

结合非高斯随机分布的粒子群优化算法在风力机叶片优化设计中的应用

孙芳锦 梁爽

(辽宁工程技术大学 建筑工程学院 辽宁 阜新 123000)

摘要: 为克服传统粒子群优化算法易陷入局部解的缺点,提出一种结合非高斯随机分布的粒子群优化算法,对风力机叶片进行优化设计。在更新速度前对每一个粒子进行限制检验,寻找全局最优解。对某 1.3 MW 风力机叶片进行优化设计,优化结果与原始叶片、传统粒子群优化算法结果进行比较。和原始叶片输出功率相比,采用结合非高斯随机分布的粒子群算法,叶片的年输出功率增加了 5.3%,计算耗时为传统粒子群优化算法的 65%。与传统粒子群优化算法相比,结合非高斯随机分布的粒子群算法的计算残差随时间步长的增加而迅速减小,说明计算效率提高,保证了全局最优解。结合非高斯随机分布的粒子群优化算法是风力机叶片优化设计的高效稳定方法。

关键词: 风力机叶片; 优化设计; 粒子群优化算法; 非高斯随机分布

中图分类号: TK83 文献标识码: A

DOI:10.16146/j.cnki.rndlgc.2015.02.030

引言

风力机是将风能转化为机械能的一种动力机械,叶片是风力机中最重要的部件之一,叶片的气动外形决定了风能的转换效率,因而风轮叶片气动外形的优化设计技术在风力机设计制造中占有至关重要的地位。风力机的叶片优化设计涉及复杂的气动性能计算及搜索寻优过程,其中优化算法是直接决定叶片设计优劣的重要指标之一^[1-2]。

对于叶片优化设计的优化算法研究,目前国内外较常用的包括遗传算法(GA)和粒子群优化算法(简称 PSO)两类。而粒子群优化算法是近年来发展起来的一种新型高效的群体智能优化算法^[3],因其没有变异环节,靠个体经验和群体经验搜索最优目标,算法容易实现等优点,近年来得到广泛发展。

粒子群优化算法也被应用于不同风力机叶片的优化设计中^[4-6]。但粒子群优化算法最大的缺点在于迭代过程中由于对所有解都进行最优化搜索,这样会导致早熟收敛现象,不能很好的进行全局搜索,而陷入局部解收敛情况^[7]。

为克服传统粒子群优化算法的上述缺点,本研究将非高斯随机分布与粒子群算法相结合,对风力机叶片进行优化设计。以动量—叶素理论为基础,利用结合非高斯随机分布的粒子群算法开发了优化设计程序,以风力机最大年发电量为目标,对某 1.3 MW 风力机叶片进行优化设计,并将结合非高斯随机分布的粒子群优化算法的结果与原始风力机叶片以及传统粒子群算法的优化结果进行了比较。

1 结合非高斯随机分布的粒子群优化算法

传统粒子群优化算法中,每个粒子用速度、位置和适应度值三项指标表示其特征,通过不断更新个体极值和全局极值在搜索空间搜寻粒子的位置来找到全局最优解。其主要迭代计算公式为:

$$V_{id}^{k+1} = \omega V_{id}^k + c_1 r_1 (P_{id}^k - X_{id}^k) + c_2 r_2 (P_{gd}^k - X_{id}^k) \quad (1)$$

$$X_{id}^{k+1} = X_{id}^k + V_{id}^{k+1} \quad (2)$$

式中: ω —惯性权重; $d = 1, 2, \dots, D$; $I = 1, 2, \dots, n$; k —当前迭代次数; V_{id} —粒子的速度; c_1, c_2 —加速度因子,非负的常数; r_1, r_2 —分布在 $[0, 1]$ 之间的随机数,为防止粒子盲目搜索,一般将其位置和速度限定在搜索范围的最大值和最小值 $[-X_{\max}, -X_{\min}]$ 之间。

