

ИССЛЕДОВАНИЕ ВОЗМОЖНОСТЕЙ ПАРАЛЛЕЛИЗМА ПРИ ИДЕНТИФИКАЦИИ КВАЗИЛИНЕЙНОГО РЕКУРРЕНТНОГО УРАВНЕНИЯ

© 2022 М.С.А. Аботалев, Т.А. Макаровских, А.В. Панюков

South Ural State University

454080 Chelyabinsk, ave. V.I. Lenin, d. 76

E-mail: abotalebmostafa@bk.ru, Makarovskikh.T.A@susu.ru, paniukovav@susu.ru

Поступила в редакцию: ДД.ММ.ГГГГ

В работе рассматривается параллельный вариант алгоритма идентификации параметров квазилинейного рекуррентного уравнения для решения задачи регрессионного анализа с взаимозависимыми наблюдаемыми переменными, основанный на обобщенном методе наименьших модулей (GLDM). В отличие от нейронных сетей, широко используемых в различных системах прогнозирования, данный подход позволяет в явном виде получать качественные квазилинейные разностные уравнения, адекватно описывающие рассматриваемый процесс. Это позволяет повысить качество анализа изучаемых процессов. Проведенные вычислительные эксперименты с использованием временных рядов с информацией о текущем количестве инфицированных Covid-19 в выбранном регионе (или стране) показывают, что максимальное ускорение алгоритма происходит при использовании количества потоков, равного половине возможных потоков для данного устройства. Модель, использующая выявленные параметры, позволяет получить долгосрочный прогноз. *Ключевые слова:* Параллелизм, квазилинейное рекуррентное уравнение, прогнозирование, моделирование, авторегрессионная модель.

ОБРАЗЕЦ ЦИТИРОВАНИЯ

Аботалев М.С.А., Макаровских Т.А., Панюков А.В. Исследование возможностей параллелизма для прогнозирования с использованием квазилинейного рекуррентного уравнения // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2022. Т. X, № Y. С. Z1–Z2. DOI: 10.14529/cmse220Y0Z.

Введение

Анализ временных рядов и прогнозирование в настоящее время являются одной из широко исследуемых областей. Данные подходы используются как при анализе и моделировании экономических процессов, транспортных систем [1], так и для исследования процессов, происходящих в технических системах [2], при анализе климатических процессов и прогнозировании погодных явлений [1], для прогнозирования состояния сельскохозяйственных угодий [3] и др. В 2020–2022 годах к разработке методов прогнозирования добавилось тысячи исследований о распространении заболеваемости Covid-19 по всему миру [4–6].

В настоящее время накоплен большой опыт измерения вибрационных сигналов, разработки методов вибродиагностики и прогнозирования состояния и ресурса механических систем. Одним из наиболее актуальных направлений остается повышение точности и скорости определения диагностических признаков. Это относится, прежде всего, к уникальным высоконагруженным механическим системам, рассмотренным, например, в работе [7]. Решение этой проблемы во многих случаях можно получить через динамические характеристики механических систем. Определению этих характеристик в значительной степени способствует правильный выбор диагностической математической модели, устанавливающей связь меж-

ду пространством состояний объекта и пространством диагностических признаков. К ним относятся динамические модели, представленные в виде разностных уравнений, феноменологические, структурные, регрессионные модели и т.п. Выбор той или иной модели зависит от определяемых характеристик и характера анализируемого процесса.

Идентификация с помощью различных статистических методов, нейронных сетей или математических моделей уже давно используется в различных областях исследований. В настоящее время эти методы применяются не только в промышленности, но и при попытках прогнозирования развития пандемии Covid-19. Например, в работе [8] сравнивается качество прогнозирования процесса распространения инфекции с помощью разных классических моделей, описывается разработанное программное обеспечение для всех этих методов и проводятся вычислительные эксперименты с использованием временных рядов по распространению заболеваемости Covid-19 в различных регионах. Авторы приходят к выводу, что разработанную систему прогнозирования можно использовать для анализа временных рядов, описывающих и другие процессы.

Большинство прогнозов, особенно при большом количестве данных, зачастую проводится с использованием различных моделей нейронных сетей. Например, в статье [9] рассматривается нейросетевая модель, с помощью которой можно прогнозировать изменение цены ферросилиция на внутреннем рынке РФ в краткосрочной перспективе. Приведенная в статье модель отличается высокой точностью прогнозирования и может быть полезна при обосновании стратегических решений в деятельности отраслевых НИИ и металлургических предприятий. В работе [10] описаны эконометрические модели для определения качественного экономического показателя металлургической отрасли, производства, применимые для оценки статистических характеристик производства черных металлов и перспектив развития черной металлургии. Тем не менее, все подобные модели выглядят как волшебный черный ящик, позволяющий получить некий адекватный ответ для каких-то входных данных. Некоторые исследователи используют так называемое когнитивное моделирование для повышения качества прогнозирования с помощью нейронных сетей. Так, статья [11] направлена на сравнение эффективности когнитивных и математических предикторов временных рядов с точки зрения их точности. Эксперимент, проведенный авторами статьи, показывает, что когнитивные модели имеют по крайней мере эквивалентную точность по сравнению с моделями ARIMA. Большинство этих подходов используются для прогнозирования некоторых экономических единиц, объема производства, некоторых параметров логистики и т. д.

Поскольку все перечисленные выше модели дают хорошую точность при краткосрочном прогнозировании, актуальна задача разработки математического подхода, позволяющего в явном виде получать качественные квазилинейные разностные уравнения (адекватно описывающие рассматриваемый процесс) в долгосрочной перспективе. Известны некоторые исследования в этой области, такие как [12], где предлагаемая модель включает очистку данных, сглаживание данных и окончательные данные после предварительной обработки, которые вводятся в регрессионную модель для прогнозирования промышленного потребления электроэнергии. Но в этой статье, как и во многих других, снова рассматриваются только известные статистические методы.

В данной работе для решения задачи регрессионного анализа с взаимозависимыми наблюдаемыми переменными рассматривается алгоритм идентификации параметров квазилинейного рекуррентного уравнения, основанный на обобщенном методе наименьших отклонений (General Least Deviation Method, GLDM). Показано, что модель, использующая

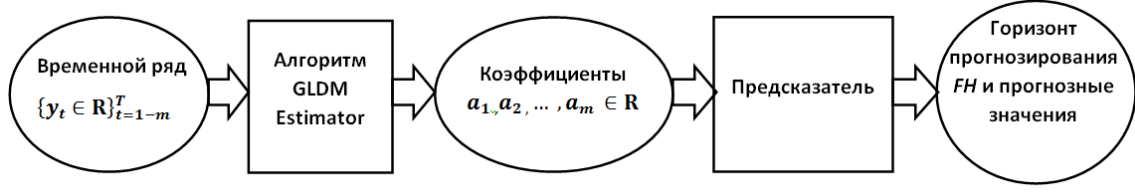


Рис. 1. Схема реализации модели

выявленные параметры, позволяет получить долгосрочный прогноз. В отличие от нейронных сетей (например, [2]) этот подход позволяет в явном виде получать качественные квазилинейные разностные уравнения (адекватно описывающие рассматриваемый процесс).

