

# 机器学习在卫星遥测分析建模中的应用

王旭<sup>1</sup>, 都晓辉<sup>2</sup>, 陈昌麟<sup>1</sup>, 付宇<sup>1</sup>, 杨涛<sup>3</sup>

(1. 航天东方红卫星有限公司, 北京 100094; 2. 山东航天电子技术研究所, 山东烟台 264670;

3. 空间电子信息研究院, 西安 710100)

**摘要:** 卫星在研制和应用的各个环节产生了庞大的特征和相关性复杂的遥测数据, 人工判读或使用专家策略进行分析均无法充分梳理出这些数据的相关性并进一步挖掘和应用它们的价值; 利用 Python 强大的机器学习生态环境和算法, 能够实现对卫星遥测数据的分析和建模; 以星载铷钟遥测数据为样本, 提出了对卫星遥测数据进行算法分析和建立模型的方法, 为实现对铷钟的健康状态进行实时评估、预报、精准控制奠定了基础; 除遥测数据外, 该方法亦可用于分析卫星其它数据, 为卫星研制和应用过程中利用机器学习技术提高智能化水平提供了参考。

**关键词:** 机器学习; Python; 卫星遥测; 建模

## Application of Machine Learning in Satellite Telemetry Data Analysis and Modeling

Wang Xu<sup>1</sup>, Du Xiaohui<sup>2</sup>, Chen Changlin<sup>1</sup>, Fu Yu<sup>1</sup>, Yang Tao<sup>3</sup>

(1. DFH Satellite Co., Ltd., Beijing 100094, China; 2. Shandong Institute of Space Electronic Technology,

Yantai 264670, China; 3. Institute of Space Electronic Information, Xi'an 710100, China)

**Abstract:** All kinds of data are generated during satellite development, production and in-orbit flights, how to deal with it is a big problem. Due to the large amount of data, diverse data types, and complex characteristics of the correlation, it's almost impossible to sort out the correlation and explore the value of these data by using of manual analysis or expert strategy. A method for analyzing, algorithmic evaluation and modeling of rubidium clock's telemetry data by using Python is presented. This model can be used to predict the state of the rubidium clock, and to enable satellite to control the rubidium clock intelligently. This method can also be used to process data from other devices on satellite, and it provides a reference for the use of machine learning in the design and manufacture of satellites.

**Keywords:** machine learning; python; satellite telemetry data; modeling

## 0 引言

机器学习作为人工智能技术的重要分支, 在数据挖掘、生物特征识别、机器人等领域有着十分广泛的应用, 提高了诸多行业的生产力水平。但在我国卫星制造和卫星在轨管理领域, 使用机器学习技术对卫星遥测进行分析和建模的应用尚不普及, 仍然较多地依赖人工分析+专家系统解决特定的问题, 在面对数量庞大、多类型且相关性复杂的数据时, 无法提供有效的综合处理能力<sup>[1]</sup>。相比专家系统, 机器学习在这些方面有天然的优势: 它能够处理更加庞大的数据, 能够梳理和关联数据的维度信息, 并且具有数据处理的实时性。使用机器学习技术后, 卫星可以产生更多的数据, 为建立各种功能模型提供参数和特征, 而不必作为遥测下传。应用这些功能模型就能更加敏感地感知和预测卫星的状态, 进而提高卫星研制效率和卫星在轨自主管理水平。

本文以对星载铷钟的遥测数据建模为例, 介绍了利用

Python 的生态环境对卫星数据进行分析, 并使用机器学习的算法来建立预测模型的通用方法, 为探索人工智能技术在卫星制造领域的应用提供支撑。

## 1 算法研究与建模的方法

### 1.1 Python 库生态圈

Python 是一门动态语言, Python 由于拥有强大而成熟的开发生态, 逐步成为了机器学习领域最受欢迎的开发语言<sup>[2]</sup>。IEEE Spectrum 第六届编程语言排名评出了 2019 年度十大流行语言, Python 连续数年稳居榜首<sup>[3]</sup>, 如图 1 所示。这在 GitHub 上也得到了证实, 现在 Python 存储库总数排名仅次于 Java 和 C 语言。Python 在数据分析和机器学习建模方面的库生态圈主要包括 SciPy 和 scikit-learn。

