计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications [doi: 10.15888/j.cnki.csa.009688]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于 VE-GEP 算法的 $PM_{2.5}$ 浓度预测①

王超学, 邹　飞

(西安建筑科技大学 信息与控制工程学院, 西安 710055)
通信作者: 邹　飞, E-mail: zoufei@xauat.edu.cn

摘　要: 准确预测 $PM_{2.5}$ 浓度对于公众健康和环境保护具有重要意义, 但其非线性、多变性以及复杂性的特点导致难以准确预测. 基于此, 本文针对传统 GEP 存在的不足, 提出了一种基于病毒进化的基因表达式编程算法 (VE-GEP) 来预测 $PM_{2.5}$ 浓度. 该算法在 GEP 的基础上引入了复活机制与诱变重启机制. 复活机制能去除种群中的劣质个体, 改善种群中个体的质量; 诱变重启机制通过引入优质基因和新的个体, 提高种群的多样性, 增强算法的寻优能力. 实验结果表明, VE-GEP 算法相较于 GEP、DSCE-GEP 和 CNN-LSTM 在春季、夏季和秋季中的预测模型均有不同程度的提高, 拟合度分别提高 1.28%/0.1%/0.13%、1.86%/1.29%/0.42%、0.57%/0.24%/0.29%, 为 $PM_{2.5}$ 浓度预测研究提供了新的思路和方法.

关键词: 基因表达式编程; 复活机制; 诱变重启机制; 病毒进化; $PM_{2.5}$ 浓度预测

引用格式: 王超学,邹飞.基于 VE-GEP 算法的 $PM_{2.5}$ 浓度预测.计算机系统应用. http://www.c-s-a.org.cn/1003-3254/9688.html

$PM_{2.5}$ Concentration Prediction Based on VE-GEP Algorithm

WANG Chao-Xue, ZOU Fei

(College of Information and Control Engineering, Xi'an University of Architecture and Technology, Xi'an 710055, China)

Abstract: Accurate prediction of $PM_{2.5}$ concentration is essential for public health and environmental protection, but its nonlinearity, variability, and complexity make it difficult. Based on this, this study proposes a gene expression programming algorithm based on virus evolution (VE-GEP) to predict $PM_{2.5}$ concentration in response to the shortcomings of traditional GEP. The algorithm introduces a resurrection mechanism and a mutagenic restart mechanism based on GEP. The resurrection mechanism removes poor-quality individuals from the population and improves individual quality in the population. The mutagenic restart mechanism increases population diversity and enhances algorithm optimization-seeking ability by introducing high-quality genes and new individuals. Experimental results show that the VE-GEP algorithm improves the prediction models to different degrees compared to GEP, DSCE-GEP, and CNN-LSTM in spring, summer, and fall, with improvements in the fitness of 1.28%/0.1%/0.13%, 1.86%/1.29%/0.42%, and 0.57%/0.24%/0.29%, respectively, which provides new ideas and methods for $PM_{2.5}$ concentration prediction studies.

Key words: gene expression programming; resurrection mechanism; mutagenic restart mechanism; virus evolution; $PM_{2.5}$ concentration prediction

雾霾天气的频繁出现已经成为中国乃至全球的严重环境问题, 而 $PM_{2.5}$ 作为构成雾霾天气的主要因素, 其浓度的增加不仅会对人体健康产生影响[1], 还会引起气候的变化[2]. 在中国, $PM_{2.5}$ 污染问题尤为严重, 已经

① 基金项目: 国家自然科学基金面上项目 (62072363); 陕西省自然科学基础研究计划面上项目 (2019JM-167)
收稿时间: 2024-04-30; 修改时间: 2024-05-20, 2024-06-12; 采用时间: 2024-06-26; csa 在线出版时间: 2024-09-24

成为主要的污染物之一. 据统计, 在全国 339 个地级及以上城市中, 有高达 86 个城市的 $PM_{2.5}$ 浓度超标, 这意味着近 1/4 的城市居民正生活在 $PM_{2.5}$ 浓度超标的环境中[3]. 因此, 探索有效的 $PM_{2.5}$ 浓度预测模型, 对于人类健康和环境保护至关重要.

