基于 Toy 模型蛋白质折叠预测的多种群微粒群优化算法研究*)

张晓龙[1] 李婷婷[1] 芦 进[2]

(武汉科技大学计算机学院 武汉 430065)[1] (华中科技大学 CAD 中心 武汉 430074)[2]

摘 要 基于 Toy 模型的蛋白质折叠结构预测问题是一个典型的 NP 问题。提出了多种群微粒群优化算法用于计算蛋白质能量最小值。该算法采用了一种新的算法结构,在该结构中,每一代的种群被分为精英子种群、开采子种群和勘探子种群三部分,通过改善种群的局部开采能力和全局勘探能力来提高算法的性能。分别采用 Fibonacci 蛋白质测试序列和真实蛋白质序列进行了折叠结构预测的仿真实验。实验结果表明该算法能够更精确地进行蛋白质折叠结构预测,为生物科学研究提供了一条有效途径。

关键词 蛋白质折叠,Toy 模型,多种群微粒群优化算法(MPSO)

Study of Multi-PSO Algorithm for Protein Folding Prediction Problem of Toy Model

ZHANG Xiao-long[1] LI Ting-ting[1] LU Jin[2]

(School of Computer Science and Technology, Wuhan University of Science and Technology, Wuhan 430065, China)[1]

(CAD Center, Huazhong University of Science and Technology, Wuhan 430074, China)[2]

Abstract Protein folding prediction problem with Toy model is a classical NP problem. A multi particle swarm optimization (MPSO) is proposed and applied successfully to protein folding prediction. MPSO introduces a new architecture that is characterized by balancing exploitation capability and exploration capability of particle swarm optimization (PSO). In the architecture, the population in each generation consists of three parts: an elitist part, an exploitative part, and an explorative part. With enhance of the global search and local search ability, MPSO can be effectively used for protein folding prediction. The algorithm has been tested in the two-dimensional Toy model for several Fibonacci protein sequences and real protein sequences. The ground state energies predicted are lower than those reported in the literatures and show that MPSO is correct and effective.

Keywords Protein folding, Toy model, Multi particle swarm optimization (MPSO)

1 前言

生物信息学是一门运用数学、计算机科学和生物学的各种工具来阐明和理解大量数据所包含的生物学意义的交叉科学。狭义上,也可称作计算生物学。蛋白质工程是生物信息学的重要研究对象之一。对蛋白质结构预测问题的求解是后基因时代蛋白质工程的最重要的课题之一,其根本目的是要将天然存在的蛋白质按照人类的设想进行改造,或根据需要设计出具有某种特殊功能的非天然的新蛋白质,而这种改造和设计的重要基础之一是蛋白质折叠结构预测。因为蛋白质折叠的形状在很大程度上决定其生物功能,所以对蛋白质折叠结构的预测和研究在蛋白质工程中有着极其重要的意义。传统的蛋白质结构测定方法有很多,如 X 射线晶体衍射方法,但其方法不但费时而且在技术上受到限制;还有核磁共振技术,而它对蛋白质结构测定的速度比较缓慢,只能限于较短蛋白质序列的结构的测定。因而,利用计算机的高效计算能力来预测蛋白质结构成为一个研究热点。

蛋白质折叠结构预测是指由一个给定的蛋白质序列预测出蛋白质的天然结构。早在 1973 年 Anfinsen[1] 在《Science》杂志上提出了"蛋白质的天然构象是自由能最低的构象"的理论,该理论奠定了蛋白质折叠预测的理论基础,即通过计算最小能量预测蛋白质结构的热力学理论基础。从数学角度上看,可以归结为一个全局优化问题。迄今为止,对蛋白质结构预测问题已提出了一些简化模型。其中,一种是 Dill 等[2] 提出的 HP 格点模型(HP Lattice Model),但该模型仅考虑了蛋白质残基之间的疏水性,忽略了残基之间的亲水性,另外相邻残基之间的夹角只能是直角或平角;另一种是由 Stillinger 等[3] 提出的 Toy 模型(AB Off-Lattice Model),该模型同时考虑了蛋白质残基之间的疏水性和亲水性,而且相邻残基之间的夹角是任意的。研究表明,Toy 模型比 HP 模型更接近真实蛋白质。因此,本文选用 Toy 模型作为蛋白质折叠预测的研究对象。Toy 模型的势能函数是高度非线性的多变量函数,且具有极多的局部极小点。文献[4]提到,粗略估计,n 个残基的蛋白质能量表面存在 10^n 个局部极小点,求得势能函数的最小值是结构预测问题的目标。所以,基于 Toy 模型的蛋白质折叠结构预测问题是一个典型 NP 问题。寻找一种有效的全局优化算法是求解结构预测问题的关键。

目前,已经有许多启发式算法应用到 Toy 模型中进行结构预测。文献[5]是用 Monte-Carlo 模拟方法来进行蛋白质折叠预测。它的优点在于基本概念简单、易于实现,但随着模

*)基金项目:国家自然科学基金(No. 60674115),教育部回国人员科研启动基金(2005—2007)。**张晓龙** 教授,博士生导师,主要研究方向为机器学习、数据挖掘与生物信息学等;**李婷婷** 硕士生,主要研究方向为机器学习与生物信息学;**芦 进** 硕士生,主要研究方向为群体智能和智能计算。

