

基于支持向量机的改进遗传算法在模糊最短路问题中的应用

孙亚莉, 陈学刚*, 陈甜甜

华北电力大学数理学院, 北京

收稿日期: 2021年8月12日; 录用日期: 2021年10月7日; 发布日期: 2021年10月14日

摘 要

最短路问题是图论中重要的优化问题之一。随着网络的增大, 求最短路径的时间复杂度越来越大。本文基于模糊图中部分路径的长度, 设计了基于支持向量机的改进遗传算法, 并用该算法求得了模糊最短路问题的较好近似解。最后通过数值算例模拟该算法, 并与标准的遗传算法作对比, 结果表明该算法是有效的。

关键词

遗传算法, 模糊最短路问题, 支持向量机, 适应度值

The Application of Improved Genetic Algorithm Based on Support Vector Machine in Fuzzy Shortest Path Problem

Yali Sun, Xuegang Chen*, Tiantian Chen

School of Mathematics and Physics, North China Electric Power University, Beijing

Received: Aug. 12th, 2021; accepted: Oct. 7th, 2021; published: Oct. 14th, 2021

Abstract

The shortest path problem is one of the important optimization problems in graph theory. As the network grows, the time complexity of finding the shortest path is getting bigger and bigger. Based on the length of some paths in the fuzzy graph, this paper designs an improved genetic algorithm

*通讯作者。

文章引用: 孙亚莉, 陈学刚, 陈甜甜. 基于支持向量机的改进遗传算法在模糊最短路问题中的应用[J]. 计算机科学与应用, 2021, 11(10): 2496-2505. DOI: 10.12677/csa.2021.1110253

based on support vector machines, and uses this algorithm to obtain a better approximate solution to the fuzzy shortest path problem. Finally, the algorithm is simulated through a numerical calculation example and compared with the standard genetic algorithm. The result shows that the algorithm is effective.

Keywords

Genetic Algorithm, Fuzzy Shortest Path Problem, Support Vector Machine, Fitness Value

Copyright © 2021 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

最短路问题是图论中经典的优化问题之一。在经典最短路问题中，两点之间的边的权重是确定的数，一般通过边的权重找到最短路径。在最短路问题的应用中，边可以表示距离、时间、成本等其他变量，而这些涉及到的变量一般都是不确定的，因而用模糊数表示边的权重不失为一种有效方法。过去的 20 年里，一些学者已经对不同类型的模糊数作为边的权重的最短路问题进行了大量的研究。对于由有限个顶点和有向边所构成的网络，边的权重都可表示为模糊数，则路的长度为组成路的所有边权的和，找到长度最短的路即为模糊最短路问题。

Dubois 和 Prade [1] 首先对模糊最短路问题进行了研究，其算法推广了经典的 Floyd 和 Ford-Moore-Bellman 算法。Chanas 和 Kamburowski [2] 引入模糊偏好关系的概念，并将其应用于求最短路问题的算法中。Takahashi 和 Yamakami [3] 提出了有模糊边长的最短路问题，扩展了 Okada [4] 提出的方法，并介绍了一种确定大型模糊图近似最短路径的遗传算法。Nayeem 和 Pal [5] 用三角模糊数和区间数表示边的权重，其提出的算法可以解决模糊最短路问题。Deng [6] 等人扩展了经典 Dijkstra 算法去求在不确定环境中的最短路问题。他们使用梯形模糊数表示模糊图边的权重，同时在算法中考虑了模糊数的分级平均值来确定模糊路径的长度，这也有助于比较两个不同模糊路径之间的模糊路径长度。Hassanzadeh [7] 等人使用基于 α 截集的法加运算来确定路径的长度，针对大规模模糊图中模糊数相加的复杂性，提出了一种求解模糊最短路问题的遗传算法。Lin [8] 等人使用梯形模糊数的分级平均值来比较模糊路径的长度，改变遗传算法的交叉过程来确定具有模糊边长的模糊图的源节点和目标节点之间的最短路径。关于最短路问题的研究可参阅文献[9] [10] [11]。

在模糊环境下，许多学者已经做了大量的工作，给出了求解最短路问题的精确算法、启发式算法和元启发式算法。精确算法能获得图的起始节点到目标节点的精确最短路径。然而当图的规模较大时，精确算法需要大量的时间来获得最短路径。元启发式算法则能够有效地解决最短路问题，其中遗传算法是求解最短路问题的一种流行而有效的算法。在该算法中，染色体提供了决策问题的潜在解决方案，一组不同的染色体产生种群，种群中的染色体进行交叉和变异来产生新的染色体，后通过选择将适应度值较大的染色体淘汰，从而得到决策问题的最优解。

