Discovering structural motifs using a structural alphabet

Sebastian Bartschat

16. Juli 2007

Inhaltsverzeichnis

1 Einführung

Wozu dient Magnesium?
Magnesium ist einer der vielseitigsten metallischen Co-Faktoren, der sowohl im intra- wie auch im extrazellulären Bereich Verwendung findet. Er ist an Reaktionen beteiligt die Phosphatgruppenübertragung realisieren (in Verbindung mit ATP), er dient zur Stabilisierung von Proteinstrukturen, Membranpotenzialen und Nukleinsäuren. Besonders ist aber auch seine calciumantagonistische Wirkung hervorzuheben, denn Magnesium kann Calciumkanäle blockieren (z.B. an der synaptischen Endplatte) und somit dessen Einstrom vermindern, was zu einer gehemmten Transmitterfreisetzung führt.

Bindungseigenschaften
Anders als für Zn^{2+} oder Ca^{2+} sind für Magnesiumbindungsstellen bis jetzt nur sehr kurze Sequenzmotive bekannt: NA**DF**D**GD** für die RNA-Polymerase und die DNA-Polymerase I, sowie das Motiv YX**DD** / LX**DD** für die Reverse Transkriptase und die Telomerase (die fett gedruckten Buchstaben sind die bindenden AS).

Magnesium neigt zur Bindung an den Seitenketten von Asp und Glu, gefolgt von den Seitenketten von Asn und Gln. Neben diesen Erkenntnissen ist vorallem über die Strukur und deren Präferenzen nicht viel bis gar nichts bekannt.

2 Fragestellungen

Welche Ziele werden in dieser Arbeit verfolgt?
Zum Einen möchte man Klarheit über die Struktur von Magnesiumbindungsstellen, d.h. besitzen diese Bindungsstellen eine bestimmte Sekundärstruktur? Da nur Proteine mit eingeschränkter Sequenzähnlichkeit betrachtet werden, interessiert natürlich, ob es trotzdem Strukturmotive gibt und wenn ja, wie sehen diese aus? Des Weiteren möchte man wissen, ob es möglich ist, diese Motive bestimmten Funktionen im Protein zuzuordnen, um so eventuell Rückschlüsse auf Proteine zu ziehen über die nicht sehr viel bekannt ist.

3 Generierung des Struktur-Alphabets

Die ganzen Untersuchungen in der vorgestellten Arbeit stützen sich auf die vorherige Codierung der Proteinsequenz in ein Struktur-Alphabet. Das Verfahren zur Generierung eben dieses Alphabets wird im Weiteren kurz vorgestellt.

Mit der Erstellung dieses Struktur-Alphabets wollten die Autoren eine genauere Beschreibung der Proteinstruktur erreichen als es mit den normalen Sekundärstrukturen möglich war.

Dazu werden einige Vorüberlegungen angestellt:
Eine konventionelle Methode zur Beschreibung der Proteinstruktur ist es, die Positionen der konsekutiven AS durch ihre Diederwinkel zu beschreiben. Zur besseren Approximation wird die Proteinsequenz in sogenannte ProteinBlöcke (PB) mit einer Länge von $M = 5$ AS zerlegt. Solch ein ProteinBlock von AS wird eindeutig durch einen Vektor mit $2(M-1)$ Diederwinkeln beschrieben:

$$V(\psi_{n-2},\, \phi_{n-1},\, \psi_{n-1},\, \phi_n,\, \psi_n,\, \phi_{n+1},\, \psi_{n+1},\, \phi_{n+2})$$

Da diese Blöcke überlappend sind, wird ein Protein der Länge L durch $L-4$ solcher ProteinBlöcke beschrieben.

