灰狼优化算法及其改进思路

Grey Wolf Optimization Algorithm and Its Improvement G20198818 张如飞

**摘要**： 灰狼优化(Grey Wolf Optimization， GWO)算法是一种新兴的群体智能优化算法，因简单高效而被成功应用于诸多领域。本报告阐述了灰狼优化算法的搜索机制和实现过程，分析灰狼优化算法的特性，并提出几种GWO算法的改进思路。最后通过与粒子群算法、差分进化算法、人工蚁群算法的对比，展示出灰狼算法的有效性和优越性。

关键词： 灰狼优化算法，群体智能优化，搜索机制，参数优化

# 引言

近年来，群体智能优化算法因具有结构简单、易于实现等特点，被广泛应用于复杂问题的求解中。比较流行的算法有遗传算法(Genetic Algorithm， GA)[1]、粒子群优化算法(Particle Swarm Optimization， PSO)[2]、差分进化算法(Differential Evolution， DE)[3]、人工蚁群算法(Anthony Colony Optimization， ACO)[4]和果蝇优化算法(Fruit Fly Optimization， FOA)[5]等。受灰狼群体捕食行为的启发，Mirjalili等[6]于2014年提出了一种新型群体智能优化算法：灰狼优化算法。GWO通过模拟灰狼群体捕食行为，基于狼群群体协作的机制来达到优化的目的。GWO算法具有结构简单、需要调节的参数少、容易实现等特点，其中存在能够自适应调整的收敛因子以及信息反馈机制，能够在局部寻优与全局搜索之间实现平衡，因此在对问题的求解精度和收敛速度方面都有良好的性能。

GWO算法自提出以来，就因具有良好的性能而引起了众多学者的广泛关注。在函数优化方面，已经证明GWO的收敛速度和求解精度均优于PSO[6]。因此，GWO算法被广泛应用于无人作战飞行器路径规划、聚类分析、特征子集选择、经济调度指派、直流电机最优控制、多输入输出电力系统等诸多领域。由于GWO算法提出的时间不长，因此其理论研究尚未成体系，具有广阔的研究前景。

# 文献综述

Mirjalili等[6]采用多个基准测试函数对GWO的性能进行测试，并将其与PSO，DE等算法进行对比分析。借鉴文献[6]的分析结果，可总结出GWO存在如下缺点：

1. 种群多样性差，这是由GWO的初始种群生成方式导致的。随机初始化生成初始种群的方式无法保证较好的种群多样性。
2. 后期收敛速度慢，这是由GWO算法的搜索机制造成的。狼群主要依据与头狼，探狼和猛狼的距离来判断与猎物之间的距离，导致后期的收敛速度较慢。
3. 易陷入局部最优，这是因为头狼狼不一定是全局最优点，在不断的迭代中，猛狼不断逼近前3匹狼，导致GWO算法陷入局部最优解。

针对GWO存在的缺陷，许多学者进行了相关改进，以便提升GWO算法的性能。

Luo等[7]提出了一种基于复数值编码的改进GWO算法。Madhiarasan等[8]将灰狼种群划分为3组，而不是4组。把从属狼群和最低等级狼群归入一个种群，从而降低了计算复杂度，提高了算法的收敛速度。Long等[9]利用佳点集理论对种群进行初始化。当取点的个数相同时，佳点序列选取的点序列的均匀化程度优于其他方法。因此，佳点集方法生成的初始种群的个体分布均匀，保证了种群的多样性，从而为算法的全局寻优奠定了基础。Saremi等[10]将进化种群动态(Evolutionary Population Dynamics， EPD)算子引入GWO。王涛、王勇、蒙丽萍[30]在狼群搜索算法中融合了非线性单纯形法，增强了狼群搜索算法的局部搜索能力。惠晓滨、郭庆、吴聘聘等人[31]将相位因子引入到狼群算法之中，改善了探狼在搜索过程中的灵活性，并提出围攻半径的概念，重新设计了猛狼的位置，来提高猛狼的围攻能力，还引入了混沌优化思想。实验结果表明，改进的狼群算法具有相对较好的优化性能。