拟次数逐渐增多，其计算分析效率较低，对于有较高复杂度的确定性分析过程，这一缺点尤其明显。文献[6]将遗传算法(GA)应用于此问题。GA 与其他传统搜索方法相比具有更强的鲁棒性，良好的全局搜索能力，减少了陷于局部最优解的风险，但同时也具有局部开采能力不足的缺点。文献[7]结合模拟退火算法和遗传算法提出了 GAA 算法，虽然该算法在性能上有一定的改进，但是仍然没有从根本上改变 GA 的局部开采能力较弱的问题。

1995 年，Kennedy 和 Eberhart 等[8]提出了局部开采能力很强的微粒群优化算法(Particle Swarm Optimization，PSO)。相对其它进化算法，PSO 局部收敛速度快，连续型数值求解效率高。文献[9]已将标准 PSO 算法应用到 Toy 模型中进行蛋白质折叠预测。当解空间维数较低时，局部搜索效率很高。然而当解空间维数变大后，全局搜索能力稍显不足，容易陷入局部最优。于是，为了提高预测速度和精度，针对 Toy 模型中势能函数多变量多极值的特点和 PSO 算法先天性的不足，本文将一种新的算法结构引入标准 PSO 算法中，提出了多种群微粒群优化算法(MPSO)。

本文首先简要地介绍了蛋白质结构预测中的 Toy 模型，简略地介绍了标准 PSO 算法模型；然后，详细描述了 MPSO 的改进策略和方法，并给出 MPSO 的结构图。最后，对 Fibonacci 蛋白质测试序列和真实蛋白质序列进行仿真实验并对实验结果进行分析。

2 Toy 模型的简单介绍

1993 年，Stillinger[3]提出了 Toy 模型，它将实际的二十种氨基酸根据疏水性和亲水性分为两类，分别由 A (hydrophobic or non-polar)和 B (hydrophilic or polar)表示，在二维平面上用单位长度的键把 A 和 B 连成一个非定向的线性链。任何一个由 n 个残基组成的蛋白质序列，对应 $n-2$ 个角度 $\theta_2,\cdots,\theta_i,\cdots\theta_{n-1}$，如图 1 所示，其中 θ_i 的取值范围为$[-\pi,\pi]$，当 $\theta_i=0$ 时，表示相邻的三个氨基酸在一条直线上。

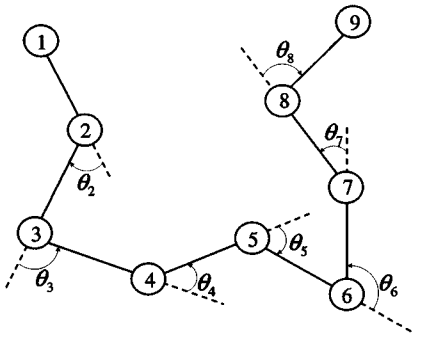

图 1　多个连续的残基形成的蛋白质序列在二维空间中的折叠表示

Toy 模型任何一个蛋白质序列的能量是由两部分组成：主链能量(V_1)和任意两个不相邻的残基之间的能量(V_2)。前者与残基的极性无关，只与折叠角度相关；后者与任意两个不相邻的残基的极性和相隔的距离相关。将残基用一组二进制变量 $\xi_1\cdots\xi_i$ 进行编码，如果第 i 个残基为 A，则 $\xi_i=1$；如果第 i 个残基为 B，则 $\xi_i=-1$。一个长度为 n 的蛋白质序列的能量势函数定义，如式(1)：

$$\Phi=\sum_{i=2}^{n-1}V_1(\theta_i)+\sum_{i=1}^{n-2}\sum_{j=i+2}^{n}V_2(r_{ij},\xi_i,\xi_j) \tag{1}$$

距离 r_{ij} 记作键角函数，如式(2)：

$$r_{ij}=\{[1+\sum_{k=i+1}^{j-1}\cos\sum_{l=i+1}^{k}\theta_l]^2+[\sum_{k=i+1}^{j-1}\sin\sum_{l=i+1}^{k}\theta_l]^2\}^{\frac{1}{2}} \tag{2}$$

V_1 是一个关于 θ_i 的简单三角函数，如式(3)：

$$V_1(\theta_i)=\frac{1}{4}(1-\cos\theta_i) \tag{3}$$

V_2 的函数表达式如式(4)：

$$V_2(r_{ij},\xi_i,\xi_j)=4(r_{ij}^{-12}-C(\xi_i,\xi_j)*r_{ij}^{-6}) \tag{4}$$

其中，系数 $C(\xi_i,\xi_j)$ 为：

$$C(\xi_i,\xi_j)=\frac{1}{8}(1+\xi_i+\xi_j+\xi_i*\xi_j) \tag{5}$$

从式(5)可以看出，当 AA 相邻时，$C(\xi_i,\xi_j)=1$；BB 相邻时，$C(\xi_i,\xi_j)=0.5$；AB 相邻时，$C(\xi_i,\xi_j)=-0.5$。表明了两个疏水残基之间有很强的引力，两个亲水残基之间有轻微的引力，疏水残基和亲水残基之间则有轻微的斥力，这在一定程度上能够真实地反映出真实蛋白质的性质。

3 多种群微粒群优化算法

PSO 算法是一种演化计算算法，其基本思想来源于对鸟群简化社会模型的研究及行为模拟，与其它的演化算法相比，PSO 具有简单、容易实现、搜索速度快的特点，同时又具有深刻的智能背景，所以从出现至今已经在很多领域有广泛的应用，而且在性能上有很大程度的改进。