目前, 为了应对 $PM_{2.5}$ 的污染问题, 许多研究人员都在寻求更好的预测方法对 $PM_{2.5}$ 浓度进行预测研究, 以便制定有效的规划和应对措施. 可以通过基于物理化学反应的原理模型与基于深度学习模型进行 $PM_{2.5}$ 浓度的预测. 基于物理化学变化原理的方式强调了对大气中物理和化学过程的深入理解, 包括污染物的生成和转化等, 从而对 $PM_{2.5}$ 浓度进行建模. Wang 等[4]使用 WRF-Chem 模型模拟了汾渭平原 $PM_{2.5}$ 的复杂空气污染情况. 秦思达等[5]使用 WRF-CMAQ 模型对辽宁中部城市群的 $PM_{2.5}$ 化学组分进行了模拟分析. 随着人工智能的快速发展, 深度学习已成为先进人工智能的前沿领域, 可通过深度学习预测 $PM_{2.5}$ 浓度. 基于深度学习的预测方法是通过构建复杂的神经网络模型来捕捉 $PM_{2.5}$ 浓度与其相关影响因素之间的关系. Hu 等[6]提出一种基于小波变换和长短期记忆的 $PM_{2.5}$ 浓度组合预测模型, 小波变换用于在多个尺度上分解和细化相关因素的时间序列. 并利用长短期记忆网络对不同尺度的时间序列进行训练, 通过重构生成最终的预测结果. Zhang 等[7]提出了一种基于时间差分的图变压器网络, 能够从时间序列 $PM_{2.5}$ 数据中学习长期时间依赖关系和复杂关系, 用于空气质量 $PM_{2.5}$ 预测. Li 等[8]通过将卷积神经网络与长短期记忆神经网络相结合, 建立了一种混合模型 CNN-LSTM, 用于预测 $PM_{2.5}$ 浓度. Liu 等[9]提出了一种基于长短期记忆网络的注意力机制, 使用长短期记忆网络对邻近区域的空气质量进行了初步预估, 随后采用基于 XGBoost 的集成方法, 将初步预估结果与天气预报进行有效结合, 从而实现了对 $PM_{2.5}$ 浓度的精准预测.

可以发现, 虽然上述模型能较好地预测 $PM_{2.5}$ 浓度, 但无法直接给出明确的函数表达式, 难以揭示 $PM_{2.5}$ 浓度与各影响因素之间的函数关系. 相比之下, 基因表达式编程不仅具备与神经网络一样强大的泛函学习能力, 还能够针对具体问题构建出相应的数学模型. 因此, 将 GEP 应用于 $PM_{2.5}$ 浓度预测中是很有价值的. 但 GEP 与其他进化算法一样, 在解决具体问题时, 常常会出现收敛速度慢、早熟收敛等情况[10]. 为解决此问题, 充分利用 GEP 算法的优势, 对传统 GEP 算法进行改进, 提出一种基于病毒进化的基因表达式编程算法 (VE-GEP) 对 $PM_{2.5}$ 浓度进行预测建模, 并分析该算法用于 $PM_{2.5}$ 浓度预测的有效性与先进性.

1 方法

1.1 GEP 简介

基因表达式编程 (GEP) 模拟一般生物的遗传和进化, 它结合了遗传算法 (GA) 和遗传编程 (GP) 的优点, 实现了基因型 (定长字符串) 与表现型 (表达式树) 之间的分离与转化, 能够使用简单的编码方式来解决复杂问题, 比传统的进化算法效率高出 2–4 个数量级. GEP 不需要先验知识为指导, 不局限于问题的具体领域, 非常适用于解决分类问题[11]和符号回归问题[12]. 目前该算法也已成功应用到了建筑物的能源性能预测[13]、土壤渗透系数预测[14]、混凝土抗压强度预测[15]和城市用水量预测[16]等多个领域.

在 GEP 中, 种群是最高实体, 它是多个个体 (染色体) 的集合. 个体的表示是算法的关键部分, GEP 个体包含一个或多个等长基因. 每个基因由头部与尾部组成, 采用线性符号串编码, 符号串即为基因的基因型. 为保证产生的基因是合法的, 头尾的长度需满足一定的关系, 具体如式 (1) 所示:

$$t = h \times (n-1) + 1 \tag{1}$$

其中, t 为基因尾部的长度, h 为基因头部的长度, n 为所有函数中的最大目数.

头部和尾部是基因的重要组成部分, 头部的基因元素是通过函数符集合 F 与终结符集合 T 中的元素构成, 而尾部的基因元素仅由终结符集合 T 中的元素构成. 其中函数符集合 F 的元素组成十分广泛, 包含问题领域相关的所有函数符号; 终结符集合 T 的元素组成通常为算法的自变量个数.

1.2 基于病毒进化的基因表达式编程算法

病毒是一类特殊的生物, 其进化速度非常快, 能够快速适应不同的环境变化. 多样性是病毒生存和进化的首要条件[17], 与一般生物相比, 病毒可以通过不同的方式快速地变异和重组, 增加病毒遗传的多样性. 病毒重组可以发生在同一种病毒的不同株之间, 也可以发生在不同种类的病毒之间. 对于活性病毒与灭活病毒可以通过交叉复活的方式, 从而产生具有新遗传特性

的病毒. 病毒的突变可以自发产生, 也可以由物理、化学诱变剂诱导产生, 使得病毒更好地适应环境变化和宿主免疫反应. 病毒依赖于宿主来进行生命活动, 利用宿主细胞内的物质和能量来维持其生命周期. 相较于生物体之间的水平基因转移, 病毒与其宿主之间的基因流动是病毒繁殖的关键特征. 病毒的进化过程就是一个不断从宿主获取基因的过程[18]. 这些优势有助于病毒在各种环境条件下快速适应和进化, 从而更好地生存和繁殖.

