

基于基因搜索算法的短时交通流预测

李杰¹, 贺莹莹², 杨矿利²

(1. 平顶山学院 电气与机械工程学院, 河南 平顶山 467036;

2. 河南平高电气股份有限公司, 河南 平顶山 467001)

摘要:为提高短时交通流的预测速度和精度,提出一种基于基因搜索算法的短时交通流预测模型,对所有相同连续时间段的历史交通流数据进行编码,按照编码值的大小顺序排列,建立短时交通流预测基因库,根据当前连续时间段的交通流数据编码,快速搜索基因库中最相似的染色体种群.计算该种群中偏差值最小的3条染色体,组合解码后对当前交通流状况做出预测.实验结果表明,该模型可通过调整连续时间段的交通流数据个数以及短时交通流预测基因库的大小来提高预测精度.

关键词:智能交通系统;编码;基因搜索;偏差值计算;解码预测

中图分类号:TP183;U491.1⁺4

文献标识码:A

文章编号:1673-1670(2023)02-0047-06

0 引言

随着我国经济持续高速增长,大量汽车进入人们的日常生活,同时也带来了交通拥堵、交通污染、交通事故等问题.在当前道路条件下,智能交通系统是一种有效解决交通领域中相关问题的方法.在智能交通管理系统中,交通流预测算法作为基础理论研究,是实现智能交通管理系统的关键,具有重要意义^[1].

目前短时交通流预测算法种类繁多、效果不一,适用条件、应用范围也大不相同,总体来说主要有基础算法和优化组合算法两大类.基础算法预测过程简单,但预测精确度不高.优化组合算法预测精确度较高,但一般预测时间较长,算法优化组合过程复杂.文献[2]提出基于GM(1,1)的短时交通流预测模型.GM(1,1)模型作为基础算法中的一种,适用于小样本、贫信息不确定系统,当时间序列近似呈指数规律变化时,预测较为准确.不足之处在于算法本身具有滞后性,最大绝对误差较大^[3].文献[4]提出一种基于日周期性和周周期性的双周期时间序列短时交通流预测算法.该算法将工作日与非工作日的交通流状况分开进行预测,提高了预测的精确度.但是,经过实地数据采集,可以发现

短时交通流具有典型的日周期性,其周周期性特点不够突出,利用短时交通流周周期性特点进行预测的误差大于利用其日周期性特点进行预测的误差^[5].文献[6]提出一种K近邻非参数回归短时交通流预测算法,该算法在基于较大历史数据库进行预测时,精确度较高.不足之处在于先要计算样本之间的距离,对历史数据库中的样本进行分类后才能对当前交通流状况做出预测.对于省会城市交通流量变化大的特点,必须建立庞大的交通流数据库,才能获得准确的预测结果,相应的计算量较大.

笔者提出一种基于基因搜索算法的短时交通流预测模型.首先,利用编码的方法将连续时间段的历史交通流数据转变为单个染色体基因,进而对所有相同连续时间段的历史交通流数据编码,从而建立短时交通流预测基因库.然后,根据当前连续时间段的交通流数据对应染色体的编码值大小,在短时交通流预测基因库中快速搜索最相似的染色体种群.最后,利用该种群中偏差值最小的3条染色体进行组合,对当前交通流状况做出预测.

1 基因搜索算法

在短时交通流预测中,每一种算法都是基于一定量的历史交通流数据样本进行预测,历史交通流

收稿日期:2022-12-30

作者简介:李杰(1988—),女,河南省平顶山市人,工学硕士,平顶山学院电气与机械工程学院讲师,主要从事智能控制技术、高压开关设备智能化关键技术研究.

数据样本的量决定了算法的预测精确度^[7]. GM(1,1)算法仅利用一天的交通流数据样本进行预测^[8];BP神经网络算法能够利用多天的交通流数据样本训练之后进行预测;而K近邻算法基于交通流数据样本库进行预测,其预测精确度高于BP神经网络算法和GM(1,1)算法^[9].