В работе использован алгоритм из [13], реализованный в виде комплекса программ [14] для определения коэффициентов $a_1, a_2, a_3 \dots, a_m \in \mathbb{R}$ квазилинейной авторегрессионной модели m -го порядка

$$y_t = \sum_{j=1}^{n(m)} a_j g_j(\{y_{t-k}\}_{k=1}^m) + \varepsilon_t, \quad t = 1, 2, \dots, T \quad (1)$$

по актуальной информации о значениях переменных состояния $\{y_t \in \mathbb{R}\}_{t=1-m}^T$ в моменты времени t . Здесь $g_j : (\{y_{t-k}\}_{k=1}^m) \rightarrow \mathbb{R}, j = 1, 2, \dots, n(m)$ – заданные $n(m)$ модельных функций; $\{\varepsilon_t \in \mathbb{R}\}_{t=1}^T$ – неизвестные ошибки.

Найденные коэффициенты $a_1, a_2, a_3 \dots, a_m \in \mathbb{R}$ используются для анализа количества кумулятивных случаев заражения Covid-19 в Челябинской области с 22 марта 2020 г. по 06 января 2022 г. (всего 655 дней распространения инфекции), прогнозирование дальнейшего развития процесса и обсуждение эффективности использования данной модели для прогнозирования событий, связанных с распространением Covid-19. В своем исследовании мы используем временной ряд, начиная с 22 марта 2020 г., поскольку до этой даты для рассматриваемого региона были нулевые значения.

Поскольку для длительного прогнозирования потребуются матрицы высокого порядка, для чего необходимы большие затраты памяти и времени, в работе исследована реализация [14] алгоритма прогнозирования, использующая параллелизм. Для исследования возможности многопоточного программирования разработанного алгоритма прогнозирования проведены вычислительные эксперименты на двух разных ПК.

1. Обозначения и описание алгоритмов

Рассматриваемый алгоритм работает следующим образом (см. рис. 1). Алгоритм GLDM [13] на входе получает временной ряд $\{y_t \in \mathbb{R}\}_{t=-1-m}^T$ длины $T + m \geq (1 + 3m + m^2)$ и определяет коэффициенты $a_1, a_2, a_3 \dots, a_m$, решая задачу оптимизации

$$\{a_j^*\}_{j=1}^{n(m)} = \arg \min_{\{a_j\}_{j=1}^{n(m)} \subset \mathbb{R}} \left[\sum_{t=1}^T \arctan \left| \sum_{j=1}^{n(m)} a_j g_j(\{y_{t-k}\}_{k=1}^m) - y_t \right| \right], \quad (2)$$

с использованием в функции потерь распределения Коши

$$F(\xi) = \frac{1}{\pi} \arctan(\xi) + \frac{1}{2},$$

которое имеет максимальную энтропию среди распределений случайных величин, не имеющих математического ожидания и дисперсии.

Ниже приведен пример полного набора квадратичных модельных функций для разностного уравнения порядка m .

$$g_j(\{y_{t-k}\}_{k=1}^m) = y_{t-j}, \quad j = 1, 2, \dots, m$$

$$g_j(\{y_{t-k}\}_{k=1}^m) = y_{t-r} \cdot y_{t-s}, \quad 1 \leq r \leq s \leq m, \\ j = m+1, m+2, \dots, n(m) = m + C_{m+1}^2 = m(m+3)/2.$$

Предиктор формирует индексированное $t = 1, 2, \dots, T-1, T$ семейство разностных уравнений m -го порядка

$$\overline{y[t]}_\tau = \sum_{j=1}^{n(m)} a_j^* g_j(\{\overline{y[t]}_{\tau-k}\}_{k=1}^m), \quad \tau = t, t+1, t+2, t+3, \dots, T-1, T, T+1, \dots \quad (3)$$

для решетчатых функций $\overline{y[t]}$ со значениями $\overline{y[t]}_\tau$, которые интерпретируются как построенный в момент времени t прогноз для y_τ . Воспользуемся решением задачи Коши для разностного уравнения (3) при начальных условиях

$$\overline{y[t]}_{t-1} = y_{t-1}, \overline{y[t]}_{t-2} = y_{t-2}, \dots, \overline{y[t]}_{t-m} = y_{t-m}, \quad t = 1, 2, \dots, T-1, T \quad (4)$$

и определим значения функции $\overline{y[t]}$.

Итак, имеем множество $\overline{Y}_\tau = \{\overline{y[t]}_\tau\}_{t=1}^T$ возможного прогнозного значения y_τ . Далее используем этот набор для оценки вероятностных характеристик величины y_τ .

1.1. Оценка по обобщенному методу наименьших отклонений

Задача (2), т.е. GLDM-оценка, представляет собой задачу многоэкстремальной оптимизации. GLDM-оценки устойчивы к наличию корреляции значений в $\{y_t \in \mathbb{R}\}_{t=-1-m}^T$ и (при соответствующих параметрах) лучше всего подходят для вероятностных распределений ошибок с более тяжелыми (чем у нормального распределения) хвостами [7]. Все вышеизложенное показывает возможность решения задачи идентификации (1) с использованием решения (2).

Используем взаимосвязь между GLDM-оценками и оценками взвешенным методом наименьших отклонений [15] (WLDM-оценки) для решения задач (2) более высокой размерности.

В данной статье рассмотрим алгоритм оценивания GLDM [16]. Начнем с алгоритма оценки взвешенного метода наименьших отклонений (Weighted Least Deviation Method, WLDM), используемого в алгоритме GLDM.

