基于复合粒子群算法的几何约束求解技术研究

曹春红[1)]　张　斌[1)]　李文辉[2)]

[1)]（东北大学信息科学与工程学院，沈阳　110004）　[2)]（吉林大学计算机科学与技术学院，长春　130012）

摘　要　在将几何约束问题的约束方程组转化为优化模型的时候，需要找到一种方法来跳出局部最优解，进而找到全局最优解。为了兼顾算法的快速性和全局性，几何约束求解时，考虑使用复合粒子群算法。这种粒子群算法是一种基于群智能方法的演化计算技术，不仅在所有的进化算法中都包括控制其自身特性的启发式参数，而且这些参数通常是与特定的问题相关，并可由用户自己定义。虽然合适的参数选择需要用户丰富的经验和对研究问题所提供信息的正确判断，更重要的是，这些启发式参数会影响到算法的收敛特性，但是即便是很有经验的用户也可能选择不恰当的参数，从而使问题得不到有效地解决，这就越来越需要对这些参数进行研究。为此可将将粒子群算法中的控制参数的选取作为一个优化问题，以便用常规遗传算法来控制粒子群算法中的启发式参数，即形成了复合粒子群优化算法，通过把复合粒子群算法成功地应用到几何约束求解技术的实验表明，该方法可以在很短的时间内找到最优解。

关键词　几何约束求解　群智能算法　复合粒子群算法

中图法分类号：TP391.72　　**文献标识码**：A　　**文章编号**：1006-8961(2007)04-0713-05

The Research Based on the Composite Particle Swarm Optimization Algorithm in the Geometric Constraint Solving

CAO Chun-hong[1)], ZHANG Bin[1)], LI Wen-hui[2)]

[1)] (*Collge of Information Science and Engineering*, *Northeastern University*, *Shenyang* 110004)

[2)] (*College of Computer Science and Technology*, *Jilin University*, *Changchun* 130012)

Abstract　When transferring a geometric constraint equation group into an optimization model, we need a method to jump out of the local beat solution so that we can find a best global solution. Considering the speed and global capability, we adopt a composite particle group optimization algorithm. Particle swarm optimization algorithm is a kind of evolution computation technology based on group intelligence. In all evolution computations heuristic function should be included to control its own characteristic. These parameters are usually correlated with a specific problem and are defined by the users. Suitable parameter choice needs user' s abundant experience and correct judgment on the information offered by the problem. More important thing is that these heuristic parameters will influence the convergence characteristic of the algorithm. Because of this even experienced users may choose an inappropriate parameter and make the problem unable to reach an effective solution. Some research on these parameters need to be carried on more and more. Here we choose the controlling parameters as an optimization solution to the particle swarm algorithm. Thus we can control the heuristic function in the PSO using the ordinal genetic algorithm and propose the composite particle swarm optimization algorithm. Finally we use this algorithm to solve the geometric constraint successfully. The experiment shows that the algorithm can find the best solution in a short time.

Keywords　geometric constraint solving, group intelligent algorithm, composite particle swarm optimization algorithm

---

基金项目：国家自然科学基金项目(60573182)；国家教育部博士后基金项目(200603900300)

收稿日期：2005-09-26；**改回日期**：2006-04-06

第一作者简介：曹春红(1976～　)，女，讲师。2005 年在吉林大学获博士学位。主要研究方向为几何约束求解、CAD 等，共发表论文 30 余篇。E-mail：chunhongcao_li@163.com

1 引 言

几何约束求解问题是当前基于约束设计研究中的热点问题。一个约束描述了一个应该被满足的关系,一旦用户已经定义了一系列的关系,那么在修改参数之后,系统就会自动选择合适的状态来满足约束,这种思想方法叫做基于约束的模型。如今国内外很多学者运用数值计算理论、人工智能理论、图论、自由度分析理论等对约束求解进行了深入的研究,归纳起来主要有整体求解法、稀疏矩阵法、连接分析法、规约构造法、约束传播法、符号代数法和辅助线法等[1]。

在将几何约束问题的约束方程组转化为优化模型的时候,通常需要找到一种方法来跳出局部最优解,进而找到全局最优解。为了兼顾算法的快速性和全局性,本文使用了群智能方法,而粒子群优化算法就是一种基于群智能的演化计算方法。若将粒子群算法中的控制参数的选取也作为一个优化问题,则可用常规遗传算法来控制粒子群算法中的启发式函数,即可形成复合粒子群优化算法。该算法最重要的优点是算法不会陷入局部最优,实际上控制参数的改变恰是算法的一种正反馈,并且控制 PSO (particle swarm optimization)算法收敛行特性的启发式参数不需要用户考虑。

