



ПАРАЛЛЕЛЬНЫЕ ВЫЧИСЛЕНИЯ В ЗАДАЧАХ УПРАВЛЕНИЯ¹

Ф.И. Бернацкий, Г.Б. Диго, Н.Б. Диго

Институт автоматики и процессов управления, г. Владивосток

Обсуждены вопросы управления сложными техническими системами в условиях неопределенности на основе многометодного и многовариантного анализа. Рассмотрена возможность применения многопроцессорных компьютеров для распараллеливания вычислений.

ВВЕДЕНИЕ

Современный этап развития теории управления характеризуется не только разработкой оптимальных систем, но и оптимизацией применяемых алгоритмов. Попадающим в сферу фундаментальных исследований и практических приложений сложным техническим системам обычно свойственны многомерность, нелинейность, неопределенность, существенно затрудняющие исследование их поведения и решение задачи управления. Из-за этого приходится сталкиваться с двумя важными и взаимосвязанными проблемами: поиском наилучшего решения и преодолением вычислительной трудоемкости методов и алгоритмов нахождения таких решений. Достигнутый уровень развития средств вычислительной техники и информационных технологий позволяет отказаться в теории управления от приемов и методов, лишь частично преодолевающих вычислительные трудности, и воспользоваться подходом, основанным на параллельных и распределенных вычислениях.

Качество управления реальными техническими системами существенно зависит от удачного подбора их математического описания и обоснованного задания требований к результату функционирования. В зависимости от вида неопределенности для ее преодоления приходится применять разные методы, но не всегда очевидно, какой из них предпочтительнее. Последовательный перебор этих

методов требует больших временных затрат, недопустимых при оперативном управлении. Параллельные вычисления на многопроцессорных компьютерах, обеспечивая одновременное проведение расчетов несколькими методами, устраняют такую ситуацию. В работе рассматривается возможность распараллеливания вычислений при применении многометодного и многовариантного анализа в задачах управления.

ПОСТАНОВКА ЗАДАЧИ И ОПИСАНИЕ ПОДХОДА

Рассматриваются стохастические технические системы, функционирующие в условиях неопределенности. В зависимости от вида неопределенности и способов ее преодоления применяются различные методы построения математических моделей. Для таких систем требования к результату их функционирования приходится задавать в виде допусков на выходные показатели и применять вероятностно-допусковые критерии.

Предложенные в работах [1, 2] методы робастного управления, позволяющие преодолевать два вида неопределенности (априори неизвестные условные законы распределения выходных переменных и невозможность задания требований к результату функционирования рассматриваемой системы в виде конкретных значений выходных показателей), заключаются в следующем.

Пусть задана структура математической модели зависимости выходной переменной y

$$y = y(X, U) \quad (1)$$

исследуемого объекта от векторов входных контролируемых $X = (x_1, \dots, x_n)^T$ и управляющих $U = (u_1, \dots, u_m)^T$ переменных и установлены ограничения $y \in [A, B]$.

¹ Работа выполнена при финансовой поддержке Дальневосточного отделения РАН (грант конкурса проектов ДВО РАН 2003г., проект «Разработка методов и алгоритмов параметрического синтеза в стохастических системах», включенный в программу фундаментальных научных исследований Отделения ЭМПУ РАН, №14).

Предположим, что условная плотность распределения вероятностей $p(y|U)$ принадлежит некоторому классу K унимодальных распределений $p_1(y|U), \dots, p_r(y|U)$.

Из-за наличия помех вычисленное по выражению (1) значение выхода является его условным математическим ожиданием, а сама модель – уравнением регрессии. Используя это выражение, в пространстве управлений U при $X = X_j$ для каждого j -го распределения класса K строится область $S_U^{(j)}$ допустимых управлений, в которой для любого вектора управлений $U \in S_U^{(j)}$ выполняется вероятностное неравенство

$$P\{y \in [A, B]\} \geq P_0, \quad (2)$$

где P_0 – заданное значение вероятности. Их пересечение $F = \bigcap_{j=1}^r S_U^{(j)}$ образует область робастных управлений, в которой целевое условие (2) мало чувствительно к конкретному виду условного закона распределения. Кроме вероятностно-допускового критерия (2) может использоваться допусковый критерий [1]. Требования к выходной переменной при этом задаются в виде допуска $[A, B]$, а задача управления состоит в нахождении в пространстве управлений области S_U , обеспечивающей выполнение целевого условия.

