

А. Ю. Морозов, канд. физ.-мат. наук, науч. сотр., morozov@infway.ru,
Федеральный исследовательский центр "Информатика и управление" Российской
академии наук (ФИЦ ИУ РАН), Москва

Параллельный алгоритм параметрической идентификации динамических систем с интервальными параметрами

Представлен параллельный алгоритм параметрической идентификации динамических систем с интервальными параметрами. В основе этого алгоритма лежит ранее разработанный, обоснованный и апробированный алгоритм адаптивной интерполяции, который позволяет в явном виде получать зависимость состояний динамической системы от интервальных параметров. Решение задачи параметрической идентификации сводится к задаче минимизации определенной целевой функции в пространстве границ интервальных оценок параметров. За счет применения алгоритма адаптивной интерполяции при вычислении градиента целевой функции нет необходимости в дополнительном анализе и моделировании исходной динамической системы, поэтому для оптимизации удобно использовать методы первого порядка. Однако задача вычисления целевой функции и градиента включает в себя набор задач условной минимизации для явных функций, решение которых можно выполнить независимо друг от друга. Рассмотрены основные аспекты и особенности распараллеливания и реализации алгоритма параметрической идентификации и выполнена его апробация на нескольких представительных примерах. Проанализированы ускорение и эффективность распараллеливания.

Ключевые слова: распараллеливание, OpenMP, алгоритм параметрической идентификации, алгоритм адаптивной интерполяции, многомерная интерполяция, интервальные системы обыкновенных дифференциальных уравнений, градиентные методы, оптимизация

Введение

Обратные задачи играют важную роль в разных современных прикладных областях [1–3]. Задача параметрической идентификации возникает на этапе, когда математическая модель процесса уже определена, но неизвестными остаются ее параметры, которые необходимо подобрать так, чтобы модель наилучшим образом воспроизводила имеющиеся экспериментальные данные.

Данные, полученные во время эксперимента, как правило имеют некоторый разброс, поэтому при построении математической модели обычно преследуется цель минимизации отклонения модельных значений от экспериментальных.

Выделим работы [4–6], посвященные интервальному аппарату. Ключевое отличие интервальных моделей от обычных заключается в том, что они дают интервальные оценки на интересующие величины ("коридоры" возможных значений). Поэтому в случае с параметрической идентифи-

кацией основная цель — нахождение таких интервальных параметров, при которых модельные интервальные оценки будут полностью покрывать экспериментальные данные (или минимизировать отклонение от них).

Существует ряд работ, посвященных интервальным обратным задачам [7–9]. В основном в них рассмотрены математические модели определенного класса, например линейные. В настоящей же работе описан подход [10], позволяющий работать с нелинейными динамическими системами, заданными в общем виде.

В основе подхода лежит ранее разработанный алгоритм адаптивной интерполяции для моделирования динамических систем с интервальными параметрами [11, 12]. Цель алгоритма заключается в построении для каждого момента времени кусочно-полиномиальной функции, интерполирующей зависимость состояния динамической системы от неизвестных параметров, или, другими словами, параметрического множества состояний.

Данный алгоритм относится к методам, восстанавливающим явную зависимость интересующих величин от интервальных параметров (в этой группе выделяются символьные методы [13–15] и полиномиальные методы [16, 17]). Для задач с большим числом неопределенностей были разработаны две модификации алгоритма [12, 18] на основе разреженных сеток [19–21] и тензорных поездов [22, 23]. В работах [24–26] рассмотрены различные аспекты и особенности распараллеливания и реализации алгоритма с использованием технологий CUDA [27] и OpenMP [28].

Идея описываемого подхода [10] к решению задач параметрической идентификации заключается в построении и минимизации целевой функции, зависящей от границ интервальных оценок параметров системы и характеризующей расстояние между параметрическими множествами состояний и соответствующими экспериментальными данными.

Для вычисления значения целевой функции необходимо:

- 1) выполнить моделирование динамической системы с интервальными параметрами;
- 2) вычислить расстояние между полученными параметрическими множествами и экспериментальными данными.

Первый пункт подразумевает применение алгоритма адаптивной интерполяции. В процессе его работы требуется многократно выполнять моделирование исходной системы с определенными значениями неизвестных параметров, что является независимыми подзадачами, которые можно решать одновременно друг с другом. Поэтому данная задача имеет высокий потенциал распараллеливания.

Второй пункт заключается в решении для каждого момента времени, в котором известна экспериментальная информация о фазовых переменных, задачи поиска расстояния между параметрическим множеством и исходными данными. В этом случае тоже речь идет о независимых подзадачах, которые можно решать параллельно.

В качестве технологии для распараллеливания вычислений используется OpenMP (*Open Multi-Processing*), так как она позволяет достаточно легко создавать многопоточные приложения на различных языках программирования. Программный комплекс OpenMP — это набор директив компилятора, библиотечных процедур и переменных окружения, которые предназначены для программирования многопоточных приложений на многопроцессорных системах с общей памятью [29]. Выбор технологии OpenMP обусловлен неоднородностью независимых подзадач, а также простотой и эффективностью данной технологии.

Постановка задачи

Будем рассматривать автономные дискретные динамические системы с интервальным начальным состоянием и интервальными параметрами:

$$\begin{cases} y_i^{k+1} = F_i(y_1^k, y_2^k, \dots, y_n^k, \theta_1, \theta_2, \dots, \theta_m), \\ y_i^0 \in [\underline{y}_i^0, \overline{y}_i^0], i = 1, \dots, n, \\ \theta_j \in [\underline{\theta}_j, \overline{\theta}_j], j = 1, \dots, m, \\ k = 0, \dots, N-1, \end{cases} \quad (1)$$

где n — размерность фазового пространства; m — число параметров; k — дискретное время; N — конечный момент времени; $\mathbf{y}^k = (y_1^k, y_2^k, \dots, y_n^k)^T$ — вектор состояния системы в момент времени k ; $\boldsymbol{\theta} = (\theta_1, \theta_2, \dots, \theta_m)^T$ — вектор параметров системы; $\underline{y}_i^0 \leq \overline{y}_i^0$, $i = 1, \dots, n$, $\underline{\theta}_j \leq \overline{\theta}_j$, $j = 1, \dots, m$ — нижние и верхние границы интервальных неопределенностей; $\mathbf{F} = (F_1, F_2, \dots, F_n)^T$ — вектор-функция, которая определяет переход системы из одного состояния в другое.

Динамическая система может быть задана в виде системы обыкновенных дифференциальных уравнений (ОДУ). В этом случае $\mathbf{F}(\mathbf{y}^k, \boldsymbol{\theta}) = \mathbf{u}(t_{k+1})$, где $\mathbf{u}(t_{k+1})$ — решение соответствующей системы ОДУ в конечный момент времени:

$$\begin{cases} \frac{du_i(t)}{dt} = f_i(u_1(t), u_2(t), \dots, u_n(t), \theta_1, \theta_2, \dots, \theta_m), \\ u_i(t_k) = y_i^k, \\ i = 1, \dots, n, \\ t \in [t_k, t_{k+1}], \end{cases}$$

у которой правая часть $\mathbf{f} = (f_1, f_2, \dots, f_n)^T$ удовлетворяет всем условиям, обеспечивающим единственность и существование решения при всех $u(t_0) \in [\underline{y}_i^0, \overline{y}_i^0]$, $i = 1, \dots, n$ и $\theta_j \in [\underline{\theta}_j, \overline{\theta}_j]$, $j = 1, \dots, m$.

