

ПОДХОД К ОРГАНИЗАЦИИ РЕКУРРЕНТНО-ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛЕНИЙ В СИСТЕМЕ АВТОМАТИЗИРОВАННОЙ СТРУКТУРНО-ПАРАМЕТРИЧЕСКОЙ ИДЕНТИФИКАЦИИ

А.В. Павлов

Международный научно-учебный центр информационных технологий и систем НАН Украины и МОН Украины

Работа нацелена на создание многопоточного программного приложения для решения прикладных задач прогнозирования, экстраполяции и аппроксимации на основе рекуррентных итерационных алгоритмов метода группового учёта аргументов. Предложен подход к организации рекуррентно-параллельных вычислений, позволяющий интегрировать различные методы и алгоритмы в современные программные приложения, что даёт возможность использовать первые широким кругом пользователей. Эффективность приложения продемонстрирована на примере решения задачи прогнозирования инвестиционной безопасности Украины. Средняя абсолютная ошибка в процентах полученных моделей на независимой выборке наблюдений укладывается в интервал $[-7; +7]$. По моделям получен прогноз на следующий год и дана его экономическая интерпретация.

Ключевые слова: многопоточное распараллеливание, паттерны проектирования, метод группового учёта аргументов, рекуррентные вычисления, библиотеки Qt и TBB.

Роботу націлено на створення багатопотокового програмного додатку для розв'язання прикладних задач прогнозування, екстраполяції та апроксимації на основі рекуррентних ітераційних алгоритмів методу групового урахування аргументів. Запропоновано підхід до організації рекуррентно-паралельних обчислень, що дозволяє інтегрувати різні методи та алгоритми в сучасні програмні додатки, даючи можливість використовувати перші широким колом користувачів. Ефективність додатку продемонстровано на прикладі розв'язання задачі прогнозування інвестиційної безпеки України. Середня абсолютна помилка у відсотках одержаних моделей на незалежній вибірці спостережень вкладається в інтервал $[-7; +7]$. За моделями одержано прогноз на наступний рік і надано його економічну інтерпретацію.

Ключові слова: багатопотокове розпаралелювання, патерни проектування, метод групового урахування аргументів, рекуррентні обчислення, бібліотеки Qt і TBB.

ВВЕДЕНИЕ

Разработка и оптимизация методов и алгоритмов решения задач моделирования по статистическим данным безусловно являются базовыми направлениями в науке. Однако даже самые эффективные алгоритмы и технологии теряют свою ценность, если остаются запрограммированными модулями, с которыми может работать только программист-разработчик. Для того, чтобы технология или метод применялись на практике, их следует интегрировать в некоторую программную систему, имеющую удобный, понятный и практичный интерфейс пользователя. Именно такие

программные системы позволяют раскрыть настоящую ценность алгоритмов и методов, заложенных в программное обеспечение.

ОБЗОР РАБОТ

Термин «рекуррентно-параллельные вычисления» появился сравнительно недавно. Впервые он упоминается в работе [1] и определяется как объединение рекуррентных и параллельных видов операций в единую парадигму параллельно-рекуррентных вычислений для перехода на более высокий уровень эффективности программных средств. Далее рекуррентно-параллельные вычисления применялись для оптимизации комбинаторного алгоритма с различными схемами генерации структур моделей [2], где показана высокая степень эффективности распараллеливания (94.2%) и загрузки вычислителей (88.4%). В работе [3] предложен один из самых быстродействующих итерационных алгоритмов метода группового учёта аргументов (МГУА), применяющий рекуррентно-параллельные вычисления. Эффективность его распараллеливания составила 97% на физических ядрах процессора Intel Core i3.

Недостатком работ [1–3] является то, что предложенные рекуррентно-параллельные алгоритмы и методы существуют сами по себе, они представляют незначительную ценность для пользователя; с ними может работать только программист-разработчик.

Цель — разработка системы моделирования на основе рекуррентно-параллельного алгоритма [3] для повышения эффективности применения разработанных методов и алгоритмов для решения задач прогнозирования, экстраполяции и аппроксимации.

ПОСТАНОВКА ЗАДАЧИ

Достижение поставленной цели включает большой спектр задач, начиная от организации процесса импортирования данных в систему до анализа, выдачи и сохранения результатов моделирования.

Данная работа является логическим продолжением работы [4], где был заложен основной каркас системы автоматизированной структурно-параметрической идентификации (САСПИ). Здесь мы сосредоточимся на решении двух задач: интеграция графического интерфейса пользователя (ГИП) с пакетом классов, предназначенных для построения моделей, использующих рекуррентно-параллельные вычисления; разработка удобного и понятного ГИП и его элементов для ввода данных, анализа, представления и выдачи результатов моделирования.

