

# РАЗРАБОТКА МОДЕЛИ РЕАКТОРА КАК ОБЪЕКТА УПРАВЛЕНИЯ НА ЯЗЫКЕ ПРОГРАММИРОВАНИЯ PYTHON

*Качанов Д.Д.<sup>1</sup>, Курганов В.В.<sup>2</sup>, Горюнов А.Г.<sup>3</sup>*

<sup>1</sup>*Томский политехнический университет, Инженерная школа информационных технологий и робототехники 8Т02, e-mail: danil3@tpu.ru*

<sup>2</sup>*Томский политехнический университет, Инженерная школа информационных технологий и робототехники, доцент, e-mail: kurganov@tpu.ru*

<sup>3</sup>*Томский политехнический университет, Инженерная школа ядерных технологий, профессор, e-mail: alex1479@tpu.ru*

## **Аннотация**

Данная работа посвящена моделированию нестационарных процессов в ядерном реакторе с использованием "точечной" модели. Реактор находится в критическом состоянии в начальный момент времени, после чего происходит скачкообразное изменение реактивности. Для анализа кинетики реактора используется система дифференциальных уравнений, учитывающая запаздывающие нейтроны. Работа использует шесть групп запаздывающих нейтронов и приводит таблицу постоянных распада и относительных долей для этих групп.

**Ключевые слова:** моделирование, ядерный реактор, кинетика реактора, запаздывающие нейтроны, постоянные распада, библиотеки Python.

## **Введение**

Актуальность проблемы эффективного управления ядерными реакторами остаётся важной в контексте обеспечения безопасности, надёжности и энергоэффективности ядерной энергетики [1]. Использование методов моделирования для анализа и оптимизации работы реакторов становится всё более актуальным, особенно в условиях стремительного развития современных алгоритмов управления, таких как Model Predictive Control [2].

Существующие модели точечной кинетики реакторов часто не учитывают нелинейности или обладают недостаточной скоростью работы, что ограничивает их применимость в реальных задачах управления реакторами. Например, модели, основанные на методе Монте-Карло, могут быть слишком медленными для использования в реальном времени, в то время как более простые модели точечной кинетики часто игнорируют нелинейности.

Цель данной работы состоит в создании библиотеки точечной кинетики реактора на языке программирования Python, которая будет учитывать нелинейности и обладать достаточно высокой скоростью работы. Это позволит исследователям и инженерам в области ядерной энергетики применять современные алгоритмы управления, такие как Model Predictive Control, для повышения безопасности и эффективности работы ядерных реакторов. Кроме того, разработка Open-source модели делает её доступной для всех заинтересованных лиц и способствует распространению знаний в области управления ядерными реакторами.

## **Описание работы**

Для рассмотрения и анализа кинетики реактора была использована система дифференциальных уравнений кинетики реактора, учитывая запаздывающие нейтроны.

Уравнения точечной кинетики представляют собой набор упрощенных уравнений, используемых для моделирования переходного процесса ядерного реактора. Эти уравнения описывают зависящие от времени изменения мощности реактора и популяции нейтронов в ответ на изменения реактивности. Уравнения точечной кинетики обычно выводятся на основе определенных допущений, таких как хорошо перемешанная активная зона реактора и постоянное время генерации нейтронов.

Обычно используют линеаризованную систему уравнений, но в данной работе система будет использоваться именно в исходном виде [3]. Основные уравнения показаны в формуле (1):

$$\begin{cases} \frac{dn(t)}{dt} = \frac{\rho(t) - \beta_{\Sigma}}{l} n(t) + \sum_{i=1}^6 \lambda_i C_i(t), \\ \frac{dC_i(t)}{dt} = \frac{\beta_{\Sigma i}}{l} n(t) - \lambda_i C_i(t), \end{cases} \quad (1)$$

Анализируя нестационарные процессы обычно используют шесть групп запаздывающих нейтронов [4]. Ядерно-физические параметры возьмем такие же, что и в модели из SimInTech для равных условий.

Результаты работы модели реактора, полученные в результате моделирования на языке программирования Python представлены ниже. На рисунке 1 отображен скачек реактивности с течением времени. Также на рисунке представлены графики концентраций нейтронов 6 групп и выдаваемая мощность.