В частности, $\mathbf{F}(\mathbf{y}^k, \boldsymbol{\theta})$ может реализовывать M шагов правила трех восьмых (четвертый порядок аппроксимации) [30]:

$$\begin{aligned} \mathbf{F}(\mathbf{y}^k, \boldsymbol{\theta}) &= \mathbf{u}^M, \\ \mathbf{u}^M &= \mathbf{u}^{M-1} + \frac{h}{8}(\mathbf{K}_1^{M-1} + 3\mathbf{K}_2^{M-1} + 3\mathbf{K}_3^{M-1} + \mathbf{K}_4^{M-1}), \\ \mathbf{u}^0 &= \mathbf{y}^k, \\ \mathbf{K}_1^{M-1} &= \mathbf{f}(\mathbf{u}^{M-1}, \boldsymbol{\theta}), \\ \mathbf{K}_2^{M-1} &= \mathbf{f}\left(\mathbf{u}^{M-1} + \frac{h}{3}\mathbf{K}_1^{M-1}, \boldsymbol{\theta}\right), \\ \mathbf{K}_3^{M-1} &= \mathbf{f}\left(\mathbf{u}^{M-1} + h\left[\mathbf{K}_2^{M-1} - \frac{1}{3}\mathbf{K}_1^{M-1}\right], \boldsymbol{\theta}\right), \\ \mathbf{K}_4^{M-1} &= \mathbf{f}\left(\mathbf{u}^{M-1} + h\left[\mathbf{K}_1^{M-1} - \mathbf{K}_2^{M-1} + \mathbf{K}_3^{M-1}\right], \boldsymbol{\theta}\right), \end{aligned}$$

где $h = (t_{k+1} - t_k)/M$.

Состояние системы (1) в каждый момент времени является параметрическим множеством:

$$Y^k = \left\{ \mathbf{y}^k(x_1, x_2, \dots, x_n, z_1, z_2, \dots, z_m) \left| \begin{array}{l} x_i \in [\underline{y}_i^0, \overline{y}_i^0], i = 1, \dots, n, \\ z_j \in [\underline{\theta}_j, \overline{\theta}_j], j = 1, \dots, m \end{array} \right. \right\}, \quad (2)$$

где $\mathbf{y}^k(x_1, x_2, \dots, x_n, z_1, z_2, \dots, z_m)$ задается следующей рекуррентной формулой:

$$\begin{aligned} \mathbf{y}^k(\mathbf{x}, \mathbf{z}) &= \mathbf{F}(\mathbf{y}^{k-1}(\mathbf{x}, \mathbf{z}), \mathbf{z}), \\ \mathbf{y}^0(\mathbf{x}, \mathbf{z}) &= \mathbf{x}, \end{aligned} \quad (3)$$

где $\mathbf{x} = (x_1, x_2, \dots, x_n)^T$ и $\mathbf{z} = (z_1, z_2, \dots, z_m)^T$.

Пусть известна $(N+1)$ экспериментальная точка в фазовом пространстве в различные моменты времени:

$$\hat{\mathbf{y}}^k = (\hat{y}_1^k, \hat{y}_2^k, \dots, \hat{y}_n^k)^T, \quad k = 0, \dots, N. \quad (4)$$

Задача параметрической идентификации заключается в нахождении таких границ интервальных оценок $\underline{y}_i^0 \leq y_i^0, i = 1, \dots, n, \underline{\theta}_j \leq \theta_j, j = 1, \dots, m$, при которых множество (2) будет содержать в себе все экспериментальные точки (4) или минимизировать отклонение от них. Отметим, что часть начальных условий и параметров может быть априори известна, в этом случае они не нуждаются в идентификации. Однако для удобства изложения будем предполагать, что все границы интервальных оценок являются неизвестными.

Алгоритм параметрической идентификации

Для нахождения интервальных оценок выполняется переход к задаче минимизации расстояния между модельными множествами состояний (2) системы (1) и соответствующими экспериментальными точками (4). Минимизируется целевая функция

$$\begin{aligned} J(\underline{y}_1^0, \overline{y}_1^0, \dots, \underline{y}_n^0, \overline{y}_n^0, \underline{\theta}_1, \overline{\theta}_1, \dots, \underline{\theta}_m, \overline{\theta}_m) &= \\ &= \sum_{k=0}^N \rho(Y^k, \hat{\mathbf{y}}^k), \end{aligned} \quad (5)$$

где

$$\rho(Y^k, \hat{\mathbf{y}}^k) = \min_{\mathbf{y}^k \in Y^k} \|\mathbf{y}^k - \hat{\mathbf{y}}^k\|^2 \quad (6)$$

— квадрат расстояния между множеством Y^k и точкой $\hat{\mathbf{y}}^k$.

В качестве нормы удобно использовать евклидову норму. Важную роль при построении градиента целевой функции (5) играют прообразы соответствующих точек минимумов (6) в пространстве начальных условий и параметров.

Запишем (6) следующим образом:

$$\rho(Y^k, \hat{\mathbf{y}}^k) = \min_{\substack{x_i \in [\underline{y}_i^0, \overline{y}_i^0], i=1, \dots, n, \\ z_j \in [\underline{\theta}_j, \overline{\theta}_j], j=1, \dots, m, \\ z_1, z_2, \dots, z_m}} J^k(x_1, x_2, \dots, x_n, \quad (7)$$

где

$$\begin{aligned} J^k(x_1, x_2, \dots, x_n, z_1, z_2, \dots, z_m) &= \\ &= \sum_{i=1}^n \left[y_i^k(x_1, x_2, \dots, x_n, z_1, z_2, \dots, z_m) - \hat{y}_i^k \right]^2. \end{aligned} \quad (8)$$

Отметим, что ключевым моментом является то, что $\mathbf{y}^k(x_1, x_2, \dots, x_n, z_1, z_2, \dots, z_m)$ можно получить в явном виде с помощью алгоритма адаптивной интерполяции, и при этом нет необходимости в применении рекуррентной формулы (3), которая эквивалентна моделированию исходной системы (1).

Обозначим через $(\tilde{\mathbf{x}}^k, \tilde{\mathbf{z}}^k) = (\tilde{x}_1^k, \tilde{x}_2^k, \dots, \tilde{x}_n^k, \tilde{z}_1^k, \tilde{z}_2^k, \dots, \tilde{z}_m^k)^T$ точки минимума (7):

$$\begin{aligned} &(\tilde{x}_1^k, \tilde{x}_2^k, \dots, \tilde{x}_n^k, \tilde{z}_1^k, \tilde{z}_2^k, \dots, \tilde{z}_m^k)^T = \\ &= \arg \min_{\substack{x_i \in [\underline{y}_i^0, \overline{y}_i^0], i=1, \dots, n, \\ z_j \in [\underline{\theta}_j, \overline{\theta}_j], j=1, \dots, m}} J^k(x_1, x_2, \dots, x_n, z_1, z_2, \dots, z_m). \end{aligned} \quad (9)$$

Запишем производные для (8):

$$\begin{aligned} \frac{dJ^k(\mathbf{x}, \mathbf{z})}{dx_i} &= 2 \sum_{l=1}^n \left[y_l^k(\mathbf{x}, \mathbf{z}) - y_l^k \right] \frac{dy_l^k(\mathbf{x}, \mathbf{z})}{dx_i}, \\ &\quad i = 1, \dots, n, \\ \frac{dJ^k(\mathbf{x}, \mathbf{z})}{dz_j} &= 2 \sum_{l=1}^n \left[y_l^k(\mathbf{x}, \mathbf{z}) - y_l^k \right] \frac{dy_l^k(\mathbf{x}, \mathbf{z})}{dz_j}, \\ &\quad j = 1, \dots, m. \end{aligned} \quad (10)$$