由于群智能方法具有很强的计算鲁棒性、隐含的内在并行性、全局搜索与局部快速收敛能力,因此将群智能算法——复合粒子群算法与约束求解相结合将能大大提高约束求解的效率,并可以很自然地求解欠约束和过约束的问题,另外,还可以对一些用其他方法无法求解的几何约束问题进行求解。

2 几何约束求解

从人工智能的角度来看[2-5],设计问题本质上是一个约束满足问题。在诸多设计约束中,几何约束最具有基础性,其是表达其他设计约束的基础,也是约束管理和求解技术中必须优先解决的问题。

约束问题可以形式化为$(\boldsymbol{E},\boldsymbol{C})$[6],其中 $\boldsymbol{E}=(e_1,e_2,\cdots,e_n)$,表示几何元素,如点、线、圆等;$\boldsymbol{C}=(c_1,c_2,\cdots,c_m)$,$c_i$ 表示加在这些几何元素之间的约束。由于一般一个约束对应一个代数方程,因此约束可以表示为

$$\begin{cases} f_1(x_0,x_1,x_2,\cdots,x_n)=0 \\ \cdots \\ f_m(x_0,x_1,x_2,\cdots,x_n)=0 \end{cases} \tag{1}$$

$$\boldsymbol{X}=(x_0,x_1,\cdots,x_n)$$

x_i 为几何元素 e_i的一些参数值,如 2 维点可以表示为(x_1,x_2)。所谓约束求解就是求出 $\boldsymbol{X}$ 满足式(1)。令

$$F(\boldsymbol{X}_j)=\sum_{i=1}^{m}|f_i| \tag{2}$$

显然,若有 $\boldsymbol{X}_j$ 满足 $F(\boldsymbol{X}_j)=0$,则 $\boldsymbol{X}_j$ 满足式(1)。因此约束求解问题要转化为优化问题,只需求 $\min(F(\boldsymbol{X}_j))<\varepsilon$就可以了,$\varepsilon$ 是某个阈值。为了提高算法的速度,可用 f_i 的绝对值的和,而不是平方和来表示约束方程组。由式(2)以及用本文的算法求解 $\min(F(\boldsymbol{X}_j))<\varepsilon$ 知道,由于并不要求 $m=n$,因此该方法显然可以求解欠约束问题和过约束问题。

3 粒子群算法

粒子群优化算法是由 Eberhart 博士和 Kennedy 博士发明的一种基于群智能方法的演化计算技术,也是演化计算领域中的一个新的分支,其源于鸟群和鱼群群体运动行为的研究。

3.1 粒子群优化算法基本原理

PSO 算法是从生物种群行为特性中得到启发并用于求解优化问题的一种方法。在 PSO 算法中,每个优化问题的潜在解都可以想象成 d 维搜索空间上的一个点,可称之为“粒子”(particle)。粒子在搜索空间中以一定的速度飞行,这个速度可根据它本身的飞行经验和同伴的飞行经验来动态调整。所有的粒子都有一个被目标函数决定的适应值(fitness value),并且知道自己到目前为止发现的最好位置(particle best,记为 p_{best})和当前的位置。这个可以看作是粒子自己的飞行经验。除此之外,每个粒子还知道到目前为止整个群体中所有粒子发现的最好位置(global best,记为 g_{best})(g_{best}是在 p_{best}中的最好值),这个可以看作是粒子的同伴的经验。

3.2 PSO 算法数学描述

PSO 算法的数学描述可以是:假设在一个 d 维的目标搜索空间中,有由 m 个代表潜在问题解的粒子组成的种群 $\boldsymbol{S}=(\hat{\boldsymbol{X}}_1,\hat{\boldsymbol{X}}_2,\cdots,\hat{\boldsymbol{X}}_m)$,其中 $\hat{\boldsymbol{X}}_1=(x_{i,1},x_{i,2},\cdots,x_{i,d};i=1,2,\cdots,m)$表示第 i 个粒子在 d 维解空间的一个矢量点。将 $\hat{\boldsymbol{X}}_i$ 代入一个与求解问题相