收稿日期: 2014-07-18; 修订日期: 2014-08-18

基金项目: 国家自然科学基金资助项目(51108345); 辽宁省教育厅基金资助项目(L2013134); 同济大学土木工程防灾国家重点实验室开放基金资助项目(SLDRCE-MB-04)

作者简介: 孙芳锦(1981-),女,辽宁沈阳人,辽宁工程技术大学副教授,工学博士。

传统粒子群优化算法虽然是全局优化搜索算法,但它在迭代过程中由于对所有解都进行最优化搜索,这样会导致早熟收敛现象,不能很好的进行全局搜索,陷入局部解收敛情况^[7]。为了克服传统粒子群优化算法的这一缺点,本研究结合风力机转轮叶片优化设计的相应内容,对粒子群算法进行相适应性的改进:将非高斯随机分布与 PSO 相结合,引入非高斯随机分布可以提高搜索效率,这里的非高斯随机分布采用 Levy 分布,因为 Levy 分布可以实现粒子在搜索空间内的多步跳,达到提高搜索效率的目的。Levy 非高斯随机分布的详细介绍可以参考文献[8]。

结合非高斯随机分布的粒子群优化算法中,每个粒子的新状态为:

$$X^{t+1} = X^t + \gamma \otimes \text{Levy}(\omega) \quad (3)$$

式中: γ —步长,与所求解问题的规模有关,这里取所有粒子直径在区间(0 2]间的任意数,即: $\gamma = \text{Ran}(D)$; D —粒子的直径; $\text{Levy}(\omega)$ —Levy 分布。

那么生成的 n 个样本如公式(4)所示^[8-9],

$$n = \text{Ran}(D) \otimes \text{Levy}(\omega) \sim 0.01 \frac{\varepsilon}{|\eta|^{1/\lambda} (x_j^t - gbest^t)} \quad (4)$$

其中 ε 和 η 是由正态分布推导出,即 $\rho \sim N(0, \sigma_\rho^2)$, $\eta \sim N(0, \sigma_\eta^2)$ 。

$$\sigma_\rho = \left\{ \frac{\Gamma(1 + \omega) \sin\left(\frac{\pi\omega}{2}\right)}{\Gamma\left[\left(\frac{1 + \omega}{2}\right)\right] \omega^{(\lambda-1)/2}} \right\}^{1/\lambda} \quad \sigma_\eta = 1 \quad (5)$$

其中 Γ 是标准 Gamma 函数。此时再利用式(4)得到的 n 值更新粒子的位置,并估计粒子新位置的适应度值。如果计算得到的粒子新适应值优于原始个体极值(简称 pbest),那么此时更新个体极值,并将试验限值定为零;否则将试验限值增加 1 继续估计粒子新位置的适应度值,重复上述步骤,直至得到的新适应值优于原始个体极值为止。

2 优化目标及约束条件

2.1 目标函数

为使用结合非高斯随机分布的粒子群优化算法寻找全局最优解,这里将风力机的年平均发电量作为优化设计的目标函数。这里取 Weibull 分布密度

函数来拟合风况,通过对不同风速下的风力机风能利用系数 C_p 的求解,利用积分求出年发电量的平均功率,进而求得风力机的年平均发电量。

其计算公式为:

$$P = F(b, \phi) = \int_{V_1}^{V_2} \frac{1}{2} \rho v^2 \pi R^2 C_p(v) W(v) dv \quad (6)$$