K近邻算法通过计算当前点和历史数据库中点的欧氏距离来评价两者之间的相似度,但距离只能反映当前点和历史数据库中点的“靠近性”,不能直接反映它们之间的“形状相似性”,而“形状相似性”则直接反映交通流的发展变化规律^[10]. 遗传算法在寻找最优解时,先从代表问题可能潜在解集的一个种群开始,种群由经过基因编码的一定数目的染色体组成,这需要实现从表现型到基因型的映射(即编码). 初始种群产生之后,按照适者生存、优胜劣汰的原理,逐代进化产生出越来越好的近似个体.

笔者在K近邻算法与遗传算法的基础上,提出了一种基因搜索算法. 将连续 m 个时间段的交通流数据经过编码转变为染色体,通过比较染色体基因的相似性来搜索相似样本,对相似样本进行组合,从而实现交通流预测.

1.1 交通流数据的编码

令连续 m 个时间段的交通流数据分别为: $q(t-m+1), q(t-m+2), \dots, q(t-1), q(t)$,其对应的二进制码分别为

$$D_1^{t-m+1}, D_1^{t-m+2}, \dots, D_1^{t-1}, D_1^t.$$

将 $D_1^{t-m+1}, D_1^{t-m+2}, \dots, D_1^{t-1}, D_1^t$ 组合后作为一个染色体 C_n 的基因. 其中, n 代表交通流预测库中染色体的数量. 染色体 C_n 的基因包含比对基因片段 D_1 和预测基因片段 D_2 . D_1 由 $D_1^{t-m+1}, D_1^{t-m+2}, \dots, D_1^{t-1}$ 组合而成, D_2 由 D_1^t 组成. 具体编码过程如下.

1.1.1 编码长度的确定

对连续 m 个时间段的交通流数据分别进行二进制编码. 设连续 m 个时间段的交通流数据均为三位数,若个别时间段的交通流数据是两位数,则百位补作0. 由于每个数字位最大为9,对应的二进制编码为1001,因此每个时间段的交通流数据编码后均为 $3 \times 4 = 12$ 位的二进制码. D_1 由 $D_1^{t-m+1}, D_1^{t-m+2}, \dots, D_1^{t-1}$ 组合而成,则 D_1 的长度为 $12(m-1)$; D_2 由 D_1^t 组成,则 D_2 的长度为12. 因此,交通流预测库中每个染色体 C_n 的基因长度为 $12m$.

1.1.2 比对基因片段 D_1 的编码

因为利用算式 $q(t-m+1), q(t-m+2), \dots, q(t-1)$ 对第 t 时间段的交通流 $q(t)$ 进行预测时,距离 t 时间段越近的交通流数据对其影响越大,所以在染色体 C_n 的每一个基因片段中都将 D_1^{t-1} 的每一位排在其余时间段对应的二进制码前. 这样在搜索相似染色体 C_n 时,距离 t 时间段近的交通流数据优先满足相似性. 具体过程如下.

1) 将 $D_1^{t-m+1}, D_1^{t-m+2}, \dots, D_1^{t-1}$ 的第1位依次排列,作为染色体 C_n 的第1个基因片段,该片段包含第1个~第 $(m-1)$ 个基因位点. 其中, D_1^{t-1} 的第1位作为染色体 C_n 基因的第1个基因位点, D_1^{t-m+2} 的第1位作为染色体 C_n 基因的第 $m-2$ 个基因位点, D_1^{t-m+1} 的第1位作为染色体 C_n 基因的第 $m-1$ 个基因位点.

2) 将 $D_1^{t-m+1}, D_1^{t-m+2}, \dots, D_1^{t-1}$ 的第2位依次排列,作为染色体 C_n 的第2个基因片段,该片段包含第 m 个~第 $2(m-1)$ 个基因位点. 其中, D_1^{t-1} 的第2位作为染色体 C_n 基因的第 m 个基因位点, D_1^{t-m+2} 的第2位作为染色体 C_n 基因的第 $2m-3$ 个基因位点, D_1^{t-m+1} 的第2位作为染色体 C_n 基因的第 $2(m-1)$ 个基因位点.

3) 依次类推,完成染色体 C_n 基因中比对基因片段 D_1 的编码工作.

