

**О. Н. Крахмалев**, канд. техн. наук, доц., olegkr64@mail.ru,  
Брянский государственный технический университет

## Алгоритмизация параллельных вычислений в динамической модели манипуляционных систем роботов

*Предложен способ алгоритмизации параллельных вычислений в динамической модели манипуляционных систем роботов, полученной методом Лагранжа—Эйлера. Алгоритм строится на основе объектно-ориентированного подхода методом декомпозиции. При этом в структуре динамической модели выделяются основообразующие составные части. В качестве таких частей динамической модели выделены геометрическая и инерционная модели манипуляционных систем. Использование языка объектно-ориентированного программирования позволяет описать классы, соответствующие структурам геометрической и инерционной моделей. На основе созданных классов могут быть инициализированы объекты. Из объектов путем соединения их в различных комбинациях с указанием допустимых операций над ними могут быть получены объектные схемы. Исходная динамическая модель рассматривается как инструкция для сборки соответствующей ей объектной схемы. Алгоритмизация параллельных вычислений основывается на независимости вычисления объектов данных классов.*

**Ключевые слова:** роботы, манипуляционные системы, динамические модели, математическое моделирование, объектно-ориентированный подход, вычислительные алгоритмы, параллельные вычисления

### Введение

При моделировании манипуляционных систем роботов разработка алгоритмов, реализующих параллельные вычисления соответствующих математических моделей, представляет собой актуальную задачу. Известны примеры разработок таких алгоритмов [1—5], однако исследование данного вопроса нельзя считать завершенным.

Рассматриваемые в статье методы объектно-ориентированного моделирования манипуляционных систем роботов позволяют повысить эффективность выполнения работ, связанных с составлением параллельных вычислительных алгоритмов. Это достигается на основе представления математических моделей в виде объектных схем, в которых каждый объект рассматривается как часть математической модели, независимая от других частей-объектов, связанных с ним в общую схему.

Динамическая модель манипуляционных систем роботов, полученная на основе метода Лагранжа — Эйлера, имеет вид

$$[M_s]\{\ddot{q}\} + \{\dot{q}\}^T[C_s]\{\dot{q}\} = Q_s, s = 1, \dots, n, \quad (1)$$

здесь  $\{q\} = [q_1, q_2, \dots, q_n]^T$ ,  $\{\dot{q}\} = [\dot{q}_1, \dot{q}_2, \dots, \dot{q}_n]^T$ ,  $\{\ddot{q}\} = [\ddot{q}_1, \ddot{q}_2, \dots, \ddot{q}_n]^T$  — векторы обобщенных ко-

ординат и их производных по времени;  $Q_s$  — обобщенная сила по  $s$ -й обобщенной координате;  $n$  — число степеней свободы манипуляционной системы;  $[M_s]$  —  $(1 \times n)$ -мерная матрица-строка инерционных характеристик

$$[M_s(q)] = \sum_{k=1}^{(1 \times n)} [m_j^{sk}],$$

$$m_j^{sk} = \text{tr} \left( \frac{\partial A_{0,k}}{\partial q_s} H_k \frac{\partial A_{0,k}^T}{\partial q_j} \right), j = 1, \dots, n; \quad (2)$$

$[C_s]$  —  $(n \times n)$ -мерная матрица центробежных и кориолисовых эффектов

$$[C_s(q)] = \sum_{k=1}^{(n \times n)} [c_{ij}^{sk}],$$

$$c_{ij}^{sk} = \text{tr} \left( \frac{\partial A_{0,k}}{\partial q_s} H_k \frac{\partial A_{0,k}^T}{\partial q_i \partial q_j} \right), i, j = 1, \dots, n, \quad (3)$$

где  $\text{tr}$  — операция вычисления суммы диагональных элементов матрицы,  $H_k$  —  $(4 \times 4)$ -мерная матрица инерции  $k$ -го звена манипуляционной системы, рассматриваемого как твердое тело.  $A_{0,k}$  —  $(4 \times 4)$ -мерная матрица преобразования однородных координат из системы координат, связанной с  $k$ -м звеном, в не-