Так как $(\tilde{x}_1^k, \tilde{x}_2^k, \dots, \tilde{x}_n^k, \tilde{z}_1^k, \tilde{z}_2^k, \dots, \tilde{z}_m^k)^T$ являются точками минимума, то

$$\frac{dJ^k(\tilde{x}_1^k, \tilde{x}_2^k, \dots, \tilde{x}_n^k, \tilde{z}_1^k, \tilde{z}_2^k, \dots, \tilde{z}_m^k)}{dx_i} \begin{cases} \leq 0, \tilde{x}_i^k = \overline{y_i^0}, \\ = 0, \tilde{x}_i^k \in (\underline{y_i^0}, \overline{y_i^0}), i = 1, \dots, n, \\ \geq 0, \tilde{x}_i^k = \underline{y_i^0}, \end{cases}$$

$$\frac{dJ^k(\tilde{x}_1^k, \tilde{x}_2^k, \dots, \tilde{x}_n^k, \tilde{z}_1^k, \tilde{z}_2^k, \dots, \tilde{z}_m^k)}{dz_j} \begin{cases} \leq 0, \tilde{z}_j^k = \overline{\theta_j}, \\ = 0, \tilde{z}_j^k \in (\underline{\theta_j}, \overline{\theta_j}), j = 1, \dots, m. \\ \geq 0, \tilde{z}_j^k = \underline{\theta_j}, \end{cases} \quad (11)$$

Если точки $(\tilde{x}_1^k, \tilde{x}_2^k, \dots, \tilde{x}_n^k, \tilde{z}_1^k, \tilde{z}_2^k, \dots, \tilde{z}_m^k)^T$ лежат внутри области неопределенности

$$\chi = [\underline{y_1^0}, \overline{y_1^0}] \times [\underline{y_2^0}, \overline{y_2^0}] \times \dots \times [\underline{y_n^0}, \overline{y_n^0}] \times [\underline{\theta_1}, \overline{\theta_1}] \times [\underline{\theta_2}, \overline{\theta_2}] \times \dots \times [\underline{\theta_m}, \overline{\theta_m}],$$

то производные целевой функции по $\underline{y_1^0}, \overline{y_1^0}, \dots, \underline{y_n^0}, \overline{y_n^0}, \underline{\theta_1}, \overline{\theta_1}, \dots, \underline{\theta_m}, \overline{\theta_m}$ будут равны нулю, а если на границе $\partial\chi$, то — соответствующим производным (11). С учетом (10) и (11) в результате получаем следующие компоненты градиента для целевой функции (5):

$$\frac{dJ}{d\underline{y_i^0}} = 2 \sum_{k=0}^N \max \left(0, \sum_{l=1}^n [y_l^k(\tilde{\mathbf{x}}^k, \tilde{\mathbf{z}}^k) - y_l^k] \frac{dy_l^k(\tilde{\mathbf{x}}^k, \tilde{\mathbf{z}}^k)}{dx_i} \right),$$

$$\frac{dJ}{d\overline{y_i^0}} = 2 \sum_{k=0}^N \min \left(0, \sum_{l=1}^n [y_l^k(\tilde{\mathbf{x}}^k, \tilde{\mathbf{z}}^k) - y_l^k] \frac{dy_l^k(\tilde{\mathbf{x}}^k, \tilde{\mathbf{z}}^k)}{dx_i} \right), i = 1, \dots, n,$$

$$\frac{dJ}{d\underline{\theta_j}} = 2 \sum_{k=0}^N \max \left(0, \sum_{l=1}^m [y_l^k(\tilde{\mathbf{x}}^k, \tilde{\mathbf{z}}^k) - y_l^k] \frac{dy_l^k(\tilde{\mathbf{x}}^k, \tilde{\mathbf{z}}^k)}{dz_j} \right),$$

$$\frac{dJ}{d\overline{\theta_j}} = 2 \sum_{k=0}^N \min \left(0, \sum_{l=1}^m [y_l^k(\tilde{\mathbf{x}}^k, \tilde{\mathbf{z}}^k) - y_l^k] \frac{dy_l^k(\tilde{\mathbf{x}}^k, \tilde{\mathbf{z}}^k)}{dz_j} \right), j = 1, \dots, m. \quad (12)$$

Так как согласно (12) ширина получающихся интервальных неопределенностей не будет уменьшаться, то дополнительно определим правила сужения интервалов:

$$\underline{y_i^0} = \min_{k=0, \dots, N} (\tilde{x}_i^k), \text{ если } \frac{dJ}{d\underline{y_i^0}} = 0,$$

$$\overline{y_i^0} = \max_{k=0, \dots, N} (\tilde{x}_i^k), \text{ если } \frac{dJ}{d\overline{y_i^0}} = 0, i = 1, \dots, n,$$

$$\underline{\theta_j} = \min_{k=0, \dots, N} (\tilde{z}_j^k), \text{ если } \frac{dJ}{d\underline{\theta_j}} = 0,$$

$$\overline{\theta_j} = \max_{k=0, \dots, N} (\tilde{z}_j^k), \text{ если } \frac{dJ}{d\overline{\theta_j}} = 0, j = 1, \dots, m. \quad (13)$$

$$\underline{y_i^{0(it+1)}} = \underline{y_i^{0(it)}} - \lambda^{(it)} \frac{dJ(\underline{y_1^{0(it)}}, \overline{y_1^{0(it)}} , \dots, \underline{\theta_1^{(it)}}, \overline{\theta_1^{(it)}} , \dots)}{d\underline{y_i^0}},$$

$$\overline{y_i^{0(it+1)}} = \overline{y_i^{0(it)}} - \lambda^{(it)} \frac{dJ(\underline{y_1^{0(it)}}, \overline{y_1^{0(it)}} , \dots, \underline{\theta_1^{(it)}}, \overline{\theta_1^{(it)}} , \dots)}{d\overline{y_i^0}},$$

$$\underline{\theta_j^{(it+1)}} = \underline{\theta_j^{(it)}} - \lambda^{(it)} \frac{dJ(\underline{y_1^{0(it)}}, \overline{y_1^{0(it)}} , \dots, \underline{\theta_1^{(it)}}, \overline{\theta_1^{(it)}} , \dots)}{d\underline{\theta_j}},$$

$$\overline{\theta_j^{(it+1)}} = \overline{\theta_j^{(it)}} - \lambda^{(it)} \frac{dJ(\underline{y_1^{0(it)}}, \overline{y_1^{0(it)}} , \dots, \underline{\theta_1^{(it)}}, \overline{\theta_1^{(it)}} , \dots)}{d\overline{\theta_j}},$$

$$j = 1, \dots, m,$$

Далее для поиска неизвестных границ интервальных оценок можно использовать методы оптимизации первого порядка [31]. Одна итерация метода градиентного спуска записывается следующим образом:

где $\lambda^{(it)}$ — скорость градиентного спуска.

После каждой итерации применяются правила (13). Начальное приближение $\underline{y}_i^{0(0)} < \overline{y}_i^{0(0)}$, $\underline{\theta}_j^{(0)} < \overline{\theta}_j^{(0)}$ задается произвольным образом, однако оно может сразу удовлетворить условию $\hat{\mathbf{y}}^0 \in Y^0$. Процесс поиска останавливается, когда $J \leq \varepsilon$ или когда $\|\nabla J\| \leq \delta$, где ε и δ — наперед заданные положительные числа, характеризующие требуемую точность.

Распараллеливание и реализация

Для того чтобы распараллелить работу алгоритма, его достаточно представить в виде набора независимых подзадач.

В рамках одной итерации градиентного спуска необходимо решить перечисленные далее задачи.

1. Решить прямую задачу (1). Для этого применяется параллельный алгоритм адаптивной интерполяции на основе разреженных сеток с нелинейным базисом, описанный в работе [25]. В результате будет получена вектор-функция $\mathbf{P}^k(\mathbf{x}, \mathbf{z})$, интерполирующая $\mathbf{y}^k(\mathbf{x}, \mathbf{z})$, $k = 0, \dots, N$.

В выражение (12) вместо $\mathbf{y}^k(\mathbf{x}, \mathbf{z})$ подставляется

$\mathbf{P}^k(\mathbf{x}, \mathbf{z})$, а вместо $\frac{dy_i^k(\mathbf{x}, \mathbf{z})}{dx_i}$ и $\frac{dy_i^k(\mathbf{x}, \mathbf{z})}{dz_j}$ подставляются $\frac{dP_i^k(\mathbf{x}, \mathbf{z})}{dx_i}$ и $\frac{dP_i^k(\mathbf{x}, \mathbf{z})}{dz_j}$, $i = 1, \dots, n$,

$i = 1, \dots, n$, $j = 1, \dots, m$, которые вычисляются аналитически.

