvent exposure of some interfaces. This will represent a computational challenge since the solvent is not explicitly considered in our computational method; an implicit solvation penalty term was introduced (EQ 2.4) for energy calculations. The implicit solvation model used in our energy function emulates the effect of solvent, but by no means it can reproduce solvent H-bonding networks that could stabilize a protein-peptide complex or even calculate meaningful energy values when the solvent effect becomes more prevalent (e.g. solvent exposed side chain). Furthermore, the pairwise energy evaluation of the algorithm inherently overestimates buried area and underestimates exposed area, which leads to a rough energy approximation of solvent effect. This could have repercussions on our predictions by potentially ranking mutations based on an unreliable energy calculation depending on the chemical environment of the protein-peptide interface. Figure 2.3.1 A-E illustrates the wide range of relative solvent exposure (blue-white-red gradient) that is found in our dataset.

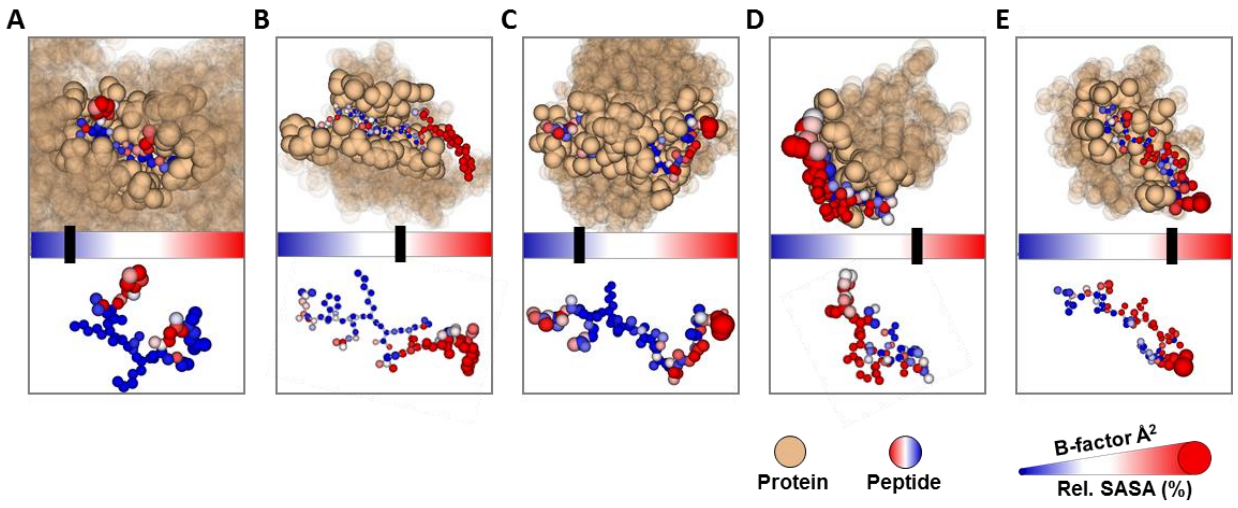


Figure 2.3.1. B–factors and Solvent accessible surface area (SASA) for each protein-peptide complexes. Protein-peptide complexes considered for further study and improvement of the computational algorithm. For each protein complex, the bound peptide is represented as blue-red gradient colored spheres, whereas the protein is shown in beige. Peptide atoms radius are scaled based on B–factor. In addition, each peptide atom is colored based on an exposure scale: blue residues are buried, and red residues are exposed. An atom with a high exposure change upon binding (fully buried by the protein) is shown in blue, whereas an atom with low to none relative exposure change is shown in red. A) SMYD2-p53, PDB ID 3TG5, shows a deeply buried peptide inside the core of the methyltransferase. Positions +3 and +4 display a tolerant mutational profile. B) SET8 – H4K20, PDB ID 2BQZ, has a shallow cleft relatively exposed to solvent. Positions -3, -2 and -1 and held in place by various electrostatic and H-bond interactions. C) ATXR5 – H3.1K27 shows a 13-amino acid long peptide going through a pore of the methyltransferase. The interface consists mainly of hydrophobic packing interactions. D) GADS – SLP76 represent an SH3-SH3 domain interaction. There are two obvious anchor residues, residues +5 (R) and +8 (K). Both residues are heavily dominated by electrostatic interactions. The rest of the positions are solvent exposed and maintained in place by hydrophobic packing. E) Erbin – ErbB2, PDB ID 1MFG complex showing a low average B–facto and a highly exposed peptide at the interface.

Another interesting feature in the dataset is the presence of interfaces with low and high B–factors. B–factors report the degree to which the electron density of a specific atom is spread out, which is often utilized to reflect the dynamicity or disorder of certain regions of a protein. It is computed when solving the crystal structure using the Debye-Waller Factors (Debye P., 1913):

$$B_i = 8\pi^2 u_i^2 \qquad \text{EQ } \mathbf{2.7}$$

The Mean square displacement (u) obtained from the x-ray diffraction experiment is used to compute the B–factor B_i in units of $Å^2$. A high B–factor peptide bound to a protein could reveal a much more complex binding mode than the simple "lock and key" model (Fischer E., 1894, Figure 1.3.2.A). Our specificity profile predictions could suffer from higher dynamicity/disorder since the algorithm used assumes a productive binding mode adopted by the peptide upon crystallization and does not allow any backbone movement during its optimization step. Figure 2.3.1 A-E illustrates the wide range of B–factors represented as atoms varying in radius based on a scale from 0-100. This is used as a proxy for dynamicity/disorder in the peptide.

Finally, another feature of the dataset under study is the broad range of interactions present at the interfaces of the protein-peptide complexes. Peptides bind to proteins with different affinities based on the specific interactions developed at the interface with the protein. Stronger electrostatic interactions like salt bridges often serve as anchor points while the rest of the protein is stabilized by weaker interactions such as H-bonds or Van der Waals interactions (London N. & al., 2010). The computational method described in section 2.1 uses discrete rotamers and does not allow backbone movement during sequence optimization, which can impact energy evaluation of directional forces such as salt bridges and H–bonds. Therefore, the importance of electrostatic versus hydrophobic interactions for a given protein-peptide complex might impact the accuracy of predictions in certain cases. Our current dataset was built while making special considerations to include interfaces with varying ratios of hydrophobic to electrostatic forces as shown in Figure 2.3.2. A-E.

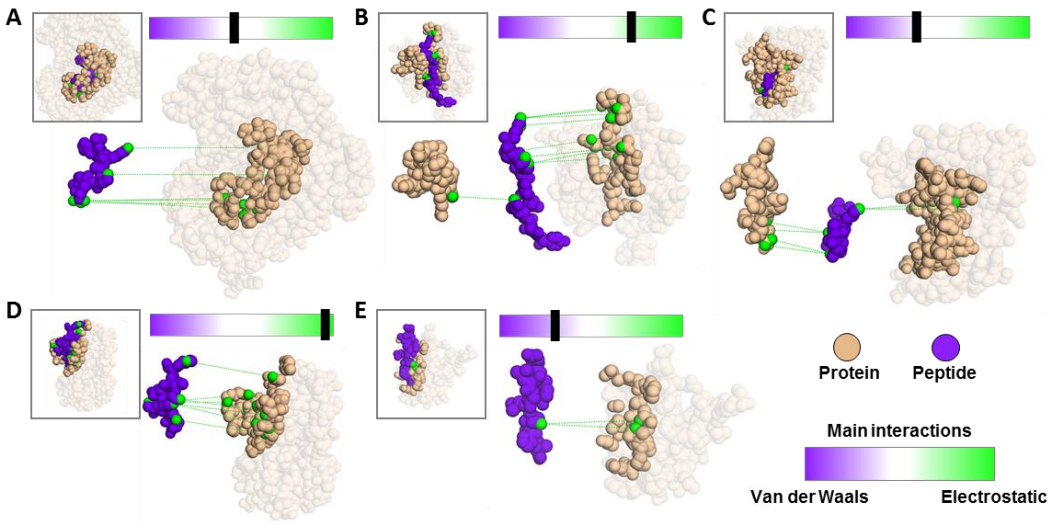


Figure 2.3.2. Exploded view of the protein-peptide complexes under study. Proteins are shown in beige and peptide in purple. Green spheres represent peptide-sidechain/protein atoms electrostatic and H-bonding stabilizing interactions. A slider opposing Van der Waals (purple side) to electrostatic (green side) interactions is shown as a qualitative measure of the dominating forces at the protein-peptide interface. It A) p53 peptide bound to Smyd2 (PDB ID 3TG5) showing one major anchor residue, while rest of the peptide side chains are stabilizing the interface mainly via hydrophobic packing. B) The H4K20 peptide of SET8 (PDB ID 2BQZ) is held in place in the protein cleft via strong N-terminal electrostatic interactions. The cleft shows a C-terminal hinge region which could open-close the cleft in a claw-like movement, securing the peptide upon closing. C) The H3.1K27 peptide binds to ATXR5 mainly through hydrophobic packing. Like SET8, ATXR5 shows a C-terminal hinge region which could open-close a pore in a claw-like movement, securing the peptide upon closing. D) GADS peptide is anchored at the surface of SLP–76 by strong electrostatic interactions. E) ErbB2 peptide binds at the surface of Erbin, but unlike SLP–76 – GADS, hydrophobic interactions dominate at the interface.

Now that we were satisfied with the diverse crystal structures at hand, we extracted from the literature permutation arrays for each protein-peptide complex. The arrays were built on the same wild–type peptide sequence scaffold as the one crystallized along the protein. Consequently, we were convinced that our predictions could be directly compared with the experimental data.

Since arrays obtained from various sources were quite different in format and data (numerical intensity values were not available and mutational order was inconsistent), raw autoradiographs (Figure 2.4.3 A) were cleaned with “ImageJ” (https://imagej.net), an image processing JAVA

software developed by the National Institute of Health in collaboration with the Laboratory for Optical and Computational Instrumentation (University of Wisconsin). The background was subtracted from the array based on a rolling ball algorithm (Sternberg S., 1983) implemented in ImageJ with a pixel radius of 50, and each spot's intensity (AA_{ij}) was measured by integration (Figure 2.4.3 B) and then divided by the wild–type amino-acid (WT_i) spot intensity (EQ 2.8). If a mutation spot intensity ratio (R_{ij}) was ≥50% of the wild–type spot intensity for a given position, it was classified as a mutation tolerated by the protein and included in the recognition motif (EQ 1.7).

$$R_{ij} = AA_{ij} / WT_i \qquad \text{EQ } \mathbf{2.8}$$

Finally, to generate standardized permutations arrays for viewing purposes, the intensity ratios calculated from EQ 2.8 were then uniformly drawn in an alphabetical order, using our in-house JAVA suite (Figure 2.3.3 C). Tolerated amino-acid mutations based on EQ 1.7 using a cut-off value of 50% were included to a recognition motif. This process was applied to each member of our dataset to extract specificity profiles (Figure 2.3.4. A-E). The amino-acids are color-coded according to their identity. This can provide insights as to which type of amino acids are tolerated at a position. These types of amino acids include hydrophobic (I, L, V, F, W, M, A), polar (T, Y, Q, N, S, C), charged (D, E, R, K, H), and special (G, P). However, if a peptide position contains 10 or more stable mutation, the whole position was declared as tolerant to mutations, where an "X" replaces all amino acids for simplicity. A red star also indicates positions targeted by the protein, usually a post-translational modification site, such as the lysine that is methylated by a

methyltransferase (Figure 2.3.4 A-C). Over the 5 recognition motifs of our dataset, 54% of the single point mutations

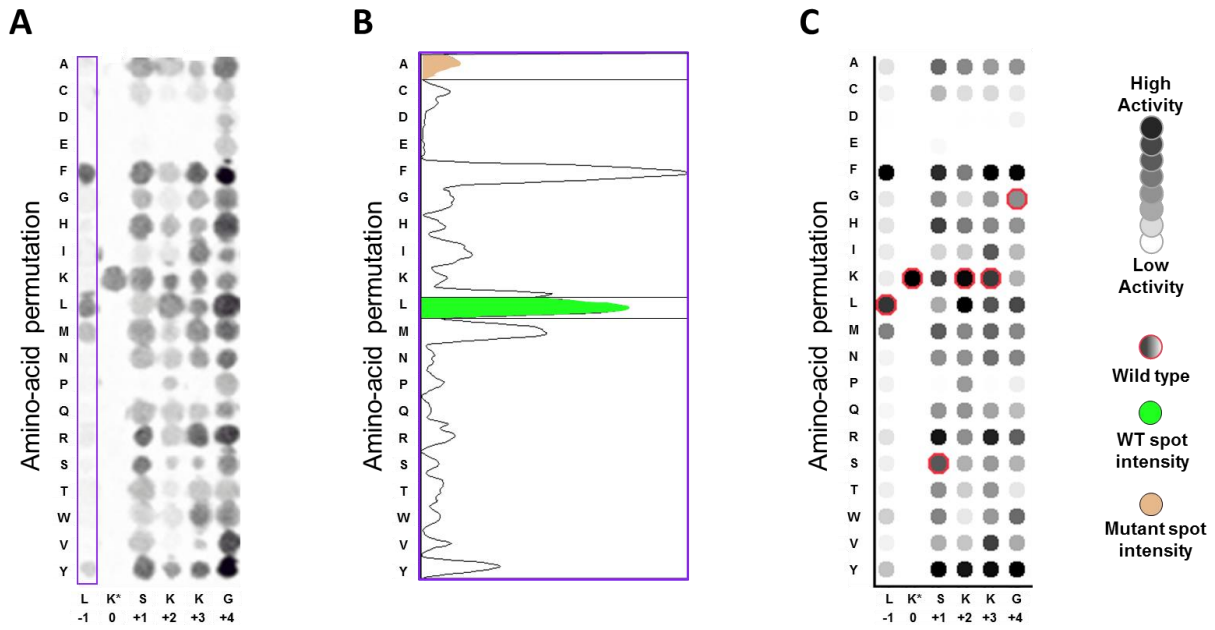


Figure 2.3.3. Analysis of raw permutation array autoradiographs from literature. A) The raw autoradiograph is analyzed, column (position) wise, by Image J, a JAVA image processing software (https://imagej.net). B) Each spot of position P–1 shown in A) is integrated to have a relative value of intensity, which are divided by the wild type amino-acid naturally found at that position. C) The ratio is drawn using in-house JAVA protein analysis framework for standardized viewing purposes. This alleviates the variability of peptide array presentation.

made to the peptide sequences were tolerated by the protein. This could be surprising, considering that proteins are known for their high specificity. However, permutation peptide arrays are only targeting single mutations, hence the sequence space explored is limited and similar, which explains this observation.

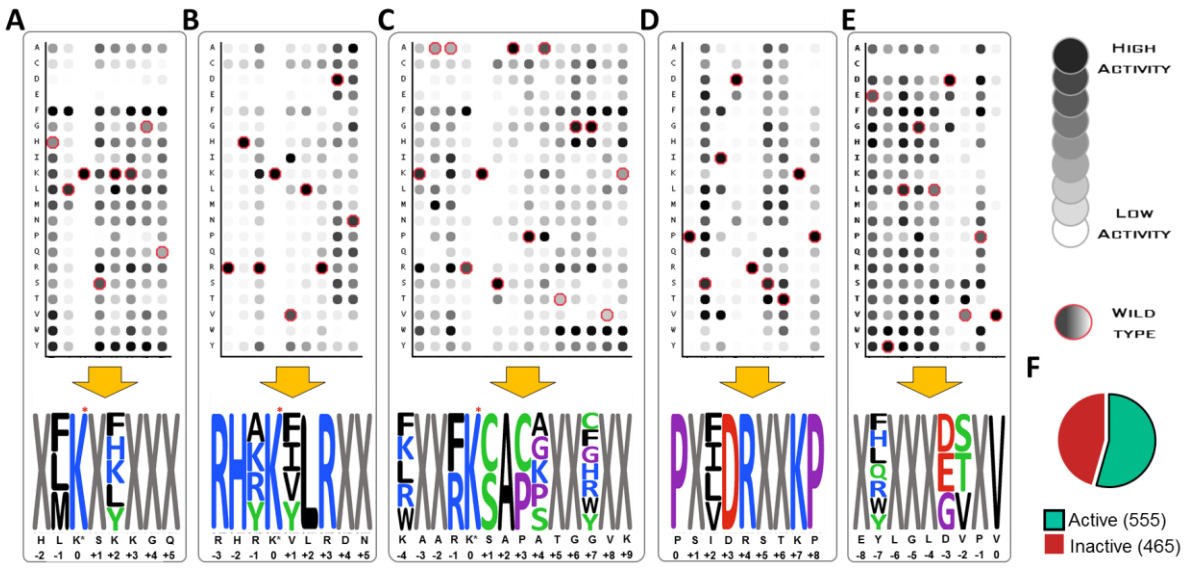


Figure 2.3.4. Curated experimental permutation arrays. A darker spot indicates more activity detected between the mutant peptide and the protein. A) Smyd2-p53, PDB ID 3TG5. Only positions p–1 and p+2 show high stringency. B) Set8 – H4K20, PDB ID 2BQZ. Stringency arise from positions p–3 up to p+3, leaving the C-termini positions tolerant to mutation. C) Atxr5 – H3K27. The positions important to peptide recognition are dispersed across the peptide. D) Gads – SLP76. Positions +5 (R) and +8 (K) are particularly stringent and are reported as the main recognition pattern for Gads. In addition, p0, p+2 and p+3 display vey high stringency. E) Erbin – ErbB2, PDB ID 1MFG. The C-terminal valine (p0) is the most important position for specificity. Positions p–7, p–3 and p–2 are also important for recognition by the PDZ domain. F) From extracted recognition motifs, covering 51 positions, 54% (555) of the single point mutations are active towards the proteins and 24 positions are tolerant to mutations, thus not important for peptide recognition events.

In summary, our dataset consists of post-translational modification enzymes and protein-protein interaction recognition domains. The peptides under study display a broad range of exposure to solvent as well as a wide range of B–factors. Ultimately, the peptides adopt different conformations and binding modes and represent a diverse dataset that should allow us to test the robustness of our computational method.

2.4 Benchmarking

Now that we have described in detail the dataset chosen for evaluating the scope of the method published in Lanouette S. & Davey J. A., 2015, this section will focus on the results and challenges faced when re-implementing the methodology, which lead us to implement various modifications in the algorithm.

The first modification we made to the motif prediction procedure in PHOENIX was switching the MSD algorithm for an MSA approach. This was needed to decrease computational resources and allow for large scale optimization of the prediction algorithm. Multi state design (MSD) requires parallel optimization runs across all the states, since all CPU slave nodes orient their search based on each other node's results simultaneously. When running the algorithm on 120 states, 120 CPU nodes must be available at the same time; heavily loading a CPU cluster. On the opposite, multi state analysis (MSA) can be run with sequential optimization runs for each state. Therefore, the same algorithm on 120 states require at least 1 CPU node available which runs sequentially all the 120 optimizations. This better balance the cluster load and is more suitable considering our resources. We do not expect any drop of accuracy for the predictions since small peptides have a limited search space, thus the sequential optimization runs in MSA should cover a combination spectrum large enough for these protein-peptide systems so that a decent overlap in sequences searched can be achieved across all independent runs. (section 2.1.3 for more details).

The results obtained from running the PHOENIX-MSA algorithm with a K-means clustering for motif generation, as described in section 2.2 are reported below (Figure 2.4.1 B). As described in 2.2, K-means cannot be applied easily as a binning method, which led us to compare the same procedure with a cut-off threshold as in EQ 1.7. The cut-off applied was chosen from the K-means

cluster analysis, which showed that the highest energy cluster member to be included in the recognition motif was on average at 4.69% kcal/mol from the wild–type residue's energy, hence we set the cut-off to 5% (Figure 2.4.1 C). The use of K-means for classifying positions does not bring any improvements to the method (Figure 2.4.1. D) since both predict recognition motifs with 64% accuracy. As a result, K-means can be replaced by the cut-off procedure, without impacting the specificity profiles.

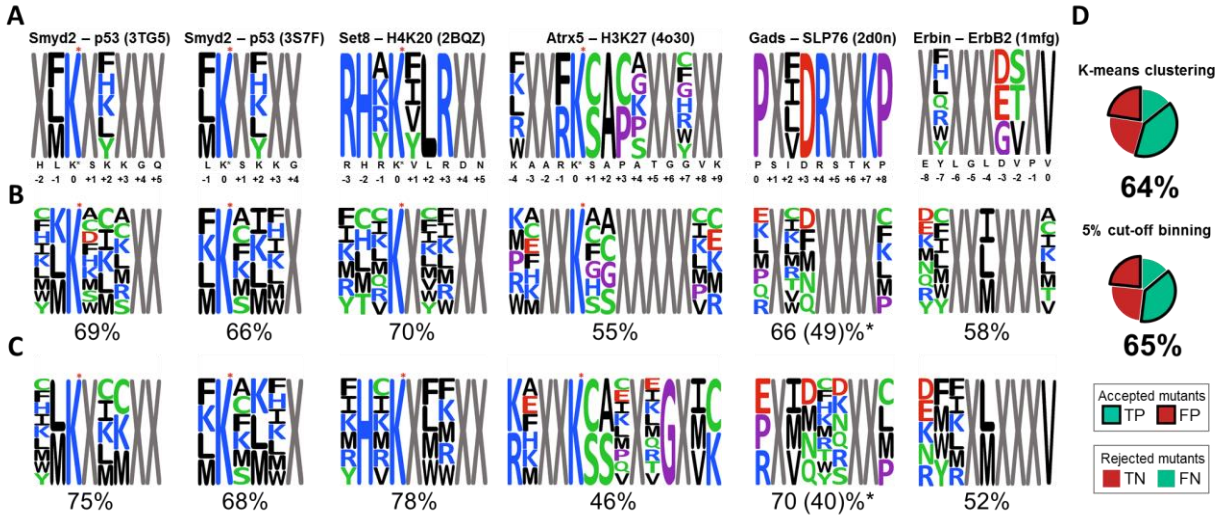


Figure 2.4.1. K-means versus threshold binning. A) Motif extracted from experimental permutation arrays. B) Motif predicted from the PHOENIX-MSA methodology and applying K-means clustering for motif generation. C) Motif predicted from the PHOENIX-MSA methodology and applying 5% cut-off (EQ 1.7) for motif generation. D) Pie chart representation of the binning statistics (dark green: true positives (TP), light green: false negative (FN), light red: true negatives (TN), dark red: false positives (FP). Overall prediction accuracy of 64% for the K-means binning and an overall accuracy of 64% for the cut-off binning. Similar results are achieved by both binning methods, despite the cut-off being much simpler to apply.

Already we can observe a clear drop in accuracy from the accuracy of 86% reported in Lanouette S. & Davey J. A. (2015). The initial predictions display many discrepancies compared to the "reference" motif (peptide array derived). Lower accuracy predictions arise from two distinct scenarios. Either stringent positions are predicted to be tolerant (10 or more amino-acid

permutation are stable at a given position, represented as an "X" in the motifs) or tolerant positions are predicted to be stringent (less than 10 amino acid mutations are tolerated by the protein). The latter is significantly more detrimental since it introduces many false negatives, which reduces the number of potential hits included in the recognition motif.

