The Threading Program of PAINT

PAINT is a PAirwise INteraction based Threading algorithm like RAPTOR(1). Environment fitness, residue mutation, secondary structure match, pairwise interactions between cores of proteins, and gap penalty are considered in PAINT. The score function of PAINT is expressed by a linear equation, where the threading is incorporated into integer variables of the score function and can be solved by integer linear programming.

In PAINT, the structure of a protein is composed of cores and loops. Cores are α helices and β sheets. When a query sequence is aligned to a template, core regions of the template are not allowed to have gaps because core regions are mostly conserved. Finally, the interaction between residues in different core regions is considered since interaction involving loop regions is relatively insignificant for fold recognition (1).

Let the ith (i=1,…,M) core of a template be denoted as $core_i = (head_i, head_i + len_i - 1)$, where $head_i$ is the starting position of $core_i$, len_i is the length of $core_i$, M is the number of cores in the template. $D[i]$ is the set of all the possible positions of a query sequence aligned to $head_i$. And $D[j|i;l]$ is the set of all the possible positions of a query sequence aligned to $head_j$, assuming that of $head_i$ is aligned the lth position of the query sequence. Assuming $j>i$, the number of $D[j|i,l]$ must be less than that of $D[j]$ because the alignment position for $head_i$ must be less than that for $head_j$. $x_{i,l}$ is a 0-1 integer and denotes that $head_i$ is aligned to position l of a

query sequence. Here threading is represented by a set of $x_{i,l}$'s. $y_{i,l;j,k}$ is also a 0-1 integer and denotes that $head_i$ is aligned to position l of the query while $head_j$ is aligned to position k of the query at the same time.

Like RAPTOR, the total score function for PAINT is

$$
\begin{aligned}
E &= w_{env}E_{env} + w_{mut}E_{mut} + w_{\sec}E_{\sec} + w_{gap}E_{gap} + w_{pair}E_{pair} \\
&= w_{env}\sum_{i=1}^{M}\sum_{l\in D[i]}\left[x_{i,l}\times\sum_{r=0}^{len_i-1} Environment(head_i+r, l+r)\right] \\
&+ w_{mut}\sum_{i=1}^{M}\sum_{l\in D[i]}\left[x_{i,l}\times\sum_{r=0}^{len_i-1} Mutation(head_i+r, l+r)\right] \\
&+ w_{\sec}\sum_{i=1}^{M}\sum_{l\in D[i]}\left[x_{i,l}\times\sum_{r=0}^{len_i-1} SS(head_i+r, l+r)\right] \\
&+ w_{gap}\sum_{1\le i\le j<M}\sum_{l\in D[i]}\sum_{k\in D[j|i,l]} y_{i,l;j,k}G(i,l,k) \\
&+ w_{pair}\sum_{1\le i\le j<M}\sum_{l\in D[i]}\sum_{k\in R[j|i,l]} y_{i,l;j,k}P(i,j,l,k) \\
&\left(P(i,j,l,k) = \sum_{u=0}^{len_i-1}\sum_{v=0}^{len_j-1}\delta(\mathrm{head}_i+u, head_j+v)Pair(l+u, k+v)\right)
\end{aligned}
\quad , \qquad \text{(S1)}
$$

where E_{env} is environment fitness score of core regions, E_{mut} is mutation score of core regions, $E_{\sec}$ is secondary structure compatibility score of core regions, E_{gap} is gap penalty score in loop regions, E_{pair} is pairwise interaction score between core regions, $w_{env}, w_{mut}, w_{\sec}, w_{gap}$ and w_{pair} are their corresponding weighting factors. *Environment*(*i*,*j*) is the score of residue *j* in environment of residue *i*. *Mutation*(*i*,*j*) is the score of residue *j* mutating to residue *i*. SS(*i*,*j*) gives the compatibility score between the predicted secondary structure of residue *j* and the secondary structure of residue *i*. Pair(*i*,*j*) is the interaction score of residue *i* and residue *j*. $\delta(i,j)=1$ if there is interaction between residue *i* and residue *j*. *G*(*i*,*j*,*k*) is the sequence alignment of the loop region of a template

(between core *i* and core *j*) to position *l* to *k* of target sequence. The gap penalty in the sequence alignment is an affine function. Gap opening is set to 10.6 and gap extension is set to 0.8. The 0-1 variables $x_{i,j}$ and $y_{i,l,j,k}$ have the following constraints:

$$x_{i,l} \in \{0,1\}, y_{i,l;j,k} \in \{0,1\}, \tag{S2}$$

$$\sum_{j \in D[i]} x_{i,j} = 1, i = 1,...,M, \tag{S3}$$

$$\sum_{k \in D[j|i,l]} y_{i,l;j,k} = x_{i,l}, \tag{S4}$$

$$\sum_{l \in D[i;j,k]} y_{i,l;j,k} = x_{j,k},. \tag{S5}$$

The energy function (S1) with constraint of Eqs. (S2) to (S5) is minimized by using integer linear programming.

Once the values of $x_{i,l}$'s are solved by inter linear programming, the corresponding alignments between query sequence to templates can be found since the alignments at the core regions of templates have no gaps and the loop regions of template are solved by dynamic programming(2).

References

1. Xu, J., Li, M., Kim, D. and Xu, Y. (2003) RAPTOR: optimal protein threading by linear programming. *J Bioinform Comput Biol*, **1**, 95-117.
2. Needleman, S.B. and Wunsch, C.D. (1970) A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J Mol Biol*, **48**, 443-453.