式中: P —年平均功率; V_1 —切入风速; V_2 —切出风速; v —无穷远来流风速值; $W(v)$ —Weibull 分布密度函数。

2.2 优化设计变量和约束条件

在风力机叶片的气动外形优化设计中,主要的优化对象是截面的翼型、弦长 b 和扭角 ϕ 。本研究将对选定后的翼型系列采取不以恒定攻角设计处理^[10]。对于同种翼型,将在某雷诺数下一定攻角 α 范围内,将升阻系数用显含攻角的形式表达为:

$$C_L = X_1(\alpha) \quad (7)$$

$$C_D = X_2(\alpha) \quad (8)$$

式中: $X_1(\alpha)$ 、 $X_2(\alpha)$ —关于攻角 α 的 Newton 插值多项式。

攻角的关系式为:

$$X(\alpha, V_\infty, \omega, R, b, \phi) = 0 \quad (9)$$

那么不显含攻角的功率系数表达式为:

$$X_p = X_p(V_\infty, \omega, R, b, \phi) \quad (10)$$

式中: V_∞ —来流速度; ω 是风轮转速; R —风轮半径,均为已知量。这样就可以确定优化设计变量为叶片弦长 b 和扭角 α 。

对弦长 b 和扭角 ϕ 按贝塞尔曲线进行分布,约束条件为:

$$\begin{cases} b_{\min} \leq b_r \leq b_{\max} \\ b_{\min} \leq b_t \leq b_{\max} \\ r_{\min} \leq r_{1c} \leq r_{2c} \leq r_{\max} \\ \phi_{\min} \leq \phi_r \leq \phi_{\max} \\ \phi_{\min} \leq \phi_t \leq \phi_{\max} \\ \phi_{\min} \leq \phi_{1c} \leq \phi_{2c} \leq \phi_{\max} \end{cases} \quad (11)$$

式中: b_r 、 ϕ_r —叶根处的弦长和扭角; b_t 、 ϕ_t —叶尖处的弦长和扭角; r_{\min} 、 r_{\max} 、 ϕ_{1c} 、 ϕ_{2c} —Bezier 的控制点坐标。

这里采用动量叶素(Blade element momentum, BEM) 理论建立风力机空气动力学模型。在建立空

气动力学模型过程中就需要考虑叶尖损失的影响。受篇幅所限,具体公式可参考文献 [11]。

2.3 优化设计程序和流程

结合非高斯分布的粒子群优化算法,叶片气动优化程序基本流程如图 1 所示。

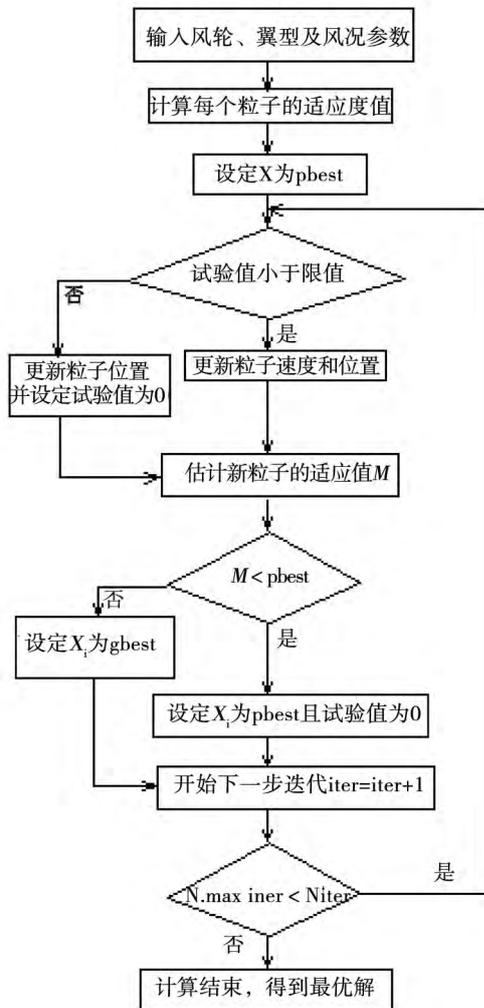


图 1 优化设计流程图

Fig. 1 Optimized design flow chart

结合非高斯随机分布的粒子群优化算法进行寻优搜索,与传统粒子群优化算法不同,结合非高斯随机分布的粒子群优化算法在进行速度更新之前,对每一个粒子进行限制检验。如果当前粒子没有超出限值,则对速度正常更新,同时其位置也予更新。如果粒子速度超过了限值,那么对超过搜索空间边界的位置进行设置,使其值等于边界值。因此本研究提出的结合非高斯随机分布的粒子群优化算法的叶片气动优化程序的关键步骤有:

(1) 对每一个粒子设定一个适应度(限)值,超过该限值的粒子群则采用非高斯随机分布在搜索空间中进行重新分布。在结合非高斯随机分布的粒子群优化算法中,起初粒子群在搜索空间中是任意分布的,需要计算这些粒子群的适应度值,并利用这些适应度值计算粒子群的个体极值(p_{best}),即当前单个粒子找到的最优解和全局极值(g_{best}),也就是整个种群目前找到的最优解。

(2) 对新确定的粒子计算其适应度值,如果计算得到的粒子新适应度值比原始 p_{best} 优,那么它们就作为 p_{best} 并作为限值;如果新适应度值与原始 p_{best} 相比并没有改进,那么粒子的限值就在原有值基础上增加 1。同样,如果计算得到的新适应度值比 g_{best} 小,那么它就作为粒子的 g_{best} 值。执行同样的步骤直至迭代结束。

(3) 限值的选定对于计算结果和计算效率有着重要影响,因此在选取限值时应该兼顾计算结果的优化和计算耗时,权衡后选用较合理的限值进行计算。

3 算例分析

应用结合非高斯随机分布的粒子群优化算法,进行风机叶片的优化设计。风轮参数如表 1 所示,风轮翼型采用 NACA6412,把叶片分成 5 个截面,每个截面都按 $0.2R$ (R 为风轮半径) 距离截取,计算出每个截面的弦长和扭角数值,并与原有叶片进行了对比。

表 1 设计参数值

Tab. 1 Design parameter values

参数	数值	参数	数值
风轮直径/m	60	尖速比 λ	6
风轮锥角/(°)	0	额定功率/MW	1.3
额定转速/ $r \cdot \min^{-1}$	19	参考高度/m	60
额定风速/ $m \cdot s^{-1}$	15	轮毂标高/m	50
切入风速/ $m \cdot s^{-1}$	4	切出风速/ $m \cdot s^{-1}$	25

优化模型基本参数选择为:种群规模 40,学习因子 $C1 = C2 = 2$,惯性权重如:

$$w = \left(\frac{N_{\max_iter} - N_{iter}}{N_{\max_iter}} \right)$$

式中: N_{max_iter} —最大迭代数; N_{iter} —目前的迭代数。Weibull 形状参数为 1.85,所有粒子在初始范围内进行初始化处理。

结合非高斯分布的粒子群优化算法中,限值 M 的选定对于优化设计结果有重要影响,因此这里对比了不同限值情况下叶片弦长的优化设计结果,并与原有叶片结果、传统粒子群优化算法结果进行了对比,如图 2 所示。