# 算法设计

3.1灰狼群体捕食行为

在经过了自然界长时间的种群进化和自然选择，为了适应生活环境，在狼群种群中，逐渐形成了一套十分严格的阶级制度。在这种阶级制度中，狼群被分为头狼、探狼和猛狼三种类型，这三种类型的狼都有自己相应的工作，这样可以保证狼群在种群中的分工明确，这样不仅可以使得行动更加高效，并且这样也更有利于捕获食物，保障狼群的生存和发展。

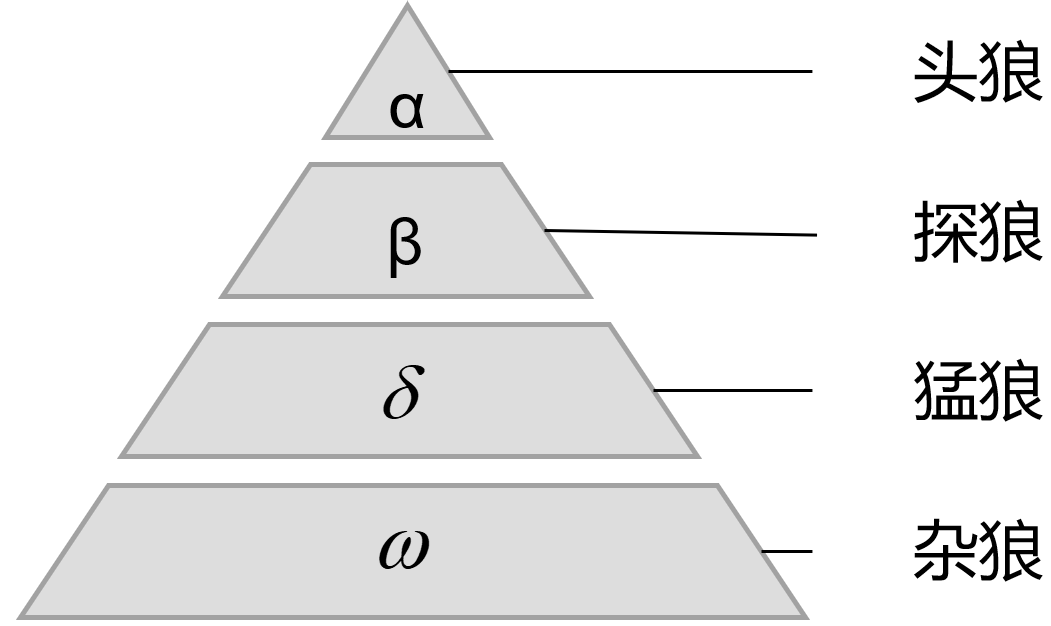


图3-1 灰狼的社会等级秩序

头狼在整个狼群中是最强壮、最有能力以及最富有头脑的。因此，头狼是在所有狼的顺应“优胜劣汰、胜者为王”的自然法则竞争出来的，在整个狼群中占有主导地位。头狼的工作十分重要，因为头狼需要根据整个狼群中其他狼所得到的信息进行整理，然后对狼群中其他狼进行工作调度，负责整个狼群的指挥工作，并带领整个狼群快速地捕获食物。

探狼是狼群中一部分比较精锐的狼。探狼作为整个狼群中的先锋成员，在环境中进行食物搜寻的时候，头狼会排出一部分狼通过嗅取空气猎物的味道，对周围环境进行局部搜索。嗅取的味道越大，说明猎物离探狼的距离也就越近。然后不断地朝着味道大的方位前进，以寻得猎物。

猛狼是狼群捕获猎物的劳动力量，当探狼发现猎物的位置后，探狼会立刻向头狼进行汇报。当头狼接收到探狼发出的信号之后，就召集周围的猛狼，让猛狼跟着自己一起往猎物的方向冲锋，然后对猎物进行围捕。

为了使得狼群能保持良好的状态，获取的猎物会优先给功劳最大的狼进行分配，然后依次类推，最后才会分配到弱小的、好吃懒做的、功劳最小的狼。因为捕获的食物是优先的，所以最后可能会出现一些弱小的狼无法获得食物的情况。为了狼群的生存和发展，这样可以使得有能力、能干活的狼生存下去，为整个狼群捕获更多的食物。

