

太阳能路灯光伏板的朝向设计问题

摘要

在能源和环境问题日益凸显的今天，太阳能作为一种清洁可再生能源，其各种利用技术的研发成为一个重要的课题。太阳能路灯便是利用太阳进行发电，其光伏板接受太阳辐射并将辐射能量转化为电能输出并储存到蓄电池中。在这过程中，光伏板的朝向及倾角会直接影响到电能的利用。本文以光伏板的朝向和倾角作为研究对象，在忽略了散射辐射的情况下，建立起计算光伏板表面太阳辐射强度和辐射总能量的数学模型，并在单个或多个不同的目标下求解最优朝向与最优倾角。

对于问题一，根据背景与题目描述，得到目标地区的经纬度与时区，并建立描述目标地区地平面太阳辐射强度与时间的关系的数学模型。结合附件数据与衰减系数恒定的假设，在清洗不合理数据后求解出目标区域的**衰减系数为** -5.6769×10^{-4} 。利用几何关系计算出太阳光线方向在赤道坐标系和地平坐标系下的坐标数据，进而计算光伏板所受的太阳直射强度。计算每月 15 日光伏板所受的最大太阳直射强度，并利用梯形法积分得到光伏板表面太阳辐射总能量。结果显示了最大太阳辐射强度和太阳辐射总能量与月份的特殊关系，展示了**夏季时倾角较大的光伏板最大太阳辐射强度和辐射总能量降低等**特殊性质

对于问题二，目标为太阳辐射日均总能量最大，要求求解最优朝向，其中隐含求解最优倾角。采用 **PSO(粒子群算法)** 来求解此多输入变量的单目标优化问题。最终求得**最优朝向为正南方向，最佳倾角为 27 度，日均直射辐射总能量为 3.96026(KW · h)**

对于问题三，本文针对此多目标优化问题，采用了两种不同的方法进行最优值计算。本文首先将两个目标函数赋予权重，将其转换为单目标优化问题，运用 PSO(粒子群算法) 进行求解最优值。本文还运用了 **MOPSO(多目标粒子群算法)** 来实现自适应权重调整，使得赋权策略中决策者影响因素减小。最后，我们调整全体粒子个数，解出密度较大的 **Pareto 解集**，并找到最优解集密度最大的范围。最后，我们对建立的模型进行了中肯的评价，细致讨论其优缺点。

关键字： 多目标优化 PSO(粒子群算法) MOPSO(多目标粒子群算法) Pareto 解集

目录

一、 问题背景和重述	4
1.1 问题背景	4
1.2 问题重述	4
二、 模型假设	4
三、 符号说明	5
四、 问题一的模型的建立和求解	5
4.1 问题分析	5
4.2 模型建立	6
4.2.1 赤道坐标系	6
4.2.2 地平坐标系	6
4.2.3 太阳直射强度与辐射能量	7
4.3 模型求解	7
4.4 求解结果	8
4.4.1 目标区域数据	8
4.4.2 衰减系数	8
4.4.3 最大直射强度和辐射总能量	9
4.4.4 结果分析	10
五、 问题二的模型的建立和求解	11
5.1 问题分析	11
5.2 模型建立	11
5.2.1 PSO(粒子群算法)	11
5.3 求解结果	12
六、 问题三的模型的建立和求解	14
6.1 问题分析	14
6.2 模型建立	14
6.2.1 多目标赋权的 PSO 优化算法	14
6.2.2 多目标粒子群算法 MOPSO	15
6.3 模型求解	15
6.3.1 多目标赋权的 PSO 优化算法	15

6.3.2 多目标粒子群算法 MOPSO	15
6.4 求解结果	16
6.4.1 多目标赋权的 PSO 求解结果	16
6.4.2 多目标粒子群算法 MOPSO 求解结果	17
七、模型的评价	18
7.1 模型的优点与创新	18
7.2 模型的缺点与改进	18
参考文献	18
A 附录 文件列表	20
B 附录 代码	20

一、问题背景和重述

1.1 问题背景

太阳能电池板通过支架固定在太阳能路灯的灯杆上端。太阳能电池板也叫光伏板，它利用光伏效应接收太阳辐射能并转化为电能输出，经过充放电控制器储存在蓄电池中。蓄电池储电量会受到太阳直射辐射总能量的影响，储电效率则会受到太阳直射强度的影响。太阳辐射由直射辐射和散射辐射组成，其中直射辐射对聚集太阳能系统起到了至关重要的影响。

由于光伏板的朝向将直接影响到光伏板获得太阳辐射能量的多少，因此为了提高路灯蓄电池的储电量和储电效率，设计出光伏板的最优朝向至关重要。

1.2 问题重述

1. 某城区位于北纬 $30^{\circ}35'$ ，东经 $114^{\circ}19'$ ，附件 sheet1 给出了该城区 2023 年 5 月 23 日晴天条件下测得地表水平面受到的太阳直射强度值，要求计算 2025 年每月 15 日在晴天条件下一块面积 1m^2 的光伏板朝向正南方、水平倾角分别为 20° 、 40° 、 60° 时，受到的最大太阳直射强度和太阳直射辐射总能量。

2. 设计该城区固定安装光伏板的朝向，使光伏板在晴天条件下受到的太阳直射辐射日均总能量最大，从而使路灯蓄电池储电量最大化。

3. 综合考虑太阳直射强度对电能转换效率以及太阳直射辐射总能量对储电量的影响，设计出光伏板的最优朝向，使得光伏板受到的太阳直射辐射上午大于 $150\text{W}/\text{m}^2$ 、下午大于 $100\text{W}/\text{m}^2$ 的时间尽可能长以及太阳直射辐射总能量尽可能大，以提高路灯蓄电池的储电效率和储电量。并且计算在晴天条件下，光伏板受到的太阳直射辐射日均总能量和太阳直射辐射（上午大于 $150\text{W}/\text{m}^2$ 、下午大于 $100\text{W}/\text{m}^2$ ）时长。

二、模型假设

为简化问题，本文做出以下假设：

- 地球公转周期为 365 天，地球自转周期为 24h，忽略平年闰年差异，忽略一天之内地球公转导致的赤纬角变化。
- 大气层视为包裹地球的厚度不发生变化的球壳，厚度是 1000km。
- 忽略大气导致的太阳光光线偏折。
- 目标地区天气始终处于天晴状况。忽略天空散射辐射和地面散射辐射，只考虑直射辐射。
- 大气层的衰减系数不受气大温度、湿度、气压、气流、大气颗粒等因素的影响，在一年内维持恒定。

三、符号说明

符号	说明	单位
d	大气厚度	km
n	日期序号	/
T_s	地区标准时	h
L	目标地区经度	度
L_s	标准时间位置经度	度
ϕ	目标地区纬度	度
δ	赤纬角	度
h	太阳高度角	度
ω	时角	度
α_s	太阳方向角	度
α_p	光伏板方向角	度
θ_s	太阳光与光伏板法线夹角	度
θ_p	光伏板倾角	度
I_0	太阳系数	W/m^2
I_p	光伏板所受太阳直射强度	W/m^2
$\beta \cdot I$	衰减系数	$W/(m^2 \cdot km)$
β	衰减系数参数	$(km)^{-1}$
S	光伏板面积	$1/m^2$
$E_{...}$	太阳直射辐射能量	$(W \cdot h) \text{ or } (KW \cdot h)$

四、问题一的模型的建立和求解

4.1 问题分析

根据地球表面太阳直射辐射强度与时间、太阳入射角度的变化规律建立数学模型。计算特定方向角和水平仰角的光伏板所受的最大太阳直射强度和每日辐射总能量，不仅需要知道太阳光线与光伏板的角度关系，还需要知道地表与大气层外太阳直射辐射强度的关系。本题难点在于未知衰减系数的求解，模型不仅要能够反映太阳光线的物理过程，还要能够排除不合理数据并求解出未知的衰减系数，通过衰减系数求解所需值。针对此问题，拟在大气衰减方程以及特定角度关系的基础上建立数学模型。

4.2 模型建立

4.2.1 赤道坐标系

首先我们需要确定太阳光线的入射方向。赤道坐标系是以地球中心为坐标原点、地轴为坐标竖轴建立的坐标系，其横纵轴位于赤道平面。太阳光线入射方向可以用赤纬角 δ 和时角 ω 两个坐标表示。

赤纬角是太阳线在地球表面直射点和地球中心联系与赤道平面的夹角，其随时间得的变化可由 Cooper 方程近似计算为：

$$\delta = 23.45 \sin(360 \times \frac{284 + n}{365}) \quad (1)$$

其中 n 为日期序号。

题目规定光伏板朝向正南方向时，方位角为零；朝向正东方向时，方位角为 $+90^\circ$ 。按照题目规定选取时角的正负方向，得到式子：

$$\omega = 180 - T_s \times 15 \pm (L_s - L) \quad (2)$$

其中 T_s 为目标地区的标准时间， L 、 L_s 分别为目标地区经度和其采用的标准时区经度。当标准时间位置经度与目标地区经度均为东经度时，式中“ \pm ”取正号；若均为西经度时，式中“ \pm ”取负号。查询目标地区经纬度，确定目标地区为武汉市，采用北京时间，因此有北京时区经度：

$$L_s = 120^\circ$$

4.2.2 地平坐标系

地平坐标系是以目标地点为原点，水平面法线为坐标竖轴建立的坐标系，其横轴沿经线指向南极。相对目标地点，太阳光线入射方向可以用高度角 h 和方位角 α_s 两个坐标表示。

太阳高度角是目标地点和太阳连与地平面之间的夹角，计算如下：

$$h = \sin \phi \sin \delta + \cos \phi \cos \delta \cos \omega \quad (3)$$

太阳方向角是目标地点和太阳连线在地平面的投影与正南方向的夹角。将太阳光线方向表示从赤道坐标系变换到地平坐标系有：

$$\begin{bmatrix} X_h \\ Y_h \\ Z_h \end{bmatrix} = \begin{bmatrix} \cos h \cdot \cos \alpha_s \\ \cos h \cdot \sin \alpha_s \\ \sin h \end{bmatrix} = \begin{bmatrix} \sin \phi & 0 & -\cos \phi \\ 0 & 1 & 0 \\ \cos \phi & 0 & \sin \phi \end{bmatrix} \times \begin{bmatrix} \cos \delta \cdot \cos \omega \\ \cos \delta \cdot \sin \omega \\ \sin \delta \end{bmatrix}$$

