

一种非线性逆优化算法

张 桐, 辜方清*

广东工业大学数学与统计学院, 广东 广州

收稿日期: 2026年1月3日; 录用日期: 2026年1月28日; 发布日期: 2026年2月4日

摘 要

针对非线性逆优化问题中最优解难以显式表示、传统方法求解受限的核心挑战, 本文提出一种嵌套进化逆优化算法。该算法构建参数种群与解种群双种群框架, 通过随机初始化生成参数向量及对应优化子问题的初始解, 利用邻域信息协同提升搜索效率; 设计保护集与待更新集动态筛选机制, 结合差分进化变异、交叉算子实现种群更新, 同时通过非支配排序快速逼近帕累托前沿。将算法应用于逆最短路径问题验证, 在7节点、15节点和25节点图上的30组实验结果表明, 所提算法求得的最优权重与初始权重的欧氏距离显著小于传统方法, 平均距离分别低至0.359、1.733和3.162, 验证了算法在全局优化与成本控制方面的优越性。该算法为地球物理学、生产计划、路径分配等领域的非线性逆优化问题提供了高效求解方案。

关键词

逆优化, 进化算法, 差分进化, 逆最短路

An Algorithm of Nonlinear Inverse Optimization

Tong Zhang, Fangqing Gu*

School of Mathematics and Statistics, Guangdong University of Technology, Guangzhou Guangdong

Received: January 3, 2026; accepted: January 28, 2026; published: February 4, 2026

Abstract

Aiming at the core challenge of nonlinear inverse optimization problems—where optimal solutions are difficult to express explicitly and traditional methods face limitations—this paper proposes a nested evolutionary inverse optimization algorithm. The algorithm constructs a dual-population framework comprising parameter and solution populations, generating initial parameter vectors and corresponding solutions for optimization subproblems through random initialization, while leveraging neighborhood information to synergistically enhance search efficiency. A dynamic screening

*通讯作者。

文章引用: 张桐, 辜方清. 一种非线性逆优化算法[J]. 应用数学进展, 2026, 15(2): 178-188.
DOI: 10.12677/aam.2026.152059

mechanism for protection and update sets is designed, integrating differential evolution mutation and crossover operators for population updating, and rapidly approximating the Pareto frontier through non-dominated sorting. The algorithm is validated on inverse shortest path problems; results from 30 experimental groups on 7-node, 15-node, and 25-node graphs demonstrate that the Euclidean distance between optimal and initial weights obtained by the proposed algorithm is significantly smaller than that of traditional methods, with average distances as low as 0.359, 1.733, and 3.162 respectively, confirming its superiority in global optimization and cost control. This algorithm provides an efficient solution framework for nonlinear inverse optimization problems in geophysics, production planning, and path allocation.

Keywords

Inverse Optimization, Evolutionary Algorithm, Differential Evolution, Inverse Shortest Path

Copyright © 2026 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

1.1. 逆优化问题的建模、应用与求解方法研究

给定一个优化问题及一个可行解, 逆优化问题旨在通过对目标函数或约束条件中的参数进行最小化调整, 使该给定可行解成为最优解[1]。逆优化问题的概念由 Tarantola 首次提出[2]。不失一般性, 逆优化问题可表述如下[3]: 设 P 为某一优化问题实例, 其中 X 为可行解集合, $c = (c_1, c_2, \dots, c_m)$ 为预先给定的参数向量, 则 P 可表示为 $P = \min_{x \in X} h(x|c)$ 对于参数向量 $d = (d_1, d_2, \dots, d_m)$, 定义 $P(d)$ 为优化问题 P 的变体, 即将 P 中的参数向量 c 替换为 d , 其表达式为:

$$P(d) = \min_{x \in X} h(x|d) \quad (1)$$

给定期望最优解 $x^0 \in X$, 记 $D(x^0) = d|h(x^0|d) = \min_{x \in X} h(x|d)$ 。基于 p 范数的逆优化问题, 是要找到参数向量 $d \in D(x^0)$, 使得 $\|d - c\|_p$ 最小, 即:

$$\min_{d \in D(x^0)} f(d) = \|d - c\|_p \quad (2)$$

其中, $\|d - c\|_p$ 表示向量 d 与向量 c 之间的 p 范数。

1.2. 逆优化的应用场景

逆优化具有广泛的应用场景, 涵盖地球物理学、生产计划[4]、医学成像与医疗规划[5]、电力分配[6]、组合优化[7]等领域。Burton 与 Toint 针对交通流和地震层析成像中的逆最短路径问题展开研究[8], 使得逆优化问题受到广泛关注。近年来, 路径分配[9]、迁移学习[10]、对抗神经网络[11]等领域的研究与工程应用发展迅速, 且诸多问题可通过逆优化的形式求解:

在大规模路径分配中, 需确保高优先级传输路径的通畅性[12], 这一问题可转化为逆优化问题——通过设备升级或减少其他路径的路由分配量, 以最小成本使给定路径成为最短路径[13];