подвижную систему координат, связанную с основанием робота;

$$Q_s = Q_{Ds} + Q_{Gs} + Q_{Fs}, \quad (4)$$

где  $Q_{Ds}$  — усилие, развиваемое приводом в  $s$ -м сочленении (шарнире);  $Q_{Gs}$  — обобщенная сила по  $s$ -й обобщенной координате, соответствующая действию сил тяжести;  $Q_{Fs}$  — обобщенная сила по  $s$ -й обобщенной координате, соответствующая действию внешних сил:

$$Q_{Gs} = \sum_{k=s}^n \text{tr} \left[ F_{Gk} \frac{\partial A^T}{\partial q_s} \right], \quad (5)$$

где  $F_{Gk}$  —  $(4 \times 4)$ -мерная матрица, содержащая силы тяжести  $k$ -го звена [6],

$$Q_{Fs} = \sum_{k=s}^n \text{tr} \left( F_{Rk} \frac{\partial A_{0,k}^T}{\partial q_s} + F_{Mk,s} A_{0,s}^T \right), \quad (6)$$

где  $F_{Rk}$  и  $F_{Mk,s}$  —  $(4 \times 4)$ -мерные матрицы, соответственно содержащие проекции главного вектора и главного момента внешних сил [6].

Объектная схема данной динамической модели может быть представлена в виде, показанном на рис. 1 [7]. Объекты, соответствующие матрицам и векторам, изображены на этой схеме прямоугольниками, скалярные величины — треугольниками, а арифметические операции — соответствующими им знаками.

На представленной объектной схеме  $Q_s$  — объект, соответствующий результирующей обобщенной силе по  $s$ -й обобщенной (шарнирной) координате;  $Q_{Ds}$  — объект, соответствующий усилию, развиваемому приводом в  $s$ -м сочленении (шарнире);  $Q_{Gs}$  — объект, соответствующий обобщенной силе  $Q_{Gs}$ ;  $Q_{Fs}$  — объект, соответствующий обобщенной силе  $Q_{Fs}$ .  $M_s$  — объект класса *Dynamic\_matrix\_Ms*, соответствующий матрице-строке  $[M_s]$ ;  $C_s$  — объект класса *Dynamic\_matrix\_Cs*, соответствующий матрице  $[C_s]$ .  $DDq$  — объект класса *Generalized\_accelerations*,  $Dq$  и  $Dq^T$  — объекты класса *Generalize\_speeds*, моделирующие векторы  $\{\ddot{q}\}$ ,  $\{\dot{q}\}$  и  $\{\dot{q}\}^T$  соответственно.

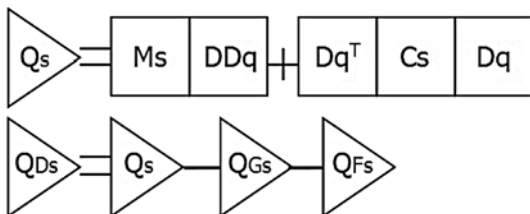


Рис. 1. Объектная схема динамической модели манипуляционных систем роботов

Структуры введенных классов могут быть созданы с использованием языков объектно-ориентированного программирования, например C++, и должны описывать объекты в соответствии с выражениями (1)–(6).

## 1. Алгоритмизация вычислений динамической модели

В вычислительном алгоритме, реализующем моделирование на основе объектной схемы (рис. 1), можно выделить несколько уровней (ярусов), на которых происходят вычисления составных частей данной объектной схемы.

На первом (I) уровне вычисляются объекты, соответствующие матрицам инерции звеньев  $H_k$ , составляющие так называемую инерционную модель манипуляционных систем (рис. 2).

На втором (II) уровне вычисляются объекты, соответствующие матрицам преобразования однородных координат  $A_{0,k}$  и их частным производным, относящиеся к геометрической модели манипуляционных систем (рис. 3).

Объекты, вычисляемые на втором (II) уровне, зависят от изменяющихся в процессе моделирования параметров, кроме того, при вычислении матриц частных производных необходимо исключить вычисление элементов этих матриц, тождественно равных нулю. В соответствии с этим число  $N_1$  вычисляемых матриц частных производных  $\partial A_{0,k} / \partial q_s$  ( $k, s = 1, \dots, n$ ) будет определяться выражением

$$N_1 = \frac{n}{2}(n+1), \quad (7)$$

а число  $N_2$  матриц вторых производных ( $\partial^2 A_{0,k} / \partial q_s^2$  и  $\partial^2 A_{0,k} / (\partial q_i \partial q_j)$ ) — выражением