1.1.3 预测基因片段 D_2 的编码

将 D_1^t 编码后直接加在比对基因片段 D_1 后,作为染色体 C_n 的预测基因片段 D_2 . 依次类推,对历史交通流数据库中所有连续 m 个时间段的交通流数据完成编码,建立短时交通流预测基因库.

本文算法对历史连续时间段的交通流数据采用编码的方法进行处理,由于短时交通流预测基因库中不同类别的连续时间段交通流数据对应的编码结果不具备相似性,在搜索过程中互不影响. 因此无需对工作日、非工作日、重大节假日的交通流分开进行预测,更具有一般性. 且经过编码后的历史交通流数据在短时交通流预测基因库中是按照它们对应的编码值大小顺序排列的. 在搜索最相似染色体种群时,能够根据当前交通流状况对应的编码值大小在短时交通流预测基因库中快速定位搜索区间,锁定最相似的染色体种群. 由于搜索的快速性,本文算法在理论上可以基于无穷大历史交通

流数据库对当前交通流状况作出预测,其预测精确度较高。

1.2 染色体种群的搜索

对当前第 m 个时间段的交通流进行预测时,将当前连续 $m-1$ 个时间段的交通流数据编码为比对基因片段 D_1 , 预测基因片段 D_2 是未知的。为了与短时交通流预测基因库中染色体 C_n 的基因进行搜索比对,可令预测基因片段 D_2 为 0000 0000 0000。该基因片段无任何意义,只是便于搜索比对而增加的。

记当前连续 m 个时间段的交通流数据对应的染色体为匹配染色体 Y 。在短时交通流预测基因库中进行搜索匹配时,根据匹配染色体 Y 基因的前 $12(m-1)$ 位大小与短时交通流预测基因库中染色体 C_n 的前 $12(m-1)$ 位大小进行快速搜索,具体过程如下。

1.2.1 搜索标准

在短时交通流预测基因库中,搜索与匹配染色体 Y 的前 $\frac{12(m-1)}{2}$ 位基因相同的染色体区间作为初始种群(当短时交通流预测基因库较大时,可以选择更多位的相同基因作为搜索标准)。

1.2.2 种群数量设定

种群数量定义为 P (当短时交通流预测基因库较大时,可以选择较大的 P 值,反之选择较小的 P 值),本文选择 $P=10$ 。

1.2.3 搜索方法

在短时交通流预测基因库中搜索定位与匹配染色体 Y 的前 $\frac{12(m-1)}{2}$ 位基因值相同的染色体种群区间,统计符合条件的染色体 C_n 总数 P 。当 $P > 10$ 时,则以匹配染色体 Y 的第 $\frac{12(m-1)}{2} + 1$ 位基因值为标准,在第 1 次搜索完成的区间中继续搜索与匹配染色体 Y 的第 $\frac{12(m-1)}{2} + 1$ 位基因值相同的染色体 C_n ,然后重新统计符合条件的染色体 C_n 总数。如果 P 仍然大于 10,则以匹配染色体 Y 的第 $\frac{12(m-1)}{2} + 2$ 位基因值为搜索标准,在第 2 次搜索完成的区间中继续搜索。直到 $P \leq 10$ 时停止,得到相似染色体种群。

K 近邻算法直接通过计算样本之间的欧氏距

离来判断样本之间的相似程度,因此,所寻找到的样本包含了大量的非相似样本,导致预测精确度相对较低。而且,K 近邻算法在锁定相似样本时,需要计算出每一个样本的欧氏距离大小,导致算法快速性较差^[11-12]。

笔者提出的基因搜索算法与 K 近邻算法在原理上具有相似性,都是通过在历史数据库中搜索与当前交通流状况最相似的样本来对下一时间段的交通流状况进行预测^[13-14]。不同之处在于本文的基因搜索算法对各个样本首先定义编码规则,使每个样本具有不同的基因特征;并且,经过编码后的各个样本基因特征按照二进制大小依次排列,编码后的 2 个样本基因二进制数值越接近,则这 2 个样本的基因特征越相似。因此,可以快速、准确地根据二进制码的大小锁定相似样本,从而提高算法的快速性和精确性。