Positions across our complete dataset for which predictions (Figure 2.4.1 C) are dominated by false positives includes p–3 of Set8, p–1/p3/p4/p7 of Atxr5, p4/p7 of Gads and p–3/p–2 of Erbin. To improve predictions at these positions, we hypothesized that it might be due to an imbalance in the electrostatic and H-bonding term of our force field, described in section 2.1. Some very stringent positions amongst the one mentioned above shows charged and polar amino-acids (blue and green color-coded residues, respectively). This is specifically the case for position p–1 of Atxr5, p4/p7 of Gads and p–3/p–2 of Erbin. As reported in London N. & al. in 2010, electrostatic and H-bond interactions are important for stabilizing and anchoring the peptide at the interface. Given the directionality of such interactions, their energy contribution is more likely to be under evaluated by our algorithm for various reasons mentioned in section 2.4, hence the reason why we decided to inflate their contribution to the total energy value. The H-bond well depth (H-bond scaling factor) was inflated from 8.0 kcal/mol to 8.2 kcal/mol (EQ 2.3) and the electrostatic dielectric constant was decreased from 40 to 10, therefore enhancing the electrostatic contribution by 4 (EQ 2.2). The improved force field predictions are reported in Figure 2.4.2.

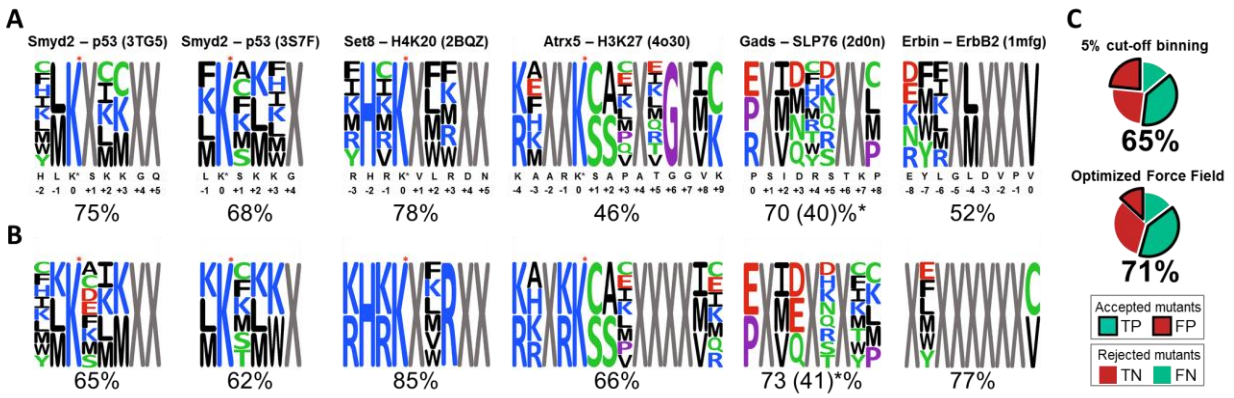


Figure 2.4.2. Forcefield optimization. Comparison between the prediction outcome from A) the original force field and B) the electrostatic biased force field. C) Pie chart representation of the binning statistics (dark green: true positives (TP), light green: false negative (FN), light red: true negatives (TN), dark red: false positives (FP). Overall prediction accuracy of 64% for original force field and a slightly better accuracy obtained from the electrostatic biased force field. Predictions are more consistent across the dataset with the improved force field, but still far behind in terms of predictability when compared to the original study from section 2.3.

2.5 Predicting tolerant positions across a recognition profile

Our PHOENIX-MSA computational algorithm, as described previously, poorly reproduces specificity of residues that were found to be tolerant by permutation array analysis for the 6 proteins used in our benchmark set. Figure 2.5.1.A illustrates the clear difference in accuracy for predictions of positions in a peptide deemed important for specificity as opposed to tolerant positions. The distribution of accuracies for stringent positions (lower quadrants) is centered at 78% accuracy on average, whereas tolerant positions are centered at 68%. Overall, the predictions are closer to random when the algorithm deals with tolerant peptide positions. Therefore, we need an additional algorithm that can identify such positions from the crystal structure, to improve global predictive power.

Figure 2.5.1. Variations in accuracies for stringent versus tolerant positions. The actual accuracies for predicting stringent and tolerant positions are reported as box plots. The experimental tolerance is measured from the number of stable substitutions found in a permutation array for that position. A) Comparison of experimental tolerance with the predicted tolerance obtained from the number of stable mutations determined by our computational method. A position stabilized by 10 or more substitutions is classified as irrelevant for specificity (tolerant). B) Comparison of experimental tolerance with the relative solvent accessible surface area calculated form the Shrake and Rupley's algorithm. A position showing relative exposure of more than 40% is classified as tolerant. C) Comparison of experimental tolerance with the number of interactions made by a residue's side chain with the protein. These interfacial contacts were included up to 4 Å away from any side-chain atoms. A residue making contacts with 2 other residues or less are considered tolerant. D) Comparison of experimental tolerance with B–factors. Residue B–factors were averaged from each of its atom's B–factor. A residue having a B–factor above the peptide's average B–factor is considered tolerant. E) Comparison of experimental tolerance against the combined previous metrics. A position not satisfying any of the constraints described from A-D is deemed tolerant.

As a mean of generating a new algorithm capable of discriminating between tolerant and stringent peptide positions solely from the available crystal structure information, we investigated the following residue-based characteristics in an attempt at correlating those with their own tolerance in the recognition profile: relative solvent-exposed surface area (SASA), number of nonbonding interactions made by side chains at the interface, and crystallographic B–factors.

First, we analyzed relative exposure of a residue (Figure 2.5.1. B), as a proxy to assess the prevalence of its interactions at the interface with the protein. The hypothesis is that a more solvent exposed residue, defined by a lower difference in solvent accessible surface area between its free and bound states, is not likely to be important for molecular recognition as it will make few stabilizing interactions at the protein-peptide interface. In the present work, we defined an exposed

residue when the variation in water accessible surface area (Lee B., Richards F. M., 1971) was less than 60%, relative to the free tri-peptide (Gly-X-Gly). Since, analytical calculation of surface area requires complex calculations, it was approximated using the Shrake-Rupley's algorithm (Shrake, A., Rupley, J. A., 1973). In this method, SASA of atoms is calculated by representing the atom surface as a mesh of 2000 points (Figure 2.5.2.), each of them carrying a small surface area. If a point is buried inside another atom's surface, its corresponding area is not counted towards the total exposure of the atom. The atom radius is augmented by the radius of a probe of 1.4 Å, which is meant to represent the size of a water molecule in contact with the atom. Looping over all atoms of the residue yields its exposure in the bound state. The relative SASA is obtained by dividing this value with the exposure of the free residue. In the implementation used (BioJava.org; Prlić A. & al., 2012), hydrogens atoms were not considered, thus the radius of atoms possessing implicit hydrogens such as methyl groups were inflated by a fixed value based on the work of Chothia C. & al., 1976. Following the SASA analysis, stringent and tolerant positions appear to be exposed or buried with no apparent correlation between both descriptors.

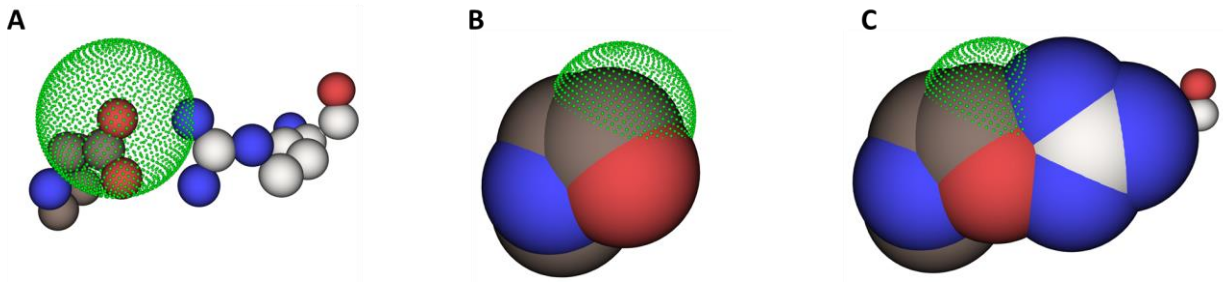


Figure 2.5.2. Shrake-Rupley's algorithm for approximation of exposed surface area. We are considering water as the solvent; thus, each atom's Van der Waals radius is inflated by 1.4 Å (large spheres) to account for a rolling water probe in contact with the atom. A) Available surface area of a glutamate (shown in brown) terminal oxygen is approximated by a mesh of 2000 points (green dots). Relative SASA is obtained by dividing surface area of the free form (B) by the exposed surface area of the bound form (C).

Second, we took a closer look at the number of nonbonding interactions that the side chain of each residue makes at the protein interface (Figure 2.5.1. C). We hypothesized that a higher number of protein contacts made by a side chain would play a greater role in stabilizing the interface. Hence, such a position should be more stringent, since those contacts would have to be maintained for a given amino acid substitution. This metric goes hand in hand with accessible surface area. The major difference between both metrics is that a residue with low relative surface area exposure might be buried by the peptide upon folding and/or binding, and not by the protein itself. Therefore, looking at the contacts made directly at the interface might the importance of a position. Interface contacts for a given residue were found by expanding every side-chain atom Van der Waals radius by 4 Å. Each protein atom lying within this augmented radius were resolved to their corresponding residue, which was then counted towards the number of contacts made at the interface. Any peptide atom returned by the search was ignored. The position is classified as tolerant if no more than 2 contacts are registered, otherwise it is classified as stringent. Then again, no correlation was found between peptide array derived tolerance profiles and side-chain interactions. We are not able to predict tolerant prediction are a rate higher than 74% (Figure 2.5.1. C).

Third, we investigated B–factors as a proxy for residue flexibility at the protein-peptide interface. Atom B–factors (B) are directly linked with electron density maps when resolving a crystal structure. A low-density map (higher mean square displacement U) reflects static or dynamic mobility of an atom, which prevents from accurately determining coordinates in the crystal lattice (Blundell T. L. & Johnson L. N., 1976). When an atom is forced into such a map, it is assigned a high B–factor value (temperature factor) as described in EQ 2.7. In our case, the absolute B–factor of a residue is not meaningful since we are interested in relative mobility of residues in the peptide. We hypothesized that a higher B–factor for a given position, relative to the whole peptide would

reveal a tolerant position. Therefore, a residue is considered tolerant if its B–factor is higher than the average B–factor of the peptide chain. Unfortunately, we observed a low correlation between B–factors and tolerance profiles. We could reproduce profiles at 73% success rate, but still could not recover more that 58% of the tolerant positions (Figure 2.5.1 D). This can be reasoned by the fact that some residues showing relatively low B–factors might be strongly held in place by the peptide itself instead of the protein interfacial residues. Furthermore, some of our permutation array data was obtained from longer peptides than the ones found in the crystal structures. The longer extremities, could potentially impact the mobility of the inner residues, possibly explaining incorrect predictions of stringent positions.

Let alone, these metrics could not recapitulate tolerance observed from permutation arrays. For example, residue exposure might not correlate with tolerance if a residue's burial is not due to protein binding. Furthermore, interfacial contacts might not correlate with tolerance if a residue is making a low number of strong contacts and vice versa. Finally, B–factors might not be predictive of tolerance if a low B–factor is reported due to strong intra-peptide contacts. These counter examples bring to light how our reported metrics are intrinsically linked altogether. On this basis, we analyzed the correlation between tolerance profiles and the combination of exposure, contacts and B–factors (Figure 2.5.1. E). For this analysis, a position is predicted stringent only if all the metrics classifies the residue as so, based on the same cut-off values as previously described. This improved our ability to predict a position as either stringent or tolerant. Using the combination of all metrics allowed us to correctly classify 100% of the tolerant positions, while maintaining an accuracy of 74% for the identification of tolerant residues. This new independent algorithm for predicting tolerant positions across a recognition profile in combination with the sequence optimization protocol presented herein is called VIPER. We applied the complete procedure and

optimized the cut-off used (Supp. Figure SB.2) to yield the best results by including 534 of 555 active sequences (96.2%) and by effectively filtering 303 out of 465 inactive sequences (65%). These statistics highlight the strength of this computational method for predicting recognition profiles: a high majority of active sequences are retained while most of the inactive sequences are removed. A comparison between the initial algorithm presented in section 2.3 and the final VIPER version is shown in figure 2.5.3.

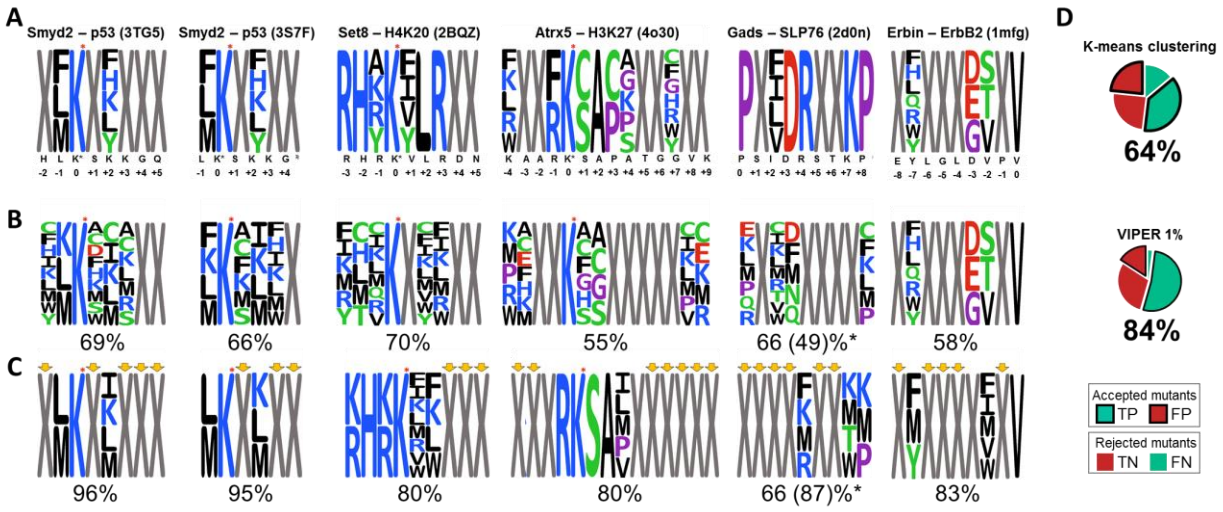


Figure 2.5.3. Comparison of the initial algorithm to the final VIPER algorithm for predicting specificity profiles by introducing the prediction of tolerant positions. A) Motif extracted with ImageJ from the experimental peptide arrays. B) PHOENIX-MSA, with K-means clustering for motif generation, as published in Lanouette S. & Davey J. A. C) PHOENIX-MSA with a 1% cut-off for motif generation with post-treatment using SASA, B–factors and interaction maps for discriminating tolerant from stringent positions – VIPER method. D) Pie charts representing the distribution of false negatives, true negatives, false positives and true positives. An improvement of 20% in accuracy was achieved from the initial methodology to VIPER.

The protein Gads bound to SLP76 is the only complex with prediction accuracy below 80%. VIPER overly predicts p0, p+3 as tolerant (Figure 2.5.3. C), whereas the experimental peptide array displays high stringency at those positions (Figure 2.5.3. A). Consequently, the predicted

motif includes many false positives, which impact negatively the accuracy. However, the peptide SLP76 was submitted to an alanine scan where SPR (Seet B. T. & al., 2007) was used to identify the most important positions in the peptide for binding affinity ($\Delta\Delta G_{binding}$). The authors found that the two crucial positions for binding are p+4 and p+7, with a significant $\Delta\Delta G_{binding}$ above any other positions. By combining their results from SPR and permutation peptide array, they postulate that the binding motif of the protein Gads is in fact -XXXX**R**XX**K**X-. Interestingly, VIPER does predict this motif at an 87% accuracy (Figure **2.5.3.** C). Therefore, despite the seemingly low accuracy for the Gads – SLP76 complex at reproducing the peptide array, it accurately predicts the SPR derived motif.

Another interesting fact about the predictions from VIPER concerns the PDZ domain complex Erbin – ErbB2. This domain is known for phosphorylated tyrosine recognition at p–7. The experimental peptide array (Figure 2.5.3. A) does not capture efficiently this known characteristic, however, the predicted motif does, and that being one of the only 3 predicted stringent positions across the whole 9-mer peptide. Furthermore, PDZ domains are known for displaying an -XϕXϕ recognition motif, where ϕ replaces any hydrophobic residues (Wiedemann U. & al., 2004). Again, the peptide array at p–2 (Figure 2.5.3. A), does not capture this feature, where a threonine and serine are the only additional amino acids tolerated other that the wild type valine. Interestingly, the VIPER motif does predict a range of hydrophobic residues at p–2.

Both experimental and computational methods have their share of variability; atomic coordinates are crucial for outcome of VIPER predictions (discussed in chapter 3), whereas reaction conditions are as crucial for experimental assays.

Overall, the diversity of the dataset and the constant quality of the predictions both confirm that VIPER is a robust method for simulating permutation peptide arrays. On a 3.2 GHz computer, a single position run for a protein run took between 1.5 and 7 days. Some speed optimization was achieved for better performance, more on that subject in chapter 3. For this study, a computer cluster was used, cutting down calculation time by 30X. Overall, a single protein recognition motif took under two weeks of calculation. Considering that permutation peptide arrays experiments, usually take may take up to 12 weeks, the VIPER procedure is a quick and predictive alternative to obtain recognition profiles.

2.6 Application of VIPER to a real case study

To further convince the scientific community that VIPER was not over-optimized on the dataset used, and that it could yield interesting results in a real case study, we used VIPER for predicting a novel binding profile of Smyd3, a methyltransferase. Smyd3 is crystallized with two different peptides in the Protein Data Bank: Smyd3 – VEGFR1 (PDB ID 5EX3) and Smyd3 – MEKK2 (PDB ID 5HQ8). Peptides are dissimilar in both sequence and geometry (Figure 2.6.1. A-B), and could yield different profiles, due to fixed backbone and single point mutations constraints (more on that subject in chapter 3). Therefore, we decided to produce recognition motifs for both sequences (Figure 2.6.1. C). Both motifs are very different at p–2 and p+2, which leads to patterns mutually exclusives (sequence space provided by the motifs does not overlap). This shows how

the crystal structure impact the

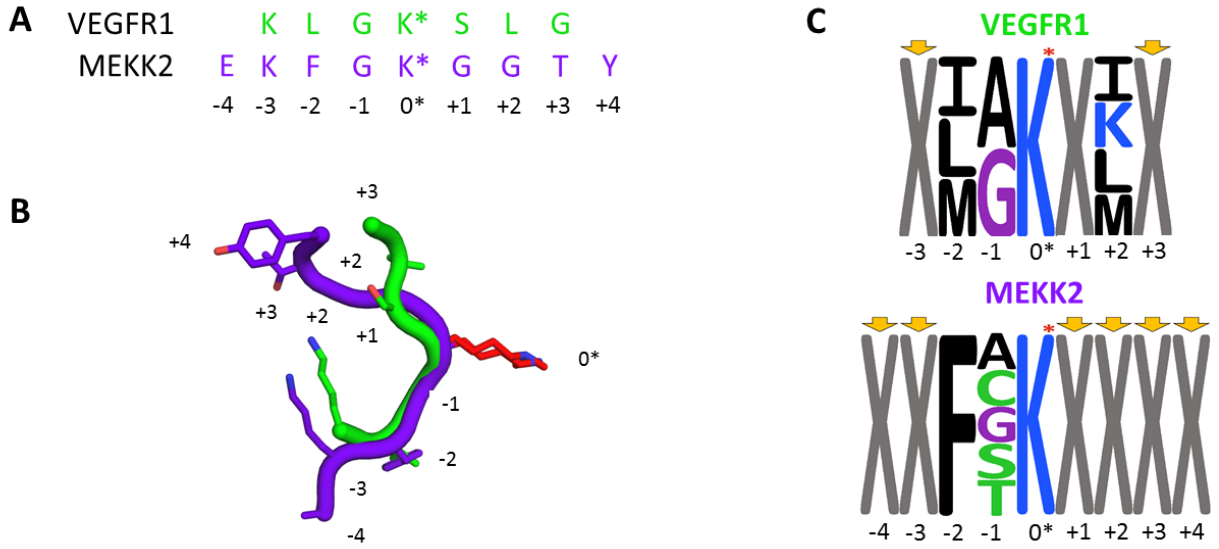


Figure 2.6.1. Comparison of Smyd3's bound peptide VEGFR1 and MEKK2. Real case application of VIPER recognition profile predictions for Smyd3, a methyltransferase known to bind A) the MEKK2 and the VEGFR1 peptides. B) Both crystal structures are available (PDB ID 5HQ8 and 5EX3 respectively) and shows very different peptide binding geometries. C) Motifs predicted for each crystal structure. Both motifs display no overlap in terms of output sequence combinations due to tolerated amino acids at p–2.

predictions from the algorithm. A more detailed analysis of this impact is found in chapter 3. We then generated experimental permutation peptide arrays to asses our predicted motifs' accuracy (Figure 2.6.2.). Since both peptides contain a lysine at p–3 (Figure 2.6.1. A), the permutation array was performed on the p–3 K/A mutant to prevent methylation at that position by Smyd3. Therefore, p–3 was ignored for accuracy calculation since a cut-off value for the wild–type lysine could not be evaluated, making the analysis futile for that position only.

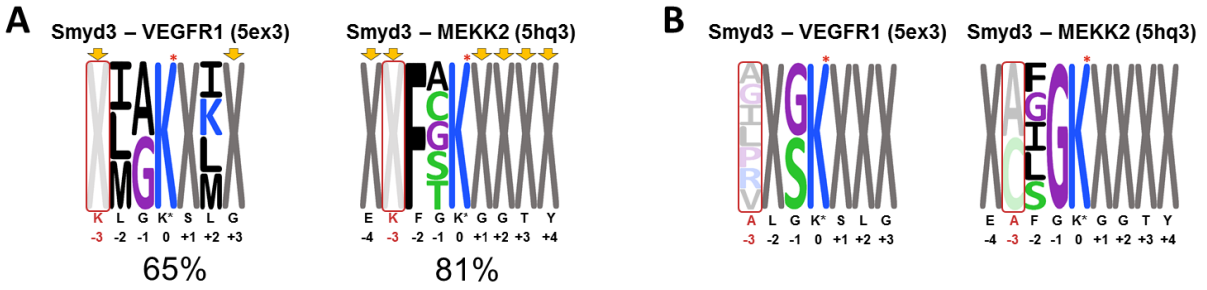


Figure 2.6.2. Predicted versus experimental motifs of Smyd3. A) SPOT peptide array derived motifs. B) Boxed positions are ignored as those were mutated specifically for the experimental assay to prevent false positives due to lysine methylation at p–3 for VEGFR1 and MEKK2 peptides.

Then, known methylated (K_{me}) peptides were selected and assembled into a library of 2550 member. That library was then screened using each motif, and the output sequences were tested for activity towards Smyd3. Less than 1% of the sequences passed the filter using the MEKK2 motif, for a total of 21 sequences. After running activity assays, 10 peptides were methylated by Smyd3, a 48% hit-rate (Figure 2.6.3. A). For the VEGFR1 motif, 20 sequences remained and 16 of which were methylated by Smyd3, an 80% hit-rate (Figure 2.6.3. B). Combining information from both peptides yielded a new motif (Figure 2.6.3. C) that was used to screen the library as well, after which 64 sequences were retained for activity assays and 45 were found methylated, a 70% hit-rate (Figure 2.5.2. C).