受此启发, 本文提出了一种基于病毒进化的基因表达式编程算法 (VE-GEP), 在 GEP 基础上引入了复活机制与诱变重启机制, 复活机制通过将种群中的不可行解复活成可行解, 来改善种群中个体的质量. 诱变重启机制通过增强种群的多样性, 避免算法过早收敛到局部最优解. 其算法具体内容如图 1 所示.

1.2.1 基本定义

VE-GEP 算法中涉及的相关概念定义如下.

定义 1. 病毒种群, 是指由病毒个体组成的种群, 对应所求问题的解集. 病毒种群又可以分为存活病毒种群与灭活病毒种群.

定义 2. 病毒个体, 在编码方式上采用与传统基因表达式编程中个体相同的策略.

定义 3. 重启病毒个体, 是指病毒个体将其基因信息传递给宿主细胞, 并从宿主细胞中获取新的基因信息, 从而产生的新个体. 其编码方式遵循了与病毒个体相同的规则, 并且两者在编码长度上保持一致.

定义 4. 灭活病毒个体, 是指在病毒个体中携带了灭活基因的个体. 例如, 在病毒种群中, 若第 i 个病毒个体的第 j 位上的函数符为“/”, 则需要对第 j 位的两个操作数进行计算. 在这个过程中, 若第 2 个操作数为 0, 这时计算后得到的值就是一个非法结果, 那么第 j 位上的基因就是一个灭活基因, 该个体就是一个灭活病毒个体.

定义 5. 存活病毒个体, 是指在病毒个体中未携带灭活基因的个体.

1.2.2 复活机制

在个体进化积累的过程中, 可能会受到基因遗传操作的破坏, 从而产生一些灭活病毒个体. 灭活病毒个体不仅增加了计算的成本, 还可能导致算法需要更多的迭代次数才能找到最优解. 为了避免或减少这些灭活病毒个体, 本文通过借鉴病毒进化中交叉复活的思想, 设计复活机制操作, 利用存活病毒个体对灭活病毒个体中的灭活基因或灭活基因片段进行等位替换.

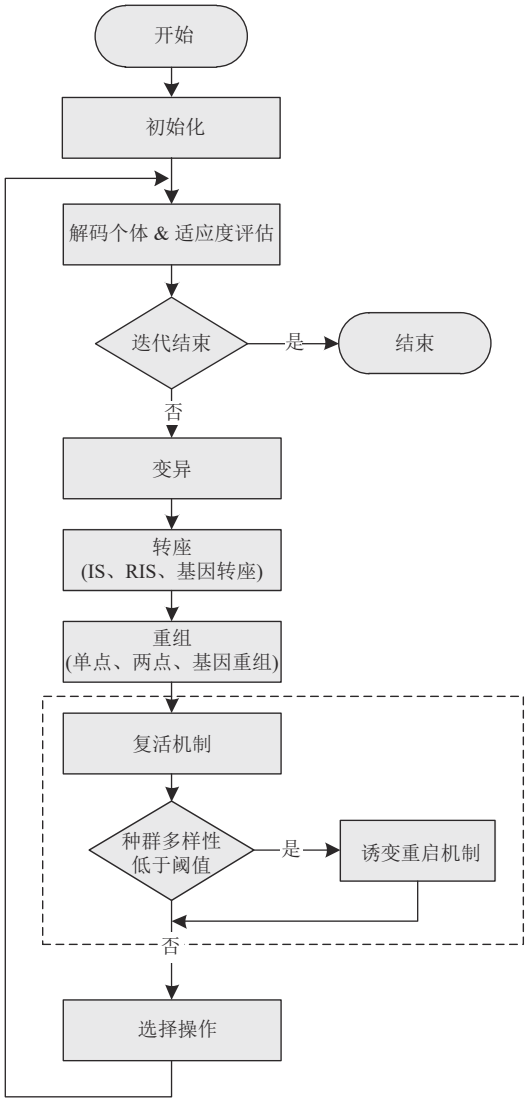

图 1 VE-GEP 算法流程图

在进行复活机制操作时, 设置相同的灭活病毒个体数量与存活病毒个体数量, 这里采用的存活病毒个体是从存活病毒种群中随机选择的优质个体. 具体分为以下两种情况, 依据灭活病毒个体基因位上的灭活基因与存活病毒个体中的等位基因元素是否相同分别进行等位基因元素的替换或等位基因元素片段的替换. 如果相同, 则进行等位基因元素片段的替换; 如果不同, 则进行等位基因元素替换. 计算式如 (2) 所示:

$$inVIRUS = \begin{cases} surVIRUS\,(x_m), & x_m \neq x_n \\ surVIRUS\,(a_m : a_n), & x_m = x_n, a_n > a_m \end{cases} \tag{2}$$

其中, *inVIRUS* 为灭活病毒个体, *surVIRUS* 为存活病毒个体, x_m 为灭活病毒个体中存在的第 m 个灭活基因元素, x_n 为存活病毒个体中与 x_m 的等位基因元素. a_m、a_n 均为替换的基因位点, $a_m \in \left(S_x^{m-1}, S_x^m\right), a_n \in \left(S_x^m, S_x^{m+1}\right)$, S_x^m为第 m 个灭活基因的基因位, a_m:a_n 为 a_m–a_n 一段基因元素片段.

