

© 2012 г. Д.А. НАЗАРОВ, канд. техн. наук  
(Институт автоматизации и процессов управления  
ДВО РАН, г. Владивосток)

## ПАРАЛЛЕЛЬНЫЕ ВЫЧИСЛЕНИЯ В ЗАДАЧЕ ПОСТРОЕНИЯ ОБЛАСТЕЙ РАБОТОСПОСОБНОСТИ<sup>1</sup>

Рассматривается задача построения областей работоспособности аналоговых технических систем, возникающая при проектировании с учетом параметрических возмущений и требований надежности. В качестве способа аппроксимации многомерной области используется дискретное множество элементарных параллелепипедов и метод многомерного зондирования. Предложен метод декомпозиции задачи для ее решения с привлечением параллельных вычислений с учетом возможности использования несимметричной вычислительной среды.

**PARALLEL COMPUTING IN THE TASK OF ACCEPTABLE REGION CONSTRUCTION** / D.A. Nazarov (Institute of Automation and Control Processes FEB RAS, 5 Radio Street, Vladivostok 690041, Russia, E-mail: nazardim@iacp.dvo.ru). The task of analogous engineering systems acceptable regions construction is considered. This task occurs at system design with the account of parameter deviations and reliability requirements. The basis of constructing of the region consists in representation of a multidimensional region with a discrete set of elementary boxes and multidimensional probing method. The method of the task decomposition for involving parallel computing with the account of non-symmetric computing environment is proposed.

### 1. Введение

Задача построения областей работоспособности (ОР) возникает на этапе параметрического синтеза при проектировании технических систем с учетом требований надежности [1]. Эта задача вытекает из процедуры исследования области допустимых значений параметров элементов проектируемой системы. При разработке сложных технических систем, а в особой степени — систем ответственного назначения, важно учитывать изменения значений параметров составляющих их элементов, возникающих при изготовлении, хранении и эксплуатации под влиянием ряда факторов как внешней среды, так и внутренних физико-химических процессов. Отклонения значений этих параметров влекут за собой отклонения основных характеристик функционирования, что может привести к отказу, в том числе с риском угрозы здоровью и жизни людей.

<sup>1</sup> Работа выполнена при частичной финансовой поддержке грантов ДВО РАН № 12-III-B-03-023 и № 12-I-OЭММПУ-01 в рамках Программы фундаментальных исследований ОЭММПУ РАН № 14 «Анализ и оптимизация функционирования систем многоуровневого, интеллектуального и децентрализованного управления в условиях неопределенности»

Проблема построения ОР заключается в невозможности получения аналитического описания границ области для подавляющего большинства сложных систем по причине отсутствия явных аналитических выражений зависимости выходных характеристик от параметров элементов. Как правило, модели, выражающие эти зависимости, задаются алгоритмически в виде численного решения систем дифференциальных уравнений или с помощью имитационных моделей типа «черного ящика». Например, поведение электрических цепей с целью получения вольт-амперных характеристик определенных их участков моделируется системой SPICE [2].

Поточечное исследование некоторой области поиска (как правило, область допусков) пространства значений параметров требует значительных вычислительных затрат. Это связано как со сложностью модели, так и необходимостью многократного расчета этой модели при различных реализациях вектора входных параметров (параметров элементов). В работе используется идея построения геометрического аналога ОР на основе многовариантного анализа системы в пространстве значений ее параметров и представлении многомерной области дискретным множеством элементарных параллелепипедов. В докладе рассматривается способ декомпозиции задачи построения ОР для ее решения с помощью технологии параллельных вычислений.

## 2. Постановка задачи построения области работоспособности

В рамках задачи построения ОР объектом исследования является модель проектируемой системы, связывающая вектор:

$$(1) \quad \mathbf{y} = (y_1, y_2, \dots, y_m),$$

выражающий выходные характеристики, которые интересуют потребителя, с вектором параметров составляющих ее элементов:

$$(2) \quad \mathbf{x} = (x_1, x_2, \dots, x_n).$$

Параметры (2) называются внутренними параметрами системы. Модель системы отражает зависимость вектора (1) от вектора (2):

$$(3) \quad y_i = y_i(\mathbf{x}), \quad \forall i = 1, 2, \dots, m,$$

которая обычно задается алгоритмически в виде численного решения системы дифференциальных уравнений или имитационной модели и реализует концепцию «черного ящика».