3.1 unsupervised cluster analyser

Zur Generierung des Alphabets wird ein Kohonennetz (auch Selbstorganisierende Karte (SOM) genannt) genutzt. Diese Karte besteht zu Beginn aus einer bestimmten Anzahl von Neuronen, die in diesem Fall je einen ProteinBlock beschreiben. Des Weiteren speichert jedes Neuron die Informationen eines Objektes, in diesem Fall den Diedervektor.
Als Abstandmessungen zwischen 2 Neuronen V_1 und V_2 dient die RMSDA-Methode (root mean square deviation on angular values), die die Euklidische Distanz zwischen den Winkelpaaren berechnet:

$$\mathrm{RMSDA}(V_1, V_2) = \sqrt{\frac{\sum_{i=1}^{i=M-1} [\psi_i(V_1) - \psi_i(V_2)]^2 + [\phi_{i+1}(V_1) - \phi_{i+1}(V_2)]^2}{2(M-1)}}$$

wobei $\{\psi_i(V_j), \phi_{i+1}(V_j)\}$ mit $j \in \{0, 1\}$ die Reihe von Diederwinkeln eines Vektors beschreibt. Der Lernprozess der SOM wurde in 2 Phasen unterteilt und anschließend die Neurone auf eine optimale Anzahl reduziert.

1. **first training**
   Zu Beginn besteht die Karte aus B Neuronen, die durch einen Vektor $W(k)$ je einen PB beschreiben.
   Nun werden von C Proteinen die Signale eingelesen. Ein Signal beschreibt dabei die konsekutive Folge der Diedervektoren eines Proteins der Länge L, d.h. einlesen der Vektoren V_1, V_2, V_3 ... V_{L-4} eines Proteines.
   Zwischen allen Neuronen der Karte wird mittels RMSDA derjenige Vektor $W(k)$ gesucht, der am dichtesten zum induzierten Vektor $V(m)$ liegt. Anschließend erfolgt eine Neugewichtung dieses Vektors :

   $$W(k) \leftarrow W(k) \; + \; (V(m) \; - \; W(k)) \; * \; \nu(c)$$

   $\nu(c) \; = \; \nu_0 / \, (1 \; + \; \tau * c)$ beschreibt eine Lernrate, die während des Lernvorganges abnimmt. (ν_0 und τ sind vorher festgelegt)

2. **second training**

   Dieser Schritt dient der Verfeinerung des Ergebnisses. Dafür werden die C Proteine nochmals als Signal konsekutiver Vektoren eingelesen und die Transitionen zwischen 2 aufeinanderfolgenden Blöcken berechnet. Anschließend berechnet man daraus Übergangswahrscheinlichkeiten, die in einer Transitionsmatrix dargestellt werden. Diese Matrix beschreibt, welcher PB mit welcher Wahrscheinlichkeit in einen anderen PB übergeht.

   Im Zweiten Trainingsschritt werden erneut C Proteine eingelesen. Für einen induzierten Vektor $V(m)$ wird mittels RMSDA die Menge der n-dichtesten Vektoren $W(k)$ bestimmt (n ist user-defined). Für die Neugewichtung wird nun aus der Menge derjenige Vektor ausgewälht, der die größte Übergangswahrscheinlichkeit des Vektors $W(j)$ hat, wobei $W(j)$ mit dem zuvor induzierten Vektor $V(m-1)$ assoziiert wurde.

3. **shrinking process**
   Das Ziel besteht darin, die Anzahl der Neurone in der Karte auf eine gut handhabbare Anzahl zu verkleinern. Dafür werden die Ähnlichkeiten zwischen je 2PB der Karte verglichen :

   - Strukturähnlichkeit
     $\text{RMSDA}(W_1, W_2) \leq \xi_0$; ξ_0 ist eine vom Nutzer gewählte Schranke
   - Transitionsähnlichkeit
     die Transitionswahrscheinlichkeiten zwischen den 2 PBs dürfen sich nur marginal unterscheiden

Zwei PBs sind genau dann ähnlich, wenn sie sowohl strukturell ähnlich sind, als auch Transitionsähnlichkeit aufweisen
Folgender Algorithmus reduziert die Neurone:

(1) `untersuche alle PB`$_i$ `und PB`$_j$ `der Karte K auf Ähnlichkeit`
($i, j \in \{1 \ldots |K|\}$)
→ strukturelle Ähnlichkeit
→ Translationsähnlichkeit