在迁移学习(尤其是基于参数的迁移学习)中, 需基于已训练模型对目标任务的参数进行训练[14][15], 而获取最优特征的过程可视为逆优化问题。

因此, 研究者们针对各类优化问题的逆优化展开研究, 包括组合优化[16]、整数规划[17]、混合整数规划[18]、凸规划[19]、线性规划[20]、可数无穷线性规划[21]等。

1.3. 优化问题的现有求解方法与挑战

对于参数向量 d , 若优化问题 $h(x|c)$ 具有线性可解性, 或 x^0 可通过显式表达式表示[22] [23], 则可通过分析与转化方程, 采用传统优化方法求解, 如序列二次规划法[24]、割平面法[25]、牛顿法[26]、罚函数法[27]、隐式规划法[28]等。此外, 部分研究者将神经网络应用于逆优化问题求解[29] [30]。然而, 当 $h(x|c)$ 为非线性函数时, 最优解通常无法通过显式形式表示, 传统优化方法可能无法得到满意解甚至失效, 因此非线性逆优化问题的求解具有极大挑战性。其核心难点在于: 需在 $d \in D(x^0)$ 的约束下最小化 $\|d - c\|_p$, 但对于给定的参数向量 d , 难以判定其是否满足 $d \in D(x^0)$ 这一条件。

1.4. 逆优化与相关优化问题的关联

当前, 非线性逆优化研究仍处于起步阶段。逆优化可被理解为一类约束优化问题[31]——在 x^0 为参数 d 对应的优化问题最优解这一约束下, 最小化参数向量 d 与 c 之间的距离, 但逆优化问题与一般约束优化问题存在本质差异。在一般约束优化中, 可通过显式约束函数直接计算解对约束的违反程度; 而逆优化问题中, 其决策变量约束缺乏显式函数表达式, 需通过额外的优化问题判断决策变量是否满足约束条件, 因此逆优化问题的求解难度更大。

从优化过程来看, 逆优化与双层优化存在一定相似性[32]。在双层优化中, 为评估给定上层决策变量对应的上层目标函数值, 需先求解其对应的下层优化问题, 这一点与逆优化问题类似。但二者的关联仅局限于逆优化问题中参数向量 d 的搜索空间——因为逆优化中参数向量 d 的最优解已给定为 x^0 , 需搜索满足 $d \in D(x^0)$ 的参数向量 d , 以实现 $\|d - c\|_p$ 最小化。近年来, 进化算法在约束优化[33] [34]与双层优化[35] [36]领域取得了显著成效, 为逆优化问题的求解提供了新思路。

2. 主要成果

所提逆优化进化算法

本文提出了一种逆优化进化算法。首先, 该算法随机初始化一组不同的参数向量, 形成基于参数的种群。每个参数向量 d 对应一个优化子问题 $P(d)$, 并拥有一个解, 从而形成基于解的种群。参数向量相似的两个优化子问题通常具有相似的最优解[37]。因此, 针对每个参数向量, 根据这些参数向量之间的距离定义一个邻域。可以像文献[37]中那样, 利用其邻域解的信息来协同提高搜索效率。

综上所述, 本文的主要贡献如下:

提出了一种嵌套进化逆优化算法。它将逆优化问题表述为一个带约束的多目标问题。

所提出的算法用于求解给定最短路径的逆最短路径问题。

3. 关键实现技术

3.1. 算法框架

所提算法的伪代码如算法 1 所示。它包含两个种群: 基于参数的种群(P)和基于解的种群 X 。具体来说, 我们在参数域内随机生成 N 个不同的参数向量, 以形成 $(P = d^1, d^2, \dots, d^N)$, 根据这些参数向量定义了 N 个优化子问题。第 i 个子问题 $P(d^i) = \min_{x \in X} h(x|d^i)$ 是由参数向量 d^i 确定的优化问题, 它可以是线性的、非线性的, 甚至是多峰的。同时, 在变量 x 的域 X 中随机生成 $2N$ 个初始解。每个子问题 $P(d^i)$ 根据 $h(x|d^i)$ 的值从初始解中选择一个最佳解, 以形成初始解种群 $X = \{x^1, x^2, \dots, x^N\}$ 。我们将在 4.2 中参数的

种群的更新策略, 在 4.3 中描述解的种群的更新策略。

算法 1. 所提逆优化算法总框架

输入: k : 每个参数向量的邻域数量
 N : 参数向量的数量
 G : 最大迭代代数
 T : 更新 X 时的最大迭代次数
 $\mathbb{P}(d)$: 由参数向量 d 确定的优化问题
 ρ : 参数种群的更新比例