3.1 标准微粒群优化算法

在标准 PSO 算法模型中，将每个个体看作寻优空间中的一个没有质量没有体积的微粒，在搜索空间中以一定的速度飞行，并根据个体自身的飞行经验以及同伴的飞行经验对自身的飞行速度进行动态调整，即每个个体通过统计迭代过程中自身的最优值和种群的最优值来不断地修正自身的前进方向和速度大小，从而形成寻优的正反馈机制，并继续搜索，最终寻找到问题的最优解。在连续空间中，第 i 个微粒同时受到自身历史最优位置(p_{ibest})和种群最优位置(p_{gbest})的吸引，通过迭代公式(6)和(7)，直到满足停止条件。

$$v_i=\omega*v_i+c_1*rand_1()*(p_{ibest}-x_i)+c_2*rand_2()*(p_{gbest}-x_i) \tag{6}$$

$$x_i=x_i+v_i \tag{7}$$

其中，ω 是惯性权重系数；$rand_1()$和 $rand_2()$是$[0,1]$之间的随机数；c_1 和 c_2 是学习因子，通常 $c_1=c_2=2.05$。

3.2 多种群微粒群优化算法原理

标准 PSO 算法模型是存在缺陷的，目前已经提出了许多的改进策略。主要包括三个方面：(1)对算法的参数设定和调整，如文献[10]分别针对 PSO 模型不同参数对收敛性能的影响进行讨论；(2)与其它智能算法的结合，如文献[11]提出的免疫微粒群算法，文献[12]提出的遗传微粒群算法等；(3)对算法的总体结构和组织模式的改进，如文献[13]提出的一种新的多种群遗传算法结构，并进一步证明了其收敛性。因为算法结构更具有一般推广性，可以直接应用于其他群集智能算法，所以对算法结构的改进受到更多的关注，本文的主要改进也是针对 PSO 算法的总体结构而提出的。

一种有效的算法应该同时具备局部开采能力和全局勘探能力。局部开采能力是引导算法朝着问题的解空间中可能最优区域进行搜索的能力；全局勘探能力是指引导算法在整个解空间中不断搜索，提高解的多样性标准 PSO 算法存在的问题是：开采能力较强，勘探能力不足。在进化后期，粒子将追随当前已知的最优位置，并且吸引其它粒子进入该区域搜索，增加了搜索到局部最优值的概率。这种进化策略利用有效信

息提高了局部开采能力，但同时减弱了种群的全局勘探能力，很难再寻找到更优解。因此，在求解复杂问题时，如何在开采和勘探之间进行有效的权衡是 PSO 算法能否获得高效的关键。基于这个认识，本文对标准 PSO 算法结构进行改进，使其能够有效地跳出局部最优解，搜索到更优解。

本文在标准 PSO 算法的框架下，基于不同分工种群建模，提出了一种新的 PSO 算法结构。在该结构中，$t+1$ 时刻的种群 $P(t+1)$ 由精英种群 $P_1(t)$，开采种群 $P_2(t)$，勘探种群 $P_3(t)$ 三部分进化而来。首先将种群 $P(t)$ 中的 n 个粒子按其适应值由低到高排序并编号，精英种群由群体中适应值较低的 $1\sim n_1$ 号粒子组成，开采种群由 $n_1+1\sim n_1+n_2$ 号粒子组成，勘探种群由适应值较高的 $n_1+n_2+1\sim n_1+n_2+n_3$ 号粒子组成（$n_1+n_2+n_3=n$）。当种群最优粒子位置在不断变化时，精英种群通过对适应值低的 n_1 个粒子进行微调获得，当种群最优粒子位置连续无变化或者连续变化非常小时，精英种群由变异策略对适应值低的 n_1 个粒子进行变异获得；开采种群在整个进化过程中选择上一代种群中适应值较低 n_2 个粒子通过标准 PSO 算法进化获得；勘探种群是从上一代种群中适应值高的 n_3 个粒子通过勘探策略获得。其中，开采种群的数量体现算法对开采能力的重视程度，而精英种群和勘探种群的数量体现算法对勘探能力的重视程度。

3.3 MPSO 算法的改进策略

MPSO 算法相对标准 PSO 算法主要有三个改进策略。分别为微调策略、变异策略和勘探策略，下面对它们进行详细说明。

3.3.1 微调策略

由公式(7)可知，每个粒子的下一时刻的位置是由当前位置和当前速度共同决定，因此可能出现粒子位置已经趋近于全局最优位置，但是由于公式(6)中 $\omega * v_i$ 这一分量较大，引起更新位置矢量 x_i 的速度矢量 v_i 较大，有可能越过全局最优位置，降低求解的效率。当解空间维度较大时，可能出现当前最优粒子某些维的位置已经达到更优解相应维度的位置，因此通过公式(7)就很难使粒子达到全局最优位置。为提高算法的精度，精英种群 $P_1(t)$ 根据公式(8)进行局部微调。

$$x_i=\begin{cases}x_i+0.001*f(\alpha)*rand() & T(t)\leqslant T_0\\ x_i+G(v_i) & T(t)>T_0\end{cases} \tag{8}$$

其中，$rand()$ 和 α 为均匀分布在[0,1]间的随机数，系数 $f(\alpha)$ 如公式(9)：

$$f(\alpha)=\begin{cases}-1 & \alpha<0.5\\ 1 & \alpha\geqslant 0.5\end{cases} \tag{9}$$