(2) `falls PB`$_i$ `und PB`$_j$ `ähnlich: lösche den PB der seltener ist und gehe zu Schritt 1`
`sonst: ENDE`

Der Algorithmus resultierte in 16 ProteinBlöcken:

				Transitions (%)			Str. II (%)			
PB label	Frequency (%)	RMSD (Å)	*anr*	1st	2nd	3rd	α	Coil	β	Coarse char.
a	3.93	0.52	1.01	54.8(*c*)	16.5(*f*)	8.0(*b*)	0.1	76.7	23.3	N-cap β
b	4.58	0.51	1.00	44.4(*d*)	17.9(*c*)	13.7(*f*)	0.2	86.7	13.1	N-cap β
c	8.63	0.51	1.28	62.2(*d*)	24.4(*f*)	5.6(*e*)	0.1	58.2	41.7	N-cap β
d	18.84	0.48	2.74	51.9(*f*)	25.6(*c*)	19.2(*e*)	0.0	28.4	71.6	β
e	2.31	0.54	1.11	80.4(*h*)	9.1(*d*)		0.0	49.8	50.2	C-cap β
f	6.72	0.50	1.00	60.7(*k*)	36.3(*b*)		0.0	72.5	27.5	C-cap β
g	1.28	0.74	1.05	37.5(*h*)	28.0(*c*)	19.1(*o*)	6.9	83.8	9.3	mainly coil
h	2.35	0.62	1.04	62.4(*i*)	18.1(*j*)	10.2(*k*)	0.0	81.5	18.5	mainly coil
i	1.62	0.56	1.01	87.7(*a*)			0.0	94.5	5.5	mainly coil
j	0.96	1.03	1.01	17.0(*a*)	16.6(*b*)	16.1(*l*)	3.7	87.9	8.4	mainly coil
k	5.46	0.59	1.00	76.2(*l*)	13.6(*b*)		35.1	64.2	0.7	N-cap α
l	5.35	0.63	1.01	68.5(*m*)	9.2(*p*)	7.0(*c*)	44.4	54.9	0.7	N-cap α
m	30.04	0.43	6.74	33.8(*n*)	18.5(*p*)	9.7(*b*)	86.7	13.2	0.1	α
n	1.93	0.61	1.03	90.9(*o*)			68.4	31.3	0.3	C-cap α
o	2.60	0.60	1.02	74.7(*p*)	8.3(*m*)		43.1	56.8	0.1	C-cap α
p	3.41	0.46	1.00	58.1(*a*)	22.7(*c*)	11.1(*m*)	11.2	87.5	1.3	C-cap α to N-cap β

3.2 Strukturvorhersage mittels Bayes-Theorem

Das Ziel besteht darin, die 3D-Struktur des Proteins bestmöglich aus der AS-Sequenz zu approximieren.

Dafür werden alle Proteine des Trainingssets nochmals eingelesen und mittels der RMSDA-Methode deren Vektoren mit je einem Neuron assoziiert; dadurch sind einem Neuron des Kohonennetzes mehrere induzierte Vektoren zugeordnet. Daraus kann man nun eine Matrix für das Vorkommen einer AS je ProteinBlock berechnen, die angibt mit welcher Wahrscheinlichkeit die AS a_i in einem PB k auftaucht.

Ein Sequenzfensters der Breite [$-\omega, +\omega$] wird auf der Proteinsequenz um das zentrale $C\alpha$ eines gegebenen PB erstellt (Mitte des ProteinBlockes ist auch Mitte des Sequenzfensters). Man berechnet daraus die absoluten Häufigkeiten $n^k_{i,j}$ (k - ProteinBlock; i - Nummer der AS; j - Position im Fenster), diese beschreiben wie oft eine AS an einer Stelle j des PB k auftrat. Aus diesen Häufigkeiten kann man nun die bedingte Wahrscheinlichkeiten $P(a_i \text{ in } j|\text{PB}_k) = \frac{n^k_{i,j}}{N_k}$ berechnen (N_k - absolute Häufigkeit des Blockes k) .