1.2.3 种群信息熵

随着进化迭代的发生, 可能会出现种群趋同, 不利于种群进化. 为此, 本文采用信息熵[19]作为种群多样性的衡量标准. 信息熵是一种定量计量标尺, 用于表示状态的多样性和丰富程度. 具体计算如下所示.

(1) 统计第 i 个函数符或终结符在种群的同一基因位置 j 上出现的次数 C_{ij}.

(2) 求第 i 个函数符或终结符在种群的同一基因位置 j 上出现的概率 P_{ij}, 计算式如式 (3) 所示:

$$P_{ij} = \frac{C_{ij}}{M} \tag{3}$$

其中, M 为种群的规模.

(3) 计算种群的信息熵, 计算式如式 (4) 所示:

$$H = \frac{1}{L}\sum_{j=1}^{L}\sum_{i=1}^{N} - P_{ij}\log P_{ij} \tag{4}$$

其中, L 为每个个体的总长度, N 为函数符和终结符的总数.

依据设定的信息熵阈值判断是否需要对当前病毒种群采用诱变重启机制. 种群的最大进化代数为 *Max*, 分别设置多样性判断阶段 1–*Max*/3、*Max*/3–2×*Max*/3 和 2×*Max*/3–*Max*, 及对应的种群多样性阈值 h, 理想种群信息熵 H_i, 当代种群信息熵 H_c. 若 H_c<h×H_i, 则对病毒种群中的病毒个体采取诱变重启机制.

1.2.4 诱变重启机制

在 GEP 中, 随着迭代次数的不断增加, 种群中的个体逐渐收敛于适应度较高的区域, 表现出较高的相似性, 致使种群的多样性降低. 在个体变异时还存在一定的盲目性, 可能会产生出与问题无关或者甚至有害的基因. 当种群中的个体接近某个局部最优解时, 变异可能无法帮助算法跳出局部区域, 而是在附近进行搜索, 这也是导致算法在后期陷入停滞的原因. 为了解决上述问题, 在 VE-GEP 算法中采用诱变重启机制, 该机制包含诱变操作和重启操作. 诱变操作是通过记录自然进化过程中的优质变异基因对个体进行变异, 探索可能存在最优值的搜索空间, 有助于提高算法的收敛性. 重启操作是将诱变操作后适应度值低的病毒个体淘汰, 同时引入等量的重启病毒个体, 保持种群个体数量的一致性, 并提高种群的多样性.

(1) 诱变操作

在进行诱变操作时, 对病毒个体中的每个基因位设置一个相对应的诱变基因库, 用来保存自然进化过程中的优质变异基因. 诱变基因库随着进化的发生始终保持动态更新, 若在自然进化中经过自然变异后的个体适应度值增加, 则称这类变异基因为优质变异基因, 这类基因将直接存入诱变基因库中; 反之, 若个体适应度值降低, 则称为劣质变异基因, 如果这类基因存在于诱变基因库中, 则需要从诱变基因库中删除. 计算如式 (5) 所示:

$$G = \begin{cases} add(j), & f_m > f_p \\ pop(j), & f_m < f_p, j \in G \end{cases} \tag{5}$$

其中, G 为诱变基因库, j 为变异基因, $add(j)$ 表示将基因 j 存入诱变基因库 G 中, $pop(j)$ 表示将基因 j 从诱变基因库 G 中删除, f_m 为变异后的个体适应度值, f_p 为变异前的个体适应度值.

诱变操作是针对最大规模的诱变基因库所对应基因位上的基因进行变异, 变异基因从诱变基因库中进行选取. 在诱变基因库中, 诱变基因选取概率通过式 (6) 进行计算.

$$P(i) = \frac{\sum_{0}^{len(G)} G(i)}{len(G)}, i \in G \tag{6}$$

其中, $P(i)$ 为选取基因 i 的概率, i 为诱变基因库中的基因, $len(G)$ 为诱变基因库规模.

将选取到的诱变基因通过式 (7) 进行诱变操作.

$$p_n^s(i) = G(k), k \in G \tag{7}$$

其中, p_n^s为第 s 代种群中的第 n 个个体, i 为个体 n 的基因, k 为诱变基因库中的基因. 若个体为最优个体时, 诱变操作后个体适应度值增加, 则进行诱变操作; 否则, 不进行诱变操作.

(2) 重启操作

为了提高种群的多样性, 采取重启操作, 将重启病毒个体引入当前病毒种群中, 帮助病毒种群进化. 在进化过程中将适应度低的病毒个体进行淘汰, 用重启病毒个体对其进行替换. 根据信息熵的计算原理, 如果种群内的个体在同一基因位上都表现出不同的等位基因

元素时, 那么该种群的信息熵将会达到最大, 此时的种群多样性表现最佳. 结合这个思想, 重启病毒个体采用随机生成的方式, 且生成的重启病毒个体与当前被淘汰的病毒个体等位基因元素不同. 计算如式 (8) 所示:

$$VIRUS(i) = newVIRUS(j),\ i \neq j \tag{8}$$

其中, *VIRUS* 为病毒个体, *newVIRUS* 为重启病毒个体, i 为 *VIRUS* 中的基因元素, j 为重启病毒个体中与基因元素 i 不相同的等位基因元素.