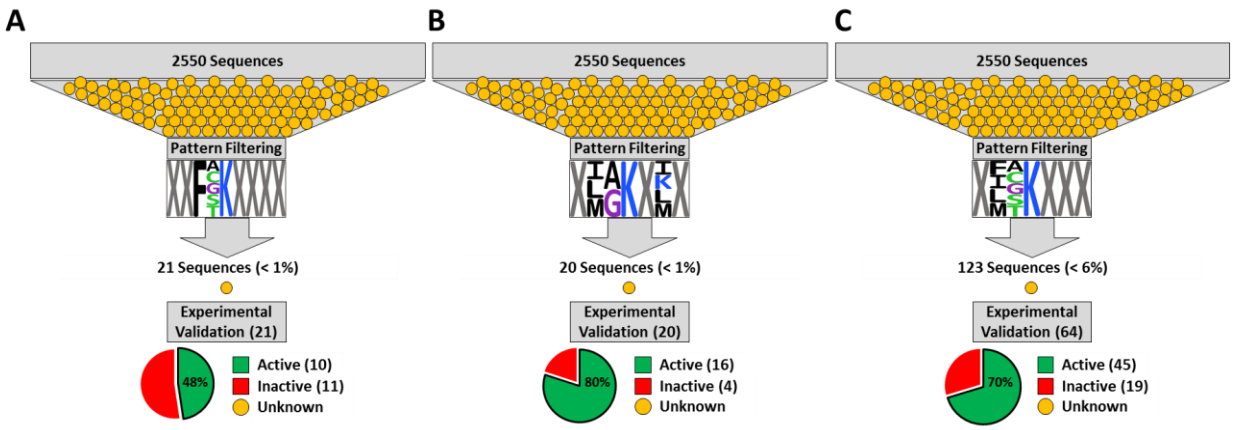


Figure 2.6.3. Blind predictions using the VIPER algorithm for uncovering new substrates of Smyd3. A) Smyd3 – VEGFR1 (PDB ID 5EX3) predicted motif used for filtering a library of 2,550 peptides yield 10 new hits. This corresponds to a 48% success rate by filtering of 99% of the library. B) Smyd3 – MAP3K2 (PDB ID 5HQ8) predicted motif used for filtering a library of 2,550 peptides yield 16 new hits. This corresponds to an 80% success rate by filtering of 99% of the library. C) Combined motif of Smyd3 used for filtering a library of 2,550 peptides. A total of 123 peptides matched the profile. Validation was done on 64 randomly selected peptides and 45 were found active, yielding a 70% hit-rate.

The high hit-rate achieved, and efficient library screening is very promising for its use as a quick method for identification of new binding partners. The complete list of methylated peptides can be found in the appendix 5.7 (Table 5.7.5, Table 5.7.6, Table 5.7.8) but includes interesting peptides such DNAPK and NHEJ1, both involved in DNA damage signaling and repair. Given the recent link discovered between Smyd3 and DNA damage repair (Chen Y-L., & al., 2017), those peptides would be worth investigating.

Overall, we have developed an algorithm for predicting specificity profiles that relies on a sequence optimization computational procedure which is then refined by crystallographic characteristics such as solvent exposure, B–factor and interfacial contacts. Overall, it can predict specificity profiles at an 84% accuracy and yield results significantly faster than experimental

assays. The algorithm has proven useful in the study of Smyd3 recognition profile as it was able to identify several (26) interacting peptides. This shows promising results for further usage and development. We called this algorithm VIPER and is made available to the scientific community as a webserver at http://viper.science.uottawa.ca. Details on the webserver application programming interface (API) and documentation can be found in chapter 3.

Chapter 3. VIPER: VIrtual PEptide array and Recognition motif Webserver

In the previous chapter, we described an improved computational protocol for predicting peptide recognition motifs of various enzymes that deposit post-translational modifications and proteins that participate in protein-protein interactions. We call this procedure VIPER for "VIrtual PEptide array and Recognition motif simulator". VIPER was shown to successfully recapitulate the recognition motif of a diverse set of protein-peptide complexes and was able to discover new binding partners of SMYD3 solely based on the recognition motif that was predicted. Unfortunately, VIPER relies on the use of third-party proprietary software, which prevents us from making it readily available to the scientific community. A web service hosting VIPER would circumvent this limitation but requires a fully automated implementation of the algorithm. This chapter describes how VIPER has been adapted for the web and provides documentation for properly using the service.

3.1 Resource management and scalability concerns

Porting to the web the VIPER algorithm as-is, is feasible, but computationally expensive. Each VIPER calculation on a single protein-peptide complex is performed using a structural ensemble of 120 templates as input to multistate analysis (MSA, see chapter 2 for more details), which requires 120 computer cores (one per template) and can take on average 1,000 CPU hours for a peptide of 6–8 residues in length. Therefore, a large amount of CPU resources would be required to adequately serve multiple clients at the same time. It is necessary to simplify the computational procedure to speed up calculations and decrease hardware requirements prior to making the VIPER method available to the scientific community.

To cut down computational resources and time, we first evaluated the possibility of reducing the template ensemble size to a minimum, without impacting the quality of predictions. Figure 3.1.1 shows how reducing the size of the ensemble from 120 to 1 backbone template could cut the CPU time needed to perform a VIPER prediction.

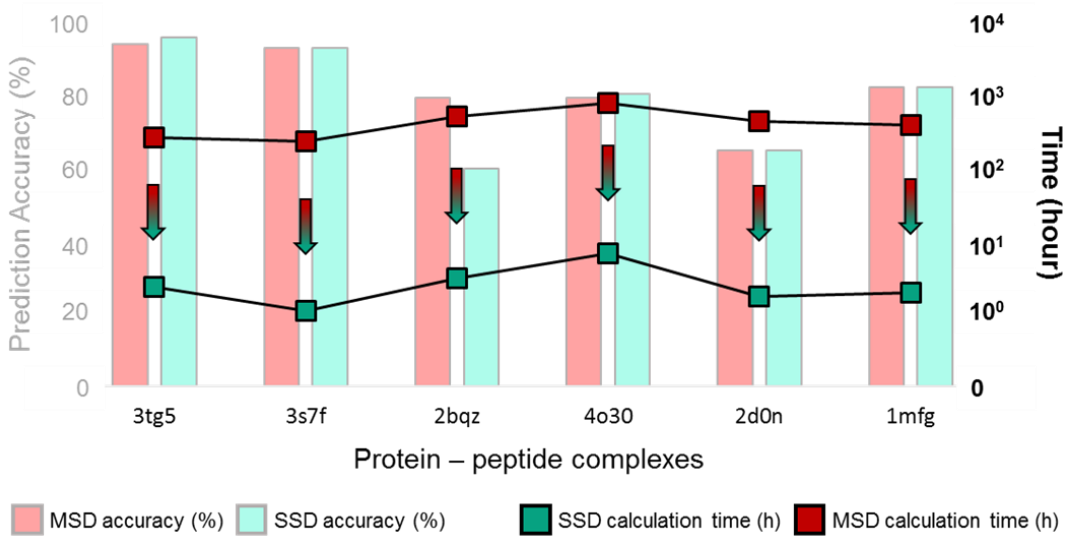


Figure 3.1.1. CPU time ($number\ of\ cores \times calculation\ time$) usage comparison of MSA vs SSD methodologies. Calculation time taken for VIPER on 120 PertMin input structures (multistate analysis, MSA, red squares) versus VIPER on a single minimized crystal structure (single-state design, SSD, green squares). It is possible to cut down calculation time by two orders of magnitude, while keeping accuracies (red bars: MSA, green bars: SSD) relatively constant. Time required to generate the PertMin ensemble for MSA was not considered, since it is highly dependent on the software use, but it took us typically 168 hours on single core.

As can be seen on Figure 3.1.1, only in the case of the Set8 methyltransferase (2bqz) did the SSD calculation yield a lower prediction accuracy compared with MSA. Analysis of the predicted recognition motifs (Figure 3.1.2, p–3 and p–1 of Set8 – H4K20) shows that positively charged residues such as lysine, histidine and arginine did not score as well in SSD (Figure 3.1.2. B) *versus* MSA (Figure 3.1.2. A). On the other hand, positions mostly stabilized by Van der Waals interactions such as p+1 of Set8 – H4K20 scored equally in SSD or MSA. At that position six

hydrophobic amino acids FILMVW are found in the top 8 scoring sequences, for both MSA and SSD. This observation matches our expectations of SSD versus MSA, since salt bridges and H-bonds are directional forces, thus are more dependent on the input structure than Van der Waals interactions. Figure 3.1.3 demonstrates how slight variations in atomic coordinates can affect computed energy values of directional forces. The arginine at p–3 of Set8 – H4K20 is held in place at the protein interface by multiple H-bonds, which have optimal geometries and distances based on molecular orbitals implicated in bonding. The residue scores badly in SSD (pink) however, small backbone perturbations found in one of the 120 ensemble members during MSA (green) allow the arginine to score in the top sequences.

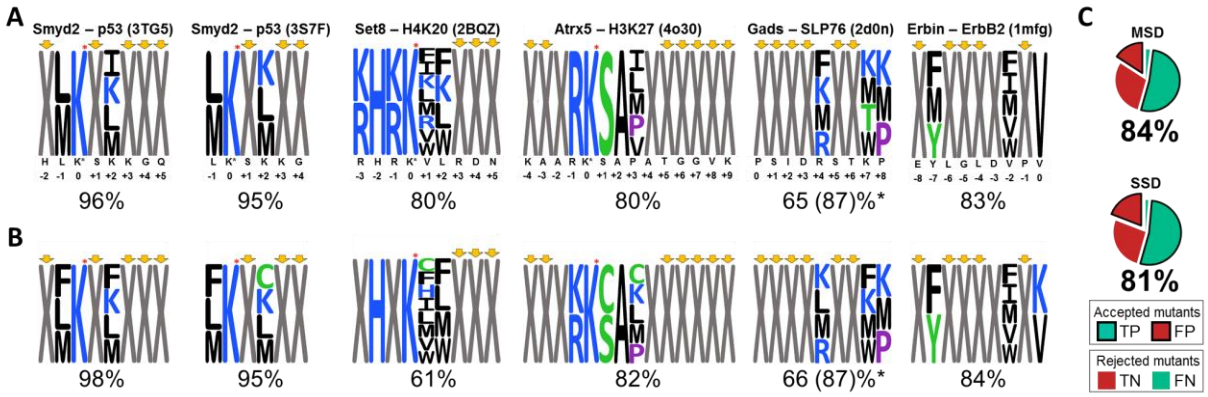


Figure 3.1.2. VIPER-MSA versus VIPER-SSD prediction accuracy. Comparison between A) VIPER on an ensemble of 120 structures (MSA) as described in chapter 2 and B) VIPER predictions using a single structure during mutational analysis: the minimized crystal structure (SSD). The accuracy is similar for both ensemble size apart from Set8 – H4K20, which shows a 20% lower accuracy using the SSD approach.

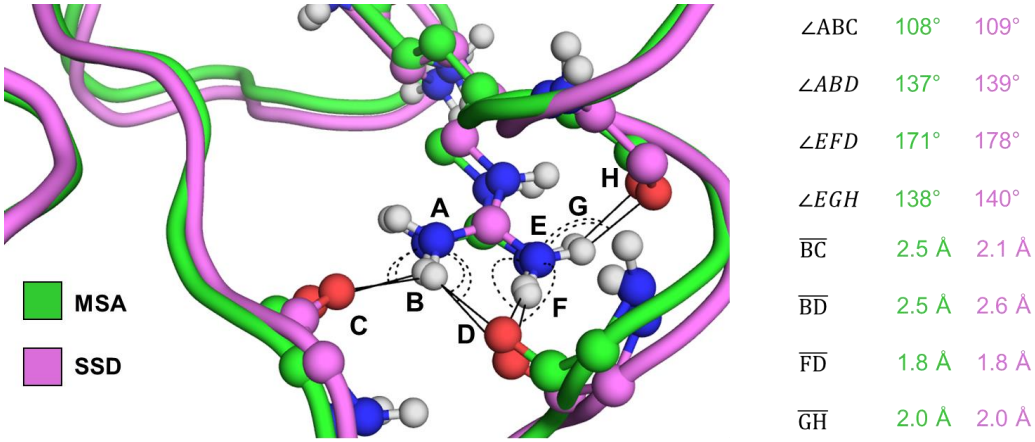


Figure 3.1.3. Best scoring backbone state for arginine at p–3 of Set8 – H4K20 (PDB ID 2BQZ) using VIPER as MSA (green) and SSD (violet). Only slight variations in backbone orientations leads the SSD state to score the wild–type arginine 12th versus 2nd (out of 20) for the MSD state. This shows how directional forces are highly impacted by initial state coordinates.

Overall, the VIPER algorithm built on top of an MSA versus SSD methodology represents better, on average, the experimental recognition motif with accuracies of 84% and 81%, respectively). From a science perspective, we would still recommend MSA over SSD when confronted to fixed backbone and discrete rotamer threading algorithms. However, since the average prediction accuracy of SSD lies 3% below the MSA implementation but reduces computational cost by 120-fold (Figure 3.1.1), modifying VIPER towards an SSD implementation would be desirable from a web perspective since users could submit a job and access the results within the same day.

3.2 Porting VIPER to the Web

Implementing VIPER behind a REST service is synonym of automation and error handling. Since the algorithm is built on multiple third-party software, various API layers are required to assemble a pipeline constituted of these programs. Consequently, a java (https://www.java.com) framework was built on pre-existing chemical libraries such as BIOJAVA 5.0 (Prlić A. & al., 2012) and CDK

2.0 (Steinbeck C. & al., 2003). This framework is aimed at easily manipulating chemical data from various input sources. It allows for standardized inputs and outputs of programs written in different languages such as java, C++, python and shell so that they can be seamlessly linked together. The overall pipeline workflow is described in Figure 3.2.1. First a PDB file is read from our framework parser, which creates a molecular data representation in memory. Then, the structure is repaired, which includes fixing missing residues, building loops and adding hydrogens (step 1). Subsequently, the structure undergoes a mutational analysis (step 2), after which physico-chemical properties are calculated (step 3) to finally output a recognition motif and a virtual peptide array (step 4). The results can then be visualized on the server and downloaded. Each of theses steps are described in more detail below.

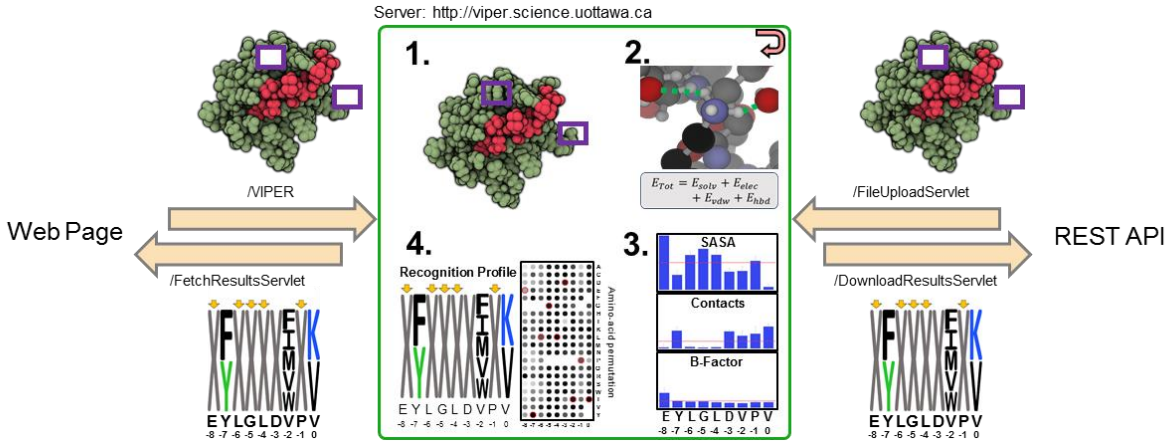


Figure 3.2.1. VIPER Webservice workflow. The service is available at "http://viper.science.uottawa.ca". A job can be submitted via the user interface served at 'http://viper.science.uottawa.ca' or by posting a form to the REST API available at 'viper.science.uottawa.ca/FileUploadServlet'. The structure provided is repaired and protonated (1), then sequence optimized (2) to rank stable mutations (section 2.1). Subsequently, tolerance predictions for each position is performed (3), according to section 2.5. Ultimately the best mutants are combined into a recognition profile (4). The profile can be retrieved via the web page or by a GET request to the API.

3.3 Uploading a Protein Structure

This first step for predicting a recognition motif is to provide structural details of the protein-peptide complex under study. Structural information can be stored in various formats, such as PDB, MMTF, SDF, CIF. As of the first release of VIPER, only the most common format—PDB file format—is supported. The biggest repository of molecular structures providing PDB files is the RCSB (www.rcsb.org). It is thus possible, from VIPER, to provide the PDB identifier (Figure 3.5.1 E), and the server will fetch from the RCSB, the file requested. If the structural data available is an "in-house" PDB file, not yet deposited, it is still possible to upload it to VIPER, using the upload manager (Figure 3.5.1 D).

Many Protein Data Bank (PDB) files at RCSB are incomplete, and thus VIPER will attempt to rebuild the missing residues and loops. These are identified by comparing the ATOM record to the SEQRES record of the PDB file. The algorithm then iterates over each residue and compares the number of atoms parsed from the ATOM record to the theoretical value of atoms for the corresponding amino acid. This information is then written to a python script (appendix 5.9) for use with MODELLER (Webb B. & Šali A., 2016). This third-party software rebuilds the coordinates of the missing loops and atoms based on a two steps protocol: the model is optimized using the variable target function method (VTFM) with conjugate gradients (CG) and then refined by molecular dynamics (MD) combined with simulated annealing (SA) (Šali A. & Blundell T. L., 1993). This is the default procedure provided by the software and has not been optimized in the context of motif predictions, as the main goal here is to produce a valid structure for energy minimization. Otherwise, gaps in a structure creates loose ends which cause local unfolding of the structure upon minimization. This step is automated, and the user has no control over the many variables of loop modelling. For full control, we suggest that the user build the missing residues

on the structure using any software available, before submitting the PDB file. It is suggested to rebuild the loops manually only if the missing residues are close to the bound peptide and might interact with it, since the final loops will greatly affect the outcome of the motif predicted. For any other loops, the automated procedure should be the preferred workflow.

The PDB file format also have the possibility to contain "ALT LOC" records, for alternate positions of residue side chains. If not removed by the user manually, VIPER will only retain the highest occupancy record of the ALT LOC for each residue. In case of equivalence between records, the first record read will take precedence. Just like loop modelling, it is advised to choose the ALT LOC records manually only if these might interact with the peptide.

Finally, PDB files also contain HETATM records (Heteroatoms). These residues can take multiple forms such as non-canonical amino acids (*e.g.* methylated lysine), inorganic (*e.g.* Zn^{2+}, Na^{+}) and organic small molecules, (*e.g.* S-adenosylmethionine) or solvents (*e.g.* H_2O). Most of these records require special treatment since most chemical computing freeware do not handle them properly when dealing with parameterization. Canonical amino acids have predefined atom types in common force fields implementations, but small molecules do not, thus it makes it difficult to guarantee error-proof methods for determining atom types for such records. VIPER has an automated procedure for parameterizing small molecules, but it is suggested that any HETATM records (apart from H_2O molecules) that does not interfere with the peptide shall be removed from the PDB file before uploading it.

For proper sequence optimization, hydrogen atoms are required. Since those are not provided in most PDB file, our in-house Java framework interfaces with REDUCE V3.23 (Word J. M., & al.,

1999) to place hydrogens in a standardized geometry and to optimize the orientations of alcohols, thiols, amines, amides, methyls and imidazole rings. However, REDUCE is dictionary based and sometimes, atom naming and/or uncommon geometries prevent the software from proper hydrogen placement (Figure 3.3.1). Therefore, we developed an in-house hydrogen adder to fix these potential issues which fundamentally relies on proper reading of the atom's chemical environment.

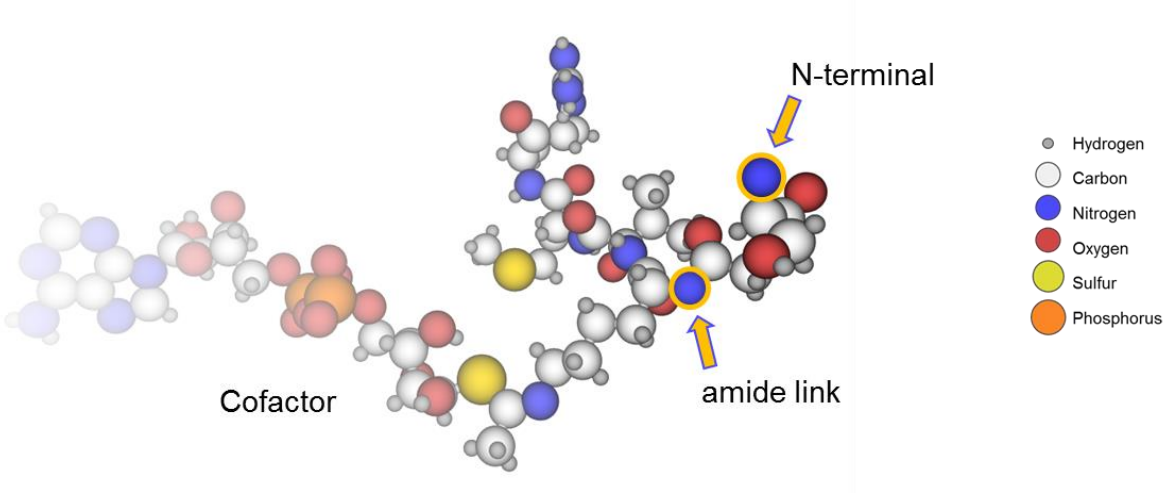


Figure 3.3.1. Sample errors output from REDUCE v3.23 upon protonation of the AceCS2 peptide chain of the human Sirt3 acetyltransferase (PDB ID 3GLT). N-terminal hydrogens as well as the cofactor amide backbone hydrogen are missing.

The algorithm starts by assigning single bonds between atoms based on pair-wise distance. Accepted distances for single bonds are computed based on theoretical covalent radius of the atom elements and within a 0.3 Å buffer to account for the inaccuracies of atomic coordinates in PDB files. For speed improvement, we devised a partitioning algorithm which caches atoms into boxes of 2 Å cubic sides, based on their atomic coordinates (Figure 3.3.2.).

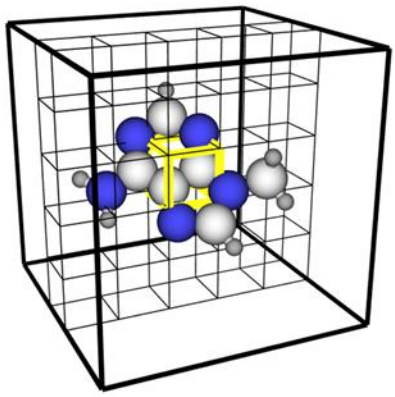


Figure 3.3.2. Bounding box of a 3D coordinate system. The box is sub-divided into 2 Å boxes, into which each atoms of the system are stored, according to their atomic cartesian coordinates. This pre-partitioned structure allows much faster access to neighboring atoms.

This provides a data-structure to query for fast identification of nearest neighbors of a target atom since looping over all atoms is not required to determine if an atom should be considered for bonding. Next, the bonded structure is morphed to its graph representation and a recursive depth-first search identifies every double bond pattern possible within delocalized systems. When the double bonds are assigned, inter-atomic distance is computed and matched to the theoretical double-bond distance of both atoms. If the distance is closer to a single bond, the double bond is not assigned. Then, bond angles are also used as a constraint to prevent incorrect geometries. Each possibility is scored according to aromaticity, charge separation, and the number of double bonds assigned. The scoring schemes goes as follow: each atom in the delocalized system contributes to +1 to the final score if it is in an aromatic ring, +0 if non-aromatic or -1 if anti-aromatic, according to the Hückel theory which defines aromaticity as cyclic delocalization of $4n + 2$ π electrons. Then each double bond assigned contributes to +1. Ultimately, any charges generated on atoms upon double bond assignment, such as a charged tertiary amine, contributes to -1 towards the final score.