1.3 染色体偏差值计算

搜索完成后的染色体种群中,每一个染色体 $C_n^i (1 \leq n \leq 10)$ 的基因与匹配染色体 Y 基因的相似程度均不同,设定偏差值 Δ_n 来衡量两者相似程度。

种群中的每个染色体均含有 12 个基因片段,每 $m-1$ 个基因位点为一个基因片段,每个染色体的后 12 位基因不参与比对,只进行预测。每个基因片段中的 $m-1$ 个基因位点在 $D_1^{i-m+1}, D_1^{i-m+2}, \dots, D_1^{i-1}$ 中的位置相同,在计算偏差值时具有同等性。

记每个基因片段的偏差值为 $\Delta_n^i (i=1, 2, \dots, 11)$ 。设种群中染色体 C_n^i 的第 i 个基因片段与匹配标准染色体 Y 的第 i 个基因片段有 b 个基因不同,则初始种群中染色体 C_n^i 的第 i 个基因片段的偏差值为

$$\Delta_n^i = b \times 2^c \times 10^d. \quad (1)$$

式中: $i=1, 2, \dots, 11$; $b=0, 1, \dots, m-1$; 2^c 为该基因片段中的二进制码在 $D_1^{i-m+1}, D_1^{i-m+2}, \dots, D_1^{i-1}$ 中所对应的十进制转化关系; $c=0, 1, 2, 3$; 10^d 为该基因片段中的二进制码在 $D_1^{i-m+1}, D_1^{i-m+2}, \dots, D_1^{i-1}$ 中对应交通流数据数字百位、十位、个位, $d=2, 1, 0$ 。

得到种群中染色体 C_n^i 的总偏差值

$$\Delta_n = \Delta_n^1 + \Delta_n^2 + \dots + \Delta_n^{11}. \quad (2)$$

根据搜索标准,种群中染色体 C_n^i 与匹配标准

染色体 Y 的前 $\frac{12(m-1)}{2}$ 位基因相同, 因此 $\Delta_n^1 = \Delta_n^2 = \dots = \Delta_n^6 = 0$, 则可得到种群中染色体 C_n^1 的偏差值

$$\Delta_n = \Delta_n^7 + \Delta_n^8 + \dots + \Delta_n^{11}. \quad (3)$$

1.4 解码预测

根据式(3)的计算, 提取偏差值 Δ_n 最小的 3 条染色体 $C_n^1 (n=1, 2, 3)$, 将每一条染色体中的预测基因片段 D_2 解码为十进制数, 分别为 $q_h^1(t)$, $q_h^2(t)$, $q_h^3(t)$, 则当前 t 时间段的交通流量预测值

$$\overline{q(t)} = \frac{q_h^1(t) + q_h^2(t) + q_h^3(t)}{3}. \quad (4)$$

2 实验与结果分析

2.1 数据来源

采用平顶山市开源路丹尼斯附近由北向南单行道车流量作为实验数据, 从周一开始采集, 采集时间为每天 7:00—19:00, 采集间隔为 10 min, 连续采集 5 d, 共 5 组, 每组共有 72 个数据. 将前 4 组数据进行编码建立短时交通流预测基因库, 选取第 5 组 8:00—19:00 之间共 66 个数据作为预测对比. 预测过程具体如下.

1) 对采集到的前 4 组历史数据的第 1 个时间段~第 4 个时间段的交通流数据编码为一个染色体, 第 2 个时间段~第 5 个时间段的交通流数据编码为一个染色体, 依次类推, 建立短时交通流预测基因库.

实例 1 设有交通流序列 [181, 186, 203, 251]:

对 203 进行二进制编码后为 0010 0000 0011;

对 186 进行二进制编码后为 0001 1000 0110;

对 181 进行二进制编码后为 0001 1000 0001;

对 251 进行二进制编码后为 0010 0101 0001.

前 3 个交通流数据对应的二进制码的第 1 位均为 0, 则组合后的染色体基因第 1 个~第 3 个基因位点为 000. 3 个二进制码的第 2 位均为 0, 则组合后的染色体基因第 4 个~第 6 个基因位点为 000. 3 个二进制码的第 3 位为 1, 0, 0, 则组合后的染色体基因第 7 个~第 9 个基因位点为 100. 依次类推, 比对基因片段 D_1 为 000 000 100 011 011 000 000 000 000 010 110 101. 251 对应的二进制码 0010 0101 0001 作为染色体中的预测基因片段 D_2

直接加在比对基因片段 D_1 后面. 所以, 该交通流序列最终的染色体基因为 000 000 100 011 011 000 000 000 000 010 110 101 0010 0101 0001.

