一种求非线性整数规划最优解的仿生算法

罗伟强，于建涛，黄家栋
LUO Wei-qiang，YU Jian-tao，HUANG Jia-dong

华北电力大学 电气工程学院，河北 保定 071003
North China Electric Power University，Baoding，Hebei 071003，China
E-mail：luoweiqiang45@163.com

LUO Wei-qiang，YU Jian-tao，HUANG Jia-dong.Bionic algorithm for solving nonlinear integer programming.Computer Engineering and Applications，2008，44（7）：57-59.

Abstract：Be inspired from natural plant growth，this paper proposes a bionic algorithm for solving the global optimization of nonlinear integer programming.By simulating the process of plant growth and growth mode，the algorithm can rapidly access to the optimal solution.The process which solved varies types of nonlinear integer programming shows that the algorithm is very effective.
Key words：net growth forces；integer programming；simulation of plant growth；Monte Carlo method

摘　要：从大自然植物生长中得到启发，提出了一种求解非线性整数规划全局最小解的仿生算法。该算法将植物生长过程及生长模式应用到非线性整数规划问题的求解，能够快速得到最优解。通过对各种不同类型非线性整数规划问题的具体求解，表明了该方法十分有效。
关键词：净生长力；整数规划；模拟植物生长；蒙特卡洛法
文章编号：1002-8331（2008）07-0057-03　**文献标识码**：A　**中图分类号**：TP301.6

整数规划是20世纪60年代发展起来的规划论中的一个分枝，由于该问题是NP-完全的，求解精确解的算法具有指数复杂性[1]。因此，尽管经过数十年的发展，整数规划，特别是非线性整数规划问题目前仍缺乏有效的通用算法。当前求解非线性整数规划问题的方法是采用近似算法，而近似算法中主要分为基于蒙特卡洛法的随机算法[6,13]和基于贪心算法类的搜索算法[3,4,7,9,10]及其他解法[2,5,8,14]。在这些方法中，文献[9]提出的模拟植物生长的仿生思想比较惹人注意，取得了一定的成效，但是由于采用了随机算法并且没有考虑植物生长中智能化的因素，因此搜索空间大、迭代次数多，导致有的问题不能很好解决（如本文算例4）。

通过观察植物生长过程中的特征，本文提出了模拟植物生长的智能模拟算法。该算法采用植物顶点变速度生长特点来减少搜索时间，利用植物生长期前期纵向型生长特性来大大减少搜索空间，因此能够在更少的时间内得到更优解。通过对不同类型的算例求解，表明了该算法是很有效的。

1　植物生长自然现象

植物的生长过程，是同化作用和异化作用相互作用的过程。在这个过程中，如果把吸收用于同化作用的水分、养料看成是生长动力，把异化作用中的呼吸、克服重力及蒸腾作用的消耗看成是生长阻力，并定义净生长力等于生长动力与生长阻力之差（生长过程中生长动力与阻力的关系如图1所示），则当净生长力大于0时，植物就开始生长；等于0时的动态平衡状态就是植物呈现长大了的成熟期状态，此时把植物所占空间称为生长空间；小于0则植物开始枯萎衰败。

植物在成长期间，总是努力的向上和四周繁殖出更多的树枝，以便接受更多的阳光，从而使生长最快。因此，植物的整个成长期（如图1的 op 阶段），就是以一种全局最优的方式尽可能长满生长空间的过程。

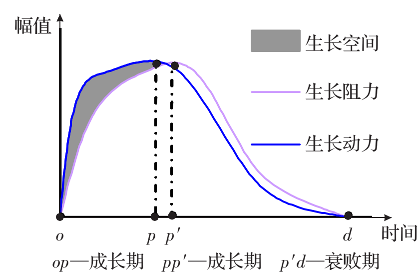


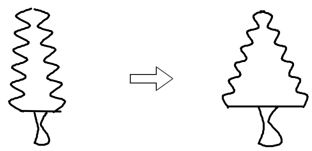
图1　植物生长过程图

当树枝的净生长力大于0时，树枝的节点就开始生长，向各个方向长出新分枝，然后各个新分枝朝尽快长满生长空间的方向生长。当存在多个待生长的树枝节点时，净生长力最大的

作者简介：罗伟强（1984-），男，硕士研究生，研究方向为电力系统继电保护、电力系统自动化控制；于建涛（1983-），男，硕士研究生，研究方向为电力系统继电保护、电力系统自动化技术；黄家栋（1961-），男，副教授，主要研究方向为电力系统自动化控制、计算机技术在电力系统中的应用。