在现实生活中，由于道路状况、交通负荷、事故和变化的天气条件等，决策者很难决定任何路线的准确成本，所以用模糊数作为边的权重。随着网络的增大，求最短路径的时间复杂度越来越大。由于时间和经费的限制，只能对模糊图中从起始节点到目标节点的部分路径的总长度进行估算。为了求模糊图从起始节点到目标节点的最短路，需要分析已知路径的长度，以预测未知路径的长度。因此，针对上述

问题, 本文提出了基于支持向量机的改进遗传算法。首先把已知的路径作为输入数据, 路径的长度作为输出数据训练支持向量机, 通过训练后的支持向量机来预测未知路径的长度。其次把路径作为遗传算法中的染色体, 把支持向量机预测的路径长度作为染色体适应度值, 通过遗传算法得到最短路径及其长度。最后通过数值算例分析表明算法的有效性。

2. 预备知识

2.1. 基本概念

定义 1 [8] 称模糊数 $\tilde{U} = \{\underline{u}, u_1, u_2, \bar{u}\}$ 为梯形模糊数, 如果其隶属度函数可以表示为:

$$\mu_{\tilde{U}}(y) = \begin{cases} \frac{y - u_1}{u_1 - \underline{u}} & \text{如果 } u_1 \geq y \geq \underline{u} \\ 1 & \text{如果 } u_1 \geq y \geq u_2 \\ \frac{\bar{u} - y}{\bar{u} - u_2} & \text{如果 } \bar{u} \geq y \geq u_2 \\ 0 & \text{如果 } y < \underline{u} \text{ 或 } y > \bar{u} \end{cases} \quad (1)$$

其中 u_1 和 u_2 是两个主要的可能值, $\underline{u}$ 和 $\bar{u}$ 分别是最小可能值和最大可能值。

定义 2 [8] 设 $\tilde{U} = \{\underline{u}, u_1, u_2, \bar{u}\}$ 为梯形模糊数, 则 $\tilde{U}$ 的分级平均值可定义为:

$$P(\tilde{U}) = \frac{1}{6}(\underline{u} + 2u_1 + 2u_2 + \bar{u}) \quad (2)$$

定义 3 [8] 设 $\tilde{U} = \{\underline{u}, u_1, u_2, \bar{u}\}$ 和 $\tilde{V} = \{\underline{v}, v_1, v_2, \bar{v}\}$ 为梯形模糊数, 则

- 1) 如果 $P(\tilde{U}) = P(\tilde{V})$, 那么 $\tilde{U} = \tilde{V}$;
- 2) 如果 $P(\tilde{U}) > P(\tilde{V})$, 那么 $\tilde{U} > \tilde{V}$;
- 3) 如果 $P(\tilde{U}) < P(\tilde{V})$, 那么 $\tilde{U} < \tilde{V}$ 。

定义 4 [8] 设 $\tilde{U} = \{\underline{u}, u_1, u_2, \bar{u}\}$ 和 $\tilde{V} = \{\underline{v}, v_1, v_2, \bar{v}\}$ 为梯形模糊数, 则 $\tilde{U}$ 和 $\tilde{V}$ 的加法运算定义为:

$$P(\tilde{U} \oplus \tilde{V}) = \frac{1}{6}(\underline{u} + 2u_1 + 2u_2 + \bar{u}) + \frac{1}{6}(\underline{v} + 2v_1 + 2v_2 + \bar{v}) \quad (3)$$

2.2. 支持向量机

支持向量机是一种泛化能力强、精度高的回归算法, 适用于解决小样本、非线性、高维数据等问题, 其基本思想是在特征空间中构建最优超平面, 使得学习器得到全局最优解。

设训练样本集为 $T = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$, 其中 x_i 为样本特征, y_i 为样本属性值。根据文献 [12], 构造损失函数并采用以下极小化优化模型来确定回归函数, 即

$$\begin{aligned} \min \quad & \frac{1}{2} \|\omega\|^2 + C \sum_{i=1}^n (\xi_i^* + \xi_i) \\ \text{s.t.} \quad & \begin{cases} y_i - \langle \omega, \phi(x_i) \rangle - b \leq \varepsilon + \xi_i^* \\ \langle \omega, \phi(x_i) \rangle + b - y_i \leq \varepsilon + \xi_i \\ \xi_i^*, \xi_i \geq 0 \quad (i=1, 2, \dots, n) \end{cases} \end{aligned} \quad (4)$$

式中: ω 为权值; $\frac{1}{2} \|\omega\|^2$ 控制模型的泛化能力; C 为惩罚因子; ξ_i^*, ξ_i 为松弛因子; $\phi(x)$ 是将数据映射到