2 实验结果与分析

2.1 数据集

本实验采用西安市 2018 年 3 月 1 日–2022 年 2 月 28 日每日的空气质量数据 (PM_{10}, NO_2, CO, SO_2, O_3) 与气象数据 (露点, 降水, 气压, 温度, 风速) 为实验的样本数据, 分别来源于中国空气质量在线监测分析平台与美国国家气候数据中心, 并将 70% 的数据组成训练集, 30% 的数据组成测试集.

2.2 评价指标

本实验分别以拟合度 R^2、均方根误差 *RMSE* 以及平均绝对误差 *MAE* 对模型的预测性能进行评价. 通过拟合优度 R^2 近似表征模型学习到的有用信息的量, 计算如式 (9) 所示. 式中 *SSE* 与 *SST* 分别为残差平方和与总离差平方和, 具体计算如式 (10) 与式 (11) 所示. *RMSE* 是实际观测值与预测值偏差的平方与观测次数 n 比值的平方根, 计算式如式 (12) 所示. *MAE* 表示实际观测值和预测值之间绝对误差的平均值, 计算式如式 (13) 所示.

$$R^2 = 1 - \frac{SSE}{SST} \tag{9}$$

$$SSE = \sum_{j=1}^{n}\left(y_j - \hat{y}_j\right)^2 \tag{10}$$

$$SST = \sum_{j=1}^{n}\left(y_j - \bar{y}\right)^2 \tag{11}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{j=1}^{n}\left(y_j - \hat{y}_j\right)^2} \tag{12}$$

$$MAE = \frac{1}{n}\sum_{j=1}^{n}\left|y_j - \hat{y}_j\right| \tag{13}$$

其中, y_j 表示实际观测值, $\hat{y}_j$ 表示预测值, $\bar{y}$ 为变量 y 的平均值.

2.3 实验环境及参数设置

实验环境: Windows 10 64 位, i5-7300HQ, 16 GB 内存. 基因重组为 0.1, 其余的具体实验运行参数设置如表 1 所示, 其中函数符集合中 S 代表平方运算, R 代表开方运算, exp 代表 e^x, ~代表 10^x.

表 1 实验参数设置

参数	设置值
种群规模	100
最大进化代数	200
函数符集合	+、−、*、/、log、exp、ln、~、S、R、sin、cos、abs、tan、cot
终结符集合	影响因素
连接符	+
头部长度	10
基因个数	6
转座率	0.1
重组率	0.3
转座元素长度	{1, 2, 3, 4, 5}
变异率	0.03
重启病毒个体规模	30
种群多样性阈值	0.2 0.15 0.1

2.4 $PM_{2.5}$ 浓度变化特征分析

$PM_{2.5}$ 浓度具有明显的季节变化趋势, $PM_{2.5}$ 的浓度常会受到气象因子和大气污染物的影响, 而这些因素往往随季节的变化而变化[20–22]. 为了更直观了解 $PM_{2.5}$ 浓度变化特征, 对 2018–2021 年四季的 $PM_{2.5}$ 浓度进行均值计算, 并做出相对应的四季 $PM_{2.5}$ 浓度均值变化图, 如图 2 所示.

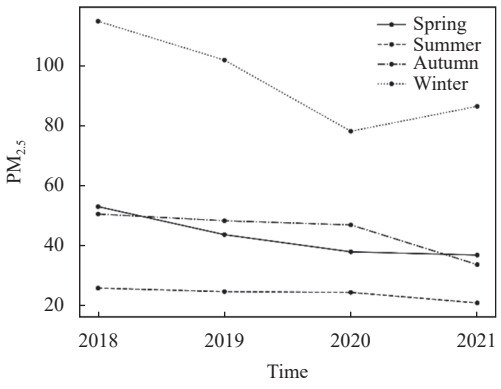

图 2 四季的 $PM_{2.5}$ 浓度均值变化图

从图 2 可以看出, 冬季的 $PM_{2.5}$ 浓度明显高于其他季节, 而春季和秋季的 $PM_{2.5}$ 浓度相对较低, 夏季则呈现出最低的 $PM_{2.5}$ 浓度. 不同季度的 $PM_{2.5}$ 浓度呈现出明显的差异性, 因此本文按四季划分对 $PM_{2.5}$ 浓度进行

预测建模.

2.5 VE-GEP 实验结果与分析

为了详细描述 VE-GEP 在 $PM_{2.5}$ 浓度预测建模中的应用, 本实验将春季、夏季、秋季和冬季分别采用 VE-GEP 算法进行建模, 最终得到四季的 $PM_{2.5}$ 浓度预测值与实际值的对比曲线分别如图 3–图 6 所示, 图中虚线代表的是预测值, 实线代表的是实际值.