因此，灰狼算法便是根据狼群在捕获猎物的过程进行模拟。游走行为负责在当前位置周围寻找出局部最优位置;号召行为指的是头狼接收到探狼找到猎物的信息后，对狼群中的猛狼进行召集的行为，狼群中的猛狼会快速朝头狼聚集，准备对猎物进行围捕;围捕行为则是狼群捕猎最后一步，由头狼召集到的猛狼对猎物进行攻击。当捕猎完毕后，获取的猎物会优先给功劳最大的狼进行分配，然后依次类推，最后才会分配到弱小的、好吃懒做的、功劳最小的狼。因为捕获的食物是优先的，所以最后可能会出现一些弱小的狼无法获得食物的情况。通过这种狼群的更新机制，有能力、能干活的狼会生存下去，保障了狼群的发展的延续。

3.2灰狼算法描述

灰狼算法的具体实现流程如下所示：

1. 种群初始化

在算法最开始的时候，需要对狼群的规模、维度、算法最大迭代次数进行设置，假设算法最大迭代次数为，狼群中狼的个数为，每匹狼的维度为，则第匹狼的位置可以表示为公式(3.1)。

 ········ (3.1)

其中，，。

然后根据公式(3.2)随机初始化则第匹狼在第0次迭代时候的初始位置。

······· (3.2)

其中表示狼群在第d维空间的最小取值，表示狼群在第d维空间的最大取值，是一个范围在[0,1]之间的随机数。

在初始化每一匹狼在每一个维度空间的位置之后，需要根据适应度函数计算每一匹狼的适应度值，然后根据适应度值，将目前最优位置的狼作为头狼。头狼在每一次迭代的过程中，是不需要执行三种智能行为的，它只是作为狼群的号召和标杆。即根据公式(3.3)选出头狼。

·········· (3.3)

其中t表示迭代次数，其取值在[0, ]之间。

1. 游走行为

在这个阶段头狼是出于原地不动的状态，然后需要挑选出（的值为之间的随机整数，表示探狼的比例因子）匹精锐的狼(除了头狼外)作为探狼形成先锋部队，沿着个不同的方向进行探索，同时需要记录自己感知到的猎物的气味浓度。当个方向进行探索完毕之后，选择气味浓度大的方向前进，最后更新自己的位置。根据公式(3.4)可以更新探狼在第维的第个方向探索后的位置，一直重复这个过程，直到某只探狼的位置优于头狼，发生头狼与探狼之间的身份将换，或者是游走次数达到最大限制。

······· (3.4)

其中，是在[1，]之间的正整数，表示游走步长。可以看出如果的值越大，虽然搜索精度会得到很大的提升，但是每一只探狼在进行探测时花费的时间也就更多。

然而如果在进行游走之前，第只探狼的位置已经比头狼的位置好，则这只探狼会代替头狼。

1. 召唤行为

在召唤阶段，头狼和探狼的位置都是不变的，并且头狼会号召周围的 （的值为，即狼群种群中除掉头狼和探狼）猛狼向猎物靠近，准备进行围捕。听到头狼在召唤的猛狼后，会以的步长快速往头狼的方向进行汇合。利用公式(3.5)可以更新第只猛狼在第次迭代时，第维的位置。

 ······· (3.5)

其中，表示第只猛狼在第次迭代时，第维的位置。表示探狼在第维的奔袭步长，表示在第次迭代时，头狼在第维的位置。

如果当第只猛狼在向头狼奔跑的过程中，突然发现自己嗅到猎物的气味比头狼描述的气味更浓郁，那么这次猛狼就会变为头狼（即），并且头狼变为猛狼，和除了变成头狼的这只猛狼外的猛狼一起，朝着这只猛狼进行聚集。

如果猛狼在朝着头狼的方向奔走的过程中，嗅到猎物的气味一直不如头狼嗅到的气味，则一直持续向着头狼奔走，直到猛狼与头狼之间的距离小于由公式(3.6)求出的距离阈值。当猛狼与头狼之间的距离小于距离阈值后，猛狼停止奔跑，准备进行围捕。

 ······· (3.6)

其中，表示问题的维度，表示距离的判定因子，表示在第维空间的最大值，表示在第维空间的最小值。

可以看出如果较大，距离阈值就会较小，这样猛狼就花更多时间进行奔走，出现猛狼嗅到猎物的气味比头狼嗅到猎物气味好的概率就会越大，这样就会造成头狼在不同的猛狼之间不断改变，这样的话，猛狼就会很难进入围捕阶段，算法的精细搜索能力就会不足。