输出: d_{best}

1. $Q = \emptyset, C = \emptyset, g = 0$
2. 分别初始化参数种群 $\mathbb{P} = \{d^1, \dots, d^N\}$ 和解种群 $\mathbb{X} = \{x^1, x^2, \dots, x^N\}$, 其中包含 N 个不同的参数向量和解
3. WHILE $g \leq G$
4. FOR $i = 1$ 到 N_p
5. 从 $\mathbb{P}$ 中随机选取 d^{n_0} 、 d^{n_1} 和 d^{n_2} , 通过公式 3 生成新的参数向量 $\bar{d}^i$
6. 根据 $h(x|\bar{d}^i)$ 的值, 从 $\mathbb{X}$ 中为 $\bar{d}^i$ 选取一个最优解 $\bar{x}^i$
7. $Q = Q \cup \bar{d}^i$, $C = C \cup \bar{x}^i$
8. $\mathbb{P}' = \mathbb{P} \cup Q$, $\mathbb{X}' = \mathbb{X} \cup C$
9. 设定 $B(i) = \{i_1, \dots, i_k\}$ 为与 d^i 最接近的 k 个参数向量的索引, $i = 1, \dots, N$, $t = 0$
10. WHILE $t \leq T$
11. FOR $x^j \in \mathbb{X}'$
12. 通过变异算子 4 和交叉算子 5 生成解 $\bar{x}^i$
13. FOR $j \in B(i)$
14. IF $h(x^j | d^j) > h(\bar{x}^i | d^j)$
15. $x^j = \bar{x}^i$
16. $t \leftarrow t + 1$
17. 通过算法 2 对种群 $\mathbb{P}'$ 执行环境选择操作
18. $g \leftarrow g + 1$

3.2. 参数种群更新

对于每个参数向量, 求解其对应的子问题都非常复杂。因此, 我们的算法无法保证精确得到最优解。所以, 直接使用进化算法得到的当前解 $\arg \min P(d)$ 来更新不合格的参数向量是不合理的。因为一个错解很可能会使种群的多样性迅速丧失, 从而导致局部收敛。为此, 本文利用子问题的求解信息设计了一种温和的参数向量更新策略。所提算法的伪代码如算法 2 所示。

算法 2. 更新参数种群

输入: x^0 : 给定的可行解
 c : 指定的参数向量
 $\mathbb{P}'$: 参数向量列表
 $\mathbb{X}'$: 参数向量对应的当前解列表

输出: $\mathbb{P}$

1. 通过公式 3 计算 $\mathbb{P}'$ 中每个个体的 f_1 、 f_2 值
2. 根据 f_2 值对种群个体进行升序排序

3. $num = |\mathbb{P}'|$, $flag_i = true, i = 1, \dots, num$
4. For $i = 1$ 到 $|\mathbb{P}'|$
5. If $f_2(d^i) < 0$ 且 $num > N + \lceil 0.5\rho N \rceil$
6. $flag_i = false, num = num - 1$
7. ELSE
8. Break
9. FOR $i = |\mathbb{P}'|$ 到 1
10. IF $num > N + \lceil 0.5\rho N \rceil$
11. $flag_i = false, num = num - 1$
12. ELSE
13. Break
14. 从 $\mathbb{P}'$ 和 $\mathbb{X}'$ 中移除 $flag_i = false$ 的参数向量及其对应的当前解
15. 根据 $(f_1, |f_2|)$, 通过非支配排序选取 N 个最优的参数向量

我们将参数向量的更新任务划分为以下双目标任务:

$$\begin{aligned} \min_{d, x} & \left(f_1 = \|d - c\|_p, f_2 = h(x|d) - h(x^0|d) \right) \\ \text{s.t.} & h(x|d) \geq h(x^0|d) \end{aligned} \quad (3)$$

特别是, 我们只取 $\|x - x^0\|_p = 0$ 所在的点。我们提出的算法中的更新参数向量主要分为三个步骤:

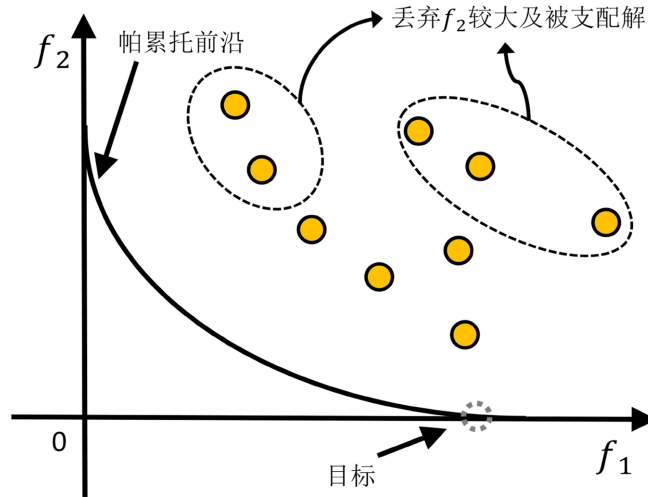


Figure 1. Update the parameter vector with a larger $f(x)$ value and a lower dominance of $[f(d), f(x)]$

图 1. 使用更大的 $f(x)$ 值和更低的 $[f(d), f(x)]$ 来更新参数向量

其中 d^{r0} 、 d^{r1} 、 d^{r2} 是从所有群体中随机选取的不同参数向量。 v^i 是 i_{th} 参数向量的变异个体。

计算保护集 P : 我们将选择一些条件优良的参数向量进行保护。保护集 P 中的参数向量不会被更新。

计算保护集有两种情况:

- 计算参数向量, 使得当前解 $P(d)$ 与 x^0 之间的距离小于 ε 。然后选择 N_p 个具有最小 $f(d) = \|d - c\|_p$ 值的个体, 将它们的索引放入保护集 P 中。

- 当计算出的保护集种群数量如果上一步中的值小于 N_p , 我们需要计算出 $N_p - \text{len}(P)$ 参数向量来补充受保护集合。计算出 $N_p - \text{len}(P)$ 个具有最小 $\|\arg \min_{x \in X} P(d) - x^0\|_p$ 值的参数向量 d , 并将它们的索引放入待保护集合 P 中。

在算法初始阶段, 该算法无法精确求解子问题, 但它通过保护特定个体来确保种群的多样性。保护集是动态计算的, 由于保护集的存在, 当前解不合格的个体不会误导参数向量的进化方向。在更新当前解信息两三代后, 当前解不合格的参数向量将被剔除, 而更优的参数向量将被加入保护集。

计算待更新集合 T : 我们选择 P 中个体之外的待更新个体, 如图 1 所示, 所选对象主要包含两部分:

- N_F 具有最大 $f(d) = \|d - c\|_p$ 值的个体
- N_D 是那些支配等级 $[f(d), f(x)]$ 处于最低水平的个体。其中 $f(d) = \|d - c\|_p$ 和 $f(x) = \|\arg \min_{x \in X} P(d) - x^0\|_p$ 。

如图 1 所示, 选择具有最低非支配水平 $[f(d), f(x)]$ 的 N_F 参数向量进行更新, 能够使目标更快地逼近帕累托前沿。选择具有最低 $f(x)$ 的 N_D 参数向量, 可使优化目标点移动到帕累托前沿中的 $f(x) = 0$ 点。

根据 P 和 T 更新参数向量: 我们将更新集合 T 中描述的所有参数向量, 因为我们无法有效判断更新后的参数向量会更好。因此, 只有一部分参数向量会被添加到 T 中, 以持续探索参数向量的优化方向。为了更新参数向量 $d^i = (d_1^i, d_2^i, \dots, d_n^i)$, 其突变体生成如下:

$$v^i = d^{r0} + F_d (d^{r1} - d^{r2}) \quad (4)$$

其中 d^{r0} 、 d^{r1} 、 d^{r2} 是从所有种群中随机选取的不同参数向量。 v^i 是参数向量 i_{th} 的突变个体。突变后, 我们执行以下操作, 将 i_{th} 突变体的 j_{th} 组件与 i_{th} 参数向量进行交叉。

$$v_j^i = \begin{cases} v_j^i & \text{随机数} > CR_d \\ d_j^i & \text{其他} \end{cases} \quad (5)$$

最后, 我们直接更新参数向量 $d^i = v^i$ 。

3.3. 当前最优解的更新

当前解的准确性对参数向量的更新有着重要影响。我们为反优化问题设计了合适的当前解更新策略。计算任意两个参数向量之间的欧氏距离, 以找出每个参数向量的 k 个最近邻参数向量。然后, 该算法利用邻居的信息来更新当前解。

不同的参数向量决定不同的子问题, 且每次参数向量更新时, 其当前的解将相应更新。根据文献[37]中的经验, 彼此接近的参数向量将具有相近的最优解。每个当前解将获取相似参数向量的当前解信息以进行更新。在所提算法的每次运行中, 参数向量的当前解将通过改进的差分进化算子进行进化, 其流程如下。

根据相似参数向量的信息, 每个参数向量的当前解将执行以下变异算子以生成不同的变异个体。

$$v^i = x^i + F_x (x^{r1} - x^{r2}) + R (x^0 - x^i) \quad (6)$$

其中 x^{r0} 、 x^{r1} 、 x^{r2} 是从与需要更新解的参数向量相似的参数向量中随机选取的不同参数向量的解。 v^i 是第 i 个解的变异体。 R 是在区间 $(0,1)$ 内随机生成的实数。 F_x 是算子的缩放因子, x^0 是给定的初始可行解。请注意, 在我们的设计中, 有两个变异体, 这使得算法能够在快速识别满足条件 $d \in D(x^0)$ 的参数向量的同时保持多样性。突变后, 参数向量的当前解将以 CR_x 的概率对突变个体进行交叉交换。 j_{th} 维的 i_{th} 新突变个体 $u^i = (u_1^i, u_2^i, \dots, u_n^i)$ 生成如下:

$$u_j^i = \begin{cases} v_j^i & rand > CR_x \\ x_j^i & otherwise \end{cases} \quad (7)$$

交叉后, 通过比较原始当前解和变异体的函数值, 将生成参数向量 i_{th} 的新当前解更新如下:

$$x^i = \begin{cases} u^i & f(u^i) < f(x^i) \\ x^i & otherwise \end{cases} \quad (8)$$