即:

$$\begin{aligned} X_h &= \cos h \cdot \cos \alpha_s = \sin \phi \cdot \cos \delta \cdot \cos \omega - \cos \phi \cdot \sin \delta \\ Y_h &= \cos h \cdot \sin \alpha_s = \cos \delta \cdot \sin \omega \\ Z_h &= \sin a = \cos \phi \cdot \cos \delta \cdot \cos \omega + \sin \phi \cdot \sin \delta \end{aligned}$$

其中 ϕ 为目标地区的纬度, α_s 为太阳方位角。由题目规定, 太阳方向角上午为正值, 下午为负值, 其计算公式为:

$$\sin \alpha_s = \frac{\cos \delta \sin \omega}{\cos h} \quad (4)$$

4.2.3 太阳直射强度与辐射能量

题目表明大气层对太阳能直射辐射的衰减变化量与其辐射强度、所穿过的大气层厚度成正比, 因此有 $\frac{dI}{dx} = \beta I$, 积分得:

$$I = I_0 e^{\frac{\beta d}{\sin h}} \quad (5)$$

其中 β 为衰减系数参数, $\frac{1}{\sin h}$ 是利用太阳高度角对太阳光线穿过大气厚度进行的近似修正。结合 β 为常量的假设与附件中所给的目标地区于 2023 年 5 月 23 日晴天状况下测得地表水平面受到的太阳直射强度值, 进而得到衰减系数 β 的近似值。

当光伏板以方位角 α_p 和水平仰角 θ_p 放置时, 太阳光线与光伏板法线的夹角满足:

$$\cos \theta_s = \cos h \sin \theta_p (\sin \alpha_s \sin \alpha_p + \cos \alpha_s \cos \alpha_p) + \sinh \cos \theta_p \quad (6)$$

此时光伏板表面的太阳直射强度 I_p 为:

$$I_p = I \cdot \cos \theta_s \cdot S \quad (7)$$

在有太阳光照时, 计算光伏板上太阳辐射总能量为:

$$\begin{aligned} E &= \int I_p dt = \int I \cdot \cos \theta_s \cdot S dt \\ s.t. \quad &\begin{cases} \sinh \geq 0 \\ \cos \theta_s \geq 0 \end{cases} \end{aligned} \quad (8)$$

4.3 模型求解

Step1: 由式 (1)(2)(3)(4) 以及题目中数据计算赤纬角、时角、太阳高度角以及太阳方位角

Step2: 由附件中地表水平面收到的太阳直射强度数据以及式 (5) 求出解衰减系数

Step3: 由式 (6)(5)(7)(8) 计算每月 15 日最大太阳直射强度和太阳直射辐射总能量

4.4 求解结果

4.4.1 目标区域数据

以 2023 年 5 月 23 日为例，目标区域的各项数据计算如下：

表 1 该城区在当天 12 点的各项数据

L	ϕ	L_s	δ	ω	h	α_s
114.3167	30.5833	120	20.53	5.68	78.72	28.32

4.4.2 衰减系数

结合附件中地表水平面收到的太阳直射强度数据，求出各时刻 β 值并画出 β 关于时间的曲线如下：

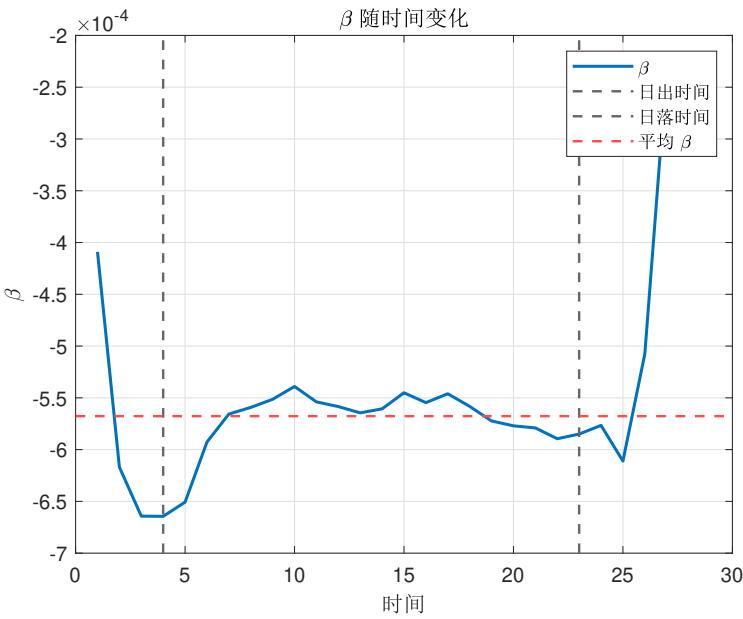


图 1 β 随时间的变化

考虑到日出后与日落后的一段时间内，太阳高度角 h 较小，无法忽略大气折射与地面曲率对太阳光线穿过的大气厚度的影响，厚度修正系数不能简单地用 $1/\sin h$ 表示。故我们在求解 β 过程中，将舍去日出后和日落后一段时间内不合理的数据，即满足 $h < 30^\circ$ 的 β 数据值 [1]，并求取合理范围内 β 数据值的平均值作为衰减系数的求解值。求解得到 $\beta = -5.6769 \times 10^{-4}$ 。

4.4.3 最大直射强度和辐射总能量

假设一个月内每天的大气层外层太阳能辐射强度相同，每个月的辐射强度值由附件得到。将求解出的衰减系数代入公式，求得每月 15 日太阳直射强度和太阳直射辐射总能量，结果如下：

表 2 问题一求解结果

月份	$I_{pmax}(\theta_p = 20)$	$I_{pmax}(\theta_p = 40)$	$I_{pmax}(\theta_p = 60)$	$E_p(\theta_p = 20)$	$E_p(\theta_p = 40)$	$E_p(\theta_p = 60)$
1	476.05	548.49	554.79	2474.60	2911.63	2997.48
2	579.99	632.80	609.29	3249.14	3600.43	3517.45
3	679.10	693.49	624.23	4091.11	4194.52	3792.02
4	735.91	696.66	573.39	4729.15	4408.46	3556.03
5	739.31	658.24	497.79	5036.97	4413.36	3257.44
6	724.27	624.63	449.64	5139.65	4432.48	3190.69
7	722.75	631.41	463.91	5039.74	4367.00	3167.53
8	725.62	667.99	529.78	4776.77	4311.71	3326.58
9	690.27	683.36	594.03	4277.43	4220.10	3653.75
10	602.71	642.13	604.11	3473.09	3745.06	3565.32
11	495.03	561.87	560.94	2630.76	3045.69	3093.26
12	440.56	515.31	527.91	2237.15	2674.71	2789.67

¹ $I_{p\max}$ 为当天光伏板所受最大太阳直射强度, E_p 为当天光伏板受到太阳直射辐射总能量

图 2 E_p 和 $I_{p\max}$ 随月份的变化

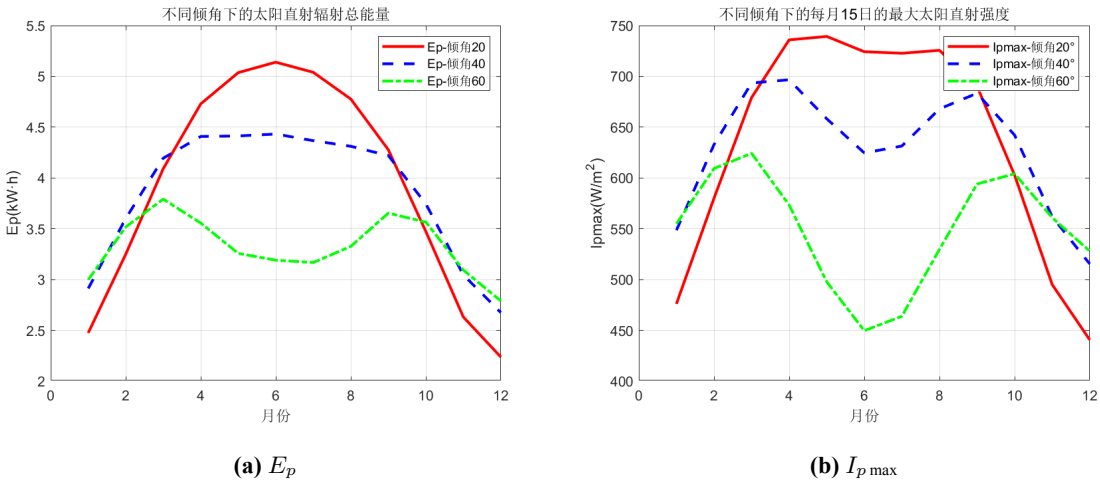
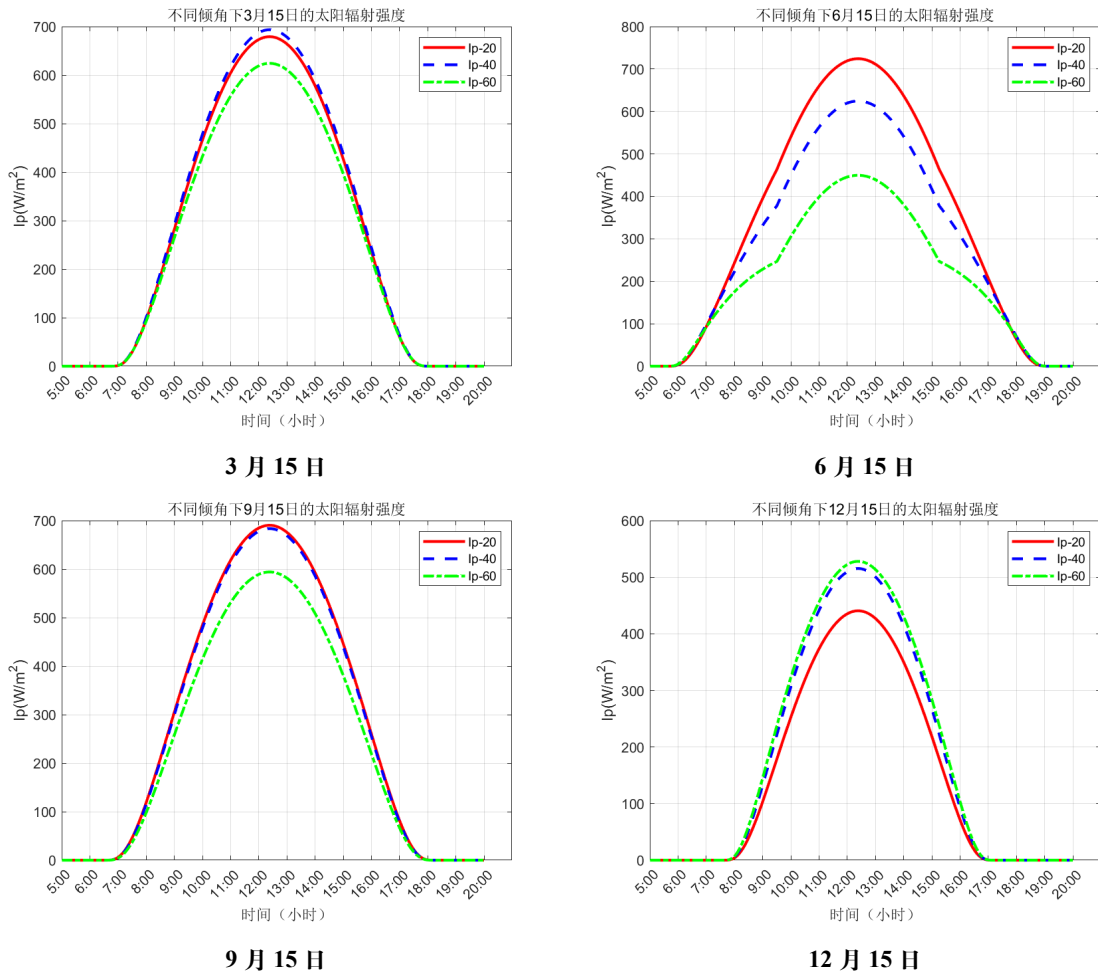