关的目标函数就可以计算出相应的适应值。用 $\hat{P}_i=(p_{i,1},p_{i,2},\cdots,p_{i,d};i=1,2,\cdots,m)$ 记录第 i 个粒子自身搜索到的最好点(所谓最好,是指计算得到的适应值为最小,即 p_{best})。而且在这个种群中,至少有一个粒子是最好的,将其编号记为 g,则 $\hat{P}_g=(p_{g,1},p_{g,2},\cdots,p_{g,d})$ 就是种群搜索到的最好值(即 g_{best}),其中 $g\in\{1,2,\cdots,m\}$。另外,每个粒子还有一个速度变量,可以用 $\hat{V}_i=(v_{i,1},v_{i,2},\cdots,v_{i,d};i=1,2,\cdots,m)$ 表示第 i 个粒子的速度。PSO 算法一般是采用下面的公式对粒子进行操作的。

$$\hat{V}_i^{(k+1)}=\hat{V}_i^{(k)}+c_1\times r_1\times(\hat{P}_i^{(k)}-\hat{X}_i^{(k)})+c_2\times r_2\times(\hat{P}_g^{(k)}-\hat{X}_i^{(k)})\tag{3}$$

$$\hat{X}_i^{(k+1)}=\hat{X}_i^{(k)}+\hat{V}_i^{(k+1)}\tag{4}$$

其中,粒子的标号 $i=1,2,\cdots,m$;k 为迭代代数;学习因子 c_1,c_2 是两个正常数,一般取值为 2;r_1,r_2 是均匀分布于[0,1]之间的两个随机数。为了控制 $\hat{V}_i^{(k)}$ 和 $\hat{X}_i^{(k)}$ 的值在合理的区域内,需要通过指定 v_{max} 和 x_{max} 来限制。

PSO 算法是一种全局优化算法,PSO 算法的具体实现过程如下:①选定 PSO 种群规模 m;设 $x[i]$ 为种群中第 i 个粒子的位置;②设 $o[i]$ 为第 i 个粒子的适应值;③设 $v[i]$ 为第 i 个粒子的速度;④设 g_{best} 为种群最好粒子的标号;⑤设 $p_{best}[i]$ 为第 i 个粒子自身搜索到的最好点位置,设 $o_{best}[i]$ 为第 i 个粒子自身搜索到的最好适应值,即 $p_{best}[i]$ 处的适应函数值。其基本算法步骤如下:

(1) 初始化　对于种群中的第 i 个粒子($i=1,2,\cdots,m$)

① 随机初始化 $x[i]$;

② 随机初始化 $v[i]$;

③ 计算 $o[i]$,并以此初始化 $o_{best}[i]$;

④ 以种群中最好适应值的粒子标号初始化 g_{best} 来优化算法(simple particle swarm optimization, SPSO);

⑤ 以 $x[i]$ 初始化 $p_{best}[i]$。

(2) 循环迭代,直到满足 PSO 算法终止条件

① 选择算法控制参数 w;

② 对每个粒子,计算其适应值 $o[i]$。若 $o[i]<o_{best}[i]$,则 $o_{best}[i]=o[i]$,且 $p_{best}[i]=x[i]$;

③ 搜索 g_{best} 值:若 $o_{best}[i]<o_{best}[g_{best}]$,则 $g_{best}=i$;

④ 对每个粒子,依据式(5),式(6)更新 $v[i]$ 和 $x[i]$ 值。

$$V_i^{(k+1)}=\omega\times\hat{V}_i^{(k)}+c_1\times r_1\times(\hat{P}_i^{(k)}-\hat{X}_i^{(k)})+c_2\times r_2\times(\hat{P}_g^{(k)}-\hat{X}_i^{(k)})\tag{5}$$

$$\hat{X}_i^{(k+1)}=\hat{X}_i^{(k)}+\hat{V}_i^{(k+1)}\tag{6}$$

4　复合粒子群优化算法

在所有的进化算法中都包括控制其自身特性的启发式参数,尽管这些参数通常是与特定的问题相关并由用户自己定义,但合适的参数选择却需要用户丰富的经验和对研究问题所提供信息的正确判断。更重要的是,这些启发式参数会影响到算法的收敛特性[7]。基于此,即便是很有经验的用户也可能会选择不恰当的参数,从而使问题得不到有效地解决,这就越来越需要对这些参数进行研究。由于本文将 PSO 算法中的控制参数的选取也作为一个优化问题,从而可用常规遗传算法(simple genetic algorithm, SGA)来控制 PSO 算法中的启发式参数,以形成复合粒子群优化算法(composite particle swarm optimization, CPSO)。

