

Ускорение кинетических расчетов в химической технологии с помощью нейросетей PINN

Хмелёв А.П.¹, Безруких П.Д.¹, Верховин М.А.¹,
Гриценко Е. Ф.²

Научный руководитель – кандидат технических наук, Щемелинин В.Л.¹

¹Университет ИТМО, ²НИ ТПУ

alexysxeightn@gmail.com

Введение

Высокоточное моделирование химико-технологических процессов в реакторах (включая каталитические системы и микрокинетiku) традиционно опирается на численное решение систем алгебраических уравнений, ОДУ/ЧДУ, а также на итерационные процедуры идентификации параметров. При росте размерности модели (детализированные механизмы, многокомпонентные смеси, многомасштабность) вычислительная стоимость становится критическим ограничением: расчеты плохо масштабируются в оптимизационных контурах, цифровых двойниках и задачах управления в реальном времени. Дополнительно, промышленные требования к достоверности вынуждают сохранять физическую корректность моделей (балансы масс/энергии, законы сохранения, термодинамические ограничения, кинетические связи), что затрудняет применение «чисто» эмпирических ML-приближений.

В ходе работы над проектом ReacModeler были выявлены следующие задачи:

- ускорение вычислений при сохранении инженерной точности,
- повышение точности за счет регуляризации физикой и устойчивого учета параметрической зависимости,
- учет исторических данных для адаптации моделей и улучшения прогноза.

Для решения этих задач предлагается использовать подход машинного обучения на основе physics-informed neural networks (PINNs), аналогичный методу, продемонстрированному для ускорения микрокинетики синтеза Фишера–Тропша (Patel et al., 2023) [1].

Основная часть

В рамках ReacModeler предлагается заменить наиболее «тяжелые» участки выч. контура (итерационное решение нелинейных систем, многократное решение ОДУ/ЧДУ при переборе условий, вычисление градиентов для оптимизации) на нейросетевой параметризованный решатель, обучаемый с учетом уравнений модели и физико-химических ограничений. Ключевая идея PINN состоит в том, что нейросеть аппроксимирует неизвестные функции состояния (например, поверхностные покрытия, концентрации, температуры), а обучение направлено на минимизацию невязок исходных уравнений (балансов, кинетики) и граничных/начальных условий, что обеспечивает физическую согласованность решения [2, 3].

Архитектура решателя позволяет провести следующую последовательность выполнения расчетов:

1. Формализация физико-химической модели и выделение вычислительного узкого места. Определяется класс уравнений (нелинейные алгебраические системы для стационарной микрокинетики; ОДУ/ЧДУ для нестационарных реакторных моделей; сопряженные уравнения тепло- и массопереноса). Выделяются компоненты, которые доминируют по времени счета: итерационные корневые решатели, жесткие интеграторы, многократные вызовы модели в калибровке и оптимизации.
2. Построение нейросетевого суррогата как параметризованного решателя.

Нейросеть задает отображение вида (T , p_i , m , геометрия, параметры кинетики) \rightarrow (состояния, скорости, потоки), то есть решает не «одну задачу», а целое семейство решений в диапазонах условий. Это принципиально важно для ReacModeler, где требуется многократно переиспользовать решатель в разных режимах и сценариях.

3. Физически-информированное обучение (physics-informed loss) и сохранение законов. Функция потерь включает:

- невязки уравнений баланса (масса/энергия), кинетических уравнений и алгебраических связей (например, site-balance на поверхности катализатора);
- выполнение граничных/начальных условий;
- ограничения физической реализуемости (неотрицательность концентраций и покрытий, допустимые диапазоны температур/давлений);
- при необходимости — регуляризацию термодинамических соотношений и ограничений стехиометрии.

4. Гибридизация с историческими данными и адаптация модели. Для учета исторических данных ReacModeler вводит дополнительный «data-driven» член в функцию потерь: расхождение между прогнозом PINN и измерениями/историческими трендами (составы, выходы, температуры, селективности). Это позволяет:

- калибровать (или дооценивать) кинетические параметры и поправочные коэффициенты под конкретный катализатор/сырье;
- компенсировать неизбежную неполноту механизма (неучтенные пути, деградация катализатора) без разрушения физических ограничений;
- выполнять перенос модели между установками при сохранении структуры уравнений.

5. Интеграция в выч.контур ReacModeler и ускорение оптимизации.

После обучения PINN используется как быстрый прямой решатель (inference) на CPU/GPU. Существенное преимущество — дифференцируемость: производные выходов по входам (условиям процесса, параметрам) вычисляются методом автоматического дифференцирования. Это ускоряет:

- градиентную оптимизацию режимов реактора;
- идентификацию параметров (inverse problem);
- задачи управления и Model Predictive Control, где требуется многократный пересчет модели и градиентов.

Выводы

Применение PINN в ReacModeler позволяет заменить вычислительно дорогие численные процедуры решения систем дифференциальных и/или нелинейных алгебраических уравнений на нейросетевой параметризованный решатель, который обеспечивает кратное ускорение вычислений за счет эффективного inference и GPU-вычислений; сохраняет физическую корректность благодаря включению законов сохранения, кинетики и ограничений в функцию потерь; поддерживает учет исторических данных через гибридную схему обучения. В результате формируется инструмент для цифровых двойников, оптимизации и управления хим. реакторами, где скорость и физическая достоверность модели являются одновременно обязательными требованиями.

Литература

1. Patel H. et al. Accurate and Fast Fischer–Tropsch Reaction Microkinetics using PINNs, arXiv:2311.10456v1, 2023.
2. Raissi M., Perdikaris P., Karniadakis G. Physics-informed neural networks, J. Comput. Phys., 2019.
3. Karniadakis G. et al. Physics-informed machine learning, Nat. Rev. Phys., 2021.