2. Решить $(N + 1)$ задач (9) поиска минимума для явной функции:

$$(\tilde{\mathbf{x}}^k, \tilde{\mathbf{z}}^k)^T = \arg \min_{\substack{x_i \in [\underline{y}_i^0, \overline{y}_i^0], i=1, \dots, n, \\ z_j \in [\underline{\theta}_j, \overline{\theta}_j], j=1, \dots, m}} \sum_{i=1}^n [P_i^k(\mathbf{x}, \mathbf{z}) - \hat{y}_i^k]^2. \quad (14)$$

Данные задачи являются независимыми друг от друга, поэтому их можно решать одновременно. Отметим, что в зависимости от исходной системы (1) для разных k может потребоваться разное вычислительное время. Поэтому при распараллеливании соответствующего цикла с помощью OpenMP необходимо

1. $\mathbf{P}^0 : [\underline{y}_1^0, \overline{y}_1^0] \times \dots \times [\underline{y}_n^0, \overline{y}_n^0] \times [\underline{\theta}_1, \overline{\theta}_1] \times \dots \times [\underline{\theta}_m, \overline{\theta}_m] \rightarrow \mathbb{R}^n$
2. $\mathbf{P}^0(\mathbf{x}, \mathbf{z}) = \mathbf{x}$
3. for $k = 1, \dots, N$:
4. $\mathbf{P}^k = \text{parallel buildPolynomial}(\mathbf{P}^{k-1}, \mathbf{F})$
5. parallel for $k = 0, \dots, N$:
6. $(\tilde{\mathbf{x}}^k, \tilde{\mathbf{z}}^k) = \arg \min \left(\sum_{i=1}^n [P_i^k(\mathbf{x}, \mathbf{z}) - \hat{y}_i^k]^2 \right)$
7. $\frac{dJ^k}{dx_i} = 2 \sum_{l=1}^n [P_l^k(\tilde{\mathbf{x}}^k, \tilde{\mathbf{z}}^k) - \hat{y}_l^k] \frac{dP_l^k(\tilde{\mathbf{x}}^k, \tilde{\mathbf{z}}^k)}{dx_i}, i = 1, \dots, n,$
8. $\frac{dJ^k}{dz_j} = 2 \sum_{l=1}^n [P_l^k(\tilde{\mathbf{x}}^k, \tilde{\mathbf{z}}^k) - \hat{y}_l^k] \frac{dP_l^k(\tilde{\mathbf{x}}^k, \tilde{\mathbf{z}}^k)}{dz_j}, j = 1, \dots, m.$
9. $J = \sum_{k=0}^N \sum_{i=1}^n [P_i^k(\tilde{\mathbf{x}}^k, \tilde{\mathbf{z}}^k) - \hat{y}_i^k]^2$
10. $\frac{dJ}{d\underline{y}_i^0} = 2 \sum_{k=0}^N \max \left(0, \frac{dJ^k}{dx_i} \right), \frac{dJ}{d\overline{y}_i^0} = 2 \sum_{k=0}^N \min \left(0, \frac{dJ^k}{dx_i} \right), i = 1, \dots, n,$
11. $\frac{dJ}{d\underline{\theta}_j} = 2 \sum_{k=0}^N \max \left(0, \frac{dJ^k}{dz_j} \right), \frac{dJ}{d\overline{\theta}_j} = 2 \sum_{k=0}^N \min \left(0, \frac{dJ^k}{dz_j} \right), j = 1, \dots, m.$

Рис. 1. Псевдокод вычисления значений целевой функции и ее градиента

применять директиву `schedule(dynamic)`, чтобы потоки динамически распределяли задачи между собой.

На рис. 1 представлен псевдокод вычисления значения целевой функции и ее градиента. На входе: $\underline{y}_i^0 \leq \overline{y}_i^0$, $\underline{\theta}_j \leq \overline{\theta}_j$, $i = 1, \dots, n$, $j = 1, \dots, m$ — нижние и верхние границы интервальных неопределенностей. На выходе: J , $\frac{dJ}{d\underline{y}_i^0}$, $\frac{dJ}{d\overline{y}_i^0}$, $\frac{dJ}{d\underline{\theta}_j}$, $\frac{dJ}{d\overline{\theta}_j}$, $i = 1, \dots, n$, $j = 1, \dots, m$ — значение целевой функции и компонент градиента.

Функция `buildPolynomial` (\mathbf{P}^{k-1} , \mathbf{F}) строит полином $\mathbf{P}^k(\mathbf{x}, \mathbf{z})$, который интерполирует $\mathbf{F}(\mathbf{P}^{k-1}(\mathbf{x}, \mathbf{z}), \mathbf{z})$, и является основой алгоритма адаптивной интерполяции.

Начинать выполнять цикл `parallel for` $k = 0, \dots, N$ можно не ожидая завершения предыдущего цикла, необходимо только, чтобы по переменной k не было опережения.

Отметим, что задача (9) допускает множество решений, в этом случае с точки зрения ширины получающихся интервальных оценок целесообразно брать решение, находящееся ближе к центру текущей области неопределенности:

$$\left(\frac{\overline{y}_1^0 + \underline{y}_1^0}{2}, \dots, \frac{\overline{y}_n^0 + \underline{y}_n^0}{2}, \frac{\overline{\theta}_1 + \underline{\theta}_1}{2}, \dots, \frac{\overline{\theta}_m + \underline{\theta}_m}{2} \right)^T.$$

На практике целесообразно найти несколько частных решений и взять ближайшее из них.

Результаты

Выполним решение задачи параметрической идентификации для двух систем ОДУ с интервальными неопределенностями. Параметры остановки метода градиентного спуска: $\varepsilon = \delta = 10^{-12}$. В качестве экспериментальных точек использовались квазиэкспериментальные точки (искусственно сгенерированные определенным образом точки):

$$\hat{\mathbf{y}}^k = \mathbf{P}^k(\hat{x}_1^k, \hat{x}_2^k, \dots, \hat{x}_n^k, \hat{z}_1^k, \hat{z}_2^k, \dots, \hat{z}_m^k),$$

где $\hat{x}_i^k = \text{rand}[\underline{y}_i^0, \overline{y}_i^0]$, $i = 1, \dots, n$; $\hat{z}_j^k = \text{rand}[\underline{\theta}_j, \overline{\theta}_j]$, $j = 1, \dots, m$ — равномерно распределенные на априори заданных интервалах случайные величины; \mathbf{P}^k — решение прямой задачи, полученное с помощью алгоритма адаптивной интерполяции.

Для каждой задачи оцениваются ускорение $S = T_1/T_p$ и эффективность $E = S/P$, где T_1 — время работы последовательной версии алгоритма; T_p — время работы параллельной версии алгоритма с использованием P вычислительных потоков.

Характеристики используемой вычислительной машины: Intel(R) Xeon(R) CPU E5-

2620 v4@2.10GHz, оперативная память 8x32 GiB DDR4 2666 MHz, режим работы процессора с памятью — двухканальный, число вычислительных ядер с гипертренингом — 8.

Вначале рассмотрим систему ОДУ с двумя интервальными параметрами:

$$\begin{cases} u' = -\frac{\alpha v}{1/2 + \sqrt{\sin^2(u) + v^2}}, \\ v' = \frac{\beta u}{1/2 + \sqrt{u^2 + \cos^2(v)}}, \\ u(0) = 5, v(0) = 0, \\ \alpha \in [\underline{\alpha}, \overline{\alpha}], \beta \in [\underline{\beta}, \overline{\beta}], \\ t \in [0, 60], \end{cases} \quad (15)$$

где $\underline{\alpha}$, $\overline{\alpha}$, $\underline{\beta}$, $\overline{\beta}$ — неизвестные границы интервальных параметров, которые подлежат определению.