1.2. Оценка по взвешенному методу наименьших отклонений

Алгоритм WLDM-оценки [13] в качестве входных данных получает временной ряд $\{y_t \in \mathbb{R}\}_{t=1-m}^T$ и весовые коэффициенты $\{p_t \in \mathbb{R}^+\}_{t=1}^T$ и вычисляет множители

$$a_1, a_2, a_3 \dots, a_{n(m)} \in \mathbb{R}$$

путем решения задачи оптимизации

$$\{a_j^*\}_{j=1}^{n(m)} = \arg \min_{\{a_j\}_{j=1}^{n(m)} \in \mathbb{R}^{n(m)}} \left[\sum_{t=1}^T p_t \cdot \left| \sum_{j=1}^{n(m)} a_j g_j(\{y_{t-k}\}_{k=1}^m) - y_t \right| \right]. \quad (5)$$

Эта задача представляет собой задачу выпуклой кусочно-линейной оптимизации, а введение дополнительных переменных сводит ее к задаче линейного программирования

$$\sum_{t=1}^T p_t z_t \rightarrow \min_{\substack{(a_1, a_2, \dots, a_{n(m)}) \in \mathbb{R}^m, \\ (z_1, z_2, \dots, z_T) \in \mathbb{R}^T}}, \quad (6)$$

$$-z_t \leq \sum_{j=1}^{n(m)} [a_j g_j(\{y_{t-k}\}_{k=1}^m)] - y_t \leq z_t, \quad t = 1, 2, \dots, T, \quad (7)$$

$$z_t \geq 0, \quad t = 1, 2, \dots, T. \quad (8)$$

Задача (6)–(8) имеет каноническую форму с переменными $n(m) + T$ и ограничениями-неравенствами $3n$, включая условия неотрицательности для переменных z_j , $j = 1, 2, \dots, T$.

Двойственная к данной задаче (6) представляет собой

$$\sum_{t=1}^T (u_t - v_t) y_t \rightarrow \max_{u, v \in \mathbb{R}^T}, \quad (9)$$

$$\sum_{t=1}^T a_j g_j(\{y_{t-k}\}_{k=1}^m) (u_t - v_t) = 0, \quad j = 1, 2, \dots, n(m), \quad (10)$$

$$u_t + v_t = p_t, \quad u_t, v_t \geq 0, \quad t = 1, 2, \dots, T. \quad (11)$$

Введем переменные $w_t = u_t - v_t$, $t = 1, 2, \dots, T$. Условия (11) подразумевают

$$u_t = \frac{p_t + w_t}{2}, \quad v_t = \frac{p_t - w_t}{2}, \quad -p_t \leq w_t \leq p_t, \quad t = 1, 2, \dots, T.$$

Поэтому оптимальное значение задачи (9)–(11) равно оптимальному решению задачи

$$\sum_{t=1}^T w_t \cdot y_t \rightarrow \max_{w \in \mathbb{R}^T}, \quad (12)$$

$$\sum_{t=1}^T g_j(\{y_{t-k}\}_{k=1}^m) \cdot w_t = 0, \quad j = 1, 2, \dots, n(m), \quad (13)$$

$$-p_t \leq w_t \leq p_t, \quad t = 1, 2, \dots, T. \quad (14)$$

Ограничения (13) определяют $(T - n(m))$ -мерное линейное подпространство \mathcal{L} с матрицей

$$S = \begin{bmatrix} g_1(\{y_{1-k}\}_{k=1}^m) & g_1(\{y_{2-k}\}_{k=1}^m) & \dots & g_1(\{y_{T+1-k}\}_{k=1}^m) \\ g_2(\{y_{1-k}\}_{k=1}^m) & g_2(\{y_{2-k}\}_{k=1}^m) & \dots & g_2(\{y_{T+1-k}\}_{k=1}^m) \\ \vdots & \vdots & \ddots & \vdots \\ g_{n(m)}(\{y_{1-k}\}_{k=1}^m) & g_{n(m)}(\{y_{2-k}\}_{k=1}^m) & \dots & g_{n(m)}(\{y_{T+1-k}\}_{k=1}^m) \end{bmatrix}$$

Ограничения (14) определяют T -мерный брус \mathcal{T} . Простая структура допустимого множества задачи (12)–(14): пересечение $(T - n(m))$ -мерного линейного подпространства \mathcal{L} (13)

и T -мерного бруса $\mathcal{T}(14)$, – позволяют найти решение алгоритмом, использующим градиентную проекцию целевой функции (12) (т.е. вектор $\nabla = \{y_t\}_{t=1}^T$) на допустимой области $\mathcal{L} \cap \mathcal{T}$, которая определяется формулой ограничения (13)–(14). Матрица оператора проектирования на \mathcal{L} имеет вид

$$S_{\mathcal{L}} = E - S^T \cdot (S \cdot S^T)^{-1} \cdot S,$$

а проекция градиента на \mathcal{L} равна $\nabla_{\mathcal{L}} = S_{\mathcal{L}} \cdot \nabla$. Кроме того, если внешняя нормаль на какой-либо грани бруса образует острый угол с проекцией градиента $\nabla_{\mathcal{L}}$, то перемещение по этой грани равно нулю.

Алгоритм **DualWLDMSolver** решает задачу (12)–(14), он начинает поиск оптимального решения с нуля, двигаясь по направлению $\nabla_{\mathcal{L}}$. Если текущая точка попадает на грань бруса \mathcal{T} , то соответствующая координата в направлении движения принимается равной нулю. Вычислительная сложность такого алгоритма не превышает величины $O(T^2)$ благодаря простой структуре допустимого множества: пересечение T -мерного кубоида (14) и $(T - n(m))$ -мерного линейного подпространства (13).

Если (w^*, R^*) – результат выполнения алгоритма проектирования градиента, то w^* – оптимальное решение задачи ((12))–((14)), а оптимальное решение задачи 9–11 равно

$$u_t^* = \frac{p_t + w_t^*}{2}, \quad v_t^* = \frac{p_t - w_t^*}{2}, \quad t = 1, 2, \dots, T.$$

1.3. Решение прямой задачи

Решение прямой задачи следует из условия дополненности для пары взаимно двойственных задач (6)–(8) и (9)–(11), которые

$$y_t = \sum_{j=1}^{n(m)} [a_j g_j(\{y_{t-k}\}_{k=1}^m)] \quad \forall t \notin R^*, \quad (15)$$

$$y_t = \sum_{j=1}^{n(m)} [a_j g_j(\{y_{t-k}\}_{k=1}^m)] + z_t^*, \quad \forall t \in R^* : w_t^* = p_t, \quad (16)$$

$$y_t = \sum_{j=1}^{n(m)} [a_j g_j(\{y_{t-k}\}_{k=1}^m)] - z_t^*, \quad \forall t \in R^* : w_t^* = -p_t. \quad (17)$$

Фактически решение $(\{a_j^*\}_{j=1}^{n(m)}, z^*)$ системы линейных алгебраических уравнений (15)–(17) представляет собой оптимальное решение двойственной задачи (12)–(14) и оптимальное решение задачи (5), что доказывает справедливость следующей теоремы.