收稿日期：2007-06-25　**修回日期**：2007-09-03

节点优先获得生长机会。当一个节点生长完成后,其他树枝的净生长力将随之动态改变(如原来树枝被新长出来的树枝遮挡住了太阳)。一般来说,植物的最高树枝并不是到成熟期才长成的,而是在成长期就已经长成。植物最高树枝(顶点)生长特点是:以很大的初速度呈减速度生长;植物生长的趋势是:如图2所示,在成长期的前期主要纵向型生长(往上生长),后期主要是横向型生长(往四周生长)。



(a)纵向型生长　　(b)横向型生长

图2　植物生长模式图

2　植物生长的算法模拟

根据植物成长期的特征,将植物生长步骤描述如下:

(1)种子破土而出,长出第一个树枝;

(2)在净生长力大于0的树枝里面选择净生长力最大的树枝去生长出新的树枝;

(3)如果存在净生长力大于0的树枝,则转向(2);

(4)植物进入成熟期。

考虑如下的一般的非线性整数规划问题:

$$\begin{cases}\min f(X)\\ \text{s.t. } X\in X_1\end{cases} \tag{1}$$

其中 $X_l\in R_l^n$,R_l^n 是 R^n 中整点的集合。

将非线性整数规划问题与植物生长过程建立如下关系:把式(1)中的初始点 X_0 看成是植物破土的第一个树枝,把 X_i 看成是植物的第 i 条树枝,把 X_0 与当前点 X_i 函数值之差 $f(X_o)-f(X_i)$ 看成点 X_i 的净生长力,把 X_l 看成是生长空间,把 $\min f(X)$ 看成是植物的最高点,则可以用植物以全局最优方式长满整个生长空间的过程来求解上述非线性整数规划问题。

2.1　普通模拟算法

根据上面建立的非线性整数规划问题与植物生长过程关系,解决非线性整数规划问题的模拟算法步骤如下:

(1)设置第一个树枝 X_0,生长步长 $step=1$,树枝集合 $allset=\{\}$,树的顶点 $X_{\min}=\{X_0\}$,树的顶点高度 $F_{\min}=f(X_0)$,生长树枝集合 $growset=\{X_0\}$。

(2)从 $growset$ 中选择一个净生长力最大的树枝 X_t,即 $X_t=\max\{f(X_0)-f(X_i),X_i\in growset\}$,并在 $growset$ 中删除 X_t 及将 X_t 加入到 $allset$ 中,然后在 X_t 的 $2n$ 个方向上依次生长新树枝 X_t^i。

如果 X_t^i 在生长空间中(即 $X_t^i\in X_l$,$X_t^i\notin allset$,$X_t^i\notin growset$),则将 X_t^i 加入到 $growset$ 中。

如果 X_t^i 满足条件 $F_{\min}==f(X_t^i)$,则将 X_t^i 加入 $X_{\min}$ 中。

如果 X_t^i 满足条件 $F_{\min}<f(X_t^i)$,则 $X_{\min}=\{X_t^i\}$,$F_{\min}=f(X_t^i)$。

X_t 在第 i 个方向上生长新树枝的方法为:$X_t^i=X_t+step*g_i$,$g_i=[0\ 0\cdots\ 0\ 1\ 0\cdots0]$,即 g_i 的第 i 个分量为1,其他分量均为0,则 X_t^i 为 X_t 在第 i 个方向上长的新树枝。

(3)如果 $growset$ 中元素个数不为0,则转向(2)。

(4)输出 $X_{\min}$ 及 $F_{\min}$,退出。

由算法执行过程可以看出,该算法能够得到非线性整数规划问题的所有最优解,但是得遍历整个生长空间。

2.2　智能模拟算法

仔细分析2.1节中算法,虽然可以搜索到所有最优解,但是如果生长空间很大,则得花费很长时间才能够得到结果。

从图2植物的生长模式可以看出,植物的最终顶点在成长期的前期就已经长成,后面都是向四周生长,并且植物顶点的生长速度是在不断变化的。因此,利用这些特点,就不必要搜索整个生长空间就可以得到全局最优解。然而,部分搜索就必须面对以下两个问题:

(1)如何知道搜索多少步骤之后就可以得到问题的全局最优解;

(2)如果初始值的选取与最优解偏离很大,如何能够减少搜索时间。

对于问题(1),考虑到最终顶点树枝是生长最快的树枝,因此只要记录下从开始到搜索出当前顶点树枝 $X_{\min}$ 所用的搜索步骤数 $numsabk=nums$,当在接下来的 $k*numsbak/s$ 搜索步骤内,$X_{\min}$ 没有任何变化,则认为已经搜索到所有全局最优解。对于 k 的选取,为了避免迭代初期提早退出和后期过多冗余迭代,因该采取 k 值满足:初始值较大,搜索过程中逐渐向1逼近的原则,同样 s 的选取满足:初始值为1,搜索过程逐渐增大。