2) 对第 5 组 t 时间段之前的 3 个时间段的交通流数据 $q(t-3)$ 、 $q(t-2)$ 、 $q(t-1)$ 进行编码, 组合生成匹配标准染色体.

实例 2 设当前连续 3 个时间段的交通流数据 [182 183 212]:

比对基因片段 D_1 为 000 000 100 011 011 000 000 100 000 000 111 010;

增加 12 位 0 后的预测基因片段 D_2 为 000 000 100 011 011 000 000 100 000 000 111 010 0000 0000 0000.

3) 在短时交通流预测基因库中搜索与当前交通流状况相似的染色体种群.

实例 3 设有交通流序列 [181, 186, 203, 251]:

该交通流序列染色体基因为 000 000 100 011 011 000 000 000 000 010 110 101 0010 0101 0001;

设当前连续 3 个时间段的交通流状况为 [182 183 212]:

编码后的基因值为 000 000 100 011 011 000 000 100 000 000 111 010 0000 0000 0000.

通过比对可以看出, 3 条染色体基因的前 21 位完全相同, 因此这 3 条染色体是相似的. 3 条染色体的第 8 个、第 10 个、第 11 个基因片段均有一个基因位点不一样, 第 12 个基因片段有 3 个基因位点不一样, 根据偏差值计算式(1)得出该染色体的偏差值 $\Delta_n = 1 \times 2^0 \times 10 + 1 \times 2^2 \times 1 + 1 \times 2^1 \times 1 + 3 \times 2^0 \times 1 = 19$.

4) 提取出偏差值最小的 3 条染色体.

5) 对提取到的染色体预测基因片段进行解码.

实例 4 设计算后偏差值最小的 3 条染色体基因分别为:

000 000 001 110 110 000 000 000 000 010 011 101 0010 0101 0001;

000 000 001 110 110 000 000 000 010 100 000 100 0010 0100 1001;

000 000 001 110 110 000 000 010 000 001 110 011 0010 0101 0101.

对这 3 条染色体的预测基因片段进行解码,



即计算出每条染色体基因的后 12 位二进制码对应的十进制数字. 计算规则为每 4 位二进制码对应一个十进制交通流数字, 则 3 条染色体预测基因片段解码后对应的十进制数字分别为 251, 249, 255.

6) 根据式(4)计算出基因搜索算法对当前 t 时间段的交通流预测输出

$$\overline{q(t)} = \frac{q_h^1(t) + q_h^2(t) + q_h^3(t)}{3} = 252.$$

2.2 预测效果评价指标

基因搜索算法的预测效果评价指标为最大绝对误差、平均相对误差、平均绝对百分比误差. 设算法的预测值为 $\overline{q(t)}$, 则算法预测的最大绝对误差

$$\max_{er} = \max | \overline{q(t)} - q(t) |. \quad (5)$$

式中, $t = (1, 2, \dots, 66)$. 平均相对误差

$$\max_{er} = \frac{1}{66} \sum_{t=1}^{66} | \overline{q(t)} - q(t) |. \quad (6)$$

平均绝对百分比误差

$$\max_{er} = \frac{1}{66} \sum_{t=1}^{66} \left| \frac{\overline{q(t)} - q(t)}{q(t)} \right|. \quad (7)$$

为了对比分析, 通过 MATLAB 分别搭建了 GM(1,1)、BP 神经网络模型、K 近邻算法模型和基因搜索算法模型, 得到了如图 1 ~ 图 4 所示的模型预测结果和实际值的对比图.