图3 春夏秋冬四季太阳辐射强度



4.4.4 结果分析

从图 (2a) 中可以发现, 倾角 20 度时曲线的峰值约为 5 千瓦时 (5 度电), 对于一个 $1m^2$ 的光伏板而言, 这是符合实际情况的。此外, 观察不同水平倾角的三条曲线随月份的变化, 可以发现在 6 月左右时, 光伏板倾角越大, 太阳能直射辐射总能量越小, 倾角 60 度时还出现了曲线下凹的情况。这或许是由于在夏季时, 太阳直射北回归线, 而该地区的纬度略大于北回归线, 在夏季时太阳高度角接近 90 度。光伏板倾角越大, 其法线与太阳光线的夹角就越大, 光伏板所受到的太阳直射辐射就越少, 从而形成曲线下凹的情况。对于倾角 20 度的光伏板, 由于 20 度较为接近北回归线的纬度, 因此光伏板所受到的直射辐射没有被削减过多, 从而曲线成上凸形。倾角 40 度则是中间的过渡状态。图2b也可以反映出这一点。

五、问题二的模型的建立和求解

5.1 问题分析

问题二是一个优化问题，题目要求我们找到光伏板的最优朝向，从而使得使光伏板在晴天条件下受到的太阳直射辐射日均总能量最大。但是求取光伏板太阳辐射总能量不仅与光伏板的朝向有关，还与光伏板的倾角有关，因此我们需要一个多变量的目标优化算法来解决该问题。其中 PSO(粒子群算法) 是一个不错的选择。

5.2 模型建立

全年光伏板上太阳辐射总能量:

$$E_{all} = \sum_{n=1}^{365} \int_{t1}^{t2} I \cdot \cos\theta_s \cdot S dt \quad (9)$$

问题二是一个优化问题模型，要求我们找到使得太阳直射辐射日均总能量最大的最优朝向和最优倾角，即:

$$\begin{aligned} \max E_{all} &= \sum_{n=1}^{365} \int_{t1}^{t2} I \cdot \cos\theta_s \cdot S dt \\ s.t. &\begin{cases} \sin h \geq 0 \\ \cos\theta_s \geq 0 \\ \sin h(t1) = 0 \\ \sin h(t2) = 0 \end{cases} \end{aligned}$$

5.2.1 PSO(粒子群算法)

PSO (粒子群算法) 源于鸟类集体活动的规律性，将群体智能应用于建立简化模型。它模拟了鸟类的觅食行为，将问题的解空间比作鸟类飞行的区域，将每只鸟抽象为一个没有体积和质量的粒子，用来表示问题的可行解。在 PSO 中，鸟群随机搜索食物，它们的目标是找到食物量最多的位置。然而，每只鸟并不知道食物的确切位置，只能感知到食物大致的方向。因此，每只鸟都沿着自己判断的方向进行搜索，并在搜索过程中记录自己曾经发现过食物量最多的位置。同时，所有鸟都共享彼此每一次发现的食物位置和食物量信息。这使得鸟群能够知道当前哪个位置的食物量最多。在搜索过程中，每只鸟都会根据自己记忆中的最优位置和当前鸟群的最优位置来调整下一步的搜索方向。经过一段时间的搜索，鸟群能够找到森林中食物量最多的位置，即全局最优解。

该算法具有收敛速度快、参数少、算法简单易实现的优点。问题二的目标函数式 (9) 是一个积分函数，求解较复杂，但该算法不涉及求导，没有复杂计算，而且全局搜索能力强，能够很好地求解出全局最优解，故适合选择该算法求解此优化问题

PSO(粒子群算法) 算法步骤:

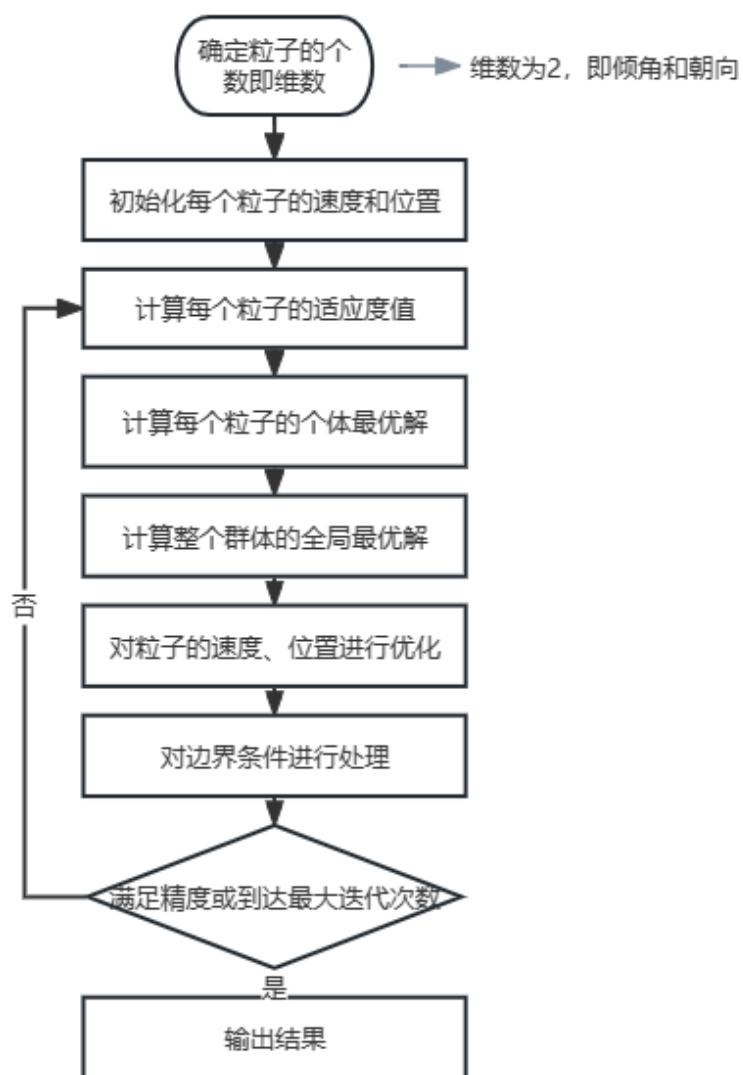


图4 PSO 求解流程

5.3 求解结果

运用 PSO(粒子群算法) 求解结果及其适应性曲线:

图5 粒子群求解结果

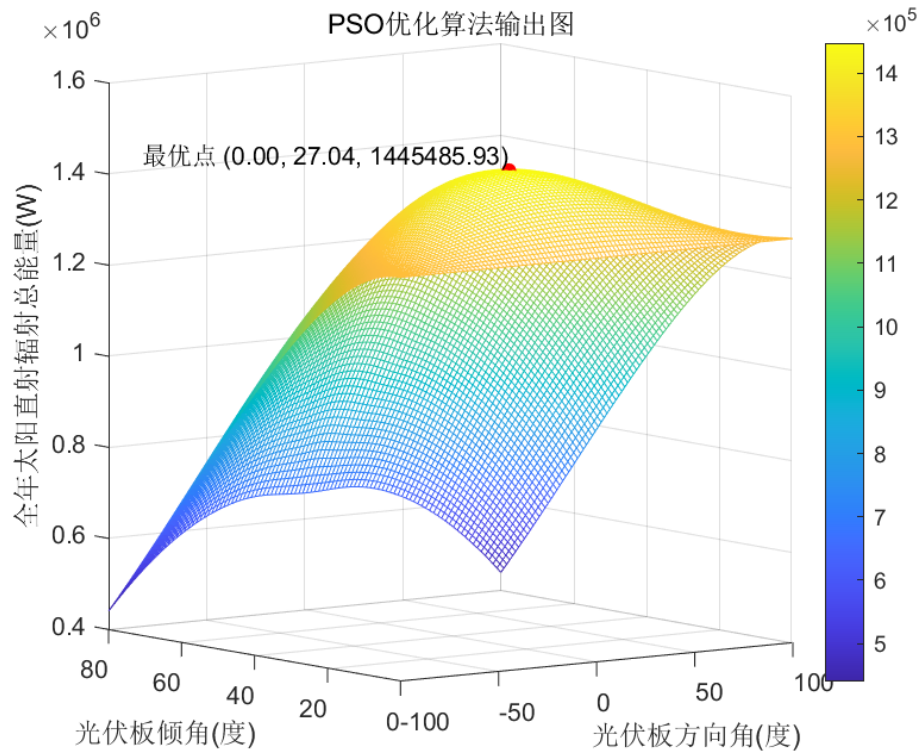
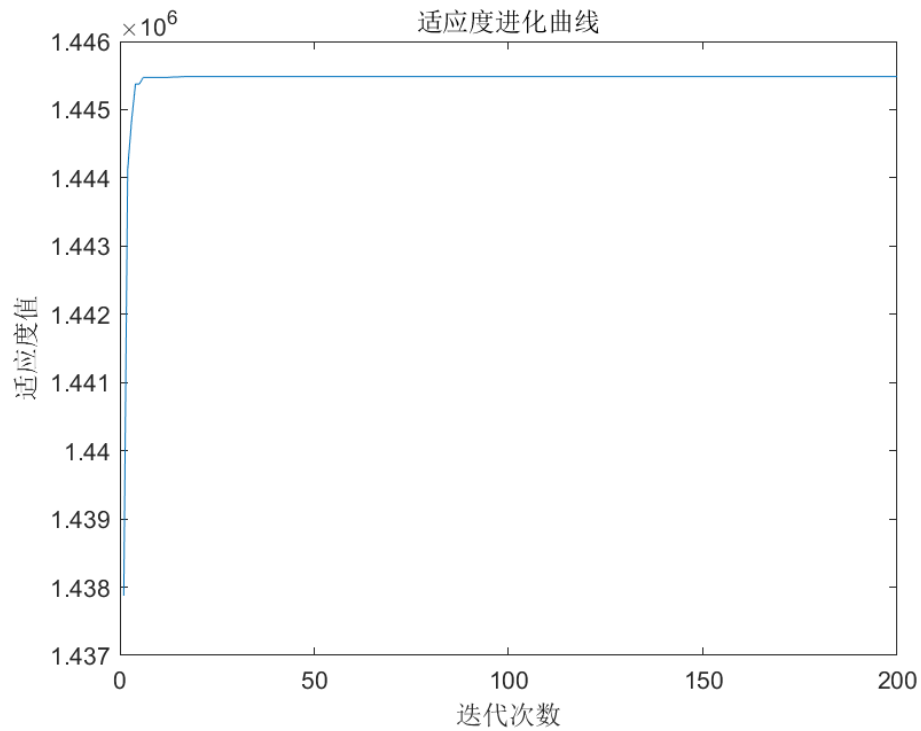


图6 粒子群适应性曲线



利用粒子群算法,我们找到了最大化全年太阳直射辐射总能的最佳朝向和最佳倾角.从图中可知,最优朝向为正南方向,最优倾角为 27.04,此时太阳直射辐射日均总能量最

大, 为 $\frac{1445485.93}{365}(W \cdot h) = 3960.26(W \cdot h) = 3.96026(KW \cdot h)$. 接近于地区的纬度的光伏板倾角往往是最优倾角 [2], 可见该最优倾角是合理有效的. 从图 (6) 中可以发现粒子群算法收敛速度是相对快的, 在第 20 次迭代时已经非常接近最优解了.

六、问题三的模型的建立和求解

6.1 问题分析

问题三是一个多目标优化问题, 有两个目标函数, 一个是日均太阳直射辐射总能量, 另一个是在利于蓄电池储电效率的太阳直射辐射时长. 题目要求综合考虑路灯蓄电池的储电效率高和储电量这两个目标, 可以赋予两个目标函数权重, 将这个多目标优化问题转换为单目标优化问题来进行求解. 但这样决策者影响因素较大, 可以选择自适应调整权重的方法, 例如 MOGA (多目标遗传算法)、MOPSO (多目标粒子群算法). 我们选取适应度缩放来进行自适应权重调整.

6.2 模型建立

$$I_p = I \cdot \cos \theta_s$$

考虑多目标优化问题:

$$\begin{cases} \max E_{all} = \sum_{n=1}^{365} \int_{t1}^{t2} I_p \cdot S dt \\ \max T_{all} = \int_{T1} dt + \int_{T2} dt \end{cases}$$

$$s.t. \begin{cases} I_p(T1) - 150 \geq 0 \\ I_p(T2) - 100 \geq 0 \\ \sin h \geq 0 \\ \cos \theta_s \geq 0 \\ 0 \leq T1 < 12 \\ 12 \leq T2 < 24 \end{cases}$$

6.2.1 多目标赋权的 PSO 优化算法

光伏板表面太阳直射辐射强度: 将两个目标函数赋予权重得到

$$\max(\lambda_1 \cdot E_{all} + \lambda_2 \cdot T_{all}) = \sum_{n=1}^{365} \left(\int_{t1}^{t2} I_p \cdot S dt + \int_{T1} dt + \int_{T2} dt \right) \quad (10)$$

其中 $\lambda_{1,2}$ 为权重, 满足 $\lambda_1 + \lambda_2 = 1$.

6.2.2 多目标粒子群算法 MOPSO

多目标粒子群算法 (Multi-Objective Particle Swarm Optimization, MOPSO) 是一种基于群体智能的优化方法, 通过调整参数 (如惯性权重、加速度因子等) 来实现自适应权重调整, 用于寻找 Pareto 最优解集。此外, 也可以采用基于非支配排序的方法来引导搜索过程, 以获得更好的 Pareto 最优解集。具体方法是利用外部种群存档 (Archive) 存储当前所有的非支配解, 并将外部存档中个体看作精英个体, 通过精英个体控制种群进化方向, 引导种群逼近真实 Pareto 前沿, 算法运行结束后将外部存档中粒子作为获得的 Pareto 最优解近似。

6.3 模型求解

6.3.1 多目标赋权的 PSO 优化算法

Step1: 赋予 λ_1, λ_2 初始值

Step2: 在第二问的基础上运用 PSO 求解式 (10)

6.3.2 多目标粒子群算法 MOPSO

Step1: 初始化粒子群和相关参数, 计算目标函数值, 设置为第一代粒子群

Step2: 初始化粒子群的外部存档, 选择外部存档中拥挤密度大的粒子作为全局最优解

Step3: 对粒子群执行 PSO, 获得下一代粒子群

Step4: 合并历史存档获得新的历史存档, 采用设定函数调整适应度函数

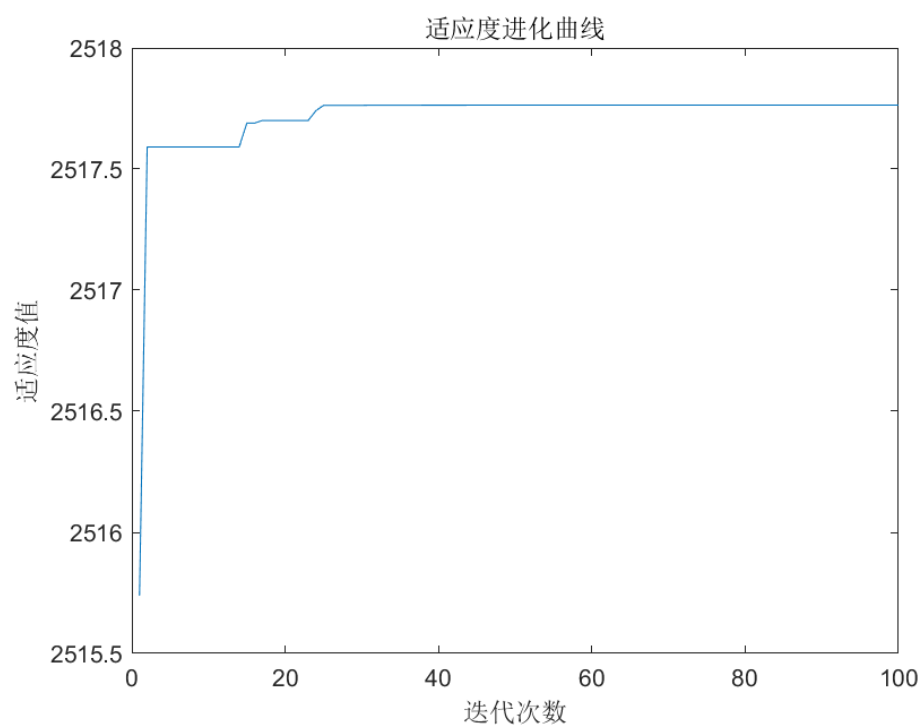
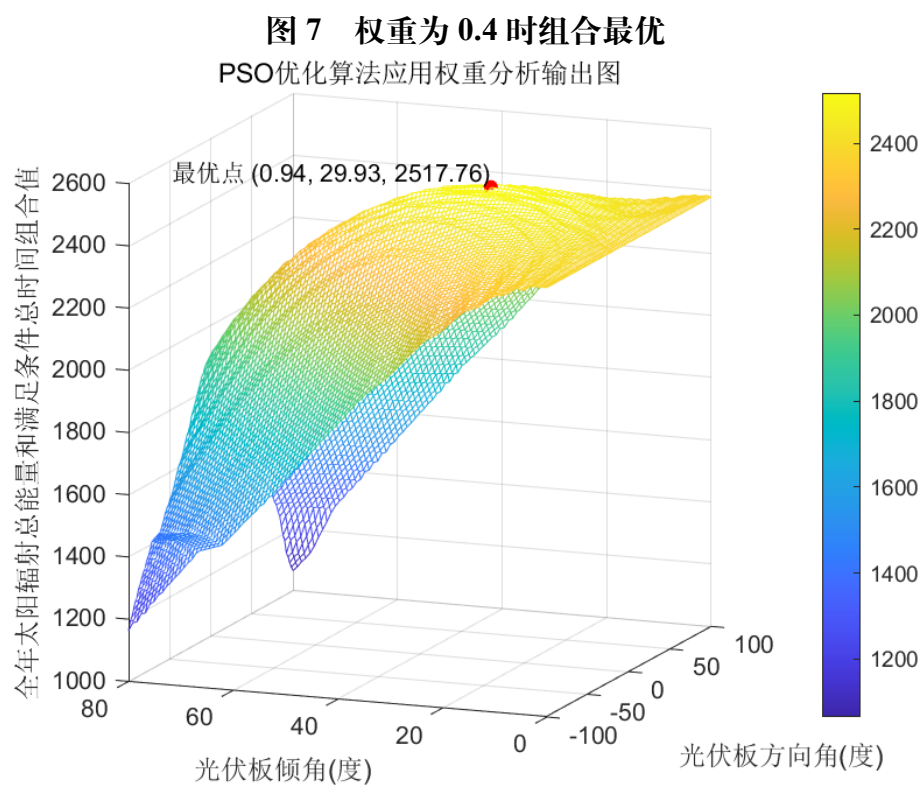
Step5: 执行支配比较更新当代粒子群的个体最优值