需要注意的是, 当前解的总数始终与候选参数向量的数量一致。使用两种不同变异算子的目的是探究 d 是否有潜力满足 $d \in D(x^0)$ 。两种变异体都会执行上述算子, 但在执行算子 7 后, 只会留下一个当前解。

3.4. 讨论

1) 设置保护集 P 和待更新集 T 的目的: 在参数向量的每次迭代后, 将使用进化算法继续评估新参数向量的当前解。然而, 对于复杂的目标优化问题, 进化算法可能无法在短时间内为每个参数向量求出最优解。因此, 对于参数向量的每次迭代, 我们可以通过制定温和的进化策略来保留和更新参数向量, 这能确保参数向量的进化方向正确。

2) 确保种群的多样性: 该算法的多样性主要取决于差分进化算子。所提出的算法为进化算法提供了一个框架。具有更强搜索能力和更好多样性的算子可以替代本文中的差分进化算子, 以提高所提算法的效率。

3) 在集合 T 的计算中使用非支配排序的目的: 逆优化问题是一种特殊的双目标优化问题。所提出的算法采用非支配排序能够使参数向量的目标函数快速接近帕累托前沿。

4) 使用相似参数向量的当前解来更新当前解的目的: 具有相似参数向量的目标优化问题具有相似的全局最优解, 这可以有效减少最优解的搜索时间。

4. 在逆最短路径优化中的应用

在路径分配任务或机器人路径规划任务中, 我们常常需要以较低成本改造设施, 以确保主路径的最优性。这可以建模为反向最短路径优化问题。也就是说, 尽可能小程度地变动图上的边, 使预定路径成为可能的最短路径。我们在反向最短路径优化问题的应用中测试了我们的算法。在图 2 所示的理想最短路径 $1 \rightarrow 4 \rightarrow 8 \rightarrow 12 \rightarrow 16 \rightarrow 20 \rightarrow 25$ 的情况下, 它可能是也可能不是该图的最短路径。我们需要通过最小改变, 使这条路径成为最短路径。重量 $c = (c_{1,4}, c_{4,8}, c_{8,12}, c_{12,16}, c_{16,20}, c_{20,25})$, $d = (d_{1,4}, d_{4,8}, d_{8,12}, d_{12,16}, d_{16,20}, d_{20,25})$ 。

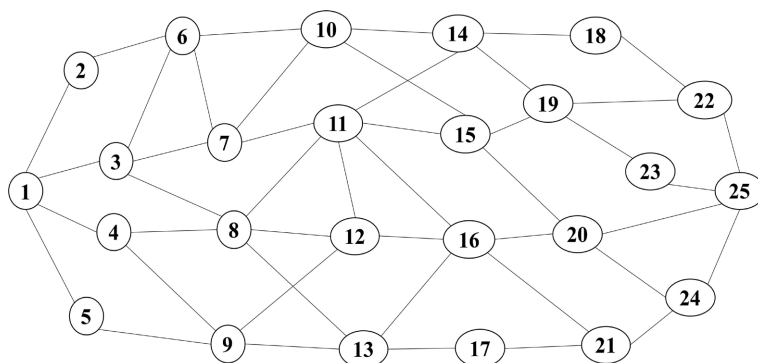


Figure 2. A graph of 25 nodes for inverse shortest path search
图 2. 用于反向最短路径搜索的 25 节点图

当给定初始最短路径及其权重 c 时, 逆优化问题可表示为:

$$\begin{aligned}
\min f(d) &= \sum_{(i,j) \in G} (d_{ij} - c_{ij})^2 \\
\text{s.t. } \sum_{(i,j) \in G} d_{ij} x_{ij}^0 &= \min \sum_{(i,j) \in G} c_{ij} x_{ij} \\
\sum_{(i,j) \in G} x_{ij} - \sum_{(j,k) \in G} x_{jk} &= 0 (\forall j \neq s, t) \\
\sum_{(s,i) \in G} x_{si} &= 1 \\
\sum_{(k,t) \in G} x_{kt} &= 1 \\
0 \leq x_{ij} &\leq 1, \forall (i, j) \in G
\end{aligned}$$

尽管经典最短路径问题本身为线性可解问题,但其对应的逆优化模型在本文框架下呈现出显著的非线性特征。具体而言,逆最短路径问题要求在参数空间中搜索一组边权,使得给定路径在全路径集合中成为最优解。该条件并不能通过有限个线性约束或 KKT 条件显式刻画,而需通过反复求解底层最短路径问题来判定,其可行性约束本质上是隐式且高度非线性的。

在本文所采用的嵌套进化框架中,参数向量与其对应子问题最优解之间形成复杂的非线性映射关系;同时,参数更新过程需在“给定路径为全局最优解”这一隐式约束下进行搜索,使得整个逆优化过程构成一个典型的非线性逆优化问题。

