

## РАЗРАБОТКА ПАКЕТА ПРОГРАММ VBA-EXCEL ДЛЯ МОДЕЛИРОВАНИЯ ЗАДАЧ ХИМИЧЕСКОЙ КИНЕТИКИ

*С. О. Травин<sup>1\*</sup>, О. Б. Громов<sup>2</sup>*

<sup>1</sup>Федеральное государственное бюджетное учреждение науки Институт химической физики им. Н.Н. Семенова Российской академии наук, Москва, \*e-mail: TravinSO@yandex.ru

<sup>2</sup>Акционерное общество «Высокотехнологический научно-исследовательский институт неорганических материалов имени академика А.А. Бочвара», Москва, Россия

Поступила в редакцию 01.04.2018 г.

Разработан пакет программ для решения прямой и обратной задач химической кинетики. Пакет реализован в виде надстройки VBA-Excel и может быть легко интегрирован с любыми другими приложениями. Пакет включает три интегратора различной сложности, которые в совокупности позволяют моделировать динамику практически любых химико-биологических систем, включая гибридные, т.е. содержащие события, изменяющие саму структуру модели. Отличается интуитивно ясным интерфейсом и простотой использования.

*Ключевые слова:* задача Коши, прямая и обратная задачи химической кинетики, квадратурная формула Гаусса.

### ВВЕДЕНИЕ

В работе [1] мы отмечали, что при рассмотрении вопросов математического и программного обеспечения для решения прямых задач химической кинетики удобство использования того или иного программного пакета определяется, в первую очередь, возможностями интерфейса ввода исходной схемы и вывода расчетных результатов. В этой связи важно еще раз сформулировать основные требования к инструментарию моделирования и, исходя из них, выбрать наиболее подходящую программную среду.

В работе [2] авторы критически отзываются о возможностях программной среды Excel в качестве средства математического моделирования сколь угодно сложных систем: «Появились новые статьи и учебники, посвященные моделированию, однако их авторы почему-то используют преимущественно программные среды для моделирования изолированных (однокомпонентных систем), такие как Matlab, MathCAD, Maple и даже Microsoft Excel в качестве среды моделирования!»

Согласиться с такой оценкой мы не можем, поскольку у этой программной среды имеется целый набор преимуществ перед ближайшими конкурентами, а именно:

- 1) С работой в Excel знаком практически любой продвинутый пользователь персонального компьютера, в связи с чем отпадает необходимость изучения дополнительных языков и перепривыкания к особенностям менее популярных пакетов. По состоянию на конец

сентября 2015 года, число пользователей Microsoft Office достигло 1,2 млрд человек, из которых добрая половина обладает навыками работы с Excel.

- 2) В Excel наиболее развитые средства графического отображения, причем они не требуют открытия дополнительных окон и позволяют использовать несколько десятков разновидностей представления результатов, включая 3D-диаграммы. Дальнейшая передача в Word или PowerPoint осуществляется самым простым и эффективным способом, гарантируя 100% совместимость.
- 3) В пакете Excel имеется встроенная полнофункциональная интегрированная среда разработки (VBA, Visual Basic for Applications). При этом режим макроккоманд позволяет создать код, повторяющий ввод клавиатурных команд, даже пользователю, имеющему самое поверхностное представление о программировании. В то же время VBA представляет собой полноценный язык программирования, допускающий работу с классами, и объектно-ориентированный подход – для пользователей с более глубокими познаниями в этой области.
- 4) В режиме отладчика имеется возможность по ходу выполнения программного кода вносить изменения как в значения переменных и/или констант выполняемой программы, так и в саму логику программы. Исключительное удобство представляет возможность редактирования формул непосредственно в ячейке.
- 5) В Excel встроено множество стандартных диалоговых окон. Их можно использовать, если требуется запросить у пользователя какую-либо информацию. Например, если нужно открыть файл, вызывается диалог открытия файла.
- 6) Режим конструктора позволяет работать с элементами ActiveX для создания необходимого интерфейса произвольной сложности и требуемого удобства на рабочем листе. Временные затраты при этом минимальны.
- 7) Помимо многочисленных встроенных функций, предусмотренных разработчиком, имеется возможность использования пользовательских функций, адаптированных в максимальной степени под решение задач «здесь и сейчас». При этом в огромном количестве случаев нет никакой необходимости прибегать к созданию собственного кода – миллиард пользователей охотно публикует свои наработки в сети. Накопленный интеллектуальный потенциал «коллективного разума» избавляет от непродуктивных временных затрат на разработку уже имеющихся решений для типовых задач.

На этом закончим перечислять достоинства программной среды Excel-VBA и перейдем к описанию возможностей разработанного в ней программного пакета «Кинетика» [3].

**ОПИСАНИЕ РАЗРАБОТАННОГО ПРОГРАММНОГО ПАКЕТА «КИНЕТИКА»**

Первичной задачей было создание эффективного инструмента для моделирования химических и химико-биологических систем, представляющих основной интерес для прогнозирования экологической безопасности. Формализм прямой задачи химической кинетики достаточно прост для того, чтобы сделать интерфейс ввода схемы механизма химической реакции и стартовых значений предельно простым, ясным и дружелюбным для пользователя, что в дальнейшем исключит возможность ошибок и/или неправильного толкования составленной системы обыкновенных дифференциальных уравнений (ОДУ).

При этом открывается возможность и для решения некоторых разновидностей обратной задачи. Заметим, что в числе штатных надстроек Excel имеется утилита «поиск решения», которая в большинстве случаев успешно производит подбор параметров, минимизируя сумму квадратов отклонений расчетной функции от реперных значений (полученных экспериментально или смоделированных другим способом). При этом расчетная функция должна содержать лишь формулы, явным образом включающие подгоночные параметры. Эта утилита может быть использована как для проверки адекватности применяемой модели, так и для установления числовых значений входящих в нее параметров, но только в том случае, если целевая функция может быть вычислена явно, т.е. с вычислением одних только алгебраических функций, без использования циклов, рекурсий, условных или безусловных переходов. Таким образом, в случае вычисления кинетических кривых в ходе интегрирования системы ОДУ эта утилита оказывалась неработоспособна, в связи с чем для решения обратных задач более общего вида нам предстояла разработка кодов специализированного модуля фитинга (с простейшими рекурсивными функциями встроенный минимизатор справляется).

Как показал опыт математического моделирования природных экосистем, лишь в отдельных, довольно редких случаях, при их описании можно ограничиться одним только химико-кинетическим формализмом, основанным на законе действующих масс. Даже включение процессов массопереноса не полностью приближает модели к реальным системам. К настоящему времени практически очевидным стало то, что чаще всего приходится иметь дело с так называемыми дискретно-непрерывными или гибридными системами.

В них осуществляется набор режимов, каждый из которых описывается своим набором дифференциальных и/или алгебраических уравнений, дающих непрерывное и гладкое поведение системы. Кроме того, в системе могут происходить дискретные *события*, которые осуществляют практически мгновенное переключение с одного набора уравнений на другой, либо вызывают скачкообразное изменение обобщенных координат и импульсов на другие координаты и импульсы, либо одновременно и то и другое. Зачастую дискретные события вызываются именно самими плавно протекающими процессами, как-то превышение предела растворимости по какому-либо из веществ, пересыщение пара при понижении температуры и т.д. Возможны,

однако, и случаи внешнего импульсного управляющего воздействия, которое напрямую не связано с внутрисистемными превращениями.

До настоящего времени распараллеливание алгоритмов решения систем алгебраических и дифференциально-алгебраических уравнений (ДАУ) практически не использовалось в коммерчески доступных программах моделирования химических систем, но это вопрос самого ближайшего будущего. В последние годы велись работы по связыванию симуляторов химических систем с программами вычислительной гидродинамики для реализации расчетов сложных с точки зрения гидродинамики объектов [4, 5].

Гибридные системы неизбежно встречаются каждый раз при математическом моделировании объектов, в которых состояние среды непрерывно изменяется в соответствии с некоторыми физическими законами, но имеется ненулевая вероятность импульсных внешних воздействий - залповых сбросов, регулируемых перетоков, скачкообразных изменений характеристик (фазовые переходы), срабатывающих в дискретные моменты времени. Примерами таких экосистем могут служить природные территории, на которых возможны аварии промышленных предприятий, внесение сельскохозяйственных удобрений и пестицидов, природные катаклизмы, миграции водорослей, насекомых и грызунов и т.п.

Неожиданные особенности подобных систем проявляются шире, чем обычных. Гибридная система может быть получена также при кусочно-линейной аппроксимации сложной нелинейной системы дифференциальных уравнений. Решения различных задач управления для нелинейной системы, таким образом, могут быть аппроксимированы решениями аналогичных задач для гибридной системы.

В связи с изложенным, в разрабатываемом пакете особое внимание было уделено разработке и адаптации алгоритмов численного интегрирования различных систем ОДУ и ДАУ, в том числе включающих дискретные события и переключения режимов. Идеального универсального алгоритма, одинаково пригодного для всех типов задач, по-видимому, не существует. По этой причине в пакет были включены хорошо себя зарекомендовавшие интеграторы по методу Гира [6, 7] и по алгоритму, любезно предоставленному авторам Е.А. Новиковым [8], а также оригинальная версия интегратора, основанного на квадратурной формуле Гаусса-Лежандра, специально разработанная нами для расчета гибридных систем.