高维空间的非线性变换; b 为偏置; ε 为误差上限。

利用拉格朗日乘子法求解该模型可得到支持向量机回归函数:

$$f(x) = \sum_{i=1}^n (\alpha_i - \alpha_i^*) k(x_i, x_j) + b \quad (5)$$

其中: α_i 和 α_i^* 为拉格朗日乘子, $k(x_i, x_j) = \phi(x_i)\phi(x_j)$ 为核函数, 需满足 Mercer 条件, 本文选取高斯核函数, 为

$$k(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right) = \exp(-g\|x_i - x_j\|^2), g > 0 \quad (6)$$

可见, 在支持向量机计算过程中涉及两个参数, 即, 惩罚因子 C 和核函数参数 g , 这两个参数的位置对于支持向量机回归模型有较大影响, 因此, 本文采用粒子群算法对参数进行优化[13]。

3. 模糊最短路问题

给定一个有向赋权的模糊图 $D=(V, E)$, 设节点集为 $V=\{v_1, v_2, \dots, v_p\}$, 边集为 $E \subseteq V \times V$ 。每条边用节点的有序对 (v_i, v_j) 来表示, 且 $v_i \neq v_j$, 并假设从节点 v_i 到 v_j 至多只有一条有向边。路径的起始节点为 u , 目标节点为 v 。考虑与边 (v_i, v_j) 有关的非负梯形模糊数 $\tilde{c}_{v_i v_j}$, $\tilde{c}_{v_i v_j}$ 表示从节点 v_i 到 v_j 的模糊费用。

给定任意一条从源节点到目标节点的路径, 对所有的边 (v_i, v_j) 赋值, 如若边 (v_i, v_j) 在该路径中, 则 $x_{v_i v_j} = 1$, 否则 $x_{v_i v_j} = 0$ 。该模糊最短路问题的线性规划模型如下:

$$\begin{aligned} \min \quad & \sum_{v_i \in V} \sum_{v_j \in V} \tilde{c}_{v_i v_j} x_{v_i v_j} \\ \text{s.t.} \quad & \sum_{v_j \in V} x_{v_i v_j} - \sum_{v_j \in V} x_{v_j v_i} = \begin{cases} 1 & v_i = u \\ 0 & v_i \neq u, v \\ -1 & v_i = v \end{cases} \\ & x_{v_i v_j} \in \{0, 1\} \\ & \forall v_i \in V \end{aligned} \quad (7)$$

4. 标准遗传算法

遗传算法是模拟自然界生物进化现象发展起来的随机全局搜索优化方法, 主要包括以下步骤:

1) 染色体编码

创建由染色体组成的大小为 N 的初始种群。一条从起始节点到目标节点的路径代表一个染色体。

2) 计算适应度值

染色体的适应度值是区分种群中染色体好坏的标准。在最短路问题中, 目标是确定模糊图中指定的起始节点 u 和目标节点 v 之间的最短路径。因此染色体的适应度值可以用模糊路径的长度来表示。一条从 u 到 v 的路径, 使用梯形模糊数的加法运算计算该模糊路径的梯形模糊数, 后使用梯形模糊数的分级平均值来计算路径的长度, 即该染色体的适应度值。

3) 选择、交叉和变异的方法

染色体通过执行选择、交叉和变异产生新的染色体, 形成新的种群。

① 选择: 选择是从旧种群中选择优质个体, 淘汰部分个体, 产生新种群的过程。一般采用轮盘赌选

择方法[14], 假设初始种群数量为 N , 种群中每个染色体的适应度值为 $f_i (i=1, 2, \dots, N)$, $g_i = \frac{1}{f_i}$, 每个染色体的选择概率为 $p_i = \frac{f_i}{\sum_{k=1}^N f_k} (i=1, 2, \dots, N)$, 累积概率为 $Q_i = \sum_{k=1}^i p_k (i=1, 2, \dots, N)$ 。由上述方法可知,

每个染色体的适应度值越小, 被选择的可能性越大。

② 交叉: 在选定的父染色体上执行交叉, 可以生成新的染色体, 即子染色体。交叉概率表示执行交叉操作的频率, 是执行交叉操作的主要参数。一般采取单点交叉[14], 同时为了避免交叉后产生不可行解, 将两条染色体之间的关系分为以下两种情况:

情况 1: 两条染色体中间存在公共边;

情况 2: 两条染色体除了前后有公共边之外, 无任何其它公共边。

交叉操作的思路是根据交叉概率选择是否执行交叉, 若交叉则随机选择两条染色体, 从两条染色体的起始边开始对比, 寻找是否存在相同边, 若存在相同边之后有不同边存在, 则将从不同边开始到目标节点的所有边进行交换。

情况 1: 从边 w 之后染色体 1 与染色体 2 进行交叉, 交叉前后分别为:

染色体 1: $1 \rightarrow 2 \rightarrow 4 \rightarrow w \rightarrow 8 \rightarrow 10 \rightarrow 14 \rightarrow 17$ 染色体 1': $1 \rightarrow 2 \rightarrow 4 \rightarrow w \rightarrow 7 \rightarrow 9 \rightarrow 11 \rightarrow 13 \rightarrow 17$

染色体 2: $1 \rightarrow 5 \rightarrow w \rightarrow 7 \rightarrow 9 \rightarrow 11 \rightarrow 13 \rightarrow 17$ 染色体 2': $1 \rightarrow 5 \rightarrow w \rightarrow 8 \rightarrow 10 \rightarrow 14 \rightarrow 17$

情况 2: 此时若进行不同边的交叉, 即染色体 1 变为了染色体 2, 染色体 2 变为了染色体 1, 无任何意义。因此采用如下方法: 寻找染色体 1 与染色体 2 中有双向边相连的边, 即: 染色体 1 中某边的邻接边同时也是染色体 2 中某边的邻接边, 此时, 引入边 w 对染色体 1 与染色体 2 进行交叉。交叉前后分别为:

染色体 1: $1 \rightarrow 2 \rightarrow 4 \rightarrow 8 \rightarrow 10 \rightarrow 14 \rightarrow 17$ 染色体 1': $1 \rightarrow 2 \rightarrow 4 \rightarrow w \rightarrow 7 \rightarrow 9 \rightarrow 11 \rightarrow 13 \rightarrow 17$

染色体 2: $1 \rightarrow 5 \rightarrow 7 \rightarrow 9 \rightarrow 11 \rightarrow 13 \rightarrow 17$ 染色体 2': $1 \rightarrow 5 \rightarrow w \rightarrow 8 \rightarrow 10 \rightarrow 14 \rightarrow 17$

若选择的两条染色体不满足情况 1, 2, 则重新选择两条染色体进行交叉。

③ 变异: 选择一条的染色体, 以指定的变异概率执行变异。一般采用的变异策略[14]为: 根据变异概率选择是否执行变异, 若变异则随机选择一条染色体进行变异, 然后随机选择要进行变异的边 w , 将染色体从该边处断开, 按照如下方法产生边 w 的前一条边 a 和后一条边 b 之间的随机路径:

步骤 1 若边 a 只有 w 一条邻接边, 则重新选择变异位置。否则, 执行步骤 2;

步骤 2 判断边 a 的邻接边中有没有终点为边 b 起点的边(除去边 w), 如有则用此边替换边 w , 终止程序。若没有, 执行步骤 3;

步骤 3 选择边 a 的所有邻接边中终点与边 b 的起点距离最短的边 c , 将其替换掉边 w , 并执行步骤 2;

步骤 4 若随机路径中插入了 5 条边, 仍没有连接到边 b , 则重新选择变异位置;

步骤 5 将产生的随机路径插入到原路径的变异位置。

4) 更新

所有新生成的染色体放入当前种群中生成新的总体, 用于进一步的计算。

5) 终止条件

判断该算法是否达到最大迭代次数, 若达到, 终止算法并返回该种群中适应度低的染色体, 该染色体为决策问题的最优解; 否则, 转到步骤(2)继续执行直到满足条件。

5. 基于支持向量机的改进遗传算法

本文提出了基于支持向量机的改进遗传算法, 其基本步骤如下:

1) 染色体编码

在算法中, 每个染色体都描述了模糊图的起始节点和目标节点之间的路径。模糊图中的每个节点都被赋予唯一的整数。在模糊最短路问题中, 染色体的第一个节点总是起始节点。为了创建编码染色体所需的路径, 首先由起始节点生成路径的下一个节点。为了解决这个问题, 从起始节点的相邻节点列表中随机选择一个节点, 例如 w 。如果所选节点 w 不存在于染色体中, 则节点 w 被用作染色体中起始节点的下一个节点, 再由 w 作为起始节点, 执行上述方法直到到达染色体的目标节点。循环上述方法直到到达染色体的初始种群数目。

2) 基于支持向量机计算染色体适应度值

染色体的适应度值是区分种群中染色体好坏的标准。给定模糊图及其边的权重, 随机生成 N 条从起始节点到目标节点的路径作为样本集, 通过梯形模糊数的加法及分级平均值的定义计算路径的长度, 将其作为样本的属性值, 将生成的 N 个样本作为训练集。采用基于粒子群算法优化支持向量机, 从而得到支持向量机回归函数。在算法中, 对任意给定的一条从起始节点到目标节点的路径, 采用支持向量机回归函数预测得到样本属性值并将其作为染色体的适应度值。