我们在分别具有 7 个、15 个和 25 个连接点的图上进行了 30 组不同的独立实验,每组实验迭代 1000 次。实验结果列于表一中。我们计算了不同条件下最优权重与初始权重之间的欧氏距离。为了进行比较,我们还计算了传统编程方法的初始权重与最优权重之间的欧氏距离。传统方法指基于 KKT 条件与对偶理论的经典逆最短路径算法,该方法通过显式构造最优性条件并求解相应线性规划模型获得参数解。

需要指出的是,在逆优化问题中,底层优化问题的全局最优解通常无法显式获得,因此传统意义下的最优性间隙难以直接计算。为此,本文采用基于搜索的最优性违反度作为替代指标:通过在当前参数下求解底层优化问题,比较给定解与搜索到的最优解之间的目标函数差值,以评估给定解的最优性满足程度。在逆最短路径问题中,该指标可通过计算给定路径长度与当前参数下最短路径长度之间的差值获得(表 1, 图 3)。

Table 1. Performance of the algorithm under different node numbers and feasible initial paths

表 1. 算法在不同节点数量和可行初始路径下的性能

点的个数	最小值	平均值	传统方法	最优性违反度
7	0.347	0.359	2.236	0.031
15	1.716	1.733	4.472	0.075
25	2.978	3.162	5.809	0.115

结果表明,所提出的算法在逆最短路优化应用中表现良好。与经典图论算法相比,该算法在欧氏空间中能够实现更低的成本。这意味着所提出的算法完全能够胜任逆最短路优化问题。我们可以得出结论,该算法在逆最短路优化应用中具有良好的潜力。

5. 结论

本文围绕非线性逆优化问题的建模与求解展开深入研究,系统梳理了逆优化问题的定义、应用场景

及现有方法的局限性, 明确其核心难点在于约束条件缺乏显式表达式、参数可行性判定困难。为此, 提出的嵌套进化逆优化算法创新性地融合了双种群进化、邻域协同搜索、动态种群筛选等机制, 有效解决了传统方法在非线性场景下的求解瓶颈。

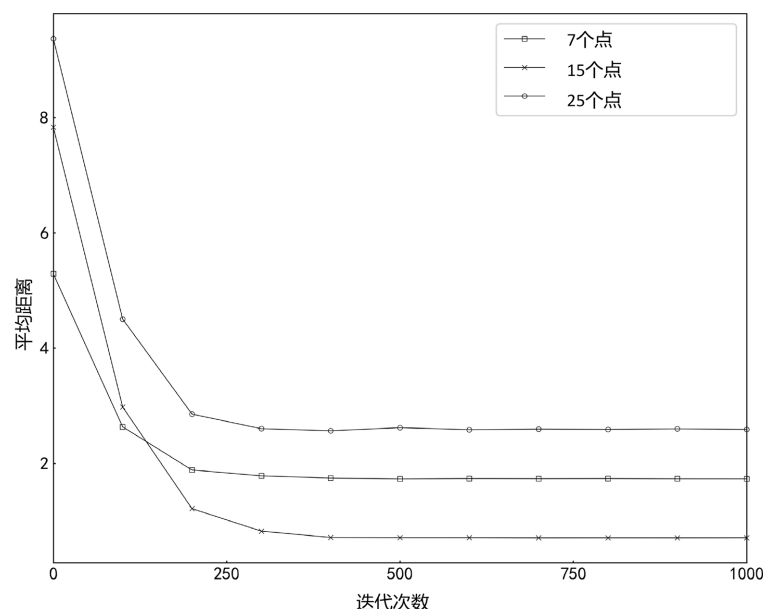


Figure 3. Performance of our proposed algorithm in inverse shortest path problem with different number of nodes

图 3. 算法在不同节点数量的反向最短路径问题中的性能

算法通过参数种群与解种群的协同进化, 利用相似参数向量对应子问题最优解相近的特性, 缩短了搜索时间; 保护集的设置保障了种群多样性, 避免局部收敛, 待更新集结合非支配排序则加速了优化目标的逼近。在逆最短路径问题的验证中, 算法在不同规模节点图上均表现出更优的性能, 相较于传统方法能以更小的参数调整成本实现预定路径的最优性, 充分证明了算法的有效性与实用性。

该研究不仅丰富了非线性逆优化问题的求解方法体系, 为进化算法在逆优化领域的应用提供了新思路, 其提出的算法框架还可通过替换更高效的搜索算子进一步拓展性能, 在更多复杂工程与科学计算场景中具有广阔的应用前景。