Zur weiteren Anaylse der Daten und zur Feststellung der Güte der Ergebnisse werden noch 2 Berechnungsmodelle eingeführt:

1. Kullback-Leibler-Divergenz

$$K_k(p_j, q) \;=\; \sum_i p_{ji} * \ln\left(\frac{p_{ji}}{q_i}\right)$$

   Diese Divergenz beschreibt den Unterschied der beobachteten AS-Verteilung an einer Stelle j im Sequenzfenster im Vergleich zur allgemeinen AS-Verteilung, d.h. damit kann man die Positionen in einem Sequenzfenster finden, die eine große Spezifität enthalten. Die relative Entropie $K_k(p_j, q)$ multipliziert mit $2N$ folgt einer χ^2-Verteilung mit 19 Freiheitsgraden, sodass $\chi^2_{19}/2N$ als untere Schranke für das KLd Profil dienen kann.

2. Z-Score

$$z \;=\; \frac{\left(n^k_{ij} \;-\; n_{ib}\right)}{\sqrt{n_{ib}}}$$

   $n_{ib} \;=\; N_k * f_i$ beschreibt den Erwartungswert der i-ten AS (N_k - Häufigkeit des Blockes k; f_i - Frequenz der Aminosäure i).
   Diese Normalisierung dient zur Suche nach AS, die eine große Spezifität für eine bestimmte Position im ProteinBlock k auweisen.

Vorhersage mittels Bayes-Theorem:

Man möchte nun mittels einer gegebenen AS-Sequenz möglichst gut auf die dahinter steckende Struktur, sprich auf den ProteinBlock schließen können.
X_s beschreibt nun eine Proteinsequenz der Breite $[-\omega, +\omega]$ um die zentrale Position s. Nun ergibt sich für die Wahrscheinlichkeit den Block k vorzufinden, wenn man die Sequenz X_s beobachtet, folgende Berechnung durch den Satz von Bayes:

$$\mathrm{P(PB}_k|X_s) \;=\; \frac{\mathrm{P(X}_s \,|\, \mathrm{PB}_k) \;*\; \mathrm{P(PB}_k)}{\mathrm{P}(X_s)}$$

Die Wahrscheinlichkeit $\mathrm{P(PB}_k)$ beschreibt das Auftreten des ProteinBlockes k an sich. $\mathrm{P}(X_s) = \prod_{j=-\omega}^{j=+\omega} \mathrm{P}(a_i)$ ist die Wahrscheinlichkeit die Sequenz X_s zu beobachten. Schießlich kann man mit $\mathrm{P(X}_s \,|\, \mathrm{PB}_k) \;=\; \prod_{j=-\omega}^{j=+\omega} \mathrm{P}(a_j \,|\, \mathrm{PB}_k)$ die Wahrscheinlichkeit berechnen von einem gegebenen PB auf eine Sequenz X_s zu schließen.

Wie findet man nun den am besten geeigneten PB* für eine gegebene Sequenz?

$$R_k \;=\; \frac{\mathrm{P(X}_s \,|\, \mathrm{PB}_k)}{\mathrm{P}(X_s)} \;=\; \frac{\mathrm{P(PB}_k \,|\, X_s)}{\mathrm{P(PB}_k)}$$

Mit diesem Verhältnis vergleicht man die Wahrscheinlichkeiten einen ProteinBlockes k zu beobachten, unter der Bedingung einer gegebenen Sequenz, mit der Wahrscheinlichkeit überhaupt diesen Block zu beobachten. Wenn nun $\ln(R_k)$ positiv ist, wird die Struktur des ProteinBlock k durch die Sequenz X_s gefördert. Nun gibt es 2 Ansätze:

- wähle den Block PB_k, bei dem R_k maximal wird
- wähle aus den r-besten Blöcken