Все три использованных сольвера («решателя» систем ОДУ и ДАУ) относятся к способам численного решения задачи Коши для системы обыкновенных дифференциальных уравнений, имеющей вид

$$\frac{du(t)}{dt} = f(t, u) \quad | \quad t > 0, u(0) = u^{(0)}. \quad (1)$$

Известно существование и единственность решения в замкнутой области D

$$D = \left\{ |t| \leq a, \quad \left| u_i - u_i^{(0)} \right| \leq b, \quad i = 1, 2, \dots, m \right\}. \quad (2)$$

Кроме того, в  $D$  функции  $f_i$  удовлетворяют условию Липшица по аргументам  $u_1, u_2, \dots, u_m$  т. е.

$$f_i(t, u'_1, u'_2, \dots, u'_m) - f_i(t, u''_1, u''_2, \dots, u''_m) \leq L\{|u'_1 - u''_1| + |u'_2 - u''_2| + \dots + |u'_m - u''_m|\} \quad (3),$$

для любых точек  $(t, u'_1, u'_2, \dots, u'_m)$  и  $(t, u''_1, u''_2, \dots, u''_m)$  из области  $D$ . Подобные системы описывают подавляющее большинство химико-технологических, биологических и природных процессов.

Наиболее подробно в учебной и научной литературе описан способ численного решения системы дифференциальных уравнений методом Рунге-Кутты [9], который базируется на использовании сокращенного ряда Тейлора. Можно разложить точное решение  $y_{n+1} = y(t_n + \tau)$  задачи

$$y' = f(t, y), \quad y(t_0) = y_0 \quad (4)$$

в кругу узла сетки  $t_n$  в ряд Тейлора и ограничить ряд членами  $p$ -го порядка:

$$y(t_{i+1}) = y(t_i) + y'(t_i)\tau + \frac{1}{2}y''(t_i)\tau^2 + \dots + \frac{1}{p!}y^{(p)}(t_i)\tau^p + O(\tau^{p+1}).$$

Основная идея построения явных методов Рунге-Кутты  $p$ -го порядка состоит в замене части ряда Тейлора [8]

$$T_p(y_i, t_i, \tau) = y'(t_i)\tau + \frac{1}{2}y''(t_i)\tau^2 + \dots + \frac{1}{p!}y^{(p)}(t_i)\tau^p$$

функцией  $\varphi_p(y_i, t_i, \tau)$ , которая содержит значение правой части уравнения  $f(t, y)$  в промежуточных точках отрезка  $[t_i, t_{i+1}]$  и приближает  $T_p(y_i, t_i, \tau)$  с точностью до  $p$ -го порядка  $|\varphi_p(y_i, t_i, \tau) - T_p(y_i, t_i, \tau)| < C\tau^{p+1}$ , где  $C$  – константа, которая не зависит от шага  $\tau$ .

Для построения вычислительных схем методов Рунге-Кутты четвертого порядка в тейлоровском разложении искомого решения  $y(x)$  учитываются члены, содержащие степени шага  $h$  до четвертой включительно. После аппроксимации производных правой части ОДУ  $f(x, y)$  получено семейство схем Рунге-Кутты четвертого порядка, из которых наиболее используемой в вычислительной практике является следующая:

$$y(x_0 + h) = y_0 + (k_1 + 2k_2 + 2k_3 + k_4)/6 + O(h^5), \quad (5)$$

где

$$\begin{aligned} k_1 &= hf(x_0, y_0), \\ k_2 &= hf(x_0 + h/2, y_0 + k_1/2), \\ k_3 &= hf(x_0 + h/2, y_0 + k_2/2), \\ k_4 &= hf(x_0 + h, y_0 + k_3). \end{aligned}$$

Вышеприведенная схема на каждом шаге  $h$  требует вычисления правой части ОДУ в четырех точках. Локальная погрешность схемы имеет пятый порядок, глобальная – четвертый. Схема обобщается для систем ОДУ, записанных в форме Коши. Для удобства программной реализации, особенно в случае систем ОДУ, формулы (5) рекомендуется преобразовать к виду

$$y_i(x_0 + h) = y_{i0} + (q_{i1} + 2q_{i2} + q_{i3} + q_{i4})/3 + O(h^5), \quad (6)$$

где

$$q_{i1} = h_2 f_i(x_0, y_{i0}), \quad h_2 = h/2,$$

$$q_{i2} = h_2 f_i(x_0 + h_2, y_{i0} + q_{i1}),$$

$$q_{i3} = h f_i(x_0 + h_2, y_{i0} + q_{i2}),$$

$$q_{i4} = h_2 f_i(x_0 + h, y_{i0} + q_{i3}),$$

$i = 1, 2, \dots, n$  - номер уравнения в системе ОДУ из  $n$  уравнений.

Оценку погрешности решений системы ОДУ, получаемых методом Рунге-Кутты четвертого порядка, можно провести по первой формуле Рунге, которая при кратности изменения шага  $k = 2$  принимает вид

$$R_0 = [y_h(x) - y_{2h}(x)]/15.$$

Недостатком этого способа является невысокая точность полученного приближенного решения системы дифференциальных уравнений, обусловленная тем, что на каждом участке изменения аргумента значения правых частей дифференциальных уравнений принимают постоянными, равными их значениям в начале этого участка, хотя, как правило, эти правые части на этом участке изменяются.

Применение формулы Рунге требует значительных затрат времени для повторного просчета. Непосредственно для приведенных выше схем не удастся построить алгоритм, подобный используемому в программах для вычисления интегралов.

Обычно полагается, что замена разложения по Тейлору на другой вид полиномиального приближения приводит к тем же самым ограничениям по точности, что и в методе Рунге-Кутты.

Однако нами было установлено, что применительно к гладким функциям правой части (которые всегда имеют место в случае выполнения кинетического закона действующих масс) использование схемы интегрирования по узлам с использованием аппроксимации Гаусса-Лежандра дает на порядки более высокую точность, чем методы Рунге-Кутты с тем же шагом и тем же порядком полинома.

На сегодняшний день стандартом *de facto* для численного решения жестких систем дифференциальных уравнений считается метод Гира, который относится к классу многошаговых методов, применяющих формулы дифференцирования назад (ФДН). Иногда такой прием называют еще методом «предиктор - корректор».

Для реализации метода Гира можно, например, выполнить следующую последовательность операций. Одним из методов Рунге-Кутты получим решения  $y_1, y_2, y_3$  задачи Коши

$$y' = f(x, y), \quad y(x_0) = y_0 \quad (7)$$

в точках  $x_1, x_2, x_3$ . В окрестности узлов  $x_0, \dots, x_4$  искомое решение  $y(x)$  приближенно заменим интерполяционным полиномом Ньютона четвертой степени, аналогичным ряду Тейлора

$$y(x) = y_0 + y_{01}(x - x_0) + y_{012}(x - x_0)(x - x_1) + y_{0123}(x - x_0)(x - x_1)(x - x_2) + y_{01234}(x - x_0)(x - x_1)(x - x_2)(x - x_3), \quad (8)$$

где  $y_{01}, \dots, y_{01234} \sim$  разделенные разности первого - четвертого порядков.

$$f(x_0; x_1) = \frac{f(x_1) - f(x_0)}{x_1 - x_0};$$

$$f(x_0; x_1; x_2) = \frac{f(x_1; x_2) - f(x_0; x_1)}{x_2 - x_0} = \frac{\frac{f(x_2) - f(x_1)}{x_2 - x_1} - \frac{f(x_1) - f(x_0)}{x_1 - x_0}}{x_2 - x_0};$$

$$f(x_0; x_1; \dots; x_{n-1}; x_n) = \frac{f(x_1; \dots; x_{n-1}; x_n) - f(x_0; x_1; \dots; x_{n-1})}{x_n - x_0}.$$

Левую часть уравнения (7) приближенно находят дифференцированием по  $x$  полинома (8)

$$y'(x) = y_{01} + y_{012}(x - x_0 + x - x_1) + y_{0123}[(x - x_0)(x - x_1) + (x - x_0)(x - x_2) + (x - x_1)(x - x_2)] + y_{01234}[(x - x_0)(x - x_1)(x - x_2)(x - x_3) + (x - x_1)(x - x_2)(x - x_3)]. \quad (9)$$

Разделенные разности для равноотстоящих узлов при шаге  $h$  выражаются через узловые значения аппроксимируемой функции

$$\begin{aligned} y_{01} &= (y_1 - y_0)/h, \\ y_{012} &= (y_2 - 2y_1 + y_0)/(2h^2), \\ y_{0123} &= (y_3 - 3y_2 + 3y_1 - y_0)/(6h^3), \\ y_{01234} &= (y_4 - 4y_3 + 6y_2 - 4y_1 - y_0)/(24h^4), \end{aligned} \quad (10)$$

где  $h = x_{i+1} - x_i$ .

Полагая в выражении для производной (9) значение аргумента  $x = x_4$  и учитывая значения разделенных разностей (10), получим

$$y'(x_4) = (3y_0 - 16y_1 + 36y_2 - 48y_3 + 25y_4) / (12h). \quad (11)$$

С другой стороны, уравнение (7) при  $x = x_4$  принимает вид

$$y'(x_4) = f(x_4, y_4) \quad (12)$$

Приравняем правые части соотношений (11), (12) и найдем

$$y_4 = [3(4hf(x_4, y_4) - y_0) + 16y_1 - 36y_2 + 48y_3] / 25. \quad (13)$$

Формула (13) представляет собой неявную схему Гира четвертого порядка для решения задачи Коши. Изменяя количество узлов  $x_i$ , можно аналогичным способом получить формулы Гира как более низких, так и более высоких порядков.

Неявные алгоритмы Гира наиболее эффективны для решения жестких

уравнений, особенностью которых является медленное изменение их решений при наличии быстро затухающих возмущений.

Жесткими уравнениями моделируются переходные процессы в нелинейных схемах, и применение неявных методов ускоряет на несколько порядков время интегрирования по сравнению с явными методами.

