НИТУ «МИСиС»
Кафедра Инженерной кибернетики

Параллельные вычисления

Лабораторная работа №4. «Технология распараллеливания OpenMP. Алгоритмы параллельного интегрирования обыкновенных дифференциальных уравнений»

Сенченко Р.В.

8 мая 2020 г.

I. Базовая часть. Пусть задана система n нелинейных обыкновенных дифференциальных уравнений (ОДУ) следующего вида:

$$\frac{\mathrm{d}u_j(t)}{\mathrm{d}t} = f_j(t, u_1(t), u_2(t), \ldots, u_n(t)), \qquad u_j(0) = \alpha_j, \qquad j = 1, 2, \ldots, n$$

где $u_j(t) \in \mathbf{C}^1(\mathbb{R})$ — искомые функции; $f_j(t, U(t))$ — заданные функции (*правые части*) времени t и вектор-функции $U(t) = (u_1(t), u_2(t), \ldots, u_n(t))$; величины $\alpha_j \in \mathbb{R}$ — заданные начальные условия, всюду $j = 1, 2, \ldots, n$.

Требуется реализовать метод Пикара для решения системы ОДУ в параллельном режиме; при этом для вычисления возникающих в ходе решения определённых интегралов использовать квадратурные формулы. Подробное описание метода см. в книге *Старченко А.В., Берцун В.Н. “Методы параллельных вычислений”. Учебник. 2013г.*.

Продемонстрировать корректность работы реализованного алгоритм, сравнив точное решение какйо-либо систему ОДУ с соответствующим численным решением. Минимальная размерность решаемой системы ОДУ равна $n = 5$, а целевой коэффициент ускорения должен удовлетворять оценке $s \geqslant 1.5$.

Подсказка. *Один из возможных путей быстрого получения тестовых систем ОДУ с известным точным решением состоит в следующем. Необходимо выбрать n функций $u_j^{\mathrm{sol.}}(t)$, а также выбрать “заготовки” правых частей $h_j(t, U)$. Далее, подставляем выбранные функции $u_j^{\mathrm{sol.}}(t)$ в правые части $h_j(t, U)$ и полагаем, что $f_j(t, U) = h_j(t, U)\big|_{U=U^{\mathrm{sol.}}}$. Очевидно, что полученная таким образом система ОДУ с правыми частями $f_j(t, U)$ и начальными условиями $\alpha_j = u_j^{\mathrm{sol.}}(0)$ имеет своим решением функции $u_j^{\mathrm{sol.}}(t)$, $j = 1, 2, \ldots, n$.*

Реализованный алгоритм должен быть хорошо документирован и снабжён исчерпывающими комментариями. Программный язык разработки: c++.

II. Дополнительная часть.

1. **Метод Рунге-Кутты 4-ого порядка точности, 3$\star$.** Реализовать параллельный явный метод Рунге-Кутты 4-ого порядка точности для решения систему ОДУ. Сравнить эффективность реализованного метода с эффективностью метода Пикара из базовой части лабораторной работы.

2. **Методы Адамса со схемой “предиктор–корректор”, 3$\star$.** Реализовать алгоритм решения системы ОДУ семейста методов Адамса с применением схемы “предиктор–корректор”. Провести оценку ускорения параллельной формы алгоритма.

3. **Задача Дирихле для уравнения Пуассона, 4$\star$.** Пусть задана двухмерная задача Дирихле для уравнения Пуассона:

$$\frac{\partial^2 u(x,y)}{\partial y^2} + \frac{\partial^2 u(x,y)}{\partial y^2} = f(x,y)$$

   в прямоугольной области $\Omega = [0,X] \times [0,Y]$ с заданными граничными условиями $u(0,y) = \alpha_0(y)$, $u(X,y) = \alpha_X(y)$ и $u(x,0) = \beta_0(x)$, $u(x,Y) = \beta_Y(x)$.

   Реализовать алгоритм паралелльного решения поставленной задачи, для чего применить метод конечных разностей, провести необходимые преобразования и свести задачу к решению СЛАУ. (Разумеется, решение полученного СЛАУ следует провести с помощью любого соответствующего метода в параллельном режиме.) Также необходимо провести контроль точности полученного решения.