SciPy 是在数学运算、科学和工程学方面被广泛应用的 Python 类库。在机器学习领域主要使用 NumPy、Matplotlib 和 Pandas 这三个类库来完成数据清洗、显示和数据分析工作<sup>[4]</sup>。

收稿日期: 2020-04-26; 修回日期: 2020-09-02。

作者简介: 王旭(1979-), 男, 北京人, 硕士, 高级工程师, 主要从事航天器星务系统设计和卫星智能化方向的研究。

引用格式: 王旭, 都晓辉, 陈昌麟, 等. 机器学习在卫星遥测分析建模中的应用[J]. 计算机测量与控制, 2021, 29(1): 210-214.

Rank	Language	Typo	Score
1	Python	☉ ☐ ☉	100.0
2	Java	☉ ☐ ☐	96.3
3	C	☐ ☐ ☉	94.4
4	C++	☐ ☐ ☉	87.5
5	R	☐	81.5
6	JavaScript	☉	79.4
7	C#	☉ ☐ ☐ ☉	74.5
8	Matlab	☐	70.6
9	Swift	☐ ☐	69.1
10	Go	☉ ☐	68.0

图 1 2019 年度编程语言排名

scikit-learn 依赖 SciPy 及其相关类库来运行, 其基本功能主要包括分类、回归、聚类、数据降维、模型选择和数据预处理六部分<sup>[5]</sup>。

### 1.2 确定算法并建模的步骤

利用 Python 进行分析数据、研究算法并预测模型共有七个基本步骤。

1) 定义问题: 这个过程主要研究和提炼问题的特征, 以帮助开发者更好地理解工程目标。

2) 导入数据: 在训练机器学习模型时需要用到大量数据, 最常用的做法是导入并利用目标工程的历史数据来训练。这些数据一般以 .csv 的格式存储, 通常有三种方法实现数据导入: 标准 Python 库导入、NumPy 导入、Pandas 导入。

3) 数据理解: 通过描述性统计和可视化数据来分析现有的数据。这个过程需要导入原始数据, 查看数据的维度、属性、类型, 并利用 Pandas 进行数据描述性统计、分析数据属性的相关性和分布情况。这个步骤一般会用到 DataFrame 的 shape 属性、Type 属性、describe () 方法、corr () 方法、skew () 方法等。其中, shape 属性可以方便获得数据集的行列维度信息; Type 属性用于获取每个字段的数据类型; describe () 方法用于审查数据的描述性统计内容; corr () 方法用于计算数据集中数据属性之间的关系矩阵; skew () 方法用于计算所有数据属性的高斯分布偏离情况。

4) 数据准备: 开始机器学习的模型训练前, 必须对数据进行格式化, 以便构建一个预测模型。这是因为不同算法对数据有不同的假定, 需要按照不同方式转换数据, 这样能够得到一个准确度比较高的模型。这个过程中一般会进行调整数据尺度、正态化数据、标准化数据或二值数据等操作。

5) 评估算法: 通过一定方法分离一部分数据, 用来评估算法模型, 并选取一部分代表数据进行分析, 以改善模型。这个过程包括主要通过使用 scikit-learn 进行数据预处理和数据特征选定, 以实现数据尺度的统一或正态化, 并降低数据的拟合度<sup>[6]</sup>。

6) 优化模型: 通过集成算法和算法调参提升预测结果的准确度。一般有三种集成算法用于优化模型: 装袋 (Bagging) 算法、提升 (Boosting) 算法和投票 (Voting) 算法。其中, 装袋算法先将训练集分离成多个子集, 然后通过各个子集训练多个模型; 提升算法是训练多个模型并组成一个序列, 序列中的每一个模型都修正前一个模型的错误; 投票算法是训练多个模型, 并采用样本统计来提高模型的准确度。算法调参是对参数化的机器学习模型进行处理, 用来提高模型的准确度。一般用网格搜索优化参数和随机搜索优化参数两种方法。