$T(t)$ 表示迭代次数计数器，如式(10)，T_0 表示 $T(t)$ 的阈值，初始状态 $T(0)=0$，

$$T(t)=\begin{cases}T(t-1)+1 & \varepsilon>\varepsilon_0\\ 0 & \varepsilon\leqslant\varepsilon_0\end{cases} \tag{10}$$

ε 表示相邻两代最低能量值之差，如式(11)：

$$\varepsilon=F_{best}(t+1)-F_{best}(t) \tag{11}$$

$F_{best}(t)$ 表示 t 时刻的最优适应值，ε_0 表示 ε 的阈值。

在公式(8)中，当 $T(t)\leqslant T_0$ 时，表示当种群历史最优位置在不断变化时，精英种群中每个粒子是在其附近很小的范围内进行扰动，提高寻找较优解的速度，避免粒子越过更优解的情况出现，在一定程度上了提高算法的效率和精度；当 $T(t)>T_0$ 时，表示当种群历史最优位置连续无变化或者变化非常小时采用的变异策略。

3.3.2 变异策略

公式(6)中粒子的当前速度由三个分量决定：粒子上一时刻速度 v_i，粒子历史最优位置 p_{ibest} 和种群历史最优位置 p_{gbest}。迭代后期，PSO 算法收敛速度较快，粒子逐渐向种群历史最优位置 p_{gbest} 聚集，粒子的速度 v_i 将会逐渐变小，所有的粒子将逐渐逼近 p_{gbest} 并且停止运动。实际上，PSO 算法并不能保证收敛到全局最优位置，而仅仅是收敛到种群的历史最优位置 p_{gbest}，算法可能出现早熟收敛。此时采用变异策略，有效地使部分粒子跳出局部搜索区域，如图 2 所示。种群历史最优位置 p_{gbest} 经过高斯变异变更为新的位置 p_{gbest}（图 2-a），通过改变公式(6)一个分量 p_{gbest} 来改变粒子的前进方向和大小，从而让粒子进入其它区域进行搜索（图 2-b）。由于仍然保留了粒子上一时刻速度 v_i 和粒子当前极值 p_{ibest} 这两个分量，使得新的搜索区域有一定的指导性，接近全局最优解的周围区域，提高搜索的速度。高斯变异函数如下：

$$p_{gbest}=p_{gbest}*(1+Gaussian(\eta*\sigma)) \tag{12}$$

其中，σ 为满足标准高斯分布的随机数，η 的初始值为 1.0，每隔 20 代 $\eta=\beta*\eta$，β 为[0.01,0.9]之间的随机数。公式(8)中的 $G(v_i)$ 即可以看作为：

$$G(v_i)=\omega*v_i+c_1*rand_1()*(p_{ibest}-x_i)+c_2*rand_2()*(p'_{gbest}-x_i) \tag{13}$$

变异策略提高种群的多样性，从而为寻找更优解提供一个较为合适的群体多样性保持策略。

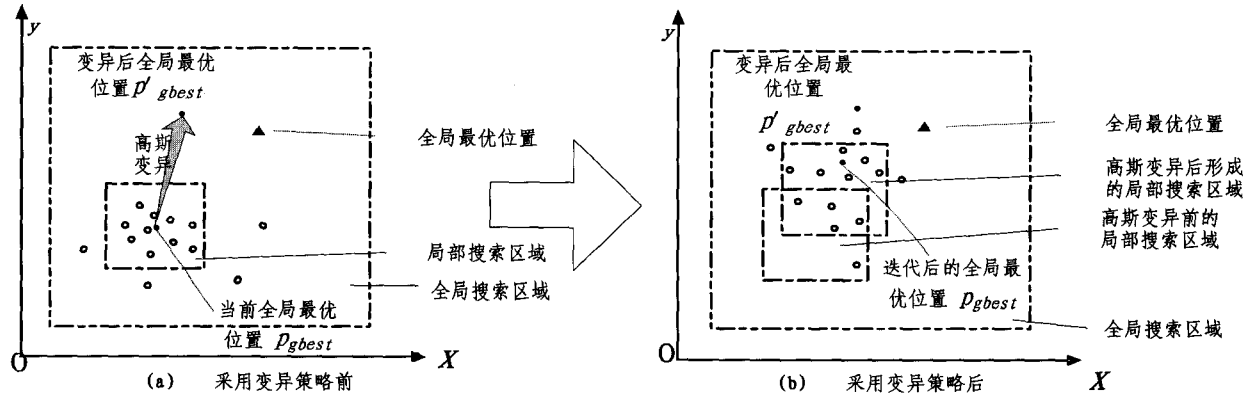

图 2 变异策略原理示意图

3.3.3 勘探策略

算法容易陷入局部最优解的主要原因是在进化后期中由于所有粒子都集中在某个局部区域内无法跳出，因此，将上一代种群中适应值高的粒子用随机搜索产生的粒子代替，加强

种群的多样性。这里的随机搜索是有指导的随机搜索，并不是盲目的随机搜索。本文定义了如下函数：

$$x_i = x_{gbest} + R * rand() * f(\alpha) \tag{14}$$

其中，x_{gbest} 表示当前种群最优位置，R 表示随机勘探半径，$rand()$ 是[−1,1]之间的随机数，$f(\alpha)$ 与公式(9)相同。公式(14)表示以上一时刻的种群历史最优位置为中心，以 R 为半径进行勘探，R 是一个动态调整函数：

$$R=\begin{cases} R*(1+\delta) & T(t)>T_0 \\ R & T(t)\leqslant T_0 \end{cases} \tag{15}$$