$$N_2 = \sum_{k=1}^n \frac{k}{2}(k+1). \quad (8)$$



Инерционная модель

Рис. 2. Уровень I алгоритма



Геометрическая модель

Рис. 3. Уровень II алгоритма

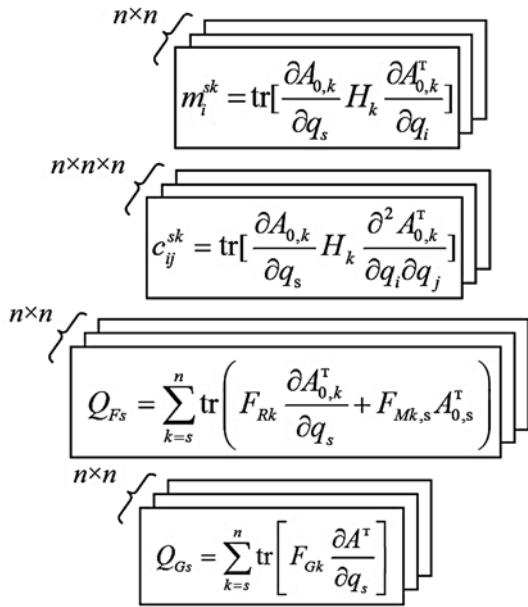


Рис. 4. Уровень III алгоритма

На третьем (III) уровне выполняются вычисления на основе объектных схем, соответствующих различным элементам матриц  $[M_s]$  и  $[C_s]$ , выполняемые на основе выражений (2) и (3), и соответствующих скалярным величинам  $Q_{Gs}$  и  $Q_{Fs}$ , выполняемые на основе выражений (5) и (6) (рис. 4).

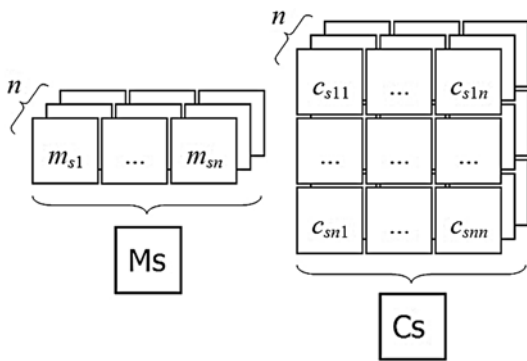


Рис. 5. Объектное представление матриц  $[M_s]$  и  $[C_s]$

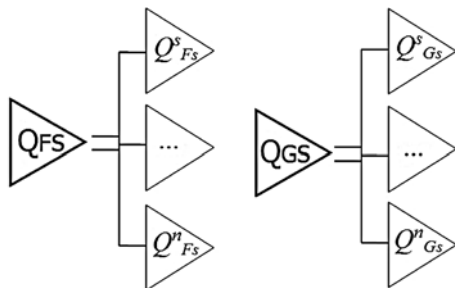


Рис. 6. Объектное представление обобщенных сил

На четвертом (IV) уровне на основе объектов  $M_s$ ,  $C_s$ ,  $QGs$  и  $QFs$ , соответствующих матрицам  $[M_s]$  и  $[C_s]$  (рис. 5) и обобщенным силам  $Q_{Gs}$  и  $Q_{Fs}$  (рис. 6), выполняется вычисление объектной схемы динамической модели (1) (см. рис. 1).

Иерархическая структура алгоритма параллельных вычислений в динамической модели (1), разбитого на несколько уровней и соответствующего объектной схеме (см. рис. 1), без моделирования внешней нагрузки может быть проиллюстрирована на общей вычислительной схеме (рис. 7).

Вычисляемая на первом уровне алгоритма инерционная модель отражает распределение масс внутри звеньев исследуемой манипуляционной системы, соответствующая ей математическая модель представляет собой множество матриц инерции звеньев  $H_k$  ( $k = 1, \dots, n$ ).

На втором уровне формируется геометрическая модель манипуляционной системы. Ее математическая модель представляет собой множество матриц преобразования однородных координат  $A_{0,k}$  ( $k = 1, \dots, n$ ), дополненное множеством частных производных от этих матриц.