7) 模型部署: 完成模型, 将其部署到项目中, 并执行模型来预测结果和展示。

## 2 问题定义及数据准备

### 2.1 定义问题

这个项目将通过使用星载铷钟在真空环境下的遥测数据建立预测模型, 数据集的每一行都是铷钟状态的描述。数据中包含以下 4 个特征和 1 609 条数据。选定的数据特征如下:

LV: 铷钟灯电压;

LOCK: 锁定信号;

LS: 光强信号;

RU: 铷信号。

### 2.2 导入数据

星载铷钟遥测数据集以 .CSV 的格式来存储, 因此可以通过标准的 Python 库、NumPy、或者 Pandas 进行导入。由于在机器学习的项目中经常使用 Pandas 做数据清洗与数据准备工作, 因此本项目使用 Pandas 导入数据集。通过 Pandas 导入数据需要使用 pandas.read\_csv () 函数, 该函数的返回值为 DataFrame, 可以方便后续处理。

首先导入在项目中需要的类库, 并导入数据集到 Python 中, 本项目所导入的类库如图 2 所示, 有 numpy、matplotlib、pandas 和 sklearn, 其中 numpy 是一种开源的数值计算扩展库, matplotlib 是跨平台交互式环境的绘图库, pandas 是基于 numpy 的为解决数据分析任务而创建的工具库, sklearn (全称 scikit-learn) 是开源的机器学习工具包。

导入项目类库后, 为数据集中的相应数据特征定义名称, 并将返回的 DataFrame 命名为 dataset, 之后就可以开展数据分析理解工作了。

### 2.3 理解数据

导入数据后需要进行分析, 便于构建合适的模型。本项目使用 dataset.shape 方法观察数据维度, 使用 dataset.types 观察数据特征。

本项目有 1 609 条记录和 4 个特征属性, 如图 3 所示。

本项目所有的特征属性都是数字, 并且都是浮点数, 如图 4 所示。

使用 set\_option 库对数据进行一次简单查看, 该数据集的前 21 行数据如图 5 所示。

```

import numpy as np
from matplotlib import pyplot
from pandas import read_csv
from pandas import set_option
from pandas.plotting import scatter_matrix
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split
from sklearn.model_selection import KFold
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import GridSearchCV
from sklearn.linear_model import LinearRegression
from sklearn.linear_model import Lasso
from sklearn.linear_model import ElasticNet
from sklearn.tree import DecisionTreeRegressor
from sklearn.neighbors import KNeighborsRegressor
from sklearn.svm import SVR
from sklearn.pipeline import Pipeline
from sklearn.ensemble import RandomForestRegressor
from sklearn.ensemble import GradientBoostingRegressor
from sklearn.ensemble import ExtraTreesRegressor
from sklearn.ensemble import AdaBoostRegressor
from sklearn.metrics import mean_squared_error

```

图 2 模型预测需要导入的类库

LV	float64
LOCK	float64
LS	float64
RU	float64

dataset. shape: (1609, 4)

图 3 数据维度信息

图 4 数据特征属性

	LV	LOCK	LS	RU
0	3.193	0.000	-0.001	-0.002
1	3.193	0.000	-0.001	-0.002
2	3.193	0.000	-0.001	-0.002
3	3.192	0.000	-0.001	-0.002
4	3.193	0.000	-0.001	-0.002
5	3.193	-0.001	-0.001	-0.002
6	3.193	0.000	-0.001	-0.002
7	3.193	0.000	-0.001	-0.002
8	3.193	0.000	-0.001	-0.002
9	3.193	-0.001	-0.001	-0.002
10	3.193	0.000	-0.001	-0.002
11	3.193	0.000	-0.001	-0.002
12	3.193	-0.001	-0.002	-0.002
13	3.193	0.000	-0.001	-0.002
14	3.193	0.000	-0.001	-0.002
15	3.193	0.000	-0.001	-0.002
16	3.193	-0.001	-0.001	-0.002
17	3.193	0.000	-0.001	-0.002
18	3.193	0.000	-0.001	-0.002
19	3.192	0.000	-0.001	-0.002
20	3.193	0.000	-0.001	-0.002