Для нахождения значения  $y_4$  из уравнения (13) можно применить метод простых итераций, однако для реализации достоинств неявного метода в отношении выбора шага при интегрировании жестких уравнений рекомендуется использовать метод Ньютона. Для любого из выбранных методов требуется знать начальное приближенное к искомой величине  $y_4$ .

Полагая в выражении для производной (9) значение аргумента  $x = x_3$ , будем иметь

$$y'(x_3) = \frac{-y_0 + 6y_1 - 18y_2 + 3y_4}{12h}. \quad (14)$$

Приравняв правые части исходного уравнения (7) при  $x = x_3$  и выражения (14), получим схему прогноза, с помощью которой можно найти начальное приближение для решения уравнения (13)

$$y_4 = 4hf(x_3, y_3) + (y_0 - 10y_3)/3 - 2y_1 + 6y_2. \quad (15)$$

В нашей программе реализован простейший вариант метода Гира четвертого порядка для системы ОДУ с одной коррекцией прогнозируемого значения. Нетрудно усовершенствовать (и усложнить) алгоритм, воспользовавшись подпрограммой метода Ньютона.

К очевидным недостаткам методов ФДН (как, впрочем, и других многошаговых) является необходимость разгонного участка и трудности при автоматическом выборе шага. Существует эквивалентная форма ФДН-методов, это методы Нордсика, в которой эти недостатки отчасти преодолены. В некоторых источниках они не выделяются в самостоятельный класс методов и называются ФДН-методами в представлении Нордсика.

Метод Гира в представлении Нордсика оказывается самостоятельно стартовым. Многозначный вариант метода Гира обладает переменным порядком аппроксимации: стартуя как метод первого порядка, по завершении разгонного участка метод стремится к максимально возможному для данной формулы порядку. При этом практическая реализация алгоритмов, основанных на методе Гира, всегда довольно сложна, не наглядна и требует значительного опыта и временных затрат при отладке и настройке, не гарантируя при этом отсутствия «проскакивания» важных событий на интегрируемой траектории. Метод Гира в реализации нашего пакета также не допускает никаких модификаций и наложения логических условий на параметры и константы системы ОДУ. В то же время исключительная эффективность сокращения временных затрат при использовании этого метода позволяет ему занимать штатное место в арсенале используемых в пакете алгоритмов.

### ПРИМЕНЕНИЕ КВАДРАТУРНЫХ ФОРМУЛ ГАУССА ДЛЯ ЧИСЛЕННОГО РЕШЕНИЯ СИСТЕМ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ

Использованный алгоритм относится к способам численного решения систем обыкновенных дифференциальных уравнений (СОДУ) химической кинетики и других аналогичных динамических систем. Техническим результатом является снижение временных затрат, необходимых для решения СОДУ при одновременном повышении точности без дополнительного дробления шага интегрирования. Также исключаются временные затраты на вычисление и подбор шага. Это достигается за счет замены аппроксимации разложения функции в степенной ряд Тейлора на точное ее решение, причем не на всем числовом отрезке, а в ограниченном наборе точек – с использованием квадратурной формулы Гаусса-Лежандра. Этот алгоритм также не требует вычисления Якобиана, что, безусловно, заметно повышает его быстродействие, но снижает его применимость к жестким системам. В то же время этот способ позволяет неограниченное сочетание с алгебраическими и логическими формулами на любой стадии, для любого компонента или константы, что делает его максимально пригодным для моделирования гибридных систем.

Подход использует общеизвестный факт, что все без исключения дифференциальные уравнения химической кинетики основываются на законе действующих масс, т.е. их правые части представляют собой полиномы степени не выше третьей. По этой причине совершенно естественным методом получения численных решений таких систем является использование метода Гаусса-Лежандра. Такой так называемый коллокационный метод (т.е. дающий для задачи Коши с начальными условиями совпадение с «точным решением» не на всем числовом отрезке, а в ограниченном наборе точек) демонстрирует ряд преимуществ, например, перед якобы аналогичным приближением Рунге-Кутты с полиномами той же степени. Специфический полиномиальный вид правых частей приводит к существенному улучшению точности при идентичном шаге, что в итоге дает заметный выигрыш вычислительного времени.

От уже перечисленных недостатков других методов свободен предлагаемый метод, основанный на использовании квадратурной формулы Гаусса

$$\int_{-1}^1 f(x) dx \approx Q = \sum_{i=1}^n \gamma_i f(\xi_i) \quad (16)$$

с разложением по полиномам Лежандра

$$\int_{-1}^1 \left( \chi_m(x) \prod_{j=1}^n (x - \xi_j) \right) dx = \sum_{i=1}^n \gamma_i \chi_m(\xi_i) \prod_{j=1}^n (\xi_i - \xi_j) = 0. \quad (17)$$

Его практическая реализация оказывается несопоставимо более удачной, чем реализация методов семейства Рунге-Кутты или Гира-Нордсика. Она не требует сколько-нибудь сложных вычислений, не содержит циклов и условных переходов и может быть легко реализована даже в электронных таблицах Excel с использованием одних только встроенных формул (не прибегая к кодам на

VBA). Тем не менее, переход от простейшего варианта к более эффективному с автоматическим подбором и регулированием шага требует программной поддержки. Именно такой вариант и реализован в описываемом пакете.

Эмпирически было найдено, что увеличение порядка аппроксимирующего полинома выше 4-го в большинстве случаев не приводит к заметному дальнейшему улучшению качества работы численного интегратора, хотя для особо жестких систем можно рекомендовать использование полиномов Лежандра 6-го порядка.

Разработанные программные блоки позволяют в десятки раз сократить время и повысить точность расчетов при вычислении решений уравнений химической кинетики, описывающих превращения реагентов при моделировании химических, биологических, химико-технологических и многих других процессов.

При включении их в автоматизированную систему моделирования, они обеспечивают повышение точности расчетов параметров систем ОДУ, и, как следствие, позволяют при конструировании оптимизировать служебные характеристики технологического оборудования, что представляет интерес как для химиков-технологов, так и разработчиков химических установок.

Нами разработан и доведен до практической реализации в среде VBA-Excel способ численного решения системы дифференциальных уравнений, который состоит в том, что для выбранного шага интегрирования разбивают интервал на промежутки, определяемые значениями корней полинома Лежандра соответствующей степени. Прирост первообразной вычисляется элементарным суммированием по квадратурной формуле Гаусса (16). Для гладких функций правой части в задаче Коши этот метод дает результаты, сопоставимые с точностью, определяемой разрядностью машины, а в случае полиномов – дает абсолютно точное решение. Нелишне подчеркнуть, что для химико-кинетических и технологических приложений правая часть задачи Коши дается уравнениями закона действующих масс и всегда может быть абсолютно точно представлена в виде полиномов (а значит и абсолютно точно проинтегрирована).

В случае квадратурных формул Гаусса, в отличие от, например, разбиений Нордсика и других аналогичных, внутренний шаг разбиения интервала внутри выбранного шага интегрирования непостоянен. Иначе говоря, смысл квадратурных формул Гаусса состоит в том, чтобы при наименьшем возможном числе узлов точно интегрировать многочлены наивысшей возможной степени. Известно, что при  $n$  гауссовых узлах по полученной формуле можно точно интегрировать многочлены степени  $\leq 2n - 1$ . При этом не требуется на каждом шаге решать систему уравнений для определения весовых коэффициентов – она решена заранее. В результате при прочих равных условиях может быть получен многократный выигрыш по времени численного решения задачи при заданной точности, либо многопорядковый выигрыш в точности при тех же временных затратах.

### ТРЕБОВАНИЯ К ИНТЕРФЕЙСУ ПРОГРАММЫ

- Задание списка идентификаторов всех веществ, включая промежуточные, в удобной нотации.
- Задание начальных концентраций. В подавляющем большинстве случаев ненулевые значения имеют концентрации двух-трех исходных веществ. Для продуктов реакции и активных интермедиатов по умолчанию подставляются нулевые концентрации, что может быть отредактировано пользователем.
- Полная схема {механизм} взаимодействия реагентов составляется в стандартной для химической кинетики нотации {число частиц не может превышать трех, что вытекает из требований формальной кинетики}. Любые более сложные многочастичные взаимодействия должны конструироваться с помощью создания промежуточных комплексов.
- По умолчанию все введенные реакции имеют нулевые константы скорости. До начала расчета необходимо подставить справочные или предполагаемые значения. Удалить реакции с нулевыми скоростями никогда не поздно, однако в ряде случаев имеет смысл зарезервировать их с целью последующего усложнения и уточнения модели.

Весьма показательное сравнение точности и быстродействия различных алгоритмов на примере системы, допускающей точное решение в квадратурах.