Определение этих двух подмоделей в структуре динамической модели манипуляционных систем роботов имеет важное методологическое значение, так как позволяет в структуре

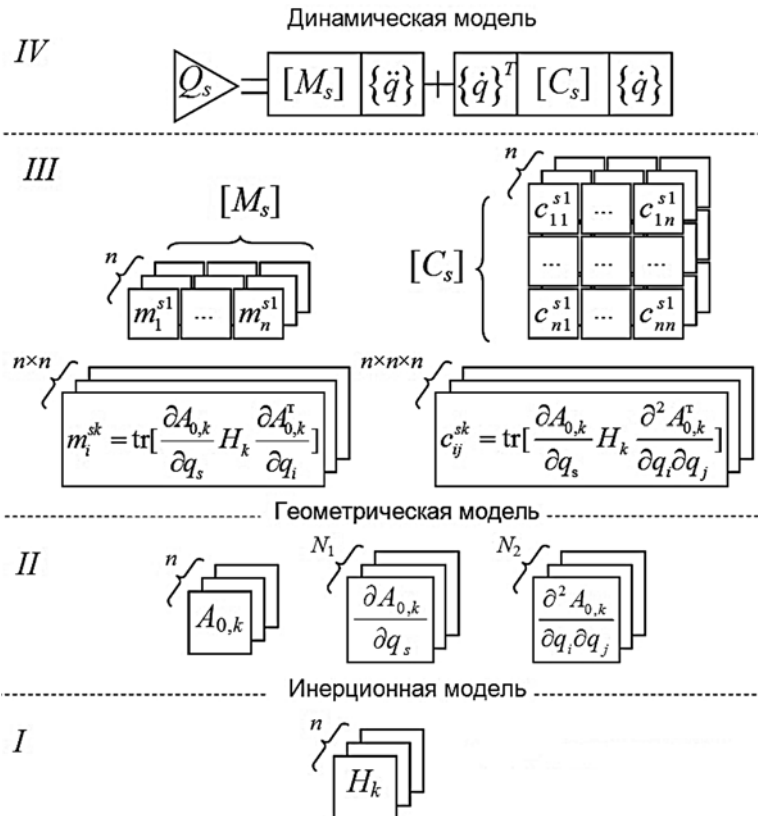


Рис. 7. Уровни алгоритма параллельных вычислений динамической модели

данных и алгоритмов динамической модели выделить части, имеющие определенный прикладной смысл, вычисление которых может выполняться независимо друг от друга.

Дальнейшее распараллеливание вычислений, выполняемых на соответствующих уровнях рассмотренного алгоритма параллельных вычислений в динамической модели манипуляционных систем роботов, может быть выполнено на основе алгоритмов параллельных вычислений матричных операций.

## 2. Алгоритмы параллельных вычислений матричных операций

Рассмотрим граф вычислительного процесса произведения матрицы размерности  $(4 \times 4)$  на вектор размерности  $(4 \times 1)$  (рис. 8), основанный на соотношении

$$w_i = \sum_{j=1}^4 a_{ij}v_j, \quad i = \{1, \dots, 4\}. \quad (9)$$

При вычислении произведения матрицы размерности  $n \times n$  на соответствующий вектор число необходимых скалярных операций сложения и умножения может быть оценено величиной

$$T_1 = 2n^2. \quad (10)$$

Анализ графа вычислительного процесса (рис. 8) показывает, что параллельные способы решения задачи произведения матрицы на вектор могут быть получены на основе использования параллельных алгоритмов суммирования, в данном случае использован каскадный алгоритм суммирования. Кроме того, способ организации параллельных вычислений будет зависеть от числа доступных для использования процессоров.

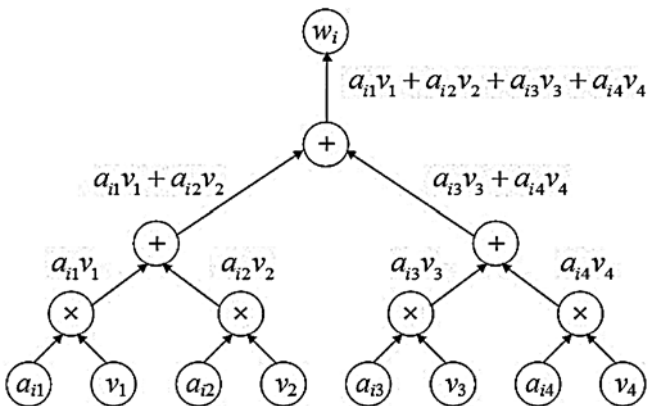


Рис. 8. Граф вычислительного процесса произведения  $(4 \times 4)$ -мерной матрицы на  $(4 \times 1)$ -мерный вектор