The best double bond network is kept. Ultimately, missing hydrogens are added in a standardized geometry to fill in any incomplete atom octet. The geometry used is 109.45° for “sp3” atoms, 120° for “sp2” atom and 180° for “sp” atoms. Unfortunately, refinement of electron density maps often leads to improper bond length and angles for better modelling of the density. This is a major issue when assigning bonding networks to unprotonated systems. Consequently, our method is not error-proof, since some atomic coordinates errors can hardly be solved without more structural knowledge but offer a name-independent algorithm to read chemical environments that complement REDUCE reasonably well. We are currently applying the algorithm to whole protein structures instead, not just small molecules. Thus, REDUCE will be replaced in the following stable version of VIPER, as we are currently finishing the benchmarking of our own protonation algorithm (Appendix 5.8).

3.4 User Interface Overview

VIPER is a web application that allow user to predict a recognition profile—according to the algorithm describe in chapter 2—for any protein-peptide complex for which 3D coordinates have been resolved. To predict a recognition motif, the user must, first, upload the desired 3D structure to the server using the predefined fields under 1. Load Structure. The crystal structure can be imported from the local file system (Figure 3.4.1 D) or from the RCSB (Figure 3.4.1 E).

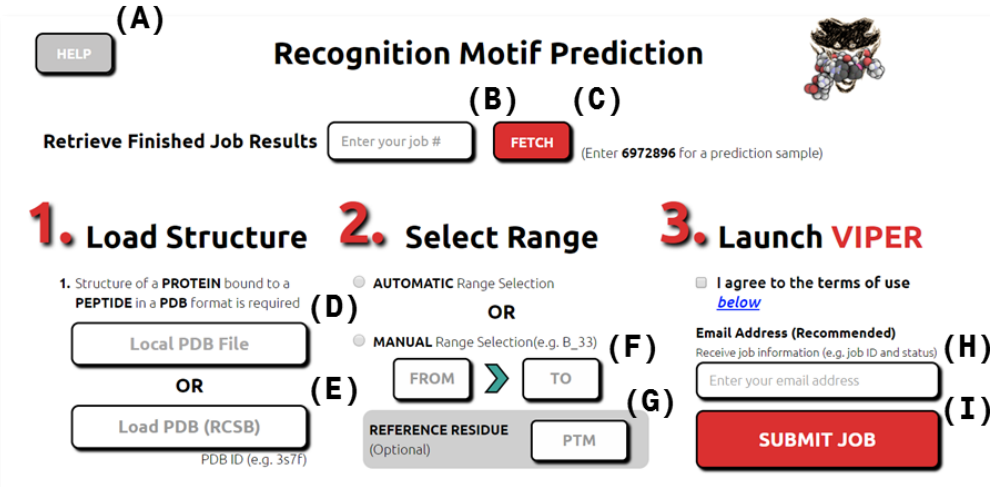


Figure 3.4.1. Overview of the main VIPER panel (http://viper.science.uottawa.ca). A) Press the help button to be redirected towards the documentation page, for various tutorials on how to use VIPER. B) This text box requires a 7-8-digit unique job identifier. Pressing the Fetch (C) button will request the result page for the job id found in textbox (B). Step 1 – To launch a recognition motif prediction on a PDB file from the local file system, press on the Local PDB File (D) button. To initiate the process on a remote PDB file from the RSCB, enter the 4-digit identifier in the textbox below (E). Step 2 – Select peptide range selection detection as either automatic where the peptide chain will be identified and in its entirety for motif prediction, or manual (F), to forcibly include / exclude certain residues of the peptide chain from the prediction. If the peptide contains an anchoring residue, such as a residue undergoing a chemical transformation upon binding, it is possible to fix its identity by providing its position in the peptide (G). Step 3 – Before launching a job (I), it is advised to provide an email address (H) to be informed on job completion.

Once the file is uploaded, the webserver will evaluate its validity; verifying first the file format and second the structure encoded. If the detection algorithm cannot identify a protein-peptide complex, the user will be notified (Figure 3.4.2.), otherwise, a summary of the structure appears for confirmation that VIPER correctly identified and assigned each protein chains.

After the file has been validated by VIPER, a residue range can be specified to restrain motif predictions. Under 2. Select Range, the residue positions to be included in the motif may be manually set or automatically determined by VIPER (Figure 3.4.1. F). It is also possible to define

specific residues in the peptide as Fixed (Figure 3.4.1. G)—these residues will appear in the predicted motif, but their relative preference will be set to 100%.

When the PDB file has been specified and validated, residues to include in the motif have been selected and fixed residues have defined, the job can be submitted to the server using the submit button under 3. Launch VIPER (Figure 3.4.1. I). The user will immediately be redirected to a webpage (Figure 3.4.3.) indicating the identification number assigned to the job. This number can be entered in the main page (Figure 3.4.1 B) to access the job status and results. The user can also choose to be sent—optionally—via email the information concerning the job submitted by filling the corresponding section (Figure 3.4.1 H).

When accessing the results page of VIPER, either via the link provided in the email, or manually by fetching the job ID from the VIPER main page, the user can verify the job status (Figure 3.4.4 A-C). The job can be in one of the following states: *Queue*, *Running*, *Done* or *Error*. A job is submitted to computer cluster via an SGE (Sun-Grid-Engine) batch queuing system, therefore, it is not immediately executed. Depending on server load, the job will stay in the *Queue* state, until

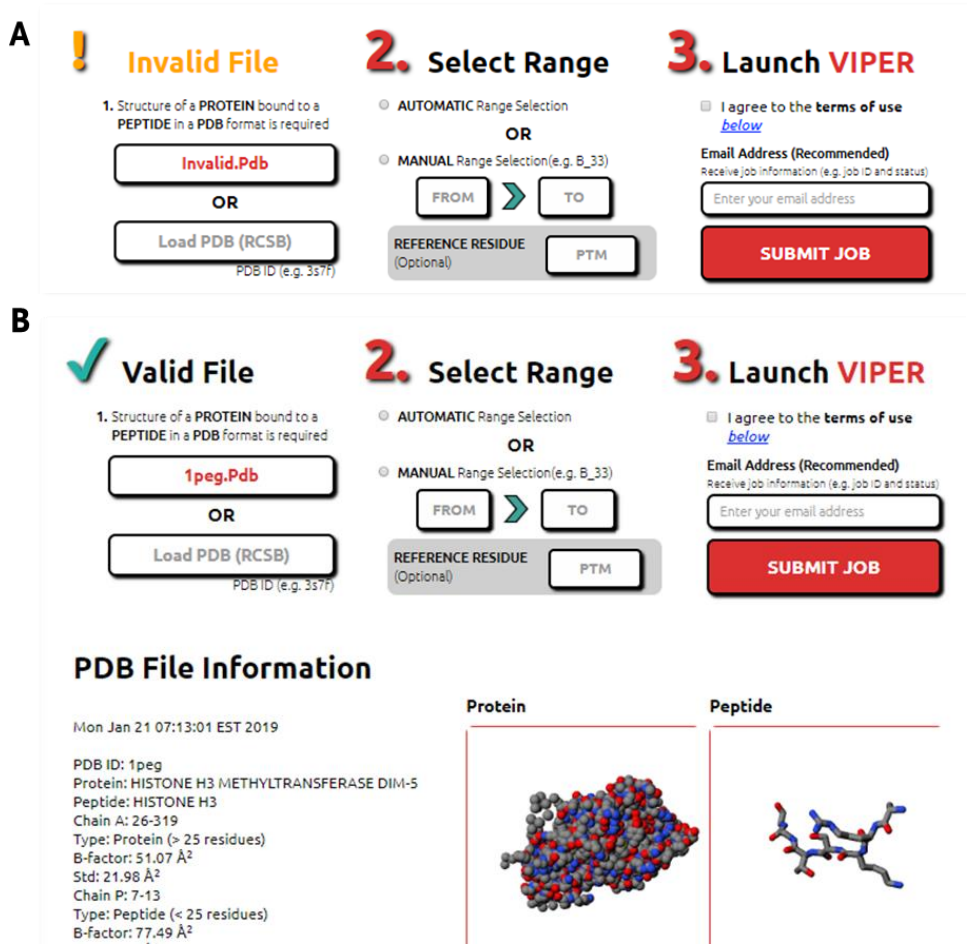


Figure 3.4.2. Submitting a structure to VIPER. A) State of the application when an invalid PDB file is uploaded. First, PDB file format validity is checked, then if passed, the algorithm tries to detect a protein and a peptide chain in the structure provided. If any of these step fails, the "1. Load Structure" is replaced by "! Invalid File". B) An valid PDB file format was provided and a protein-peptide complex was identified. A recognition motif prediction can be done on the provided structure. Details information on the uploaded structure is shown for the user to make sure the file uploaded corresponds to what is expected.

computer resources are available to run the job. From our benchmarks, (Figure 3.1.1), recognition motifs for 4-8 residues will run within a 12h, after leaving the queue. When the predicted motif is available, the status changes to *Done*.

The Recognition Motif for the protein structure submitted is shown in the main section (Figure 3.4.4 F). Motifs are read from left-to-right and match the residue range defined by the user. Each letter in a column represent one of the 20 amino acids. The relative height of the letters has no significance. The purpose of the motif is simply to show which permutations are allowed, as per some predefined rules, explained in detail in chapter 2. A color code has been implemented to quickly observe trends in the motifs, where Blue, red, green, black, purple represents basic, acidic, polar, hydrophobic and special amino acids respectively. A position marked by a red star indicates that this residue was set to *Fixed* at submission.

A virtual peptide array is provided in the results page. The array reflects the energy differences calculated by VIPER for every residue at each position. By comparing two residues for the same position in the array, the color tone indirectly indicates the relative stability of both residues. The darker amino acid spot is more stable (relative to other residues for that position). The wild–type amino acid at each position is marked by a red circle.

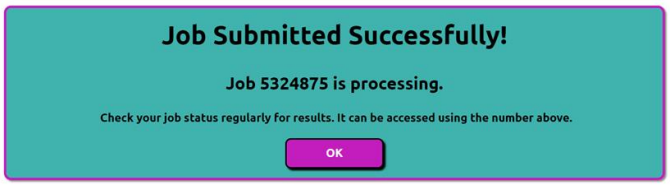


Figure 3.4.3. Job submission webpage indicating the job number used for future reference. This number can be entered in the VIPER main page to fetch a finished job.

Finally, a *Download* button can be triggered to obtain various output source files from the algorithm used to build the peptide array and the motif. Computed energy values for each amino acid at every position as well as PNG images of the peptide array, recognition motif and various graphs used to predict tolerant positions (Figure 3.4.4 E). Information on the bound peptide such

as relative solvent exposure, backbone ensemble RMSD, protein contacts can be extracted from the plots provided. These graphs were used in determination of tolerant positions. Please refer to chapter 2 for more details on how this information was used.

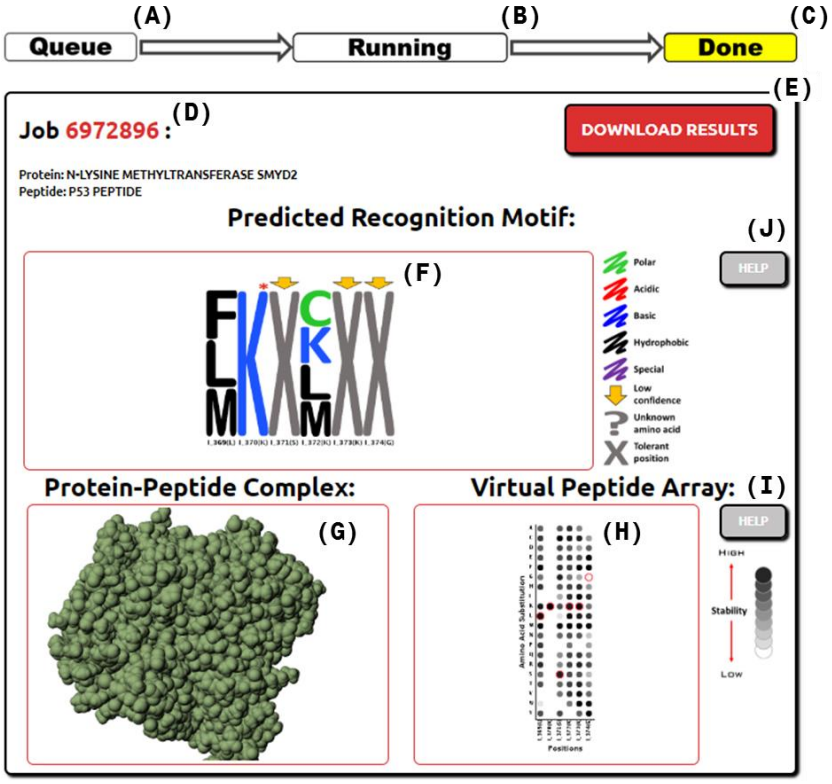


Figure 3.4.4. The VIPER Result page groups the job status, identifier, predicted motif and predicted peptide array into a single panel. The status appears in the page header and is highlighted in yellow. It can adopt 3 states: queued (A), running (B) or done (C). The Job identifier is indicated below (D). When the job is done, a Download results (E) button is available to get a local copy of the recognition motif, the virtual peptide array, and the energy values calculated for every amino acid at each position in the recognition motif. The recognition motif predicted is displayed (F) alongside its corresponding peptide array (H). A visual inspection of the PDB file submitted can be made (G) to uncover potential abnormalities, if any, in the prediction. Documentation on the recognition motif and peptide array can be accessed by pressing the help buttons (J, I) respectively.

Additionally, the results page provides a molecular viewer (http://www.jmol.org) for quick inspection of the final molecule used for calculations. Just like any molecular viewer, the user can

rotate, zoom, select, show and hide parts of the molecule. This tool can be used to detect any abnormalities in the preparation and cleaning steps of the structure by VIPER. It is thus recommended to inspect the final molecule to validate the output results from VIPER.

3.5 Data generation and analysis

When submitting a recognition motif prediction job to VIPER, it automatically performs a mutational analysis on the whole peptide. A peptide is detected when a chain found in the structure provided contains 25 residues or less. If no such chain is detected, the job will exit with error. Any modified amino acids found in the peptide chain will be reverted to their canonical form (e.g. methylated lysine is reverted to lysine) but if the HETATM record cannot be mapped to any amino acid, it will be excluded from the motif and any calculations. To prevent a residue from being part of the mutational analysis, while being kept in the motif profile output, it can be explicitly marked as a Reference residue (Figure 3.4.1 G). To make sure that the range desired in the motif is included in the calculations, it is possible to override the automated VIPER detection procedure by entering manually the residue range (Figure 3.4.1 F).

The next step in VIPER consists of launching the PHOENIX sequence optimization protocol. Energy calculation using force fields requires extensive description (parameterization) of each atom in a system. PHOENIX provides parameters for all 20 canonical amino acid atoms, but any small molecules such as cofactors, must be parameterized individually. The PHOENIX force field (section 2.1) is based on the Dreiding (Mayo S. L. & al., 1990) atom types and the PARSE charge model for energy calculation (Yang Q. & Sharp K. A., 2006).

Using our in-house chemical environment reader, a DFS assigns functional groups (FG) to atoms based on connectivity. Using a recursive DFS on every hetero-atoms speed up the process as hetero-atoms are found in most FGs, whereas carbon atoms and hydrogens are not. Atom charges were assigned based on the FG detected, as described by the PARSE model. Finally, we assigned Dreiding atom types based on their respective hybridization, delocalization, elements and number of bonds.

After PHOENIX has produced the mutational analysis and amino acid ranking for the defined residue range, SASA, B–factor and interactions are computed to generate the final motif. (Figure 3.5.1). For each peptide position, a set of amino acids in the form of the 1-letter code arranged in columns are produced. Put side-by-side, these create the recognition motif for the protein. A position marked with a red star indicates a position that was flagged as Reference. This position was thus not part of the mutational analysis and is outputted as is.

A position where the 1-letter code corresponds to an X denoted a tolerant position. This position was predicted as such for various reasons (consult chapter 2 for in depth details of the algorithm), such as high relative B–factor, low number of protein interactions, high relative solvent exposure or most of mutations were evaluated as stable at the interface. Consequently, VIPER suggests that these positions are not relevant for peptide binding or recognition. Of note, the recognition motif does not provide any information as to which mutations are more likely at any given positions.

Each amino acid in the motif is colored according to polarity, charge, hydrophobicity or phi-psi Ramachandran maps. Blue corresponds to basic amino acids, where as red corresponds to acidic amino acids. Green corresponds to polar, uncharged amino acids and black corresponds to

hydrophobic amino acids. Finally, purple corresponds to amino acids with particularities, such as low (glycine) or high (proline) hindrance to backbone rotation. This color scheme is mainly used for quick inspection of the motif. It allows to observe trends at certain regions and positions.

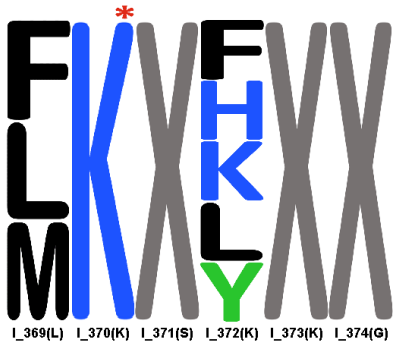


Figure 3.5.1 Typical recognition motif obtained from VIPER. Each column identified by the residue position is populated with 1-letter code amino acids. These amino-acids were predicted to be stable at the interface with the protein in the peptide context. A column marked with a red star indicates a residue that was flagged as Reference by the user. It was not included in the mutational analysis, but still appears in the motif. When a column contains an X, VIPER evaluated that this position was tolerant, by analyzing various factors including B–factor, contact map, accessible surface area, and stable amino acids. Each amino acid type is colored differently according to polarity, basicity, acidity, hydrophobicity or phi-psi maps.

The recognition motif predicted provides a quick overview of tolerated amino acids across the peptide, which can then be combinatorially assembled to generate numerous sequences. These sequences can be BLASTed (https://blast.ncbi.nlm.nih.gov/Blast.cgi) against databases to find potential peptide binder. The user might want to use custom restraints to limit the sheer amount of hits possibly generated, such as co-location, protein families or biological roles. The final remaining sequences represents a pruned dataset that is enriched with binding peptides (hits), thus enhancing the success rate of experimental screening, as demonstrated in chapter 2.

Next to the recognition motif in the VIPER result page lies a virtual peptide array (Figure 3.5.2). It is a matrix representing the relative stability of each mutation per peptide position. The wild type amino acid in the original bound peptide is identified by a red circle. The spots are colored from white to black with a continuous gradient. A black spot represents a mutation with lowest computed energy (more stable at the protein interface) and a white spot corresponds to the highest computed energy. The spots are colored based on their energy normalized within the distribution of energy computed position-wise. It is important to note that spot intensity cannot be compared across peptide positions, since the distributions of energy varies significantly from one position to another. Thus, two similar energy values may be represented with different shades when belonging to two distinct position. Nevertheless, the peptide array provides a mean to visually analyze the computed energy values available from VIPER.

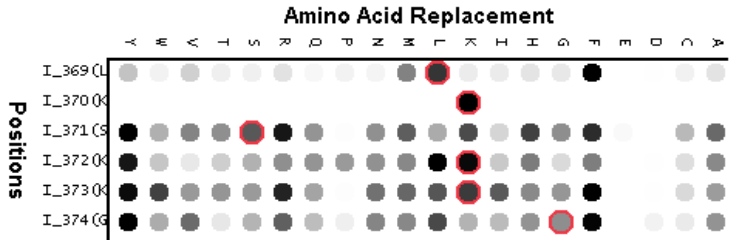


Figure 3.5.2. Typical virtual peptide array outputted by VIPER. Each row corresponds to a different peptide position. All 20 amino acids are represented at each position with a dot colored from white to black. The spot intensity is computed based on the energy of the mutation normalized within the distribution of energies at the given position the lighter the spot, less stable is the mutation at the interface. The wildtype amino acid is identified with a red circle. Since spots are normalized position-wise, they cannot be compared across positions.

3.6 Common Errors

The most common reason for a motif prediction job to fail and exit with error is when an unknown HETATM record is encountered during mutational analysis. These unknown residues will most

likely cause energy calculation errors, since they cannot be parameterized properly by the algorithm. The easiest way to prevent such errors is to remove these molecules from the PDB file or mutate them to their canonical amino acid counterpart.

Other job failures might occur when missing loops in the provided structure cannot be recovered properly. For example, two consecutive missing loops separated by only 2 residues or less, might not be rebuilt correctly, and such atypical cases should be handled by the user manually with loop modelling software.

3.7 Limitations of the method

Recognition profiles are made from *in-silico* binding predictions. Care must be taken if the protein for which a profile is being generated is an enzyme. Since catalysis does not only require binding, but also many other steps along the way, binding predictions are not chemical transformation predictions. During our benchmarking, methyltransferases were considered and found successful, but a more complex chemical transformation might involve others steps in between, which are not accounted for in our methodology. What might have helped us in being successful concerning methyltransferases is that the binding pose found in the crystal structures were representative of the most probable conformation leading to catalysis. By optimizing the closest conformation leading to a chemical transformation we were likely to be successful. In summary, the algorithm will be successful as the quality of the input conformations. This is true for chemical transformations, but also for any other binding predictions made during the benchmarking.

The challenge comes from judging the quality of the input for predictions. In this case, chemical intuition may help, but the best insight comes for experimental data. More experimental data

necessarily offers more understanding of the protein and its mode of action. Nevertheless, in absence of much data, one can look at the electron density of the input structure to make sure all atoms are really accounted for. Missing atoms, or a full loop around the bound peptide is a hint that the structure is not well resolved and that important interactions might be missed when comparing mutants. VIPER will build all missing atoms and loops, but homology models may be off-target. For this reason, any missing atoms near the peptide may lead to inaccurate predictions made by VIPER.

3.8 Conclusions

The algorithm devised initially for predicting recognition motifs of proteins (chapter 2) was requiring too many computational resources for a web service. A quick performance analysis of the procedure revealed that most of the time was spent during the mutational analysis step which was done on every member of the structural ensemble considered (MSA). For that reason, we experimented with the number of structures used in the PertMin ensemble to cut down mutational analysis time to a minimum. Overall, we realized that considering 1 single structure during that step instead of 120 did not affect much the accuracy of the predicted motif (81% vs. 84% on average, respectively) but cut down the time drastically. It was surprising that a single state approach was giving similar results as a MSA methodology but since the VIPER algorithm does not only rely on energy stabilisation but also B–factors, interaction maps, and accessible surface area of the peptide, this behavior can be rationalized. We were able to predict a recognition motif consistently under 12 hours and using a single CPU. This improvement was significant and solved the initial concern on server load.

With a viable algorithm at hand, we linked each step altogether using an in-house API layer to abstract the data manipulations that had to be done when juggling from one third-party software to another. On top of that API, we designed a simple web interface for submitting recognition motif predictions. The web service is available at http://viper.science.uottawa.ca. The job submission interface is simple to use and provides a mean to visualize and download the results. Prediction motifs should be available within 12 hours after the job submission, depending on the server load.

Chapter 4. Conclusions and Perspectives

Current experimental methods for obtaining protein recognition motif such as peptide arrays are costly and time consuming. They require specific lab equipment and skilled workers. These constraints motivated us to devise an algorithm capable of successfully generating recognition motifs for proteins of comparable quality to peptide arrays but at no cost and much quicker. To undertake this challenge, we used computational protein design (CPD) methodologies. We hypothesized that full mutational analysis of the bound peptide at the interface of a protein partner should yield most stable mutations which would, in turn, represent the recognition profile. To test our hypothesis, we scanned the literature for protein-peptide complexes for which a crystal structure was available and of reliable quality but also for which an experimental peptide array has been published. We were able to assemble a dataset of 6 protein-peptide complexes: Smyd2-p53 (PDB id 3S7F,), Smyd2-p53 (PDB id 3TG5), Set8-H4K20 (PDB id 2BQZ), Atxr5-H3.1k27 (PDB id 4O30), Erbin-ErbB2 (PDB id 1MFG), Gads-SLP76 (PDB id 2D0N).