На выходные характеристики (1), как правило, налагаются определенные в техническом задании требования:

$$(4) \quad \mathbf{y}_{\min} \leq \mathbf{y}(\mathbf{x}) \leq \mathbf{y}_{\max}.$$

Эти зависимости называются условиями работоспособности (УР). Ограничения (4) определяют в пространстве внутренних параметров область:

$$(5) \quad D_x = \{x \in \mathbb{R}^n \mid \mathbf{y}_{\min} \leq \mathbf{y}(\mathbf{x}) \leq \mathbf{y}_{\max}\},$$

которая называется областью работоспособности [1].

Задача построения ОР (5) состоит в нахождении в пространстве внутренних параметров многомерную фигуру с известной конфигурацией, которая бы являлась приближением к неизвестной области  $D_x$  (5) при заданной модели (3), известных УР (4) и допусках на внутренние параметры, которые ограничивают область поиска.

Представление многомерной области дискретным множеством элементарных гиперпараллелепипедов по результатам многовариантного анализа системы в пространстве значений ее внутренних параметров описывается моделью (6):

$$(6) \quad G_R = (n, B, Q, S),$$

где  $n$  — размерность пространства внутренних параметров (2),  $B$  — гиперпараллелепипед, вершины которого определяются значениями допусков на внутренние параметры  $x_{i \min} \leq x_i \leq x_{i \max}, \forall i=1, 2, \dots, n$ ,  $Q = (q_1, q_2, \dots, q_n)$  - вектор, каждый  $i$ -й компонент которого показывает количество интервалов, на которое разбивается диапазон значений  $i$ -го параметра,  $S = (s_1, s_2, \dots, s_R)$  — набор индикаторов  $s_i \in \{0, 1\}$  принадлежности соответствующего элементарного параллелепипеда ОР, количество которых равно количеству элементарных параллелепипедов  $R = q_1 \cdot q_2 \cdot \dots \cdot q_n$ . Значением каждого индикатора является бинарный результат вычисления значения сеточной функции принадлежности (7), представляющей собой проверку УР (4), для пробной точки соответствующего ему параллелепипеда (Рис. 1) [3].

$$(7) \quad F(\mathbf{x}_c(k_1, k_2, \dots, k_n)) = \left\{ \begin{array}{l} 1, \quad \mathbf{y}_{\min} \leq \mathbf{y}(\mathbf{x}_c(k_1, k_2, \dots, k_n)) \leq \mathbf{y}_{\max}, \\ 0, \quad \mathbf{y}_{\min} > \mathbf{y}(\mathbf{x}_c(k_1, k_2, \dots, k_n)) \vee \mathbf{y}(\mathbf{x}_c) > \mathbf{y}_{\max} \end{array} \right\},$$

где  $\mathbf{x}_c(k_1, k_2, \dots, k_n)$  — центральная точка элементарного параллелепипеда, заданного набором индексов  $(k_1, k_2, \dots, k_n)$ , где  $k_i = 1, 2, \dots, q_i$ . Каждая координата этой точки вычисляются по значению соответствующего индекса и величины шага сетки:

$$(8) \quad x_{i_c} = x_{i \min} + (k_i - 1) \cdot h_i / 2,$$

где  $h_i = x_{i \max} - x_{i \min} / q_i$  — величина шага сетки по  $i$ -му параметру внутри его допуска.

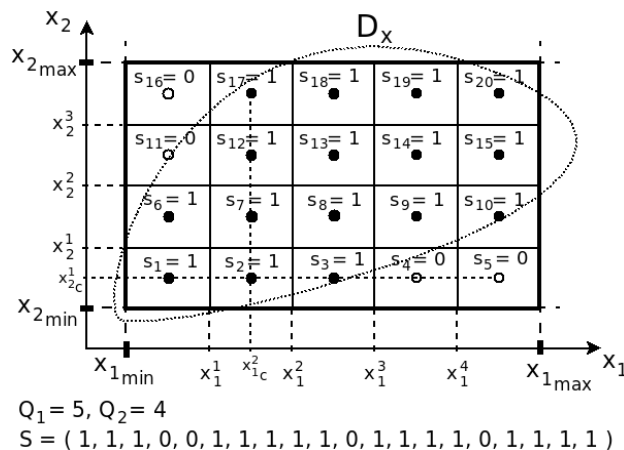


Рисунок 1. Дискретное представление двумерной ОР с помощью модели (6).