Нужно отметить, что последовательности выполняемых операций умножения отдельных  $n$  строк матрицы на вектор являются независимыми и могут быть выполнены параллельно. Операции умножения каждой строки на вектор включают  $n$  независимых операций поэлементного умножения и тоже могут быть выполнены параллельно. Таким образом, максимально необходимое число процессоров определяется величиной

$$p = n^2. \quad (11)$$

Время выполнения параллельного алгоритма, отнесенное ко времени выполнения одной скалярной операции, при использовании  $p$  процессоров определяется временем выполнения операции умножения и временем выполнения каскадного алгоритма суммирования [8]

$$T_p = 1 + \log_2 n. \quad (12)$$

Показатели эффективности параллельного алгоритма определяется выражением

$$S_p = \frac{T_1}{T_p} = \frac{2n^2}{1 + \log_2 n}, \quad (13)$$

где  $S_p$  — ускорение для  $p$  процессоров, получаемое при использовании параллельного алгоритма, и

$$E_p = \frac{S_p}{p} = \frac{2}{p(1 + \log_2 n)}, \quad (14)$$

где  $E_p$  — эффективность использования процессоров.

Информационный граф (рис. 8), соответствующий вычислительному процессу умножения  $i$ -й строки  $(4 \times 4)$ -мерной матрицы на  $(4 \times 1)$ -мерный вектор, построенный на основе четырех процессоров, представлен на рис. 9.

Задача умножения прямоугольных матриц определяется соотношением

$$c_{ij} = \sum_{l=1}^k a_{il}b_{lj}, \quad i = \{1, \dots, n\}, j = \{1, \dots, m\}. \quad (15)$$

В случае умножения квадратных матриц размерности  $n \times n$  число скалярных операций умножения и сложения пропорционально  $n^3$ , а для их оценки может быть использована величина

$$T_1 = 2n^3. \quad (16)$$

Последовательности выполняемых операций умножения отдельных строк первой матрицы на соответствующий столбец вто-

рой матрицы являются независимыми и могут быть выполнены параллельно. Алгоритм параллельного вычисления при умножении матриц может быть построен на основе рассмотренного вычислительного алгоритма умножения матрицы на вектор. Соответственно, необходимое число процессоров для максимального распараллеливания вычислительного алгоритма будет определяться величиной

$$p = n^3. \quad (17)$$

Информационный граф, соответствующий вычислительному процессу умножения  $(4 \times 4)$ -мерных матриц, построенный на основе 64 процессоров, представлен на рис. 10.

Время выполнения параллельного алгоритма, как и прежде, будет определяться выраже-

нием (12). Получаемое ускорение для  $p$  процессоров повысится:

$$S_p = \frac{T_1}{T_p} = \frac{2n^3}{1 + \log_2 n}, \quad (18)$$

а эффективность использования процессоров в соответствии с (14) не изменится.

Снижение числа используемых процессоров потребует перестройки информационного графа (рис. 10), отражающего вычислительный процесс умножения матриц. Это повлечет увеличение времени выполнения вычислений.

### Заключение

Рассмотренный в статье алгоритм параллельных вычислений динамической модели (1) имеет четыре уровня распараллеливания (см. рис. 7) и может быть дополнен начальным (нулевым) уровнем, соответствующим инициализации данных.

Число независимо вычисляемых операций на уровне III максимально. На этом уровне при уменьшении числа используемых процессоров будет наблюдаться наибольшее замедление.

На уровне IV информационный граф вычислительного процесса умножения  $(n \times n)$ -мерной матрицы на  $(n \times 1)$ -мерный вектор для динамической модели (1) должен строиться на основе  $n$  процессоров, по аналогии с информационным графом, построенным на основе четырех процессоров (см. рис. 9).

Рассмотренный алгоритм параллельных вычислений в динамической модели (1) манипуляционных систем роботов может быть реализован на основе объектно-ориентированного подхода с использованием объектных схем. Такие объектные схемы строятся из объектов, соответствующих геометрической и инерционной моделям исследуемой манипуляционной системы, а также из объектов, соответствующих модели внешней нагрузки.