Для генерации квазиэкспериментальных данных из интервала интегрирования были взяты 100 точек с постоянным шагом по времени и использовались значения параметров $\alpha \in [0,975, 1,025]$ и $\beta \in [0,900, 1,100]$. На рис. 2 показаны полученные точки на фазовой плоскости.

Начальное приближение в методе градиентного спуска: $\alpha^{(0)} \in [0,70, 0,75]$ и $\beta^{(0)} \in [0,70, 0,75]$. На рис. 3 продемонстрирован процесс решения задачи. Серым цветом показаны получающиеся модельные интервальные оценки фазовых переменных, а черным — экспериментальные данные. Найденные значения параметров $\alpha^{(29)} \in [0,975, 1,025]$ и $\beta^{(29)} \in [0,900, 1,100]$ совпадают с исходными.

На рис. 4 серым цветом показаны полученные параметрические множества Y^k , $k = 1, 3, 5, \dots, 43$, на последней итерации градиентного спуска.

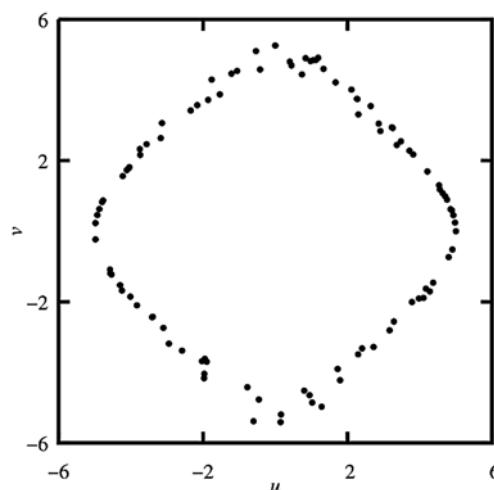


Рис. 2. Квазиэкспериментальные точки на фазовой плоскости

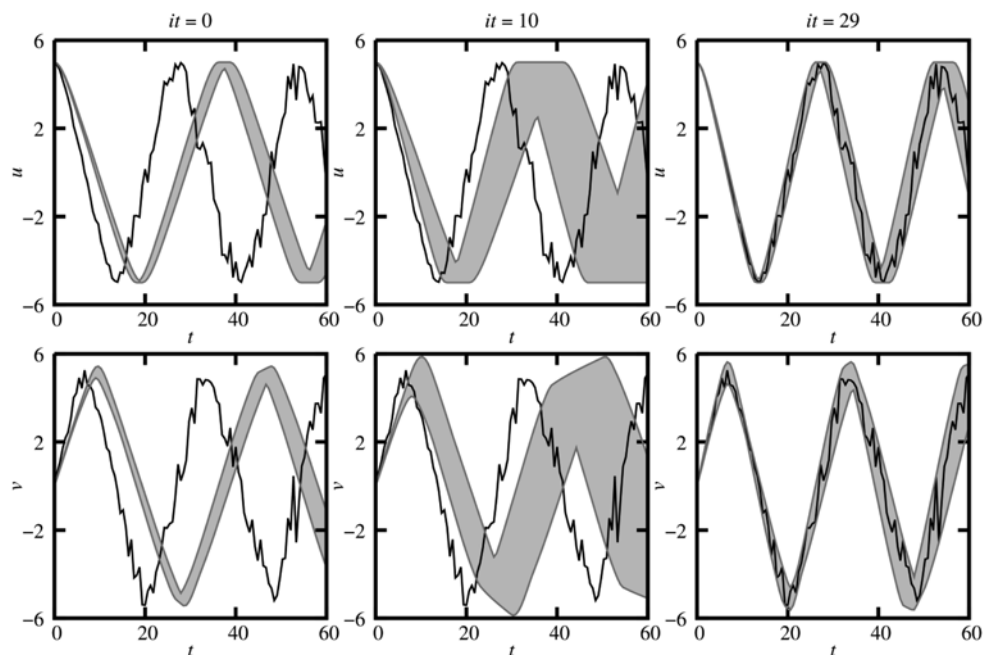


Рис. 3. Иллюстрация решения задачи параметрической идентификации системы (15)

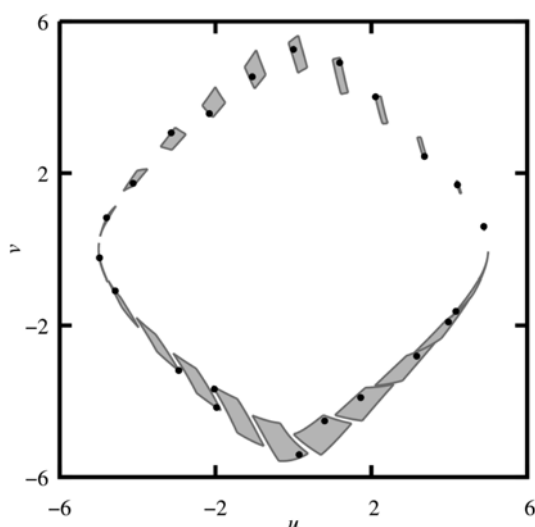


Рис. 4. Некоторые множества состояний системы (15) на 29-й итерации

На рис. 4 можно видеть, что все экспериментальные точки принадлежат соответствующим множествам.

На рис. 5 представлены графики зависимости ускорения S и эффективности E от числа вычислительных потоков P . За счет распараллеливания удалось сократить вычислительное время в 5 раз. Получение небольшого коэффициента распараллеливания связано, во-первых, с тем, что реальных ядер 8, во-вторых, с особенностями распараллеливания алгоритма адаптивной интерполяции, описанными в работе [25], а в-третьих, с неоднородностью задач (14) для разных k .

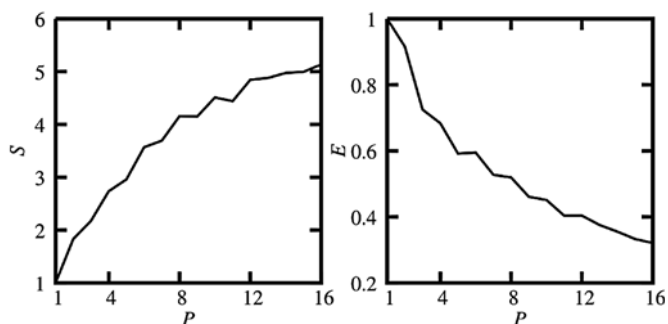


Рис. 5. Ускорение и эффективность распараллеливания при решении задачи параметрической идентификации системы (15)

Процесс вычисления вектор-функции $\mathbf{P}^k(\mathbf{x}, \mathbf{z})$ подразумевает частое обращение к памяти, и при числе потоков более 16 будет наблюдаться увеличение времени работы программы, связанное с особенностями взаимодействия с оперативной памятью.

Далее рассмотрим систему ОДУ, соответствующую модели осциллятора Дуффинга [32] с интервальными двумя начальными условиями и одним параметром:

$$\begin{cases} u' = v, \\ v' = u - \frac{1}{4}v - u^3 + \alpha \cos(\tau), \\ \tau' = 1, \\ \tau(0) = 0, \\ u(0) = u_0 \in [\underline{u}_0, \overline{u}_0], \\ v(0) = v_0 \in [\underline{v}_0, \overline{v}_0], \alpha \in [\underline{\alpha}, \overline{\alpha}], \\ t \in [0, 6]. \end{cases} \quad (16)$$

Как и в предыдущем примере, для генерации квазиэкспериментальных данных были взяты 100 точек с постоянным шагом по времени. Исходные значения начальных условий и параметра: $u_0 \in [1,90, 2,10]$, $v_0 \in [1,90, 2,10]$, $\alpha \in [0,27, 0,33]$. Начальное приближение: $u_0^{(0)} \in [1,50, 1,60]$, $v_0^{(0)} \in [1,60, 1,70]$, $\alpha^{(0)} \in [0,48, 0,78]$. На рис. 6 показаны зависимости интервальных оценок фазовых переменных от времени для различных итераций it градиентного спуска. На рис. 7 представлены полученные параметрические множества Y^k , $k = 1, 5, 9, \dots, 97$, на фазовой плоскости и соответствующие экспериментальные точки.