图 5 数据特征

接下来分析数据的描述性统计信息。该信息中包含数据的最大值、最小值、中位值、四分位值等，分析这些数据能够加深对数据分布、数据结构等的理解，如图 6 所示。

	LV	LOCK	LS	RU
count	1.6e+03	1.6e+03	1.6e+03	1.6e+03
mean	2.5e+00	2.0e+00	1.6e+00	1.3e+00
std	9.1e-01	1.7e+00	1.2e+00	1.1e+00
min	-2.0e-03	-1.0e-03	-2.0e-03	-3.0e-03
25%	2.7e+00	0.0e+00	-1.0e-03	-2.0e-03
50%	2.7e+00	3.5e+00	1.9e+00	2.2e+00
75%	2.7e+00	3.5e+00	2.2e+00	2.3e+00
max	3.2e+00	3.5e+00	3.6e+00	2.3e+00

图 6 数据描述性统计信息

然后通过图 7 查看皮尔逊相关系数，或通过图 8 相关性矩阵图，找到数据特征之间的关系。

由皮尔逊相关系数和相关矩阵图可以看到，部分数据特征具有强关联关系。在机器学习建模中，当数据的关联

	LV	LOCK	LS	RU
LV	1.00	0.25	0.28	0.24
LOCK	0.25	1.00	0.55	1.00
LS	0.28	0.55	1.00	0.52
RU	0.24	1.00	0.52	1.00

图 7 皮尔逊相关系数

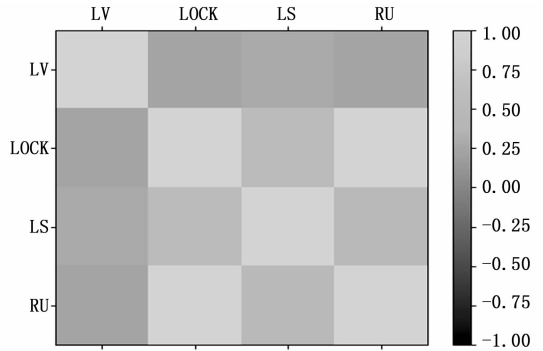


图 8 相关性矩阵

性比较高时，有些算法（如 linear、逻辑回归等）的性能会降低。所以在开始训练算法前，应对数据特征进行降维处理<sup>[7]</sup>。

### 3 算法评估及模型优化

#### 3.1 数据分析

在评估机器学习建模的算法时，由于存在过度拟合的原因，不能将训练集直接作为评估数据集，一般将评估数据集和训练数据集完全分开，采用评估数据集来评估算法模型，这样做的目的是用来评判生成模型的准确度<sup>[8]</sup>。一般工程上会将数据总量的 70% 到 90% 分离出来用于模型训练生成，10% 到 30% 用于评估算法模型。本项目分离 80% 的数据用于模型训练，20% 的数据用于算法评估。

通过对本项目数据集的初步分析不难发现部分数据表现出线性分布的特征，因此选择弹性网络回归和线性回归等算法处理数据的效果可能较好。同时，考虑到数据集中不同时刻采样到的数据表现出了离散化的特点，因此在算法选择上也可以同时考虑引入决策树或支持向量机等算法。如何评估选择合适的算法，需要进一步评估。

本项目使用十折交叉验证法将数据集中的数据进行分离，再带入各个算法中产生模型预测结果。由于带入算法过程前分离了 20% 的数据用于算法评估，因此将预测结果与评估数据进行均方差计算，均方差值小的算法准确度高。

本项目选择三个线性算法和三个非线性算法进行比较。

线性算法：线性回归（LR）、套索回归（LASSO）、弹性网络回归（EN）。

非线性算法：分类与回归树（CART）、支持向量机（SVM）、K 近邻算法（KNN）。

执行结果如图 9 所示。从结果看，CART 具有最优的 MSE，其次是 LR 和 KNN<sup>[9-11]</sup>。

#### 3.2 评估算法

本项目数据集中的四个参数采用了不同的度量单位，

```
LR:-0.003260(0.003150)
LASSO:-0.351965(0.005978)
EN:-0.180422(0.003854)
KNN:-0.001580(0.001985)
CART:-0.000952(0.001340)
SVM:-0.008898(0.002225)
```