By applying a multi-state analysis methodology as reported by Lanouette S. et al., 2015, we were not able to reproduce accurately recognition motifs as reported by the experimental peptide arrays (63% on average). The algorithm performed well for stringent positions but demonstrated low accuracy for tolerant positions. Tolerant peptide position might be more dynamic than other positions—this behavior cannot be captured by the ensemble, and thus yield results biased towards the crystal structure's binding pose. To solve this issue, we added an additional computational layer for dealing with such positions. The algorithm involves evaluating solvent exposure, protein contacts and residue B–factors in addition to the mutational analysis data obtained using CPD to detect tolerant positions. Both algorithms combined for predicting recognition motif yielded 84% accuracy on average for the 6 proteins of the dataset.

To make this methodology available for a large community, VIPER was slightly modified for a web implementation. The methodology employed was revised to use a single structure (SSD) instead of 120 (MSA) to cut down on computational resources. We compared the accuracy of MSA versus SSD and observed a 3% average decrease in accuracy (84% and 81% respectively) but a speed improvement of 120-fold. The speed improvement was significant and allowed us to provide VIPER as a web service (http://viper.science.uottawa.ca).

4.1 Impacts

VIPER was used to predict a recognition profile for Smyd3 and identify new binding partners of that novel protein. Smyd3 is a methyltransferase for which no recognition motif has been published at the time of writing this thesis. Two crystal structures bound to two different peptides existed in the Protein Data Bank (PDB ids 5EX3 and 5HQ8). VIPER predicted a recognition motif for each structure and a collaborator (Kyle Biggar) screened a peptide library of size 2500 to find potential hits. The Smyd3-VEGFR1 and Smyd3-MEKK2 predicted motif retained 21 and 20 peptides respectively. These peptides were tested on a peptide array experimentally to validate if they were truly binding partners of Smyd3 or simply false positives. Finally, 10 and 16 peptides out of the 21 and 20 hits respectively were binding to Smyd3, as indicated on the experimental array. This demonstrates how VIPER can be used as a cheap and swift alternative to traditional peptide arrays when searching for new substrates of a protein.

4.2 Future work

The promising results obtained with VIPER for discovering new binding peptides of Smyd3 could motivate us to push forward the algorithm and include in the pipeline predicted motif screening of large databases and guided blast searches. Motif screening would provide a large quantity of

sequences matching the motif and the guided blast search could limit the list of potential hits to proteins or peptides matching specified criteria such as a specific organism or cellular location.

As by writing this thesis, most of the literature, to our knowledge, was scanned to find all potential peptide arrays and crystallized matching protein-peptide complexes. Further benchmarking and fine-tuning of the method could be performed as the literature gets richer in experimental SPOT peptide arrays and crystallized protein-peptide complexes. As more protein-peptide complexes are processed by VIPER, we can provide sound ground to prove its true predictive power but also identify its limitations.

It would also be interesting to generate a 2nd version of VIPER in which robustness of data handling would be enhanced. PDB files contain many non-standard amino acids, special chain breakages, missing electron densities, multiple chains, etc. This makes automated data preparation challenging, and prone to failure. In addition, a proper API layer on top of the VIPER algorithm would make the service broadly applicable by third-party software, which would greatly improve visibility of VIPER. It would also allow for high-throughput and batch generation of recognition motifs, if needed.

References

L. Brocchieri and S. Karlin, Protein length in eukaryotic and prokaryotic proteomes, Nucleic Acids Research, 33:3390, **2005**

Kubelka J, Chiu TK, Davies DR, Eaton WA, Hofrichter J., Sub-microsecond protein folding, Journal of Molecular Biology, 359(3):546-53, **2006**

Levinthal C., Mossbauer Spectroscopy in Biological Systems Proceedings, University of Illinois Press, 22-24, **1969**

Karplus, M. & Weaver, D.L., Protein-folding dynamics. Nature, 260, 404-406, **1976**.

Karplus, M. & Weaver, D.L., Protein folding dynamics: the diffusion - collision model and experimental data. Protein Sci, 3, 650-668, **1994**.

Nolting, B. & Agard, D.A., How general is the nucleation-condensation mechanism? Proteins, 73, 754-764., **2008**.

Dill, K.A., Theory for the folding and stability of globular proteins. Biochemistry, 24, 1501-1509, **1985**.

Frauenfelder, H.; Sligar, S.G. & Wolynes, P.G., The energy landscapes and motions of proteins. Science, 254, 1598-1603., **1991**.

Dill, K.A., Chan H.S, From Levinthal to Pathways to Funnels, Nature Structural Biology, 4, 10-19, **1997**.

Dunitz, J.D., Win some, lose some: enthalpy-entropy compensation in weak intermolecular interactions. Chemical Biology, 2, 709-712, **1995**.

Gilli, P.; Ferretti, V.; Gilli, G. & Borea, P.A., Enthalpy-entropy compensation in drug receptor binding. Journal Physical Chemistry B, 98, 1515-1518, **1994**.

Sali, A. & Blundell, T.L., Comparative protein modelling by satisfaction of spatial restraints. Journal Molecular Biology, 234, 779-815, **1993**.

Kauzmann, W., Denaturation of proteins and enzymes. In: The mechanism of enzyme reaction, W.D. McElroy, & B. Glass, (Eds.), pp. 70-120, Johns Hopkins Press, Baltimore, **1954**.

Kauzmann, W., Some factors in the interpretation of protein denaturation, Advance Protein Chemistry, 14, 1-63, **1959**.

Lum, K.; Chandler, D. & Weeks, J.D., Hydrophobicity at small and large length scales, Journal Physical Chemistry B, 103, 4570-4577, **1999**.

Stillinger, F.H., Structure in aqueous solutions of nonpolar solutes from the standpoint of scaled-particle theory, J Solution Chem, 2, 141-158, **1973**.

Zhou, R.; Huang, X.; Margulis, C.J. & Berne, B.J., Hydrophobic collapse in multidomain protein folding, Science, 305, 1605-1609, **2004**.

Eaton B.E, Gold L. & Zichi A.D. Let's get specific: the relationship between specificity and affinity, Crosstalk: Chemistry and Biology, 2, 633-638, **1995**.

Fischer, E., Einfluss der configuration auf die wirkung der enzyme, Berichte der Deutschen Chemischen Gesellschaft, 27, 2984-2993, **1894**.

Koshland, D.E.J., Application of a theory of enzyme specificity to protein synthesis, Proceedings National Academic Science USA, 44, 98-104, **1958**.

Ma, B.; Kumar, S.; Tsai, C.J. & Nussinov, R., Folding funnels and binding mechanisms. Protein Engineering, 12, 713-720, **1999**.

Tsai, C.J.; Kumar, S.; Ma, B. & Nussinov, R., Folding funnels, binding funnels, and protein function. Protein Sci, 8, 1181-1190, **1999a**.

Liu S. Q., & al., Protein Folding, Binding and Energy Landscape: A Synthesis, **2014**.

Apic G., Ignjatovic T., Boyer S., Russel R. B., Illuminating drug discovery with biological pathways, FEBS Letters, 579 (8), 1872-1877, **2005**.

Tomar N., De R. K., Comparing methods for metabolic network analysis and an application to metabolic engineering, Gene, 521 (1), 1-14, **2013**.

Thompson M.C., Barad B.A., Wolff A.M., Cho H.S., Schotte F., Schwarz D.M.C., Anfinrud P., Fraser J.S., Temperature-Jump Solution X-ray Scattering Reveals Distinct Motions in a Dynamic Enzyme., Submitted - Preprint on BioRxiv., **2018**

St-Jacques A. D., Gagnon O. & Chica R. A., Computational Enzyme Design: Successes, Challenges and Future Directions. In: Williams G & Hall M (Ed.) Modern Biocatalysis: Advances Towards Synthetic Biological Systems, Royal Society of Chemistry, 88-116. **2018**.

Volkmer R., Tapia V., Landgraf C., Synthetic peptide arrays for investigating protein interaction domains, FEBS Letters, 586 (17), **2012**.

Landgraf, C. et al., Protein interaction networks by proteome peptide scanning. PLoS Biology, 2 (1), **2004**.

Frank, R., The SPOT-synthesis technique. Synthetic peptide arrays on membrane supports–principles and applications. J. Immunol. Methods 267, 13–26, **2002**.

Lanouette S., Davey J. A., & al., Discovery of substrates for a SET domain lysine methyltransferase predicted by multistate computational protein design, Structure, 23(1), 206-215, **2015**.

Rathert P., Dhayalan A. & al., Protein lysine methyltransferase G9a acts on non-histone targets, Nature Chemical Biology, 4, 344-346, **2008**.

Young, K. H., Yeast Two-Hybrid: So Many Interactions, (in) So Little Time ..., Biology of Reproduction, 58, 302-311, **1998**.

Kudithipudi S., Identifying Novel Substrates by Specificity Profile Analysis of Protein Lysine Methyltransferases, Doctoral Thesis, Jacobs University, 106p, **2012**.

Kudithipudi S., Dhayalan A., Kebede A. F., & Jeltsch A., The SET8 H4K20 protein lysine methyltransferase has a long recognition sequence covering seven amino acid residues. Biochimie, 94(11), 2212–2218, **2012**.

Reineke U., Sabat R., Antibody epitope mapping using SPOT peptide arrays. Methods Molecular Biology, 524, 145-67, **2009**.

Wu C., Li S. S., CelluSpots: a reproducible means of making peptide arrays for the determination of SH2 domain binding specificity. Methods Molecular Biology, 570, 197-202, **2009**.

Zhang Y., Jurkowska R., Soeroes S., Rajavelu A., Dhayalan A., Bock I., Rathert P., Brandt O., Reinhardt R., Fischle W., Jeltsch A., Chromatin methylation activity of Dnmt3a and Dnmt3a/3L is guided by interaction of the ADD domain with the histone H3 tail, Nucleic Acids Research, 38(13):4246-53, **2010**.

Mayo S. L., Olafson B. D. and Goddard W. A., Dreiding - a generic force-field for molecular simulations. Journal of Physical Chemistry, 94(26):8897-8909, **1990**.

Gasteiger J., Marsili M., Iterative partial equalization of orbital electronegativity - a rapid access to atomic charges. Tetrahedron, 36(22):3219-3228, **1980**.

Dahiyat B. I., Mayo S. L., Probing the role of packing specificity in protein design, Proceedings of the National Academy of Sciences of the United States of America, 94(19):10172-10177, **1997**.

Boas F.E., Harbury P.B., Potential energy functions for protein design. Current Opinion in Structural Biology, 17(2):199-204, **2007**.

Archer D. G., Wang P., The Dielectric Constant of Water and Debye-Hückel Limiting Law Slopes Journal of Physical and Chemical Reference Data, 19, 371, **1990**.

Dunbrack R. L. & Cohen F. E., Bayesian statistical analysis of protein side-chain rotamer preferences, Protein Science, 6(8):1661-1681, **1997**.

Allen B. D. & Mayo S. L., Dramatic performance enhancements for the faster optimization algorithm, Journal of Computational Chemistry, 27(10):1071-1075, **2006**.

Metropolis N., Rosenbluth A. W., Rosenbluth M. N., Teller A. H. and Teller E., Equation of state calculations by fast computing machines. Journal of Chemical Physics, 21(6):1087-1092, **1953**.

Kirkpatrick S., Gelatt C. D. and Vecchi M. P., Optimization by simulated annealing. Science, 220(4598):671-680, **1983**.

Choi E. J., Guntas G. and Kuhlman B., Future challenges of computational protein design, "Protein engineering and design", Park SJ and Cochran JR. Boca Raton, Florida, United States of America, 18:367-385, **2009**.

Davis I.W., Arendall III W. B., Richardson D. C., Richardson J. S., The Backrub Motion: How Protein Backbone Shrugs When a Sidechain Dances, Structure, 14 (2):265-274, **2006**.

Davey J. A. & Chica R. A., Improving the accuracy of protein stability predictions with multistate design using a variety of backbone ensembles. Proteins: Structure, Function, and Bioinformatics, 82(5):771-784, **2014**.

KanungoT., Mount D. M., Netanyahu N. S., Piatko C. D., Silverman R. and Wu A. Y., An efficient k-means clustering algorithm: Analysis and implementation, IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(7):881-892, **2002**.

Bergamin E., Sarvan S., Malette J., Eram M. S., Yeung S., Mongeon V., Joshi M., Brunzelle J. S., Michaels S. D., Blais A., Vedadi M., Couture J-F., Molecular basis for the methylation specificity of ATXR5 for histone H3, Nucleic Acids Research, 45 (11):6375–6387, **2017**.

Seet B. T., Berry D. M., Maltzman J. S., Shabason J., Raina M., Koretzky G. A., McGlade C. J., Pawson T., Efficient T-cell receptor signaling requires a high affinity interaction between the Gads C-SH3domain and the SLP–76 RxxK motif, The EMBO Journal, 26, 678–689, **2007**.

Wiedemann U., Boisguerin P., Leben R., Leitner D., Krause G., Moelling K., Volkmer-Engert R., Oschkinat H., Quantification of PDZ domain specificity, prediction of ligand affinity and rational design of super-binding peptides, Journal of Molecular Biology, 343(3):703-18, **2004**.

Debye P., Interferenz von Röntgenstrahlen und Wärmebewegung, Annalen der Physik, 348 (1):49–92, **1913**.

Sternberg S., Biomedical Image Processing, IEEE Computer, 16 (1):22-34, **1983**

London N., Movshovitz-Attias D. & Schueler-Furman O., The Structural Basis of Peptide-Protein Binding Strategies, Structure 18:188–199, **2010**.

Lee B., Richards F. M., The interpretation of protein structures: estimation of static accessibility, Journal of Molecular Biology, 55 (3): 379–400, **1971**.

Shrake, A., Rupley, J. A., Environment and exposure to solvent of protein atoms. Lysozyme and insulin, Journal of Molecular Biology. 79 (2): 351–71, **1973**.

Chothia C., The nature of the accessible and buried surfaces in proteins, Journal of Molecular Biology,105:1-14, **1976**.

Blundell T. L. & Johnson L. N., Protein Crystallography, Academic Press Inc. London, **1976**.

Chen Y-J., Tsai C-H., Wang P-Y. & Teng S-C., SMYD3 Promotes Homologous Recombination via Regulation of H3K4-mediated Gene Expression, Scientific Reports, 7(3842), **2017**.

Prlić A. & al., BioJava: an open-source framework for bioinformatics, Bioinformatics, 28(20):2693–2695, **2012**.

Steinbeck C. & al. The Chemistry Development Kit (CDK): An Open-Source Java Library for Chemo- and Bioinformatics. J. Chem. Inf. Comput. Sci. 2003 Mar-Apr; 43(2):493-500, **2003**.

Webb B., Šali A., Comparative Protein Structure Modeling Using Modeller, Current Protocols in Bioinformatics 54, John Wiley & Sons, Inc., 5.6.1-5.6.37, **2016**.

Word J. M., Lovell, S.C., Richardson, J. S., & Richardson, D.C., Asparagine and glutamine: using hydrogen atom contacts in the choice of sidechain amide orientation, Journal of Molecular Biology, 285:1735-1747, **1999**.

Yang Q., & Sharp K. A., Atomic Charge Parameters for the Finite Difference Poisson-Boltzmann Method Using Electronegativity Neutralization, Journal of Chemical Theory and Computation, 2:1152-1167, **2006**.

Jmol: an open-source Java viewer for chemical structures in 3D. http://www.jmol.org/

5. Appendix

5.1. VIPER benchmarking: raw data

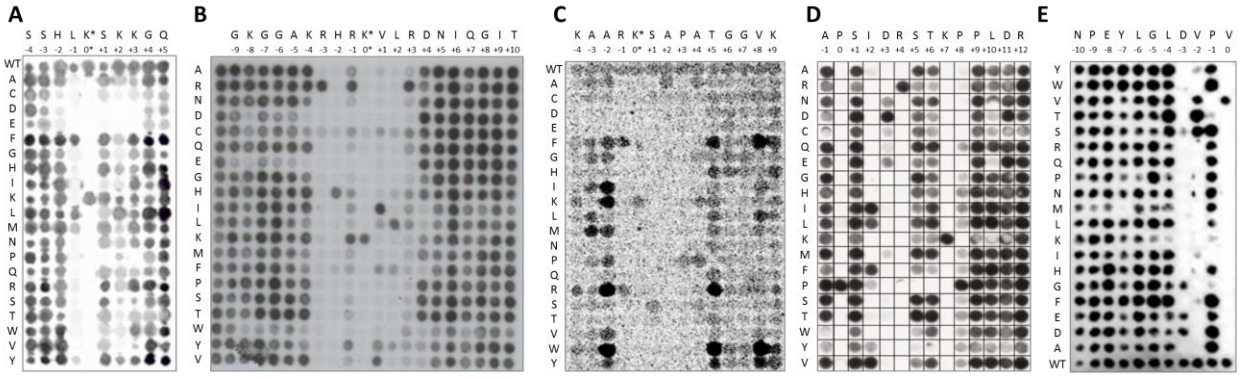


Figure 5.1.1. Raw permutation peptide arrays used for benchmarking VIPER. A) Smyd2 – p53 (PDB ID 3S7F; Lanouette S. & Davey J. A., 2015). B) Set8 – H4K20 (PDB ID 2BQZ; Kudithipudi S. & al. 2012). C) Atxr5 – H3.1K27 (PDB ID 4O30; Bergamin E., 2017). D) Gads – SPL76 (PDB ID 2D0N, Seet B. T. & al., 2007). E) Erbin – ErbB2 (PDB ID 1MFG, Wiedemann U. & al., 2004)

3tg5	Relative Spot Intensity						
Substitution	-2	-1	1	2	3	4	5
A	0.87	0.14	0.90	0.48	0.51	0.97	1.05
C	0.42	0.07	0.42	0.13	0.20	0.44	0.41
D	0.22	0.01	0.01	0.00	0.01	0.39	0.49
E	0.18	0.01	0.04	0.00	0.00	0.31	0.36
F	2.00	1.26	1.26	0.53	1.30	1.85	2.29
G	0.91	0.12	0.68	0.15	0.54	1.00	0.94
H	1.00	0.14	1.14	0.53	0.60	0.97	1.44
I	1.37	0.10	0.25	0.22	0.84	0.72	1.48

K	1.46	0.10	1.08	1.00	1.00	0.78	1.16
L	1.62	1.00	0.50	1.04	0.86	1.43	1.69
M	1.43	0.62	0.97	0.49	0.78	1.04	1.03
N	0.94	0.06	0.67	0.45	0.73	1.06	1.15
P	1.07	0.07	0.01	0.41	0.01	0.40	1.06
Q	0.99	0.05	0.64	0.44	0.46	0.71	1.00
R	0.98	0.15	1.39	0.46	1.12	1.29	1.49
S	0.85	0.08	1.00	0.32	0.51	0.77	0.97
T	0.96	0.08	0.67	0.20	0.54	0.45	0.54
V	1.76	0.10	0.74	0.09	0.52	1.22	1.11
W	1.76	0.07	0.47	0.23	0.97	0.82	0.97
Y	1.72	0.30	1.53	0.95	1.25	1.88	1.86

Table 5.1.1. Experimental peptide array's spot intensities for 3TG5 treated with ImageJ. Their spot relative intensities to wild–type is reported in this table. The data was used to extract recognition profiles of Smyd2 (PDB ID 3TG5) by using a 50% cut-off.

3s7f	Relative Spot Intensity				
Substitution	-1	1	2	3	4
A	0.14	0.90	0.48	0.51	0.97
C	0.07	0.42	0.13	0.20	0.44
D	0.01	0.01	0.00	0.01	0.39
E	0.01	0.04	0.00	0.00	0.31
F	1.26	1.26	0.53	1.30	1.85
G	0.12	0.68	0.15	0.54	1.00
H	0.14	1.14	0.53	0.60	0.97
I	0.10	0.25	0.22	0.84	0.72
K	0.10	1.08	1.00	1.00	0.78
L	1.00	0.50	1.04	0.86	1.43
M	0.62	0.97	0.49	0.78	1.04
N	0.06	0.67	0.45	0.73	1.06
P	0.07	0.01	0.41	0.01	0.40
Q	0.05	0.64	0.44	0.46	0.71
R	0.15	1.39	0.46	1.12	1.29
S	0.08	1.00	0.32	0.51	0.77
T	0.08	0.67	0.20	0.54	0.45
V	0.10	0.74	0.09	0.52	1.22
W	0.07	0.47	0.23	0.97	0.82
Y	0.30	1.53	0.95	1.25	1.88

Table 5.1.2. Experimental peptide array's spot intensities for 3S7F treated with ImageJ. Their spot relative intensities to wild–type is reported in this table. The data was used to extract recognition profiles of Smyd2 (PDB ID 3S7F) by using a 50% cut-off.

2bqz	Relative Spot Intensity							
Substitution	-3	-2	-1	1	2	3	4	5
A	0.12	0.10	0.52	0.22	0.10	0.21	0.70	1.21
C	0.21	0.21	0.40	0.46	0.31	0.45	0.64	0.74
D	0.07	0.08	0.07	0.06	0.06	0.11	1.00	0.93
E	0.06	0.08	0.05	0.11	0.05	0.09	0.71	0.75
F	0.11	0.22	0.29	0.53	0.45	0.17	0.09	0.46
G	0.07	0.10	0.13	0.08	0.05	0.14	0.35	1.00
H	0.09	1.00	0.41	0.17	0.24	0.37	0.54	0.61
I	0.10	0.10	0.40	1.45	0.16	0.35	0.10	0.52
K	0.10	0.10	0.90	0.12	0.10	0.15	0.33	0.52
L	0.07	0.11	0.21	0.41	1.00	0.40	0.12	0.37
M	0.09	0.12	0.27	0.14	0.08	0.18	0.49	0.48
N	0.07	0.08	0.23	0.08	0.06	0.20	0.62	1.00
P	0.08	0.09	0.08	0.10	0.12	0.14	0.76	0.73
Q	0.08	0.09	0.20	0.14	0.08	0.36	0.68	0.94
R	1.00	0.12	1.00	0.11	0.10	1.00	0.50	0.67
S	0.07	0.08	0.21	0.08	0.14	0.13	0.74	0.76
T	0.08	0.11	0.26	0.10	0.07	0.22	0.71	0.92
V	0.07	0.11	0.26	1.00	0.10	0.18	0.15	0.51
W	0.08	0.05	0.10	0.15	0.06	0.29	0.10	0.30
Y	0.09	0.15	0.61	0.79	0.39	0.30	0.24	0.49

Table 5.1.3. Experimental peptide array's spot intensities for 2BQZ treated with ImageJ. Their spot relative intensities to wild–type is reported in this table. The data was used to extract recognition profiles of Set8 (PDB ID 2BQZ) by using a 50% cut-off.

