蛋白质在格子模型中改进的 PERM 算法*)

李小妹
(广东工业大学计算机学院　广州 510006)

摘　要　PERM 算法是当前蛋白质结构预测的格子模型优化算法中最为有效的一种算法，在该算法的基础上，我们提出了一种改进的增长算法 IPERM。该方法简化了 PERM 算法中的权重计算公式，在遇到不同类型的残基时选用不同的上下限阈值以提高算法的有效性，并根据链长的大小使用不同的网格尺寸。实验结果表明，改进的增长算法使得 HP 序列在格子模型中能更快地找到其能量最低构象。

关键词　蛋白质折叠，格子模型，PERM

Improved PERM Algorithm of Protein Folding Simulations

LI Xiao-Mei
(Faculty of Computer, Guangdong University of Technology, Guangzhou 510006)

Abstract　PERM is the most efficient aproach for protein folding problem in lattice model. In this paper we ppresent an improved growth algorithm which simplify the calculation of weight when choosing different branches and apply different upper thresholds according to different type of monomers. The experiment results show that the improved growth algorithm can find the knwon lowest ground state faster for the given HP sepquences in square lattive model.

Keywords　Protein folding, Lattice model, PERM

1　引言

在致密结构中，对所有 HP 序列实施全枚举的快速算法。虽然这些快速算法对减少计算量起到了一定的作用，但全枚举仍局限于规模较小的短链。在实际蛋白质结构预测中，需要研究较长链的 HP 序列折叠，由于计算量的限制，不可能对所有结构实施全搜索，因此必须寻求优化算法来实现长链的 HP 序列折叠搜索。

我们知道致密结构中构象数会随着序列长度的增加呈指数增加，已提出的各种模型均为有规则外形的格子模型，而实际上 HP 序列的最小能量构象虽然结构致密，但不一定具有规则的几何外形。因此在蛋白质结构预测的格子模型中需要在所有可能的构象中寻求能量最低的结构。

蛋白质的折叠问题[1,2]就是蛋白质的结构预测问题。一般常用的预测算法有两种，一种是迭代算法[3]，包括模拟退火[4]、蒙特卡罗算法[5]、遗传算法[6]和禁忌算法[7]。这种算法很容易陷入局部最小值。第二种算法就是链的增长算法，包括以核为指导的增长算法[8]和 PERM 算法[9,10]，为了改进算法效果，加入了各种启发式的结构信息。这是算法的一种设计方向，本文就是通过序列和结构的已知信息来提高算法的有效性。蛋白质结构预测的 PERM 算法和改进的增长算法是一种利用部分已折叠信息来实施预测的一种算法，其优点是预测所需时间少，效率高。但缺点是不适合搜索最低能量构象中存在远程疏水残基间形成拓扑接触对的 HP 序列。

2　HP 模型

在 HP 格子模型中，一个蛋白质的氨基酸链可表示为每个氨基酸疏水性组成的链串。蛋白质的构象用一条自回避路径来表示。蛋白质由疏水氨基酸(P)和亲水氨基酸(H)组成，一个蛋白质构像的能量为疏水氨基酸形成的拓扑接触对数目取负值。

我们希望得到能量最低的构像。从上面能量的计算模式可看出，格子模型的全局最优构象为 HH 拓扑接触对数目最多的构象，那么蛋白质折叠问题就转化为 HH 拓扑接触对数目最多的优化问题。如二维 HP 序列为 HPHPPHHPHP-PHPHHPPHPH 的能量最低构象，见图 1，其中白色方格表示亲水残基，黑色方格表示疏水残基，箭头表示开始点，实线表示链连接，虚线表示疏水残基间的拓扑接触对。

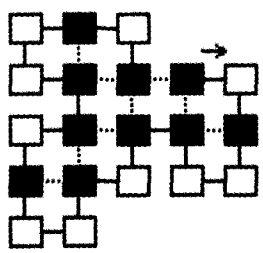

图 1　二维 HP 序列为 HPHPPHHPHPPHPHHPPHPH 的能量最低构象

3　改进的增长算法 IPERM

改进的增长算法(IPERM)是一种带权的有偏增长的 PERM 算法，该方法对每一个已形成的部分链构象赋予一个权值。在部分链构象中加入一个氨基酸时，权重就会随着增加的 HH 的拓扑接触对数而变化，PERM 算法抑制权重过低的构象增长，而高权重的部分链构象则让其在多个合法方向上增长。