图 9 原始数据集建模结果

这会影机器学习的结果。因此需要先对数据集中的参数进行正态化处理, 然后再次评估这些算法<sup>[12]</sup>。为了避免在数据正态化时出现数据采样偏差, 使用 Pipeline 对模型进行评估。为了与前面的处理结果比较, 采用与上面相同的均方差计算来评估算法模型。

从结果看, 仍然是 CART 具有最优的 MSE, 其次是 LR、KNN 和 SVM, 其中 SVM 在数据处理后精度得到明显提高。执行后结果如图 10 所示。

```
ScalerLR:-0.003260(0.003150)
ScalerLASSO:-1.005951(0.014296)
ScalerEN:-0.512914(0.007222)
ScalerKNN:-0.001526(0.001700)
ScalerCART:-0.000952(0.001340)
ScalerSVM:-0.003260(0.003150)
```

图 10 正态化后结果

### 3.3 优化模型

通过评估算法, 发现几个优秀算法的准确度比较接近, 这时可以使用集成算法进一步提高算法准确度。本项目使用以下两种流行的集成算法对表现较好的线性回归、K 近邻、分类回归树算法进行集成。袋装算法 (Bagging) 是将训练集分离成多个子集, 然后通过各个子集训练多个模型。提升算法 (Boosting) 是训练多个模型并组成一个序列, 序列中的每一个模型都会修正前一个模型的错误。本项目采用以下四种模型:

袋装算法: 随机森林 (RF)、极端随机树 (ET);

提升算法: AdaBoost (AB)、随机梯度上升 (GBM)。

在分析算法时依然采用和前面相同的评估框架和正态化后的数据进行。执行后结果如图 11 所示。从结果看, 线性算法和非线性算法的准确度都有提高, 其中随机梯度上升算法的准确度最好。

```
ScaledAB:-0.001578(0.001276)
ScaledAB-KNN:-0.001292(0.001193)
ScaledAB-LR:-0.003300(0.003216)
ScaledRFR:-0.000941(0.001225)
ScaledETR:-0.000950(0.001339)
ScaledGBR:-0.000927(0.001351)
```

图 11 集成算法后结果

集成算法都有一个参数 `n_estimators`, 可以通过调整这个参数进一步优化算法的准确度。调参后程序给出最优的 `n_estimators` 结果为 100, 如图 12 所示。

由此得到本项目最终的算法是随机梯度上升, 参数 `n_estimators=100`, 然后通过该算法对象的 `.fit` 方法进行训练并生成最终的模型。使用模型对象的 `.predict` 方法就可以进行数据预测了。最后通过评估数据集来评估该模型的准确

```
最优GBM: -0.0009286357686460576
使用 {'n_estimators':100}
```

图 12 算法调参后结果

度。本项目使用了 sklearn 库的 `mean_squared_error` 模块来评估模型的准确度, 以均方差来表示, 最终结果为: 5.077。

### 3.4 模型部署

模型部署是机器学习项目的最后一步, 也是最重要的步骤之一。选定算法后, 对算法训练生成模型, 通过 pickle 或 joblib 将其序列化, 并发布到硬件环境上。之后就可以利用机器学习解决实际问题, 如对新产生的遥测与模型进行比较, 并预测其变化趋势等<sup>[13]</sup>。

对于使用卫星遥测数据进行机器学习的结果既可以应用于卫星地面测试系统, 提高测试系统的数据分析能力和趋势预报能力, 也可以加载到星上嵌入式设备中, 使卫星具备数据现场分析和预测的能力, 这是卫星智能化的必要前提。