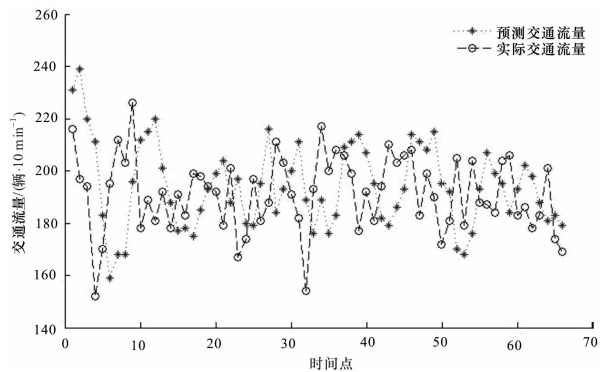


图 1 GM(1,1) 模型预测结果与实际值对比

通过表 1 可以看出, 基因搜索模型的最大绝对误差较 GM(1,1) 模型、BP 模型、K 近邻模型都有所降低, 平均相对误差以及平均绝对百分比误差也最小. 由于样本的质量直接影响预测结果的精确度, 本文算法特殊的编码规则保证了相似样本的高质量, 因此降低了预测结果的平均相对误差和平均绝对百分比误差. 并且, 预测结果采用 3 个相似样本的平均值作为最终预测结果, 有效防止了偶然性

事件导致预测结果误差较大的情况, 降低了预测结果的最大绝对误差.

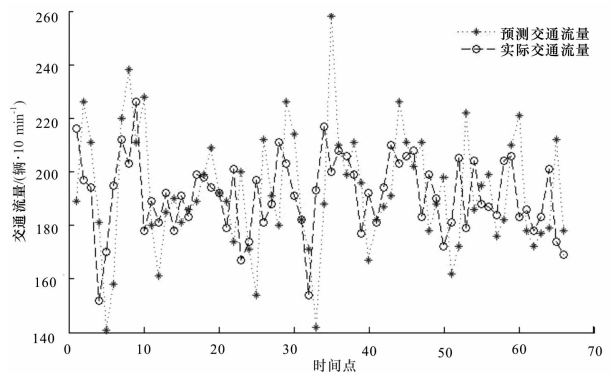


图 2 BP 模型预测结果与实际值对比

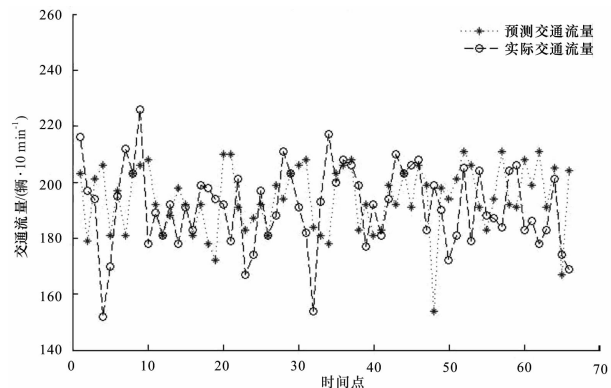


图 3 K 近邻模型预测结果与实际值对比

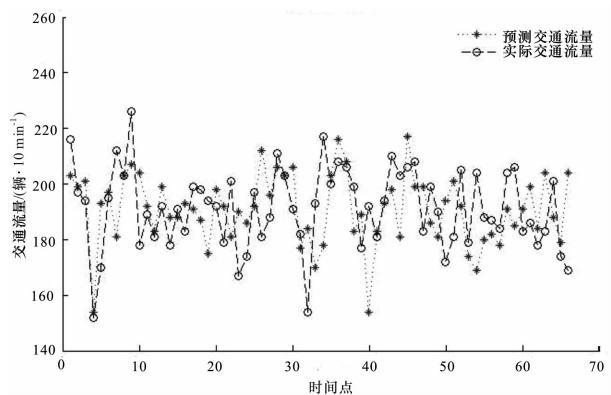


图 4 基因搜索模型预测结果与实际值对比

表 1 4 种模型评价指标统计

模型	最大绝对误差	平均相对误差	平均绝对百分比误差/%
GM(1,1) 模型	59	19.864	10.377
BP 模型	58	18.954	9.913
K 近邻模型	54	14.015	8.449
基因搜索模型	49	13.970	8.348