1. 围捕行为

在猛狼都靠近头狼距离之后，猛狼这个时候需要和探狼一起合作，对猎物进行围捕。由于对于猎物位置的精确感知，所以在这个阶段是利用头狼的位置来代替猎物的位置，即将头狼视为猎物。假设在第次迭代时，猎物在第位空间的位置为，则探狼和猛狼的位置更新如公式(3.7)所示。

······· (3.7)

其中，表示第只猛狼在第次迭代时，第维的位置。表示探狼和猛狼在第维的围捕步长。是一个在[-1,1]之间的一个随机数。

需要注意的是，在围捕阶段，只有探狼和猛狼在位置更新后，猎物的浓度大于在探狼和猛狼在位置没有更新之前，探狼和猛狼的位置才会更新，否则，探狼和猛狼的位置不会变。

1. 淘汰机制

该分配机制是根据“论功行赏”的规矩来制定的，获取的猎物会优先给功劳最大的狼进行分配，然后依次类推，最后才会分配到弱小的、好吃懒做的、功劳最小的狼。因为捕获的食物是优先的，所以最后可能会出现一些弱小的狼无法获得食物的情况。虽然这个方式很残忍，但是为了狼群的生存和发展，这样可以使得有能力、能干活的狼生存下去，为整个狼群捕获更多的食物。

在淘汰阶段，会根据适应度函数选出最差的匹狼，与此同时会重新随机生成匹新的狼出来，代替被淘汰的狼。

如果过大的话，算法就会丢失很多已经经过训练的狼，又重新生成，如果的值趋向于种群数的话，就变成了随机收缩，失去了群智能算法中，对于迭代的意义。如果特别小的话，由不能扩大种群的多样性，在算法后期，很容易会陷入局部最优，降低算法局部勘探能力。所以根据实际情况的不同，即需要捕获猎物的大小、数量是不同的，从而造成了不同数量的狼被淘汰。在原始算法中， 每一次淘汰的狼的个数是取值在范围中的一个随机正整数，其中表示种群淘汰的比例因子。

在算法的最后，算法还给出了在第维，狼群的游走步长、奔袭步长以及围捕步长之间的关系。其关系公式如公式(3.8)所示。

 ··· (3.8)

其中，表示在第维空间的最大值，表示在第维空间的最小值，表示步长因子，代表了灰狼算法在解空间中寻找最优解的精度。

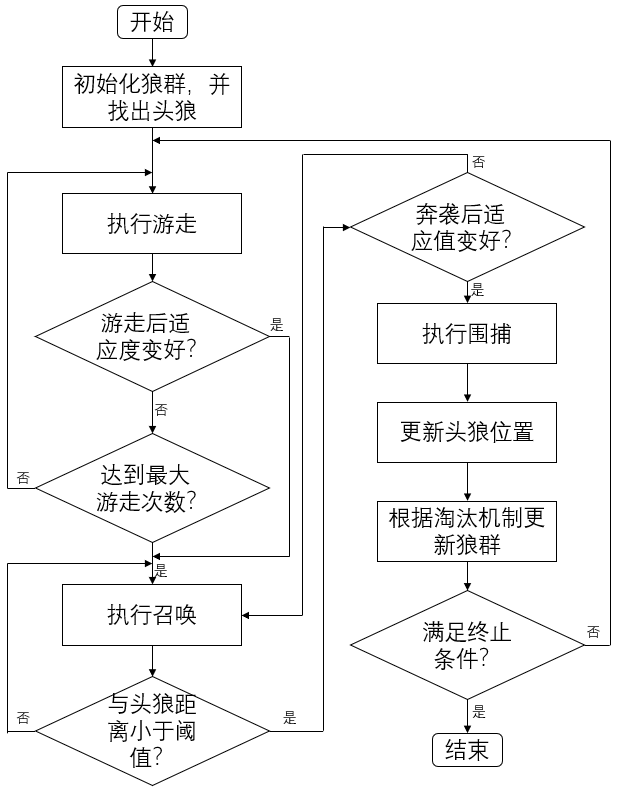


图3-1 灰狼算法流程图

以上步骤简单描述为：

Step 1 初始化，，等参数以及灰狼群体，以及每个灰狼的位置;