Erweiterungen

(i) *1 fold - n Sequences*
Im bisherigen Modell stützte man sich auf eine Matrix für das Vorkommen der AS je Position in einem ProteinBlock und somit konnten auch signifikant unterschiedliche Sequenzen der selben Struktur zugeordnet werden. Dieser Ansatz versucht nun verschiedene Sequenzfamilien getrennt für einen ProteinBlock zu betrachten.
Dazu werden alle Sequenzen, die dem PB zugeordnet sind, willkürlich in f verschieden Familien unterteilt. Nun folgt ein Trainingsalgorithmus ähnlich dem PB-Training:

1. Berechne für jeden PB k die f verschiedenen Erscheinungsmatrizen (für jede einzelne Familie): also $n^l_{i,j}$ für $i \in \{1 \dots 20\}$, $j \in \{-\omega \dots + \omega\}$ und $l \in \{1 \dots f\}$
2. Berechne für jede Sequenz X_s die f-verschiedenen Wahrscheinlichkeiten $\mathrm{P(X_s|PB^l_k)}$ und füge X_s derjenigen Gruppe mit der Größten hinzu. Starte die Prozedur neu, bis sich die Erscheinungsmatrizen nur noch geringfügig ändern.

Dadurch erreicht man eine genauere Vorhersage.

(ii) *1 Sequence - n folds*
Hier geht man davon aus, dass eine Sequenz durch die Wahrscheinlichkeitsverteilung zu mehreren PB passen würde. (große Homogenität zwischen den ersten Scores R_k)
Mit Hilfe der Shannon-Entropie lässt sich der Informationsgehalt des Scores berechnen:

$$H \;=\; -\sum_k S_k \ln(S_k)$$

Wobei die Entropie nun mit Wahrscheinlichkeiten $S_k = R_k / \sum_l R_k$ berechnet wird.
Die transformierte Entropie $N_{eq} = \exp\{H\}$ gibt nun Auskunft, wie groß die Menge der r-besten PB gewählt werden sollte um eine bestimmte Vorhersagegenauigkeit zu erreichen.

4 Anwendung auf Mg^{2+}-Bindungsstellen

4.1 Vorbereitungen

Alle Proteine wurden aus der PDB bezogen, dabei wurden folgende Kriterien festgelegt:

- Sequenzähnlichkeit der Proteine $< 30\%$
- Auflösung $< 2,5$ Å
- Anzahl AS, die an Bindung beteiligt sind ≥ 3 AS

Diese Eingrenzungen ergaben in diesem Beispiel ein Proteinset von 70 Proteinen mit insgesamt 77 zu untersuchenden Bindungen. Diese wurden mittels des ProteinBlockExperts (einem Webinterface) in das jeweilige Struktur-Alphabet codiert.

Wie werden Bindungsmotive definiert?
Ein Motiv muss aus den selben Buchstaben des Sturktur-Alphabets bestehen, die Zwischenräume müssen einen ähnlichen Bereich umfassen und diese Sequenz muss mindestens dreimal im Proteinset auftauchen.

4.2 Analyse

Zu Beginn werden eventuelle strukturelle Präferenzen der Bindungsstellen untersucht.
Dafür wird das Verhältnis zwischen einem Buchstaben in der 1^{st}- oder 2^{nd}- shell Bindungstelle mit dem allgemeinen Auftreten dieses Buchstabens verglichen.