Теорема 1. Пусть w^* – оптимальное решение задачи (12)–(14), $(\{a_j^*\}_{j=1}^{n(m)}, z^*)$ – решение системы линейных алгебраических уравнений (15)–(17), то $(\{a_j^*\}_{j=1}^{n(m)})$ – оптимальное решение задачи (5).

Вышеизложенное позволяет нам предложить алгоритм WLDМ-оценки. Основной проблемой при использовании этого алгоритма является отсутствие общих формальных правил выбора весовых коэффициентов. Следовательно, такой подход требует дополнительных исследований.

Установленные в [16] результаты позволяют свести задачу определения GLDM-оценки к итерационной процедуре с WLDМ-оценками.

1.4. Алгоритм оценки по обобщенному методу наименьших отклонений

Задача (2) оценки GLDM является задачей вогнутой оптимизации. GLDM-оценки устойчивы к наличию корреляции значений в $\{S_{jt} : t = 1, 2, \dots, T; j = 1, 2, \dots, N\}$, и (с соответствующими настройками) как лучший для вероятностных распределений ошибок с более тяжелыми (чем у нормального распределения) хвостами [7]. Вышеизложенное показывает возможность решения задачи идентификации (1) путем решения последовательности задач (15) с соответствующими весами. Установленные в [16] результаты позволяют свести задачу определения GLDM-оценки к итерационной процедуре с WLDM-оценками.

Теорема 2. Последовательность $\{(A^{(k)}, z^{(k)})\}_{k=1}^{\infty}$, построенная по алгоритму GLDM-оценки, сходится к глобальному минимуму (a^*, z^*) задачи (2).

Описание алгоритма **GLDM-оценка** показывает, что его вычислительная сложность пропорциональна вычислительной сложности алгоритма решения простых и/или двойственных задач WLDM (5). Многочисленные вычислительные эксперименты показывают, что среднее число итераций алгоритма **GLDM-оценки** равно количеству коэффициентов в идентифицированном уравнении. Если эта гипотеза верна, то вычислительная сложность решения практических задач не превышает $O((n(m))^3 T + n(m) \cdot T^2)$. Необходимо учитывать, что поиск и нахождение уравнения авторегрессии высокого порядка имеют свои специфические условия. Одним из таких условий, в частности, является высокая чувствительность алгоритма к ошибкам округления. Для исключения возможности ошибки в вычислениях необходимо точно выполнять основные арифметические операции над полем рациональных чисел [17] и дополнять их параллельным программным кодом.

1.5. Алгоритм прогнозирования

Предиктор формирует индексированное $t = 1, 2, \dots, T-1, T$ семейство разностных уравнений m -го порядка (3) для решетчатых функций $\overline{y[t]}$ со значениями $\overline{y[t]}_{\tau}$ и интерпретируется как построенный на момент времени t прогноз для y_{τ} . Воспользуемся решением задачи Коши для ее разностного уравнения (3) при начальных условиях (4), чтобы найти значения функции $\overline{y[t]}$.

Итак, у нас есть множество $\overline{Y}_{\tau} = \{\overline{y[t]}_{\tau}\}_{t=1}^T$ возможного предсказания значения y_{τ} . Далее мы используем этот набор для оценки вероятностных характеристик величины y_{τ} .

Схема алгоритма приведена на рисунке 2.

2. Программная реализация

В рассматриваемой программе используется распараллеливание средствами OpenMP, что предполагает модель параллельного программирования SPMD. В этом случае для всех параллельных потоков используется один и тот же код, а в программу вставляются специальные директивы. Программа имеет последовательные области, когда выполняется один процесс (поток), а при входе в параллельную область порождается определенное количество процессов, между которыми в дальнейшем распределяются части кода. Максимально возможное количество потоков в рассматриваемой программе определяется с помощью функции `omp_get_max_threads()` в зависимости от ресурсов компьютера, на котором запущена программа. В вычислительном эксперименте, представленном ниже, программа запуска-