参考文献

- [1] Li, H. (2014) An Evolutionary Algorithm for Multi-Criteria Inverse Optimal Value Problems Using a Bilevel Optimization Model. *Applied Soft Computing*, **23**, 308-318. <https://doi.org/10.1016/j.asoc.2014.06.044>
- [2] Tarantola, A. (2005) Inverse Problem Theory and Methods for Model Parameter Estimation. Society for Industrial and Applied Mathematics. <https://doi.org/10.1137/1.9780898717921>
- [3] Ahuja, R.K. and Orlin, J.B. (2001) Inverse Optimization. *Operations Research*, **49**, 771-783. <https://doi.org/10.1287/opre.49.5.771.10607>
- [4] Troutt, M.D., Pang, W. and Hou, S. (2006) Behavioral Estimation of Mathematical Programming Objective Function Coefficients. *Management Science*, **52**, 422-434. <https://doi.org/10.1287/mnsc.1050.0445>
- [5] Ayer, T. (2014) Inverse Optimization for Assessing Emerging Technologies in Breast Cancer Screening. *Annals of Operations Research*, **230**, 57-85. <https://doi.org/10.1007/s10479-013-1520-3>
- [6] Birge, J.R., Hortaçsu, A. and Pavlin, J.M. (2017) Inverse Optimization for the Recovery of Market Structure from Market Outcomes: An Application to the MISO Electricity Market. *Operations Research*, **65**, 837-855.

- <https://doi.org/10.1287/opre.2017.1606>
- [7] Bertsimas, D., Gupta, V. and Paschalidis, I.C. (2012) Inverse Optimization: A New Perspective on the Black-Litterman Model. *Operations Research*, **60**, 1389-1403. <https://doi.org/10.1287/opre.1120.1115>
 - [8] Burton, D. and Toint, P.L. (1994) On the Use of an Inverse Shortest Paths Algorithm for Recovering Linearly Correlated Costs. *Mathematical Programming*, **63**, 1-22. <https://doi.org/10.1007/bf01582056>
 - [9] We, C. (2013) An Optimized Service Routing Allocation Method for Electric Power Communication Network Considering Reliability. *Power System Technology*, **37**, 3541-3545.
 - [10] Qiu, J., Lu, X., Wang, X. and Hu, X. (2021) Research on Rice Disease Identification Model Based on Migration Learning in VGG Network. *IOP Conference Series: Earth and Environmental Science*, **680**, Article ID: 012087. <https://doi.org/10.1088/1755-1315/680/1/012087>
 - [11] Wang, Y., Ma, F., Jin, Z., Yuan, Y., Xun, G., Jha, K., et al. (2018) EANN: Event Adversarial Neural Networks for Multi-Modal Fake News Detection. *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, London, 19-23 August 2018, 849-857. <https://doi.org/10.1145/3219819.3219903>
 - [12] Adil, M., Song, H., Ali, J., Jan, M.A., Attique, M., Abbas, S., et al. (2022) Enhanced-AODV: A Robust Three Phase Priority-Based Traffic Load Balancing Scheme for Internet of Things. *IEEE Internet of Things Journal*, **9**, 14426-14437. <https://doi.org/10.1109/jiot.2021.3072984>
 - [13] Mamashli, Z., Bozorgi-Amiri, A., Dadashpour, I., Nayeri, S. and Heydari, J. (2021) A Heuristic-Based Multi-Choice Goal Programming for the Stochastic Sustainable-Resilient Routing-Allocation Problem in Relief Logistics. *Neural Computing and Applications*, **33**, 14283-14309. <https://doi.org/10.1007/s00521-021-06074-8>
 - [14] Yang, L., Jing, L., Yu, J. and Ng, M.K. (2016) Learning Transferred Weights from Co-Occurrence Data for Heterogeneous Transfer Learning. *IEEE Transactions on Neural Networks and Learning Systems*, **27**, 2187-2200. <https://doi.org/10.1109/tnnls.2015.2472457>
 - [15] Guo, W., Wen, Z., Guo, Z., Xu, C., Xiao, Q. and Mao, L. (2020) Varying Balancing Transfer Learning for BN Parameter Estimation. *2020 Chinese Control and Decision Conference (CCDC)*, Hefei, 22-24 August 2020, 2179-2184. <https://doi.org/10.1109/ccdc49329.2020.9164198>
 - [16] Ahuja, R.K. and Orlin, J.B. (2002) Combinatorial Algorithms for Inverse Network Flow Problems. *Networks*, **40**, 181-187. <https://doi.org/10.1002/net.10048>
 - [17] Schaefer, A.J. (2009) Inverse Integer Programming. *Optimization Letters*, **3**, 483-489. <https://doi.org/10.1007/s11590-009-0131-z>
 - [18] Lamperski, J.B. and Schaefer, A.J. (2015) A Polyhedral Characterization of the Inverse-Feasible Region of a Mixed-Integer Program. *Operations Research Letters*, **43**, 575-578. <https://doi.org/10.1016/j.orl.2015.08.010>
 - [19] Thai, J. and Bayen, A.M. (2018) Imputing a Variational Inequality Function or a Convex Objective Function: A Robust Approach. *Journal of Mathematical Analysis and Applications*, **457**, 1675-1695. <https://doi.org/10.1016/j.jmaa.2016.09.031>
 - [20] Tayyebi, J. and Aman, M. (2018) On Inverse Linear Programming Problems under the Bottleneck-Type Weighted Hamming Distance. *Discrete Applied Mathematics*, **240**, 92-101. <https://doi.org/10.1016/j.dam.2015.12.017>
 - [21] Ghatge, A. (2015) Inverse Optimization in Countably Infinite Linear Programs. *Operations Research Letters*, **43**, 231-235. <https://doi.org/10.1016/j.orl.2015.02.004>
 - [22] Shahmoradi, Z. and Lee, T. (2022) Quantile Inverse Optimization: Improving Stability in Inverse Linear Programming. *Operations Research*, **70**, 2538-2562. <https://doi.org/10.1287/opre.2021.2143>
 - [23] Naghavi, M., Foroughi, A.A. and Zarepisheh, M. (2019) Inverse Optimization for Multi-Objective Linear Programming. *Optimization Letters*, **13**, 281-294. <https://doi.org/10.1007/s11590-019-01394-0>
 - [24] Jiang, H. and Ralph, D. (2000) Smooth SQP Methods for Mathematical Programs with Nonlinear Complementarity Constraints. *SIAM Journal on Optimization*, **10**, 779-808. <https://doi.org/10.1137/s1052623497332329>
 - [25] Wang, L. (2009) Cutting Plane Algorithms for the Inverse Mixed Integer Linear Programming Problem. *Operations Research Letters*, **37**, 114-116. <https://doi.org/10.1016/j.orl.2008.12.001>
 - [26] Zhang, Y., Jiang, Y., Zhang, L. and Zhang, J. (2013) A Perturbation Approach for an Inverse Linear Second-Order Cone Programming. *Journal of Industrial & Management Optimization*, **9**, 171-189. <https://doi.org/10.3934/jimo.2013.9.171>
 - [27] Lv, Y., Chen, Z. and Wan, Z. (2010) A Penalty Function Method Based on Bilevel Programming for Solving Inverse Optimal Value Problems. *Applied Mathematics Letters*, **23**, 170-175. <https://doi.org/10.1016/j.aml.2009.09.007>
 - [28] Chen, X. and Fukushima, M. (2004) A Smoothing Method for a Mathematical Program with P-Matrix Linear Complementarity Constraints. *Computational Optimization and Applications*, **27**, 223-246. <https://doi.org/10.1023/b:coap.0000013057.54647.6d>
 - [29] Wu, H., Wang, K., Guo, Q., Xu, G. and Li, N. (2012) Design of a Kind of Nonlinear Neural Networks for Solving the