Рассмотрим модельное ДУ:

$$y'' + 4y = \cos(3x), \quad \text{с начальными условиями}$$

$$y(0) = 0.8, \quad y'(0) = 2.0, \quad x \in [0.0, 3.0], \quad h = 0.1.$$

Оно легко сводимо к системе двух уравнений подстановкой  $z(x) = y'(x)$

$$\begin{cases} y'(x) = z(x) \\ z'(x) = \cos(3x) - 4y \end{cases} \quad \begin{cases} y(0) = 0.8 \\ z(0) = 2.0 \end{cases}$$

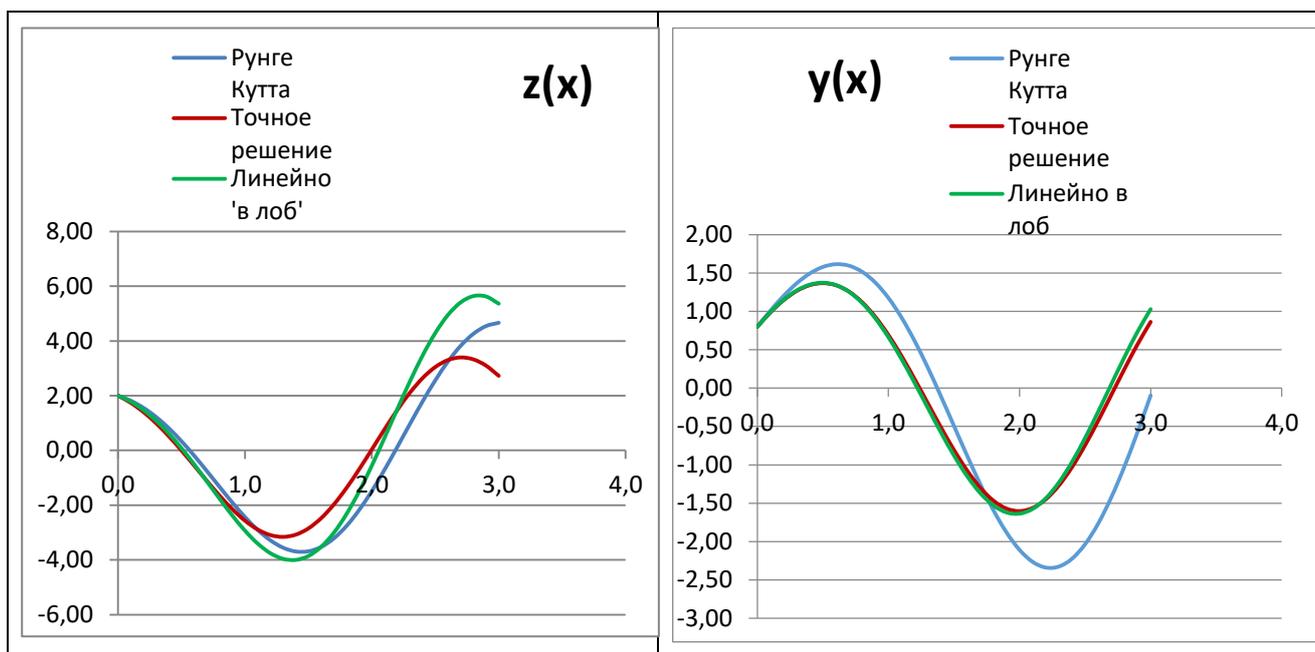
Эта система может быть численно проинтегрирована методом Рунге-Кутты 4-го порядка и методом с использованием квадратурной формулы Гаусса-Лежандра того же порядка.

Но уравнение было выбрано в качестве модельного, поскольку оно допускает точное решение в квадратурах, что может быть использовано в качестве репера для оценки отклонений, получающихся при использовании приближенных методов:

$$z = 2\sqrt{2} * \cos\left(2x + \frac{\pi}{4}\right) - 0,6 * \sin(3 * x)$$

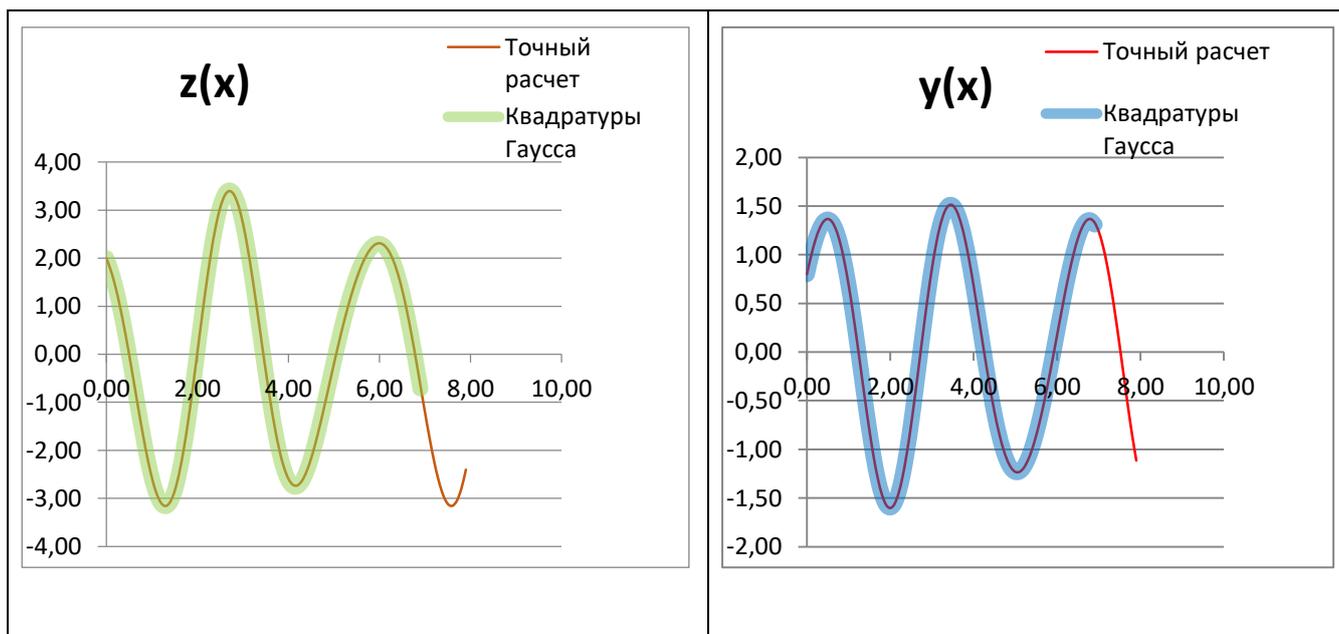
$$y = 0,2 * \cos(3 * x) + \sqrt{2} * \sin\left(2x + \frac{\pi}{4}\right).$$

В случае применения метода Рунге-Кутты после первого численного интегрирования наблюдается некоторое отклонение уже на первом периоде осцилляций (рис. 1). При этом отклонения для  $y(x)$  существенно больше, чем для  $z(x)$ , поскольку его вычисление включает в себя на одну процедуру приближенного интегрирования больше (ошибка накапливается).



**Рис. 1.** Графики решений модельной системы уравнений, полученные различными способами (см. пояснения в тексте).

От этого недостатка практически свободен алгоритм, основанный на использовании квадратурных формул Гаусса-Лежандра, что демонстрируют графики на рис. 2.



**Рис. 2.** Графики точного решения модельной системы и расчета с применением квадратурной формулы Гаусса-Лежандра (см. пояснения в тексте).

### ПРИМЕНЕНИЕ ПАКЕТА ДЛЯ МОДЕЛИРОВАНИЯ СИСТЕМЫ ЛОТКИ-ВОЛЬТЕРРЫ

Одной из тестовых задач для выяснения возможностей пакета моделирования является система Лотки-Вольтерры, в которой рассматривается

динамика популяций в замкнутой системе с пищевой цепочкой «питательная среда – жертва – хищник». Подразумевается закрытый ареал, в котором обитают два вида – травоядные («жертвы») и хищники. Предполагается, что животные не мигрируют (нет ни ухода за пределы ареала, ни прихода извне), и что еды для травоядных животных имеется с избытком. Тогда уравнение изменения количества жертв (без учета хищников) принимает вид:

$$\frac{dx}{dt} = \alpha x,$$

где  $\alpha$  – коэффициент рождаемости жертв,  $x$  – величина популяции жертв,  $\frac{dx}{dt}$  – скорость прироста популяции жертв.

Пока хищники не охотятся, они вымирают, следовательно, уравнение для численности хищников (без учета численности жертв) принимает вид:

$$\frac{dy}{dt} = -\gamma y,$$

где  $\gamma$  – коэффициент убыли хищников,  $y$  – величина популяции хищников,  $\frac{dy}{dt}$  – скорость прироста популяции хищников.

При встречах хищников и жертв (частота которых прямо пропорциональна величине  $xy$  происходит убийство жертв с коэффициентом  $\beta$ , сытые хищники способны к воспроизводству с коэффициентом  $\delta$ . С учетом этого, система уравнений модели такова:

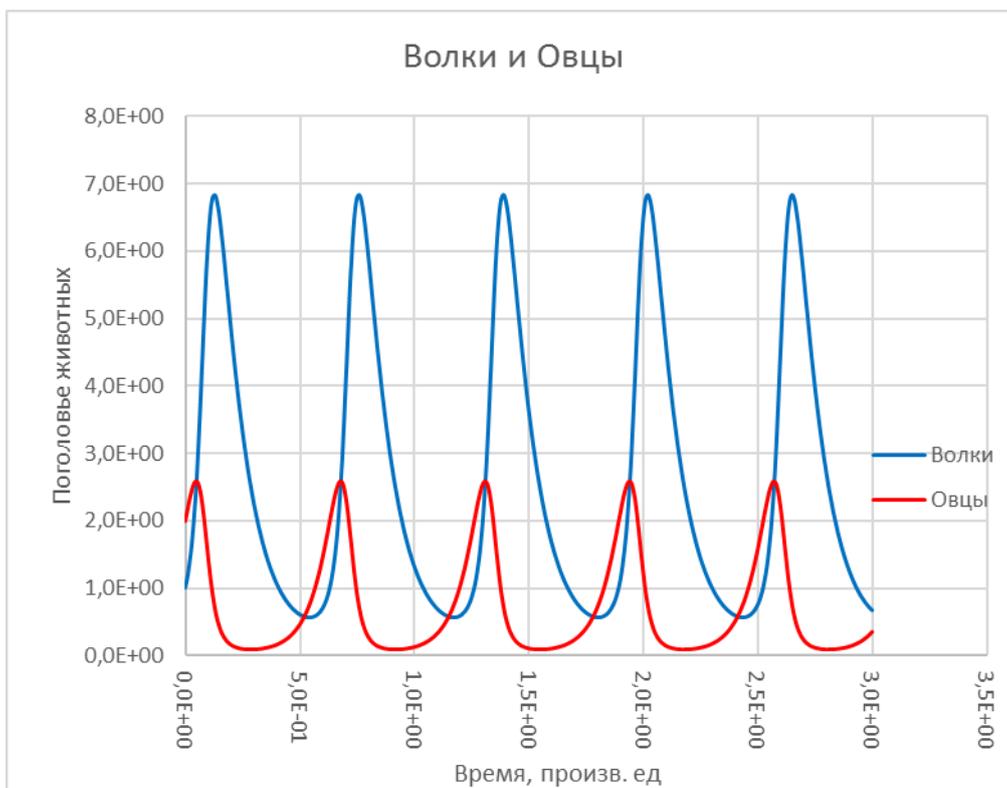
$$\begin{cases} \frac{dx}{dt} = \alpha x - \beta xy = x(\alpha - \beta y) \\ \frac{dy}{dt} = -\gamma y + \delta xy. \end{cases}$$