ПАТТЕРНЫ ПРОЕКТИРОВАНИЯ РЕКУРРЕНТНО-ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛЕНИЙ В САСПИ

Интерфейс пользователя САСПИ построен на основе библиотеки Qt от Nokia [5]. Библиотека хорошо спроектирована, имеет множество

возможностей и предоставляет разработчику базовый набор элементов для построения систем любой сложности.

ГИП системы, реализованный без использования параллелизма, имеет недостаток, заключающийся в «замораживании» главного окна программы при выполнении каких-либо операций, вызываемых пользователем. Во избежание этого, все вычисления или любые другие задачи должны выполняться не в главном потоке, а в дополнительном.

Библиотека Qt имеет набор классов, служащих для создания и управления дополнительными потоками на высоком уровне. Главной задачей здесь является объединение двух механизмов параллелизма: механизм распараллеливания с помощью библиотеки TBB, выполняющий построение моделей [3] и механизм распараллеливания в библиотеке Qt, позволяющий приложению обрабатывать события пользователя во время выполнения вычислений в программе. Идея решения задачи следующая: каждая операция (включая вызов модуля построения моделей, создающий свои TBB-потоки), инициируемая пользователем в программе, будет выполняться в дополнительном Qt-потоке. Проиллюстрируем сказанное на рис. 1.

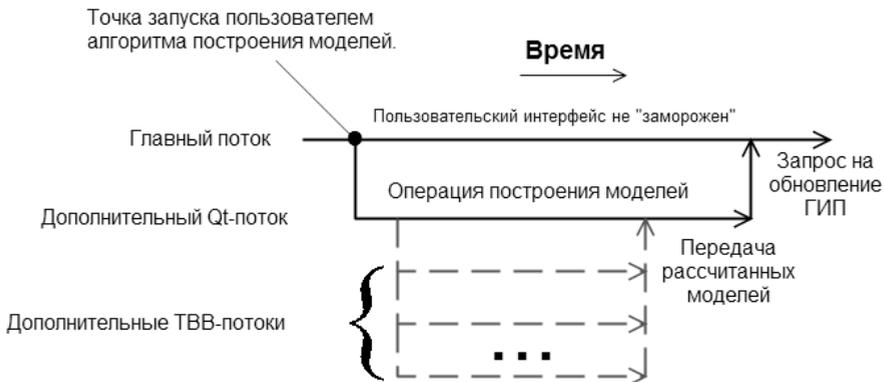


Рис. 1. Идея объединения двух механизмов параллелизма

Библиотека Qt спроектирована так, что классы, реализующие интерфейс пользователя, всегда находятся в главном потоке программы. Поэтому основным требованием, предъявляемым к паттерну проектирования, решающему данную задачу, является обновление ГИП в главном потоке, а не в дополнительном, где происходили вычисления. На рис. 2 представлен паттерн, удовлетворяющий предъявленному требованию.

Класс Thread предназначен для создания дополнительного Qt-потока и запуска задачи в этом потоке. Приведём реализацию метода run данного класса:

```
thread = new QThread;
task->moveToThread( thread );
QObject::connect( thread, SIGNAL( started() ), task,
SLOT( process() ) );
QObject::connect( thread, SIGNAL( finished() ), thread,
SLOT( deleteLater() ));
```

```

QObject::connect( task, SIGNAL( finished() ), thread,
SLOT( quit() ) );
QObject::connect( task, SIGNAL( finished() ), task,
SLOT( deleteLater() ) );
thread->start();

```

Для того чтобы какой-либо код мог быть вызван в параллельном Qt-потоке, он должен содержаться в слоте, поэтому любой класс-задача должен быть унаследован от QObject. Примерами таких классов-задач являются классы ParallelLoadData и ParallelRun. Согласно реализации метода Thread::run, эти классы должны иметь слот process(), который содержит код, запускаемый в параллельном потоке, а также сигнал finished(), говорящий об окончании выполнения параллельных операций.