Применение параллельных вычислительных алгоритмов в задачах моделирования требует от вычислительных систем наличия достаточного числа процессоров. Это требование может быть реализовано на основе многокристальных реконфигурируемых вычислительных систем, созданных на основе применения программируемых логических интегральных схем (ПЛИС). Такие вычислительные системы могут адаптироваться под структуру решаемой задачи и обеспечить реализацию сложных многоуровневых параллельных алгоритмов [9, 10].

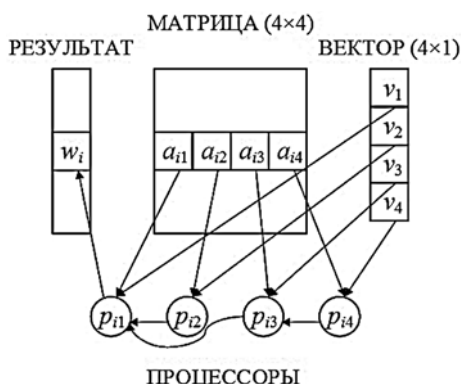


Рис. 9. Информационный граф вычислительного процесса произведения  $i$ -й строки  $(4 \times 4)$ -мерной матрицы на  $(4 \times 1)$ -мерный вектор

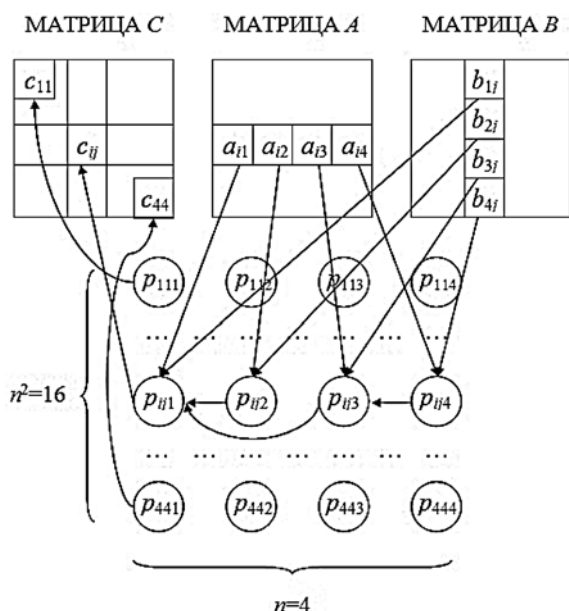


Рис. 10. Информационный граф вычислительного процесса умножения  $(4 \times 4)$ -мерных матриц на основе 64 процессоров

## Список литературы

1. Белоусов И. Р. Применение метода символьных преобразований для формирования алгоритма параллельных вычислений в задачах кинематики и динамики роботов // Отчет ИПМ им. М. В. Келдыша РАН № 5-19-93. 1993. 25 с.
2. Второв В. Б., Иншаков Д. Ю. Моделирование динамики робота-манипулятора на параллельных вычислительных структурах // Известия ТЭТУ. 1997. № 513. С. 65–70.
3. Малашонок Г. И., Аветисян А. И., Валеев Ю. Д., Зуев М. С. Параллельные алгоритмы компьютерной алгебры // Труды ИСП РАН. 2004. № 2. Т. 8. С. 169–180.
4. Внуков А. А., Лисенков М. А. Параллельный алгоритм обработки данных на примере решения основных задач динамики роботов-манипуляторов // Доклады пятой международной конференции "Параллельные вычисления и задачи управления", Москва. 2010. С. 1025–1036.
5. Внуков А. А., Лисенков М. А. Разработка алгоритмов параллельной обработки данных в многопроцессорных системах на примере решения задач динамики промышленных роботов // Вестник РУДН серия Инженерные исследования. 2010. № 4. С. 60–72.
6. Крахмалев О. Н. Моделирование манипуляционных систем роботов: Учеб. пособ. Саратов: Ай Пи Эр Медиа, 2018. 165 с.
7. Крахмалев О. Н. Объектно-ориентированное моделирование манипуляционных систем роботов // Робототехника и техническая кибернетика. 2018. № 4(21). С. 41–47.
8. Гергель В. П., Стронгин Р. Г. Основы параллельных вычислений для многопроцессорных вычислительных систем: учебное пособие. Н. Новгород: ННГУ, 2003. 184 с.
9. Каляев И. А., Дордопуло А. И., Левин И. И., Федоров А. М. Развитие отечественных многокристальных реконфигурируемых вычислительных систем: от воздушного к жидкостному охлаждению // Труды СПИИРАН. 2017. № 1 (50). С. 5–31.
10. Kalyaev I. A., Levin I. I., Semernikov E. A., Shmoilov V. I. Reconfigurable multipipeline computing structures // New York: Nova Science Publishers. 2012. 330 p.