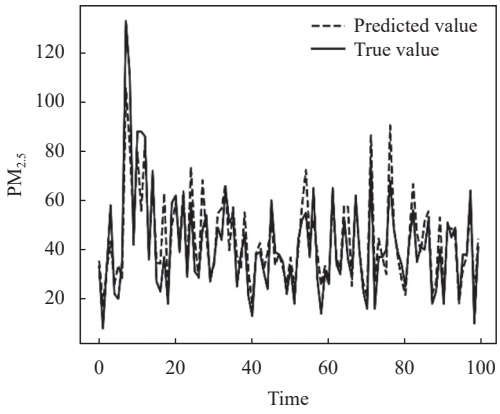

图 3 春季预测结果对比图

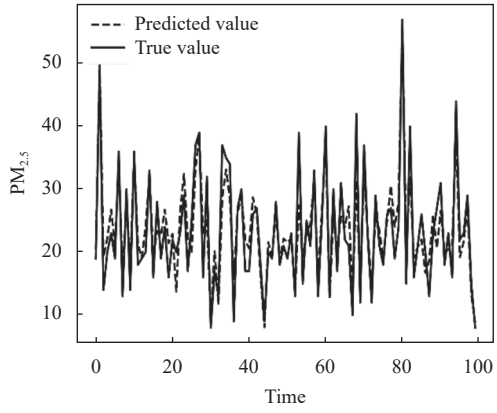

图 4 夏季预测结果对比图

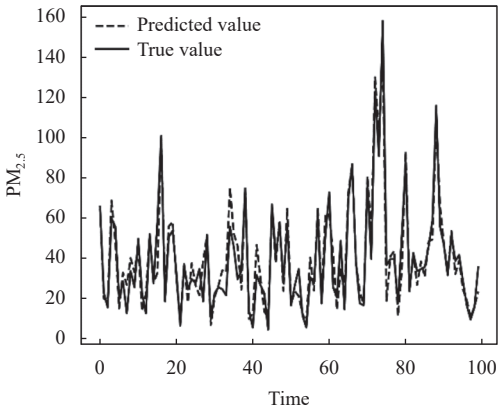

图 5 秋季预测结果对比图

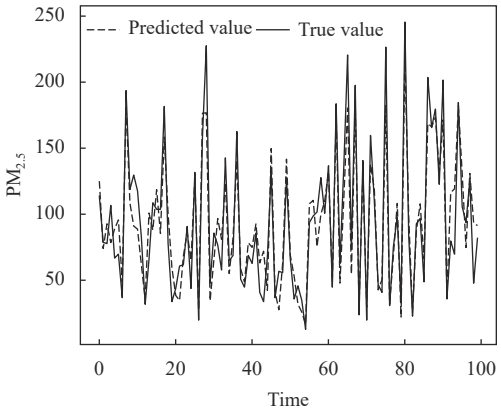

图 6 冬季预测结果对比图

通过观察图 3–图 6, 可以发现采用 VE-GEP 算法对 $PM_{2.5}$ 浓度进行预测, 得到的预测值与实际值基本一致, 该算法能较好地预测 $PM_{2.5}$ 浓度. 基于该算法得到的四季的预测模型分别如式 (14)–式 (17) 所示. 其中 x_0–x_9 分别代表 PM_{10}、NO_2、CO、SO_2、O_3、露点、降水、气压、温度和风速.

$$y=\sqrt{x_4}+x_3+\mathrm{e}^{x_2}+\frac{|x_3+2x_8-x_0|}{\ln|x_4+x_9|}+10^{x_2}+x_2 \tag{14}$$

$$y=\sqrt{x_0}\times x_2+\sqrt{x_0}\times\log x_0+\frac{x_0\times x_2}{x_3}+x_2+\cos 10^{x_6} \tag{15}$$

$$y=\frac{x_0}{\log x_0}+\cos x_8+x_0\times\log x_2+\cos(\log 10^{x_6})+\frac{x_0}{x_1}+(\cos x_6)^2 \tag{16}$$

$$y=x_0+x_2+x_6+2x_5-\frac{x_0^2}{x_7\times|\log x_1^2|}-2x_8+\sin x_2-\frac{{x_0}^2}{x_7\times|x_2|} \tag{17}$$

通过式 (14)–式 (17) 可以看到, $PM_{2.5}$ 的浓度与各影响因素之间的函数关系. 在不同季节中污染物对 $PM_{2.5}$ 浓度产生的影响也各不相同, 在各个季节中 PM_{10} 对 $PM_{2.5}$ 浓度都产生了较大的影响. NO_2、SO_2 和 CO 对 $PM_{2.5}$ 浓度的影响则因季节而异, 其中秋季和冬季影响更强的因素是 CO 和 NO_2, 而春季和夏季影响更强的因素是 CO 和 SO_2. 此外, 温度、风速和气压等因素对 $PM_{2.5}$ 浓度的形成也具有季节性的不同影响. 因此, 采用 VE-GEP 算法对 $PM_{2.5}$ 浓度进行预测建模, 还能够捕捉与当季的 $PM_{2.5}$ 浓度密切相关成函数关系的影响因子, 综合考虑各种污染物的排放特征和季节变化

特点, 采取针对性的措施来降低 $PM_{2.5}$ 浓度.