4o30	Relative Spot Intensity												
Substitution	-4	-3	-2	-1	1	2	3	4	5	6	7	8	9
A	0.37	1.00	1.00	0.12	0.00	1.00	0.08	1.00	0.53	0.36	0.42	0.60	0.76
C	0.30	0.61	0.59	0.22	0.51	0.31	0.59	0.36	0.84	0.65	0.77	1.16	1.05
D	0.12	0.21	0.29	0.08	0.09	0.20	0.30	0.04	0.65	0.67	0.47	0.56	1.02
E	0.21	0.18	0.00	0.10	0.03	0.00	0.33	0.20	0.32	0.68	0.27	0.45	0.79
F	0.67	0.97	1.88	1.46	0.00	0.07	0.17	0.34	2.30	0.70	0.96	3.23	1.91
G	0.31	1.26	1.27	0.08	0.10	0.28	0.24	0.73	0.81	1.00	1.00	0.72	0.91
H	0.39	0.55	0.51	0.15	0.11	0.18	0.00	0.29	1.55	1.11	0.87	0.90	1.82
I	0.26	1.26	2.62	0.11	0.12	0.03	0.04	0.23	0.76	0.23	0.20	1.42	0.67
K	1.00	0.28	2.83	0.19	0.00	0.01	0.29	1.04	1.13	0.15	0.33	0.77	1.00
L	0.52	1.42	0.42	0.16	0.00	0.05	0.19	0.19	1.34	0.36	0.32	1.83	1.05
M	0.21	2.53	2.01	0.05	0.00	0.03	0.35	0.13	1.14	0.33	0.38	1.01	0.58
N	0.19	0.34	0.39	0.18	0.13	0.20	0.08	0.44	0.88	0.37	0.45	0.42	0.68
P	0.12	0.86	1.51	0.07	0.01	0.15	1.00	1.48	0.17	0.52	0.34	0.37	0.40

Q	0.15	0.12	0.58	0.11	0.00	0.00	0.13	0.12	1.17	0.36	0.50	0.83	0.59
R	1.20	0.61	3.29	1.00	0.13	0.05	0.10	0.40	2.88	0.66	0.93	1.05	0.94
S	0.42	0.68	0.55	0.14	1.00	0.18	0.19	0.74	0.44	0.29	0.15	0.52	0.75
T	0.29	0.27	0.22	0.10	0.48	0.05	0.24	0.41	1.00	0.32	0.20	0.46	0.77
V	0.46	0.77	1.78	0.13	0.00	0.00	0.14	0.17	0.56	0.15	0.21	1.00	0.50
W	0.90	0.71	3.03	0.12	0.10	0.07	0.17	0.33	3.41	1.10	0.95	3.36	2.02
Y	0.47	0.47	1.34	0.33	0.16	0.12	0.27	0.19	1.52	0.61	0.50	2.21	1.65

Table 5.1.4. Experimental peptide array's spot intensities for 4O30 treated with ImageJ. Their spot relative intensities to wild–type is reported in this table. The data was used to extract recognition profiles of Atxr5 (PDB ID 4O30) by using a 50% cut-off.

2d0n	Relative Spot Intensity								
Substitution	0	1	2	3	4	5	6	7	8
A	0.01	0.84	0.08	0.01	0.02	0.81	0.55	0.02	0.07
C	0.00	0.72	0.00	0.19	0.02	0.63	0.29	0.01	0.03
D	0.00	0.92	0.01	1.00	0.02	0.61	0.27	0.01	0.02
E	0.00	1.06	0.00	0.48	0.03	0.48	0.42	0.02	0.03
F	0.02	0.84	0.87	0.02	0.02	0.43	0.19	0.02	0.22
G	0.01	0.86	0.01	0.00	0.02	0.99	0.53	0.02	0.17
H	0.03	0.88	0.01	0.02	0.03	0.78	0.49	0.03	0.42
I	0.02	0.90	1.00	0.04	0.02	0.43	0.69	0.02	0.38
K	0.01	0.59	0.00	0.01	0.01	0.32	0.35	1.00	0.03
L	0.08	1.07	0.74	0.02	0.01	0.74	0.80	0.03	0.30
M	0.02	1.13	0.01	0.14	0.02	0.99	0.81	0.02	0.24
N	0.03	0.88	0.00	0.45	0.03	0.74	0.57	0.02	0.04
P	1.00	1.12	0.24	0.01	0.07	0.01	0.00	0.03	1.00
Q	0.02	0.97	0.01	0.12	0.02	0.88	0.73	0.02	0.28
R	0.01	0.45	0.00	0.01	1.00	0.39	0.20	0.02	0.07
S	0.03	1.00	0.01	0.03	0.02	1.00	0.78	0.02	0.20
T	0.04	1.11	0.05	0.06	0.02	1.20	1.00	0.03	0.36
V	0.05	1.10	1.08	0.04	0.01	0.46	0.67	0.02	0.29
W	0.00	0.55	0.00	0.00	0.01	0.24	0.59	0.02	0.24
Y	0.01	0.55	0.35	0.01	0.01	0.07	0.10	0.01	0.29

Table 5.1.5. Experimental peptide array's spot intensities for 2D0N treated with ImageJ. Their spot relative intensities to wild–type is reported in this table. The data was used to extract recognition profiles of Gads (PDB ID 2D0N) by using a 50% cut-off.

1mfg	Relative Spot Intensity								
Substitution	-8	-7	-6	-5	-4	-3	-2	-1	0
A	0.93	0.37	0.61	0.65	0.60	0.00	0.05	1.20	0.05
C	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
D	1.15	0.42	0.87	0.49	0.37	1.00	0.04	1.48	0.10
E	1.00	0.32	1.08	0.52	0.58	0.70	0.04	0.89	0.01

F	0.98	0.82	1.17	1.19	1.45	0.01	0.21	1.58	0.09
G	1.49	0.41	1.03	1.00	0.52	0.83	0.06	0.14	0.00
H	1.56	0.50	1.09	0.90	1.06	0.00	0.10	0.81	0.00
I	0.67	0.35	0.73	0.70	1.14	0.02	0.19	0.00	0.00
K	1.05	0.35	0.60	0.23	0.09	0.01	0.03	0.14	0.00
L	1.18	0.60	1.00	0.96	1.00	0.04	0.02	0.45	0.00
M	0.46	0.03	0.69	0.37	0.49	0.02	0.01	0.80	0.04
N	1.03	0.48	1.01	0.52	0.78	0.04	0.26	1.07	0.00
P	0.90	0.28	0.26	1.12	0.39	0.02	0.14	1.00	0.00
Q	1.00	0.54	0.98	0.74	1.03	0.03	0.02	0.99	0.00
R	1.20	0.59	0.88	0.93	0.51	0.02	0.03	0.79	0.00
S	1.09	0.48	0.56	0.42	1.21	0.40	1.60	1.40	0.00
T	1.01	0.40	0.91	0.65	1.86	0.17	1.99	0.19	0.00
V	1.00	0.33	0.87	0.92	1.20	0.01	1.00	0.01	1.00
W	1.29	1.07	1.16	1.07	1.01	0.09	0.02	1.46	0.00
Y	1.30	1.00	1.22	1.00	1.50	0.02	0.23	0.86	0.00

Table 5.1.6. Experimental peptide array's spot intensities for 1MFG treated with ImageJ. Their spot relative intensities to wild–type is reported in this table. The data was used to extract recognition profiles of Erbin (PDB ID 1MFG) by using a 50% cut-off.

5.2. VIPER cut-off optimization

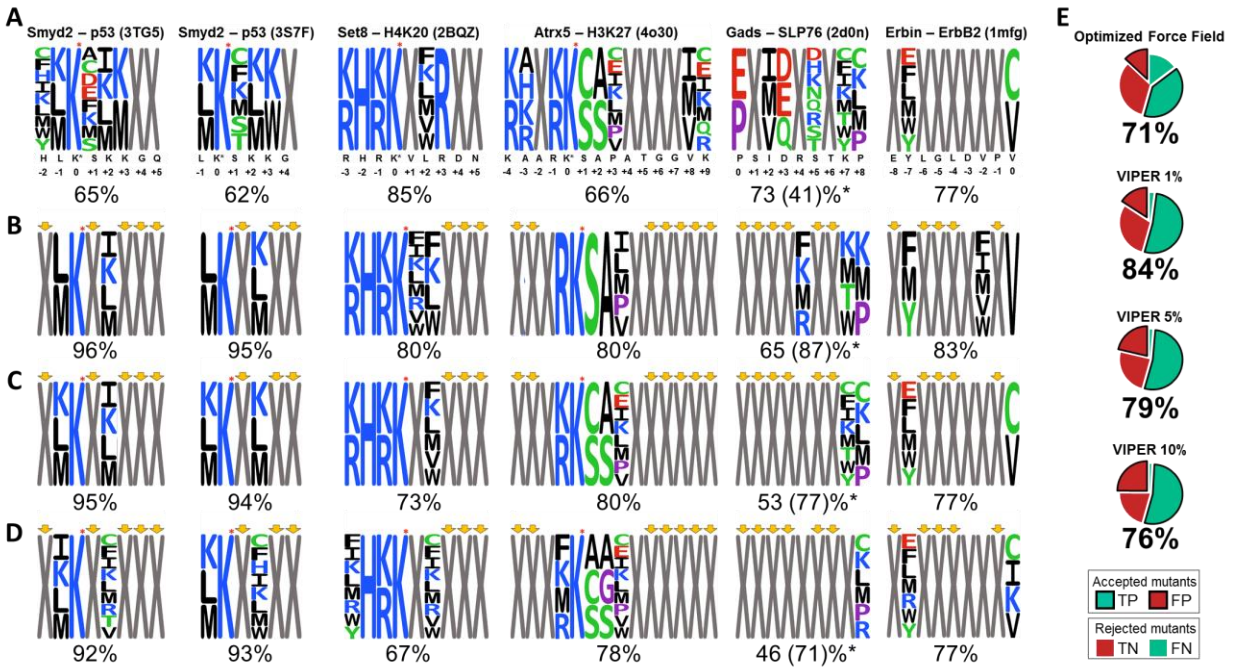


Figure 5.2.1. VIPER cut-off optimization. Predicting tolerant positions greatly improves the accuracy. The cut-off was optimized to yield the final version and optimized version of the algorithm. A) Pre-VIPER optimization stage. B) Predicting tolerant positions (VIPER) at a 1% cut-off. This yields the best results. C) VIPER at 5% cut-off. D) VIPER at 10% cut-off. E) Binning statistics for each method.

5.3. Initial Calculated PHOENIX Energies

3tg5	Energies per peptide positions (kcal/mol), centered at K370						
Substitution	-2	-1	1	2	3	4	5
A	-6.474	-21.202	-50.771	-57.465	-35.471	-9.609	-30.141
C	-10.376	-27.02	-55.794	-63.265	-39.223	-13.193	-33.748
D	-7.678	-16.608	-51.627	-54.108	-33.081	-6.41	-23.594
E	-8.126	-21.885	-49.946	-57.855	-34.343	-10.637	-30.652
F	-14.83	-21.277	-53.328	-59.01	-33.859	-16.576	-27.793
G	-6.543	-17.538	-47.903	-53.407	-33.308	-11.17	-26.292
H	-10.006	-23.862	-50.914	-60.51	-30.881	-12.406	-24.79
I	-11.576	-24.774	-46.869	-66.079	-33.536	-10.685	-34.579
K	-11.208	-31.439	-54.599	-65.632	-40.44	-15.244	-30.792
L	-12.168	-33.211	-42.029	-67.582	-36.887	-18.404	-31.13
M	-13.642	-33.577	-57.709	-69.985	-44.828	-17.228	-34.588
N	-7.424	-14.738	-48.919	-50.478	-32.133	-8.88	-21.281
P	-8.345	-27.441	-19.599	-60.796	4.517	N/E	-33.086
Q	-8.554	-21.276	-46.515	-56.797	-32.666	-9.484	-29.33
R	-7.623	-26.091	-48.299	-59.213	-37.992	-8.424	-27.268
S	-7.84	-17.133	-52.545	-54.018	-35.307	-9.64	-28.991
T	-7.625	-20.357	-47.155	-57.615	-27.632	-10.565	-32.28
V	-8.011	-24.596	-46.584	-61.05	-33.219	-13.183	-33.483
W	-12.01	8.02	-52.869	-46.777	-29.89	-16.454	-19.803
Y	-9.794	-3.745	-49.286	-48.556	-26.684	-13.807	-26.135

Table 5.3.1. Raw calculated energies using parameters from Lanouette S & Davey J. A. (2015) for Smyd2 – p53 (PDB ID 3TG5).

3s7f	Energies per peptide positions (kcal/mol), centered at K370				
Substitution	-1	1	2	3	4
A	-62.39	-84.319	-65.329	-66.368	-65.802
C	-72.63	-86.541	-75.799	-78.259	-79.549
D	-64.146	-64.495	-52.538	-78.714	-63.46
E	-69.103	-72.444	-58.719	-71.044	-75.294
F	-92.885	-79.088	-61.627	-87.668	-88.329
G	-59.271	-73.023	-51.821	-66.749	-70.415
H	-79.145	-63.851	-72.254	-82.528	-69.765
I	-79.589	-16.605	-80.294	-84.945	-86.251
K	-91.293	-86.263	-84.674	-86.029	-80.466
L	-88.033	-75.055	-82.419	-85.062	-82.802
M	-90.029	-86.401	-82.688	-86.659	-86.803
N	-65.459	-65.052	-52.967	-66.885	-58.996
P	N/E	N/E	N/E	N/E	N/E
Q	-68.757	-75.15	-60.728	-78.916	-72.314
R	-71.981	-75.934	-36.089	-79.699	-76.488
S	-65.188	-81.272	-55.109	-71.861	-64.824
T	-75.2	-73.804	-61.75	-75.084	-70.494
V	-74.471	-51.396	-75.637	-74.064	-79.822
W	-78.562	-69.126	62.68	-86.162	-71.173
Y	-77.503	-76.631	140.521	-80.215	-71.398

Table 5.3.2. Raw calculated energies using parameters from Lanouette S & Davey J. A. (2015) for Smyd2 – p53 (PDB ID 3S7F).

2bqz	Energies per peptide positions (kcal/mol), centered at K20							
Substitution	-3	-2	-1	1	2	3	4	5
A	-0.821	-28.774	-15.918	-16.645	-37.263	3.558	9.127	-14.258
C	-4.355	-34.336	-20.023	-21.153	-46.88	0.342	6.729	-22.345
D	1.355	-21.705	-11.562	-15.734	-34.786	2.289	7.627	-13.193
E	-4.004	-26.914	-16.094	-23.246	-39.551	-0.067	5.302	-18.072
F	-8.057	-26.559	-14.595	-24.32	-54.315	-3.915	4.006	-21.947
G	0.849	-19.796	-11.612	-11.161	-30.815	4.993	8.722	-12.702
H	-4.175	-40.435	-14.5	-19.49	-40.975	1.631	6.181	-19.846
I	-7.261	-17.57	-19.94	-23.373	-44.758	-1.695	4.394	-23.576
K	-9.297	-24.907	-23.209	-24.857	-45.913	-4.339	3.47	-23.274
L	-6.838	-37.731	-18.424	-23.092	-50.276	-1.702	3.319	-21.895
M	-9.214	-37.434	-20.452	-23.363	-51.323	-4.109	2.489	-27.217
N	-1.172	-23.155	-13.806	-16.611	-33.849	2.862	6.587	-14.6
P	-3.16	-23.598	-13.547	29.399	N/E	2.888	7.353	4.881
Q	-3.855	-24.478	-17.574	-23.518	-38.12	-0.724	3.668	-18.096
R	-7.615	-14.136	-19.466	-22.218	-40.068	-2.954	3.314	-22.275
S	-1.608	-29.974	-14.046	-17.062	-37.282	3.555	7.051	-17.652
T	-3.892	-34.763	-13.954	-16.094	-37.203	1.734	5.431	-23.428
V	-4.334	-27.463	-19.464	-22.12	-46.171	0.889	6.504	-22.964
W	-7.069	3.363	-11.474	-25.358	-53.645	-5.398	3.782	-21.734
Y	-8.31	-8.292	-15.63	-20.742	-46.126	-0.09	6.753	-17.453

Table 5.3.3. Raw calculated energies using parameters from Lanouette S & Davey J. A. (2015) for Set8 – H4K20 (PDB ID 2BQZ).

4o30	Energies per peptide positions (kcal/mol), centered at K27												
Substitution	-4	-3	-2	-1	1	2	3	4	5	6	7	8	9
A	11.799	-5.898	-15.134	-54.485	-38.98	-60.81	-14.82	-30.771	-0.396	7.996	-6.338	-29.816	-7.82
C	10.158	-5.179	-18.461	-63.59	-42.111	-50.6	-19.634	-30.642	-2.884	4.576	-10.003	-35.491	-13.273
D	9.682	3.262	-13.376	-53.75	-25.621	-37.893	-12.759	-30.518	-2.103	4.628	-9.018	-25.313	-8.775
E	9.11	-6.547	-16.967	-56.919	-19.859	-32.226	-19.618	-29.249	-3.88	4.977	-10.772	-29.516	-11.371
F	8.546	-5.688	-25.27	-70.333	-36.948	N/E	-15.606	-28.266	-1.653	8.165	-10.955	-10.272	-5.838
G	10.488	-0.643	-9.042	-50.579	-32.524	-56.326	-10.857	-28.216	1.203	1.713	-8.74	-24.705	-3.374
H	8.62	-5.792	-17.653	-63.217	-32.136	0.744	-11.929	-27.002	-1.666	6.991	-11.147	-21.156	-7.49
I	9.984	-0.329	-23.672	-59.465	24.827	10.288	-23.806	-26.401	-4.978	10.204	1.346	-42.591	-8.005
K	6.105	-6.586	-23.772	-65.274	-11.706	-18.341	-19.23	-26.015	-5.146	3.16	-10.71	-35.048	-12.789
L	9.134	-3.887	-24.71	-69.049	N/E	-1.554	-21.095	-25.686	-4.777	6.279	-9.835	-33.928	-8.377
M	7.445	-10.436	-24.174	-69.36	-24.215	-35.417	-22.801	-25.624	-4.663	4.674	-11.081	-42.068	-11.173
N	9.029	2.426	-14.478	-53.361	-19.234	-35.388	-12.515	-25.523	-1.385	4.443	-8.489	-25.086	-8.178
P	7.221	N/E	-20	N/E	12.372	-30.785	-19.882	-25.262	1.81	N/E	N/E	-36.237	N/E
Q	8.518	-0.902	-17.812	-55.617	-16.024	-31.014	-19.273	-24.968	-4.006	4.906	-10.673	-30.846	-10.031
R	6.226	-3.968	-20.621	-61.711	10.79	20.374	-18.321	-24.534	-5.385	5.119	-12.265	-28.938	-11.976
S	9.667	-2.294	-13.416	-54.33	-44.291	-58.244	-12.332	-23.716	-1.795	2.287	-12.024	-25.115	-9.043
T	8.923	6.673	-15.01	-61.978	-13.895	-31.606	-16.921	-23.69	-3.497	3.57	-4.013	-31.073	-7.388
V	11.491	1.594	-20.652	-53.477	30.713	7.609	-20.22	-23.411	-4.395	10.673	1.29	-38.994	-7.521
W	6.778	-2.344	-26.702	-63.352	9.153	N/E	-17.686	-23.157	-2.405	5.71	-15.454	-28.362	-7.034
Y	8.853	-2.058	-21.709	-56.402	-27.885	N/E	-16.169	13.04	-3.141	6.795	-10.146	-8.196	-7.999

Table 5.3.4. Raw calculated energies using parameters from Lanouette S & Davey J. A. (2015) for Atxr5 – H3.1K27 (PDB ID 4O30).

2d0n	Energies per peptide positions (kcal/mol), starting at P233								
Substitution	0	1	2	3	4	5	6	7	8
A	4.94	-7.844	-46.799	-1.899	-37.821	2.847	-18.036	-31.852	-42.937
C	3.12	-8.255	-51.783	-5.392	-43.623	1.779	-23.469	-40.751	-47.009
D	4.185	-10.072	-47.044	-8.713	-39.541	-1.542	-20.794	-31.492	-37.373
E	-2.19	-10.064	-49.163	-5.937	-42.26	1.479	-24.482	-31.706	-40.944
F	3.213	-8.398	-49.728	-7.283	-48.166	3.205	-21.64	-41.4	-45.476
G	6.099	-9.182	-44.408	-2.192	-35.27	2.587	-14.373	-25.836	-38.859
H	4.79	-9.134	-46.944	-3.676	-43.725	2.572	-21.827	-34.667	-38.449
I	5.005	-8.636	-57.075	-5.066	-43.056	5.009	-23.81	-41.258	-44.63
K	1.602	-10.794	-52.918	-6.242	-45.201	-0.295	-24.839	-39.783	-46.084
L	1.353	-9.334	-54.035	-5.918	-41.894	5.244	-22.032	-35.759	-49.305
M	1.24	-11.785	-55.102	-8.599	-44.403	2.304	-22.515	-43.048	-50.837
N	3.131	-11.649	-46.297	-8.78	-39.614	-1.413	-21.309	-34.371	-36.161
P	-0.314	-9.662	-47.182	-5.125	-40.721	5.382	-23.35	N/E	-49.096
Q	-0.004	-11.955	-49.296	-8.394	-41.633	0.267	-24.661	-32.152	-39.213
R	-1.924	-11.613	-50.828	-6.424	-45.473	-3.856	-24.318	-39.036	-43.254
S	3.627	-9.933	-47.119	-3.471	-38.684	0.44	-20.996	-35.168	-38.47
T	6.909	-9.375	-51.259	-3.982	-43.738	1.405	-22.355	-42.226	-39.188
V	5.605	-7.215	-54.31	-3.174	-42.373	3.967	-22.102	-40.94	-41.501
W	3.762	-9.665	-54.146	-4.975	-48.578	6.877	-19.244	-44.179	-39.9
Y	3.449	-9.349	-48.552	-6.516	-45.143	1.392	-21.266	-42.092	-39.098

Table 5.3.5. Raw calculated energies using parameters from Lanouette S & Davey J. A. (2015) for Gads – SLP76 (PDB ID 2D0N).

1mfg	Calculated Energies (kcal / mol)								
Substitution	-8	-7	-6	-5	-4	-3	-2	-1	0
A	-5.996	3.791	-0.208	-6.522	8.228	-22.076	-13.69	-16.766	-40.504
C	-6.905	-3.018	-2.466	-8.286	4.292	-24.451	-16.406	-19.315	-46.442
D	-10.262	8.62	-1.457	-8.922	7.349	-17.407	-13.641	-18.077	-36.418
E	-8.781	0.133	-1.917	-9.559	2.512	-19.762	-15.835	-20.405	-35.928
F	-6.998	-13.933	-3.683	-11.004	4.003	-22.723	-19.533	-22.387	-0.53
G	-6.571	8.926	0.488	-8.86	10.003	-18.541	-9.479	-15.197	-33.843
H	-7.307	-0.355	-1.588	-9.161	5.968	-23.071	-16.878	-18.727	-37.776
I	-6.744	-1.965	-3.74	-7.952	-0.728	-5.249	-19.415	-22.976	-45.417
K	-9.674	-0.762	-3.955	-12.073	1.651	-27.51	-17.681	-21.293	-42.512
L	-6.911	-6.123	-3.517	-11.212	-0.783	-24.42	-17.467	-20.969	-43.944
M	-7.716	-8.271	-6.207	-11.374	-2.061	-24.458	-20.135	-25.022	-43.272
N	-9.967	8.157	-2.461	-9.09	6.222	-16.36	-13.078	-18.073	-32.94
P	-6.159	-1.156	2.11	N/E	6.027	-20.445	-13.259	-19.503	-6.936
Q	-8.303	2.266	-2.915	-9.49	3.142	-20.178	-17.496	-19.54	-35.28
R	-9.395	-1.412	-5.73	-11.039	1.231	-24.765	-17.6	-19.633	-36.463
S	-7.176	5.981	-1.322	-8.236	5.712	-22.592	-13.681	-16.638	-34.766
T	-6.434	1.967	-2.963	-8.733	3.952	-20.173	-14.629	-18.559	-40.754
V	-6.326	0.418	-2.664	-7.819	2.69	-7.701	-17.557	-20.847	-49.551
W	-7.399	-8.557	-1.448	-10.379	1.475	-22.507	-23.834	-26.435	N/E
Y	-7.646	-6.579	-2.875	-8.647	5.26	-23.881	-19.03	-20.466	N/E

Table 5.3.6. Raw calculated energies from the using parameters from Lanouette S & Davey J. A. (2015) for Erbin – ErbB2 (PDB ID 1MFG).