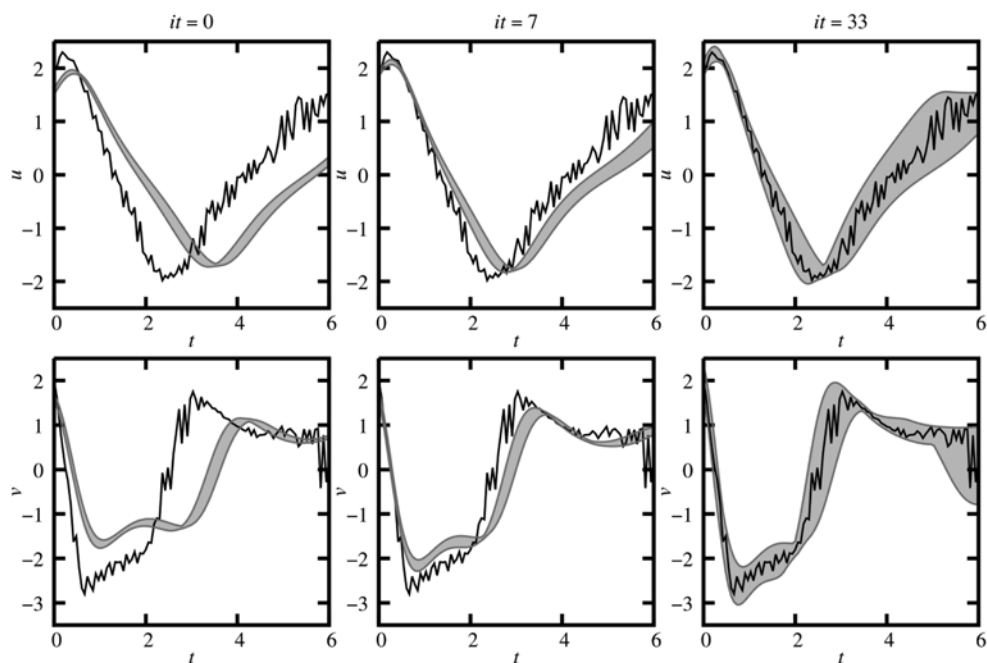


Рис. 6. Зависимости интервальных оценок фазовых переменных от времени в процессе параметрической идентификации системы (16)

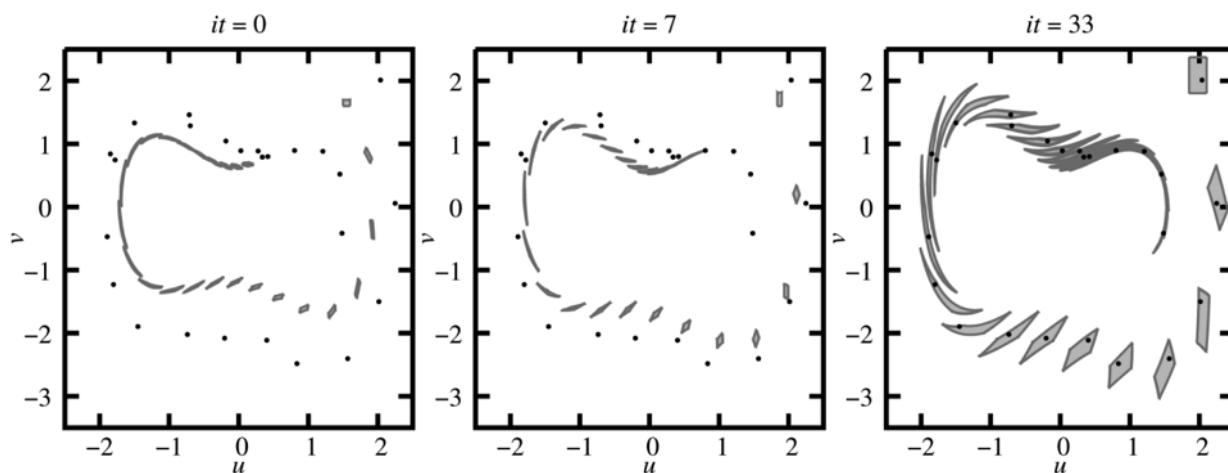


Рис. 7. Иллюстрация процесса решения задачи параметрической идентификации для системы (16) на фазовой плоскости

На 33-й итерации полученные параметрические множества Y^k полностью содержат в себе экспериментальные точки: $J^{(33)} = 0$. Найденные интервальные оценки: $u_0^{(33)} \in [1,85, 2,11]$, $v_0^{(33)} \in [1,80, 2,37]$, $\alpha^{(33)} \in [0,46, 0,78]$. Их отличие от исходных оценок связано с многоэкстремальностью целевой функции.

На рис. 8 показаны графики зависимости ускорения S и эффективности E от числа вычислительных потоков P . Максимальное достигнутое ускорение — более 6 раз.

Алгоритм параметрической идентификации условно состоит из двух частей — решения прямой интервальной задачи и решения множества

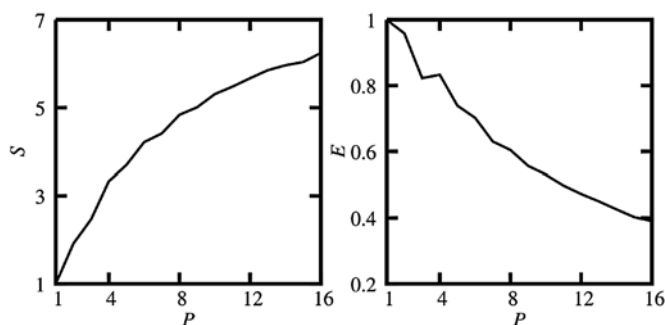


Рис. 8. Ускорение и эффективность распараллеливания при решении задачи параметрической идентификации системы (16)

задач минимизации для явных функций. Благодаря распараллеливанию данных операций можно существенно сократить время работы алгоритма.

Заключение

В работе рассматривается алгоритм параметрической идентификации интервальных динамических систем с позиции распараллеливания. Поиск неизвестных границ интервальных параметров сводится к решению задачи минимизации. При этом основные вычислительные затраты связаны с получением значения целевой функции и ее градиента. Вычисление целевой функции происходит в два этапа. Сначала решается прямая задача моделирования динамической системы с интервальными параметрами с помощью алгоритма адаптивной интерполяции. После этого решается множество независимых задач минимизации для явных функций. Оба этапа допускают распараллеливание. При апробации параллельной реализации алгоритма на двух системах ОДУ получено ускорение в 5–6 раз.

Список литературы

1. Nenarokomov A. V., Alifanov O. M., Krainova I. V., Titov D. M., Morzhukhina A. V. Estimation of environmental influence on spacecraft materials radiative properties by inverse problems technique // *Acta Astronautica*. 2019. Vol. 160. P. 323–330. DOI: 10.1016/j.actaastro.2019.04.014.
2. Кабанихин С. И., Куликов И. М., Шишленин М. А. Алгоритм восстановления характеристик начального состояния сверхновой звезды // *Журнал вычислительной математики и математической физики*. 2020. Том 60, № 6. С. 1035–1044. DOI: 10.31857/S0044466920060137.
3. Абгарян К. К., Носков Р. Г., Ревизников Д. Л. Обратная коэффициентная задача теплопереноса в слоистых наноструктурах // *Известия высших учебных заведений. Материалы электронной техники*. 2017. Том 20, № 3. С. 213–219. DOI: 10.17073/1609-3577-2017-3-213-219.
4. Moore R. E., Kearfott R. B., Cloud M. J. *Introduction to Interval Analysis*, SIAM, 2009. 223 p.
5. Шарый С. П. Конечномерный интервальный анализ. Новосибирск: Институт вычислительных технологий СО РАН. Изд. XYZ, 2022. 653 с.