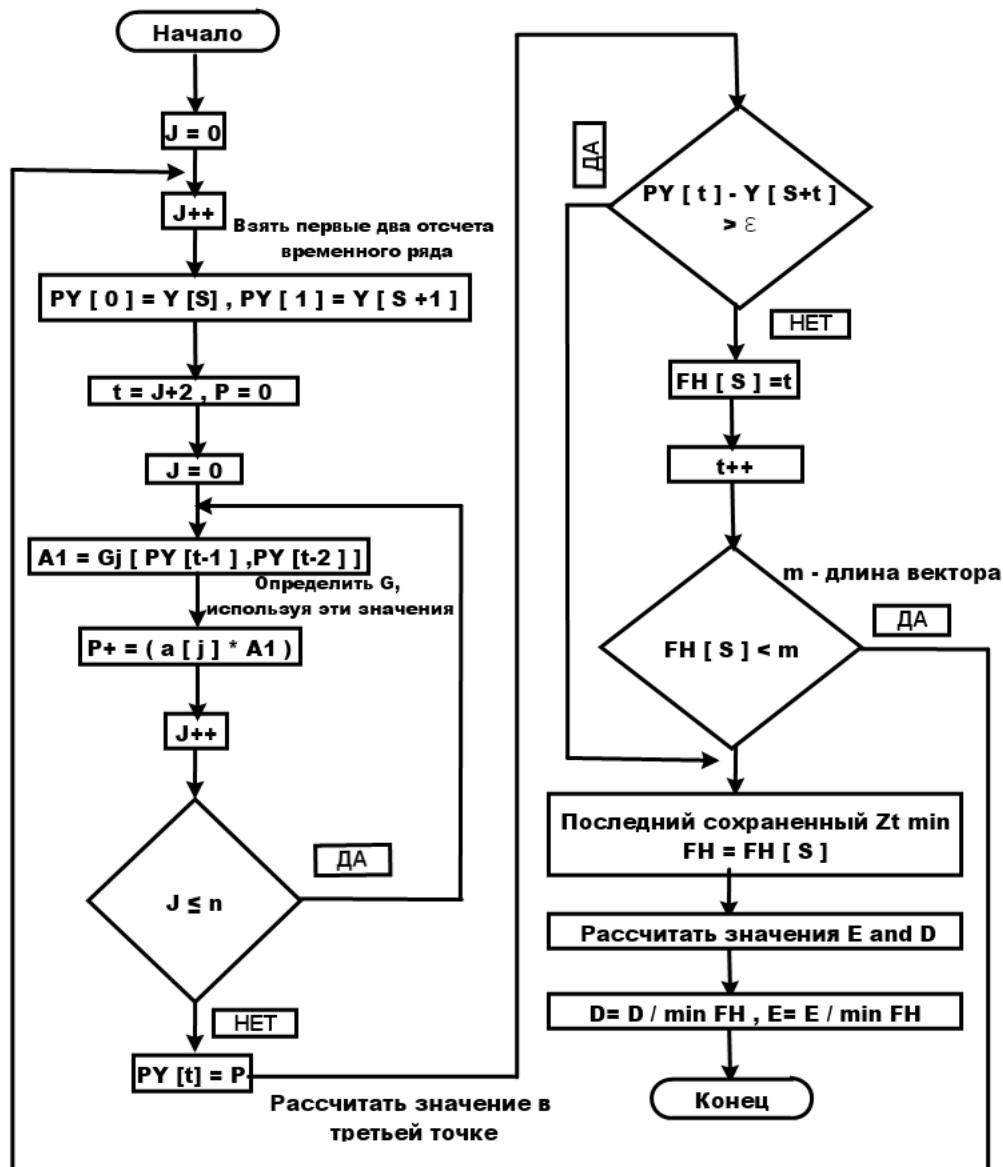


Рис. 2. Схема алгоритма прогнозирования

ется с количеством потоков от 1 до максимально возможного значения для конкретного компьютера.

Схема разработанного алгоритма показана на рисунке 3. На рис. 3 приведена последовательность вызова разработанных процедур из функции `void main()`, используемая для запуска алгоритма. Листинги всех функций, вызываемых из `main()`, приведены в [14].

В разработанной программе распараллелено выполнение всех возможных циклов один за другим. Циклы, которые не могут быть адекватно распараллелены и остаются последовательными областями: (1) первый цикл функции основного решения WLDM; (2) базовый цикл `do... while` алгоритма оценки GLDM (вложенные циклы распараллелены). Все остальные циклы могут выполняться как параллельные участки. В функции прогнозирования `ForecastingEst()`, внешний цикл `do...while` также работает с одним потоком, однако его вложенные циклы работают параллельно.

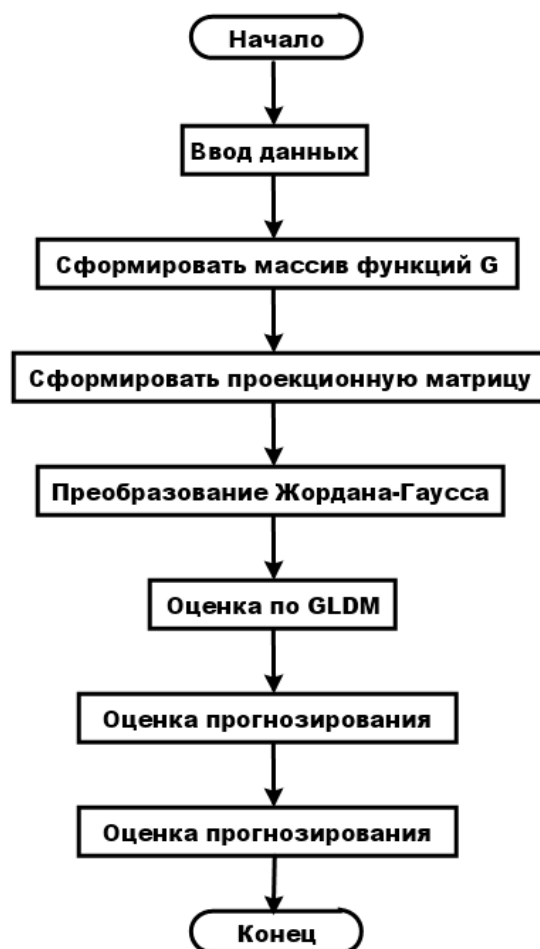


Рис. 3. Последовательность вызова функций из основной функции программы

Эта функция формирует массив $PY[] []$ (см. листинг 1), каждый элемент которого равен значениям решеточных функций $\overline{y[t]}_\tau$ семейства разностных уравнений m -го порядка (3). Используя этот массив, можно получить значения ошибок E и D :

$$E = \sum_{t=3}^{T_R} \left(\overline{y[t]} - \overline{y[t]}_\tau \right),$$

$$D = \sum_{t=3}^{T_R} \left| \overline{y[t]} - \overline{y[t]}_\tau \right|,$$

где T_R – надежный горизонт прогнозирования, т.е.

$$T_R = \min \left\{ \tau : z > \left| \overline{y[t]} - \overline{y[t]}_\tau \right| \right\}.$$

Листинг 1: Расчет средних ошибок прогнозирования

```
void ForecastingEst() {
    int t, Strt=0, Et;
    do {
        Strt++;
        PY[ Strt ][0]=Y[ Strt ];
        PY[ Strt ][1]=Y[ Strt +1];
        for (t=Strt+2; t<m; t++){
            double py=0;
            #pragma omp parallel for reduction(+: py)
            for (int j=0; j<=n; j++){
                double A1=G[j](PY[ Strt ][ t-1 ],PY[ Strt ][ t-2]);
                double R=a[j]*A1;
                py+=R;
            }
            PY[ Strt ][ t]=py;
            if (fabs(PY[ Strt ][ t]-Y[( Strt)+t]) > SZ) break;
        }
        FH[ Strt]=t;
    } while (FH[ Strt]<m);
    LastStrt=t;
    minFH=FH[ Strt ];
    #pragma omp parallel for private (minFHp)
    {
        int minFHp=minFH;
        for (int t=3; t<Strt; t++){
            if (FH[t]<minFH) minFHp=FH[t];
        }
        #pragma omp critical
        minFH=(minFHp<minFH)? minFHp : minFH;
    }
    E=D=0;
```