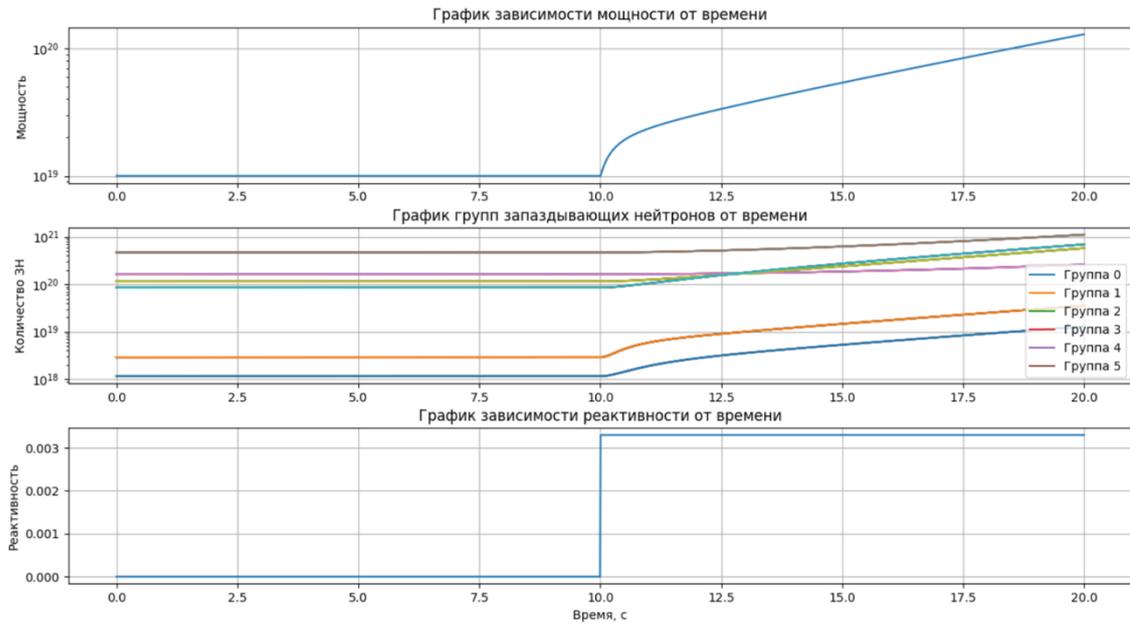


Рис. 1. Модель мгновенного скачка в Python

Сравним полученную модель с уже готовыми аналогами. За эталон взята модель в программе SimInTech, результаты сравнения представлены на рисунке 2.



Рис. 2. Сравнение с эталонной моделью

Исходя из полученных графиков видно, что модели схожи. Относительная погрешность полученной мощности за одинаковый промежуток времени (20 секунд) составила 3.13 % и представлена в формуле (3). Также модель, разработанная на языке программирования Python, показывает график реактивности и концентрации групп запаздывающих нейтронов, чего нет в эталонной модели. Еще значительной особенностью Python модели является то, что это «Open-source» проект, то есть, доступ к его исходному коду открыт каждому. Используя формулу относительной погрешности, с особенностью, что за эталон были взяты значения из модели SimInTech.

$$\delta = \frac{|11.81 - 12.18|}{11.81} \cdot 100\% = 3.13\% . \quad (2)$$

На рисунке 3 представлен интерфейс программы, написанный при помощи библиотеки Tkinter в Python. На главном окне задаются начальная мощность, реактивность и время симуляции.

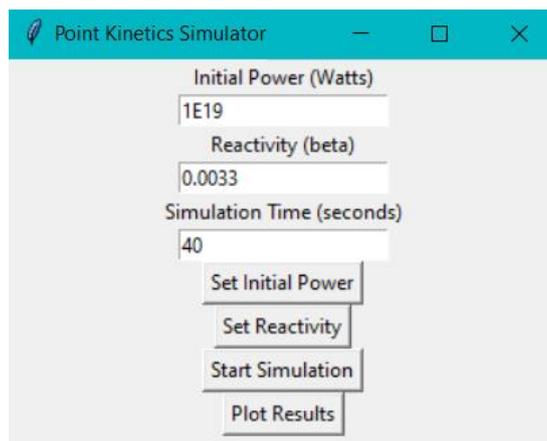


Рис. 3. Интерфейс программы

Затем добавим реактивность с тепловой обратной связью в полученную ранее модель. Основные уравнения показаны в формуле (2):

$$\left\{ \begin{array}{l} \frac{dn(t)}{dt} = \frac{\rho(t) - \beta_{\Sigma}}{l} n(t) + \sum_{i=1}^6 \lambda_i C_i(t), \\ \frac{dC_i(t)}{dt} = \frac{\beta_{\Sigma i}}{l} n(t) - \lambda_i C_i(t), \\ m_T c_T \frac{dT_T(t)}{dt} = n(t) - n_0, \\ \rho = \rho_0 + \alpha_T (T_T - T_{T_0}); \end{array} \right. \quad (3)$$

Предположим, что регулирующие стержни подняли и ввели некоторую реактивность, не изменяя потребность в паре [5]. Можно увидеть самокорректирующийся переходный процесс мощности, самоограничивающийся изменением температуры. Это означает, что в рабочем режиме стержни можно использовать для контроля температуры активной зоны, а не мощности.