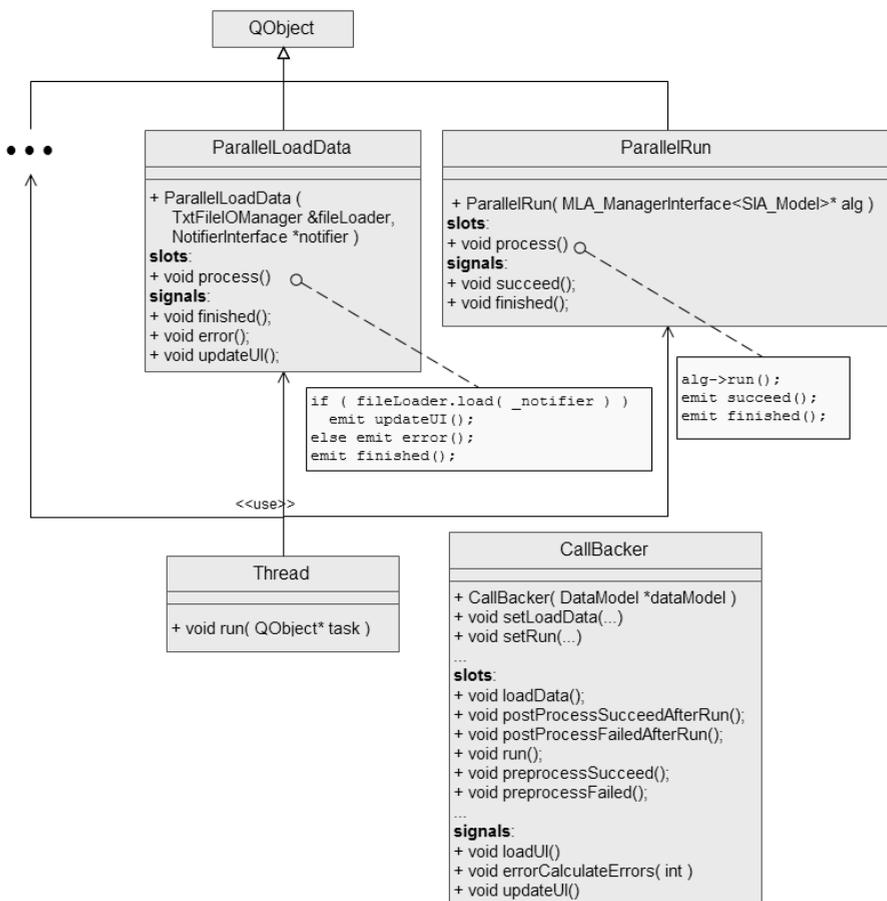


Рис. 2. Диаграмма классов параллельного вызова операций в САСПИ

Таким образом, классы ParallelLoadData и ParallelRun служат обёрткой для выполнения операций в параллельном потоке. Они делегируют операции загрузки данных и расчёта моделей классам TxtFileManager и MLA_ManagerInterface соответственно.

Дабы не усложнять класс `DataModel`, наследуя им класс `QObject`, а также логически отделить вызов `callback`-методов, был введён класс `CallBaker`. Его предназначение — обновлять данные модели `DataModel` с помощью «сигнал-слот» механизма после того, как были окончены вычисления какой-либо задачи в параллельном потоке. Следовательно, для каждой параллельной задачи `CallBaker` должен иметь: 1) метод установки параметров, которые будут использоваться при обновлении `DataModel`; 2) слот, который будет вызываться и обновлять `DataModel` после того, как параллельная задача была успешно/неуспешно выполнена.

Рассмотрим последовательность взаимодействия объектов системы при вызове операции построения моделей во времени. Процесс построения моделей состоит из трёх этапов: 1) предварительная обработка данных; 2) построение моделей; 3) расчёт ошибок моделей.

На рис. 3 приведена диаграмма последовательности операций для первого этапа. Метод `callBack_preprocSucceed()` конфигурирует алгоритм построения модели и запускает процесс её расчёта.

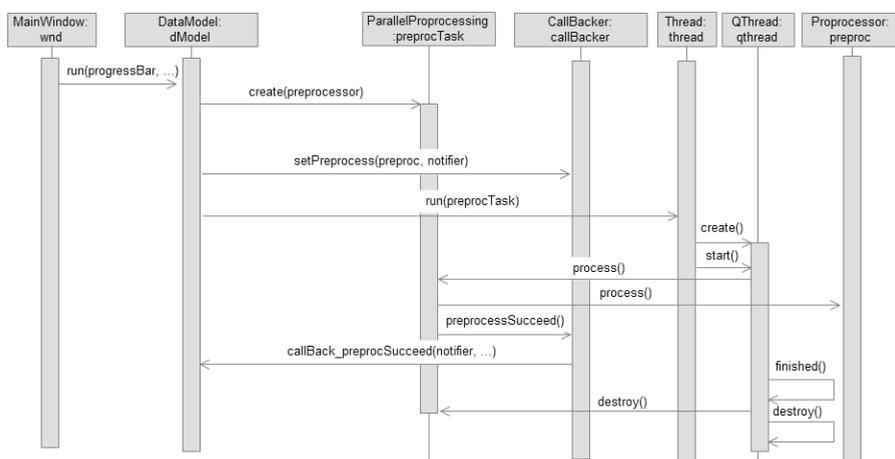


Рис. 3. Диаграмма последовательности операций для параллельной предварительной обработки данных