Созданная надстройка VBA–Excel избавляет от необходимости выписывать уравнения «руками» (хотя такая возможность тоже имеется) и позволяет просто рисовать пищевые цепочки:

Реагент#	Реагент#	Реагент#	Продукт#	Продукт#	Продукт#	Константа				
1	2	3	1	2	3					
1	Овцы		-->	Овцы	+	Овцы	1,50E+01			
2	Овцы	+	Волки	-->	Волки		6,00E+00			
3	Волки		-->				9,00E+00			
4	Волки	+	Овцы	-->	Волки	+	Волки	+	Овцы	1,20E+01

Решением приведенной системы уравнений является колебательная динамика по каждой из популяций со сдвигом по фазе на четверть периода: максимум численности овец соответствует наибольшей скорости воспроизводства волков, показанная на рис. 3.

Эта система дает предельный цикл, который может быть представлен либо в виде циклической фазовой диаграммы (количество вида против скорости изменения его поголовья), либо, что несколько предпочтительнее, – цикл в координатах «Волки – Овцы».



**Рис. 3.** Типовой график колебательного поведения в системе Лотки-Вольтерры (пищевая цепочка «хищник» - «жертва» - «растительный субстрат»).

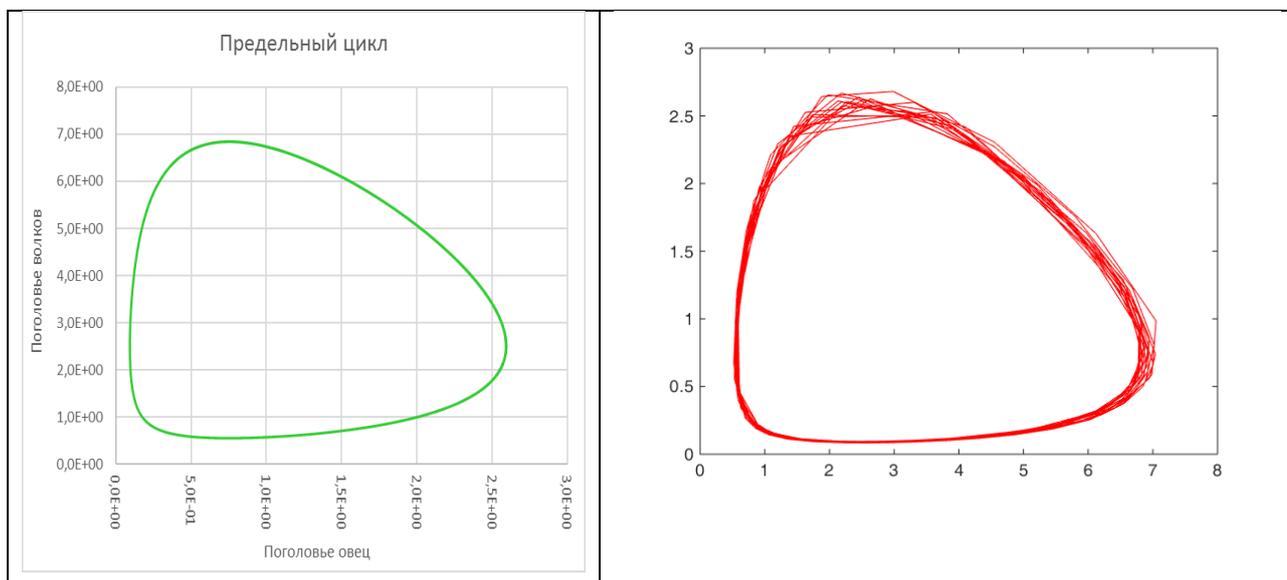
Приведенные диаграммы получены в упомянутой надстройке, отчего может возникнуть ложное ощущение легкости численного моделирования. На самом деле, это далеко не так. На рис. 4 справа представлен лучший результат, который путем длительных и изнурительных подборов удалось получить в лучшем (и самом сложном и медленном) сольвере популярного пакета Matlab ODE113.

Результаты моделирования в более простых интеграторах (типа ODE23 или ODE45) дают еще менее удовлетворительные результаты. Всюду имеется «распухание» фазовой траектории, на которое в свое время обращал внимание еще А.Н. Тихонов. Были протестированы все без исключения «решатели» из Матлаба и из нашей предыдущей версии программы «Кинетика», и ни один из них конкуренции с новейшим алгоритмом не выдержал.

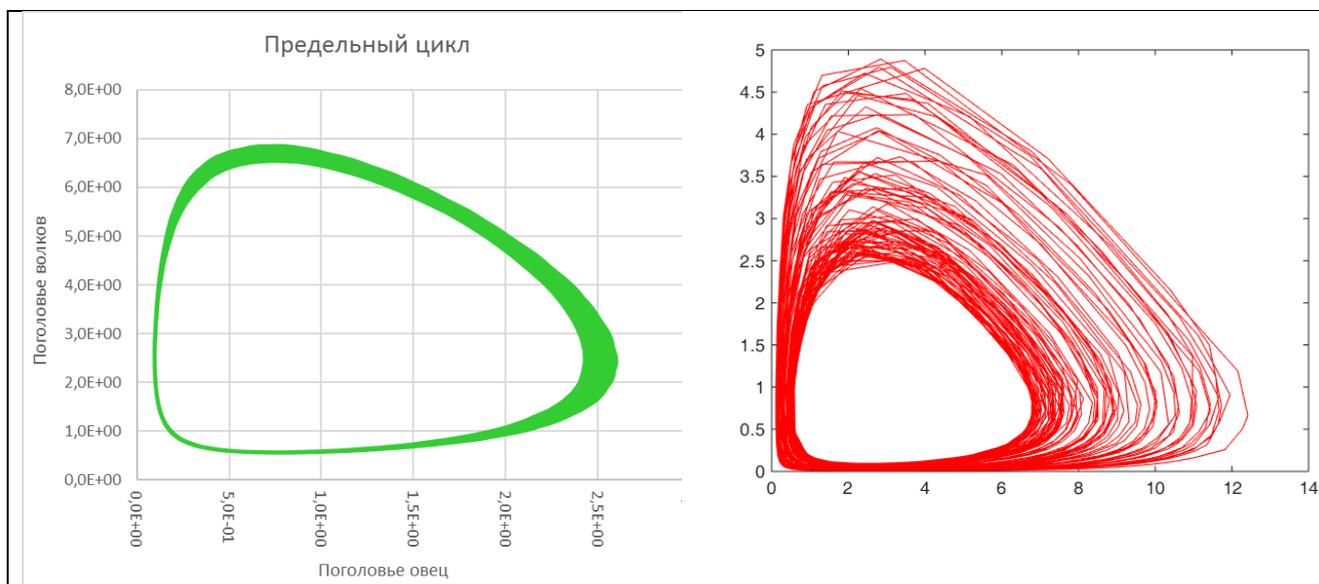
Разумеется, при увеличении интервала интегрирования степень «распухания» растет. На рис. 5 показаны фазовые диаграммы предельного цикла при интегрировании до 60 секунд (100 циклов) на SODE113.

Эмпирически было установлено, что с ростом длительности интервала ошибка (расцветание) накапливается и растет как экспонента по времени.

Но созданная надстройка не только дает более высокую точность при выбранном шаге интегрирования или позволяет использовать более крупный шаг при избранной точности, но и позволяет усложнять задачу добавлением событий, которые приводят к изменению самого вида исходной системы ОДУ.



**Рис. 4.** Фазовые траектории в системе Лотки-Вольтерры, 16 циклов. Слева расчет по алгоритму Гаусса-Лежандра, справа – расчет в Matlab.