2.6 实验结果比较与分析

为了进一步验证 VE-GEP 算法的先进性, 将 VE-GEP 算法与 GEP、DSCE-GEP[23]和 CNN-LSTM[8]进行对比实验, 通过实验得到拟合度、均方根误差和平均绝对误差如表 2 所示.

表 2 实验对比结果

季节	VE-GEP			GEP			DSCE-GEP			CNN-LSTM		
	R^2	*RMSE*	*MAE*	R^2	*RMSE*	*MAE*	R^2	*RMSE*	*MAE*	R^2	*RMSE*	*MAE*
春季	0.8115	10.178	7.469	0.7987	10.516	7.495	0.8105	10.203	7.153	0.8102	10.211	7.650
夏季	0.8294	3.770	3.073	0.8108	3.970	3.223	0.8165	3.910	3.224	0.8252	3.443	2.773
秋季	0.8995	8.233	6.276	0.8938	8.464	6.556	0.8971	8.333	6.290	0.8966	8.224	6.804
冬季	0.8638	19.637	16.127	0.8493	20.651	17.062	0.8632	19.678	15.353	0.8766	18.691	14.621

通过表 2 可以看出, 这些算法的预测值与真实值都比较接近. VE-GEP 算法相较于其他算法在春季、夏季和秋季中的预测模型均有不同程度的提高, 冬季预测模型稍差于 CNN-LSTM, 但它无法得到 $PM_{2.5}$ 浓度与各影响因素之间的具体函数关系. 整体来说, VE-GEP 算法性能更好, 能有效应用在 $PM_{2.5}$ 浓度预测中, 为 $PM_{2.5}$ 浓度预测提供新的解决方案和思路.

3 算法性能分析

从前面的研究可以看出, VE-GEP 算法具有明显的优势. 为了进一步探究算法复活机制和诱变重启机制的改进效果, 借助 *F* 函数[24]对 VE-GEP 算法性能进行分析.

F 函数由 Ferreira[24]首次应用于 GEP 中, 具有维度高, 自变量个数少的特点, 通常被用于算法的性能分析及比较, 具体如式 (18) 所示:

$$F = 5a_n^4 + 4a_n^3 + 3a_n^2 + 2a_n + 1 \tag{18}$$

连续实验 200 次, 实验参数设置如表 3 所示, 实验结果如表 4 所示.

表 3 算法性能分析实验参数设置

参数	设置值
种群规模	70
最大进化代数	100
函数符集合	+、−、*、/
连接符	+
头部长度	6
基因个数	5
转座率	0.1
重组率	0.1
转座元素长度	{1, 2, 3}
变异率	0.022
重启病毒个体规模	20
种群多样性阈值	0.6、0.5、0.4

表 4 实验结果

算法	复活机制	诱变重启机制	成功率 (%)
GEP	—	—	76.0
GEP_1	√	—	77.5
GEP_2	—	√	84.0
VE-GEP	√	√	85.0

从表 4 可以看出, 加入复活机制后, 算法的成功率提高了 1.5%, 充分说明了该机制能够通过改善种群中解的质量来提高算法的全局寻优能力; 而加入诱变重启机制后, 算法的成功率提高了 8%, 充分说明了该机制通过增强种群的多样性, 提高了算法的全局寻优能力. 同时加入这两种机制后, 算法的成功率最高. 充分说明了算法能够在改善解质量的同时提高对优质解的开采能力, 进一步提高了算法的全局寻优能力.

4 结论

受病毒进化的启发, 本文提出了一种新的基因表达式编程算法 (VE-GEP). 该算法在 GEP 的基础上引入复活机制与诱变重启机制, 不仅可以改善解的质量, 还可以提高算法对优质解的开采能力, 增强算法的寻优能力. 并将该算法应用到 $PM_{2.5}$ 浓度预测中, 依据 $PM_{2.5}$ 浓度的季节变化特点, 分别建立了不同季节的 $PM_{2.5}$ 浓度预测模型. 同时, 将 VE-GEP 算法与其他预测模型进行对比实验. 结果表明该算法不仅预测精度更高, 还能够得到 $PM_{2.5}$ 浓度与各影响因素之间的函数关系, 对于 $PM_{2.5}$ 浓度预测研究具有重要的现实意义.

虽然该算法在 $PM_{2.5}$ 浓度预测中具有较好的实用性与较高的预测精度, 但由于引入了一些新的遗传算子, 需要更多的计算资源作为代价. 在下一步工作中, 将致力于解决这些约束, 并进一步改善算法性能.

参考文献

1 Kim Y, Manley J, Radoias V. Medium- and long-term consequences of pollution on labor supply: Evidence from Indonesia. IZA Journal of Labor Economics, 2017, 6: 5. [doi: 10.1186/s40172-017-0055-2]

2 王薇, 陈明. 城市绿地空气负离子和 $PM_{2.5}$ 浓度分布特征及其与微气候关系——以合肥天鹅湖为例. 生态环境学报, 2016, 25(9): 1499–1507.