На рис. 4 представлена диаграмма последовательности операций при расчёте моделей в параллельных потоках. Метод `callBack_runSucceed()` удаляет алгоритм и обновляет объект модели данными построенными моделями. Метод `callBack_run()` удаляет фасад и препроцессор, и создаёт объекты для параллельного расчёта ошибок моделей (рис. 4).

На рис. 5 представлен завершающий этап процесса построения моделей — расчёт ошибок моделей. Метод `callback_postProcessSucceedAfterRun()` обновляет модель данных и вызывает метод `modelChanged()`, обновляющий ГИП.

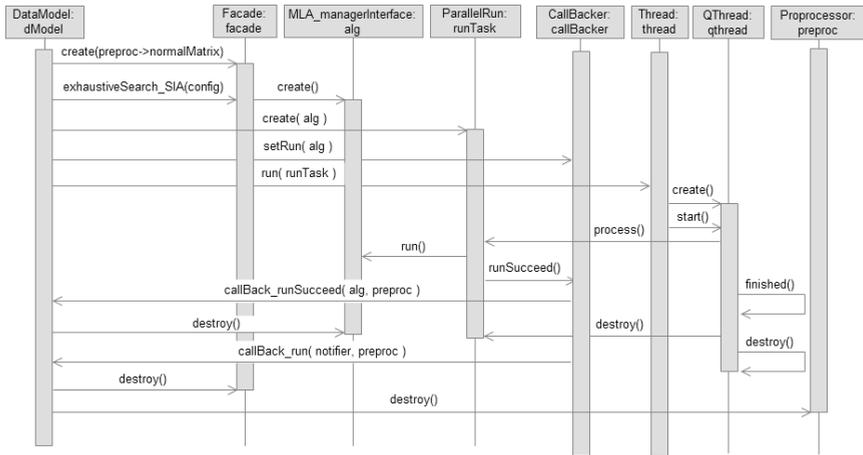


Рис. 4. Диаграмма последовательности операций для процесса расчёта моделей

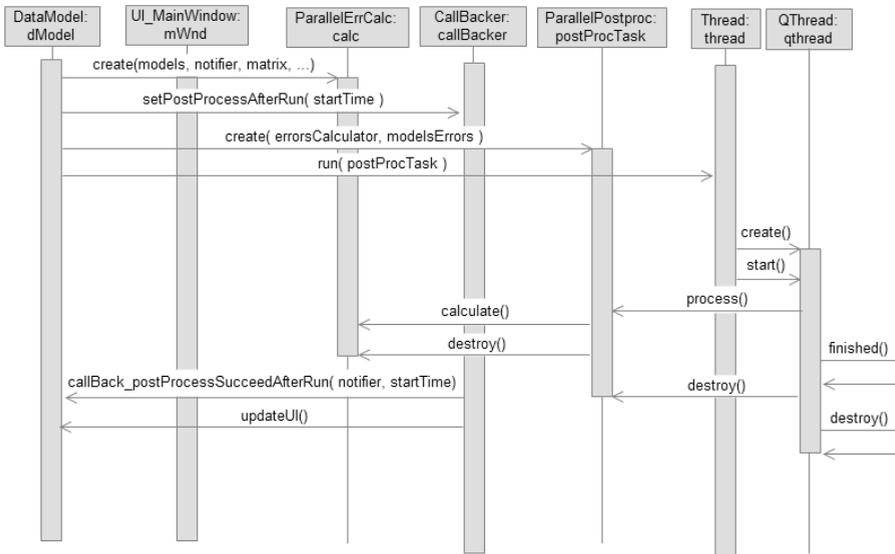


Рис. 5. Диаграмма последовательности операций для процесса расчёта ошибок моделей

Рассмотрим работу с ГИП САСПИ на примере решения задачи прогнозирования инвестиционной безопасности Украины.