由于构象具有旋转对称性，不失一般性，我们可以固定前面两个氨基酸位置。由于 HP 序列的能量最低构象具有一定

*)基金项目：校博士基金(编号：063011)。李小妹　博士，主要研究方向：生物信息学和复杂系统的演化分析。

的致密性，我们可根据 HP 序列的长度适当选定网格大小。PERM 算法中一条长为 N 的链使用的网格为 $(2N+1)\times(2N+1)$，也就是说 PERM 算法不限制网格大小。在 IPERM 中我们选取的网格大小为 $5\sqrt{N}\times 5\sqrt{N}$。则对于长为 100 的 HP 链，PERM 算法使用的网格大小为 201×201，IPERM 使用的网格大小为 50×50，通过网格规模的减小，从而也就大大减小了构象的搜索空间。

在 IPERM 算法的权重初始化阶段，首先固定前两个氨基酸位置，后面氨基酸的放置位置在相邻的格点中随机选取，但必须保证构象的合法性（即链的连接性和独占性），在增长过程中，长为 n 的部分链构象的初始化权重表示为：

$$W_1=W_2=1;W_n=W_{n-1}e^{\frac{\Delta E}{T}}\ (N\geqslant n>2)$$

其中 ΔE 表示第 n 个氨基酸放置后增加的能量，T 为温度系数，其值的选取范围为 0.25～0.35。用 Z_n 来表示算法的搜索过程中已生成的长为 n 的部分链构象权重的平均值，其初始化值等于部分链的权重初始化值，即 $Z_n=W_n(N\leqslant n\leqslant 1)$。

若已有长为 $n-1$ 的部分链构象，链的增长和 IPERM 算法权重的更新过程描述如下：

(1)第 n 个氨基酸可放置的合法位置的平均质量表示为：

$q_n=(\sum_{k=1}^{k=k_{free}}\Delta E_k)/k_{free}$，其中 f_{free} 表示第 n 个氨基酸可放置的合法位置个数，ΔE_k 表示第 n 个氨基酸放置到第 k 个合法位置时增加的能量。

(2)上下阈值 $W_n^>$ 和 $W_n^<$ 的计算，这两个阈值的计算依赖于 Z_n。

这两个阈值的计算公式分别为 $W_n^>=C^>Z_n$ 和 $W_n^<=C^<W_n^>$，其中 $C^>$ 和 $C^<$ 分别为上、下限阈值系数，也是一个可调参数，该参数值越小，则对应的上下限阈值越小，使得搜索的构象空间越大。对不同的氨基酸，我们可选用不同的下限阈值系数值，若第 n 个氨基酸为疏水氨基酸，则其值的选取的范围为 0.6～0.8，若第 n 个氨基酸为亲水氨基酸则其值的选取的范围为 0.05～0.2。

(3)若已有了一条长为 $n-1$ 的部分链构象，则可预计长为 n 的部分链构象的权重 W_n^{pred} 为：

$$W_n^{pred}=W_{n-1}K_{free}$$

(4)部分链的抑制和繁殖根据 $W_n^>$，$W_n^<$ 和 W_n^{pred} 值来确定，若长为 n 的部分链构象权重的期望值 W_n^{pred} 小于 $W_n^<$，则该链以 0.5 的概率丢弃该格局，以 0.5 的概率保留，并在合法分支中等概率选取一个方向生长，其权重交加倍；若长为 n 的部分链构象权重的期望值 W_n^{pred} 大于 $W_n^<$ 且小于 $W_n^>$，则链任选其中一个合法位置增长；若期望值 W_n^{pred} 大于 $W_n^>$，则链任选其中 k 个合法位置增长，其中 $k=\min\{k_{free},ceiling(W_n^{pred}/W_n^>)\}$，而增长的部分链构象权重乘以 k/k_{free}。

(5)实施链增长并根据链初始化阶段给出的公式计算部分链构象权重，并将该权重乘以(4)中给出的系数因子，同时更新 Z_n 值。

改进的 PERM 算法在原来算法的基础上简化了权重的计算方式，在生长的过程中灵活地选取上下门限值，保证了可能出现最优能量的区域枝叶生长得更茂森。在实际的算法搜索过程中，灵活控制上下门限，或使用更为精细的控制策略，从而使得算法得到的结果更为有效。

4 实验结果

我们应用一组数据来测试改进的增长算法，表 1 中的这些三维 HP 序列为各文献中常常用来作为测试的公认为较难的样例。序列中的 H_i、P_i 以及 $()_i$ 分别表示串中连续的 H、P 和()内 HP 串的个数。