引入控制参数 $T(t)$ 和 T_0 与公式(10)中表示的含义相同，目的是判断勘探范围是否扩大；δ 表示勘探范围的扩大率。种群历史最优粒子位置在不断变化时，随机勘探的范围不变，以此提高搜索的速度；当种群最优粒子的位置连续无变化或者连续变化非常小时，将不断扩大随机勘探的范围直至整个解空间，进行大范围的勘探，能够更好地搜索到更优解。

3.4 MPSO 算法的算法结构

根据 3.2 节所述的 MPSO 算法的原理和 3.3 节所述的改进策略，面向 Toy 模型的 MPSO 算法的寻优过程是：随机产生一个包含 n 个粒子的群体 P，并根据 Toy 模型中的势能函数计算群体中每个粒子的能量值，然后按照升序排列，同时保存最低能量值和具有最低能量值的粒子。按能量值由低到高将种群划分为三部分：精英种群、开采种群、勘探种群。在每次的进化过程中按照微调策略，变异策略和勘探策略进行全局和局部搜索，再对群体进行升序排序，并记录最低能量值和对应粒子的位置，直到算法结束，如图 3 所示。

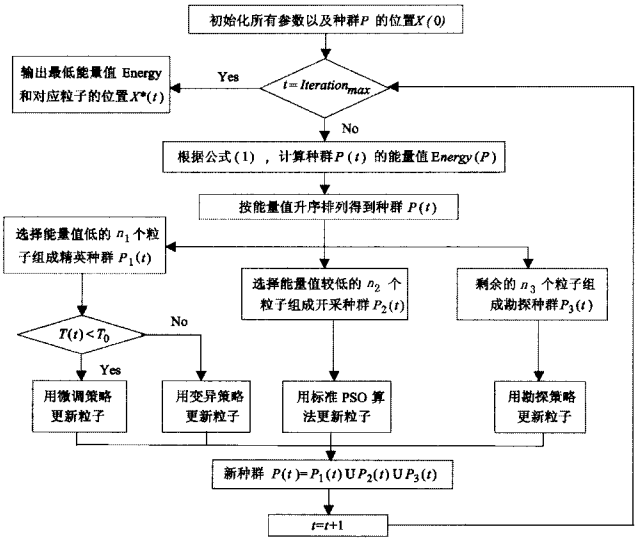

图 3 MPSO 算法的流程图

4 实验结果

为了评价和比较算法的性能，本文基于 Toy 模型采用若干人工和真实蛋白质序列进行了折叠结构预测的实验，应用 MPSO 算法求解蛋白质序列的能量最低值。仿真实验条件：Intel IA Sever(Sever)，Intel Xeon 3.06GHZ(CPU)，1GB DRR2/667MHZ (RAM)，Window Sever 2003(Operation System)，VC++6.0，Matlab7.0。

4.1 人工蛋白质序列仿真实验

首先选用标准 Fibonacci 测试序列进行仿真实验。Fibonacci 序列为：$S_0=A$，$S_1=B$，$S_{i+1}=S_i-1*S_i$，其中 * 是连接算子。则 $S_2=AB$，$S_3=BAB$，$S_4=ABBAB$ 等，S_i 由 Fibonacci 序列给出，在序列中疏水性残基 A 被孤立，亲水性残基 B 成对或单个被孤立。本实验首先采用文献[4]中较短的 Fibonacci 序列作为测试序列，判断本算法是否得到最低能量值。再对采用文献中长度为 13，21，34 和 55 的蛋白质序列作为测试序列，并与之比对。在预测复杂度方面，长度为 34，55 的测试蛋白质序列接近于真实的蛋白质序列，具有结构复杂、目标势能函数中极小值数量巨大等特点。这里的较短序列指的是序列长度在 3～5 的序列，而长序列指的是长度为 13～55 的序列。

对较短的蛋白质测试序列进行预测时，迭代次数 $Iteration_{max}=50$；对长的测试序列时，迭代次数 $Iteration_{max}=5000$。其它参数初始化如下：种群个数 $n=2000$，在公式(6)中，惯性权重系数 ω 是从 0.9 到 0.4 线性递减，保留因子 $r_1=100$，开采因子 $r_2=1300$，勘探因子 $r_3=600$，$T_0=50$，$\varepsilon_0=-0.01$，$\delta=0.01$，$R=0.5$。

表 1 列出了对短序列的测试结果，与文献[4]中的结果相符，表明通过本算法可以迅速得到蛋白质序列的最低能量值，说明本算法是可行有效的。

表 1 短 Fibonacci 测试序列的最低能量值

Sequence list	E	Sequence list	E
AAA	−0.65821	AAAAA	−2.84828
AAB	0.03223	AAAAB	−1.58944
ABA	−0.65821	AAABA	−2.44493
ABB	0.03223	AAABB	−0.54688
BAB	−0.03027	AABAA	−2.53170
BBB	−0.03027	AABAB	−1.34774
		AABBA	−0.92662
		AABBB	0.04017
AAAA	−1.67633	ABAAB	−1.37647
AAAB	−0.58527	ABABA	−2.22020
AABA	−1.45098	ABABB	−0.61680
AABB	0.06720	ABBAB	−0.00565
ABAB	−0.64938	ABBBA	−0.39804
ABBA	−0.03617	ABBBB	−0.06596
ABBB	0.00470	BAAAB	−0.52108
BAAB	0.06172	BAABB	0.09621
BABB	−0.00078	BABAB	−0.64803
BBBB	−0.13974	BABBB	−0.18266
		BBABB	−0.24020
		BBBBB-	−0.45266