Step 2 计算每个灰狼个体的适应度值，记录适应度值排列前三的灰狼个体的位置，并将适应度值最好的位置记为最优解;

Step 3 按照计算剩余个体与适应度前三位置的距离，并更新各狼和猎物的位置;

Step 4更新各参数因子的值;

Step 5 若算法到达到最大迭代次数，那么算法结束并输出最优解；否则，返回 Step 2。

# 灰狼算法改进

上文已提到原始灰狼优化算法存在着诸多缺陷：

1. 种群多样性差；
2. 后期收敛速度慢；
3. 易陷入局部最优。

针对这些缺陷，本次课程实验对其进行了相关改进，以便提升GWO算法的性能。

主要改进点：

1 在猛狼进行更新位置更新的时候，考虑狼群中，所有狼群所达到过的最好位置，加入猛狼的历史最优位置，每次进行位置迭代时，将历史最优位置作为计算参数之一，提高搜索效率。

2 将莱维飞行[13]与灰狼算法相结合

虽然灰狼算法收敛速度很快，但在探索猎物的过程中易早熟，难以找到全局最优解。莱维飞行是一种随机游走过程，其步长服从重尾分布，是一种非高斯随机过程。重尾分布可以以较大概率取到极大的值，即可以较大的概率进行大幅度的跳跃，从而避免局部最优、扩大搜索的范围，以获得全局最优解。现将莱维飞行与灰狼算法结合，以拓展灰狼算法提高其全局搜索能力，改善早熟现象。位置更新公式更改为：



3 在淘汰机制中，参考细菌觅食优化算法[14]中的复制过程，在该过程中，同样是遵守了“优胜劣汰，适者生存”的准则。

首先计算每匹狼的适应度值，然后根据适应度值，淘汰较差的半数狼群，剩下一般较好的狼群中的每一匹狼都复制成两匹。这样做可以做到简化参数的同时，保留多次迭代后较好的狼，也不需要对被淘汰的狼进行重新初始化。

4 为了使得算法在前期拥有更好的全局搜索能力，在后期拥有较强的局部勘探能力，提出了自适应步长公式。

在算法的执行过程中游走步长 、奔袭步长以及围捕步长是一直不变的。然而在群智能算法中，当算法迭代多次趋近收敛之后，需要降低个体移动的距离，这样有利于算法的局部勘探能力，所以，本文提出自适应步长公式，用于改进在算法过程中不会变化的游走步长、奔袭步长以及围捕步长 。



其中，表示算法开始时候第0次迭代时，初始化的步长因子，表示当前迭代次数，表示自适应强化狼群算法中的最大迭代次数，是一个取值范围在[0,1] 之间的随机数。

则狼群的游走步长、奔袭步长以及围捕步长分别为







其中，表示在第维空间的最大值，表示在第维空间的最小值。

# 复杂度分析

根据算法的执行步骤，下面讨论GWO算法时间复杂性，这里只计算每一步的运算次数。在Step 1中，对维搜索空间下的个灰狼进行初始化赋值需要次运算;在Step 2中，计算个灰狼个体的适应度值需要次运算，而适应度函数的复杂度一般是，选出适应度值前三的个体至多需要次运算，记录最优解位置其运算次数加1；在Step 3中，计算剩余个体与头狼探狼的距离需要次运算，而距离函数的复杂度是 。更新各灰狼和猎物的位置需 次运算;在Step4和Step5中更新步长因子及判断算法是否满足终止条件，其复杂度均为常数时间。由于算法最多执行次（是用户设定的最大迭代次数或者是为了达到某计算精度所需的迭代次数）。因此，通过近似和简化运算，GWO算法的时间复杂度约为。

通过利用经验法即实验测量与群体大小、最大迭代次数、数据维度的关系，得到的关系图分别如图5-1、5-2、5-3所示。验证了GWO算法的时间复杂度。

|  |  |
| --- | --- |
|  |  |
| 图5-1 运行时间与群体大小的关系 | 图5-2 运行时间与最大迭代次数的关系 |
|  |  |
| 图5-2 运行时间与数据维度的关系 |  |

# 性能验证

将灰狼算法、粒子群算法、差分进化算法、人工蜂群算法应用于K-means算法寻优过程，并对得到的有效性以及聚类准确性进行对比分析。选取的数据集包含250个样本，每个样本是一个二维数据，可以分为5个类。其中每一个类都是均值分布的，并且每一个类包含50个样本。