Взаимнооднозначное соответствие между набором индексов и индексом индикатора в массиве  $S$  устанавливается следующими соотношениями. Соотношение (9) устанавливает связь индекса  $p$  элемента массива индикаторов с набором индексов  $(k_1, k_2, \dots, k_n)$ :

$$(9) \quad p = k_1 + (k_2 - 1) \cdot q_1 + (k_3 - 1) \cdot q_1 \cdot q_2 + \dots + (k_n - 1) \cdot q_1 \cdot q_2 \cdot \dots \cdot q_{n-1}.$$

Процедура вычисления набора индексов  $(k_1, k_2, \dots, k_n)$  по заданному индексу  $p$  элемента массива индикаторов состоит в решении системы уравнений (10):

$$(10) \quad \begin{cases} k_n = \left[ \frac{p-1}{\prod_{i=1}^{n-1} q_i} \right] + 1, \\ k_{n-1} = \left[ \frac{p - (k_n - 1) \cdot \prod_{i=1}^{n-1} q_i}{\prod_{i=1}^{n-2} q_i} \right] + 1, \\ \dots \\ k_1 = p - \sum_{i=2}^n \left[ (k_i - 1) \cdot \prod_{j=1}^{i-1} q_j \right], n > 1 \end{cases},$$

где квадратные скобки у дробей означают операцию взятия целой части числа (антье).

Алгоритм построения ОР на основе модели (6) сводится к полному перебору элементарных параллелепипедов и вычислением функции принадлежности (7) для центральной точки каждого из них с записью результата в соответствующий по формуле (9) элемент массива индикаторов [3].

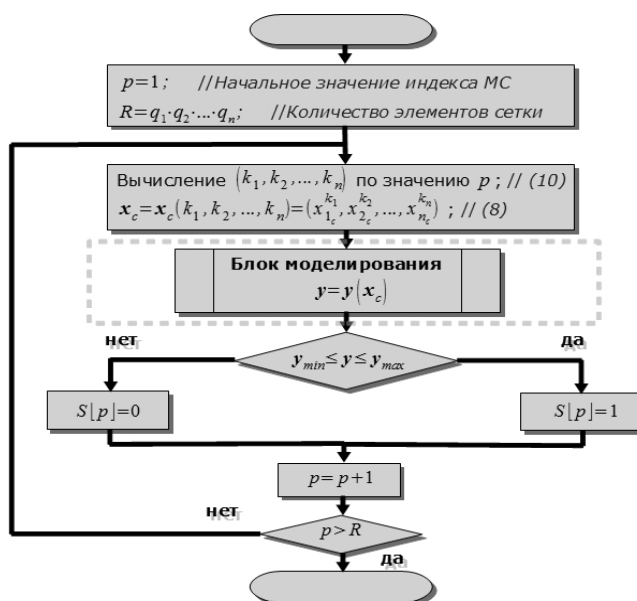


Рисунок 2. Алгоритм перебора элементарных параллелепипедов

Основными проблемами представления ОР с помощью модели (6) являются хранение и обработка большого объема данных. Больше всего ресурсов требуется для набора индикаторов  $S$ . Для повышения эффективности его обработки необходимы методы, позволяющие выполнять ее параллельно.

Главной особенностью предлагаемого подхода к перебору элементов сетки (элементарных параллелепипедов) является организация цикла не по индексам многомерной структуры, а по элементам одномерного массива индикаторов. Это позволяет произвести SPMD-декомпозицию задачи по данным. В этом случае один набор команд выполняется на нескольких различных процессорах для параллельной обработки разных частей массива. Таким образом, единственный цикл, блок-схема которого приведена на рисунке 2, разрывается на несколько параллельно выполняемых циклов.

### 3. Декомпозиция и распараллеливание задачи построения ОР

Как было сказано, процесс последовательной обработки независимых друг от друга элементов одномерного массива разбивается на  $K$  параллельно и независимо выполняемых процессов. В этом случае каждому  $i$ -му процессу сообщается диапазон (11) индексов элементов массива индикаторов  $S$

$$(11) \quad (p_{i\min}, p_{i\max}), \quad i=1,2,\dots,K, \quad p_{i\min}, p_{i\max} \in \{1,2,\dots,R\}.$$

Отдельный процесс помимо граничных значений диапазонов (11) должен знать параметры сетки (6) для корректного преобразования значений из этого диапазона в набор индексов  $(k_1, k_2, \dots, k_n)$  соответствующего элемента сетки с последующим вычислением его центральной точки. Для вычисления выходных характеристик (1) исследуемой системы, каждому процессу также должна быть передана ее модель (3) и УР (4).