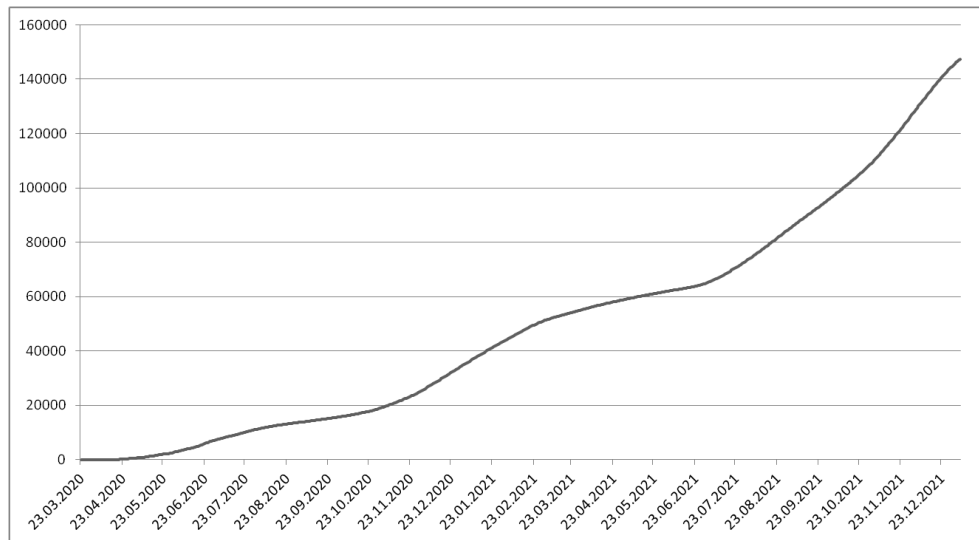


Рис. 4. Прирост числа кумулятивных случаев Covid-19 по Челябинской области (данные с 23.03.2020 по 06.01.2022)

```
#pragma omp parallel for shared(Y,PY) reduction(+:D,E)
for (int t=3; t<=minFH; t++){
    D+=fabs(Y[t+Strt]-PY[ Strt ][ t ]);
    E+=(Y[t+Strt]-PY[ Strt ][ t ]);
}
D/=minFH; E/=minFH;
}
```

3. Вычислительный эксперимент

Рассмотрим временной ряд, содержащий кумулятивные данные по случаям заражения вирусом Covid-19 в Челябинской области. В эксперименте рассматриваются векторы разной длины, самый длинный из которых имеет 655 значений. График процесса показан на рисунке 4. Полученные средние ошибки прогноза разработанной модели составляют $E = 1,615210 \cdot 10^{-1}$ и $D = 9,852680 \cdot 10^{-1}$ при горизонте прогнозирования 645 дней. Этот результат лучше результатов, полученных с помощью классической статистической модели и модели нейронных сетей, поскольку наша модель позволяет получить долгосрочный прогноз (тем не менее, ARIMA, линейная модель Хольта, NNAR, LSTM позволяют получать хорошие прогнозы только на 30–60 дней, т.к. это было показано в вычислительных экспериментах, описанных в [18]).

Исследуем время выполнения и ускорение при распараллеливании для векторов длиной 150, 300, 500 и 655 элементов. Эксперименты проводились на двух персональных компьютерах:

1. Intel Core i7-4770, 3.4GHz, RAM 16Gb;
2. Intel Core i5-2430M, 2.4 GHz, RAM 4 Gb.

Для компьютера (1) программа определила максимальное количество потоков равное 8, а для компьютера (2) — 4 потока. Для нашего эксперимента зафиксируем количество потоков, равное 8 для обоих компьютеров. Время выполнения разработанной программы для разного количества N потоков показано в таблице 1 и на рисунках 5 и 6.

Таблица 1. Время вычислений для разного количества потоков и длин векторов

N	Число нитей	1	2	3	4	5	6	7	8
150	Comp (1) (10^{-4} с)	8.105	7.326	5.832	5.812	5.773	5.788	5.942	5.926
300	Comp (1) (10^{-3} с)	3.475	3.648	2.363	2.245	2.506	3.802	2.533	2.459
500	Comp (1) (10^{-2} с)	1.371	1.093	1.028	1.118	1.156	1.082	1.094	1.105
655	Comp (1) (10^{-2} с)	1.783	1.140	1.085	1.029	1.006	1.062	1.010	1.029
150	Comp (2) (10^{-2} с)	1.007	0.951	1.016	0.980	1.289	0.991	0.909	0.993
300	Comp (2) (10^{-2} с)	3.685	3.701	3.560	3.592	3.600	3.513	3.477	3.421
500	Comp (2) (10^{-2} с)	10.102	9.819	9.714	9.692	9.861	9.744	9.920	9.650
655	Comp (2) (10^{-2} с)	17.349	16.379	16.257	16.221	16.219	16.229	16.332	16.362

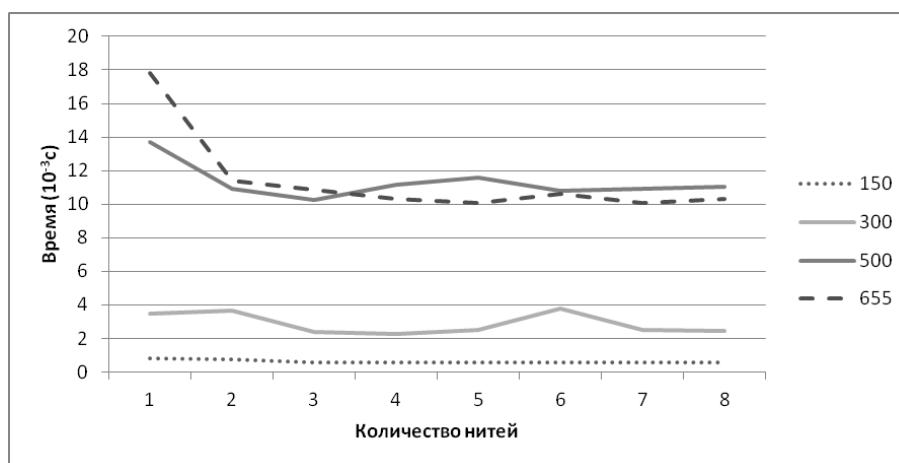


Рис. 5. График времени работы программы на компьютере (1) (время в сек· 10^{-3}) для временных рядов длиной 150, 300, 500, 655 точек

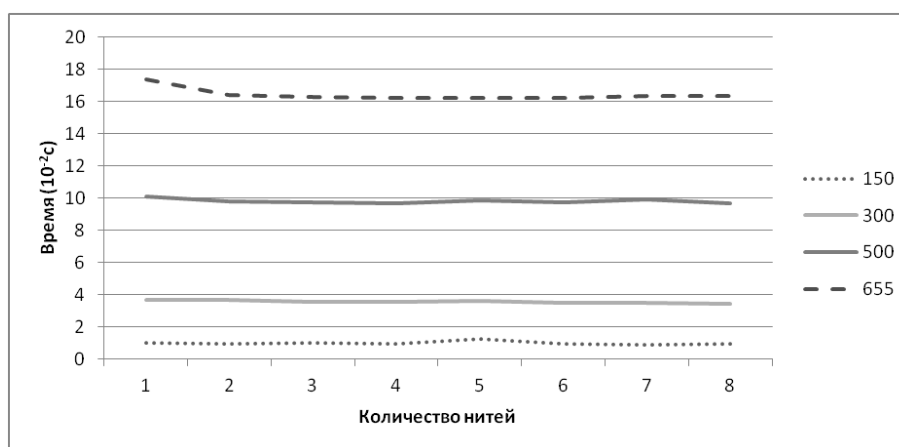


Рис. 6. 87 / 5 000 Результаты перевода График времени работы программы на компьютере (2) (время в сек· 10^{-2}) для временных рядов длиной 150, 300, 500, 655 точек

Легко видеть, что ускорения почти нет. Причина, вероятно, в малом объеме используемой оперативной памяти используемых компьютеров, а так же небольшой длине исследуемых векторов.