3) 选择、交叉和变异的方法

① 选择: 选择方法用于选择更多数量的具有较低适应度值的染色体, 随机选取的方式, 保证染色体选择的偶然性和随机性。该算法采用二元锦标赛选择技术, 从当前种群中随机选择两个染色体, 根据每个个体的适应度值, 选择其中适应度值低的个体进入下一代种群。多次重复上述过程, 直到新的种群规模达到原来的种群规模。

② 交叉: 采用交叉方法重组两条父染色体, 生成新的子染色体。在该算法中, 使用了以下交叉方法:

步骤 1 从当前种群中随机选择两个父染色体 F_1 和 F_2 。

步骤 2 从 F_1 中随机选取一段基因(起始节点和目标节点除外)。

步骤 3 使用获取的该段基因生成子染色体, 该段基因在子染色体中的位置与在 F_1 中的位置相同, 子染色体的起始节点和目标节点与 F_1 相同, 且子染色体的长度与 F_1 相同。

步骤 4 从左到右依次选取 F_2 中的节点(起始节点和目标节点除外), 如果 F_2 中的节点没有出现在子染色体中, 那么将该节点放入到子染色体中第一个空缺位置中, 否则重新选取 F_2 中的节点。重复上述选取方法直到将子染色体的所有空缺位置填满。

若通过上述方法无法将子染色体的空位填满, 则重新选择两条染色体进行交叉。

③ 变异: 染色体以指定的变异概率执行变异操作。该算法先随机选取一条染色体, 后随机指定染色体上的某一节点进行变异。

4) 更新染色体

为了减少运行时间, 本算法采用上述交叉和变异方法。但上述交叉和变异方法产生的个体不一定为可行解, 为了增加可行解的个数, 此算法要尽可能增大交叉和变异的次数。因此需要判断交叉和变异的次数是否等于 M , 若等于则停止交叉变异; 否则, 继续交叉变异。

通过上述交叉和变异方法会生成的新的个体, 该算法需要判断新生成的个体是否为染色体。若新生成的个体为染色体, 则该个体是一条从起始节点 u 到目标节点 v 的路径。首先假设新生成的个体的起始节点 u 的下一个节点为 w , 判断节点 w 是否为起始节点 u 的相邻节点, 若节点 w 不为起始节点 u 的相邻节点, 则新生成的个体不为染色体, 若节点 w 为起始节点 u 的相邻节点, 则从节点 w 作为起始节点并执行上述方法直到到达新生成个体的目标节点 v 。将所有新生成的染色体放入当前种群中生成新的种群, 用于进一步的计算。