5.4. VIPER Calculated PHOENIX Energies

3tg5	Energies per peptide positions (kcal/mol), centered at K370						
Substitution	-2	-1	1	2	3	4	5
A	-11.064	-28.483	-70.861	-66.547	-47.88	-20.675	-38.869
C	-15.482	-33.733	-75.128	-71.747	-51.692	-23.186	-42.383
D	-12.105	-24.901	-72.109	-63.337	-44.557	-16.036	-34.754
E	-12.353	-29.097	-70.369	-68.023	-44.551	-20.978	-42.567
F	-18.798	-25.752	-70.245	-70.028	-41.989	-26.011	-35.994
G	-11.048	-26.973	-67.83	-55.922	-45.155	-20.869	-35.268
H	-14.871	-34.104	-69.128	-67.321	-41.197	-22.81	-36.481
I	-15.472	-36.917	-60.268	-76.582	-44.859	-22.366	-43.156
K	-15.772	-39.224	-74.459	-76.35	-55.09	-25.735	-38.601
L	-15.914	-39.878	-50.644	-77.032	-48.424	-26.982	-40.348
M	-17.396	-40.059	-76.899	-79.368	-57.249	-26.009	-43.109
N	-12.648	-22.889	-68.677	-63.209	-44.847	-19.773	-30.437
P	-12.037	-34.788	-45.7	-67.088	-12.539	N/E	-41.823
Q	-13.717	-28.553	-68.229	-66.196	-47.02	-19.324	-39.26
R	-12.876	-35.681	-68.429	-70.303	-50.649	-18.711	-35.99
S	-12.656	-26.201	-73.006	-63.704	-44.623	-22.004	-38.172
T	-11.628	-26.8	-67.925	-69.766	-41.982	-20.7	-41.619
V	-12.291	-32.623	-65.649	-72.363	-45.269	-23.758	-42.038
W	-16.107	4.07	-67.899	-56.881	-38.221	-25.508	-28.591
Y	-15.599	-9.534	-66.611	-58.98	-33.499	-23.324	-35.31

Table 5.4.1. Raw calculated energies from the VIPER method for Smyd2 – p53 (PDB ID 3TG5).

3s7f	Energies per peptide positions (kcal/mol), centered at K370				
Substitution	-1	1	2	3	4
A	-17.071	-31.854	-39.577	-21.015	-21.872
C	-19.533	-35.414	-44.49	-34.383	-24.984
D	-9.226	-29.356	-32.099	-29.448	-24.867
E	-15.894	-26.294	-38.732	-26.593	-31.069
F	-19.589	-36.644	-45.307	-36.833	-26.805
G	-12.545	-28.181	-31.608	-19.229	-19.649
H	-20.584	-29.351	-45.024	-35.969	-26.202
I	-18.764	-24.171	-44.146	-34.845	-29.616
K	-23.471	-34.639	-48.409	-39.859	-26.749
L	-24.03	-30.929	-48.538	-29.397	-29.579
M	-23.954	-35.697	-49.712	-36.448	-29.914
N	-9.312	-29.351	-33.808	-24.927	-21.043
P	-20.224	N/E	-40.121	-15.569	-24.143
Q	-15.19	-28.363	-35.817	-33.687	-21.514
R	-20.099	-29.146	-40.282	-35.691	-28.455
S	-14.717	-34.496	-36.225	-31.412	-21.419
T	-17.281	-36.181	-41.405	-35.054	-25.433
V	-15.968	-24.996	-42.307	-31.567	-27.092
W	-16.259	-31.866	-44.249	-39.708	-28.029
Y	-8.666	-28.625	-27.28	-32.802	-30.573

Table 5.4.2. Raw calculated energies from the VIPER method for Smyd2 – p53 (PDB ID 3S7F).

2bqz	Energies per peptide positions (kcal/mol), centered at K20							
Substitution	-3	-2	-1	1	2	3	4	5
A	-46.96	-43.039	-23.357	-43.89	-65.156	-22.189	-22.095	-49.902
C	-51.281	-52.16	-25.005	-50.39	-72.004	-23.983	-23.256	-54.322
D	-40.453	-35.283	-13.464	-38.511	-55.946	-17.049	-22.009	-41.193
E	-44.097	-38.746	-19.391	-48.138	-59.987	-25.338	-23.066	-42.224
F	-53.938	-41.762	-14.881	-53.892	-77.819	-26.887	-24.821	-55.539
G	-45.185	-41.084	-15.862	-36.873	-48.821	-19.256	-18.407	-44.037
H	-50.728	-67.448	-20.532	-48.381	-68.553	-27.731	-27.121	-51.305
I	-53.693	-31.841	-23.928	-53.751	-69.765	-25.664	-24.434	-56.069
K	-60.33	-54.54	-37.865	-55.909	-76.528	-36.107	-31.603	-59.687
L	-53.1	-53.964	-20.311	-52.195	-76.512	-25.95	-24.109	-51.807
M	-55.136	-54.909	-23.865	-53.495	-75.464	-26.205	-24.972	-58.255
N	-46.141	-43.515	-22.854	-42.284	-60.782	-25.266	-26.973	-48.194
P	-52.05	-44.446	-15.879	-0.877	N/E	-21.86	-26.099	-21.053
Q	-49.696	-45.795	-28.047	-49.894	-64.653	-26.262	-28.712	-49.125
R	-58.139	-38.734	-35.582	-52.199	-69.334	-39.611	-33.996	-58.258
S	-47.296	-53.298	-21.961	-43.446	-61.969	-22.834	-22.684	-50.004
T	-47.867	-51.94	-22.362	-44.063	-65.63	-23.688	-23.083	-52.434
V	-51.245	-40.152	-23.619	-51.929	-73.242	-24.406	-23.021	-53.458
W	-52.541	-9.869	-11.65	-54.178	-77.18	-32.235	-26.865	-51.652
Y	-53.602	-19.739	-17.414	-49.366	-67.675	-28.948	-22.928	-49.681

Table 5.4.3. Raw calculated energies from the VIPER method for Set8 – H4K20 (PDB ID 2BQZ).

4o30	Energies per peptide positions (kcal/mol), centered at K27												
Substitution	-4	-3	-2	-1	1	2	3	4	5	6	7	8	9
A	-21.057	-51.307	-50.952	-70.566	-72.069	-74.469	-29.762	-38.206	-8.89	-2.104	-9.68	-30.454	-17.605
C	-22.336	-46.478	-55.663	-81.037	-75.174	-62.517	-35.984	-42.529	-11.395	-5.146	-17.287	-36.75	-23.811
D	-19.981	-41.603	-49.552	-72.262	-64.122	-51.539	-30.546	-39.235	-10.142	-1.711	-13.95	-26.639	-18.437
E	-23.755	-47.921	-54.704	-73.43	-55.259	-42.931	-35.126	-41.232	-12.186	-3.082	-16.618	-33.926	-23.536
F	-24.971	-44.819	-59.614	-83.267	-67.539	N/E	-31.539	-35.834	-11.777	-1.571	-12.735	-10.889	-18.582
G	-19.109	-42.934	-42.603	-64.31	-65.154	-68.383	-27.202	-36.491	-7.298	-2.139	-10.483	-24.453	-12.999
H	-26.243	-49.985	-54.217	-77.209	-67.702	-7.768	-25.168	-40.784	-10.459	-1.303	-14.22	-20.413	-16.727
I	-23.497	-34.266	-60.911	-72.258	-3.894	-4.059	-40.526	-35.668	-14.242	-3.083	-9.191	-43.476	-20.767
K	-32.483	-50.885	-60.666	-89.175	-47.679	-25.029	-35.891	-42.524	-14.582	-2.955	-13.436	-36.679	-21.395
L	-24.03	-39.404	-62.979	-80.223	N/E	1.28	-37.122	-40.892	-13.186	-2.045	-11.146	-34.815	-19.721
M	-25.09	-47.212	-57.697	-81.94	-58.659	-47.083	-36.991	-41.831	-13.364	-5.132	-13.096	-42.783	-22.114
N	-23.021	-46.207	-51.523	-72.71	-55.867	-44.684	-25.749	-41.937	-10.181	-3.815	-13.609	-30.048	-15.863
P	-23.118	N/E	-61.989	N/E	-27.202	-39.175	-36.689	-41.424	-7.891	N/E	N/E	-36.927	36.62
Q	-23.841	-44.488	-55.58	-74.359	-50.369	-39.351	-31.161	-39.676	-12.676	-3.264	-15.513	-32.958	-20.667
R	-35.449	-53.241	-61.096	-90.091	-31.274	6.939	-29.527	-42.694	-14.237	-1.783	-15.08	-33.839	-21.703
S	-21.094	-47.262	-49.884	-74.761	-78.506	-72.34	-26.054	-36.995	-10.1	-2.74	-17.127	-28.122	-17.631
T	-21.82	-35.313	-54.536	-76.342	-51.439	-42.319	-31.558	-38.597	-12.158	-2.187	-13.157	-33.564	-19.26
V	-22.592	-39.009	-54.746	-65.099	-1.2	-5.557	-37.091	-32.469	-13.162	-3.309	-9.79	-39.676	-19.598
W	-25.578	-41.099	-63.94	-78.145	-9.287	N/E	-33.812	-0.823	-12.42	-2.043	-19.307	-26.209	-19.208
Y	-24.871	-43.231	-57.161	-70.112	-62.139	N/E	-30.192	-38.242	-12.595	-2.11	-14.112	-5.547	-18.997

Table 5.4.4. Raw calculated energies from the VIPER method for Atxr5 – H3.1K27 (PDB ID 4O30).

2d0n	Energies per peptide positions (kcal/mol), starting at P233								
Substitution	0	1	2	3	4	5	6	7	8
A	-3.264	-14.846	-65.864	-21.159	-68.872	-18.103	-45.848	-42.506	-51.824
C	-5.451	-15.554	-71.889	-27.113	-79.144	-19.081	-49.771	-50.409	-55.919
D	-2.471	-15.3	-63.906	-34.183	-71.86	-21.824	-45.955	-37.733	-44.803
E	-10.627	-16.705	-66.122	-33.216	-73.474	-17.653	-47.761	-39.854	-48.361
F	-4.217	-17.177	-70.164	-25.087	-83.863	-19.95	-50.579	-51.462	-52.657
G	-1.432	-14.453	-61.124	-20.618	-60.829	-17.535	-42.685	-36.166	-47.628
H	-4.381	-16.398	-65.892	-23.986	-74.186	-21.692	-48.652	-45.644	-46.862
I	-3.213	-16.56	-78.116	-23.884	-78.304	-19.275	-49.126	-52.156	-51.804
K	-6.882	-18.393	-70.249	-29.802	-82.004	-23.999	-51.195	-52.699	-58.295
L	-6.53	-16.9	-73.436	-24.044	-78.779	-19.693	-49.38	-49.825	-57.827
M	-6.292	-17.386	-74.756	-25.731	-80.617	-20.199	-50.036	-56.249	-59.963
N	-2.945	-16.72	-61.207	-32.166	-72.663	-24.213	-48.318	-42.22	-46.276
P	-10.286	-17.74	-67.961	-23.453	-70.7	-11.285	-51.873	N/E	-58.525
Q	-7.17	-18.733	-64.897	-34.112	-72.406	-23.196	-50.47	-41.108	-48.497
R	-8.018	-21.528	-67.602	-31.307	-80.711	-32.547	-50.547	-50.005	-55.258
S	-3.613	-15.081	-64.413	-27.975	-72.028	-21.325	-46.565	-47.04	-47.433
T	0.266	-15.608	-68.78	-28.652	-79.031	-21.572	-47.472	-53.681	-49.521
V	-3.261	-15.723	-75.303	-22.843	-74.408	-18.693	-46.335	-49.536	-50.575
W	-4.086	-18.052	-74.145	-27.366	-78.109	-20.151	-47.932	-52.791	-48.935
Y	-5.021	-17.246	-67.454	-27.335	-77.045	-20.038	-48.567	-51.048	-48.512

Table 5.4.5. Raw calculated energies from the VIPER method for Gads – SLP76 (PDB ID 2D0N).

1mfg	Energies per peptide positions (kcal/mol), centered at C-terminal								
Substitution	-8	-7	-6	-5	-4	-3	-2	-1	0
A	-15.537	-13.339	-15.483	-18.154	-3.827	-35.131	-27.255	-21.093	-49.023
C	-16.254	-20.274	-17.477	-18.834	-5.554	-38.866	-30.496	-23.153	-55.242
D	-16.108	-9.775	-16.187	-18.788	-13.913	-29.52	-25.378	-25.241	-50
E	-17.605	-24.619	-20.342	-19.541	-17.227	-29.984	-27.251	-27.483	-47.413
F	-17.584	-31.059	-19.162	-20.022	-6.573	-38.807	-33.881	-24.553	2.03
G	-15.306	-9.132	-14.303	-18.083	-3.171	-33.754	-22.826	-19.284	-43.585
H	-17.344	-20.457	-21.537	-19.695	-8.711	-34.268	-30.282	-23.448	-47.967
I	-16.816	-13.617	-17.842	-19.557	-7.674	-13.67	-33.236	-25.083	-52.993
K	-21.689	-22.169	-22.836	-24.971	-10.763	-38.778	-31.002	-28.965	-52.672
L	-16.852	-25.315	-16.934	-19.689	-7.998	-37.875	-31.163	-23.678	-50.053
M	-17.861	-26.202	-18.163	-20.598	-8.576	-37.505	-33.487	-27.497	-48.682
N	-16.544	-11.473	-20.791	-19.173	-13.083	-28.712	-24.55	-23.056	-45.043
P	-17.204	-17.731	-9.203	N/E	-2.745	-36.169	-24.259	-24.115	-13.61
Q	-19.046	-16.685	-21.294	-22.033	-13.717	-27.543	-29.605	-28.288	-46.257
R	-22.553	-23.342	-27.254	-27.234	-13.963	-31.782	-30.695	-29.551	-49.83
S	-15.604	-15.824	-16.098	-18.398	-7.631	-37.413	-26.049	-22.279	-47.483
T	-16.132	-17.447	-16.303	-18.609	-7.512	-34.487	-27.793	-23.026	-52.099
V	-16.257	-16.2	-17.031	-18.969	-5.857	-17.579	-31.839	-23.651	-58.047
W	-18.041	-24.787	-19.997	-20.962	-7.443	-31.809	-37.543	-26.4	N/E
Y	-17.242	-25.845	-18.336	-20.1	-6.737	-37.507	-31.254	-24.26	N/E

Table 5.4.6. Raw calculated energies from the VIPER method for Erbin – ErbB2 (PDB ID 1MFG).

5.5. Peptide Residues Analysis

3tg5	Peptide Positions						
Attribute	-2	-1	1	2	3	4	5[a]
Exp. Aas	17	3	14	5	17	17	17
Pred. Aas	9	3	8	4	2	15	12
SASA (%)	N/A	4	13	16	6	68	N/A
B–factor (Å^2)	41.3	29.9	26.5	29.2	36.4	42.6	45.5
Contacts	2	9	2	8	8	0	7
WT Boltz E kcal/mol)	-14.871	-39.878	-73.006	-76.35	-55.09	-20.869	-39.26

a. This position contains missing atoms, therefore it is considered dynamic and a B–factor penalty will be applied.

Table 5.5.1. P53 (PDB ID 3TG5) peptide attributes used for predicting tolerant positions. Tolerated amino acid substitutions in the permutation array experiment (Exp. Aas) are compared to predicted tolerated amino acid substitutions (Pred. Aas), solvent accessible surface area (SASA), B–factor and the number of contacts made at the interface (Contacts).

3s7f	Peptide Positions				
Attribute	-1	1	2	3	4[a]
Exp. Aas	3	14	5	17	17
Pred. Aas	3	6	3	2	20
SASA (%)	N/A	12	12	32	N/A
B–factor (Å^2)	69.3	71.1	66.3	89.4	89.4
Contacts	8	2	6	7	1
WT Boltz E kcal/mol)	-24.03	-34.496	-48.409	-39.859	-19.649

a. This position contains missing atoms, therefore it is considered dynamic and a B–factor penalty will be applied.

Table 5.5.2. P53 (PDB ID 3S7F) peptide attributes used for predicting tolerant positions. Tolerated amino acid substitutions in the permutation array experiment (Exp. Aas) are compared to predicted tolerated amino acid substitutions (Pred. Aas), solvent accessible surface area (SASA), B–factor and the number of contacts made at the interface (Contacts).

2bqz	Peptide Positions								
Attribute	-3	-2	-1	1	2	3	4	5	6[a]
Exp. Aas	1	1	4	4	1	1	10	15	
Pred. Aas	2	1	1	11	6	1	19	16	
SASA (%)	N/A	8	32	25	14	52	91	29	N/A
B–factor (Å^2)	16.0	7.2	11.4	7.0	11.0	19.7	28.4	33.4	35.9
Contacts	7	7	6	3	8	2	0	3	0
WT Boltz E kcal/mol)	-58.139	-67.448	-35.582	-51.929	-76.512	-39.611	-22.009	-48.194	N/A

a. This position was mutated for the experimental peptide array, thus was not considered in the benchmarking.

Table 5.5.3. H4K20 (PDB ID 2BQZ) peptide attributes used for predicting tolerant positions. Tolerated amino acid substitutions in the permutation array experiment (Exp. Aas) are compared to predicted tolerated amino acid substitutions (Pred. Aas), solvent accessible surface area (SASA), B–factor and the number of contacts made at the interface (Contacts).

4o30	Peptide Positions												
Attribute	-4[a,b]	-3	-2	-1	1	2	3	4	5	6	7	8	9[a]
Exp. Aas	5	13	15	2	2	1	2	5	17	10	7	16	19
Pred. Aas	2	4	19	2	2	2	8	16	12	15	16	3	7
SASA (%)	N/A	17	21	12	1	0	14	4	54	80	21	12	N/A
B–factor (Å^2)	57.83	50.4	42.8	41.9	34.6	34.6	34.1	36.8	40.3	41.8	44.0	50.2	61.7
Contacts	0	5	4	8	9	5	5	5	3	0	1	7	0
WT Boltz E kcal/mol)	-32.483	-51.307	-50.952	-90.091	-78.506	-74.469	-36.689	-38.206	-12.158	-2.139	-10.483	-39.676	-21.395

a. This position contains missing atoms, therefore it is considered dynamic and a B–factor penalty will be applied.

Table 5.5.4. H3.1K27 (PDB ID 4O30) peptide attributes used for predicting tolerant positions. Tolerated amino acid substitutions in the permutation array experiment (Exp. Aas) are compared to predicted tolerated amino acid substitutions (Pred. Aas), solvent accessible surface area (SASA), B–factor and the number of contacts made at the interface (Contacts).

2d0n	Peptide Positions								
Attribute	0	1	2	3	4	5	6	7	8
Exp. Aas	1	19	4	1	1	11	11	1	1
Pred. Aas	2	20	3	3	10	8	19	8	5
SASA (%)	N/A	80	36	57	29	80	44	15	N/A
B–factor (Å^2)	49.8	46.5	37.6	28.6	22.5	25.3	21.5	19.1	21.6
Contacts	3	0	4	0	7	0	2	6	4
WT Boltz E kcal/mol)	-10.286	-15.081	-78.116	-34.183	-80.711	-21.325	-47.472	-52.699	-58.525

Table 5.5.5. SLP76 (PDB ID 2D0N) peptide attributes used for predicting tolerant positions. Tolerated amino acid substitutions in the permutation array experiment (Exp. Aas) are compared to predicted tolerated amino acid substitutions (Pred. Aas), solvent accessible surface area (SASA), B–factor and the number of contacts made at the interface (Contacts).

1mfg	Peptide Positions								
Attribute	-8	-7	-6	-5	-4	-3	-2	-1	0
Exp. Aas	18	7	18	15	15	3	3	13	1
Pred. Aas	12	6	17	19	11	17	11	17	2
SASA (%)	N/A	22	74	54	86	28	31	57	N/A
B–factor (Å^2)	45.9	12.7	13.3	11.5	10.4	10.0	9.7	11.6	10.4
Contacts	1	6	1	0	0	5	3	1	7
WT Boltz E kcal/mol)	-17.605	-25.845	-16.934	-18.083	-7.998	-29.52	-31.839	-24.115	-58.047

Table 5.5.6. ErbB2 (PDB ID 1MFG) peptide attributes used for predicting tolerant positions. Tolerated amino acid substitutions in the permutation array experiment (Exp. Aas) are compared to predicted tolerated amino acid substitutions (Pred. Aas), solvent accessible surface area (SASA), B–factor and the number of contacts made at the interface (Contacts).

5.7. Smyd3 methylation assays

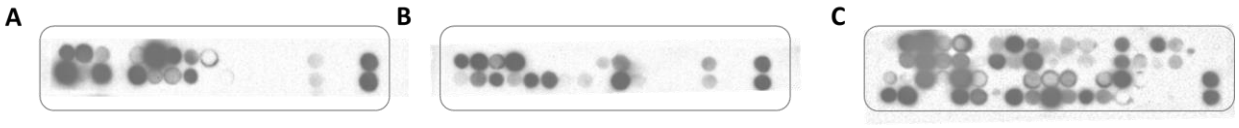


Figure 5.7.1. Peptide arrays for methylation assays of Smyd3 with VEGFR1 and MEKK2 motif screened peptides. Each peptide is incubated with SAM (H3) and Smyd3 for several hours before detection by autoradiography. Various screens were performed based on different predicted recognition profiles. A) Smyd3 - VEGFR1 predicted motif (VIPER), B) Smyd3 – MEKK2 predicted motif (VIPER) and C) combination of both predicted motifs used in A) and B).

5ex3	Calculated Energies (kcal / mol)					
Substitution	-3	-2	-1	1	2	3
A	-1.481	-35.796	-26.422	-27.665	-44.996	-31.516
C	-2.8	-40.677	-22.739	-34.066	-49.813	-31.94
D	-6.664	-34.535	-17.393	-27.241	-43.563	-24.303
E	-5.919	-40.686	-5.519	-30.869	-46.187	-25.056
F	-7.104	-43.995	N/E	-37.194	-26.319	-37.052
G	-2.016	-32.343	-26.665	-22.903	-39.784	-38.681
H	-7.278	-38.304	4.489	-31.117	-43.497	-28.571
I	-4.402	-46.331	N/E	-32.888	-53.278	-25.175
K	-15.636	-42.46	N/E	-32.782	-60.317	-32.556
L	-4.718	-46.75	N/E	-36.043	-53.528	-36.325
M	-5.821	-47.569	3.149	-37.252	-54.347	-36.247
N	-7.869	-32.254	-12.269	-22.867	-41.254	-21.859
P	-3.694	-39.72	N/E	-24.065	-48.725	N/E
Q	-9.229	-36.774	6.983	-31.235	-44.114	-22.307
R	-15.59	-37.096	N/E	-25.597	-51.645	-18.921
S	-1.568	-32.018	-20.613	-28.984	-40.535	-25.362
T	-2.12	-37.849	-7.849	-25.718	-44.812	-24.596
V	-2.939	-42.651	N/E	-30.069	-50.584	-22.642
W	-12.775	-40.425	N/E	-35.682	-34.987	-25.849
Y	-12.196	-38.893	N/E	-31.712	N/E	-31.705

Table 5.7.1. Raw calculated energies from the VIPER method for Smyd3 – VEGFR1 (PDB ID 5EX3).