ПРОГНОЗИРОВАНИЕ ИНВЕСТИЦИОННОЙ БЕЗОПАСНОСТИ УКРАИНЫ С ПОМОЩЬЮ САСПИ

Важным условием для устойчивого экономического развития страны служит активная и управляемая инвестиционная деятельность. Уровни инвестиционных возможностей, инвестиционной привлекательности и эффективности инвестиционных процессов — важные показатели развития национального хозяйства. Инвестиционная безопасность государства — это уровень соотношения между величинами инвестиций страны за границу и

полученными инвестициями, что удовлетворяет потребности внутренней экономики и поддерживает положительный платежный баланс государства.

В системе показателей инвестиционной безопасности Украины необходимо выделить следующие: x_1 — валовой внутренний продукт, млрд. грн.; x_2 — объёмы инвестиций в основной капитал в фактических ценах, млрд. грн.; x_3 — реальный прирост/снижение инвестиций в основной капитал, процент от предыдущего года; x_4 — степень износа основных фондов, процент от предыдущего года; x_5 — чистый прирост прямых внешних инвестиций (ПВН), млн. дол. США; x_6 — чистый прирост ПВН, млрд. грн.; x_7 — часть иностранных инвестиций относительно объема инвестиций в основной капитал, процент; x_8 — общий объём инвестиций, млрд. грн.

Пренебрежение инвестиционной безопасностью может привести к катастрофическим последствиям: остановке отраслей народного хозяйства, банкротству предприятий и, в конечном итоге, к подрыву системы жизнеобеспечения нации вплоть до потери суверенитета. Поэтому важной научно-прикладной задачей является оценка системы показателей инвестиционной безопасности Украины в будущем.

Задача моделирования состоит в построении прогнозных моделей на один шаг вперёд для каждого из вышеперечисленных индикаторов. С целью получения наиболее точных результатов предполагается построение системы моделей вида:

$$\begin{cases} x_{1,t+1} = f(x_{1,t}, \dots, x_{1,t-l}, x_{2,t}, \dots, x_{2,t-l}, \dots, x_{8,t}, \dots, x_{8,t-l}) \\ \dots \\ x_{8,t+1} = f(x_{1,t}, \dots, x_{1,t-l}, x_{2,t}, \dots, x_{2,t-l}, \dots, x_{8,t}, \dots, x_{8,t-l}) \end{cases}, \quad (1)$$

где $x_i, i = \overline{1,8}$ — показатели, l — количество запаздываний, t — время.

Алгоритм построения моделей САСПИ строит модели линейные по параметрам и линейные/нелинейные по входным переменным. Функция f имеет вид:

$$f(z_1, \dots, z_m, \theta_1, \dots, \theta_m) = \sum_{i=1}^m \theta_i z_i,$$

где в качестве z_i могут выступать как входные переменные x_i , так и их

нелинейные преобразования вида $\prod_{j=1}^k x_j$ (степень k одночлена задаётся

заранее), θ_i — параметры, определяемые по рекуррентному методу с использованием метода наименьших квадратов (МНК) [2]. В таблице 1 представлены исследуемые показатели.

Таблица 1.

Показатели инвестиционной безопасности Украины (1996-2012 гг.)

	1996	1997	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012
x_1	81.52	93.37	102.59	130.44	170.07	204.19	225.81	267.34	345.11	441.45	544.15	720.73	948.06	913.35	1082.57	1302.08	1408.89
x_2	12.56	12.4	13.96	17.55	23.63	32.57	37.18	51.01	75.71	93.1	125.25	188.49	233.08	151.78	171.09	238.17	263.73
x_3	-22	-8.8	-6.1	0.4	14.4	20.8	8.9	31.3	28	1.9	19	29.8	-2.6	-41.5	-0.6	22.4	8.3
x_4	40	38	40.4	42.3	43.7	45	47.8	48	49.3	49	51.5	52.6	61.2	60	74.9	74.9	74.9
x_5	541.3	625.4	747.1	471.1	593.2	680.3	916.51	1318.61	1696.31	8021.3	4810.8	8303.4	6234	4303.4	4681.2	4654.3	5100.1
x_6	0.99	1.16	1.83	1.95	3.23	3.65	4.89	7.03	9.02	41.11	24.29	41.93	32.84	33.53	37.15	37.04	40.75
x_7	7.89	9.39	13.11	11.09	12.02	10.09	11.62	12.11	10.65	30.63	16.25	18.2	12.35	18.09	17.84	13.47	13.38
x_8	13.55	13.56	15.79	19.5	26.86	36.23	42.07	58.04	84.74	134.2	149.55	230.42	265.92	185.31	208.24	275.26	304.48

Процесс моделирования. Загруженные данные отображаются в таблице «Содержимое файла» (рис. 6). Загруженные столбцы таблицы отображаются в списке «Переменные». Поскольку выборка совсем небольшая ($n = 17$), ограничимся количеством лагов, $l = 1$.