**Рис. 5.** Слева расчет по алгоритму Гаусса-Лежандра – 1000 циклов, справа – расчет в Matlab, 100 циклов.

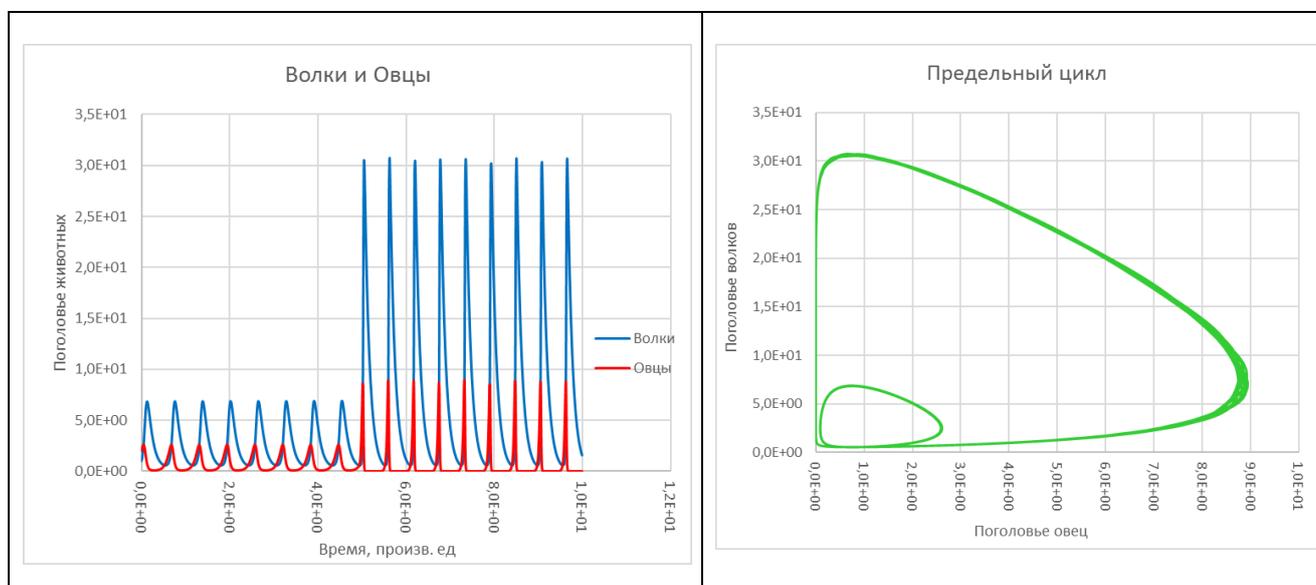
Рассмотрим демонстрационный вариант появления «события» все в той же системе Лотки-Вольтерры. В какой-то момент скачком изменяется кормовая база овец, например, появляется новый сорт кукурузы с более сочной силосной массой. В моделировании химического процесса это могло бы быть срабатывание на открытие одного из подводящих «клапанов»: константа скорости первой стадии в середине срока наблюдения скачкообразно возрастает втрое.

$$k_1 = 15 + 30 * \Theta(t - 4,95).$$

Здесь  $\Theta(t)$  – функция Хевисайда.

Тем самым мы ввели скачок в константу скорости размножения овец. Начиная с пятой единицы времени, овцы переходят на «усиленное питание» и их становится в три раза больше. Результат трудно назвать неожиданным: овец становится (в максимуме их численности) в три раза больше. Точно также ничего неожиданного в том, что на увеличенном поголовье овец увеличилось и поголовье волков.

На рис. 6 представлена совместная картинка динамики популяций.



**Рис. 6.** Слева динамика популяций в системе Лотки-Вольтерры с событием. Справа – фазовая диаграмма с переходом с «маленького колечка» на большое. При этом центр колебаний не сместился, просто увеличилась амплитуда.

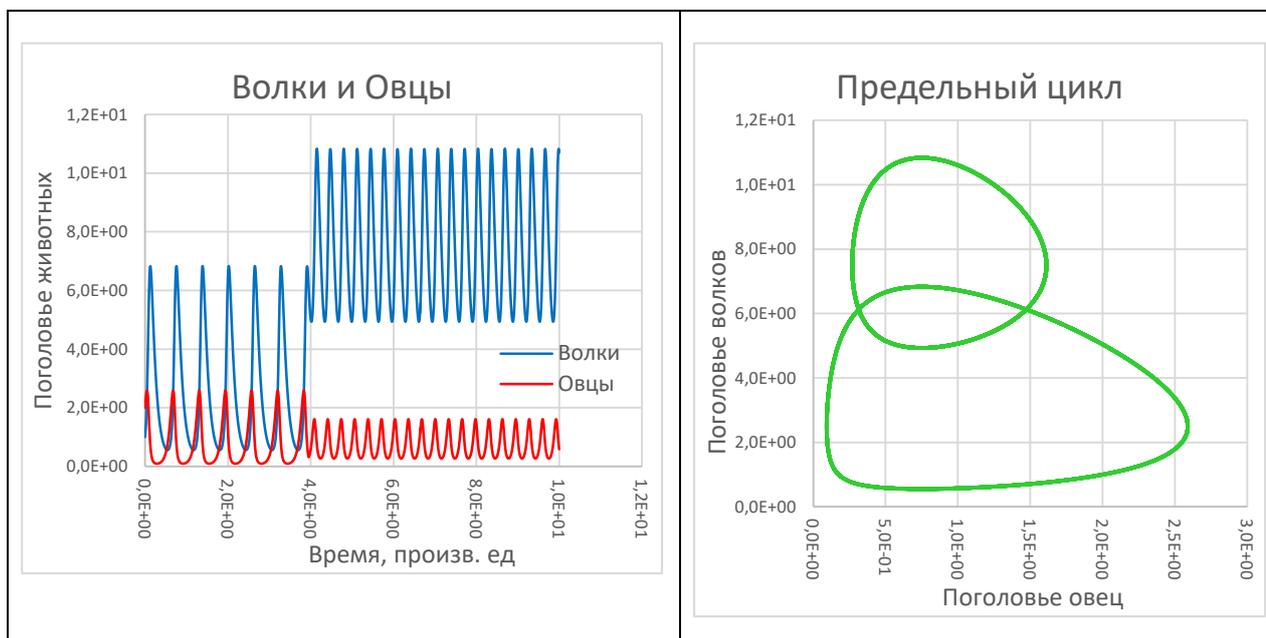
Помимо совершенно ожидаемого отклика системы на событие, надстройка позволяет исследовать разные режимы и «нащупать» совершенно неожиданный отклик динамической системы. На рис. 7 показано «включение» события появления нового сорта более сочной кукурузы на секунду раньше.

$$k_1 = 15 + 30 * \Theta(t - 3,95).$$

Результат оказывается совершенно неожиданным, предугадать его не удалось.

В отличие от первого случая, «подталкивание» в сторону увеличения поголовья овец сделано в фазе их максимальной убыли (наибольшая скорость гибели), т.е. получается как бы «толчок в противофазе». В результате количество овец не только не увеличивается, но даже уменьшается:  $k_1$  увеличивается с 15 до 45 в момент времени 3,95.

Не менее удивительно и изменение динамики популяции волков. Овец, якобы, стало меньше, а волков не просто прибавилось, но и падение их численности (минимумы на графике) стало не на всю амплитуду, а как бы наполовину.

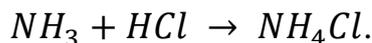


**Рис. 7.** Событие в системе Лотки-Вольтерры происходит в иной момент времени. Слева динамика популяций, справа – фазовая диаграмма. На этот раз центр колебаний сместился вверх, но при этом амплитуда колебаний даже уменьшилась.

## ДРУГИЕ ПРИМЕРЫ ПРИМЕНЕНИЯ РАЗРАБОТАННОГО ПАКЕТА ПРОГРАММ

### *Событие - фазовый переход*

Известная школьная реакция получения хлорида аммония («дым без огня»):

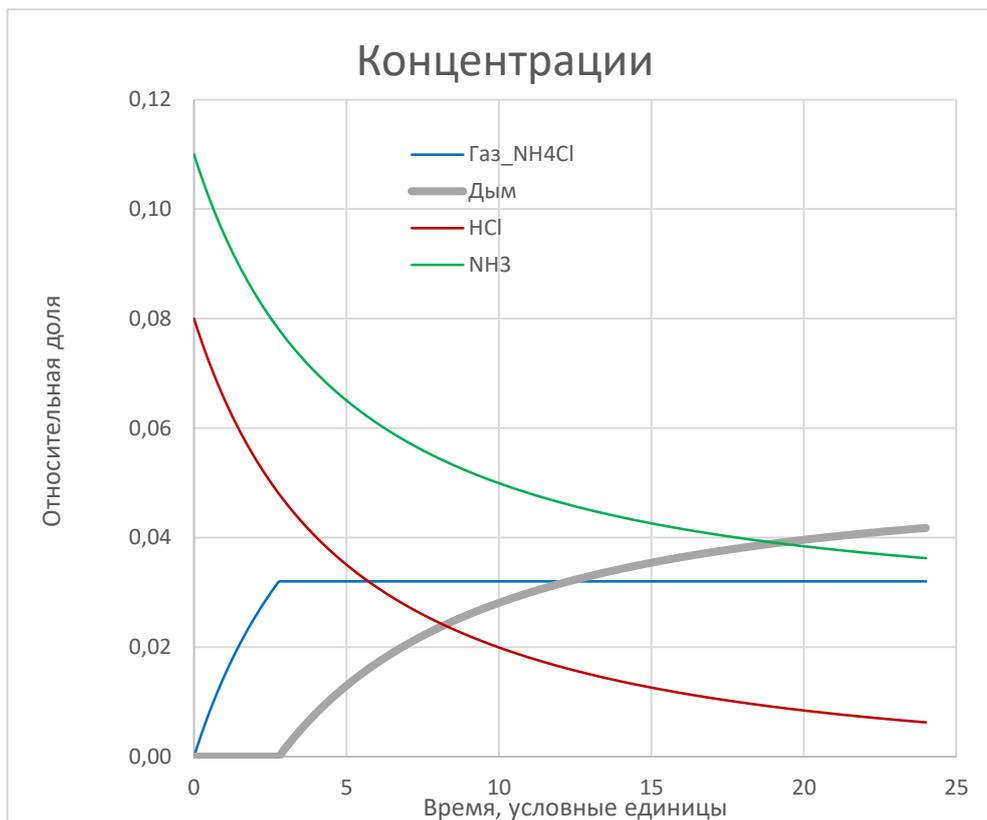


Появление твердой фазы (дыма) происходит при достижении предельного парциального давления хлорида аммония. На рис. 8 показаны модельные графики.