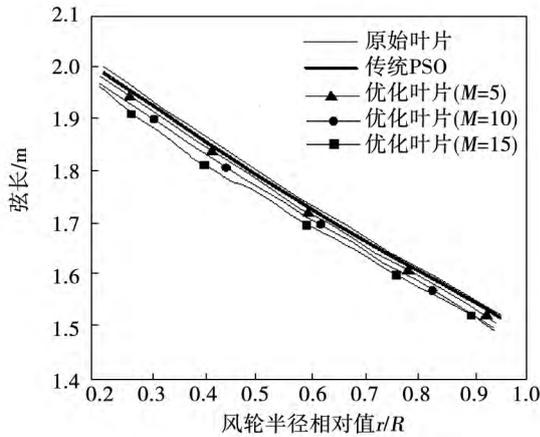


图 2 叶片弦长对比

Fig. 2 Comparison of blade chord lengths

由图 2 可以看到,结合非高斯分布的粒子群优化算法和传统粒子群优化算法优化设计后的叶片弦长都小于原始的风力机叶片的弦长,但采用结合非高斯分布的粒子群优化算法的叶片弦长要小于传统粒子群优化算法,说明与比传统粒子群优化算法相比,结合非高斯分布的粒子群优化算法优化后的叶片具有更好的气动特性,特别是叶片弦长的减小对于叶片质量的聚集区—叶片根部来讲,很好地实现了风轮输出功率和成本的控制,证明了结合非高斯分布的粒子群优化算的先进性。

结合非高斯分布的粒子群优化算法中,对于限值 M 来说,可以看出随着限值的增大优化效果越好,这是由于非高斯随机 Levy 分布可以实现粒子在搜索空间内的多步跳,其搜索范围的扩大使得结果越精确。但与此同时计算时间也在增大, $M = 15$ 时的计算时间是 $M = 10$ 时计算时间的 1.2 倍, $M = 10$ 时的计算时间是 $M = 5$ 时计算时间的 0.7 倍。综合考虑,这里选定限值 $M = 10$ 是比较合理的,可以达到提高优化结果和计算成本间的平衡。因此在下面

的优化计算中,都按照限值 $M = 10$ 进行计算。

图 3 给出了分别采用结合非高斯分布的粒子群优化算法、传统粒子群优化算法优化后,叶片扭角与原叶片的对比。可以看到,结合非高斯分布的粒子群优化算法、传统粒子群优化算法优化后的叶片扭角与原有风力机的扭角相比变化不大,这是由于尽管扭角的减小可以增加风轮的输出功率,但同时会增加轴向推力,从而缩短叶片寿命、增加成本,所以采用不同优化算法优化后叶片的扭角的变化并不大。

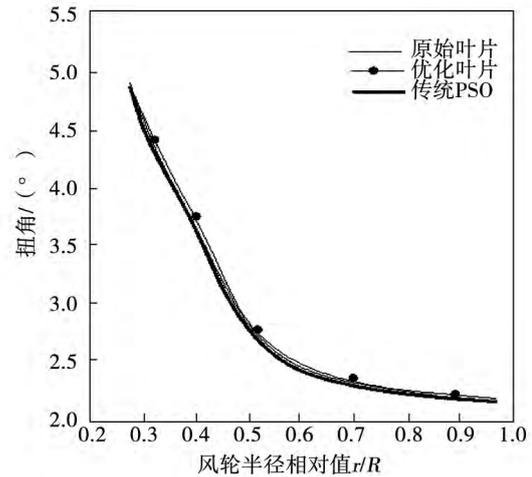


图 3 叶片扭角对比

Fig. 3 Comparison of blade twist angles

图 4 给出了分别采用结合非高斯分布的粒子群优化算法、传统粒子群优化算法优化后,风力机叶片输出功率和原有叶片输出功率进行了对比。可以看出,采用结合非高斯分布的粒子群优化算法优化后,在不同风速下,风力机叶片的输出功率均较原有风力机叶片和传统粒子群优化算法优化后的叶片输出功率有所增加,说明结合非高斯分布的粒子群优化算法较传统粒子群优化算法能更好的进行风力机叶片的优化设计,且结合非高斯分布的粒子群优化算法优化后,叶片的输出功率一直控制在额定功率 1.3 MW 内,说明结合非高斯分布的粒子群优化算法优化后的风力机具有良好的失速控制特性。

进一步对比发现,采用传统粒子群优化算法优化后和原有叶片的输出功率相比,风轮在额定风速 15 m/s 时的输出功率增加了 2.1%,年输出功率增加了 5.3%;而采用结合非高斯分布的粒子群优化