- Inverse Optimal Value Problem with Convex Constraints. *International Journal of Machine Learning and Cybernetics*, **5**, 85-92. <https://doi.org/10.1007/s13042-012-0138-0>
- [30] Kamyab, S., Azimifar, Z., Sabzi, R. and Fieguth, P. (2022) Deep Learning Methods for Inverse Problems. *PeerJ Computer Science*, **8**, e951. <https://doi.org/10.7717/peerj-cs.951>
- [31] Kohli, M. and Arora, S. (2017) Chaotic Grey Wolf Optimization Algorithm for Constrained Optimization Problems. *Journal of Computational Design and Engineering*, **5**, 458-472. <https://doi.org/10.1016/j.jcde.2017.02.005>
- [32] Deb, K. and Sinha, A. (2010) An Efficient and Accurate Solution Methodology for Bilevel Multi-Objective Programming Problems Using a Hybrid Evolutionary-Local-Search Algorithm. *Evolutionary Computation*, **18**, 403-449. https://doi.org/10.1162/evco_a_00015
- [33] Ang, K.M., Lim, W.H., Isa, N.A.M., Tiang, S.S. and Wong, C.H. (2020) A Constrained Multi-Swarm Particle Swarm Optimization without Velocity for Constrained Optimization Problems. *Expert Systems with Applications*, **140**, Article ID: 112882. <https://doi.org/10.1016/j.eswa.2019.112882>
- [34] Tian, Y., Zhang, T., Xiao, J., Zhang, X. and Jin, Y. (2021) A Coevolutionary Framework for Constrained Multiobjective Optimization Problems. *IEEE Transactions on Evolutionary Computation*, **25**, 102-116. <https://doi.org/10.1109/tevc.2020.3004012>
- [35] Chaabani, A., Bechikh, S. and Ben Said, L. (2016) A Memetic Evolutionary Algorithm for Bi-Level Combinatorial Optimization: A Realization between Bi-MDVRP and Bi-CVRP. 2016 *IEEE Congress on Evolutionary Computation (CEC)*, Vancouver, 24-29 July 2016, 1666-1673. <https://doi.org/10.1109/cec.2016.7743988>
- [36] Angelo, J.S., Krempser, E. and Barbosa, H.J.C. (2013) Differential Evolution for Bilevel Programming. 2013 *IEEE Congress on Evolutionary Computation*, Cancun, 20-23 June 2013, 470-477. <https://doi.org/10.1109/cec.2013.6557606>
- [37] Zhang, Q.F. and Li, H. (2007) MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. *IEEE Transactions on Evolutionary Computation*, **11**, 712-731. <https://doi.org/10.1109/tevc.2007.892759>