Если требования к результату функционирования объекта заданы в виде допуска на значения выходной переменной y , известны ее регрессионная зависимость от управляющих переменных при фиксированных входных переменных $y = y(X, U)$ и ограничения на управляющие переменные, то область S_U описывается системой неравенств

$$\left. \begin{aligned} A \leq y(u_1, \dots, u_n) \leq B \\ u_j^{\min} \leq u_j \leq u_j^{\max}, \quad j = 1, \dots, m \end{aligned} \right\}. \quad (3)$$

При векторном выходе $Y = (y_1, \dots, y_k)^T$ область робастных управлений исследуемой системы строится сначала для каждой его компоненты. Их пересечение, если оно не пусто, и является областью робастных управлений всей системы. Если же оно пусто, то приходится анализировать и при необходимости корректировать допуски $[A_i, B_i]$ на компоненты вектора выхода, уточнять для них классы условных законов распределения и проверять правильность задания принятых значений вероятностей. Но в связи с тем, что заранее неизвестно, будет ли в случае векторного выхода пересечение F непустым, кроме вероятностно-допускового критерия, дополнительно к критерию (2) по каждой выходной переменной применяется допусковый критерий [2]

$$A_i \leq y_i \leq B_i, \quad i = 1, \dots, k, \quad (4)$$

дополненный ограничениями на управляющие переменные. Это позволяет при $F = \emptyset$ попытаться скорректировать ограничения $A_i, B_i, i = 1, \dots, k$.

Изложенный подход требует уточнения параметров модели по текущим данным, выбора класса условных распределений выходных переменных, решения систем нелинейных неравенств, нахождения областей допустимых и робастных управлений. Поскольку это связано с большими временными и вычислительными затратами, возникает задача исследования возможности распараллеливания вычислений с помощью многометодного и многовариантного анализа, чтобы предложить алгоритмы распараллеливания и способы организации вычислений, построить их структурные схемы, выбрать средства программной реализации.

АНАЛИЗ ЗАДАЧИ И ВЫЯВЛЕНИЕ ЕЕ ПОТЕНЦИАЛЬНОГО ПАРАЛЛЕЛИЗМА

На этапе построения и уточнения параметров математической модели (1) в описанных выше условиях путем организации параллельных вычислительных потоков обеспечивается одновременное проведение расчетов несколькими методами с разными наборами данных. Каждый поток реализовывает один метод построения модели на отдельном процессоре, и по заданному критерию адекватности выбирается наилучшая модель [3, 4].

Параллельные вычисления позволяют заменить сложно формализуемую процедуру выбора класса условных распределений одновременным рассмотрением различных вариантов возможных из технологических соображений классов распределений с последующим выбором лучшего результата [3].

При построении областей допустимых и робастных управлений приходится исследовать и решать системы в общем случае нелинейных неравенств (3), общих методов решения которых не существует. Предложенный в работе [5] алгоритм обеспечивает переход к линейным системам, решаемым известными методами. Распараллеливание ускоряет вычисления при проверке нелинейных неравенств на выпуклость, замене каждого выпуклого неравенства несколькими линейными неравенствами, поиске опорных гиперплоскостей для невыпуклых неравенств и решении системы линейных неравенств на этапе нахождения координат вершин выпуклого многогранника, описывающего область робастных управлений.

При нахождении областей допустимых и робастных управлений для векторного выхода распараллеливание применяется при одновременном использовании допускового и вероятностно-допускового критериев. Кроме ускорения процесса вычислений, расширяется возможность получения непустой области робастных управлений.



СТРУКТУРНЫЕ СХЕМЫ АЛГОРИТМОВ РАСПАРалЛЕЛИВАНИЯ

Схема реализации алгоритма распараллеливания при выборе текущего управления в случае скалярного выхода приведена на рис. 1. Предполагается, что модель (1) уточняется по трем модификациям стандартного регрессионного анализа (метод текущих наименьших квадратов, взвешенный метод наименьших квадратов и метод наименьших квадратов с переменной глубиной памяти) с использованием, соответственно, разных длин текущих интервалов, весовых коэффициентов и глубин памяти [3]. Для построения областей допустимых управлений и последующего оперативного управления выбирается по заданному критерию наилучшая модель. Одновременное применение разных методов идентификации и расчет по ним нескольких вариантов моделей формируют группу алгоритмов, каждый из которых работает достаточно эффективно только в определенной ситуации. При этом обеспечивается автоматический анализ различных вариантов построения моделей разными методами по заданному критерию.