算法优化后和原有叶片的输出功率相比,在额定风速 15 m/s 时,风力机叶片的输出功率增加了 3.4%, 年输出功率增加了 7.6%。因此从计算效率看,结合非高斯分布的粒子群优化算法的计算耗时约为传统粒子群优化算法耗时的 65%,结合非高斯分布的粒子群优化算法的计算效率大大提高,这是由于非高斯 Levy 分布实现了粒子在搜索空间内的多步跳,提高了搜索效率,从而节约了计算耗时。因此可以看出,结合非高斯分布的粒子群优化算法与传统粒子群优化算法相比,在进行风力机叶片的优化设计时,是一种更优、更高效的方法。

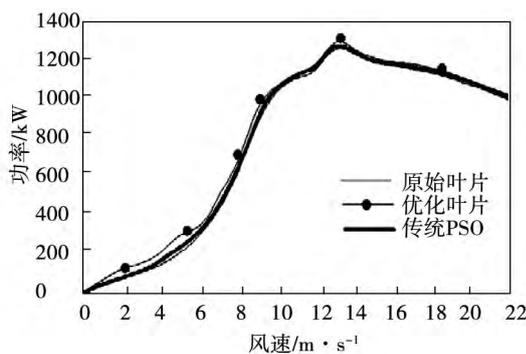


图 4 优化设计前后的输出功率对比
Fig. 4 Comparison of the output power before and after optimization

为了说明结合非高斯分布的粒子群优化算法的计算效率,这里与采用传统粒子群优化算法进行优化设计的计算收敛情况进行了对比,表 2 给出了不同方法计算的残差。

表 2 不同方法计算残差对比

Tab. 2 Comparison of residuals calculated by using different methods

残 差	时间步长				
	10	30	40	70	90
传统 PSO	10^1	10^{-0}	10^{-1}	10^{-1}	10^{-1}
结合非高斯分布的 PSO	10^{-1}	10^{-3}	10^{-5}	10^{-6}	10^{-7}

从表 2 中可以看出,随着时间步长的增加,传统粒子群优化算法的计算残差变化并不明显,且当时间步长到达某一值时,计算残差不再变化,说明传统 PSO 方法的收敛性差、陷入了局部收敛的情况;而结合非高斯随机分布的粒子群优化算法则随着时间步

长的增加,计算残差迅速减小,且减小幅度较大,说明结合非高斯随机分布的粒子群优化算法很好的克服了传统 PSO 方法收敛性差、易陷入局部解收敛的缺点,具有收敛性好的特点,原因是结合非高斯随机分布的粒子群优化算法中粒子的 pbest 值是任意选择的,而不像传统粒子群优化算法那样,需要从粒子的 pbest 和 gbest 值中通过学习得到,因此结合非高斯随机分布的粒子群优化算法保证了种群的多样性,避免了解陷入局部最小值的情况发生。

为说明结合非高斯随机分布的粒子群优化算法的计算稳定性,这里在不同时间步长时,分别对传统粒子群优化算法和结合非高斯随机分布的粒子群优化算法的误差进行了统计分析,得到了不同优化算法的计算误差的平均值和均方差。这里计算的最优值均取为零,且当结果小于 10^{-6} 时认为计算结果收敛,统计结果如表 3 所示。

表 3 不同方法误差统计结果分析

Tab. 3 Analysis of the statistic results of errors calculated by using different methods

时间步长	传统 PSO		结合非高斯分布的 PSO	
	均值	均方差	均值	均方差
20	$8.67e-1$	$2.5e-1$	$4.35e-3$	$3.43e-3$
40	$6.45e-1$	$3.23e-1$	$5.34e-5$	$3.76e-5$
50	$6.11e-1$	$8.23e-2$	$1.23e-5$	$6.45e-6$
60	$3.25e-1$	$5.23e-2$	$5.45e-6$	$9.45e-7$
70	$2.23e-1$	$6.34e-2$	$4.76e-7$	$1.23e-7$