表 2 长 Fibonacci 测试序列的结果比较

Length	Sequence list	E_{HTMC}	E_{PERM}	E_{nPERM}	E_{MPSO}
13	ABBABBABABBAB	−3.2235	−3.2167	−3.2939	−3.2941
21	BABABBABABBAB BABABBAB	−5.2281	−5.7501	−6.1976	−6.1977
34	ABBABBABABBABBABAB BABABBABBABABBAB	−8.9749	−9.2195	−10.7001	−10.7036
55	BABABBABABBABBABABB ABABBABBABABBABBA BBABABBABBABABBAB	−14.4089	−14.9050	−18.5154	−18.6701

表 2 列出了不同算法对长度为 13，21，34 和 55 的 Fibonacci 序列进行预测得到的最低能量值。E_{HTMC} 是文献[5]得到的最低能量值，E_{PERM} 是通过 PREM[14] 的方法得到的最低能量值，E_{nPERM} 是先用 PREM 方法再用共扼梯度法得到的最低能量值，即首先采用 PREM 方法得到的最低能量值和最优粒子，并将最优粒子作为新种群的初始化的基础，接着采用共扼梯度法进行再次预测，得到新的最低能量值和最优粒子。

因此，用 PREM 方法和共扼梯度法得到的最低能量值 E_{nPERM} 被认为是最低能量值。E_{MPSO} 是本文提出的 MPSO 算法得到的最低能量值。

由表 2 可见，对于长度为 21，34，55 的序列，用本文算法得到的最低能量值均明显优于用 HTML 方法和用 PREM 方法得到的最低能量值；对于长度为 13 的序列，本文的最低能量值略有提高。对于所有序列，用本文算法得到的最低能量值与最低能量值（先用 PREM 方法再用共扼梯度法得到的最优能量值）近似，而是对长度为 55 的序列有较好的改善。

从 MPSO 算法得到的蛋白质序列的构像（如图 4）中发现：在长度为 13 的蛋白质序列的构像中，A 类残基（疏水性残基）形成了一个紧密的疏水核，被 B 类残基（亲水性残基）包围，完全符合蛋白质的特性。在长度分别为 21，34，55 的蛋白质序列的构像中，A 类残基形成多个束，基本上是被 B 类残基包围，较为符合真实蛋白质的特性。

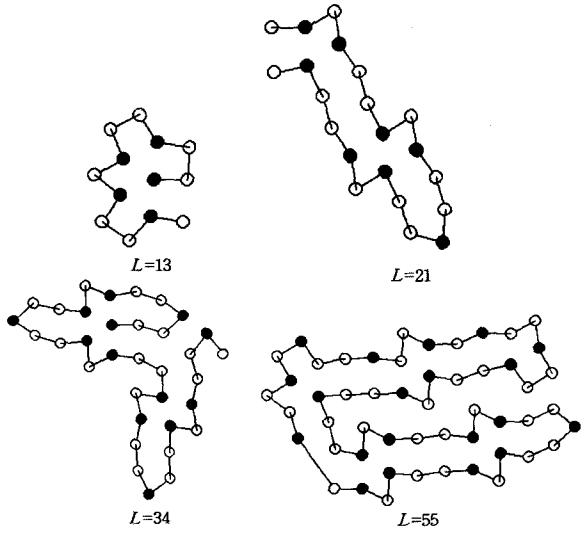

图 4　测试蛋白质序列二维最低能量构像（黑色球和白色球分别代表疏水性残基 A 和亲水性残基 B）

为了更好地说明 MPSO 算法用于蛋白质折叠预测的性能和特点，图 5 表示长度为 55 的蛋白质序列的最低能量值在迭代过程中的变化曲线图，图 6 表示在相应的迭代过程中随机勘探半径 R 的变化曲线图。

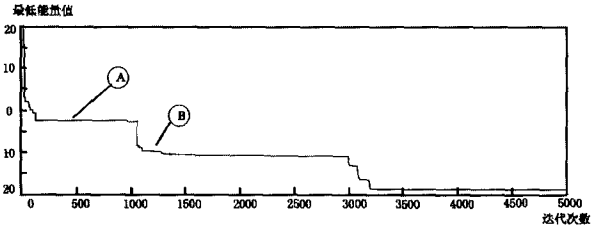

图 5　L_{55} 的最低能量值曲线

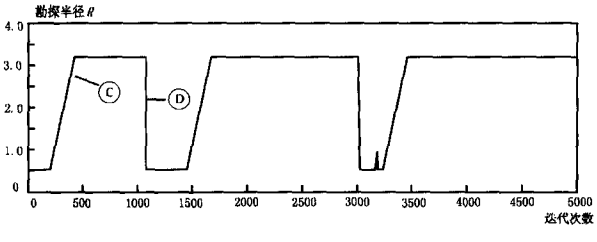

图 6　L_{55} 的随机勘探半径 R 曲线

由图 5 可见，在迭代初期，算法的收敛速度很快，进入局部区域搜索最优解（A 区），当预测蛋白质序列的最低能量值长时间无变化或变化非常小时，勘探种群通过扩大随机勘探半径进行搜索，对应图 6 中的迭代过程（C 区）中的随机勘探半径逐渐增大，直至随机勘探区域扩大到整个解空间。迭代多次后，粒子跳出局部最优解（B 区）寻找到更优解。随之，随机勘探半径回到初始值（D 区）。实验结果表明，MPSO 算法能够有效逃离各个阶段的局部最优解，并能在保持高精度的条件下快速收敛到更优解。