表 1 三维格子模型中蛋白质折叠问题的 HP 序列及其最优或已知的最低的能量值

No	Length	E	Protein sequence
			3D HP sequence
1	48	−32	$HPH_2P_2H_4PH_3P_2H_2P_2HPH_3PHPH_2P_2H_2P_3HP_8H_2$
2	48	−34	$H_4PH_2PH_5P_2HP_2H_2P_2HP_6HP_2HP_3HP_2H_2P_2H_3PH$
3	48	−34	$PHPH_2PH_6P_2HPHP_2HPH_2(PH)_2P_3H(P_2H_2)_2P_2HPHP_2HP$
4	48	−33	$PHPH_2P_2HPH_3P_2H_2PH_2P_3H_5P_2HPH_2(PH)_2P_4HP_2(HP)_2$
5	48	−32	$P_2HP_3HPH_4P_2H_4PH_2PH_3P_2(HP)_2HP_2HP_6H_2PH_2PH$
6	48	−32	$H_3P_3H_2PH(PH_2)_3PHP_7HPHP_2HP_3HP_2H_6PH$
7	48	−32	$PHP_4HPH_3PHPH_4PH_2PH_2P_3HPHP_3H_3(P_2H_2)_2P_3H$
8	48	−31	$PH_2PH_3PH_4P_2H_3P_6HPH_2P_2H_2PHP_3H_2(PH)_2PH_2P_3$
9	48	−34	$(PH)_2P_4(HP)_2HP_2HPH_6P_2H_3PHP_2HPH_2P_2HPH_3P_4H$
10	48	−33	$PH_2P_6H_2P_3H_3PHP_2HPH_2(P_2H)_2P_2H_2P_2H_7P_2H_2$

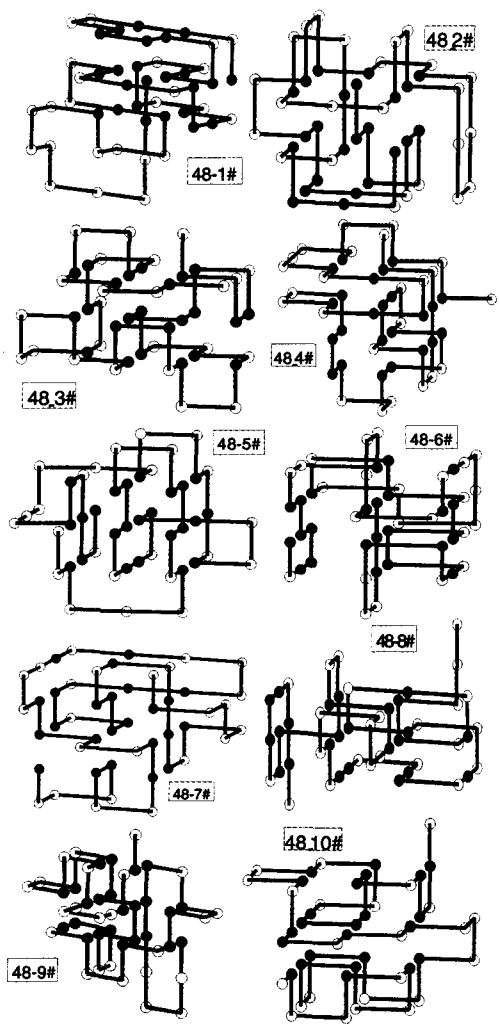

图 2 在三维网格模型中利用改进的增长算法得到的长为 48 的 10 条链的最低能量构象

我们对10条三维HP序列进行了测试，得到的其中一个最优构象见图2，并与PERM算法的实验结果进行了比较，见表2。PERM算法给出的时间是在2.4 GHz Pentium IV微处理器上每条序列运行50～100次后得到的最好结果，我们的实验是在SANYO 1.7G微处理器上每条序列运行5次后得到的最好结果。表3中黑体字显示的能量为当前所知的最低能量值。从表3可以看出利用IPERM算法平均0.4～80秒时间可找到一条能量最低构象。通过比较可以看出PERM算法较以前的优化算法得到的结果更优。而改进的增长算法除了5号和6号链，其他链均能在比PERM算法更短的时间内找到最小能量构象。

表2　三维方形网格模型中10条链的各种算法比较

No/E	HZ	CHCC	CG	CI	ACO	PERM	IPERM
1/−32	−31(4 hrs)	−32(30 min)	−32(9.4 min)	−32	−32(30min)	−32(0.2 min)	−32(0.046 min)
2/−34	−32(18 hrs)	−34(2.3 min)	−34(35 min)	−33	−34(420 min)	−34(0.6 min)	−34(0.081 min)
3/−34	−31(23 hrs)	−34(30 min)	−34(62 min)	−32	−34(120 min)	−34(0.2 min)	−34(0.156 min)
4/−33	−30(19 days)	−33(71 min)	−33(29 min)	−32	−33(300 min)	−33(3 min)	−33(0.033 min)
5/−32	−30(1.3 days)	−32(32 min)	−32(12 min)	−32	−32(15 min)	−32(1 min)	−32(1.338 min)
6/−32	−29(2.1 days)	−32(80 min)	−32(460 min)	−30	−32(720 min)	−32(0.2 min)	−32(0.759 min)
7/−32	−29(2.5 days)	−32(110 min)	−32(64 min)	−30	−32(720 min)	−32(1 min)	−32(0.006 min)
8/−31	−29(4 hrs)	−31(530 min)	−31(38 min)	−30	−31(120 min)	−31(0.6 min)	−31(0.296 min)
9/−34	−31(4.5 hrs)	−34(8.3 min)	−33	−32	−34(450min)	−34 (7 min)	−34 (0.654 min)
10/−33	−33(1.1 hr)	−33(4.8 min)	−33(1.1min)	−32	−33(60 min)	−33(0.01 min)	−33(0.078 min)