分析表 3 可以看出,采用不同的时间步时,结合非高斯随机分布的粒子群优化算法得到的误差的均值和均方差均小于传统粒子群优化算法,说明结合非高斯随机分布的粒子群优化算法的优化结果优于传统粒子群优化算法,可以获得更稳定的结果。这是由于结合非高斯随机分布的粒子群优化算法在进行速度更新之前,对每一个粒子进行限制检验,避免了在低效空间内进行搜索,从而大大提高了计算效率和收敛性。

4 结 论

(1) 提出了结合非高斯随机分布的粒子群优化算法。对某 1.3 MW 风力机叶片进行了优化计算和

分析。计算结果表明,与传统粒子群优化算法相比,结合非高斯随机分布的粒子群优化算法由于扩大了粒子的搜索范围,优化结果随着搜索限值的增大而提高,使得优化结果更加精确。

(2) 结合非高斯分布的粒子群优化算法进一步提高了风力机叶片的输出功率:结合非高斯随机分布的粒子群优化算法优化叶片后,在额定风速 15 m/s 时,风力机叶片的输出功率增加了 3.4%,年输出功率增加了 7.6%,而传统粒子群优化算法的相应结果分别增加了 2.1% 和 5.3%,且结合非高斯随机分布的粒子群优化算法的计算耗时约为传统粒子群优化算法耗时的 65%,这是由于结合非高斯随机分布的粒子群优化算法中的非高斯 Levy 分布实现了粒子在搜索空间内的多步跳,提高了搜索效率,从而节约了计算耗时。

(3) 在计算效率和计算稳定性方面,与传统粒子群优化算法相比,随着时间步的增加,结合非高斯随机分布的粒子群优化算法的计算残差迅速减小,说明结合非高斯随机分布的粒子群优化算法克服了传统粒子群优化算法的收敛性差、易陷入局部解收敛的缺点;且结合非高斯随机分布的粒子群优化算法可以获得更稳定的结果。

参考文献:

- [1] 包耳,邵晓荣,刘德庸. 风力机叶片设计的新方法[J]. 机械设计 2005 22(2): 24-26.
BAO Er, SHAO Xiao-rong, LIU De-yong. New method for designing wind turbine blades [J]. Journal of Mechanical Design, 2005, 22(2): 24-26.
- [2] 申振华,于国亮. 翼型弯度对风力机性能的影响[J]. 动力工程 2007 27(1): 136-139.
SHEN Zhen-hua, YU Guo-liang. Influence of the bending degree of the airfoil on the performance of a wind turbine [J]. Journal of Power Engineering 2007 27(1): 136-139.
- [3] N. Nouaouria, M. Boukadoum. Particle swarm classification for high dimensional data sets, in: Proceedings of 22th International IEEE Conference on Tools with Artificial Intelligence [C], 1(27-29): 87-93, October 2010, Arras, France.
- [4] 彭茂林,杨自春,曹跃云等. 基于贝赛尔曲线和粒子群算法的涡轮叶片型线参数化建模[J]. 中国电机工程学报, 2012, 32(32): 101-108.
PENG Mao-lin, YANG Zi-chun, CAO Yue-yun et al. Parameterization modelling of turbine blades based on Bezier curves and particle swarm algorithm [J]. Proceedings of China Electric Machinery Engineering 2012 32(32): 101-108.
- [5] 刘雄,陈严,叶枝全. 水平轴风力机风轮叶片优化设计模型研究[J]. 汕头大学学报(自然科学版), 2006, 21(1): 44-49.
LIU Xiong, CHEN Yan, YE Zhi-quan. Research of a model for optimized design of wind wheel blades of a horizontal shaft wind turbine [J]. Journal of Shantou University (Natural Science Edition) 2006 21(1): 44-49.
- [6] C. C. Liao, X. L. Zhao, J. Z. Xu. Blade layers optimization of wind turbines using FAST and improved PSO Algorithm [J]. Renewable Energy 2012 42: 227-233.
- [7] 王维刚,刘占生,倪红梅. 求解混合变量约束优化问题的改进粒子群算法[J]. 系统仿真学报 2012(6): 1175-1179
WANG Wei-gang, LIU Zhan-sheng, NI Hong-mei. Improved particle swarm algorithm for seeking solutions to the mixed variable and constraint optimization problems [J]. Journal of System Simulation, 2012(6): 1175-1179.
- [8] Mehrdad Ghaemi, Zahra Zabihinpour, Yazdan Asgari. Computer simulation study of the Levy flight process [J]. Physica A 2009, 388: 1509-1514.
- [9] J. Rada-Vilela, M. Zhang, W. Seah. A performance study on synchronicity and neighborhood size in particle swarm optimization [J]. Soft Comput 2013 17(6): 1019-1030.
- [10] 刘占芳,周振凯,邓智春. 20 kW 定桨距风力机叶片优化设计[J]. 农业机械学报 2011(5): 121-124.
LIU Zhan-fang, ZHOU Zhen-kai, DENG Zhi-chun. Optimized design of the blades in a 20 kW constant-pitch wind turbine [J]. Journal of Agricultural Machinery 2011(5): 121-124.
- [11] 贺德馨. 风工程与工业空气动力学[M]. 北京: 国防工业出版社 2006.
HE De-xin. Wind engineering and industrial aerodynamics [M]. Beijing: National Defense Industry Press 2006.