4.1　常规遗传算法

遗传算法是模拟遗传选择和自然淘汰的生物进化过程的计算模型。常规遗传算法主要包括以下的主要处理步骤:(1)对优化问题的解进行编码;(2)适应函数的构造和应用,该适应函数基本上依据优化问题的目标函数而定;(3)染色体的组合,包括选择和交叉;(4)变异。由于新解的产生过程中可能发生基因变异,变异使某些解的编码发生变化,从而使解有更大的遍历性。本文基因组是采用实数编码,而选择操作则采用于转盘式选择(roulette wheel selection)、一点交叉及 $N(0,\sigma^2)$ 分布的变异算子,方差按算式 $\sigma^2=1-t/t_{max}$ 递减。

4.2　复合粒子群算法

复合 PSO 算法是将基本 PSO 算法和选择机制相结合而得到的,基本 PSO 算法的搜索过程在很大程度上依赖 p_{best} 和 g_{best},它的搜索区域受到 p_{best} 和 g_{best} 的限制。在通常的进化算法中,选择机制用来选择相对较好的区域和淘汰较差的区域,以便可以更合理地分配有限的资源。复合 PSO 算法首先计算每个个体基于当前位置的适应值,并将这些适应值排序;然后将群体中一半适应值差的个体的当前位置和速度替换为另一半好的个体的当前位置和速度,但保留每个个体的最好位置 p_{best}。虽然群体搜索可集中到相对较优的空间,但还受到个体自身以

前最好位置的影响。

在优化过程中，CPSO算法的3个启发式参数ω、c_1和c_2形成3维空间，可采用3个实数编码的遗传算法对其进行优化。其主要实施步骤为：

(1) 选定GA(genetic algorithm)种群规模n_c、交叉率P_c和变异率P_m，并给定问题的求解精度ε或最大进化代数T；

(2) 初始化GA种群；

(3) 计算GA个体适应值；

① 选定PSO种群规模M；

② 初始化粒子速度和位移；

③ 计算每个粒子的适应值；

④ 比较各个粒子的适应值，并记录其自身最优值p_i，和群体最优粒子标号g；

⑤ 根据前面式(3)和式(4)改变粒子的速度和位移；

⑥ 检查是否满足PSO算法终止条件，若否，则转至步骤④；

继续；若是，则求出最优值；

⑦ PSO算法所找到的最优值即为GA个体的适应值；

(4) 根据适应值计算每个个体的选择概率，通常按线性排名计算；

(5) 进行选择、交叉和变异操作，生成新一代种群；

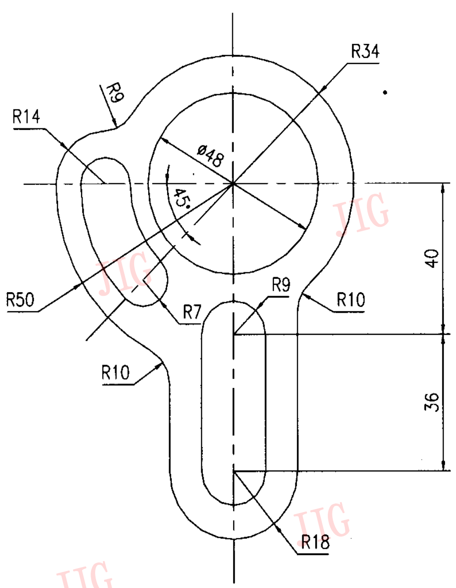
(6) 检查是否满足GA终止条件，若否，则转至步骤3，继续；若是，则求出最优解，即为参数估计值。

很明显，由于要对PSO算法控制参数进行优化，因此该算法要比常规的粒子群优化算法慢。考虑到遗传算法主要用来优化3维的控制参数，因此对SGA的种群规模和迭代次数进行控制，可有效控制其所需计算时间。该算法最重要的优点有以下2个：(1)算法不会陷入局部最小，实际上控制参数的改变恰是算法的一种正反馈；(2)控制PSO算法收敛行特性的启发式参数不需要用户考虑，并且从理论上分析，当进化足够多代数时，算法应收敛到最优解。

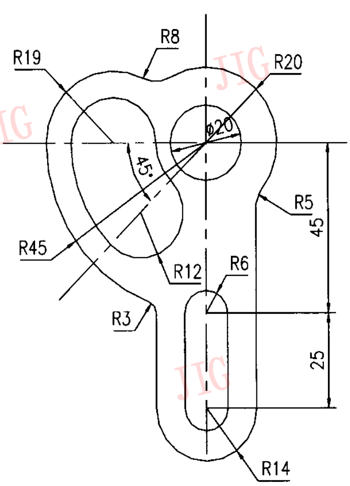
4.3 实验结果

图1(a)和图2(a)是设计的原图，图1(b)与图2(b)都是有部分尺寸或者角度有了变化后，用复合粒子群算法得到的新的图形。从图1、图2的图可看出，在用户定了约束关系后，系统在参数修改之后就可以自动地满足约束条件。