模型生成之后, 也需要定期对模型进行更新, 使模型处于最新、最有效的状态。

### 3.5 结果分析

对部署后的模型进行了预报能力评估测试, 测试的工况选择卫星在轨运行期间需要经历的 6 种典型环境, 因此将测试工况抽象为评估测试项 A~测试项 F。由于 LOCK 参数的数据聚合度较高, 因此选择以参数 LOCK 为测试对象进行预报准确度评估。模型预测结果见下表所示。从测试结果看, 模型预报的准确度优于  $6E-3$ , 说明模型训练和优化的效果较好, 能够起到数据分析和准确预测的作用。

表 1 模型预测结果评估表

	LS	RU	LV	LOCK	LOCK 预报	准确度( $\xi \rightarrow 0$ )
测试项 A	2.04	2.22	2.73	3.46	3.47	0.002 9
测试项 B	1.97	2.23	2.72	3.47	3.47	0
测试项 C	1.89	2.27	2.74	3.48	3.46	-0.005 8
测试项 D	2	2.22	2.73	3.47	3.47	0
测试项 E	1.98	2.23	2.73	3.45	3.45	0
测试项 F	1.91	2.34	2.74	3.47	3.46	-0.002 9

## 4 结束语

当前, 我国卫星设计和在轨自主管理的智能化主要通过专家策略来实现, 故障的类别与处理的过程是事先设计出来的, 这对于典型故障的识别与处理有很好的效果, 但由于无法充分挖掘星上遥测数据的价值, 无法进一步预测故障类型, 导致对于意料之外的故障往往束手无策。随着科技发展, 卫星智能化是必然的趋势。卫星产生的绝大部分数据都将直接用于实时分析与处理, 而不再需要生成遥测信息。作为卫星设计人员, 找到人工智能技术在卫星设计和应用中的着力点, 并利用机器学习的算法对大数据进行挖掘处理, 是需要认真思考和实践的。

本文利用 Python 强大的生态环境和开源的算法, 使用星载铷钟遥测数据, 按定义问题、导入数据、理解数据、数据分析、评估算法、优化模型到结果部署七个步骤, 详细介绍了模型建立的过程, 该模型可用于星上数据的实时处理和预报, 并使卫星具备实时自主控制的能力。本文是机器学习在卫星设计和应用上的探索, 所使用的数据集为星载铷钟真空高、低温环境下开机过程的数据, 因此训练出的模型是否适用于铷钟长期稳定后的状态还需要进一步测试验证<sup>[14-15]</sup>。

**参考文献:**

[1] 张 弓, 翟君武, 杨海峰. 导航卫星遥测数据趋势预测技术研究 [J]. 航天器工程, 2017, 26 (3): 70-77.

[2] Niko Wilbert, Tiziano Zito, Rike-Benjamin Schuppner. Building extensible frameworks for data processing: The case of MDP, Modular toolkit for Data Processing [J]. Journal of Computational Science, 2013, 4 (5): 345-351.

[3] Stephen Cass. The Top Programming Languages 2019 [EB/OL]. <https://spectrum.ieee.org/computing/software/the-top-programming-languages-2019>.

[4] Joey Bernard. Running Scientific Code Using IPython and SciPy [J]. Linux journal, 2013, TN. 228: 94-103.

[5] Varoquaux G, Louppe G, Pedregosa F. Scikit-learn: Machine Learning Without Learning the Machinery [J]. Mobile computing & communications review, 2015, 19 (1): 29-33.

[6] Pedro Marcelino, Maria de Lurdes Antunes, Eduardo Fortunato, Marta Castilho Gomes. Machine Learning for Pavement Friction Prediction Using Scikit-Learn [J]. Lecture Notes in Computer Science, 2017, 10423 (1): 331-342.

[7] Mohammad Hasan Shojaeefard, Abolfazl Khalkhali, Sadegh

Yarmohammadisatri. An efficient sensitivity analysis method for modified geometry of Macpherson suspension based on Pearson correlation coefficient [J]. Vehicle System Dynamics, 2017, 55 (6): 827-852.