Заключение

Говоря о качестве самой модели, можно отметить, что она работает не хуже нейросетевых моделей или классических статистических моделей. У нее есть одно существенное преимущество по сравнению с этими моделями, заключающееся в возможности интерпретировать коэффициенты модели с точки зрения задачи исследования. Проведенное нами тестирование этой модели на данных о кумулятивных случаях заражения вирусом Covid-19 в Челябинской области показало, что ошибка прогноза меньше, чем для остальных моделей. Поскольку для запуска разработанного алгоритма необходимо использовать матрицы высокого порядка, полезно распараллелить алгоритм с целью ускорения процесса получения результата. Проведенные вычислительные эксперименты показывают, что при простейших способах распараллеливания рассмотренных алгоритмов имеем незначительное ускорение алгоритма для количества потоков, равного половине возможных потоков для данного устройства. Увеличение количества потоков позволяет получить почти такое же слабое ускорение.

Исследование выполнено при финансовой поддержке Министерства науки и высшего образования РФ (государственное задание FENU-2020-0022).

Литература

1. Li Q., Wang J., Zhang H. A wind speed interval forecasting system based on constrained lower upper bound estimation and parallel feature selection // Knowl. Based Syst. 2021. Vol. 231. DOI: 10.1016/j.knosys.2021.107435.
2. Khashei M., Chahkoutahi F. A comprehensive low-risk and cost parallel hybrid method for electricity load forecasting // Comput. Ind. Eng. 2021. Vol. 155. DOI: 10.1016/j.cie.2021.107182.
3. Supuwiningsih N., Kadeksukerti N., Putra A., P. D. Forecasting of Agricultural Production Results in South Denpasar Using Quadratic Trend Method Based GIS // International Journal of Engineering Technologies and Management Research. 2018. Vol. 5, issue 2. DOI: 10.5281/zenodo.1186523.
4. Hamdi F., Raby H., Hakim G., *et al.* A Generalized Mechanistic Model for Assessing and Forecasting the Spread of the COVID-19 Pandemic // IEEE Access. 2021. Vol. 9. P. 13266–13285. DOI: 10.1109/ACCESS.2021.3051929.
5. Dash S., Chakraborty C., Giri S.K., *et al.* BIFM: Big-Data Driven Intelligent Forecasting Model for COVID-19 // IEEE Access. 2021. Vol. 9. P. 97505–97517. DOI: 10.1109/ACCESS.2021.3094658.
6. F. C.-B., M.A. M.-B. An Autoregressive Disease Mapping Model for Spatio-Temporal Forecasting // Mathematics. 2021. Vol. 9, no. 384. DOI: 10.3390/math9040384.
7. Panyukov A., Tyrsin A. Stable Parametric Identification of Vibratory Diagnostics Objects // Journal of Vibroengineering. 2008. Vol. 10, no. 2. P. 142–146. URL: <http://elibrary.ru/item.asp?id=14876532>.

8. Makarovskikh T., Abotaleb M. Comparison Between Two Systems for Forecasting Covid-19 Infected Cases // IFIP Advances in Information and Communication Technology. 2021. Vol. 616. P. 107–114. DOI: 10.1007/978-3-030-86582-5_10.
9. Sirotin D. Neural network approach to forecasting the cost of ferroalloy products // Izvestiya. Ferrous Metallurgy. 2020. Vol. 63(1). P. 78–83. DOI: 10.17073/0368-0797-2020-1-78-83.
10. Yakubova D. Econometric models of development and forecasting of black metallurgy of Uzbekistan // Asian Journal of Multidimensional Research (AJMR). 2019. Vol. 8, issue 5. P. 310–314. DOI: 10.5958/2278-4853.2019.00205.2.
11. Neto A., Ferreira T., Batista M., Firmino P. Studying the Performance of Cognitive Models in Time Series Forecasting // Revista de Informatica Teorica e Aplicada. 2020. Vol. 27, no. 1. P. 83–91. DOI: 10.22456/2175-2745.96181.
12. Panchal R., Kumar B. Forecasting industrial electric power consumption using regression based predictive model // Recent Trends in Communication and Electronics. 2021. DOI: 10.1201/9781003193838-26.
13. Panyukov A.V., Mezaal Y.A. Improving of the Identification Algorithm for a Quasilinear Recurrence Equation // Advances in Optimization and Applications. Vol. 1340 / ed. by N. Olenov, Y. Evtushenko, M. Khachay, V. Malkova. Springer International Publishing, 2020. P. 15–26. DOI: 10.1007/978-3-030-65739-0_2.
14. Makarovskikh T., Panyukov A., Abotaleb M. Generalized least deviation method for identification of quasi-linear autoregressive model. URL: <https://github.com/tmakarovskikh/GLDMPredictor.git> (accessed: 27.7.2022).
15. Pan J., Wang H., Qiwei Y. Weighted Least Absolute Deviations Estimation for ARMA Models with Infinite Variance // Econometric Theory. 2007. Vol. 23(3). P. 852–879.
16. Stable estimation of autoregressive model parameters with exogenous variables on the basis of the generalized least absolute deviation method. 51(11). 01/2018. P. 1666–1669. Open access DOI: 10.1016/j.ifacol.2018.08.217.
17. Panyukov A. Scalability of Algorithms for Arithmetic Operations in Radix Notation // Reliable Computing. 2015. Vol. 19. P. 417–434. URL: <http://interval.louisiana.edu/reliable-computing-journal/volume-19/reliable-computing-19-pp-417-434.pdf>.
18. Abotaleb M.S., Makarovskikh T. Analysis of Neural Network and Statistical Models Used for Forecasting of a Disease Infection Cases // 2021 International Conference on Information Technology and Nanotechnology (ITNT). 2021. P. 1–7. DOI: 10.1109/ITNT52450.2021.9649126.