5) 终止条件

判断该算法是否达到最大迭代次数, 若达到, 终止算法并返回该种群中具有最小适应度值的染色体,

该染色体为模糊图中从起始节点到目标节点的最短路；否则，转到步骤(2)继续执行直到满足条件。
算法的流程图如图 1 所示。

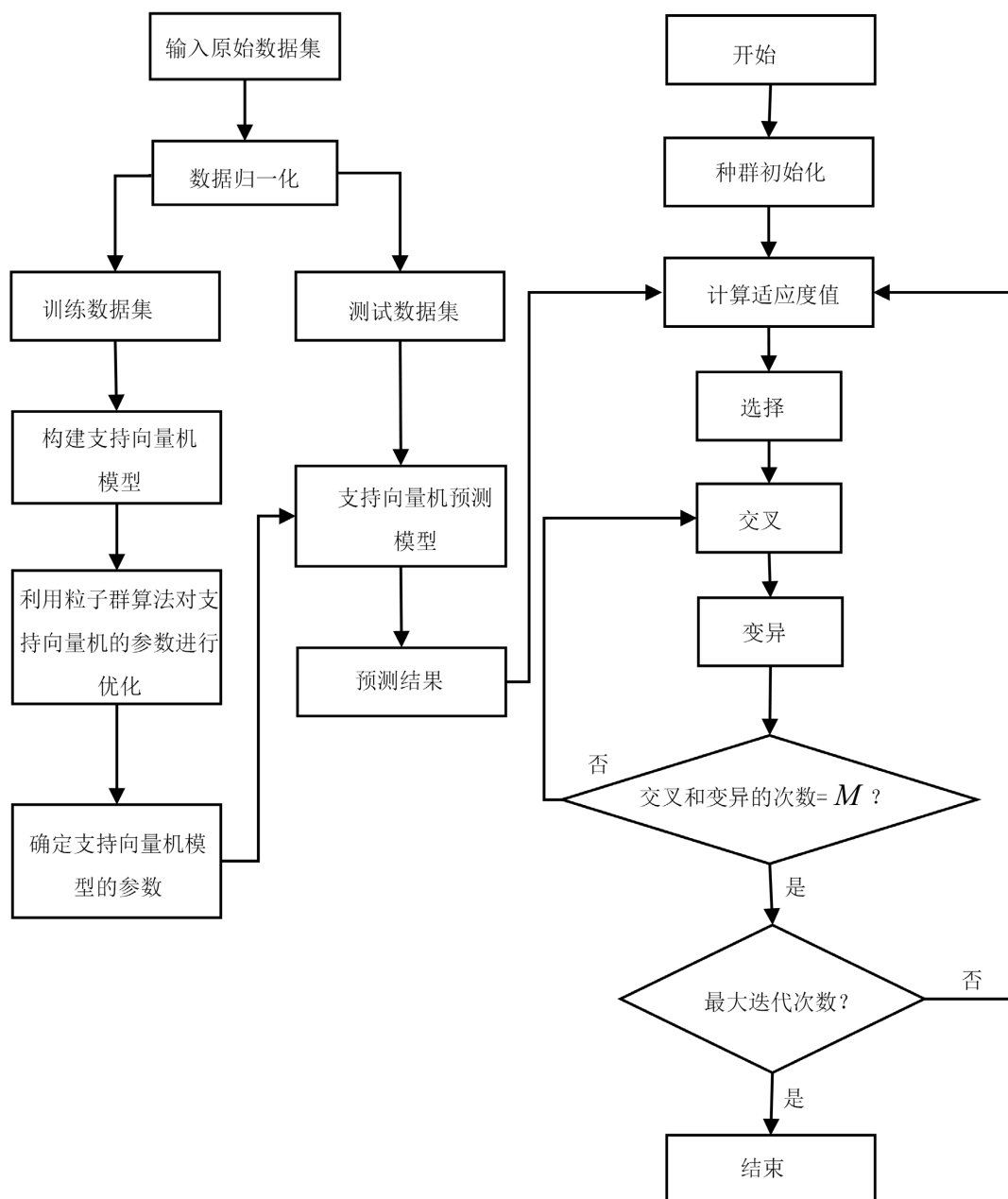


Figure 1. Flow chart of improved genetic algorithm based on support vector machine
图 1. 基于支持向量机的改进遗传算法流程图

6. 数值算例分析

考虑图 2 所示的交通网络模糊图，包含 24 个节点和 43 条边，在此模糊图上执行标准遗传算法与基于支持向量机的改进遗传算法。假设模糊图的起始节点为 1，目标节点为 24。表 1 给出了模糊图边的权重，表示从一个节点到另一点节点的模糊成本。

算法采用 MATLAB R2019b 软件，在 Intel(R) Core(TM) i7-10510U CPU @ 1.80GHz 2.30 GHz，内存 8.00 GB，Window10 平台上编码实现。本文对两种遗传算法的参数设置均相同，并取迭代次数为 20 次，种群规模为 24，每一代交叉和变异的次数为 100 次，交叉概率与变异概率分别为 0.7 与 0.6。