Результаты работы модели реактора, полученные в результате моделирования представлены ниже. На рисунке 4 отображается результат работы полученной модели.

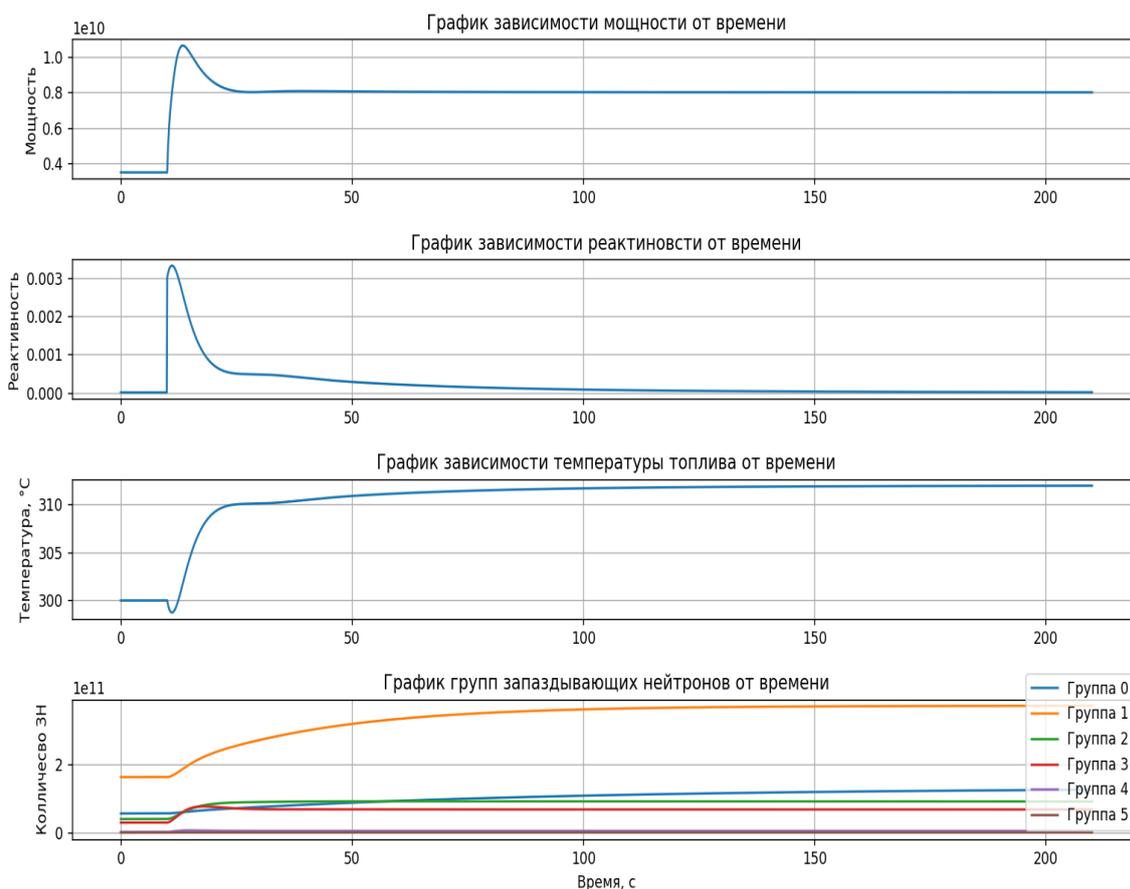


Рис. 4. График работы модели с обратной тепловой связью

### Заключение

В результате данной работы была получена модель кинетики реактора, реализованная на языке Python, что дает быстрое действие системы за счет встроенных библиотек. Сравнив полученные значения выдаваемой мощности с эталонной моделью, относительная погрешность составила 3.13 %. Модель основана на 7 дифференциальных уравнениях, включающих в себя константы для урана-235. Расчеты проводились при помощи метода интегрирования Эйлера. Интерфейс модели реализован при помощи библиотеки Tkinter.

### Список используемых источников

1. Климов, А.Н. Ядерная физика и ядерные реакторы: учебник / А.Н. Климов. – 3-е изд., стер. – Екатеринбург: АТП. – 2015. – 350 с. – Текст: непосредственный.
2. Bequette W., Process Control Modeling Design an Simulation, Prentice Hall, 2003. – 564 с.
3. Владимиров Владимир Иванович. Физика ядерных реакторов: практические задачи по их эксплуатации / В. И. Владимиров. – 5-е изд., перераб. и доп. – Москва: URSS, 2009. – 478 с.: ил. – Текст: непосредственный.
4. Демченко В.А. Автоматизация и моделирование технологических процессов АЭС и ТЭС / В.А. Демченко. – Одесса: Астропринт. – 2001. – 395с.
5. Кипин Дж.Р. Физические основы кинетики ядерных реакторов. М.: Атомиздат. – 1967.