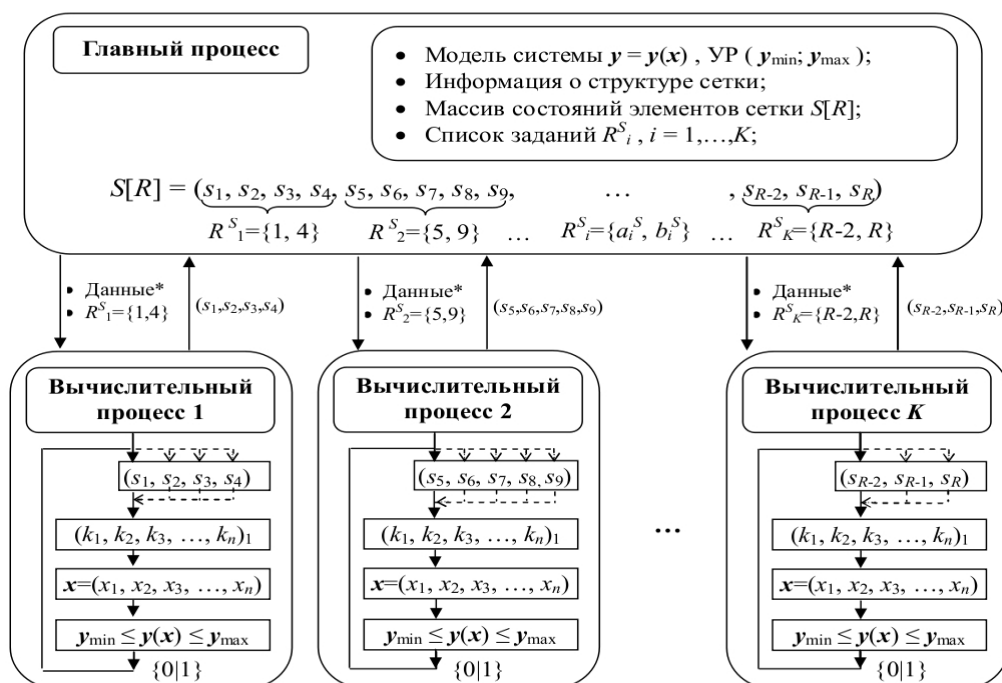


Рисунок 3. Алгоритм параллельной обработки массива индикаторов

На основе этой информации каждый отдельный процесс выполняет приведенный выше алгоритм перебора и заполняет свою часть массива индикаторов. Для координирования работы вычислительных узлов и сбора результатов их работы необходим главный процесс. Таким образом, топология параллельной вычислительной системы имеет централизованный тип «звезда». Алгоритм параллельной обработки массива индикаторов приведен на рисунке 3.

Приведенный способ декомпозиции позволяет разбивать обрабатываемый массив данных на различные по объему части. Потребность в таком разбиении может возникнуть при необходимости сбалансировать загрузку вычислительных узлов в случае несимметричной вычислительной среды. Для этого предлагается два способа декомпозиции [4].

### 3.1. Балансировка по вычислительной мощности

Очевидным решением задачи сбалансированной загрузки вычислительных узлов является раздача таких объемов данных, чтобы каждый отдельный процесс в силу доступных ему вычислительных возможностей завершил их обработку в один момент времени с остальными узлами. Для этого требуется предварительный замер производительности путем выполнения тестовой задачи. Пусть каждый из доступных  $K$  вычислительных узлов выполнил тестовое задание за  $T_i$  секунд. Тогда длина  $N_i$  обрабатываемой части массива индикаторов для каждого из этих узлов будет вычисляться следующим образом:

$$(12) \quad N_1 = \frac{R}{\left(1 + \frac{T_1}{T_2} + \frac{T_1}{T_3} + \dots + \frac{T_1}{T_K}\right)},$$

$$N_i = N_1 \frac{T_1}{T_i}, i = 2, 3, \dots, K$$

Из-за округления до целых при подсчете длины последнего диапазона  $N_K$  (12) желательно выполнить расчет по формуле:  $N_K = R - N_1 \cdot N_2 \cdot \dots \cdot N_{K-1}$ .

В результате применения этого метода балансировки на кластере, состоящем из пользовательских ПК локальной сети удалось добиться результатов, проиллюстрированных на рисунке 4.