最后我们也对19条二维HP三角网格模型序列进行了测试，得到的其中一个最优构象见图3。PERM算法没有给出三角网格的实验结果，但是与已有的实验结果相比，改进的PERM算法能得到能量更低的构象，特别是对14、17和18号链而言，其他算法只能给出次优能量构象，对19号链只能给出能量为−26的构象。而改进的PERM算法，对全部19条链均得到了能量最优构象。

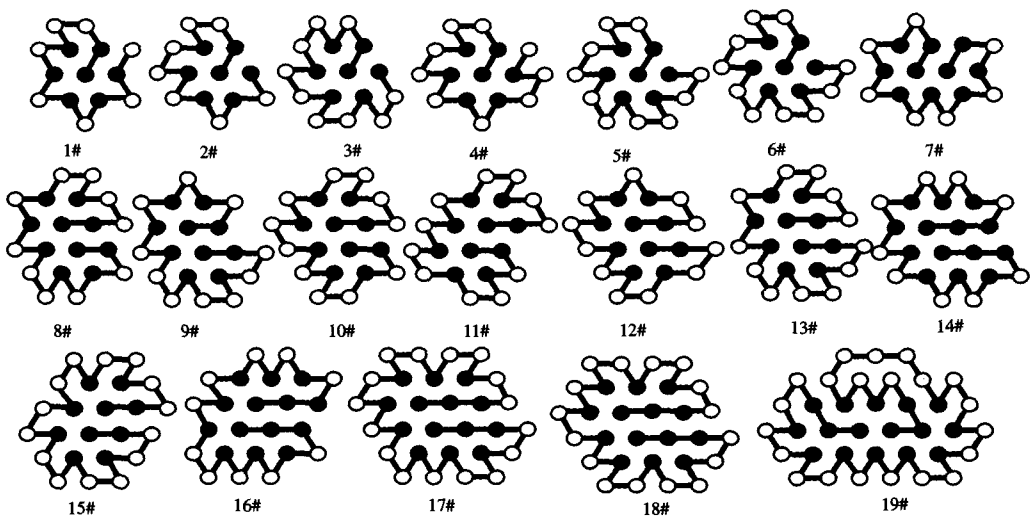

图3　在2D三角网格模型中利用改进的增长算法得到的19条链的最低能量构象

结论　本文提出了二维和三维格子模型的改进的增长算法，通过实验发现，PERM算法和改进的增长算法在当前国际上公认的最难的几个算例的计算上均达到了最优解，且改进的增长算法较PERM算法可更快地找到能量最低构象(除了一条二维HP链和两条三维HP链)。这些结果表明对PERM算法的改进是明显有效的。

参考文献

1　Berger B, Leighton T. Protein folding in the hydrophobic-hydrophilic model is NP complete. J. Comput. Biol, 1998,5: 27～40

2　Paterson M, Przytycka T. On the complexity of string folding, Discrete. Appl. Math., 1996,71: 217～230

3　Kirkpatrick S, Gelatt C D Jr, Vecchi M P. Optimization by simulated annealing. Science, 1983,220:671

4　Unger R, Moult J. Genetic algorithms for protein folding simulations. J. Mol. Biol.,1993, 231:75

5　Konig R, Dandekar T. Improving genetic algorithms for protein folding simulations by systematic crossover. Biosystems, 1999, 50: 17～25

6　Liang F. Evolutionary Monte Carlo for protein folding simulations. J. Chem. Phys., 2001, 115:3374

7　Jiang Tianzi. Protein folding simulations of the hydrophobic-hydrophilic model by combining tabu search with genetic algorithms. J. Chem. Phys., 2003, 119:4592

8　Beutler T C, Dill K A. A fast conformational search strategy for finding low energy structures of model proteins. Protein Science, 1996, 5:2037～2043

9　Zhang J L, Liu J S. A new sequential importance sampling method and its application to the two-dimensional Hydrophobic- Hydrophilic model. J. Chem. Phys., 2002, 117(7): 3492～3498

10　Grassberger P. The pruned-enriched Rosenbluth method: simulations of Theta polymers of chain length up to 1000000. Phys. Rev. E,1997, 56(3): 3682～3693