4.2　真实蛋白质序列仿真实验

从 PDB 库（http://www.rcsb.org/pdb/）中选取两条真实的蛋白质序列，采用 MPSO 算法求解其能量最低值并进行结构预测，检验 MPSO 算法对真实蛋白质序列结构预测的有效性。本文同样采用 K-D 方法[15]来区别它们的疏水和亲水性质，简单地说，I，V，L，P，C，M，A，G 是疏水性；D，E，F，H，K，N，Q，R，S，T，W，Y 是亲水性。为评价 MPSO 算法的性能，将得到的实验结果与文献[9，16]中结果相比较，并且得到最低能量的构像，如图 7 和图 8 所示。

表 3　两个真实蛋白质的最低能量

PDB (ID)	Sequence list	E_{SA}	E_{PSO}	E_{MPSO}
1AGT	GVPINVSCTGSPQCIKPCKDQG MRFGKCMNRKCHCTPK	−17.3628	−19.6168	−19.6241
1AHO	VKDGYIVDDVNCTYFCG RNAYCNEECTKLKGESG YCQWASPYGNACYCYK LPDHVRTKGPGRCH	−14.9612	−15.1911	−19.3571

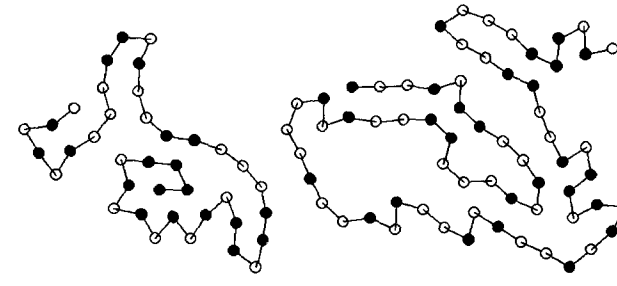

图 7　1AGT 的最低能量构象

图 8 1AHO 的最低能量构形

从表 3 中的结果可以看出，对于“1AGT”，MPSO 得出的最低能量值比通过 PSO 得到的最低能量值略优，但明显优于 SA 算法；对于“1AHO”，我们的结果明显优于通过 PSO 和 SA 算法得到的结果。同时从图 7 和图 8 中也可以看出，“1AGT”的构像中具有一个较为稀疏的疏水核，“1AHO”的构像中疏水性残基形成多个束，被亲水性残基包围。虽然 Toy 模型能够模拟真实蛋白质的一些特性，但是与真正的蛋白质的特性相比，还是有一些不同之处，Toy 模型还有待进一步改善。

结束语　本文提出了一种基于新结构的多种群微粒群优化算法，并将此算法应用于二维的 Toy 模型中进行蛋白质折叠结构预测。通过本文算法得到的结果与已有的研究结果相比，显示了多种群微粒群优化算法良好的性能。并且通过实验结果的分析发现二维的 Toy 模型能在一定程度上反映蛋白质天然结构的一些特点，即在蛋白质序列的构像中，疏水性氨基酸形成束，总是被亲水性氨基酸包围。然而，如何提高多种群微粒群优化算法的计算性能，如何将多种群微粒群优化算法并行化并推广到蛋白质折叠问题的三维 Toy 模型中，使

其成为蛋白质折叠预测问题的有效预测方法将是我们下一步的主要研究内容。

参 考 文 献

[1] Anfinsen C B. Principles that Govern the Folding of Protein Chains. Science,1973,181(4096):223-227

[2] Dill K A. Theory for the Folding and Stability of Globular Proteins. Biochemistry,1985,24:1501-1512

[3] Stillinger F H,Gordon T H,Hirshfeld C L. Toy Model for Protein Folding. Physical Review E,1993,48(2):1469-1477

[4] 邹承鲁. 第二遗传密码:新生肽链及蛋白质折叠的研究. 长沙:湖南科学技术出版社,1997:24-96

[5] Stillinger F H. Collective Aspects of Protein Folding Illustrated by a Toy Model. Physical Review E,1995,52:2872-2877

[6] Rainer K,Thomas D. Improving Genetic Algorithms for Protein Folding Simulation by Systematic Crossover. BioSystems,1999,50(5):17-25

[7] Zhang X L,Lin X L. Protein Folding Prediction Using an Improved Genetic-Annealing Algorithm // The 19th Australian Joint Conference on Artificial Intelligence. Australian, 2006:1196-1120

[8] Kennedy J,Eberhart R. Particle swarm optimization// IEEE International Conference on Neural Networks- Conference Proceedings. Perth,Aust,1995:1942-1948

[9] Liu J,Wang L H,He L L. Analysis of Toy Model for Protein Folding Based on Particle Swarm Optimization Algorithm// International Conference on Natural Computation. 2005:636-645

[10] Eberhart R C,Shi Y. Comparing inertia weights and constriction factors in particle swarm optimization// Proceedings of the IEEE Conference on Evolutionary Computation. California, 2000:84-88