Мостафа Салахелдин Абделсалам Аботалеб, аспирант, кафедра системного программирования, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация)

Татьяна Анатольевна Макаровских, д.ф.-м.н., доцент, кафедра системного программирования, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация)

Анатолий Васильевич Панюков, д.ф.-м.н., профессор, кафедра системного программирования, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация)

EXPLORING THE POSSIBILITIES OF PARALLELISM UNDER IDENTIFYING A QUASI-LINEAR RECURRENT EQUATION *

© 2022 M.S.A. Abotaleb, T.A. Makarovskikh, A.V. Panyukov

¹South Ural State University (pr. Lenina 76, Chelyabinsk, 454080 Russia),

²Lomonosov Moscow State University (GSP-1, Leninskie Gory 1, Moscow, 119991 Russia)

E-mail: abotalebmostafa@bk.ru, Makarovskikh.T.A@susu.ru, paniukovav@susu.ru

Received: ДД.ММ.ГГГГ

The paper considers a parallel version of the algorithm for identifying the parameters of a quasi-linear recurrent equation, since it is necessary to use high-order matrices for this algorithm to work. The algorithm is used to solve the problem of regression analysis with interdependent observed variables, which makes it possible to implement the generalized last deviations (GLDM) method. Using this model, a computational experiment was conducted using time series with information about the current number of infected Covid-19 in the selected region (or country). Our computational experiments show that the acceleration of the algorithm occurs when using the number of threads equal to half of the possible threads for a given device. Increasing the number of threads allows you to get almost the same weak acceleration. A model using the identified parameters allows you to get a long-term forecast. Unlike neural networks, which are widely used in various forecasting systems, this approach allows us to explicitly obtain qualitative quasi-linear difference equations that adequately describe the process under consideration.

Ключевые слова: Parallelism, quasi-linear recurrent equation, forecasting, modeling, autoregressive model.

FOR CITATION

Abotaleb M.S.A., Makarovskikh T.A., Panyukov A.V. Investigation of the possibilities of parallelism for forecasting using a quasi-linear recurrent equation. Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering. 2022. Vol. X, no. Y. P. Z1–Z2. (in Russian) DOI: 10.14529/cmse220Y0Z.

This paper is distributed under the terms of the Creative Commons Attribution-Non Commercial 4.0 License which permits non-commercial use, reproduction and distribution of the work without further permission provided the original work is properly cited.

References

1. Li Q., Wang J., Zhang H. A wind speed interval forecasting system based on constrained lower upper bound estimation and parallel feature selection. Knowl. Based Syst. 2021. Vol. 231. DOI: 10.1016/j.knosys.2021.107435.
2. Khashei M., Chahkoutahi F. A comprehensive low-risk and cost parallel hybrid method for electricity load forecasting. Comput. Ind. Eng. 2021. Vol. 155. DOI: 10.1016/j.cie.2021.107182.
3. Supuwiningasih N., Kadeksukerti N., Putra A., P. D. Forecasting of Agricultural Production Results in South Denpasar Using Quadratic Trend Method Based GIS. International Journal

*The work was carried out with the support of the Ministry of Science and Higher Education of the Russian Federation (state task FENU-2020-0022)

- of Engineering Technologies and Management Research. 2018. Vol. 5, issue 2. DOI: 10.5281/zenodo.1186523.
4. Hamdi F., Raby H., Hakim G., *et al.* A Generalized Mechanistic Model for Assessing and Forecasting the Spread of the COVID-19 Pandemic. IEEE Access. 2021. Vol. 9. P. 13266–13285. DOI: 10.1109/ACCESS.2021.3051929.
 5. Dash S., Chakraborty C., Giri S.K., *et al.* BIFM: Big-Data Driven Intelligent Forecasting Model for COVID-19. IEEE Access. 2021. Vol. 9. P. 97505–97517. DOI: 10.1109/ACCESS.2021.3094658.
 6. F. C.-B., M.A. M.-B. An Autoregressive Disease Mapping Model for Spatio-Temporal Forecasting. Mathematics. 2021. Vol. 9, no. 384. DOI: 10.3390/math9040384.
 7. Panyukov A., Tyrsin A. Stable Parametric Identification of Vibratory Diagnostics Objects. Journal of Vibroengineering. 2008. Vol. 10, no. 2. P. 142–146. URL: <http://elibrary.ru/item.asp?id=14876532>.
 8. Makarovskikh T., Abotaleb M. Comparison Between Two Systems for Forecasting Covid-19 Infected Cases. IFIP Advances in Information and Communication Technology. 2021. Vol. 616. P. 107–114. DOI: 10.1007/978-3-030-86582-5_10.
 9. Sirotin D. Neural network approach to forecasting the cost of ferroalloy products. Izvestiya. Ferrous Metallurgy. 2020. Vol. 63(1). P. 78–83. DOI: 10.17073/0368-0797-2020-1-78-83.
 10. Yakubova D. Econometric models of development and forecasting of black metallurgy of Uzbekistan. Asian Journal of Multidimensional Research (AJMR). 2019. Vol. 8, issue 5. P. 310–314. DOI: 10.5958/2278-4853.2019.00205.2.
 11. Neto A., Ferreira T., Batista M., Firmino P. Studying the Performance of Cognitive Models in Time Series Forecasting. Revista de Informatica Teorica e Aplicada. 2020. Vol. 27, no. 1. P. 83–91. DOI: 10.22456/2175-2745.96181.
 12. Panchal R., Kumar B. Forecasting industrial electric power consumption using regression based predictive model. Recent Trends in Communication and Electronics. 2021. DOI: 10.1201/9781003193838-26.
 13. Panyukov A.V., Mezaal Y.A. Improving of the Identification Algorithm for a Quasilinear Recurrence Equation. Advances in Optimization and Applications. Vol. 1340 / ed. by N. Oleney, Y. Evtushenko, M. Khachay, V. Malkova. Springer International Publishing, 2020. P. 15–26. DOI: 10.1007/978-3-030-65739-0_2.
 14. Makarovskikh T., Panyukov A., Abotaleb M. Generalized least deviation method for identification of quasi-linear autoregressive model. URL: <https://github.com/tmakarovskikh/GLDMPredictor.git> (accessed: 27.7.2022).
 15. Pan J., Wang H., Qiwei Y. Weighted Least Absolute Deviations Estimation for ARMA Models with Infinite Variance. Econometric Theory. 2007. Vol. 23(3). P. 852–879.
 16. Stable estimation of autoregressive model parameters with exogenous variables on the basis of the generalized least absolute deviation method. 51(11). 01/2018. P. 1666–1669. Open access DOI: 10.1016/j.ifacol.2018.08.217.
 17. Panyukov A. Scalability of Algorithms for Arithmetic Operations in Radix Notation. Reliable Computing. 2015. Vol. 19. P. 417–434. URL: <http://interval.louisiana.edu/reliable-computing-journal/volume-19/reliable-computing-19-pp-417-434.pdf>.

18. Abotaleb M.S., Makarovskikh T. Analysis of Neural Network and Statistical Models Used for Forecasting of a Disease Infection Cases. 2021 International Conference on Information Technology and Nanotechnology (ITNT). 2021. P. 1–7. DOI: 10.1109/ITNT52450.2021.9649126.