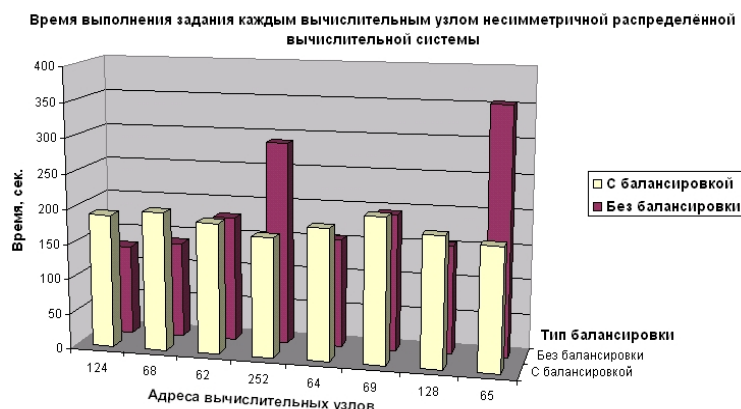


Рисунок 4. Применение балансировки по вычислительной мощности

К недостаткам такого типа балансировки стоит отнести, во-первых, тот факт, что после выполнения расчетов каждый вычислительный узел отправляет главному процессу свою часть массива, что вызывает повышенную нагрузку как на коммуникационную среду, так и на сам центральный узел, во-вторых, в случае ненадежных каналов коммуникации или выхода из строя одного из узлов весьма существенно скажется на эффективности параллельной обработки негативный эффект потери одной из части результатов вычислений. В третьих, такой тип балансировки не эффективен для случая использования пользовательских ПК в качестве вычислительных узлов, поскольку он не учитывает динамику изменения загруженности процессора пользовательскими задачами.

### 3.2. Балансировка очередью заданий

Другим способом балансировки является организация запроса со стороны вычислительных узлов заданий, имеющих на центральном узле, и их раздача по мере выполнения предыдущих. В этом случае массив индикаторов разбивается на относительно большое количество малых и одинаковых по объему заданий. Каждый вычислительный процесс к концу решения задачи обработает столько заданий, сколько ему позволят доступные ему вычислительные ресурсы. Преимущества такого способа декомпозиции преодолевают указанные выше недостатки балансировки по вычислительным мощностям, но при этом возникает проблема оптимального объема одного задания, в котором должны быть учтены такие факторы как пропускная способность коммуникационной среды, время выполнения одного задания на самом мощном вычислительном узле. Увеличение количества заданий влечет за собой рост передаваемой служебной информации. Однако стоит отметить, что в случае различных вычислительных мощностей узлов возникает «перераспределение» вычислительных и сетевых ресурсов таким образом, что вычислительные узлы обмениваются данными с центральным в разное время, а коллизии случаются относительно редко. При неоправданном увеличении количества заданий отмечается падение эффективности параллельной обработки из-за увеличивающегося числа коллизий и роста объема передаваемых данных, что проиллюстрировано на рисунке 5.



Рисунок 5. Влияние количества заданий в очереди на время решения задачи

Эффективность балансировки вычислительной загрузки узлов зависит от объема данных в каждом задании. С одной стороны, чем мельче разбиение, тем более качественной балансировки можно достичь, однако с другой стороны, как уже говорилось, с измельчением заданий и увеличением их количества возникает больше коллизий при передаче результатов центральному узлу, что вызывает простаивание вычислительных узлов. На рисунке 6 приведена гистограмма балансировки загрузки узлов.



Рисунок 6. Загрузка вычислительных узлов при балансировке очередью заданий

#### 4. Заключение

Применение описанного в работе параллельного алгоритма построения ОР на основе многовариантного анализа и представлении многомерной области дискретным множеством элементарных гиперпараллелепипедов на распределенной в локальной вычислительной сети из 8 пользовательских компьютеров с учетом балансировки вычислительной загрузки узлов позволило достичь ускорения, равного 5,33. Ввиду необходимости хранения и обработки большого объема данных, перспективой развития предложенного подхода к построению ОР является реализация параллельного алгоритма для высокопроизводительного вычислительного кластера и организация доступа к этой вычислительной системе на основе технологии облачных вычислений [5].

#### СПИСОК ЛИТЕРАТУРЫ

1. *Абрамов О.В.* Параметрический синтез стохастических систем с учетом требований надежности. М.: Наука, 1992.
2. *Kundert K.S., Zinke O.* The Designer's Guide to SPICE & SPECTRE. Springer, 2004.
3. *Назаров Д.А.* Использование областей работоспособности для оптимального выбора номиналов параметров // Информатика и системы управления. 2011. № 2(28). С. 59 — 69.
4. *Назаров Д.А.* Разработка алгоритмических и программных средств построения и анализа областей работоспособности аналоговых технических систем: Дис... канд. техн. наук / Назаров Д.А. Владивосток: ИАПУ ДВО РАН, 2011.
5. *N. Antonopoulos, L. Gillam* Cloud Computing: Principles, Systems and Applications. Springer, 2010.