O. N. Krakhmalev, Ph.D., Associate Professor, olegkr64@mail.ru,  
Bryansk State Technical University, Bryansk, 241036, Russian Federation

## Algorithmization of Parallel Computing in a Dynamic Model of Robot Manipulation Systems

*The task of developing an algorithm for parallel computing in a dynamic model of robot manipulation systems obtained by the Lagrange-Euler method is considered.*

*The algorithm is built on the basis of the object-oriented approach by the decomposition method. In the structure of the dynamic model highlights the fundamental components. Geometric and inertial models of manipulation systems are defined as such components. Using the language of object-oriented programming, classes are created that describe the structure of the geometric and inertial models. Independent objects can be initialized based on the created classes. By combining objects in various combinations, object schemas can be obtained. The initial dynamic model is considered as an instruction for building the corresponding object schema. Algorithmization of parallel computing is based on the independence of the calculation of data objects of classes. The use of parallel computational algorithms in modeling problems requires a sufficient number of processors to implement such calculations. This requirement can be implemented on the basis of multi-chip reconfigurable computer systems created using programmable logic integrated circuits. Such computing systems can adapt to the structure of the problem to be solved and provide the implementation of complex multi-level parallel algorithms.*

**Keywords:** robots, manipulation systems, dynamic models, mathematical modeling, object-oriented approach, computational algorithms, parallel computing.

DOI: 10.17587/it.26.354-359

### References

1. Belousov I. R. The use of the method of symbolic transformations for the formation of parallel computing algorithms in the problems of kinematics and dynamics of robots, *Report of IPM im. M. V. Keldysh RAS*, no. 5-19-93, 1993, 25 p. (in Russian).
2. Vtorov V.B, Inshakov D. Yu. Modeling the dynamics of a robotic arm on parallel computing structures, *Izvestiya TETU*, 1997, no. 513, pp. 65–70 (in Russian).
3. Malashonok G. I., Avetisyan A. I., Valeev Yu. D., Zuev M. S. Parallel Algorithms of Computer Algebra, *Trudy ISP RAN* (Proceedings of ISP RAS), 2004, no. 2, vol. 8, pp. 169–180.
4. Vnukov A. A., Lisenkov M. A. Parallel data processing algorithm by the example of solving the main problems of the dynamics of robotic manipulators, *Reports of the fifth international conference "Parallel Computing and Control Problems"*, Moscow, 2010, pp. 1025–1036 (in Russian).
5. Vnukov A. A., Lisenkov M. A. The development of parallel data processing algorithms in multiprocessor systems using the example of solving the problems of the dynamics of industrial robots, *Bulletin of RUDN University series Engineering Research*, 2010, no. 4, pp. 60–72.
6. Krakhmalev O. N. Modeling of robot manipulation systems: textbook, Saratov, IPR Media, 2018, 165 p. (in Russian).
7. Krakhmalev O. N. Object-oriented modeling of robots' manipulation systems, *Robotics and Technical Cybernetics*, 2018, no. 4 (21), pp. 41–47 (in Russian).
8. Gergel V. P., Strongin R. G. Basics of parallel computing for multiprocessor computing systems: a tutorial, N. Novgorod: NNSU, 2003, 184 p. (in Russian).
9. Kalyaev I. A., Dordopulo A. I., Levin I. I., Fedorov A. M. Development of domestic multi-chip reconfigurable computer systems: from air to liquid cooling, *Proceedings of SPIIRAS*, 2017, no. 1 (50), pp. 5–31 (in Russian).
10. Kalyaev I. A., Levin I. I., Semernikov E. A., Shmoilov V. I. Reconfigurable multipipeline computing structures, New York, Nova Science Publishers, 2012, 330 p.