Схема распараллеливания вычислений при построении области робастных управлений для одного класса условных распределений выходной переменной приведена на рис. 2. Робастные области управления строятся для каждого класса, среди них определяется лучшая по заданному критерию, и в ней выбирается текущее управление.

При преобразовании системы нелинейных неравенств в систему линейных неравенств распараллеливание ускоряет проверку на выпуклость нелинейных неравенств, замену каждого выпуклого неравенства несколькими линейными и поиск опорных гиперплоскостей для невыпуклых неравенств [5]. Схема реализации алгоритма для k неравенств (k – число доступных процессоров) приведена на рис. 3, где $Ineq_i$ – проверка на выпуклость i -го нелинейного неравенства, Con – рассылка данных о выпуклых неравенствах на k_1 процессоров, $Ncon$ – рассылка данных о невыпуклых неравенствах на k_2 процессоров, Ap_i – аппроксимация i -го выпуклого неравенства i_p линейными неравенствами, Bas_i – замена i -го невыпуклого неравенства ограничивающей гиперплоскостью, Система – формирование системы линейных неравенств.

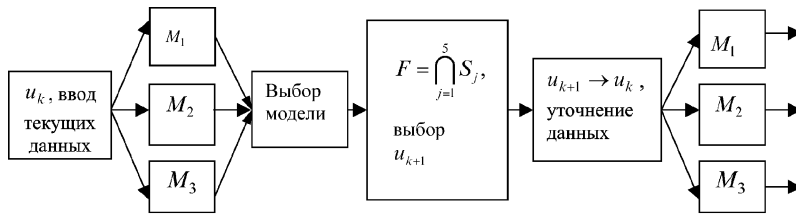
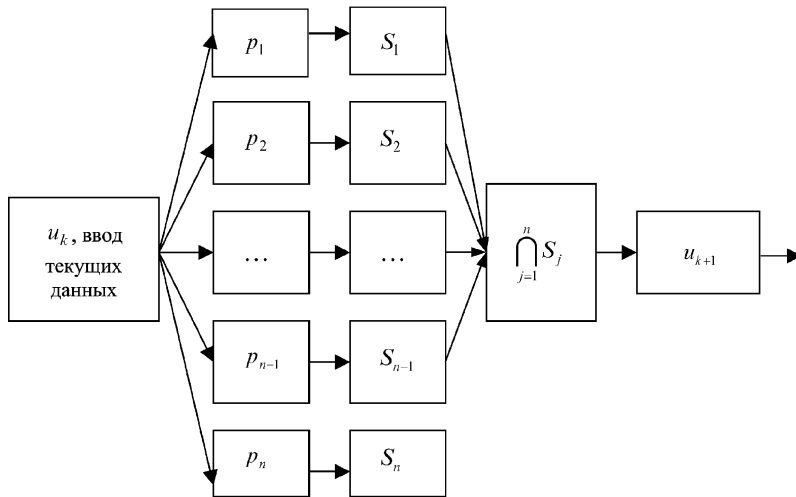
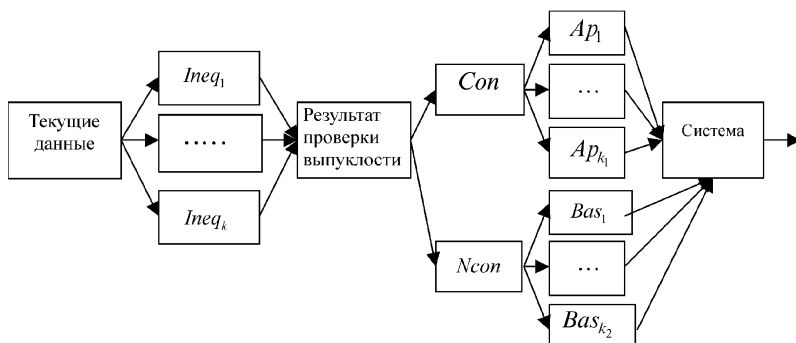
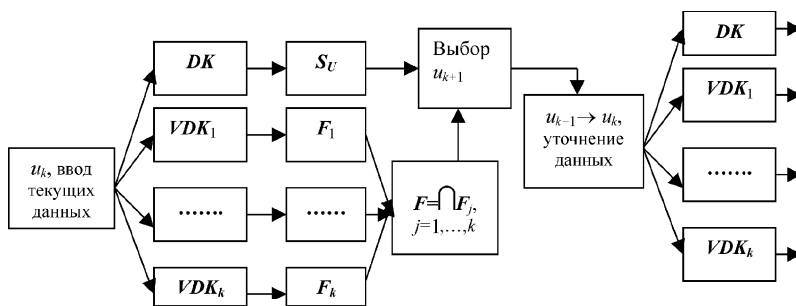
Схема распараллеливания вычислений при построении областей допустимых и робастных управлений для векторного выхода приведена на рис. 4. Распараллеливание применяется при одновременном использовании допускового и вероятностно-допускового критериев. В случае допускового критерия исходная система неравенств включает