3 结论

笔者设计一种基因搜索短时交通流预测模型,

该模型通过对连续时间段的历史交通流数据进行编码排序,能够在短时交通流预测基因库中快速定位搜索到与当前交通流状况最相似的染色体种群区间;而且可以通过调整连续时间段的交通流数据个数以及短时交通流预测基因库的大小来提高预测的精确度.该算法对数据的适应性强,无需对交通流数据进行分类,预测精确度高,预测速度快,是一种快速有效的短时交通流预测方法.不足之处在于,对大都市交通流基数大、变化幅度大的情况,需要建立更加庞大的短时交通流预测基因库才能达到精确的预测.下一步工作将研究短时交通流预测基因库大小与预测精确度之间的数学关系,以及相邻路段短时交通流预测基因库是否通用或者具有一定的关联性等问题.

参考文献:

- [1] 李松江,祝绍淞,杨华民,等.基于时空相关性多任务神经网络的交通预测[J].计算机应用与软件,2021(9):286-292.
- [2] 郭欢,肖新平.基于GM(1,1,t,r)的城市道路短时交通流预测[J].交通运输系统工程与信息,2013(6):60-66.
- [3] 马梅琴,李风军,赵菊萍.基于改进粒子群算法优化小波神经网络的短时交通流预测[J].宁夏师范学院学报,2019(1):71-77.
- [4] 邱敦国,杨红雨.一种基于双周期时间序列的短时交通流预测算法[J].四川大学学报,2013(5):64-68.
- [5] LAN J H, GUO M, LU H F, et al. Short-term traffic flow combination forecast by co-integration theory[J]. Journal of Transportation Systems Engineering and Information Technology, 2011, 11(3): 71-73.
- [6] 程山英.基于K近邻非参数回归的短时交通流预测模型研究[J].数字技术与应用,2017(4):130-132.
- [7] 李明明,雷菊阳,赵从健.基于LSTM-BP组合模型的短时交通流预测[J].计算机系统应用,2019(10):152-156.
- [8] 唐少虎,周进,尚春琳.降雨情景下城市道路交通信号控制优化模型与方法[J].系统仿真学报,2020(2):149-156.
- [9] 夏元清,闫策,王笑京.智能交通信息物理融合云控制系统[J].自动化学报,2019(1):132-142.
- [10] HU X J, WANG W. Urban traffic flow prediction with variable cell transmission model[J]. Journal of Transportation Systems Engineering and Information Technology, 2010, 10(4): 73-78.
- [11] 张军,王寒凝.大波动短时公路交通流K-近邻预测的稳健组合方法[J].天津大学学报,2011(2):107-112.
- [12] 黄欣,莫海森,赵志刚.基于自适应烟花算法和K近邻算法的特征选择算法[J].计算机应用与软件,2020(5):268-274.
- [13] 王云鹏,郭戈.城市交叉口车辆速度与交通信号协同优化控制[J].控制与决策,2019(11):2397-2406.
- [14] 陈如清,李嘉春,俞金寿.基于FWADE-ELM的短时交通流预测方法[J].控制与决策,2021(4):925-932.

(责任编辑:王彦江)

Short-term Traffic Flow Prediction Based on Genetic Searching Algorithm

LI Jie¹, HE Yingying², YANG Kuangli²

(1. School of Electrical and Mechanical Engineering, Pingdingshan University, Pingdingshan, Henan 467036, China; 2. Henan Pinggao Electric Co., Ltd., Pingdingshan, Henan 467001, China)

Abstract: A short-term traffic flow prediction model based on the genetic searching algorithm was proposed in order to improve the prediction speed and accuracy of short-term traffic flow. Firstly, all the previous traffic flow dates in the fixed continuous period were encoded, and ordered by the size to develop the gene pool of the short-term traffic flow forecasting. Secondly, according to the code of current traffic flow dates, the most similar chromosome populations were quickly searched in the gene pool. Finally, three chromosomes with the minimum deviation in the population were calculated and combined to predict the current traffic flow conditions. The experiment results show that, the proposed method is faster and owns better precision. Moreover, the accuracy of the prediction can be improved by adjusting the quantity of the traffic flow data of the same continuous period and the size of the gene pool of the short-term traffic flow forecasting.

Key words: intelligent traffic system; coding; genetic searching; deviation calculation; decoding prediction