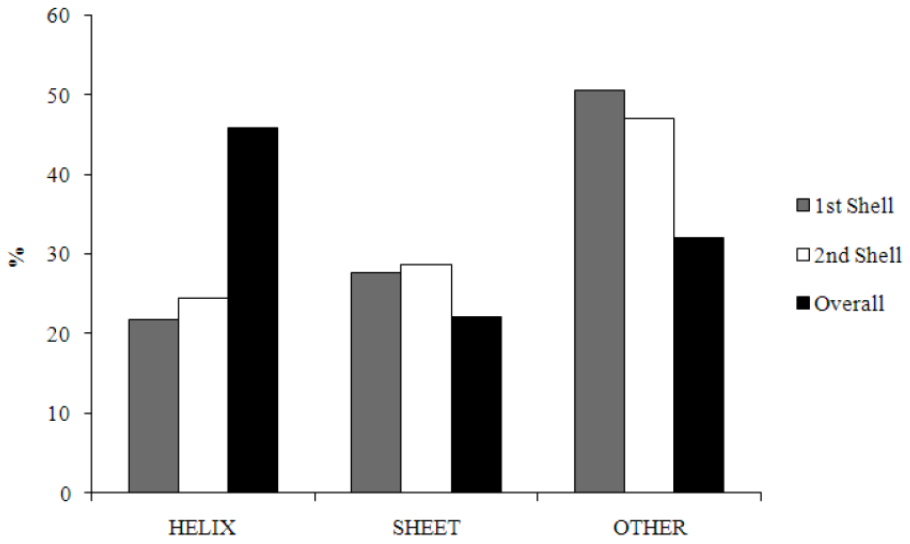
	1st-shell vs. all residues			2nd-shell vs. all residues		
Letter, *x*[a]	$\nu_{x,1}/\nu_{x,\,all}$[b]	T-test[c]	p-value[c,d]	$\nu_{x,2}/\nu_{x,\,all}$[e]	T-test[c]	p-value[c,d]
a	1.47	1.4037	0.0802	0.57	2.4731	**0.0067**
b	1.86	2.7909	**0.0027**	1.20	1.2200	0.1113
c	0.56	2.0160	**0.0219**	0.50	4.3510	**<0.0001**
d	1.23	1.7376	**0.0412**	1.23	3.1829	**0.0008**
e	1.46	1.0111	0.1560	2.03	4.1825	**<0.0001**
f	1.47	1.9389	**0.0263**	1.70	5.4060	**<0.0001**
g	1.15	0.2494	0.4015	1.18	0.5381	0.2953
h	5.29	9.3752	**< 0.0001**	1.19	0.7921	0.2142
i	0	1.8928	**0.0292**	1.34	1.1910	0.1168
j	2.21	1.6156	0.0531	1.54	1.3401	0.0901
k	1.40	1.4992	0.0669	1.60	4.1820	**<0.0001**
l	0.76	0.9209	0.1786	1.08	0.5978	0.275
m	0.52	2.9377	**0.0017**	0.74	5.2192	**<0.0001**
n	0.53	1.1306	0.1291	0.88	0.5208	0.3013
o	1.52	1.4066	0.0798	0.35	3.3637	**0.0004**
p	0	3.1174	**0.0009**	0.77	1.3204	0.0934
SSE, *x*						
Loop	1.56	2.5575	**0.0053**	1.47	2.1874	**0.0144**
β-strands	1.30	1.0780	0.1405	1.34	1.2170	0.1118
α-helices	0.47	3.6454	**0.0002**	0.51	3.3621	**0.0004**

Anhand der Tabelle sieht man, dass z.B. first-shell AS (in der Spalte $\nu_{x,1}$ / $\nu_{x,\mathrm{all}}$) eine große Neigung zu Strukturen des Typs *'b', 'd', 'f'* und *'h'* haben, da das entsprechende Verhältnis über 1 liegt und der dazugehörige p-Wert unter der Grenze von 0.05 .

Second-shell AS neigen hingegen zu Strukturen des Types *'d', 'e', 'f'* und *'k'* (das Verhältnis ist in der Spalte $\nu_{x,2}$ / $\nu_{x,\mathrm{all}}$ zu sehen).

Solch ein Verhältnis kann man auch für Sekundärstrukturen (siehe die letzen drei Zeilen) berechnen. Man sieht für Mg^{2+}Bindungsstellen, dass die Häufigkeit eines Loops sowohl für

1^{st}- als auch 2^{nd}-shell Liganden signifikant erhöht ist, wohingegen die α-Helizes in solchen Bindungsstellen signifikant seltener sind. Dieses Ereignis ist auch noch mal in folgenden Diagramm zu sehen:



Welche Motive sind im Proteinset enthalten?
Anhand der vorher definierten Kriterien für ein Motiv wurden alle 77 Bindugsstellen untersucht und man fand 4 Bindungsmotive für first-shell AS und 5 partielle Motive für second-shell AS. Da zuviele AS an den 2^{nd}-shell Bindungen beteiligt sind, konnte man dort keine kompletten Motive extrahieren.