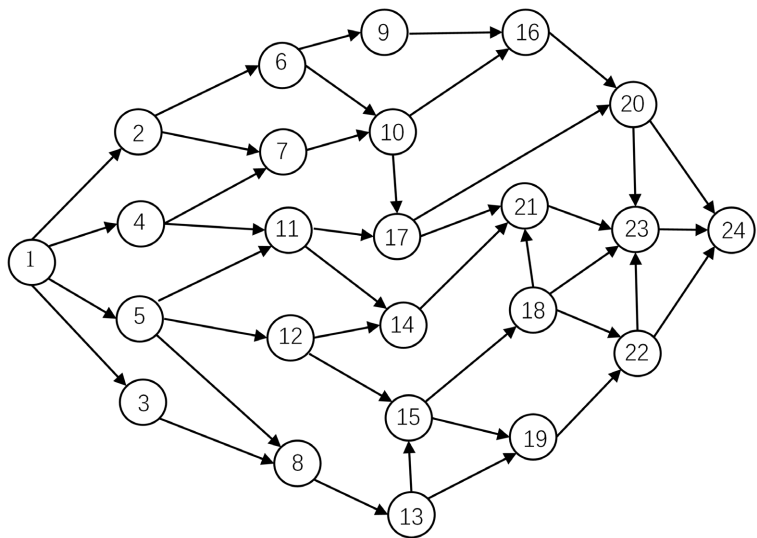


Figure 2. Fuzzy graph of traffic network
图 2. 交通网络模糊图

Table 1. The length of fuzzy graph
表 1. 模糊图的弧长

边	模糊成本	边	模糊成本
(1, 2)	(8, 9, 10, 12)	(1, 3)	(13, 15, 16, 17)
(1, 4)	(1, 2, 3, 7)	(2, 7)	(13, 15, 16, 17)
(1, 5)	(7, 9, 10, 12)	(2, 6)	(5, 8, 9, 10)
(3, 8)	(8, 9, 10, 12)	(4, 7)	(3, 5, 7, 9)
(4, 11)	(2, 4, 5, 10)	(5, 8)	(5, 8, 9, 10)
(5, 11)	(8, 9, 10, 12)	(5, 12)	(13, 15, 16, 17)
(6, 9)	(32, 33, 35, 37)	(6, 10)	(13, 16, 20, 22)
(7, 10)	(9, 11, 12, 13)	(7, 11)	(2, 4, 5, 10)
(8, 12)	(7, 10, 12, 19)	(8, 13)	(5, 8, 9, 10)
(9, 16)	(13, 15, 16, 17)	(10, 16)	(2, 4, 5, 10)
(10, 17)	(22, 23, 24, 26)	(11, 14)	(1, 2, 3, 7)
(11, 17)	(3, 5, 7, 9)	(12, 14)	(5, 8, 9, 10)
(12, 15)	(10, 15, 16, 17)	(13, 15)	(10, 11, 13, 15)
(13, 19)	(13, 15, 16, 17)	(14, 21)	(5, 8, 9, 10)
(15, 18)	(9, 10, 12, 16)	(15, 19)	(9, 10, 12, 14)
(16, 20)	(1, 2, 3, 7)	(17, 20)	(9, 10, 13, 16)
(17, 21)	(1, 2, 3, 7)	(18, 21)	(1, 6, 9, 15)
(18, 22)	(18, 20, 21, 23)	(18, 23)	(21, 25, 26, 28)
(19, 22)	(13, 15, 16, 17)	(20, 23)	(10, 12, 15, 16)
(21, 23)	(12, 16, 18, 20)	(22, 23)	(7, 8, 9, 10)
(22, 24)	(6, 7, 8, 9)	(20, 24)	(3, 5, 7, 9)
(23, 24)	(5, 8, 8, 10)		

为验证本文所提算法的有效性，本文分别用标准的遗传算法和基于支持向量机的改进遗传算法进行 20 次实验，程序结果列于表 2。

Table 2. Program results of the two algorithms
表 2. 两种算法的程序结果

运行次数	标准的遗传算法		基于支持向量机的遗传算法	
	模糊最短路	最短路径长度	模糊最短路	最短路径长度
1	1→4→7→10→16→20→24	34.3333	1→4→7→10→16→20→24	34.3434
2	1→4→11→17→21→23→24	46.5001	1→4→7→10→16→20→24	34.3433
3	1→4→11→17→20→24	36.5000	1→4→11→17→20→24	36.5100
4	1→4→11→14→21→23→24	48.6668	1→4→7→10→16→20→24	34.3434
5	1→4→7→11→17→21→23→24	47.8334	1→5→11→17→20→24	43.0100
6	1→4→7→10→16→20→24	34.3333	1→4→7→10→16→20→24	34.3435
7	1→4→11→17→21→23→24	46.5001	1→5→11→17→20→24	40.1359
8	1→4→11→14→21→23→24	48.6668	1→5→11→17→20→24	42.9904
9	1→4→11→17→20→24	36.5000	1→4→7→10→16→20→24	34.3431
10	1→4→7→11→17→21→23→24	47.8334	1→4→7→10→17→20→24	39.3016
11	1→4→7→10→16→20→24	34.3333	1→4→11→17→20→24	47.1072
12	1→4→7→10→16→20→24	34.3333	1→4→7→10→16→20→24	34.3431
13	1→4→11→17→20→24	36.5000	1→4→11→17→21→23→24	46.5104
14	1→4→11→14→21→23→24	48.6668	1→4→11→17→20→24	36.5100
15	1→4→11→17→21→23→24	46.5001	1→4→7→10→16→20→24	34.3433
16	1→4→11→17→20→24	36.5000	1→4→7→10→16→20→24	34.3435
17	1→4→7→10→16→20→24	34.3333	1→5→11→17→20→24	40.2378
18	1→4→7→10→16→20→24	34.3333	1→4→11→17→21→23→24	46.5101
19	1→4→7→11→17→20→24	37.8333	1→4→7→10→16→20→24	34.3433
20	1→4→11→17→20→24	36.5000	1→4→11→17→20→24	36.5099

通过表 2 可知两种算法的对比结果列于表 3。

Table 3. Comparison results of the two algorithms
表 3. 两种算法的对比结果

算法	模糊最短路	模糊最短路平均长度	最短路出现次数
标准的遗传算法	1→4→7→10→16→20→24	34.3333	6
基于支持向量机的遗传算法	1→4→7→10→16→20→24	34.34332	9

