

## МОДЕЛИРОВАНИЕ ФЛОТАЦИОННЫХ ПРОЦЕССОВ С ИСПОЛЬЗОВАНИЕМ PYTHON

**Ю. Ш. Абдуганиева**

*старший преподаватель Алмалыкского филиала  
ТашГТУ имени И. Каримова г. Алмалык, Узбекистан.*

**Аннотация:** Современные подходы к изучению и оптимизации флотации требуют применения методов математического моделирования и вычислительных инструментов. В данной статье представлен опыт использования языка программирования Python для построения моделей флотационного процесса. Рассматриваются методы численного моделирования, визуализации, анализа данных, а также применение Python-библиотек (NumPy, SciPy, Matplotlib, scikit-learn) для симуляции поведения флотационной системы. Приведены примеры реализации моделей кинетики флотации, оптимизации дозировки реагентов и прогнозирования эффективности извлечения.

**Ключевые слова:** флотация, моделирование, Python, кинетика, оптимизация, машинное обучение, NumPy, SciPy.

Флотационные процессы играют важную роль в горно-обогатительной промышленности, обеспечивая селективное извлечение ценных минералов из руд. Однако флотация представляет собой сложную многофакторную систему, поведение которой трудно описать аналитически. Традиционные методы расчётов уступают место гибким и настраиваемым численным моделям. Язык программирования Python, обладая высокой доступностью, богатой экосистемой библиотек и поддержкой научных расчётов, становится эффективным инструментом моделирования флотации. Он позволяет строить как эмпирические, так и механистические модели, обрабатывать данные с реальных фабрик и применять методы машинного обучения для прогнозов и оптимизации.

### 1. Методология моделирования флотации

#### 1.1. Кинетические модели флотации

Наиболее распространённой формой описания флотации являются модели первого и новейших порядков:

Модель первого порядка:

$$R(t) = R_{\infty} \cdot (1 - e^{-kt})$$

где  $R(t)$  — степень извлечения,  $R_{\infty}$  — максимальное извлечение,  $k$  — кинетическая константа,  $t$  — время.

Модели этого типа легко реализуются с помощью библиотеки SciPy для численного анализа и подбора параметров.

## 1.2. Моделирование дозирования реагентов

Используются модели зависимости извлечения от концентрации коллектора, например:

$$R = a \cdot C / b + C$$

где  $C$  — концентрация реагента,  $a, b$  — эмпирические коэффициенты.

## 1.3. Прогнозирование и оптимизация

Машинное обучение (например, библиотеки `scikit-learn`) позволяет:

- классифицировать режимы флотации (эффективный/неэффективный);
- прогнозировать степень извлечения по входным параметрам (рН, плотность пульпы, расход воздуха);
- оптимизировать параметры управления (например, дозировку, обороты аэрации).

## 2. Реализация на Python: примеры

### 2.1. Построение кинетической модели

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.optimize import curve_fit
def model(t, R_inf, k):
    return R_inf * (1 - np.exp(-k * t))
t_exp = np.array([0, 1, 2, 3, 4, 5])
R_exp = np.array([0, 20, 35, 45, 50, 52])
params, _ = curve_fit(model, t_exp, R_exp)
t_fit = np.linspace(0, 5, 100)
R_fit = model(t_fit, *params)
plt.plot(t_exp, R_exp, 'o', label='Эксперимент')
plt.plot(t_fit, R_fit, '-', label='Модель')
plt.xlabel('Время, мин')
plt.ylabel('Извлечение, %')
plt.legend()
plt.show()
```

### 2.2. Машинное обучение для предсказания извлечения

```
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
# Примерные данные
X = np.array([[7.0, 1.2], [6.8, 1.5], [7.2, 1.1]]) # рН, концентрация реагента
y = np.array([65, 72, 63]) # извлечение
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3)
model = LinearRegression()
```

```

model.fit(X_train, y_train)
y_pred = model.predict(X_test)

```

### 3. Результаты и обсуждение

Разработанные модели позволяют:

- адекватно описывать экспериментальные данные кинетики флотации;
- подбирать оптимальные параметры реагентов;
- проводить численные эксперименты без затрат на лабораторные исследования;
- строить интеллектуальные прогнозные модели на основе накопленных данных.

Python оказался эффективной платформой как для обучения студентов основам моделирования флотации, так и для научных исследований.

Преимущества использования Python:

- открытость и бесплатность;
- широкий спектр библиотек (NumPy, SciPy, Pandas, Matplotlib, Scikit-learn, TensorFlow);
- простота интеграции с базами данных, системами визуализации и автоматизации.

Моделирование флотационных процессов с использованием Python открывает широкие возможности как в научной, так и в образовательной деятельности. Язык позволяет реализовывать как классические, так и современные методы анализа, визуализации и оптимизации процессов. Внедрение таких подходов способствует цифровизации горно-обогатительной отрасли и формированию новой генерации инженерных кадров, обладающих навыками моделирования и анализа данных.

#### Использованная литература

1. Пташинский Б.Б. Основы флотации. — Екатеринбург: УрГУ, 2018.
2. Абдуганиева Ю.Ш. Автоматизация технологических процессов - European Journal of Humanities and Educational Advancements 3 (12), 130-131, 2022 Millen, M., & Jowett, A. (2021). Flotation process modeling using machine learning. Minerals Engineering, 170, 106963.
3. van der Walt, S. et al. (2011). The NumPy Array: A Structure for Efficient Numerical Computation. Computing in Science & Engineering, 13(2), 22–30.
4. Pedregosa, F. et al. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, 2825–2830.
5. Shevtsov V.I. Применение Python в задачах инженерного моделирования. // Горный журнал, 2023, №2, с. 41–47.