Step6: 选择外部存档中拥挤密度大的粒子作为全局最优解

Step7: 重复 Step3 到 Step6, 直到满足精度或到达最大迭代次数

6.4 求解结果

6.4.1 多目标赋权的 PSO 求解结果



可以看到，此多目标优化问题转换到单目标优化问题后，求出的最优方向角为 0.94 度，倾角为 29.93 度。计算得出光伏板受到的太阳直射辐射日均总能量约为 $3.87(KW \cdot h)$ ，全年太阳直射辐射满足条件的时长为 3030.5 小时。

6.4.2 多目标粒子群算法 MOPSO 求解结果

图 8 MOPSO 求解得到的最优解集

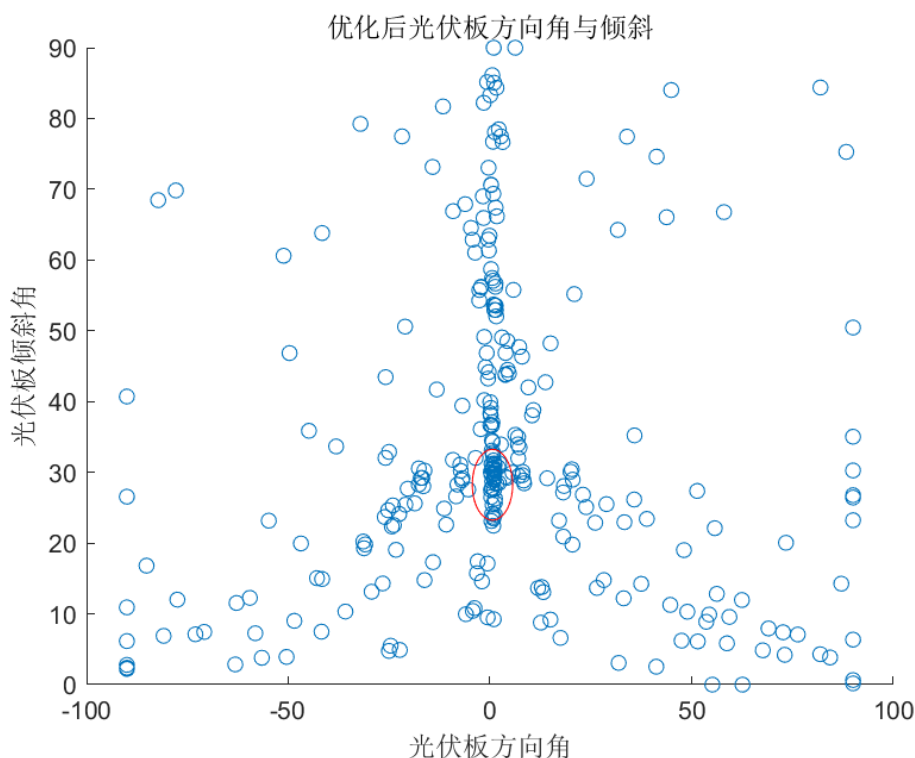


图 9 红色曲线: $(x - 3.41)^2 + (y - 28.24)^2 = 5^2$

图中可以发现，Pareto 解集中密度最大的区域是红色曲线所包围的区域，此外，将多目标赋权的 PSO 算法求得的最优解也在该红色曲线区域内。

取红色区域的圆心值作为最优值，即最优方向角为 3.41 度，最佳倾角为 29.93 度，代入计算可得光伏板受到的太阳直射辐射日均总能量约为 $3.96(KW \cdot h)$ ，全年太阳直射辐射满足条件的时长为 3117.5 小时。总的来说，光伏板朝向正南左右，倾角在 $27 \sim 30$ 度时是综合性能最好的。

七、模型的评价

7.1 模型的优点与创新

- 问题一构建的数学模型充分考虑到了大气折射与地面曲率对太阳光线的影响,清洗不合理数据之后解出的衰减系数符合实际,较好地拟合了光伏板太阳直射辐射强度与辐射总能量随时间的变化关系,与实际测量数据吻合。

- 问题二运用的 PSO(粒子群算法),其最优值搜索过程是从问题解的一个集合开始的,而不是从单个个体开始,具有隐含并行搜索特性,从而减小了陷入局部极小的可能性。此外,相较于 GA(遗传算法)更擅长处理连续变量问题,能够很好的求出全局最优解。

- 问题三运用了两种不同的算法处理了多目标优化问题。第一种利用权重将多目标优化问题转化为单目标优化问题,可以利用 PSO 直解搜索最优值,简化了搜索过程降低了搜索时间,还能按照决策的倾向自行设定权重得到适宜的最优解。第二种则利用 MOPSO 得到了两个目标函数的最优解集,最优解集合中的每个解都是帕累托最优解,这反映了多目标优化问题的复杂性和多样性。在实际应用中,研究人员或决策者可以根据特定的需求和偏好从这些最优解中选择最合适的解决方案,较为方便。

7.2 模型的缺点与改进

- 问题一中所取的 I_0 是将附件中的每月的数据近似作为该月份每天的 I_0 ,此时计算日均太阳直射总能量存在偏差。可以根据太阳光强与测量点到太阳距离的二次方成反比的关系,设计算法计算出每天的大气层外光强,使得我们的计算更加精准。

- 问题二中的 PSO 算法由于初始时随机性较大,若目标函数不够平滑,多次运行可能有不同结果。可以尝试修改惯性因子大小和迭代方式,使得 PSO 算法能较好符合我们的最优值求取要求。还可以尝试多种优化算法,比较不同算法的优劣,一边更好地求解光伏板取向的优化问题。

- 当积分求解式 (9) 求解时设定的步长较小时,会导致 PSO 算法迭代次数过多,计算较慢。选取步长大小时,应该衡量精确度和运行时间。

- 多目标赋权的 PSO 算法,权重设置较为粗糙,没有综合考虑储电效率和储电量的取舍关系。可以取不同的权重,得到具有适合不同要求的最优解。

参考文献

- [1] 祖文超,左廷荣,戎卫国. 太阳辐射分析与集热器倾角优化求解. 中国建筑学会建筑热能动力分会第十七届学术交流大会,2011

[2] 朱丹丹, 燕达. 太阳能板放置最佳倾角研究 [J]. 建筑科学, 2012, 28(S2): 277-281.

附录 A 文件列表

文件名	功能描述
cal_beta.m	问题一估计 β 的值
Ep_int.m	计算每天的光照强度
t1_data.m	计算每月 15 日的数据
t2_PSO.m	PSO 求解最优朝向
Ep_int_and_totaltime.m	计算每天的辐射强度和高效率时长
All_year_and_time.m	给定朝向和倾角计算全年辐射强度和时长
t3PSO_k.m	赋权后 PSO 算法
t3_MOPSO_k.m	多目标 PSO 算法

附录 B 代码

cal_beta.m

```
1 clear
2 close all
3 clc
4
5 n=31+28+31+30+23; %日期序号
6 L=114+19/60; %目标经度
7 Ls=120; %时区经度
8 phi=30+35/60; %目标纬度
9 I0=1334; %5月太阳常数
10 d=1000; %大气厚度
11
12 %表格数据
13 Ts=6:0.5:19;
14 I=[21.132 64.906 155.472 265.660 372.830 499.623 587.170
    643.019 ...
    688.302 727.547 736.604 747.170 750.189 754.717 762.264
    744.151 733.585 ...
    698.868 653.585 605.283 546.415 464.906 377.358 271.698
    125.283 48.302 10.566];
```

```

17
18 delta=23.45*sind(360*(284+n)/365); %赤纬角
19 omega=180-Ts*15+(Ls-L); %时角
20 sinh=sind(phi)*sind(delta)+cosd(phi)*cosd(delta)*cosd(omega);
    %太阳高度角正弦值
21 beta=sinh/d.*(log(I)-log(I0)); %衰减系数的取值
22 beta1=beta;
23 sinAs=cosd(delta)*sind(omega)./sqrt(1-sinh.^2); %太阳方向角正
    弦值
24 %找到sinh中小于0.5项
25 indices = sinh < 0.5;
26 %把对应的项消除
27 beta(indices) = [];
28 %求beta的平均值
29 beta_ave=sum(beta)/length(beta)
30
31
32 figure
33 plot(beta1, 'LineWidth', 1.5) % 增加线宽
34 hold on
35 xline(4, '--', 'LineWidth', 1.2) % 添加虚线，调整线宽
36 xline(23, '--', 'LineWidth', 1.2) % 添加虚线，调整线宽
37 yline(beta_ave, '--r', 'LineWidth', 1.2) % 添加虚线，调整线宽
    和颜色
38 xlabel('时间') % 添加 x 轴标签
39 ylabel('\beta') % 添加 y 轴标签
40 title('\beta 随时间变化') % 添加图标题
41 grid on % 添加网格线
42 legend('\beta', '日出时间', '日落时间', '平均 \beta') % 添加图
    例
43
44 % %把对应的项消除
45 % sinh(indices) = [];
46 % I(indices) = [];
47 % %线性拟合求衰减系数

```

```

48 % x=d./sinh;
49 % y=log(I)-log(I0);
50 % p=polyfit(x,y,1);
51 % 求beta的线性拟合值
52 % beta_sti=p(1)