对于问题(2),为了加快搜索速度,可以对生长步长 $step$ 采用变值,遵循原则是先采用大步长,后采用小步长,最后阶段采用 $step=1$。一种可行的变步长方案为:置初值 $step=2^n$,当生长出来的新树枝中有位于生长空间之外时,$step=step-1$,直到 $step=1$。

上面智能模拟算法方案一般都能够快速搜索到全局最大值,但是如果初值选的很不恰当,如图3所示的情况,此时初值落在离全局最优值很远的一个极值附近,则会把这个极值当成全局最优值,这时,只能靠选取合适初值或者强制增加迭代步数来解决。

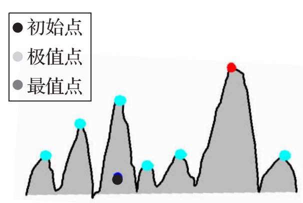


图3　初值选取不当的智能模拟算法图

根据以上分析,得到智能模拟算法步骤如下:

(1)设置第一个树枝 X_0,生长步长 $step=Step_0$,树枝集合 $allset=\{\}$,树的顶点 $X_{\min}=\{X_0\}$,树的顶点高度 $F_{\min}=f(X_0)$,生长树枝集合 $growset=\{X_0\}$,搜索步数 $nums=0$,顶点高度保持步数 $nochanged=0$,$k=K_0$,$s=1$,搜索最优值所耗步数 $numsbak=n$。

(2)从 $growset$ 中选择一个净生长力最大的树枝 X_t,即 $X_t=\max\{f(X_0)-f(X_i),X_i\in growset\}$,并在 $growset$ 中删除 X_t 及将 X_t 加入到 $allset$ 中,置 $nums=nums+1$。在 X_t 的 $2n$ 个方向上依次生长新树枝 X_t^i。

如果 X_t^i 在生长空间,则将 X_t^i 加入到 $growset$ 中。

表1 算例1迭代结果

	初始值 X_0	迭代次数 *nums*	最优解 $X_{\min}$	函数值 $f(X_{\min})$
智能模拟算法	(0,0,0)	309	(1,1,1),(1,1,−1),(1,−1,1),(−1,1,1),(1,−1,−1),(−1,1,−1),(−1,−1,1),(−1,−1,−1)	−11.7
	(1,1,1)	31	(1,1,1),(1,1,−1),(1,−1,1),(−1,1,1),(1,−1,−1),(−1,1,−1),(−1,−1,1),(−1,−1,−1)	−11.7
文献[9]	(0,0,0)	51	(−1,−1,1),(1,1,1),(−1,1,1)	−11.7

如果 X_t^i 不在生长空间中并且 $step>1$，则将 $step=step-1$。

如果 X_t^i 满足条件 $F_{\min}==f(X_t^i)$，则将 X_t^i 加入 $X_{\min}$ 中，$nochanged=0$。

如果 X_t^i 满足条件 $F_{\min}<f(X_t^i)$，则 $X_{\min}=\{X_t^i\}$，$F_{\min}=f(X_t^i)$，$nochanged=0$，$numsbak=nums$，$s=s+1/(S_0*n)$。

如果 $k>1$，则 $k=k-1$。如果 X_t^i 满足条件 $F_{\min}>f(X_t^i)$，则 $nochanged=nochanged+1$。

(3)如果 $nochanged<k*numsbak/s$，则转向(2)。

(4)输出 $X_{\min}$ 及 $F_{\min}$，退出。

对于步骤(2)中的 S_0，一般取值在2~5，迭代次数越多则选取越小的值。

3 算例

在下面所有算例里，智能模拟算法中相关参数设置如下：$Step_0=2^n$，$K_0=2^n$，$S_0=5$。

算例1 求解以下无约束非线性整数规划问题(见文献[14])：

$$\min \sum_{i=1}^{3}(x_i^4-4.9x_i^2)$$

$$\text{s.t.}\ |x_i|\leqslant 5$$

根据算法，编写程序计算结果如表1所示，从表中可以看出，智能模拟算法找到所有的最优解，初值与最优解越接近，迭代的次数就越少，而文献[14]找到了1个解，文献[9]找到了3个最优解。

算例2 求解以下无约束非线性整数规划问题(见文献[9])：

$$\min f(x)=(x_1-1)^2+(x_2-2)^2+(x_3-4)^2$$

$$\begin{cases}-x_1+x_2+x_3\leqslant 3.5\\ x_1+x_2-x_3\leqslant 6\\ x_1,x_2,x_3\geqslant 0\text{，且为整数}\end{cases}$$