由表 3 可知，使用标准的遗传算法求得的模糊最短路为：1→4→7→10→16→20→24，模糊最短路的平均长度为 34.3333；使用基于支持向量机的改进遗传算法求得的模糊最短路同样为：1→4→7→10→16→20→24，模糊最短路的平均长度为 34.34332，两种算法求得的最短路相同，长度近似相等。由此可知，本文所提出的基于支持向量机的改进遗传算法可行的。与此同时，在两种算法各进行 20 实验后，标准的遗传算法得到正确最短路结果的次数为 6 次，而本文提出的基于支持向量机的改进遗传算法得到正确最短路结果的次数为 9 次，由此可知，本文所提算法在保证结果正确的同时具有更为稳定的结果。

7. 结论

模糊最短路问题是模糊图论领域的一个非常流行的组合优化问题。随着网络的增大，求最短路径的时间复杂度越来越大。由于时间和经费的限制，只能对模糊图从起始节点到目标节点的部分路径总长度进行估算。为了找出模糊图中起始节点到目标节点的最短路径，本文使用支持向量机算法通过已知路径及其长度对未知路径的长度进行预测，并将其作为遗传算法中染色体的适应度值。然后使用遗传算法对

该模糊图的最短路问题进行求解。最后,在数值实例中,通过与标准的遗传算法进行对比,表明本文所提出的基于支持向量机的改进遗传算法是有效且较稳定的。

参考文献

- [1] Dubis, D. and Prade, H. (1980) Theory and Application/Fuzzy Sets and Systems. Academic Press, New York.
- [2] Chanas, S. and Kamburowski, J. (1983) The Fuzzy Shortest Route Problem. Interval and Fuzzy Mathematics, Poznan,: 35-41.
- [3] Takahashi, M.T. and Yamakami, A. (2005) On Fuzzy Shortest Path Problems with Fuzzy Parameters: An Algorithmic Approach. *NAFIPS 2005-2005 Annual Meeting of the North American Fuzzy Information Processing Society*, Detroit, 26-28 June 2005, 654-657. <https://doi.org/10.1109/NAFIPS.2005.1548615>
- [4] Shinkoh, O. (2004) Fuzzy Shortest Path Problems Incorporating Interactivity among Paths. *Fuzzy Sets and Systems*, **142**, 335-357. [https://doi.org/10.1016/S0165-0114\(03\)00225-2](https://doi.org/10.1016/S0165-0114(03)00225-2)
- [5] Nayeem, S. and Pal, M. (2005) Shortest Path Problem on a Network with Imprecise Edge Weight. *Fuzzy Optimization and Decision Making*, **4**, 293-312. <https://doi.org/10.1007/s10700-005-3665-2>
- [6] Deng, Y., Chen, Y., Zhang, Y. and Mahadevan, S. (2012) Fuzzy Dijkstra Algorithm for Shortest Path Problem under Uncertain Environment. *Applied Soft Computing*, **12**, 1231-1237. <https://doi.org/10.1016/j.asoc.2011.11.011>
- [7] Hassanzadeh, R., Mahdavi, I., Mahdavi-Amiri, N., et al. (2013) A Genetic Algorithm for Solving Fuzzy Shortest Path Problems with Mixed Fuzzy Arc Lengths. *Mathematical & Computer Modelling*, **57**, 84-99. <https://doi.org/10.1016/j.mcm.2011.03.040>
- [8] Lin, L., Wu, C. and Ma, L. (2020) A Genetic Algorithm for the Fuzzy Shortest Path Problem in a Fuzzy Network. *Complex & Intelligent Systems*, **3**, 1-10. <https://doi.org/10.1007/s40747-020-00195-8>
- [9] Hernandez, F., Lamata, M.T., Verdegay, J.L. and Yamakami, A. (2007) The Shortest Path Problem on Networks with Fuzzy Parameters. *Fuzzy Sets and Systems*, **158**, 1561-1570. <https://doi.org/10.1016/j.fss.2007.02.022>
- [10] Klein, C.M. (1991) Fuzzy Shortest Paths. *Fuzzy Sets and Systems*, **39**, 27-41. [https://doi.org/10.1016/0165-0114\(91\)90063-V](https://doi.org/10.1016/0165-0114(91)90063-V)
- [11] Iraj, M., Rahele, N., Armaghan, H. and Mahdavi, A.N. (2009) A Dynamic Programming Approach for Finding Shortest Chains in a Fuzzy Network. *Applied Soft Computing*, **9**, 503-511. <https://doi.org/10.1016/j.asoc.2008.07.002>
- [12] 王宁, 谢敏, 邓佳梁, 等. 基于支持向量机回归组合模型的中长期降温负荷预测[J]. 电力系统保护与控制, 2016, 44(3): 92-97.
- [13] 鞠秋文. PSO-SVM 算法在网络入侵检测中的研究[J]. 计算机仿真, 2011, 28(4): 130-132.
- [14] 王江, 孙劲光. 遗传算法在最短路径问题中的应用[J]. 微计算机信息, 2010, 26(36): 226-227, 114.