В большинстве случаев, линейная по переменным модель была не в состоянии дать достаточно точный прогноз, поэтому ко входным переменным, кроме исходных $x_i, i = \overline{1,8}$, были добавлены обратные их величины $1/x_i, i = \overline{1,8}$, а также одночлены второй, и в редких случаях третьей степени.

Поскольку стоит задача построения прогноза на один шаг вперед, экзамен (выборка D) составил последнее наблюдение. Оно будет использовано для проверки качества и выбора наиболее адекватной модели. Рабочую выборку C разделим на подвыборки обучения A и проверки B в соотношении 2/1 соответственно. Критерием отбора лучших моделей алгоритма будет нормированный внешний критерий регулярности NAR_B (NRSS проверки на рис. 6).

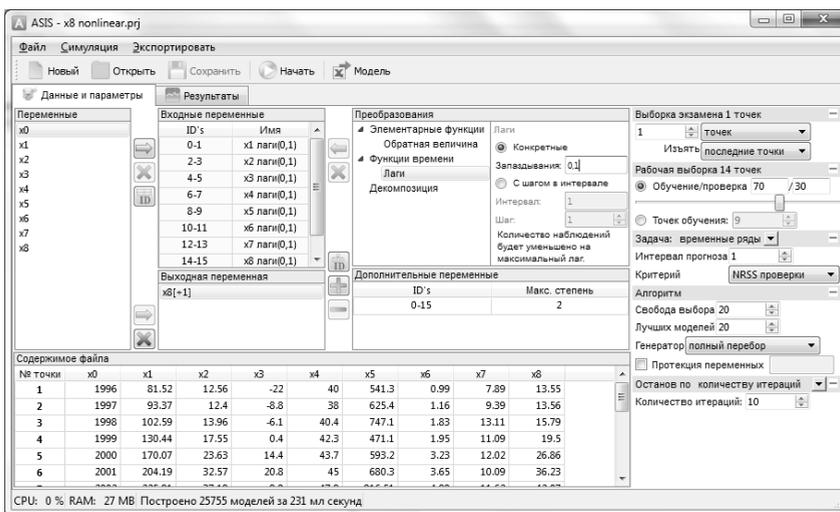


Рис. 6. Настройка параметров моделирования

Количество моделей, передаваемых от итерации к итерации — свобода выбора — равна 20 моделям; количество лучших моделей равно двадцати.

Остановку алгоритма будем осуществлять по количеству итераций, а лучшую из 20 моделей определим по следующим трём критериям: 1) точность на экзамене; 2) простота структуры модели; 3) величина коэффициентов при одночленах модели. Установим максимальное количество итераций алгоритма равным десяти.

Результаты моделирования. Результаты моделирования для показателя реального прирост/снижение инвестиций в основной капитал (x_3) представлены на рис. 7. Они отображаются как в графическом, так и числовом виде. Дополнительно рассчитываются остатки моделей и доверительный интервал, равный $[-2\sigma; +2\sigma]$.

Аналитический вид полученных моделей следующий:

$$\begin{aligned}
 x_{1,t+1} &= 1.29116x_{1,t} - 1.04917 \cdot 10^{-5} x_{8,t} x_{8,t-1} x_{8,t-1} - 12.2856, \\
 x_{2,t+1} &= 0.008213x_{5,t} + 4.41185x_{7,t-1} + 5.17354x_{4,t} - 0.0847269x_{3,t-1} - 240.877, \\
 x_{3,t+1} &= -624.88 \frac{1}{x_{7,t}} - 4.3247 \cdot 10^{-5} x_{5,t} x_{8,t-1} + 0.005263x_{7,t-1} x_{8,t} - \\
 &\quad - 0.0008958x_{3,t-1} x_{8,t} + 67.68 \\
 x_{4,t+1} &= 0.083198x_{1,t} + 0.00319x_{4,t} x_{8,t} + 35.2331, \\
 x_{5,t+1} &= -500580 \frac{1}{x_{1,t-1}} + 5601, \quad x_{7,t+1} = 0.0668215x_{7,t-1} + 13.2782 \\
 x_{6,t+1} &= 0.001549x_{4,t-1} x_{7,t} x_{7,t-1} + 4.84545 \cdot 10^{-8} x_{5,t} x_{8,t-1}^2 - \\
 &\quad - 2.06235 \frac{1}{x_{3,t-1}} - 2.5223 \cdot 10^{-5} x_{4,t} x_{4,t-1} x_{7,t} + 1.5618 \\
 x_{8,t+1} &= 0.0408775x_{7,t} x_{8,t-1} + 0.11994x_{3,t} x_{7,t-1} + 0.0336721x_4^2 - 42.6608.
 \end{aligned}$$