5hq8	Calculated Energies (kcal / mol)							
Substitution	-4	-3	-2	-1	1	2	3	4
A	-1.555	-17.139	-24.39	-45.993	-25.946	-15.682	-25.186	-63.829
C	-4.832	-19.024	-30.619	-45.426	-29.297	-23.299	-31.92	-68.992
D	-9.38	-16.58	-20.724	-33.218	-20.819	-16.49	-20.704	-62.482
E	-9.142	-20.353	-24.503	-20.449	-21.13	-13.212	-22.687	-64.753
F	-5.567	-24.503	-41.186	N/E	-37.377	-21.51	-29.835	-78.096
G	-2.081	-16.744	-25.055	-44.059	-25.031	-14.653	-19.06	-60.482
H	-3.822	-19.242	-34.653	-30.277	-27.807	-13.742	-31.266	-69.739
I	-2.816	-21.662	-31.261	-10.829	-28.447	-19.409	-30.901	-57.503
K	-4.086	-23.691	-28.515	-9.548	-27.916	-14.084	-28.472	-68.993
L	-2.664	-22.098	-32.071	N/E	-29.557	-19.142	-26.196	-72.622
M	-3.798	-23.336	-34.737	-32.658	-30.688	-23.324	-33.374	-75.717
N	-6.152	-19.309	-23.537	-27.527	-22.14	-14.16	-20.297	-63.805
P	-2.28	-13.424	4.38	N/E	N/E	-12.287	-30.778	-66.156
Q	-3.111	-19.653	-22.508	-19.577	-20.996	-10.863	-20.737	-66.118
R	-11.064	-23.676	-26.488	-15.669	-22.749	-11.217	-28.839	-56.744
S	-5.908	-24.922	-27.125	-47.228	-22.6	-17.651	-31.238	-62.099
T	-6.446	-26.027	-24.17	-48.604	-21.85	-16.828	-35.272	-59.895
V	-2.06	-19.826	-25.263	-36.74	-27.158	-17.319	-25.226	-56.626
W	-6.835	-23.233	-13.966	N/E	-36.924	-21.073	-14.689	-78.102
Y	-5.534	-24.086	-32.156	N/E	-31.472	-15.457	-24.366	-76.253

Table 5.7.2. Raw calculated energies from the VIPER method for Smyd3– MEKK2 (PDB ID 5HQ8).

5ex3	Peptide Positions					
Attribute	-3	-2	-1	1	2	3[a]
SASA (%)	N/A	24	5	14	11	N/A
B–factor (Å^2)	105.9	73.9	74.0	63.7	61.9	76.0
Contacts	2	7	4	3	8	4
WT Boltz E kcal/mol)	-15.636	-46.75	-26.665	-28.984	-53.528	-38.681

a. This position contains missing atoms, therefore it is considered dynamic and a B–factor penalty will be applied.

Table 5.7.3. VEGFR1 (PDB ID 5EX3) peptide attributes used for predicting tolerant positions. Solvent accessible surface area (SASA), B–factor and the number of contacts made at the interface (Contacts) are shown above.

5hq8	Peptide Positions							
Attributes	-4	-3	-2	-1	1	2	3	4[a]
SASA (%)	N/A	65	12	10	32	17	15	N/A
B–factor (Å^2)	69.7	55.7	34.7	29.5	35.0	53.5	55.8	48.6
Contacts	1	2	7	4	2	2	4	10
WT Boltz E kcal/mol)	-9.142	-23.691	-41.186	-44.059	-25.031	-14.653	-35.272	-76.253

a. This position contains missing atoms, therefore it is considered dynamic and a B–factor penalty will be applied.

Table 5.7.4. VEGFR1 (PDB ID 5EX3) peptide attributes used for predicting tolerant positions. Solvent accessible surface area (SASA), B–factor and the number of contacts made at the interface (Contacts) are shown above.

UniProt ID	Peptide sequence	Spot intensity	Relative to WT	Efficient methylation (> 0.5)
Q9Y6A5	VPPKNLAKAMKVTFQAA	22566	6.17	yes
Q8BMF4	VFVSPLAKKLAAEKGAA	19642	7.55	yes
Q6ZV50	TYLSNMAKTMRMVLKAA	14036	6.68	yes
P17948	RERLKLGKSLGRGAFAA	20179	9.61	yes
O95785	PSPKALAKMMGGAGPAA	21680	10.32	yes
Q96HE9	LRKPSLAKALQAGPLAA	8734	4.16	yes
Q92736	LLSVRMGKEEEKLMIAA	3780	1.80	yes
Q7Z2G1	LLPGQMGKLAESEGTAA	662	0.32	no
P35579	KSKSLAKLKNKHEAA	22447	10.69	yes
P09661	KRGAQLAKDIARRSKAA	10660	5.07	yes
Q13217	KHLELGKKLLAAGQLAA	14251	6.78	yes
P68104	KEAAEMGKGSFKYAWAA	22550	10.73	yes
Q96A33	IMNYIIGKNKNSRLAAA	11600	5.52	yes
Q8BMF4	FVSPLAKKLAAEKGIAA	22030	10.49	yes
Q2NKX8	FKLFNLAKDIFPNEKAA	16697	7.95	yes
Q8TD57	ETTKDLAKALAKQCVAA	13795	6.57	yes
Q5VW36	ELYISIAKCLLEMTDAA	499	0.24	no
Q05639	EKEAAEMGKGSFKAA	4225	2.01	yes
P68104	EAAEMGKGSFKYAAA	1139	0.54	no
O95153	AELAVIAKRLEERARAA	637	0.30	no
Blank	A	549	0.26	no
Blank	A	463	0.22	no
Blank	A	490	0.23	no
Blank	A	463	0.22	no
Blank	A	586	0.28	no
Blank	A	424	0.20	no
Blank	A	493	0.23	no
Blank	A	531	0.25	no
VEGFR1 K828A (WT)	RERLALGKSLGRGAFAA	3706	1.76	yes
VEGFR1 K828A (WT)	RERLALGKSLGRGAFAA	3608	1.72	yes
VEGFR1 K828A/K830A	RERLALGASLGRGAFAA	493	0.23	no
VEGFR1 K828A/K830A	RERLALGASLGRGAFAA	596	0.28	no

Table 5.7.5. Smyd3 methylation assay results (VEGFR1 derived motif). In total, 20 peptides (after filtering a library of 2500 peptides with a combined recognition motif of Smyd3 – VEGFR1) were incubated with SAM (H3) and Smyd3. The autoradiograph spots' intensity was compared to the wild–type (VEGFR1) lowest average intensity. If a peptide's intensity was equal or higher than 50% of the wild–type, it is considered methylated by Smyd3.

UniProt ID	Peptide sequence	Spot intensity	Relative to WT	Efficient methylation (> 0.5)
Q14157	VHSPFTKRQAFTPAA	2894	0.15	no
Q8TC05	TSKNDFTKKESRAVSAA	18336	0.93	yes
P78527	SVGPDFGKKRLGLPGAA	12891	0.65	yes
Q8NEG2	SSPTNFSKLISNGYKAA	22347	1.13	yes
Q8TC05	SKNDFTKKESRAVSLAA	16151	0.82	yes
Q9Y4L5	RFFCHFCKGEVSPKLAA	14224	0.72	yes
P21266	QSDQFCKMPINNKAA	7321	0.37	no
Q9Y2U5	PIFEKFGKGGTYPRRAA	23232	1.18	yes
P06276	PALEFTKKFSEWGNNAA	18462	0.94	yes
P02768	NEVTEFAKTCVADESAA	4514	0.23	no
Q9Y657	MKTPFGKTPGQRSRAA	18708	0.95	yes
Q9BPU6	MGKEDFTKIPHGVSGAA	1217	0.06	no
Q8WWX8	LFIYIFTKISVDMYAAA	2606	0.13	no
O95785	LEMNFSKADPPPEAA	578	0.03	no
Q07890	KDLINFSKRRKVAEIAA	3329	0.17	no
Q80V94	ISTEDFGKLWLSFANAA	513	0.03	no
Q05516	IQRELFSKLGELAVGAA	944	0.05	no
P16471-4	GRLAVFTKATLTTVQAA	3892	0.20	no
Q9NZN4	EMPSVFGKENKKKQLAA	21759	1.10	yes
Q9H223	EMPSVFGKENKKRELAA	16412	0.83	yes
Q9Y4G6	AQKAAFGKADDDDVVAA	6312	0.32	no
Blank	A	940	0.05	no
Blank	A	660	0.03	no
Blank	A	843	0.04	no
Blank	A	609	0.03	no
Blank	A	703	0.04	no
Blank	A	595	0.03	no
Blank	A	594	0.03	no
Blank	A	684	0.03	no
Blank	A	763	0.04	no
MEKK2 (WT)	PIFEAFGKGGTYPRRAA	21053	1.07	yes
MEKK2 (WT)	PIFEAFGKGGTYPRRAA	18376	0.93	yes
MEKK2 K260A	PIFEAFGAGGTYPRRAA	859	0.04	no
MEKK2 K260A	PIFEAFGAGGTYPRRAA	747	0.04	no

Table 5.7.6. Smyd3 methylation assay (MEKK2 derived motif). In total, 21 peptides (after filtering a library of 2550 peptides with a combined recognition motif of Smyd3 – MEKK2) were incubated with SAM (H3) and Smyd3. The autoradiograph spots' intensity was compared to the wild–type (MEKK2) lowest average intensity. If a peptide's intensity was equal or higher than 50% of the wild–type, it is considered methylated by Smyd3.

UniProt ID	Peptide sequence	Spot intensity	Relative to WT	Efficient methylation (> 0.5)
O35098	YVTKVMSKGAADMVAAA	20274	4.25	yes
Q9ULC5	YDAENLGKEHFRKPVAA	4053	0.85	yes
Q5VUA4	VTPSISKEEILESAA	394	0.08	no
Q13315	VNLLQLSKMAINHTGAA	2252	0.47	no
Q6SJ93	VLEMDISKKKALQQKAA	22803	4.78	yes
Q9NYR8	TMRDLGKKETLEAAAAA	2729	0.57	yes
P78527	SVGPDFGKKRLGLPGAA	11824	2.48	yes
Q8NEG2	SSPTNFSKLISNGYKAA	18966	3.98	yes
G3UW68	SKTVLISKTELTDVQAA	3109	0.65	yes
Q6DIC7	SKKSGMSKKTNRGSQAA	17649	3.70	yes
Q96A08	SKGATISKKGFKKAVAA	17355	3.64	yes
Q07133	SGSFKLSKKAASGNDAA	17941	3.76	yes
Q9NVM6	RSHSGLSKGSVLSERAA	803	0.17	no
Q9Y6X0	RGTIYIGKKRGRKPRAA	4923	1.03	yes
P17948	RERLKLGKSLGRGAFAA	11851	2.49	yes
A2ABF8	RARKTMSKPGNGQPPAA	9208	1.93	yes
Q9Y232	PKALVIGKDHESKNSAA	21711	4.55	yes
Q9Y2U5	PIFEKFGKGGTYPRRAA	18472	3.87	yes
Q9Y657	MKTPFGKTPGQRSRAA	14662	3.08	yes
Q14789	MEYETLSKKFQSLMSAA	11914	2.50	yes
Q91Z83	LYDNHLGKSNNFQKPAA	16936	3.55	yes
Q9H9Q4	LQRPQLSKVKRKKPRAA	6685	1.40	yes
Q92736	LLSVRMGKEEEKLMIAA	1645	0.35	no
Q7Z2G1	LLPGQMGKLAESEGTAA	1753	0.37	no
P30046	LESWQIGKIGTVMTFAA	1979	0.42	no
O95785	LEMNFSKADPPPEAA	486	0.10	no
Q9NP78	KYYKRLSKEVQNALAAA	20000	4.19	yes
P35232	KVFESIGKFGLALAVAA	6351	1.33	yes
O76021	KSPSLGKKDARQTPKAA	23014	4.83	yes
Q8NF91	KSEVLGKLQELQSAA	832	0.17	no
P23249	KPGSNISKQHRSLLAAA	7137	1.50	yes
Q6DIC7	KKSGMSKKTNRGSQLAA	23110	4.85	yes
Q13217	KHLELGKKLLAAGQLAA	13724	2.88	yes
Q99MY8	KGTIYIGKRRGRKPKAA	9514	2	yes
P62805	KGGKGLGKGGAKRHRAA	21291	4.47	yes
O15131	KFRKLLSKEPNPPIDAA	13157	2.76	yes
P68104	KEAAEMGKGSFKYAWAA	22283	4.67	yes
Q07890	KDLINFSKRRKVAEIAA	7327	1.54	yes
Q80V94	ISTEDFGKLWLSFANAA	2627	0.55	yes
Q05516	IQRELFSKLGELAVGAA	3418	0.72	yes
Q9P2N5	IQKMMSKPQTSGAAA	14314	3	yes
Q96A33	IMNYIIGKNKNSRLAAA	11083	2.32	yes

Q8NDG6	HSLTRISKFRCVWIEAA	1009	0.21	no
P32119	GNARIGKPAPDFKAA	5204	1.09	yes
Q9P2N5	GIQKMMSKPQTSGAYAA	18865	3.96	yes
Q9NVA2	GETGIGKSTLMDTAA	647	0.14	no
Q6UX07	GANSGIGKMTALELAAA	2444	0.51	yes
Q6F5E8	FPRSTLGKLFRRPTPAA	3271	0.69	yes
Q9UHD8	FMKRLSKVVNIVPAA	10950	2.30	yes
P55008	FLRMMLGKRSAILKMAA	10567	2.22	yes
P68104	FEAGISKNGQTREAA	785	0.16	no
P61406	FDAETMSKDSPVVRSAA	751	0.16	no
P17480	EVKDSLGKQWSQLSDAA	1784	0.37	no
Q9NZN4	EMPSVFGKENKKKQLAA	22302	4.68	yes
Q9H223	EMPSVFGKENKKRELAA	16994	3.56	yes
Q6GSS7	ELNKLLGKVTIAQGGAA	15153	3.18	yes
Q0VBD0	EIKMDISKLNAQEAFAA	794	0.17	no
P68104	EAAEMGKGSFKYAAA	2251	0.47	no
E9Q401	DTKSKMSKAAISDQEAA	3768	0.79	yes
Q92736	DTKSKMSKAAVSDQEAA	833	0.17	no
Q68DQ2	DTEGDIGKIEVIPMMAA	478	0.10	no
P11142	DNNLLGKFELTGIAA	500	0.10	no
Q9Y4G6	AQKAAFGKADDDDVVAA	1310	0.27	no
Q07666	AKISVLGKGSMRDKAAA	18404	3.86	yes
Blank	A	720	0.15	no
Blank	A	449	0.09	no
VEGFR1 K828A (WT)	RERLALGKSLGRGAFAA	4696	0.98	yes
VEGFR1 K828A (WT)	RERLALGKSLGRGAFAA	4840	1.02	yes
VEGFR1 K828A/K830A	RERLALGASLGRGAFAA	641	0.13	no
VEGFR1 K828A/K830A	RERLALGASLGRGAFAA	652	0.14	no
Blank	A	483	0.10	no
Blank	A	1728	0.36	no
MEKK2 (WT)	PIFEAFGKGGTYPRRAA	19855	4.16	yes
MEKK2 (WT)	PIFEAFGKGGTYPRRAA	20862	4.38	yes
MEKK2 K260A	PIFEAFGAGGTYPRRAA	928	0.19	no
MEKK2 K260A	PIFEAFGAGGTYPRRAA	747	0.16	no

Table 5.7.7. Smyd3 methylation assay (VEGFR1-MEKK2 combined motif). In total, 64 peptides (after filtering a library of 2500 peptides with a combined recognition motif of Smyd3 – VEGFR1 and Smyd3 – MEKK2 predicted with VIPER) were incubated with SAM (H3) and Smyd3. The autoradiograph spots' intensity was compared to the wild–type (MEKK2, VEGFR1) lowest average intensity. If a peptide's intensity was equal or higher than 50% of the wild–type, it is considered methylated by Smyd3.

5ex3	Relative Spot Intensity					
Substitution	-3	-2	-1	1	2	3
A	1.00	2.95	0.14	1.38	1.30	1.32
C	0.16	0.10	0.01	4.23	1.76	1.67
D	0.05	0.04	0.04	4.12	3.39	3.81
E	0.10	0.07	0.05	0.98	0.21	1.26
F	0.16	0.35	0.05	0.94	1.10	0.90
G	1.78	3.48	1.00	1.06	1.56	1.00
H	0.14	0.29	0.09	2.90	2.25	2.63
I	0.54	1.38	0.10	0.12	1.09	1.41
K	N/E	N/E	N/E	N/E	N/E	N/E
L	0.67	1.00	0.20	0.19	1.00	0.76
M	0.43	1.88	0.14	0.23	0.59	1.27
N	0.33	1.11	0.16	1.30	1.41	1.93
P	0.99	3.99	0.04	0.99	1.29	1.12
Q	0.40	2.20	0.18	1.09	1.21	1.47
R	0.51	1.93	0.34	0.88	1.97	2.60
S	0.35	2.37	0.60	1.00	0.93	0.81
T	0.33	2.27	0.29	0.25	0.79	0.90
V	0.61	2.13	0.17	0.15	0.43	1.12
W	0.04	0.11	0.03	1.66	0.86	0.70
Y	0.17	1.45	0.12	0.59	0.63	0.69

Table 5.7.8. Smyd3-VEGFR1 peptide array spot intensities analyzed with Image J. Their spot relative intensities to wild–type is reported in this table. The data was used to extract an experimental recognition profiles of Smyd3 (PDB ID 5EX3) by using a 50% cut-off and was then used to asses the accuracy of VIPER regarding the Smyd3 protein.

5hq8	Relative Spot Intensity							
Substitution	-4	-3	-2	-1	1	2	3	4
A	0.82	1.00	0.10	0.17	0.74	0.58	0.27	0.23
C	1.60	1.21	0.09	0.11	0.03	0.04	0.04	0.05
D	1.51	0.19	0.04	0.04	0.03	0.02	0.03	0.06
E	1.00	0.11	0.03	0.02	0.02	0.02	0.03	0.05
F	0.99	0.09	1.00	0.05	0.53	0.80	0.53	1.97
G	1.27	0.49	1.21	1.00	1.00	1.00	0.64	0.90
H	0.75	0.08	0.08	0.04	0.40	0.14	0.23	0.13
I	0.69	0.10	0.53	0.05	0.95	0.65	0.55	0.78
K	N/E	N/E	N/E	N/E	N/E	N/E	N/E	N/E
L	0.58	0.07	1.43	0.04	1.30	1.07	1.05	1.32
M	0.76	0.09	0.30	0.10	0.46	0.50	0.55	1.29
N	1.07	0.15	0.07	0.04	0.59	0.14	0.52	0.09
P	1.29	0.15	0.05	0.05	0.16	0.45	0.22	0.10
Q	1.23	0.18	0.14	0.07	0.85	0.32	0.67	0.76
R	0.64	0.09	0.03	0.03	0.06	0.06	0.10	0.12
S	0.92	0.11	1.35	0.11	0.74	0.75	0.37	1.00
T	0.72	0.18	0.27	0.18	1.26	1.12	1.00	1.31
V	0.37	0.07	0.08	0.03	0.73	0.54	0.92	1.97
W	0.56	0.12	0.01	0.01	0.03	0.03	0.03	0.03
Y	0.58	0.08	0.05	0.04	0.50	0.73	0.75	1.00

Table 5.7.9. Smyd3-MEKK2 peptide array spot intensities analyzed with Image J. Their spot relative intensities to wild–type is reported in this table. The data was used to extract an experimental recognition profiles of Smyd3 (PDB ID 5HQ8) by using a 50% cut-off and was then used to asses the accuracy of VIPER regarding the Smyd3 protein.

5.8. Binning statistics of various protein-peptide complexes

Protein-Peptide Complex	PDB ID	Method	False Negative	True Positive	True Negative	False Positive	Accuracy
Smyd2 – p53	3TG5	Kmeans	39	69	28	4	69.3
		Cut-off (5%)	32	76	29	3	75.0
		Improved FF	45	63	29	3	65.7
		VIPER (1%)	4	104	30	2	95.7
		VIPER (5%)	4	104	29	3	95.0
		VIPER (10%)	3	105	24	8	92.1
		VIPER SSD	2	106	31	1	97.9
Smyd2 – p53	3S7F	Kmeans	29	39	27	5	66.0
		Cut-off (5%)	30	38	30	2	68.0
		Improved FF	36	32	30	2	62.0
		VIPER (1%)	4	64	31	1	95.0
		VIPER (5%)	4	64	30	2	94.0
		VIPER (10%)	2	66	27	5	93.0
		VIPER SSD	3	65	30	2	95.0
Gads – SLP76	2D0N	Kmeans	1	68	52	59	66.7
		Cut-off (5%)	17	52	75	36	70.6
		Improved FF	14	55	77	34	73.3
		VIPER (1%)	0	69	49	62	65.6
		VIPER (5%)	0	69	27	84	53.3
		VIPER (10%)	0	69	14	97	46.1
		VIPER SSD	0	69	49	62	65.6
Erbin – ERBb2	1MFG	Kmeans	32	82	22	44	57.8
		Cut-off (5%)	51	63	31	35	52.2
		Improved FF	32	82	22	44	57.8
		VIPER (1%)	7	107	44	22	83.9
		VIPER (5%)	3	111	29	37	77.8
		VIPER (10%)	2	112	27	39	77.2
		VIPER SSD	7	107	45	21	84.4
ATXR5 – H3.1K27	4O30	Kmeans	43	101	43	73	55.4
		Cut-off (5%)	85	59	62	54	46.5
		Improved FF	50	94	80	36	66.9
		VIPER (1%)	3	141	69	47	80.8
		VIPER (5%)	1	143	65	51	80.0
		VIPER (10%)	0	144	61	55	78.8
		VIPER SSD	1	143	69	47	81.5
Set8 – fH4K20	2BQZ	Kmeans	2	50	63	45	70.6
		Cut-off (5%)	2	50	76	32	78.8
		Improved FF	2	50	86	22	85.0
		VIPER (1%)	3	49	80	28	80.6
		VIPER (5%)	2	50	67	41	73.1

		VIPER (10%)	2	50	58	50	67.5
		VIPER SSD	1	51	46	62	60.6
Smyd3 – VEGFR1	5EX3	VIPER (1%)	34	48	17	1	65.0
Smyd3 – MAP3K2	5HQ8	VIPER (1%)	23	83	30	4	80.7

Table 5.8.1. Binning statistics for each method tested on various protein-peptide complexes. Refer to chapter 2 for the description of each method.

5.9. MODELLER python script example for loop modelling

```
from modeller import *
from modeller.automodel import *    # Load the automodel class


log.verbose()
env = environ()


      # directories for input atom files


path = '/home/ogagnon/scratch/PTM_JOBS/2287247/Prep_Input/'
env.io.atom_files_directory = [path]
env.edat.dynamic_sphere = True
env.io.hydrogen = True
env.io.hetatm = True
env.io.water = True
env.libs.topology.read(file='$(LIB)/top_heav.lib')
env.libs.parameters.read(file='$(LIB)/par.lib')



PDB = '3glt'
struct = '3glt'
seq = '3glt_seq'


class MyModel(automodel):
    def special_patches(self, aln):


          # Rename chains and renumber the residues


        self.rename_segments(segment_ids=['A','B'], renumber_residues=[121,640])


    def select_atoms(self):
        return selection( self.residue_range('642:B','642:B') )


mdl1 = model(env)
mdl1.read(file=PDB)
aln1 = alignment(env)
aln1.append_model(mdl1, align_codes=struct)
aln1.write(file=path+PDB+'.seq')


loop = MyModel(env, alnfile=path+PDB+'.ali', knowns=(struct), sequence = seq)
loop.starting_model= 1
loop.ending_model  = 1
loop.make()
loop.write(file=path+PDB+'.repair.2.pdb', model_format='PDB', no_ter=False)
```