6. Добронеев Б. С. *Интервальная математика*. Красноярск: Краснояр. гос. ун-т, 2004. 216 с.
7. Дилигенская А. Н., Самокиш А. В. Параметрическая идентификация в обратных задачах теплопроводности в условиях интервальной неопределенности на основе нейронных сетей // *Вестник Самарского государственного технического университета*. 2020. Т. 28. № 4 (68). С. 6–18.
8. Петрикевич Я. И. Структурно-параметрическая идентификация динамических объектов по интервальным исходным данным: дис. ... канд. техн. наук: 05.13.18. Кемерово: Кемеровский государственный университет, 2006. 225 с.
9. Xiao N., Fedele F., Muhanna R. L. Inverse Problems Under Uncertainties-An Interval Solution for the Beam Finite Element // *Conference Paper: 11th International Conference on Structural Safety & Reliability*, New York, NY, USA, 2013. 8 p. URL: <https://www.researchgate.net/publication/269518192>. DOI: 10.1201/b16387-430.
10. Морозов А. Ю., Ревизников Д. Л. Интервальный подход к решению задач параметрической идентификации динамических систем // *Дифференциальные уравнения*. 2022. Том 58, № 7. С. 962–976. DOI: 10.31857/S0374064122070081.
11. Морозов А. Ю., Ревизников Д. Л., Гидаспов В. Ю. Алгоритм адаптивной интерполяции на основе kd-дерева для решения задач химической кинетики с интервальными параметрами // *Математическое моделирование*. 2018. Т. 30. № 12. С. 129–144. DOI: 10.31857/S023408790001940-8.
12. Морозов А. Ю., Ревизников Д. Л. Алгоритм адаптивной интерполяции на разреженных сетках для численного интегрирования систем обыкновенных дифференциальных уравнений с интервальными неопределенностями // *Дифференциальные уравнения*. 2021. Т. 57. № 7. С. 976–987. DOI: 10.31857/S0374064121070104.
13. Makino K., Berz M. *Models and Their Applications // Numerical Software Verification 2017: conference*. Heidelberg, Germany, July 22–23, 2017. Springer International Publishing AG, 2017. P. 3–13.
14. Nataraj P. S. V., Sundur S. The Extrapolated Taylor Model // *Reliable Computing*. July, 2011. P. 251–278.
15. Рогалев А. Н. Гарантированные методы решения систем обыкновенных дифференциальных уравнений на основе преобразования символьных формул // *Вычислительные технологии*. 2003. Том 8, № 5. С. 102–116.
16. Fu C., Ren X., Yang Y.-F. et al. Steady-state response analysis of cracked rotors with uncertain but bounded parameters using a polynomial surrogate method // *Commun. Nonlinear Sci. Numer. Simul.* 2019. Vol. 68. P. 240–256. DOI: 10.1016/j.cnsns.2018.08.004.
17. Fu C., Xu Y., Yang Y. et al. Response analysis of an accelerating unbalanced rotating system with both random and interval variables // *J. Sound Vib.* 2020. Vol. 466. Article 115047. DOI: 10.1016/j.jsv.2019.115047.
18. Гидаспов В. Ю., Морозов А. Ю., Ревизников Д. Л. Алгоритм адаптивной интерполяции с использованием ТТ-разложения для моделирования динамических систем с интервальными параметрами // *Журнал вычислительной математики и математической физики*. 2021. Том 61, № 9. С. 1416–1430. DOI: 10.31857/S0044466921090106.
19. Смоляк С. А. Квадратурные и интерполяционные формулы на тензорных произведениях некоторых классов функций // *Докл. АН СССР*, 1963. Том 148, № 5. С. 1042–1045.
20. Bungartz H.-J., Griebel M. Sparse grids // *Acta Numerica*. 2004. Vol. 13, No. 1. P. 147–269.
21. Gerstner T., Griebel M. Sparse grids // *Encyclopedia of Quantitative Finance*. 2008. Vol. 13. P. 5.
22. Oseledets I. V. Tensor-train decomposition // *SIAM Journal on Scientific Computing*. 2011. Vol. 33, No. 5. P. 2295–2317. DOI: 10.1137/090752286.
23. Oseledets I., Tyrtshnikov E. TT-cross approximation for multidimensional arrays // *Linear Algebra and its Applications*. 2010. Vol. 432, Iss. 1. P. 70–88. DOI: 10.1016/j.laa.2009.07.024.
24. Morozov A. Yu., Reviznikov D. L. Modelling of Dynamic Systems with Interval Parameters on Graphic Processors // *Про-*

граммная инженерия. 2019. Том 10, № 2. С. 69–76. DOI: 10.17587/prin.10.69-76.

25. Морозов А. Ю. Параллельный алгоритм адаптивной интерполяции на основе разреженных сеток для моделирования динамических систем с интервальными параметрами // Программная инженерия. 2021. Том 12, № 8. С. 395–403. DOI: 10.17587/prin.12.395-403.

26. Капралов Н. С., Морозов А. Ю., Никулин С. П. Параллельная аппроксимация многомерных тензоров с использованием графических процессоров // Программная инженерия. 2022. Том 13, № 2. С. 94–101. DOI: 10.17587/prin.13.94-101.

27. CUDA Zone. URL: <https://developer.nvidia.com/cuda-zone>

28. OpenMP. URL: <https://www.openmp.org/>

29. OpenMP technology. URL: <https://pvs-studio.com/ru/a/0057>

30. Kutta M. Beitrag zur näherungsweise Integration totaler Differentialgleichungen // Zeitschrift für Mathematik und Physik. 1901. Vol. 46. P. 435–453.

31. Гилл Ф., Мюррей У., Райт М. Практическая оптимизация. М.: Мир, 1985. 509 с.

32. Kovacic I., Brennan M. J. The Duffing Equation: Nonlinear Oscillators and their Behaviour. John Wiley & Sons, Hoboken, 2011. 369 p. DOI: 10.1002/9780470977859.

Parallel Algorithm for Parametric Identification of Dynamical Systems with Interval Parameters

A. Yu. Morozov, morozov@infway.ru,

Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences (FRC CSC RAS), Moscow, 119333, Russian Federation

Corresponding author:

Alexander Yu. Morozov, Researcher,

Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences (FRC CSC RAS), Moscow, 119333, Russian Federation

E-mail: morozov@infway.ru

Received on July 28, 2022

Accepted on August 10, 2022

The paper presents a parallel algorithm for the parametric identification of dynamical systems with interval parameters. The algorithm is based on the previously developed, substantiated and tested adaptive interpolation algorithm, which makes it possible to explicitly obtain the dependence of the states of a dynamic system on interval parameters. The solution of the problem of parametric identification is reduced to the problem of minimizing a certain objective function in the space of boundaries of interval parameter estimates. Due to the use of the adaptive interpolation algorithm when calculating the gradient of the objective function, there is no need for additional analysis and modeling of the original dynamic system, so it is convenient to use first-order methods for optimization. However, the task of calculating the objective function and the gradient includes a set of conditional minimization problems for explicit functions that can be solved independently of each other. The article discusses the main aspects and features of parallelization and implementation of the parametric identification algorithm and tests it on several representative examples. The acceleration and efficiency of parallelization are analyzed.

Keywords: parallelization, OpenMP, parametric identification algorithm, adaptive interpolation algorithm, multidimensional interpolation, interval systems of ordinary differential equations, gradient methods, optimization

For citation:

Morozov A. Yu. Parallel Algorithm for Parametric Identification of Dynamical Systems with Interval Parameters, *Programmnaya Ingeneria*, 2022, vol. 13, no. 10, pp. 497–507.

DOI: 10.17587/prin.13.497-507

References

1. Nenarokomov A. V., Alifanov O. M., Krainova I. V. et al. Estimation of environmental influence on spacecraft materials radiative properties by inverse problems technique, *Acta Astronautica*, 2019, vol. 160, pp. 323–330. DOI: 10.1016/j.actaastro.2019.04.014.