все ограничения на выходные переменные из выражений (3) и (4), а для вероятностно-допускового критерия – только на конкретный выходной показатель. При такой схеме вычислений выбор текущего управления осуществляется из области $S_U \cap F$, если $F \neq \emptyset$ и $S_U \neq \emptyset$; из S_U , если $F = \emptyset$ и $S_U \neq \emptyset$; из F , если $F \neq \emptyset$ и $S_U = \emptyset$. На рис. 4 приняты обозначения: DK – формирование системы неравенств для построения области допустимых управлений по допусковому критерию; S_U – область допустимых управлений, полученная по допусковому критерию; VDK_i – формирование системы неравенств для построения области робастных управлений i -й выходной переменной по вероятностно-допусковому критерию; F_i – область робастных управлений i -й выходной переменной, полученная по вероятностно-допусковому критерию.

СРЕДСТВА ПРОГРАММНОЙ РЕАЛИЗАЦИИ И СХЕМА ВЫЧИСЛЕНИЙ

Программная реализация параллельных вычислений во многом определяется языковыми и инструментальными средствами программирования, архитектурой используемых компьютеров. Поскольку предлагаемые алгоритмы предполагают распараллеливание как по данным (одна параллельная инструкция воздействует на разные потоки данных), так и по процессам (различные потоки данных участвуют в вычислительном процессе под управлением различных потоков команд), применяются системы с массовым параллелизмом.

Для распараллеливания описанных алгоритмов используется MPMD-модель программирования (Multiple Program – Multiple Data) [6], обеспечивающая выполнение различных программ над разными потоками данных и учитывающая основные признаки, присущие параллельным программам (параллелизм, масштабируемость, локальность, модульность). Для уменьшения числа и объема сообщений, которыми обмениваются параллельно работающие процессоры, и сокращения общего времени решения задачи распараллеливание осуществляется на уровне крупных блоков. Структура обменов однородна в пределах одного блока и между блоками и не всегда однородна при передаче параметров отдельных методов. Как основное средство программирования применяется система передачи сообщений MPI (Message Passing Interface) [6], практически являющаяся стандартом для программирования алгоритмов с массовым параллелизмом. Это объясняется тем, что MPI обеспечивает единый механизм взаимодействия ветвей внутри параллельного приложения независимо от машинной архитектуры, взаимного расположения ветвей


Рис. 1. Схема выбора управления на текущем $(k + 1)$ -м шаге

Рис. 2. Схема выбора текущего управления на основе областей допустимых управлений

Рис. 3. Схема реализации алгоритма преобразования k нелинейных неравенств в линейные

Рис. 4. Схема выбора управления для векторного выхода на текущем $(k + 1)$ -м шаге

и интерфейса разработчика приложений операционной системы. Кроме того, программы, использующие MPI, легче отлаживаются (сужается простор для совершения стереотипных ошибок параллельного программирования) и быстрее переносятся на другие платформы (имеются компиляторы для Си, Си++ и Фортрана). Программная реализация алгоритма ориентирована на многопроцессорную систему МВС-1000/16 под управлением ОС LINUX. С учетом того, что максимально доступное количество процессоров в системе относительно невелико и равно 16, выбрана крупноблочная схема распараллеливания.