[11] 高鹰,谢胜利. 免疫粒子群优化算法. 计算机工程与应用,2004(1):47-50

[12] 高尚,杨静宇,吴小俊. 求解指派问题的交叉粒子群优化算法. 计算机工程与应用,2004(8):54-55

[13] 江瑞,罗予频,胡东成,等. 一种协调勘探和开采的遗传算法:收敛性及性能分析. 计算机学报,2001(12):1233-1241

[14] Hsiao P H,Vishal M,Peter G. Structure Optimization in an Off-lattice Protein Model. Physical Review E, 2003, 68(3):037703

[15] Mount D W. sequence and genome analysis. Bioinformatics, 2001

[16] Wang L H,Zhou H. Perspective roles of short-and long-range interactions in protein folding. Wuhan University Journal of Natural Sciences,2004,9:182-187

(上接第 222 页)

下,改变节点运动轨迹的采样次数,即每个节点添加虚节点的数目,与 MA-MDS-MAP(P)算法进行比较。结果如图 5 所示。随着采样次数的增加,添加的虚节点数目也越多,两种算法的定位误差都呈下降的趋势。对 MA-MDS-MAP(P)算法而言,采样次数越多,其最短路径算法得到的距离误差也就越小;对 NMDS-LRA(M)算法而言,采样次数越多,得到的距离信息也就越多,相异性矩阵的重构误差也就越小。从图中不难看出,在不同采样次数条件下 NMDS-LRA(M)算法都能取得较小的定位误差。

以上是所有节点均可移动的情况,下面考察网络中只有部分节点可移动的情形。同样,在上面实验的两种拓扑条件下,改变网络中移动节点所占比重,比较了两种算法的定位性能。结果如图 6 所示。不难看出,改变移动节点数目,NMDS-LRA(M)均能取得较好的性能,可移动节点数目越多定位精度越高。即使只有少数移动节点,也可以显著提升定位性能。当移动节点的比重超过全部节点数的 40%就可取得较为理想的定位性能。在保证足够定位精度的前提下,减少移动节点,降低能耗,延长网络生存周期。

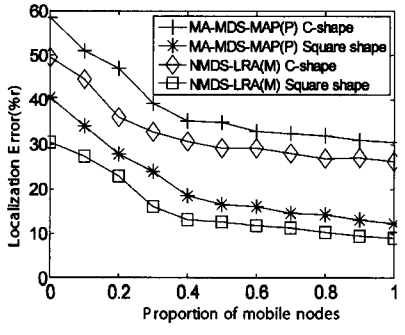

图 6 不同移动节点数的定位性能

结束语 针对移动定位问题,无线传感器网络节点距离矩阵各元素间存在冗余的特性,本文提出了 NMDS-LRA(M)算法。该算法通过对移动节点运动轨迹采样添加虚节点,利用部分节点的距离信息通过奇异值分解计算距离矩阵的逼近阵,然后用非度量多维标度技术进行定位,从而避免以往定位算法采用节点间最短路径算法来构造距离矩阵而引入的误差。仿真分析表明,在方形和 C 形网络拓扑条件下该算法能有效地提高定位精度,对存在测距误差和较低网络连通度环境下的定位具备较强的适应性以及较高的可靠性。

参 考 文 献

[1] Tilak S,Kolar V,Abu-Ghazaleh N B,et al. Dynamic localization control for mobile sensor networks[A]// Proceedings of IEEE International Workshop on Strategies for Energy Efficiency in Ad Hoc and Sensor Networks[A]. New York, USA: IEEE, 2005:587-592

[2] Cheung K W,So H C. A multidimensional scaling framework for mobile location using time-of-arrival measurements[J]// IEEE Trans. Signal Process,2005,53(2):460-470

[3] Chen Zhang-xin,Wan Q,Jiang B,et al. Dynamic Multidimensional Scaling Algorithm for Mobile Location[A]// ENCON 2006. 2006 IEEE Region 10 Conference[C]. Hong Kong,China:IEEE,2006:1-4

[4] Wang C, Ding Y, Xiao L. Virtual ruler : Mobile beacon based distance measurements for indoor sensor localization[A]// The Third International Conference on Mobile Ad-hoc and Sensor Systems (MASS06)[C]. 2006

[5] Shang Y,Ruml W,Zhang Y. Localization from Connectivity in Sensor Networks[J]. IEEE Transactions on Parallel and Distributed Systems,2004,15(11):961-974

[6] Shang Y, Ruml W. Improved MDS-Based Localization[A]// Proc. of the IEEE Infocom[C]. Hong Kong, China. IEEE, 2004:2640-2651

[7] Costa J A,Patwari N,Hero A O. Distributed Weighted-Multidimensional Scaling for Node Localization in Sensor Networks [J]. ACM Transactions on Sensor Networks Journal, 2006, 2(1):39-64

[8] Wu Chang-jua,Sheng Weihua,Zhang Ying. Mobile Sensor Networks Self Localization based on Multi-dimensional Scaling[A]// 2007 IEEE International Conference on Robotics and Automation[C]. Roma,Italy:IEEE,2007:4038-4043

[9] Arora S,Hazan E,Kale S. A Fast Random Sampling Algorithm for Sparsifying Matrices[A]// Proc. of the RANDOM[C]. 2006:272-279

[10] Deshpande A, Varadarajan K. Sampling-based dimensional reduction for subspace approximation[A]// Proc. of the ACM symposium on Theory of computing[C]. San Diego, USA, ACM:2007:641-650

[11] Borg I, Groenen P. Modern Multidimensional Scaling: Theory and Applications[M]. New York:Springer-Verlag,1997