1^{st}-shell : 4 Motive, die 21% aller Bindungsstellen repräsentieren
- → *e(24-47)h(24)k*
- → *f(1)h(109-349)b*
- → *f(2)h(126-158)m*
- → *k(26-29)h(1)a*

2^{nd}-shell : zu viele AS definieren eine Bindungsstelle, daher nur partielle Motive
- → *f(1)lm* , *kl(0-1)m*, *d(1-2)ff* u.a

Welche Möglichkeiten ergeben sich aus den Ergebnissen?
Die Analysen der Proteine mit gleichen Bindungsmotiven zeigen, dass diese Proteine meist zu derselben Superfamilie gehören (sie besitzen gleiche CATH-Nummern), d.h. solche Proteine, die noch nicht klassifiziert sind, könnten durch ihre Bindungsstellenstruktur evtl. eingeordnet werden.

Ist es möglich durch die Struktur der Bindungsstelle auf eine Funktion des Proteins zu schließen?

Um dies zu klären, muss man die funktionellen Gruppen des Proteins und die EC-Codes untersuchen. Allerdings zeigte sich schon bei den wenigen Proteinen, die ein gemeinsames Motiv aufweisen, dass es durchaus sein kann, dass die Proteine unterschiedliche Funktionen ausführen.

Die Bindungsstellen im Überblick:

Motif[a]	*PDB code*	*Mg^{2+}-Ligands*	*CATH number*[b]	*Functional Group*[c]	*EC code*[d]
e(24–47)h(24)k	1SJC	D^{189}, E^{214}, D^{239}	3.20.20.120	Lyase[e], Isomerase[f]	-
	1TKK	D^{191}, E^{219}, D^{244}	3.20.20.120	Isomerase[f]	-
	2AKZ	D^{244}, E^{292}, D^{317}	-	Lyase[e]	4.2.1.11
f(1)h(109–349)b	1O08	D^{1008}, D^{1010}, D^{1170}	3.40.50.1000	Isomerase[f]	5.4.2.6
	1U7P	D^{11}, D^{13}, D^{123}	NYC	Hydrolase[g]	-
	1WPG	D^{351}, T^{353}, D^{703}	3.40.50.1000	Hydrolase[g]	3.6.3.8
	2B82	D^{44}, D^{46}, D^{167}	3.40.50.1000	Hydrolase[g]	3.1.3.2
	2C4N	D^{9}, D^{11}, D^{201}	NYC	Hydrolase[g]	-
f(2)h(126–158)m	1KA1	D^{142}, D^{145}, D^{294}	3.30.540.10	Hydrolase[g]	3.1.3.7
	1NUY	D^{1118}, D^{1121}, E^{1280}	3.30.540.10+ 3.40.190.80	Hydrolase[g]	3.1.3.11
	2BJI	E^{1090}, D^{1093}, D^{1220}	3.30.540.10+ 3.40.190.80	Hydrolase[g]	3.1.3.25
k(26–29)h(1)a	1ITZ	D^{168}, N^{198}, I^{200}	3.40.50.970	Transferase[h]	2.2.1.1
	1POX	D^{447}, N^{474}, Q^{476}	3.40.50.970+ 3.40.50.1220	Oxidoreductase[i]	1.2.3.3
	1UMD	D^{175}, N^{204}, Y^{206}	3.40.50.970	Oxidoreductase[i]	1.2.4.4
	1ZPD	D^{440}, N^{467}, G^{469}	3.40.50.970	Lyase[e]	4.1.1.1
	2C3M	D^{963}, T^{991}, V^{993}	3.40.50.970	Oxidoreductase[i]	1.2.7.1

Spezifität der Motive
Als Erstes untersucht man die Spezifität für Metallproteine. Dafür untersucht man Nichtmetallproteine auf ihre Bindungsstellen hin, diese hat man aus de Brevern's Datenbank (enthält Proteine, die in das 3D-Struktu-Alphabet codiert wurden) entnommen.
(Auswahl der Proteine geschieht nach vorab definierten Kriterien)