Модель требует использования всего пяти компонентов и трех реакций, из которых только одна соответствует реальному химическому превращению, а две являются виртуальными, в которых компонент превращается сам в себя. Фактически включение этих стадий в схему обусловлено лишь тем, что логика программного пакета подразумевает отображение на графике лишь тех компонентов, которые принимают участие хотя бы в каких-то реакциях.

Химическая часть модели дополнена двумя логическими условиями: для концентраций газообразного хлористого аммония и его аэрозольных частиц («дыма»), которые переходят один в другой при переходе граничного значения парциального давления газовой компоненты. Понятно, что столь простая модель считается практически мгновенно. Более интересным и даже неожиданным оказался тот факт, что моделирование успешно проходит с использованием любого из трех интеграторов пакета, включая метод Гира. Обычно сочетание алгоритма Гира с алгебраическими уравнениями приводит к появлению не вполне корректных результатов (в том смысле, что

алгебраические уравнения игнорируются и подавляются внутренней логикой интегратора). Но для данной модели все три интегратора дают идентичные результаты.



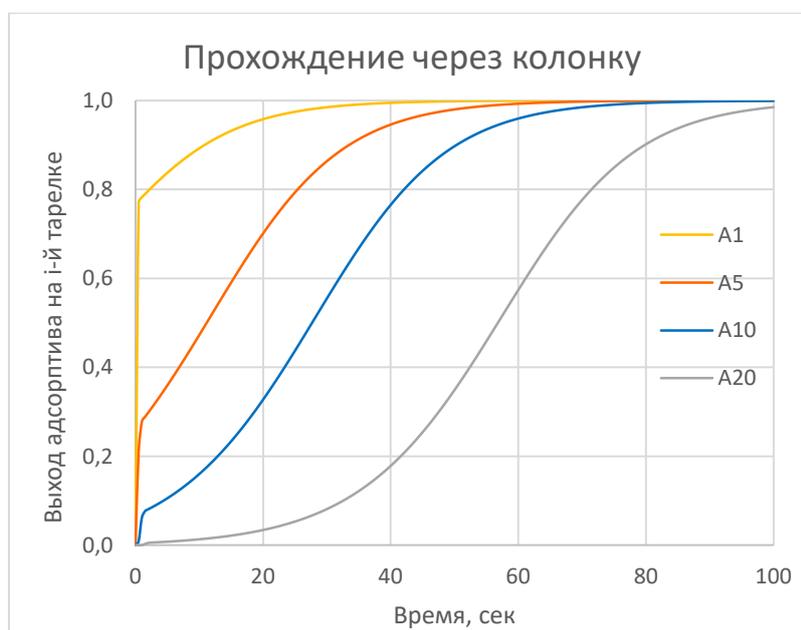
**Рис. 8.** Событие – выпадение аэрозоля при превышении предельного давления газообразного компонента.

### МОДЕЛИРОВАНИЕ АДСОРБЦИИ

Программный пакет имеет еще ряд возможностей, в качестве примера приведем моделирование адсорбции.

На рис. 9 показана динамика прохождения адсорбента через разные слои поглощающей колонки. Колонка разделена на 20 условных слоев («тарелок»). В моделирование включено по 3 компонента (A, S, AS) для каждого слоя. Модель включает 63 компонента, 61 элементарную стадию и считается за секунды.

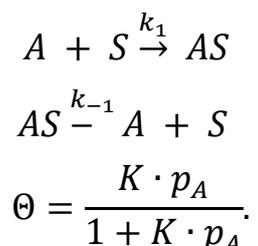
Обычно пакет, предназначенный для моделирования кинетических схем, дает лишь возможность получить информацию о временном ходе концентраций веществ и, возможно, об их установившихся равновесных значениях. В типичных случаях использование средства моделирования для получения зависимости равновесных значений от начальных концентраций требует многократного прогона вычислительной процедуры и составления итоговой таблицы равновесных значений концентрации продукта от концентраций исходных реагентов вручную.



**Рис. 9.** Моделирование продувания адсорбата через колонку с адсорбентом.

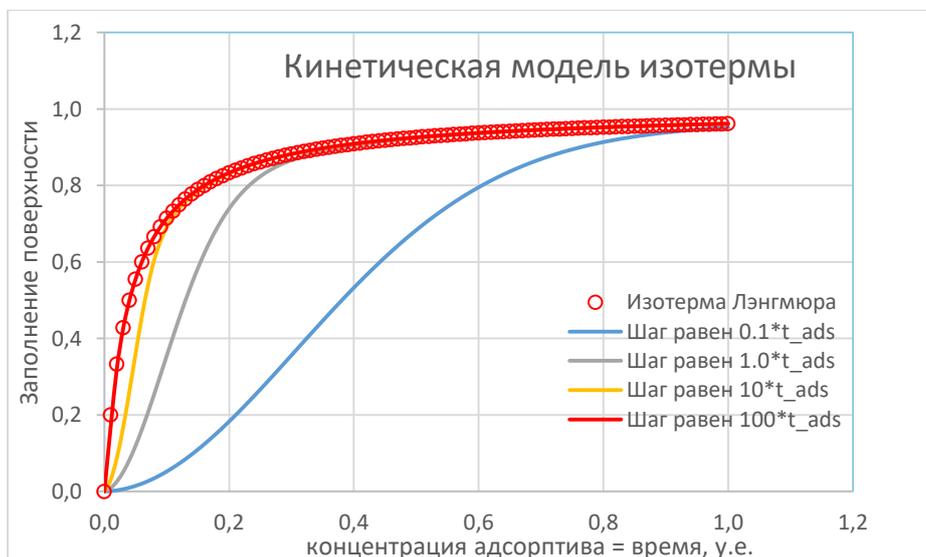
В нашем случае пакет легко использовать не только для вычисления динамики адсорбции, но и для построения графиков адсорбционных равновесий – изотерм адсорбции.

1) Вводим простейшую модель Лэнгмюра:



2) Вводим «развертку» по концентрации, т.е. добавляем к дифференциальным уравнениям алгебраическое  $[A] = t$ . Возможен также вариант чисто химического моделирования развертки: для этого достаточно добавить реакцию с единичной константой скорости, в которой вещество А является продуктом превращения из нерасходуемого источника. Правда, в последнем случае потребуются дополнительные ухищрения, чтобы компенсировать связывание части адсорбата с поверхностью адсорбента.

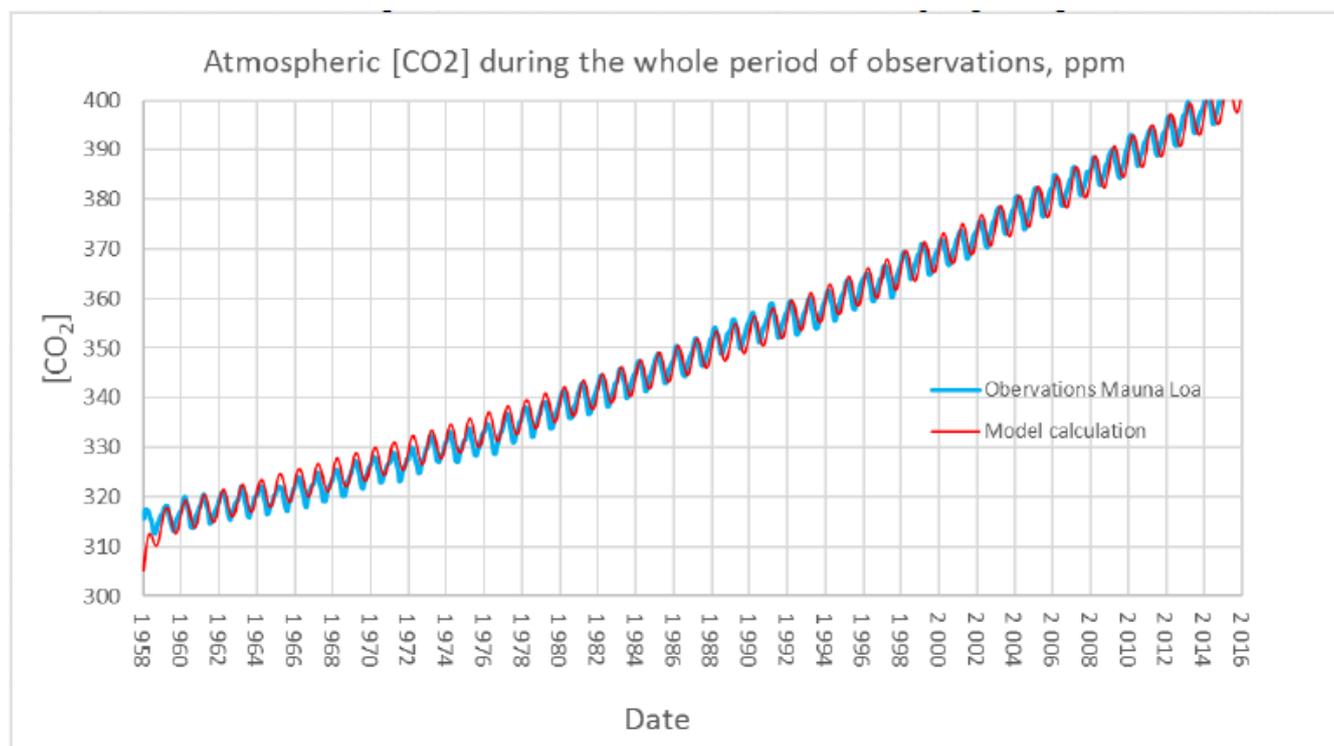
3) Подбираем достаточно быстрые константы сорбции и десорбции, чтобы равновесие устанавливалось быстрее, чем шаг по времени. На рис. 10 показано, при каких условиях получается аутентичное моделирование изотермы Лэнгмюра.