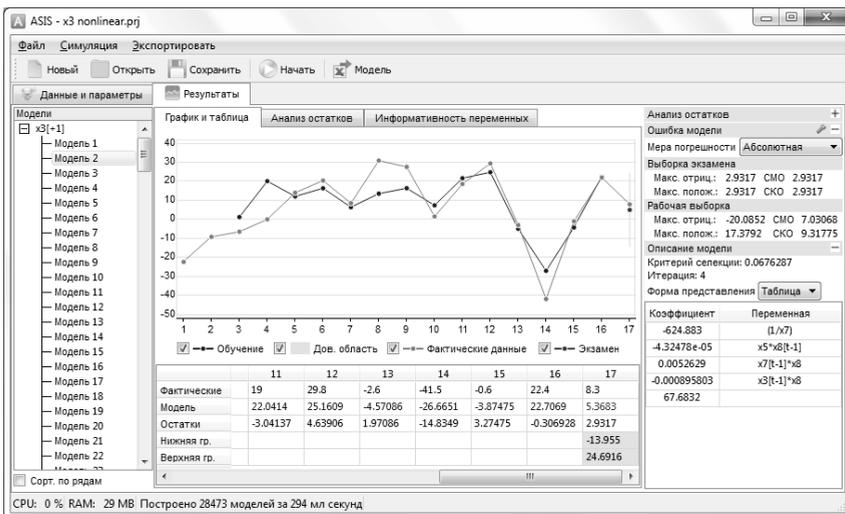


Рис. 7. Результаты моделирования для показателя «Реальный прирост/снижение инвестиций в основной капитал»

Расчёт каждой модели занял меньше секунды. САСПИ также рассчитывает ряд ошибок модели. Значения средней абсолютной ошибки в процентах (САОП) и средней абсолютной ошибки (САО) представлены в таблице 2.

Таблица 2.

Точность моделей на рабочей и экзаменационной выборках

		x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
Рабочая выборка	САОП	4.92	26.52	338	2.39	71.4	74.7	21.78	21.41
	САО	24.11	17.95	7.03	1.29	38.47	4.53	3.53	14.61
Выборка экзамена	САОП	6.58	0.047	35.32	3.86	0.754	0.024	5.96	0.911
	САО	92.76	0.124	2.93	2.89	1408.9	0.01	0.798	2.77

Ошибки прогноза полученных моделей на экзамене укладывается в интервал $[-7\%; +7\%]$ почти для всех моделей (кроме модели для x_3) что говорит об отличных прогностических способностях. Известно, что меру САОП некорректно использовать при малых значений показателя, что наблюдается для показателя x_3 . В таких случаях точность модели лучше оценить по САО или графику (рис. 7). Чистый прогноз по моделям представлен в таблице 3.

Таблица 3.

Прогноз на 2013 год по всем показателям

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
2012	1408.89	263.73	8.3	74.9	5100.1	40.75	13.38	304.48
2013	1538.84	246.03	-24.25	79.70	5216.5	39.21	14.17	309.7

По результатам прогнозирования предполагается увеличение валового внутреннего продукта, степени износа основных фондов, чистого прироста ПВН (дол. США), части иностранных инвестиций относительно объема инвестиций в основной капитал, общего объема инвестиций; уменьшении объемов инвестиций в основной капитал в фактических ценах, реального прироста/снижения инвестиций в основной капитал, чистого прироста ПВН в грн.

Выводы

Предложенный подход к организации рекуррентно-параллельных вычислений в системе автоматизированной структурно-параметрической идентификации (САСПИ) позволил интегрировать эффективные методы решения задач моделирования по экспериментальным данным в удобной, практичной и целостной системе, позволяющей пользователю разного уровня решать задачи прогнозирования, экстраполяции и аппроксимации с высокой эффективностью.

Работа системы продемонстрирована на примере решения задачи прогнозирования инвестиционной безопасности Украины. Полученные модели имеют погрешность на экзамене в интервале $[-7\%; +7\%]$, что говорит о хорошей точности и отличных прогностических свойствах.