```

Ep_int.m

```

1 function [Ep_int,Ipmax,Ip]=Ep_int(n,I0,Ap,Thetap)
2     L=114+19/60; %目标经度
3     Ls=120; %时区经度
4     phi=30+35/60; %目标纬度
5     delta=23.45*sind(360*(284+n)/365); %赤纬角
6     Ts=[5:0.01:20]; %步距0.01
7     ns=length(Ts); %步数
8     beta=-5.6769e-4; %衰减系数
9     d=1000;
10    omega=180-Ts*15+(Ls-L); %时角
11
12    sinh=sind(phi)*sind(delta)+cosd(phi)*cosd(delta)*cosd(
13    omega); %太阳高度角正弦值
14    sinAs=cosd(delta)*sind(omega)./sqrt(1-sinh.^2); %太阳方向
    角正弦值
15    costhetas=sqrt(1-sinh.^2)*sin(Thetap).*(sinAs*sin(Ap)+sqrt
16    (1-sinAs.^2)*cos(Ap))+sinh*cos(Thetap); %法线夹角余弦值
17    Ip=zeros(1,ns); %直射强度初始化
18    %太阳板表面光强
19    for i=1:ns
20        if costhetas(i)<=0 || sinh(i)<=0
21            Ip(i)=0;
22        else
23            Ip(i)=I0*exp(d*beta/sinh(i)).*costhetas(i);
24        end
25    end
26    Ep_int=trapz(Ts,Ip); %对光强进行梯形法积分
27    Ipmax=max(Ip);

```

26 end

tl_data.m

```
1 days_in_month = [31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30,
    31]; % 一年中每个月的天数
2 n = [15,zeros(1,length(days_in_month)-1)];
3 for i=2:length(days_in_month)
4     n(i)=cumsum(days_in_month(i-1))+n(i-1);
5 end
6 I0
    =[1405,1394,1378,1353,1334,1316,1308,1315,1330,1350,1372,1392];

7 %初始化
8 Ep_int_20list=zeros(1,12);
9 Ep_int_40list=zeros(1,12);
10 Ep_int_60list=zeros(1,12);
11 Ipmax_20=zeros(1,12);
12 Ipmax_40=zeros(1,12);
13 Ipmax_60=zeros(1,12);
14 Ip_20= zeros(1501,12);
15 Ip_40=zeros(1501,12);
16 Ip_60=zeros(1501,12);
17 Thetap=[pi/9,2*pi/9,pi/3];
18
19 for month = 1:12
20     [Ep_int_20list(month),Ipmax_20(month),Ip_20(:,month)] =
        Ep_int(n(month),I0(month),0,Thetap(1));
21     [Ep_int_40list(month),Ipmax_40(month),Ip_40(:,month)] =
        Ep_int(n(month),I0(month),0,Thetap(2));
22     [Ep_int_60list(month),Ipmax_60(month),Ip_60(:,month)] =
        Ep_int(n(month),I0(month),0,Thetap(3));
23 end
24
25 figure;
26 plot(Ipmax_20,'r-','LineWidth',2)
```

```

27 hold on
28 plot(Ipmax_40,'b--','LineWidth',2)
29 plot(Ipmax_60,'g-.','LineWidth',2)
30 legend('Ipmax-倾角20°','Ipmax-倾角40°','Ipmax-倾角60°')
31 xlabel('月份')
32 ylabel('Ipmax(W/m^2)')
33 title('不同倾角下的每月15日的最大太阳直射强度');
34 grid on;
35 hold off
36 filename = sprintf('Ipmax_.png');
37 saveas(gcf, filename);
38 close(gcf);
39
40 figure;
41 plot(Ep_int_20list/1000,'r-','LineWidth',2);
42 hold on;
43 plot(Ep_int_40list/1000,'b--','LineWidth',2);
44 plot(Ep_int_60list/1000,'g-.','LineWidth',2);
45 legend('Ep-倾角20°','Ep-倾角40°','Ep-倾角60°');
46 xlabel('月份');
47 ylabel('Ep(kW·h)');
48 title('不同倾角下的太阳直射辐射总能量');
49 grid on;
50 hold off;
51 filename = sprintf('Ep_.png');
52 saveas(gcf, filename);
53 close(gcf);
54
55 for month = 1:12
56     figure;
57     plot(Ip_20(:, month), 'r-', 'LineWidth', 2);
58     hold on;
59     plot(Ip_40(:, month), 'b--', 'LineWidth', 2);
60     plot(Ip_60(:, month), 'g-.', 'LineWidth', 2);
61     xticks([0:100.066:1501]);

```



```

62     xticklabels({'5:00', '6:00', '7:00', '8:00', '9:00', '
10:00', '11:00', '12:00', '13:00', '14:00', '15:00', '16:00'
, '17:00', '18:00', '19:00', '20:00'}));
63     xtickangle(45); % 将 x 轴标签旋转 45 度
64     legend('Ip-20', 'Ip-40', 'Ip-60');
65     xlabel('时间 (小时) ');
66     ylabel('Ip(W/m^2)');
67     title(['不同倾角下', num2str(month), '月15日的太阳辐射强度
']);
68     grid on;
69     hold off;
70     filename = sprintf('Ip_%d.png', month);
71     saveas(gcf, filename);
72     close(gcf);
73 end

```

t2_PSO.m

```

1  clear
2  close all
3  clc
4
5  L=114+19/60; %目标经度
6  Ls=120; %时区经度
7  Ld=Ls-L; %经度差值
8  phi=30+35/60; %目标纬度
9  d=1000; %大气厚度
10 beta=-5.6769e-4; %衰减系数
11
12
13 %全年太阳常数数据
14 I_monthly = [1405, 1394, 1378, 1353, 1334, 1316, 1308, 1315,
1330, 1350, 1372, 1392]; %每月的太阳常数
15 days_in_month = [31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30,
31]; % 一年中每个月的天数
16 I_daily = [];

```

```

17 for month = 1:12
18     I_daily = [I_daily, repmat(I_monthly(month), 1,
    days_in_month(month))]; % 将每个月的辐射强度重复相应的天数
19 end
20
21 N=100; %群体粒子个数
22 D=2; %粒子维数
23 T=200; %最大迭代次数
24 c1=1.5; %学习因子1
25 c2=1.5; %学习因子2
26 w=0.8; %惯性权重
27 Xmax=20; %位置最大值
28 Xmin=-20; %位置最小值
29 Vmax=10; %速度最大值
30 Vmin=-10; %速度最小值
31 Wmax=0.8; %惯性权重最大值
32 Wmin=0.4; %惯性权重最小值
33 %初始化个体
34 x=rand(N,D)*(Xmax-Xmin)+Xmin;
35 v=rand(N,D)*(Vmax-Vmin)+Vmin;
36
37 %初始化个体最优位置和最优值
38 p=x;
39 pbest=ones(N,1);
40 for i=1:N
41     pbest(i)=func1(x(i,:), phi, d, beta, I_daily, Ld);
42 end
43
44 %初始化全局最优位置和最优值
45 g=ones(1,D);
46 gbest=0;
47 for i=1:N
48     if (pbest(i)>gbest)
49         g=p(i,:);
50         gbest=pbest(i);

```

```

51     end
52 end
53 gb=ones(1,T);
54
55 %按照公式依次迭代直到满足精度或者迭代次数
56 for i=1:T
57     for j=1:N
58         new_fitness = func1(x(j,:), phi, d, beta, I_daily, Ld)
59         ;
60         if new_fitness > pbest(j)
61             p(j,:)=x(j,:);
62             pbest(j)=new_fitness;
63         end
64         if (pbest(j)>gbest)
65             g=p(j,:);
66             gbest=pbest(j);
67         end
68 %         %计算动态惯性权重值
69 w=Wmax-(Wmax-Wmin)*i/T;
70 v(j,:)=w*v(j,:)+c1*rand*(p(j,:)-x(j,:))+c2*rand*(g-x(j
71 ,:));
72 x(j,:)=x(j,:)+v(j,:);
73 %边界条件处理
74 for ii=1:D
75     if (v(j,ii)<Vmin)|| (v(j,ii)>Vmax)
76         v(j,ii)=rand*(Vmax-Vmin)+Vmin;
77     end
78     if (x(j,ii)<Xmin)|| (x(j,ii)>Xmax)
79         x(j,ii)=rand*(Xmax-Xmin)+Xmin;
80     end
81 end
82 %记录全局最优值
83 gb(i)=gbest;
84 end

```

```

84 g;%最优个体
85 gb(end);%最优值
86 figure
87 plot(gb)
88 xlabel('迭代次数')
89 ylabel('适应度值')
90 title('适应度进化曲线')
91 %最优值处理:
92 %适应度函数
93
94 % 绘制三维表面图
95 [Ap_grid, Thetap_grid] = meshgrid(linspace(-20, 20, 100),
    linspace(-20, 20, 100));
96 E_values = arrayfun(@(ap, tp) func1([ap, tp], phi, d, beta,
    I_daily, Ld), Ap_grid, Thetap_grid);
97
98
99 figure;
100 mesh(Ap_grid, Thetap_grid, E_values);
101 xlabel('光伏板方向角(度)');
102 ylabel('光伏板倾角(度)');
103 zlabel('全年太阳直射辐射总能量(W)');
104 title('PSO优化算法输出图');
105 colorbar;
106 hold on;
107 % 标记最优值, 使用实心点标记
108 optimal_Ap = g(1); % 最优Ap角度
109 optimal_Thetap = g(2); % 最优Thetap角度
110 plot3(optimal_Ap, optimal_Thetap, gbest, 'r.', 'MarkerSize',
    20, 'LineWidth', 2); % 使用点(.)代替圆圈(o)
111 text(optimal_Ap, optimal_Thetap, gbest, sprintf('最优值 (%.2f,
    %.2f, %.2f)', optimal_Ap/40*180, optimal_Thetap/80*180+45,
    gbest), 'VerticalAlignment', 'bottom', 'HorizontalAlignment'
    , 'right');
112 hold off;