基于加入了自适应步长强化灰狼算法的K-means算法和粒子群优化算法、差分进化算法、人工蜂群算法结合K-means算法在人工数据中的平均适应度值、标准差适应度值、最优适应度值和最差适应度值显示在表6-1中。其中加粗的数字表示的是在这几个算法中的最优值。

表6-1 几种算法适应度比较

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 算法类型 | 平均值 | 标准值 | 最优值 | 最差值 |
| 自适应步长  灰狼算法 | 1972.76 | 104.52 | 1726.43 | 2106.39 |
| PSO | 2205.41 | 312.73 | 1773.82 | 2444.89 |
| DE | 1992.03 | 200.67 | 1752.89 | 2495.90 |
| ABC | 2079.42 | 107.49 | 1902.56 | 2206.78 |

从表 6-1 中可以看出，自适应步长灰狼算法的K-means算法比应用PSO、DE、ABC的K-means算法在人工数据集的测试中，在平均适应度值、标准差适应度值、最优适应度值和最差适应度值结果中均取得最好的效果。

# 结论

灰狼算法作为一种具有广阔应用前景的群优化搜索算法，已吸引越来越多学者目光。本报告详细阐述了灰狼算法的原理和运行机制，并指出原始灰狼算法在初始化种类多样性差、后期搜索速度慢、易陷入局部最优等缺点。为了改善灰狼算法的综合性能，本报告提出了四点改进思路，并最终通过与粒子群算法、差分进化算法等的实际应用结果对比，展示出灰狼算法的极其可观的研究开发前景。

参考文献:

[1] WHITLEY D. A genetic algorithm tutorial[J]. Statistics and Computing,1994,4(2):65-85.

[2] TRELEA I C. The particle swarm optimization algorithm: convergence analysis and parameter selection[J]. Information Processing Letters,2003,85(6):317-325.

[3] STORN R, PRICE K. Differential evolution-A simple and efficient heuristic for global optimization over continuous spaces[J]. Journal of Global Optimization,1997,11(4):341-359.

[4] DORIGO M, BIRATTARI M, STUTZLE T. Ant colony optimization[J].

Computational Intelligence Magazine IEEE,2007,1(4):28-39.

[5] PAN W T. A new fruit fly optimization algorithm: taking the financial distress model as an example[J]. Knowledge-Based Systems,2012,26(2):69-74.

[6] MIRJALILI S, MIRJALILI S M, LEWIS A. Grey Wolf Optimizer[J]. Advances in Engineering Software,2014,69(3):46-61.

[7] LUO Q, ZHANG S, LI Z.et al. A Novel Complex-Valued Encoding Grey Wolf Optimization Algorithm[J]. Algorithms ,2015,9(1):1-23.

[8] MADHIARASAN M, DEEPA S N. Long-Term Wind Speed Forecasting using Spiking Neural Network Optimized by Improved Modified Grey Wolf Optimization Algorithm[J]. International Journal of Advanced Research,2016,4(7):356-368.

[9] LONG W, ZHAO D Q, XU S J. Improved grey wolf optimization algorithm for constrained optimization problem[J]. Journal of Computer Applications,2015,35(9):2590-2595.(in Chinese) 龙文,赵东泉,徐松金.求解约束优化问题的改进灰狼优化算法[J].计算机应用,2015,35(9):2590-2595.

[10] SAREMI S, MIRJALILI S Z, MIRJALILI S M. Evolutionary population dynamics and grey wolf optimizer[J]. Neural Computing & Applications,2015,26(5):1257-1263.

[11] 王涛, 王勇, 蒙丽萍. 一种改进的狼群搜索算法及用于解决聚类问题[J].计算机应用与软件, 2016(12):257-263.

[12] 惠晓滨、郭庆、吴聘聘、赵昱.一种改进的狼群算法[J].控制与决策，2017,32(7):1163-1172.

[13] 郑国雨,李伟鹏,邱晓宁.基于改进狼群算法的井下排水采气超声速喷嘴优化[J].科学技术与工程,2019,19(31):129-133.

[14] 周爱武, 于亚飞. K-Means 聚类算法的研究[J]. 计算机技术与发展, 2011,21(2):62-65.