Согласно экономической интерпретации прогнозов, полученных по моделям, в 2013 году Украина получит больше инвестиций из-за рубежа, нежели от внутренних инвесторов. Производство Украины будет понемногу наращивать темпы. На фоне роста общего объема инвестиций в страну, инвестиции в основной её капитал будут падать.

1. Ефименко С. Н., Степашко В.С. Основы рекуррентно-параллельных вычислений в комбинаторном алгоритме COMBI МГУА // Управляющие системы и машины. — 2014. — № 6. — С. 27–33.
2. Єфіменко С. М. Комбінаторний алгоритм МГУА з послідовним ускладненням структур моделей на основі рекуррентно-паралельних обчислень // Індуктивне моделювання складних систем. — 2014. — Вип. 6. — С. 81–89.
3. Pavlov A.V. Parallel relaxational iterative algorithm of GMDH. // Індуктивне моделювання складних систем. Зб. наук. праць. — Вип. 6. — К.: МННЦІТС НАНУ. — 2014. — С. 33–40.
4. Павлов А.В. Проектирование системы автоматизированной структурно-параметрической идентификации // Індуктивне моделювання складних систем. Зб. наук. праць. — Вип. 7. — К.: МННЦІТС НАНУ. — 2015. — С. 95–108.
5. Интернет ресурс <https://ru.wikipedia.org/wiki/Qt>

UDC 681.513.8

APPROACH TO ORGANIZATION OF RECURRENT-AND-PARALLEL COMPUTATIONS IN AUTOMIZED STRUCTURE-PARAMETRIC IDENTIFICATION SYSTEM

A.V. Pavlov

International Research and Training Center of Information Technologies and Systems of National Academy of Sciences of Ukraine and Ministry of Education and Science of Ukraine (Kiev)

Introduction. Development and optimization of methods and algorithms for solving statistical modeling problems are the basic directions in science undoubtedly. Although even the most efficient methods and technologies lose their value if they stay just program modules which can be used only by programmers. To be used in practice they should be integrated in some software that has intuitive user-friendly interface. Such software helps to discover a real value of the methods behind it.

The purpose of the paper is increasing the usage effectiveness of a recurrent-and-parallel iterative algorithm by developing a full-fledged modern software based on it for solving forecasting, extrapolation and approximation problems.

Results. The main task of integration of GUI and computational module is a union of two mechanisms of parallization: Threading Building Blocks (TBB) parallization and Qt-parallization. The main idea of proposed approach is that every operation (including the model building operation that create own TBB-threads) initiated by a user should perform in additional Qt-thread. A design pattern that solve this task was developed. The pattern was used to finally implement the ASIS.

The system applied to forecast investments security of Ukraine. A system of forecasting models that describe the state of Ukrainian investments security was

build. Mean absolute percentage error of the models hit the [-7; +7] interval on independent dataset, that indicate a good forecasting ability of the models.

Conclusion. The work suggests an approach to organization of recurrent-and-parallel computations in ASIS that allow integrating the most effective methods for solving statistical modeling problems in user-friendly intuitive full-fledge system that allows any user to solve forecasting, regression and approximation problems with high efficiency. The system was applied to forecast Ukrainian investments security. The economic interpretation of the obtained forecasts says that in 2013 year Ukraine will gain more investments from abroad than from inner investors Ukrainian manufacturing will increase rates; on the background rise of overall country investments, the investments in basic capital will decline.

Keywords: Multithreaded parallelization, design patterns, group method of data handling, recurrent-and-parallel computations, Qt, TBB.

1. Efimenko S., Stepashko V. S, Basics of recurrent-and-parallel computations in combinatorial algorithm COMBI of GMDH // *Controll systems and machines* — 2014. — № 6. — P. 27–33. (in Russian)
2. Efimenko S.M. Combinatorial algorithm of GMDH with sequential complication of structures based on recurrent-and-parallel computations // *Inductive modeling of complex systems* — Vol. 6. — 2014. — P. 81–89. (in Ukrainian)
3. Pavlov A.V. Parallel relaxational iterative algorithm of GMDH // *Inductive modeling of complex systems* — Vol. 6. — 2014. — P. 33–40.
4. Pavlov A.V. Design of automated structure-parametric identification system // *Inductive modeling of complex systems* — Vol.7— 2015. — P. 33–40.
5. Internet resource <https://ru.wikipedia.org/wiki/Qt>.

Получено 28.10.2015