3 2022 年中国生态环境状况公报 (摘录). 环境保护, 2023, 51(Z2): 64–81.

4 Wang YX, Cao L, Zhang T, *et al*. Simulations of summertime ozone and $PM_{2.5}$ pollution in Fenwei Plain (FWP) using the WRF-Chem model. Atmosphere, 2023, 14(2): 292. [doi: 10.3390/atmos14020292]

5 秦思达, 王帆, 王堃, 等. 基于 WRF-CMAQ 模型的辽宁中部城市群 $PM_{2.5}$ 化学组分特征. 环境科学研究, 2021, 34(6): 1277–1286.

6 Hu XK, Shi JH, He CL, *et al*. Combined prediction model of $PM_{2.5}$ concentration based on wavelet transform and LSTM. Journal of Physics: Conference Series, 2023, 2555(1): 012009. [doi: 10.1088/1742-6596/2555/1/012009]

7 Zhang Z, Zhang SQ, Zhao XM, *et al*. Temporal difference-based graph transformer networks for air quality $PM_{2.5}$ prediction: A case study in China. Frontiers in Environmental Science, 2022, 10: 924986. [doi: 10.3389/fenvs.2022.924986]

8 Li TY, Hua M, Wu X. A hybrid CNN-LSTM model for forecasting particulate matter ($PM_{2.5}$). IEEE Access, 2020, 8: 26933–26940. [doi: 10.1109/ACCESS.2020.2971348]

9 Liu DR, Lee SJ, Huang Y, *et al*. Air pollution forecasting based on attention-based LSTM neural network and ensemble learning. Expert Systems, 2020, 37(3): e12511. [doi: 10.1111/exsy.12511]

10 Xiong ZJ, Wang XJ, Li Y, *et al*. A problem transformation-based and decomposition-based evolutionary algorithm for large-scale multiobjective optimization. Applied Soft Computing, 2024, 150: 111081. [doi: 10.1016/j.asoc.2023.111081]

11 Hanandeh S. Evaluation circular failure of soil slopes using classification and predictive gene expression programming schemes. Frontiers in Built Environment, 2022, 8: 858020. [doi: 10.3389/fbuil.2022.858020]

12 Lu Q, Xu CW, Luo J, *et al*. AB-GEP: Adversarial bandit gene expression programming for symbolic regression. Swarm and Evolutionary Computation, 2022, 75: 101197. [doi: 10.1016/j.swevo.2022.101197]

13 Alzara M, Rehman MF, Farooq F, *et al*. Prediction of building energy performance using mathematical gene-expression programming for a selected region of dry-summer climate. Engineering Applications of Artificial Intelligence, 2023, 126: 106958. [doi: 10.1016/j.engappai.2023.106958]

14 Zhang RL, Zhang S. Coefficient of permeability prediction of soils using gene expression programming. Engineering Applications of Artificial Intelligence, 2024, 128: 107504. [doi: 10.1016/j.engappai.2023.107504]

15 Alabduljabbar H, Khan M, Awan HH, *et al*. Predicting ultra-high-performance concrete compressive strength using gene expression programming method. Case Studies in Construction Materials, 2023, 18: e02074. [doi: 10.1016/j.cscm.2023.e02074]

16 Mousavi-Mirkalaei P, Roozbahani A, Banihabib ME, *et al*. Forecasting urban water consumption using Bayesian networks and gene expression programming. Earth Science Informatics, 2022, 15(1): 623–633. [doi: 10.1007/s12145-021-00733-z]

17 Domingo E, Parrish C R, Holland JJ. Origin and Evolution of Viruses. 2nd ed., Amsterdam: Elsevier, 2008.

18 Koonin EV, Dolja VV, Krupovic M. The logic of virus evolution. Cell Host & Microbe, 2022, 30(7): 917–929.

19 沈慧慧, 韩生廉. 免疫算法多样性及亲和性的一种计算方法. 重庆职业技术学院学报, 2004, 13(4): 125–126.

20 孟昭伟, 张同军, 雷佩玉, 等. 西安市 $PM_{2.5}$ 浓度季节变化特征及气象影响因素解析. 实用预防医学, 2020, 27(8): 934–937.

21 张怡文, 郭傲东, 吴海龙, 等. 基于 PCA-BP 神经网络的 $PM_{2.5}$ 季节性预测方法研究. 南京林业大学学报 (自然科学版), 2020, 44(5): 231–238.

22 曾江毅, 李志生, 欧耀春, 等. 季节指数改进的 $PM_{2.5}$ 质量浓度组合预测模型研究. 广东工业大学学报, 2022, 39(3): 89–94.

23 王超学, 贾晓莉, 孙嘉诚. DSCE-GEP 算法在 $PM_{2.5}$ 浓度预测中的应用. 计算机测量与控制, 2021, 29(10): 71–76.

24 Ferreira C. Gene expression programming: A new adaptive algorithm for solving problems. Complex Systems, 2001, 13(2): 87–129.

(校对责编: 孙君艳)