从表1可以看出，遗传算法和常规粒子群优化算法、复合粒子群优化算法在有限的代数内都可以找到最优解。另外，测试还表明，两种粒子群优化算法质量都好于常规的遗传算法，而且复合粒子群优化算法可以在更少的代数内收敛到最优解。这表明复合粒子群优化算法比遗传算法具有更高的效率。但应指出，本文所有利用复合粒子群的计算实例中，



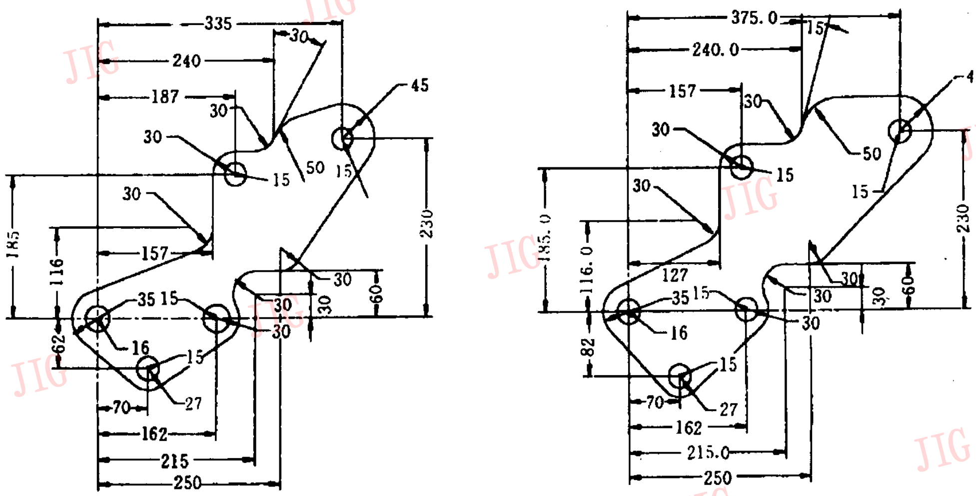
(a) 原设计图



(b) 变动设计图

图1 设计草图及用复合粒子群算法得到的新图形

Fig. 1 Primary sketch and the sketch using composite particle swarm optimization algorithm



(a) 原设计图　　(b) 变动设计图

图 2　另一设计草图及用复合粒子群算法得到的新图形

Fig. 2　Another primary sketch and the sketch using composite particle swarm optimization algorithm

表 1　遗传算法、常规粒子群算法、复合粒子群算法的实验结果比较

Tab. 1　The comparison of genetic algorithm, simple PSO and composite PSO

算法	进化的代数	占用 CPU 时间 (s)	出现最好解的代数
遗传算法	500	2 880	488
常规粒子群算法	20	10	15
复合粒子群算法	12	100	8

计算精度的提高是以牺牲进化速度为代价的，亦即该算法适合实时性要求不是很高的情况。

参考文献(References)

1　Yuan Bo. The Research and Implement of Deometric Constraint Solving[D]. Beijing: Tsinghua University, 1999.［袁波. 几何约束求解技术研究与实现[D]. 北京：清华大学，1999.］

2　Fu Jing-sun, Cai Zi-xing, Xu Guang-you. Intelligence and Its Application[M]. Beijing: Tsinghua University Press, 1987.［傅京孙，蔡自兴，徐光祐. 智能及其应用[M]. 北京：清华大学出版社，1987.］

3　Zhou Ji, Zha Jian-zhong, Xiao Ren-bin. Intelligence Design[M]. Beijing: Higher Education Press, 1998.［周济，查建中，肖人彬. 智能设计[M]. 北京：高等教育出版社，1998.］

4　Liu Sheng-li, Tang Min, Dong Jin-xiang. Two spatial constraint solving algorithms [J]. Journal of Computer-Aided Design and Computer Graphic, 2003, **15**(8): 1011～1029.［刘生礼，唐敏，董金祥. 两种空间约束求解算法[J]. 计算机辅助设计与图形学学报，2003，**15**(8)：1011～1029.］

5　Parsopoulos K E, Vrahatis M N. Recent approaches to global optimization problems through particle swarm optimization [J]. Natural Computing, 2002, (1): 235～300.