При реализации описанных алгоритмов распараллеливания вычислений работа на корневом процессоре начинается с ввода матрицы исходных данных и формирования классов условных распределений выходных переменных, для которых будут строиться области робастных управлений. Основные этапы вычислений на каждом временном шаге представлены следующей схемой:

- обновление матрицы данных на корневом процессоре;
- рассылка обновленных данных всем остальным процессорам;
- обновление и уточнение параметров конкретных методов на корневом процессоре для построения различных вариантов моделей;
- рассылка обновленных параметров всем остальным процессорам;
- построение разных вариантов модели по каждому методу на параллельных процессорах;
- сборка полученных результатов на корневом процессоре;
- выбор лучшего варианта модели по каждому методу на корневом процессоре;
- выбор наилучшей модели на корневом процессоре;
- рассылка полученных результатов остальным процессорам;
- построение областей допустимых управлений на отдельных процессорах;
- построение области робастных управлений на отдельных процессорах;
- сборка полученных результатов на корневом процессоре;



- выбор наилучшей области робастных управлений на корневом процессоре;
- выбор текущего управления на корневом процессоре;
- рассылка полученных результатов по остальным процессорам.

ЗАКЛЮЧЕНИЕ

В работе исследованы возможности применения параллельных вычислений для оперативного управления. На примере алгоритма робастного управления рассмотрено применение многометодного и многовариантного анализа. Изложены основные этапы распараллеливания: анализ задачи и выявление ее потенциального параллелизма, выбор схемы алгоритмов распараллеливания, схемы вычислений и средств программирования задачи. Выполненные исследования показывают, что предлагаемый алгоритм обладает значительным объемом потенциального параллелизма и хорошей, с точки зрения распараллеливания, структурой.

ЛИТЕРАТУРА

1. *Бернацкий Ф. И., Пащенко Ф. Ф.* Синтез робастных алгоритмов управления технологическими объектами // Автоматика и телемеханика. — 1997. — № 12. — С. 150–162.
2. *Бернацкий Ф. И., Диго Г. Б., Диго Н. Б.* Моделирование алгоритмов робастного управления технологическими процессами // Информатика — Машиностроение. — 1998. — № 4. — С. 22–27.
3. *Бернацкий Ф. И., Диго Г. Б., Диго Н. Б.* Применение многометодной технологии в робастном управлении // Информатика и системы управления. — 2002. — № 2(4). — С. 88–96.
4. *Бернацкий Ф. И., Диго Г. Б., Диго Н. Б.* Многометодная технология в задачах идентификации // Тр. II Междунар. конф. «Идентификация систем и задачи управления» (SICPRO'03). Москва, 2003. — С. 631–636.
5. *Бернацкий Ф. И., Диго Г. Б., Диго Н. Б.* Построение робастных областей допустимых управлений // Информатика и системы управления. — 2003. — № 1(5). — С. 92–100.
6. *Воеводин В. В., Воеводин Вл. В.* Параллельные вычисления. — СПб.: БХВ-Петербург, 2002. — 609 с.

☎ (4232) 31-02-02

E-mail: bernatsk@iacp.dvo.ru



ABSTRACTS

Popkov Yu.S.

MACROSYSTEMS AND GRID TECHNOLOGIES: DYNAMIC STOCHASTIC NETWORK MODELING

The paper discusses the development of Internet in the near-term outlook (Grid technologies) as a dynamic stochastic network whose state is characterized by the spatial distribution of data-processing resources and information flows. Based on the generalized entropy maximization principle, it offers the models of locally-steady states of the net, which are a component part of a data-processing resources evolution model. This model is classified as a positive dynamic system with an entropy operator. — P. 10.

Krapukhina N.V., Pastukhova K.M., Sviridov P.A.

ARTIFICIAL INTELLIGENCE TECHNIQUES IN ON-LINE CONTROL AND OPTIMIZATION OF COMPLEX PROCESS SYSTEMS

Some features of applying genetic algorithms in the optimization of complex continuous stochastic systems are discussed. The paper makes an attempt to improve the convergence of iterative techniques used for such systems calculation. — P. 21.

Bernatsky F.I., Digo G.B., Digo N.B.

PARALLEL COMPUTATIONS IN CONTROL PROBLEMS

The paper considers the problems of complex technical systems control under uncertainty conditions based on multimethod and multivariant analysis. It discusses the possibility of applying multiprocessor computers for computation multisequencing. — P. 25.