[8] Walt Stefan vander, Colbert S. Chris, Varoquaux Gae. The NumPy Array: A Structure for Efficient Numerical Computation [J]. Computing in Science & Engineering, 2011, 13 (2): 22-30.

[9] 郭一帆, 唐家银. 基于机器学习算法的寿命预测与故障诊断技术的发展综述 [J]. 计算机测量与控制, 2019, 27 (3): 7-13.

[10] 帅 勇, 宋太亮, 郑玉杰, 等. 基于机器学习的军用软件过时淘汰评估方法研究 [J]. 计算机测量与控制, 2019, 27 (5): 131-135.

[11] 贺园园, 胡小敏, 梁腾飞. 基于深度机器学习的霾污染监测技术 [J]. 计算机测量与控制, 2020, 28 (8): 18-22.

[12] Emmanuely L. Souza, Regivan H. N. Santiago, Anne M. P. Canuto, Romulo O. Nunes. Gradual Complex Numbers and Their Application for Performance Evaluation Classifiers [J]. IEEE Transactions on Fuzzy Systems, 2018, 26 (2): 1058-1065

[13] Patrick Stanula, Amina Ziegenbein, Joachim Metternich. Machine learning algorithms in production: A guideline for efficient data source selection [J]. Procedia CIRP, 2018, 78: 261-266.

[14] 张馨予. 基于机器学习的车用主动防撞预警雷达信号识别系统研究 [J]. 计算机测量与控制, 2020, 28 (7): 225-229.

[15] 高翔霄, 任月慧, 李 思, 等. 基于遥测技术的新型传感器柔性结构振动优化控制方法 [J]. 计算机测量与控制, 2020, 28 (7): 86-90.

(上接第 209 页)

2) 采用双履带式运行结构设计, 极大地提高了除雪机工作的稳定性, 除雪机的防滑效果明显提升, 可以适应于多种复杂路面的除雪工作;

3) 利用网络 RTK 实时动态差分定位模块, 实现了对除雪车 5 厘米以内的高精度定位, 除雪作业时不会出现重扫和漏扫问题, 与同类除雪机器人相比, 除雪效率大大提高, 综合性能优于现有小型除雪机械。

4) 一种小型全自动除雪机的设计, 其应用范围得到了极大的提高, 可以适用于人行道、街道、小型广场、公园、社区、城区等区域路面的积雪清扫作业, 同时为我国智能化、全自动化的小型除雪机械的研制提供了一种可行性设计方案。

**参考文献:**

[1] 朱康熹. 小型除雪机手机控制系统的设计与研究 [D]. 重庆: 西南大学, 2017.

[2] 司癸卯, 李光辉, 王 贝, 等. 国内外道路除雪车发展现状及前景 [J]. 工程机械与维修, 2019 (2): 32-36.

[3] 董 杭. 校园用小型简易除雪车设计 [J]. 科技经济导刊, 2019, 27 (15): 97.

[4] 李天生, 李 方, 曾 理, 等. 多功能小型扫雪除冰车的研制 [J]. 机械设计与制造, 2011 (8): 129-131.

[5] 王旭景. 小型除雪车的设计及应用 [J]. 时代农机, 2019, 46 (4): 123-124.

[6] 尚佑峰. 一种适合城市路肩使用的除雪滚刷 [J]. 科学技术创新, 2019 (21): 171-172.

[7] 阿依提拉·伊力哈木. 扫雪机结构优化设计与动力稳定性分析 [D]. 乌鲁木齐: 新疆大学, 2014.

[8] 刘喜平, 刘春香, 闫晓丽, 等. 新型破冰除雪机设计与试验 [J]. 佳木斯大学学报 (自然科学版), 2014, 32 (3): 401-404.

[9] 李峰华. 自动行走除雪机的研究 [D]. 北京: 中国农业大学, 2005.

[10] 彭冬生, 刘 富, 王 震. 多功能除雪机的设计 [J]. 湖北汽车工业学院学报, 2011, 25 (2): 78-80.

[11] 张安康, 张 攀, 王 选, 等. 新型多功能道路清扫车设计 [J]. 科技创新导报, 2019, 16 (13): 87-88.