(单丽华 编辑)

the second generation serving as the prototype a calculation was performed of the transient forces produced by the water hammer in the system. It has been found that when the containment dome sprinkling system is used to conduct a direct sprinkling the transient forces produced by the water hammer in the pipelines are far less than the self-weight of the sprinkling vertical tubes and annular tubes when they are filled with water. The displacement of the pipelines arisen from the water hammer-resulted transient forces is very small thus the force resulted from such a displacement is within the range bearable by the supporters and hangers on the pipelines. **Key Words:** nuclear power plant (NPP) ,sprinkling inside a containment dome ,water hammer

结合非高斯随机分布的粒子群优化算法在风力机叶片优化设计中的应用 = **Applications of the Particle Swarm Optimization Algorithm in Combination with Non-Gaussian Stochastic Distribution in the Optimized Design of Wind Turbine Blades** [刊 汉] SUN Fang-jin ,LIANG Shuang (College of Architectural Engineering , Liaoning Engineering Technology University , Fuxin , China , Post Code: 123000) // Journal of Engineering for Thermal Energy & Power. - 2015 , 30(2) . - 291 - 296

To overcome the shortcoming of the traditional particle swarm optimization algorithm , i. e. easy to be trapped in the solutions for local areas , proposed was a particle swarm optimization algorithm in combination with non-Gaussian stochastic distribution to conduct an optimized design of wind turbine blades. Before the particle velocity had been updated a limitation check was performed of each particle to seek an optimum solution in a comprehensive way. An optimized design was performed of the blades of a 1.3 MW wind turbine and the optimization results were compared with those of the original blades obtained by using the traditional particle swarm optimization algorithm. Compared with the output power of the original blades , the annual output power of the blades optimized by using the particle swarm optimization algorithm in combination with non-Gaussian stochastic distribution had increased by 5.3% and the time consumed by the calculation was 65% of the time consumed by using traditional particle swarm optimization algorithm. Compared with the traditional particle swarm optimization algorithm , the calculation residue of the particle swarm optimization algorithm in combination with the non-Gaussian stochastic distribution decreased rapidly with an increase of the time step , indicating that the calculation efficiency has been enhanced , thus guaranteeing the optimum solution in a comprehensive way. The particle swarm optimization algorithm in combination with the non-Gaussian stochastic distribution offers an even higher and more stable method for design of wind turbine blades. **Key Words:** wind turbine blade , optimized design , particle swarm optimization algorithm , non-Gaussian stochastic distribution