2. Kabanikhin S. I., Kulikov I. M., Shishlenin M. A. An Algorithm for Recovering the Characteristics of the Initial State of Supernova, *Computational Mathematics and Mathematical Physics*. 2020, vol. 60, pp. 1008–1016. DOI: 10.1134/S0965542520060135.

3. Abgarian K. K., Noskov R. G., Reviznikov D. L. The inverse coefficient problem of heat transfer in layered nanostructures, *Materials of Electronics Engineering*. 2017, vol. 20, no. 3, pp. 213–219. DOI: 10.17073/1609-3577-2017-3-213-219 (in Russian).

4. Moore R. E., Kearfott R. B., Cloud M. J. *Introduction to Interval Analysis*, SIAM, 2009, 223 p.

5. Shary S. P. *Finite dimensional interval analysis*. Novosibirsk, Institute of Computational Technologies SB RAS, XYZ Publisher, 2022, 653 p. (in Russian).

6. Dobronets B. S. *Interval Mathematics*, Krasnoyarsk State University, Krasnoyarsk, 2004, 216 p. (in Russian).

7. **Diligenskaya A. N., Samokish A. V.** Parametric identification in inverse heat conduction problems under conditions of interval uncertainty based on neural networks, *Bulletin of the Samara State Technical University*, 2020, vol. 28, no. 4 (68), pp. 6–18 (in Russian).

8. **Petrikevich Ya. I.** Structural-parametric identification of dynamic objects by interval initial data: dis. cand. tech. Sciences: 05.13.18. Kemerovo: Kemerovo State University, 2006. 225 p. (in Russian).

9. **Xiao N., Fedele F., Muhanna R. L.** Inverse Problems Under Uncertainties—An Interval Solution for the Beam Finite Element, *Conference Paper: 11th International Conference on Structural Safety & Reliability*, New York, NY, USA, 2013. 8 p. available at: <https://www.researchgate.net/publication/269518192>. DOI: 10.1201/b16387-430.

10. **Morozov A. Yu., Reviznikov D. L.** Interval approach to solving problems of parametric identification of dynamical systems, *Differential Equations*, 2022. Vol. 58, No. 7. pp. 962–976. DOI: 10.31857/S0374064122070 (in Russian).

11. **Morozov A. Yu., Reviznikov D. L., Gidasov V. Yu.** Adaptive Interpolation Algorithm Based on a KD-Tree for the Problems of Chemical Kinetics with Interval Parameters, *Mathematical Models and Computer Simulations*, 2019, vol. 11, no. 4, pp. 622–633. DOI: 10.1134/S2070048219040100.

12. **Morozov A. Yu., Reviznikov D. L.** Adaptive Interpolation Algorithm on Sparse Meshes for Numerical Integration of Systems of Ordinary Differential Equations with Interval Uncertainties, *Differential Equations*, 2021, vol. 57, no. 7, pp. 947–958. DOI: 10.1134/S0012266121070107.

13. **Makino K., Berz M.** Models and Their Applications, *Numerical Software Verification 2017: conference*, Heidelberg, Germany, July 22–23, 2017, Springer International Publishing AG 2017, pp. 3–13.

14. **Nataraj P. S. V., Sundur S.** The Extrapolated Taylor Model, *Reliable Computing*, July, 2011, pp. 251–278.

15. **Rogalev A. N.** Guaranteed methods for solving systems of ordinary differential equations based on the transformation of symbolic formulas, *Computational technologies*, 2003, vol. 8, no. 5, pp. 102–116 (in Russian).

16. **Fu C., Ren X., Yang Y.-F., Lu K., Qin W.** Steady-state response analysis of cracked rotors with uncertain but bounded parameters using a polynomial surrogate method, *Commun. Nonlinear Sci. Numer. Simul.*, 2019, vol. 68, pp. 240–256. DOI: 10.1016/j.cnsns.2018.08.004.

17. **Fu C., Xu Y., Yang Y., Lu K., Gu F., Ball A.** Response analysis of an accelerating unbalanced rotating system with both

random and interval variables, *J. Sound Vib.*, 2020, vol. 466, article 115047. DOI: 10.1016/j.jsv.2019.115047.

18. **Gidasov V. Yu., Morozov A. Yu., Reviznikov D. L.** Adaptive Interpolation Algorithm Using TT-Decomposition for Modeling Dynamical Systems with Interval Parameters, *Computational Mathematics and Mathematical Physics*, 2021, vol. 61, no. 9, pp. 1387–1400. DOI: 10.1134/S0965542521090098.

19. **Smolyak S. A.** Quadrature and interpolation formulas for tensor products of certain classes of functions, *Dokl. Akad. Nauk SSSR*, 1963, vol. 148, no. 5, pp. 1042–1045 (in Russian).

20. **Bungartz H.-J., Griebel M.** Sparse grids, *Acta Numerica*, 2004, vol. 13, no. 1, pp. 147–269.

21. **Gerstner T., Griebel M.** Sparse grids, *Encyclopedia of Quantitative Finance*, 2008, vol. 13, p. 5.

22. **Oseledets I. V.** Tensor-train decomposition, *SIAM Journal on Scientific Computing*, 2011, vol. 33, no. 5, pp. 2295–2317. DOI: 10.1137/090752286.

23. **Oseledets I., Tyrtshnikov E.** TT-cross approximation for multidimensional arrays, *Linear Algebra and its Applications*, 2010, vol. 432, iss. 1, pp. 70–88. DOI: 10.1016/j.laa.2009.07.024.

24. **Morozov A. Yu., Reviznikov D. L.** Modelling of Dynamic Systems with Interval Parameters on Graphic Processors, *Programmnaya Ingeneria*, 2019, vol. 10, no. 2, pp. 69–76. DOI: 10.17587/prin.10.69-76.

25. **Morozov A. Yu.** Parallel Adaptive Interpolation Algorithm based on Sparse Grids for Modeling Dynamic Systems with Interval Parameters, *Programmnaya Ingeneria*, 2021, vol. 12, no. 8, pp. 395–403. DOI: 10.17587/prin.12.395-403.

26. **Kapralov N. S., Morozov A. Yu., Nikulin S. P.** Parallel Approximation of Multivariate Tensors using GPUs, *Programmnaya Ingeneria*, 2022, vol. 13, no. 2, pp. 94–101. DOI: 10.17587/prin.13.94-101.

27. **CUDA Zone**, available at: URL: <https://developer.nvidia.com/cuda-zone>

28. **OpenMP**, available at: URL: <https://www.openmp.org/>

29. **OpenMP technology**, available at: <https://pvs-studio.com/ru/a/0057/>

30. **Kutta M.** Beitrag zur näherungsweise Integration totaler Differentialgleichungen. *Zeitschrift für Mathematik und Physik*, 1901, vol. 46, pp. 435–453.

31. **Gill P. E., Murray W., Wright M. H.** Practical Optimization, ACADEMIC PRESS, INC. San Diego, 1997, 509 p.

32. **Kovacic I., Brennan M. J.** *The Duffing Equation: Nonlinear Oscillators and their Behaviour*, John Wiley & Sons, Hoboken, 2011, 369 p. DOI: 10.1002/9780470977859.

ИНФОРМАЦИЯ

XVI Международная конференция инженерии программного обеспечения SECR/2023

17–18 марта 2023 г., Москва, Отель «Holiday Inn Сокольники»

SECR — одно из старейших и авторитетных ИТ-событий, посвященных индустрии разработки ПО.

Уникальность конференции — это широкий охват тем. SECR представляет разрез всей ИТ-индустрии и преподносит его участникам в виде 100+ различных докладов, мастер-классов и дискуссий за два дня. Современные подходы, тренды, прогнозы, экспертные мнения — все это SECR.

Программа SECR

- Выступления ключевых спикеров — приглашенных экспертов ИТ-рынка
- Доклады в несколько потоков, отобранные на конкурсной основе
- Дискуссии и круглые столы
- Полноценный трек мастер-классов
- Специальная SECR Party в конце первого дня.

Подробности: <https://secrus.ru>