**Рис. 10.** При стократном превышении шага над характерным временем адсорбции получаем равновесную изотерму Лэнгмюра.

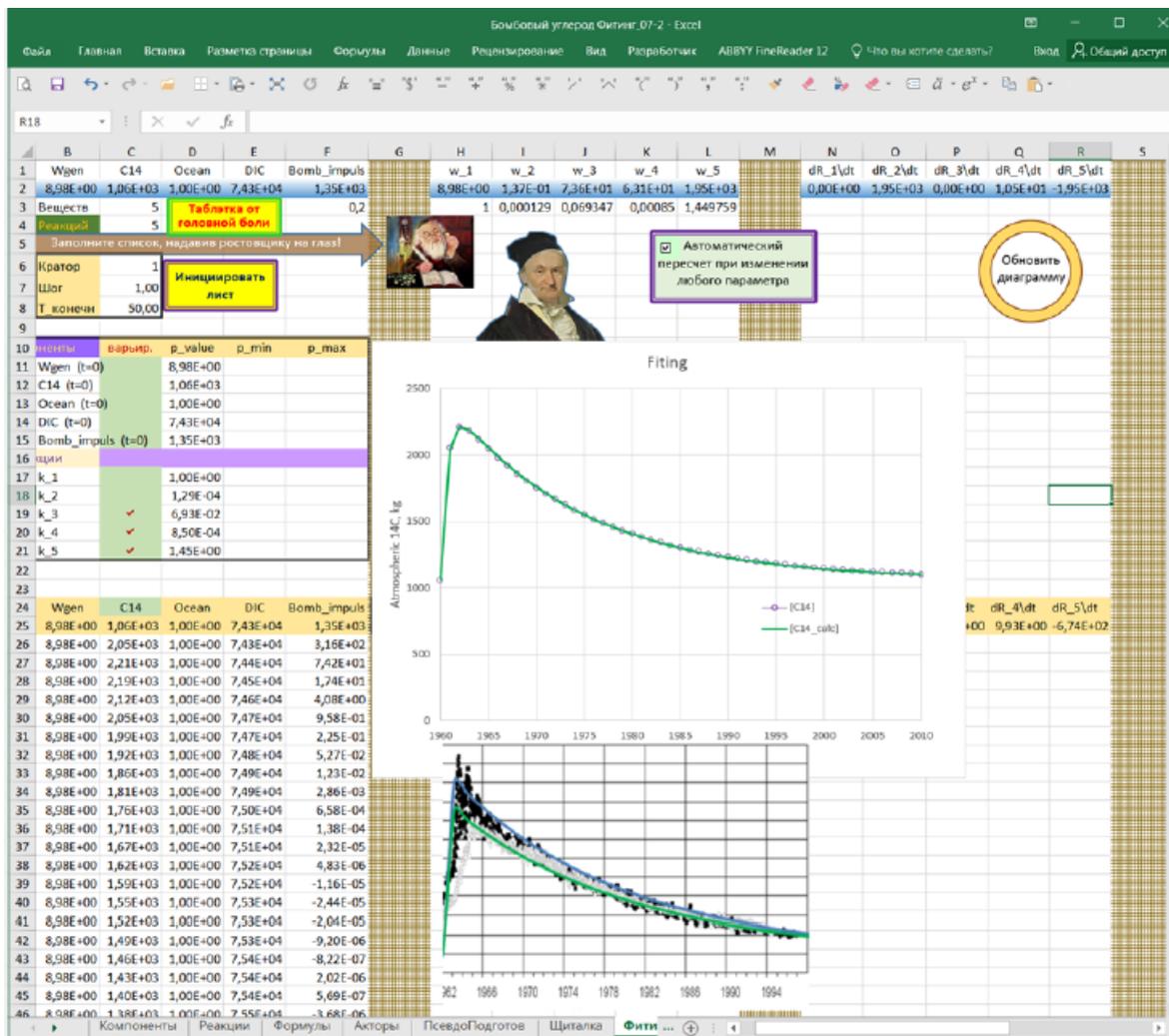
### МОДЕЛИРОВАНИЕ ОКЕАНИЧЕСКОГО КАРБОНАТНОГО БУФЕРА

С помощью разработанного нами пакета была рассчитана модель, учитывающая все известные кислотно-основные равновесия и равновесия газ – жидкость применительно к миграции углекислого газа. Она позволяет промоделировать наблюдаемый рост атмосферного  $\text{CO}_2$  и опровергнуть гипотезу о его антропогенной природе (рис. 11).



**Рис. 11.** Моделирование океанического карбонатного буфера. Голубой график – экспериментальные данные обсерватории Мауна-Лоа. Красная линия – моделирование в пакете «Кинетика».

В пакете также реализована возможность решения обратной задачи – определение констант скорости и иных числовых параметров модели путем наилучшего в смысле наименьших квадратов «подгона» расчетной кривой под экспериментальные данные. Получение выражений в явном виде для целевой функции и интегрирование системы ДАУ в квадратурах не требуется. На рис. 12 показан скриншот работы программы по решению обратной задачи методом Левенберга-Марквардта для подбора некоторых констант равновесия между атмосферным и океаническим содержанием радиоуглерода.



**Рис. 12.** Определение константы скорости спада концентрации атмосферного радиоуглерода  $^{14}\text{C}$  после бомбовых испытаний.

Это фрагмент моделирования более общей задачи динамики океанического карбонатного буфера, приведенной на предыдущем рисунке.

### ЗАКЛЮЧЕНИЕ

Таким образом, разработанный пакет программ «Кинетика» на основе VBA-Excel имеет следующие характеристики:

- В текущей конфигурации программная среда позволяет моделировать системы, содержащие до 100 индивидуальных компонентов между

- которыми возможно до 200 реакций. Эти цифры, при необходимости, могут быть увеличены в разы и даже на порядки.
- Обеспечивает наглядность счета, позволяет легко вводить произвольные математические и логические ограничения на значения констант, концентраций, скоростей процессов.
  - Содержит три хорошо отлаженных интегратора, что позволяет сравнивать результаты счета и иметь контроль над осмысленностью получаемых результатов.
  - Два интегратора из трех (за исключением метода Гира) позволяют моделирование гибридных систем путем добавления произвольных логических условий и алгебраических уравнений для любых констант скорости, концентраций или скоростей реакций.
  - Имеется встроенная утилита решения обратной задачи по алгоритму Левенберга-Марквардта.

*Работа выполнена в рамках Программы фундаментальных исследований ИХФ РАН № 46.15. Номер темы ФАНО 0082-2014-0005. Номер государственной регистрации ЦИТИС: АААА-А17-117091220076-4.*

#### Список литературы

1. Травин С.О., Быков А.А., Громов О.Б., Михеев П.И. // Сборник: XII всероссийское совещание по проблемам управления ВСПУ-2014. М.: Институт проблем управления им. В.А. Трапезникова РАН, 2014. С. 4237.
2. Инихов Д.Б., Колесов Ю.Б., Сениченков Ю.Б. // Компьютерные инструменты в образовании. 2012. № 6. С. 44.
3. Надеждин И.С., Горюнов А.Г., Дьяченко А.Н., Бахтадзе Н.Н., Быков А.А., Громов О.Б., Максимов Е.М., Михеев П.И., Травин С.О. Патент РФ по заявке № 2017120953/11(036214) от 14.06.2017.
4. Третья волна Вычислительной Гидродинамики. <https://habr.com/post/367463/> (дата обращения 30.03.2018).
5. Арутюнов А.В. // Сборник конференции «Ломоносов-2016». Секция «Вычислительная математика и кибернетика». М.: МГУ, МАКС Пресс, 2016. С. 56.
6. Самарский А.А., Гулин А.В. Численные методы. М.: Наука, 1989.
7. Мудров А.Е. Численные методы для ПЭВМ на языках Бейсик, Фортран Паскаль. Томск: МП «РАСКО», 1991. С. 182.
8. Новиков Е.А., Шорников Ю.В. Компьютерное моделирование жестких гибридных систем : монография. Новосибирск: Изд-во НГТУ, 2012.
9. Зельдович Я.Б., Мышкис А.Д. Элементы прикладной математики. М.: Наука, 1972. С. 261.

# DEVELOPMENT OF VBA-EXCEL SOFTWARE PACKAGE FOR SIMULATION CHEMICAL KINETICS TASKS

*S. O. Travin<sup>1\*</sup> and O. B. Gromov<sup>2</sup>*

<sup>1</sup>Semenov Institute of Chemical Physics, Russian Academy of Sciences, Moscow, Russia,  
\*e-mail: TravinSO@yandex.ru

<sup>2</sup>JSC “VNIINM” A.A. Bochvar High-Technology Scientific Research Institute for Inorganic Materials, Moscow, Russia

Received April 1, 2018

**Abstract** – A software package has been developed for solving direct and inverse problems of chemical kinetics. The package is implemented as a VBA-Excel add-in and can be easily integrated with any other applications. The package includes three integrators of different levels of complexity, which taken together allow modeling dynamics of virtually any chemical and biological system, including hybrid ones, i.e. containing events that change the model structure itself. The package is characterized by a user-friendly interface and ease of use.

*Keywords:* Cauchy problem, direct and inverse chemical kinetics problem, Gaussian quadrature.