根据算法，编写程序计算结果如表2所示，从表可以看出，智能模拟算法找到同等数量的最优解所花的迭代次数要少，同样，初值与最优解越接近，迭代的次数就越少，而且任意给初值不一定能够得到全部的最优解。

表2 算例2迭代结果

	初始值 X_0	迭代次数 *nums*	最优解 $X_{\min}$	函数值 $f(X_{\min})$
智能	(0,0,0)	29	(1,1,3),(2,2,3),(2,1,4)	2
模拟	(1,1,0)	13	(1,1,3),(2,2,3),(2,1,4)	2
算法	(2,2,0)	7	(2,2,3)	2
文献[9]	(0,0,0)	37	(1,1,3),(2,2,3),(2,1,4)	2
文献[15]	(8,8,8)	未知	(1,1,3)，(2,2,3)	2

算例3 求解以下无约束非线性整数规划问题(见文献[12])：

$$\min f(x)=x_1^2+2x_2^2+x_3^2+2x_4^2-27x_1-31x_2-41x_3-34x_4$$

$$\text{s.t.}\begin{cases}x_1^2+2x_2^2+2x_3^2+x_4^2-x_1+x_2-x_3-61\leqslant 0\\ 2x_1^2+3x_2^2+3x_3^2+x_4^2-x_2-2x_3+x_4-90\leqslant 0\\ x_1^2+2x_2^2+2x_4^2+2x_1-x_2+3x_4-43\leqslant 0\\ 0\leqslant x_i\leqslant 20\ (i=1,\cdots,4)\text{，且 } x_i \text{ 为整数}\end{cases}$$

根据算法，编写程序计算结果如表3所示。

表3 算例3迭代结果

	初始值 X_0	迭代次数 *nums*	最优解 $X_{\min}$	函数值 $f(X_{\min})$
智能模拟算法	(0,0,0,0)	33	(3,4,3,2)	−338
	(1,1,1,1)	33	(3,4,3,2)	−338
	(0,0,1,2)	27	(3,4,3,2)	−338
	(2,2,2,2)	33	(3,4,3,2)	−338
文献[12]	未知	未知	(3,4,3,2)	−338

算例4 求解以下无约束非线性整数规划问题(见文献[13])：

$$\text{Max}\ z=x_1^2+x_2^2+3x_3^2+4x_4^2+2x_5^2-8x_1-2x_2-3x_3-x_4-2x_5$$

$$\text{s.t.}\begin{cases}0\leqslant x_i\leqslant 99\ (i=1,\cdots,5)\\ x_1+x_2+x_3+x_4+x_5\leqslant 400\\ x_1+2x_2+2x_3+x_4+6x_5\leqslant 800\\ 2x_1+x_2+6x_3\leqslant 200\\ x_3+x_4+5x_5\leqslant 200\end{cases}$$

此规划问题的生长空间十分大，大约 $(100)^5=10^{10}$ 个点，且存在与最优值很接近的极值，目前尚无有效方法求出准确解[13]。将 Max z 改写成 min−z 形式，根据算法，编写程序计算结果如表4所示，从表中可以看出，智能模拟算法可以在搜索大概 441*(2*5−1)=3 969个点就得到最优值，明显优于文献[13]中蒙特卡洛法。但是当初始值为(0,10,0,0,0)时，得到的是一个局部最优值，说明初始点落在如图3所示的区域并且此时的迭代次数过少，让算法强制迭代303*2次就得到了全局最优值。

表4 算例4迭代结果

	初始值 X_0	迭代次数 *nums*	最优解 $X_{\min}$	函数值 Max z
智能模拟算法	(0,0,0,0,10)	441	(50,99,0,99,20)	51 568
	(50,50,0,0,0)	195	(50,99,0,99,20)	51 568
	(0,50,0,99,0)	89	(50,99,0,99,20)	51 568
	(0,10,0,0,0)	303	(0,99,16,99,17)	49 972
	(0,10,0,0,0)	强制迭代 303*2=606次	(50,99,0,99,20)	51 568
	(10,10,0,0,0)	377	(50,99,0,99,20)	51 568
文献[13]			(23,99,7,98,10)	48 392
蒙特卡洛法			(30,90,6,99,3)	47 775

4 结论

本文针对自然界植物生长的特点，利用植物生长过程及模式，提出了一种新的植物生长智能模拟算法。通过上面的分析讨论，针对该智能算法有如下结论：

(1)该算法原理简单，很容易通过编程实现，能够快速得到非线性整数规划问题的最优解，并且在大部分情况下，能够得到所有全局最优解，这优于文献[9]提出的算法。

(下转68页)