```

```

113
114 view(-60, 30);
115 % 自定义x轴刻度标签
116 xticklabels({'-100', '-50', '0', '50', '100'});
117 % 自定义y轴刻度标签
118 yticklabels({'0', '20', '40', '60', '80', '100'});
119
120 % 标记最优点，使用实心点标记
121 optimal_Ap = g(1); % 最优Ap角度
122 optimal_Thetap = g(2); % 最优Thetap角度
123 plot3(optimal_Ap, optimal_Thetap, gbest, 'r.', 'MarkerSize',
        20, 'LineWidth', 2); % 使用点(.)代替圆圈(o)
124 text(optimal_Ap, optimal_Thetap, gbest, sprintf('最优点 (%.2f,
        %.2f, %.2f)', optimal_Ap/40*180, optimal_Thetap/80*180+45,
        gbest), 'VerticalAlignment', 'bottom', 'HorizontalAlignment'
        , 'right');
125 hold off;
126
127 view(-60, 30);
128
129 function value=func1(position, phi, d, beta, I_daily, Ld)
130     Ap=position(1)/40*pi; %光伏板方向角赋值
131     Thetap=position(2)/80*pi+pi/4; %光伏板倾斜角赋值
132     E=0; %总能量初始化
133     for n=1:365
134         I0=I_daily(n);
135         E=E+Ip_int(n,I0,Ap,Thetap, Ld, phi, d, beta);
136     end
137     value=E;
138 end
139
140 %积分函数
141 function Ep=Ip_int(n,I0,Ap,Thetap, Ld, phi, d, beta)
142     delta=23.45*sind((284+n)/365*360); %赤纬角
143 %IP是一天某一时刻光强

```

```

144 %梯形法求积分
145 Ts=[5:0.1:20.0]; %步距0.01
146 ns=length(Ts); %步数
147 omega=180-Ts*15+Ld; %时角
148 sinh=sind(phi)*sind(delta)+cosd(phi)*cosd(delta)*cosd(
omega); %太阳高度角正弦值
149 sinAs=cosd(delta)*sind(omega)./sqrt(1-sinh.^2); %太阳方向
角正弦值
150 costhetas=sqrt(1-sinh.^2)*sin(Thetap).*(sinAs*sin(Ap)+sqrt
(1-sinAs.^2)*cos(Ap))+sinh*cos(Thetap); %法线夹角余弦值
151 Ip=zeros(1,ns); %直射强度初始化
152 %太阳板表面光强
153 for k=1:ns
154     if costhetas(k)<=0 || sinh(k)<=0
155         Ip(k)=0;
156     else
157         Ip(k)=I0*exp(d*beta/sinh(k)).*costhetas(k);
158     end
159 end
160 %一天总光强EP
161 Ep=trapz(Ts,Ip); %对光强进行梯形法积分
162 end

```

Ep_int_and_totalltime.m

```

1 function [AllEp_int,totalTime]=Ep_int_and_totalltime(n,I0,Ap,
    Thetap)
2     L=114+19/60; %目标经度
3     Ls=120; %时区经度
4     phi=30+35/60; %目标纬度
5     delta=23.45*sind(360*(284+n)/365); %赤纬角
6
7     beta=-5.6769e-4; %衰减系数
8     d=1000;
9     dt=0.1;
10    Ts=5:dt:20; %步距0.1

```

```

11     ns=length(Ts); %步数
12     omega=180-Ts*15+(Ls-L); %时角
13     sinh=sind(phi)*sind(delta)+cosd(phi)*cosd(delta)*cosd(
14     omega); %太阳高度角正弦值
15     sinAs=cosd(delta)*sind(omega)./sqrt(1-sinh.^2); %太阳方向
    角正弦值
16     costhetas=sqrt(1-sinh.^2)*sin(Thetap).*(sinAs*sin(Ap)+sqrt
17     (1-sinAs.^2)*cos(Ap))+sinh*cos(Thetap); %法线夹角余弦值
18     Ip=zeros(1,ns); %直射强度初始化
19     %太阳板表面光强
20     for i=1:ns
21         if costhetas(i)<=0 || sinh(i)<=0
22             Ip(i)=0;
23         else
24             Ip(i)=I0*exp(d*beta/sinh(i)).*costhetas(i);
25         end
26     end
27     AllEp_int=trapz(Ts,Ip); %对光强进行梯形法积分
28
29     morningThresh = 150; % 上午阈值
30     afternoonThresh = 100; % 下午阈值
31     morningTime = Ts <= 12; % 上午的时间点
32     afternoonTime = Ts > 12; % 下午的时间点
33
34     % 上午和下午分别满足条件的时间点数
35     morningSum = sum(Ip(morningTime) > morningThresh);
36     afternoonSum = sum(Ip(afternoonTime) > afternoonThresh);
37     % 计算总时长
38     totalTime = (morningSum + afternoonSum) * dt; % 总时间，乘
    以时间步长转换为小时
39 end

```

All_year_and_time.m

```

1 function All_year= All_year_and_time(Ap,Thetap)
2 %全年太阳常数数据

```

```

3 I_monthly = [1405, 1394, 1378, 1353, 1334, 1316, 1308, 1315,
    1330, 1350, 1372, 1392]; %每月的太阳常数
4 days_in_month = [31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30,
    31]; % 一年中每个月的天数
5 I_daily = [];
6 for month = 1:12
7     I_daily = [I_daily, repmat(I_monthly(month), 1,
        days_in_month(month))]; % 将每个月的辐射强度重复相应的天数
8 end
9 All_year_E=zeros(1,365);
10 All_year_Time=zeros(1,365);
11 for n = 1:365
12     [AllEp_int,totalTime]=Ep_int_and_totaltime(n,I_daily(n),Ap
        ,Thetap);
13     All_year_E(n)=AllEp_int;
14     All_year_Time(n) = totalTime;
15 end
16 All_year(1)= sum(All_year_E);
17 All_year(2)= sum(All_year_Time);
18 end

```

t3PSO_k.m

```

1 N=100; %群体粒子个数
2 D=2; %粒子维数
3 T=100; %最大迭代次数
4 c1=1.5; %学习因子1
5 c2=1.5; %学习因子2
6 w=0.8; %惯性权重
7 Xmax=20; %位置最大值
8 Xmin=-20; %位置最小值
9 Vmax=10; %速度最大值
10 Vmin=-10; %速度最小值
11 Wmax=0.8; %惯性权重最大值
12 Wmin=0.4; %惯性权重最小值
13 %初始化个体

```



```

14 x=rand(N,D)*(Xmax-Xmin)+Xmin;
15 v=rand(N,D)*(Vmax-Vmin)+Vmin;
16
17 %初始化个体最优位置和最优值
18 p=x;
19 pbest=ones(N,1);
20 for i=1:N
21     pbest(i)=kwAllyear(x(i,:));
22 end
23
24
25 %初始化全局最优位置和最优值
26 g=ones(1,D);
27 gbest=0;
28 for i=1:N
29     if (pbest(i)>gbest)
30         g=p(i,:);
31         gbest=pbest(i);
32     end
33 end
34 gb=ones(1,T);
35
36
37 %按照公式依次迭代直到满足精度或者迭代次数
38 for i=1:T
39     for j=1:N
40         new_fitness = kwAllyear(x(j,:));
41         if new_fitness > pbest(j)
42             p(j,:)=x(j,:);
43             pbest(j)=new_fitness;
44         end
45         if (pbest(j)>gbest)
46             g=p(j,:);
47             gbest=pbest(j);
48         end

```

```

49 %           %计算动态惯性权重值
50     w=Wmax-(Wmax-Wmin)*i/T;
51     v(j,:)=w*v(j,:)+c1*rand*(p(j,:)-x(j,:))+c2*rand*(g-x(j
    ,:));
52     x(j,:)=x(j,:)+v(j,:);
53     %边界条件处理
54     for ii=1:D
55         if (v(j,ii)<Vmin)||(v(j,ii)>Vmax)
56             v(j,ii)=rand*(Vmax-Vmin)+Vmin;
57         end
58         if (x(j,ii)<Xmin)||(x(j,ii)>Xmax)
59             x(j,ii)=rand*(Xmax-Xmin)+Xmin;
60         end
61     end
62 end
63 %记录全局最优值
64 gb(i)=gbest;
65 end
66
67 g;%最优个体
68 gb(end);%最优值
69 figure
70 plot(gb)
71 xlabel('迭代次数')
72 ylabel('适应度值')
73 title('适应度进化曲线(w=0.8)')
74 %最优值处理:
75 %适应度函数
76
77
78
79
80
81 % 绘制三维表面图
82 [Ap_grid, Thetap_grid] = meshgrid(linspace(-20, 20, 100),

```

```

    linspace(-20, 20, 100));
83 values = arrayfun(@(ap, tp) kWAllyear([ap, tp]), Ap_grid,
    Thetap_grid);
84
85 figure;
86 mesh(Ap_grid, Thetap_grid, values);
87 xlabel('光伏板方向角(度)');
88 ylabel('光伏板倾角(度)');
89 zlabel('全年太阳辐射总能量和满足条件总时间组合值');
90 title('PSO优化算法应用权重分析输出图(w=0.8)');
91 colorbar;
92 hold on;
93 % 标记最优点，使用实心点标记
94 optimal_Ap = g(1); % 最优Ap角度
95 optimal_Thetap = g(2); % 最优Thetap角度
96 plot3(optimal_Ap, optimal_Thetap, gbest, 'r.', 'MarkerSize',
    20, 'LineWidth', 2); % 使用点(.)代替圆圈(o)
97 text(optimal_Ap, optimal_Thetap, gbest, sprintf('最优点 (%.2f,
    %.2f, %.2f)', optimal_Ap/40*180, optimal_Thetap/80*180+45,
    gbest), 'VerticalAlignment', 'bottom', 'HorizontalAlignment'
    , 'right');
98 hold off;
99
100
101 % 自定义x轴刻度标签
102 xticklabels({'-100', '-50', '0', '50', '100'});
103 % 自定义y轴刻度标签
104 yticklabels({'0', '20', '40', '60', '80', '100'});
105
106 function kWAllyear = kWAllyear(position)
107 w=0.8;
108 Ap=position(1)/40*pi; %光伏板方向角赋值
109 Thetap=position(2)/80*pi+pi/4; %光伏板倾斜角赋值
110 Allyear=All_year_and_time(Ap,Thetap);
111 kWAllyear=w*Allyear(1)/1000+(1-w)*Allyear(2);%能量转换成千瓦