→ matches für *f(1)h(109-349)b* und *k(26-29)h(1)a*

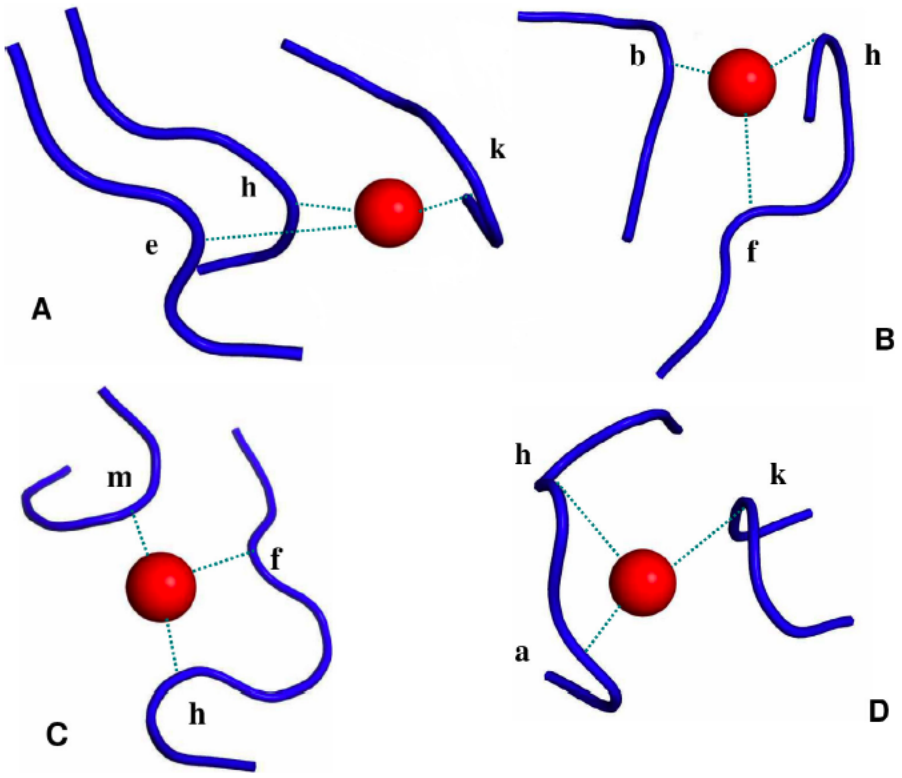
→ *e(24-47)h(24)k* & *f(2)h(126-158)m* sind metallspezifisch, da sie in keinem untersuchten Protein vorkommen

Den Abgleich gegen calciumbindende Proteine führt man deswegen durch, weil Calcium der Co-Faktor ist, der Magnesium am Nächsten kommt und man so die Magnesiumspezifität prüfen kann. Man untersucht die Ca^{2+}-Bindungsstellen mit dem selben Verfahren wie vorher für Magnesiumbindungsstellen beschrieben.

→ *f(1)h(109-349)b* und *k(26-29)h(1)a* wurden in 1 bzw. 2 Proteinen gefunden

→ *e(24-47)h(24)k & f(2)h(126-158)m* sind metall- und magnesiumspezifisch, da sie in keinem Nichtmetallprotein und auch in keinem calciumbindenden Protein vorkamen.

Ergebnis: 4 Bindungsmotive für magnesiumbindende Proteine



5 Bemerkungen

- Validierung der Methode erfolgte nur gegen 42 Zink-Finger Bindungsstellen. Diese zeigten allerdings alle ein bestimmtes Motiv.

Literatur

[1] Minko Dudev, Carmay Lim *Discovering structurel motifs using a structural alphabet: Application to magnesium-binding sites*, Bioinformatics 2007 Mar 28; 8(106)

[2] de Brevern AG, Etchebest C, Hazout S *Bayesian Probabilistic Approach for Predicting Backbone Sturctures in Terms of Protein Blocks* PROTEINS: Structures, Function and Genetics 2000