```

```

112 end
113
114 function All_year= All_year_and_time(Ap,Thetap)
115 %全年太阳常数数据
116 I_monthly = [1405, 1394, 1378, 1353, 1334, 1316, 1308, 1315,
1330, 1350, 1372, 1392]; %每月的太阳常数
117 days_in_month = [31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30,
31]; % 一年中每个月的天数
118 I_daily = [];
119 for month = 1:12
120     I_daily = [I_daily, repmat(I_monthly(month), 1,
days_in_month(month))]; % 将每个月的辐射强度重复相应的天数
121 end
122 All_year_E=zeros(1,365);
123 All_year_Time=zeros(1,365);
124 for n = 1:365
125     [AllEp_int,totalTime]=Ep_int_and_totalltime(n,I_daily(n),Ap
,Thetap);
126     All_year_E(n)=AllEp_int;
127     All_year_Time(n) = totalTime;
128 end
129 All_year(1)= sum(All_year_E);
130 All_year(2)= sum(All_year_Time);
131 end
132
133 function [AllEp_int,totalTime]=Ep_int_and_totalltime(n,I0,Ap,
Thetap)
134     L=114+19/60; %目标经度
135     Ls=120; %时区经度
136     phi=30+35/60; %目标纬度
137     delta=23.45*sind((284+n)/365); %赤纬角
138
139     beta=-5.745e-4; %衰减系数
140     d=1000;
141     dt=0.1;

```

```

142     Ts=[5:dt:20]; %步距0.01
143     ns=length(Ts); %步数
144     omega=180-Ts*15+(Ls-L); %时角
145     sinh=sind(phi)*sind(delta)+cosd(phi)*cosd(delta)*cosd(
omega); %太阳高度角正弦值
146     sinAs=cosd(delta)*sind(omega)./sqrt(1-sinh.^2); %太阳方向
角正弦值
147     costhetas=sqrt(1-sinh.^2)*sin(Thetap).*(sinAs*sin(Ap)+sqrt
(1-sinAs.^2)*cos(Ap))+sinh*cos(Thetap); %法线夹角余弦值
148     Ip=zeros(1,ns); %直射强度初始化
149     %太阳板表面光强
150     for i=1:ns
151         if costhetas(i)<=0 || sinh(i)<=0
152             Ip(i)=0;
153         else
154             Ip(i)=I0*exp(d*beta/sinh(i)).*costhetas(i);
155         end
156     end
157     AllEp_int=trapz(Ts,Ip); %对光强进行梯形法积分
158
159     morningThresh = 150; % 上午阈值
160     afternoonThresh = 100; % 下午阈值
161     morningTime = Ts <= 12; % 上午的时间点
162     afternoonTime = Ts > 12; % 下午的时间点
163
164     % 上午和下午分别满足条件的时间点数
165     morningSum = sum(Ip(morningTime) > morningThresh);
166     afternoonSum = sum(Ip(afternoonTime) > afternoonThresh);
167     % 计算总时长
168     totalTime = (morningSum + afternoonSum) * dt; % 总时间，乘
以时间步长转换为小时
169 end

```

t3_MOPSO_k.m

```

1 N=100; %群体粒子个数

```

```

2  D=2;                                %粒子维数
3  T=100;                              %最大迭代次数
4  c1=1.5;                             %学习因子1
5  c2=1.5;                             %学习因子2
6  w=0.8;                              %惯性权重
7  Xmax=20;                            %位置最大值
8  Xmin=-20;                           %位置最小值
9  Vmax=10;                            %速度最大值
10 Vmin=-10;                           %速度最小值
11 Wmax=0.8;                           %惯性权重最大值
12 Wmin=0.4;                           %惯性权重最小值
13 %初始化个体
14 x=rand(N,D)*(Xmax-Xmin)+Xmin;
15 v=rand(N,D)*(Vmax-Vmin)+Vmin;
16
17 %初始化个体最优位置和最优值
18 p = x;
19 pbest = zeros(N, 2);
20 for i = 1:N
21     pbest_m = All_year_and_time(x(i, :));
22     pbest(i, 1) = pbest_m(1);
23     pbest(i, 2) = pbest_m(2);
24 end
25
26 % 初始化全局最优解集
27 g = p;
28 gbest = pbest;
29
30
31 % 迭代优化
32 for t = 1:T
33     % 计算适应度缩放因子
34     fitness = calculateFitness(pbest);
35     totalFitness = sum(fitness);
36     weight = fitness / totalFitness;

```

```

37
38 % 更新粒子速度和位置
39 for i = 1:N
40     r1 = rand(1, D);
41     r2 = rand(1, D);
42     v(i, :) = w * v(i, :) + c1 * r1 .* (p(i, :) - x(i, :))
+ c2 * r2 .* (g(i, :) - x(i, :));
43     x(i, :) = x(i, :) + v(i, :);
44
45 % 边界条件处理
46     x(i, :) = max(Xmin, min(Xmax, x(i, :)));
47     v(i, :) = max(Vmin, min(Vmax, v(i, :)));
48 end
49
50 % 更新个体最优位置和最优值
51 for i = 1:N
52     pbest_m = All_year_and_time(x(i, :));
53     f1 = pbest_m(1);
54     f2 = pbest_m(2);
55
56     if f1 > pbest(i, 1) && f2 > pbest(i, 2)
57         p(i, :) = x(i, :);
58         pbest(i, :) = [f1, f2];
59     end
60 end
61
62
63 % 更新全局最优解集
64 [g, gbest] = updateGlobalBestSet(x, pbest, g, gbest);
65 end
66
67 % 输出结果
68 disp('最优解集: ');
69 disp(gbest);
70 disp('最优值: ');

```

```

71 disp(g);
72
73 %画出散点图
74 a = g(:,1)';
75 b = g(:,2)';
76 % 生成逻辑索引
77 indices = abs(a) == 20 & abs(b) == 20; % 去除掉边界上的点
78 % 删除满足条件的元素
79 a(indices) = [];
80 b(indices) = [];
81 % 绘制散点图
82 figure;
83 scatter(a/20*90, b/40*90 + 45);
84 xlabel('光伏板方向角');
85 ylabel('光伏板倾斜角');
86 title('优化后光伏板方向角与倾斜角关系');
87
88 % 更新全局最优解集函数
89 function [g, gbest] = updateGlobalBestSet(x, pbest, g, gbest)
90     % 遍历每个粒子
91     for i = 1:size(pbest, 1)
92         % 如果当前粒子的最优值优于全局最优解，则更新全局最优解集
93         if pbest(i, 1) > gbest(i, 1) && pbest(i, 2) > gbest(i,
94             2)
95             g(i, :) = x(i, :);
96             gbest(i, :) = pbest(i, :);
97         end
98     end
99 end
100 % 适应度缩放函数
101 function fitness = calculateFitness(pbest)
102     % 适应度缩放因子可以根据不同的方法进行计算，这里使用简单的
    线性缩放

```



```

103     minFitness = min(pbest, [], 1);
104     maxFitness = max(pbest, [], 1);
105     fitness = (pbest - minFitness) ./ (maxFitness - minFitness
    );
106 end
107
108
109 function All_year= All_year_and_time(position)
110 Ap=position(1)/40*pi; %光伏板方向角赋值
111 Thetap=position(2)/80*pi+pi/4; %光伏板倾斜角赋值
112 % 太阳常数数据
113 I_monthly = [1405, 1394, 1378, 1353, 1334, 1316, 1308, 1315,
    1330, 1350, 1372, 1392];
114 % 一年中每个月的天数
115 days_in_month = [31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30,
    31];
116 %全年太阳常数数据
117 I_daily = repelem(I_monthly, days_in_month);
118 All_year_E=zeros(1,365);
119 All_year_Time=zeros(1,365);
120 for n = 1:365
121     [AllEp_int,totalTime]=Ep_int_and_totalltime(n,I_daily(n),Ap
    ,Thetap);
122     All_year_E(n)=AllEp_int;
123     All_year_Time(n) = totalTime;
124 end
125 All_year(1)= sum(All_year_E);
126 All_year(2)= sum(All_year_Time);
127 end
128
129
130 function [AllEp_int,totalTime]=Ep_int_and_totalltime(n,I0,Ap,
    Thetap)
131     L=114+19/60; %目标经度
132     Ls=120; %时区经度

```

```

133     phi=30+35/60; %目标纬度
134     delta=23.45*sind((284+n)/365); %赤纬角
135
136     beta=-5.745e-4; %衰减系数
137     d=1000;
138     dt=0.1;
139     Ts=[5:dt:20]; %步距0.01
140     ns=length(Ts); %步数
141     omega=180-Ts*15+(Ls-L); %时角
142     sinh=sind(phi)*sind(delta)+cosd(phi)*cosd(delta)*cosd(
omega); %太阳高度角正弦值
143     sinAs=cosd(delta)*sind(omega)./sqrt(1-sinh.^2); %太阳方向
角正弦值
144     costhetas=sqrt(1-sinh.^2)*sin(Thetap).*(sinAs*sin(Ap)+sqrt
(1-sinAs.^2)*cos(Ap))+sinh*cos(Thetap); %法线夹角余弦值
145     Ip=zeros(1,ns); %直射强度初始化
146     %太阳板表面光强
147     for i=1:ns
148         if costhetas(i)<=0 || sinh(i)<=0
149             Ip(i)=0;
150         else
151             Ip(i)=I0*exp(d*beta/sinh(i)).*costhetas(i);
152         end
153     end
154     AllEp_int=trapz(Ts,Ip)/1000; %对光强进行梯形法积分并换算为
千瓦时
155
156     morningThresh = 150; % 上午阈值
157     afternoonThresh = 100; % 下午阈值
158     morningTime = Ts <= 12; % 上午的时间点
159     afternoonTime = Ts > 12; % 下午的时间点
160
161     % 上午和下午分别满足条件的时间点数
162     morningSum = sum(Ip(morningTime) > morningThresh);
163     afternoonSum = sum(Ip(afternoonTime) > afternoonThresh);

```

```
164     % 计算总时长
165     totalTime = (morningSum + afternoonSum) * dt; % 总时间，乘
    以时间步长转换为小时
166 end
```