УДК 519.872, 519.688, 681.63

С.И. ЛАВРЕНЮК, А.Ю. ШЕЛЕСТОВ

ИДЕНТИФИКАЦИЯ МОДЕЛЕЙ И ОЦЕНКА СОСТОЯНИЯ GRID-СИСТЕМ

Ключевые слова: *Grid-система, идентификация моделей, обобщенный вектор показателей системы, структурно-функциональный анализ, нечеткие эллипсоидальные множества, нейросетевая модель*.

Введение. Постановка задачи оптимизации Grid-системы, рассматриваемой как функционирующий в условиях неопределенности объект управления, к которому можно применять методы идентификации и оценивания и на основе обратной связи обеспечить управление этим объектом, — это планирование выполнения задач, для этого нужно оценить текущее состояние распределенной системы и загрузку ее отдельных ресурсов. Отметим, что до сих пор теория управления для разработки и исследования Grid-систем практически не использовалась, поскольку такие системы сложные и иерархические, чем обусловлена объективная трудность их строгого аналитического описания. Среди литературных источников доступны лишь несколько работ, связанных с применением теории управления для моделирования нагрузки и планирования вычислений в Grid-среде. Так, в одной из них для оценки состояний линейной модели Grid-системы применен фильтр Калмана [1].

В статье на основе результатов структурно-функционального анализа (СФА), полученных в [2], для описания функционирования разных Grid-ресурсов предложен ряд моделей таких систем. В частности нагрузку на отдельный узел Grid-системы можно описать моделью линейного объекта управления с неизвестными возмущениями, а неопределенности представить с помощью теоретико-множественного подхода [3–5]. При этом для записи граничных значений неизвестных величин оценивание неизвестных переменных выполняется на основе нечетких эллипсоидальных множеств [6]. Для описания функционирования сегмента или Grid-системы в целом и более полного учета показателей функционирования предложен кибернетический подход к моделированию, когда система рассматривается как «черный ящик» и с учетом ее нелинейности описывается нейросетевой моделью. Для адаптивности и робастности процесса идентификации матриц весовых коэффициентов использован метод рекуррентного оценивания параметров этой модели [7].

Задачи оценивания состояний и параметров компонентов структурной модели Grid-системы. В [2, 8] рассмотрена задача СФА Grid-системы для задач исследования Земли (EOGrid) и получено формальное описание такой системы. Отметим, что структурную декомпозицию без ограничения общности можно применять для Grid-систем. Именно в таком контексте следует рассматривать данную статью.

Результат структурной декомпозиции системы показан на рис. 1.

Количество показателей функциональных элементов (ФЭ) всех уровней иерархии системы (системы в целом, сегментов, узлов и компонентов) определяет соотношение

$$X = \{\mathbf{x}_{qp},\ p = \overline{1, P},\ q = \overline{1, N}\}, \tag{1}$$

а обобщенный вектор показателей — формула

$$\mathbf{x} = (x_1, x_2, \ldots, x_{N_0})^{\mathrm{T}},\ \mathbf{x} \in \mathbf{R}^{N_0}. \tag{2}$$

Как следует из [2], одной из главных задач структурно-функционального анализа является определение преобразования

$$F: X \to Y \tag{3}$$

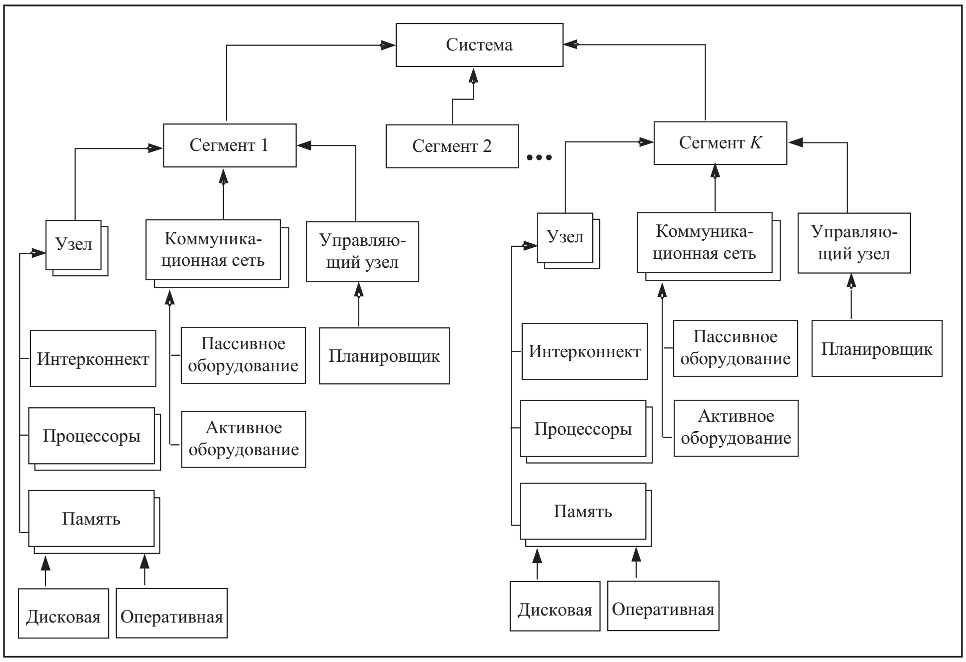

Рис. 1

из множества X допустимых показателей системы в пространство Y необходимых свойств по набору количественных и качественных требований. При этом задача структурно-параметрической идентификации разрешает одновременно определять структуру иерархической системы в целом и ФЭ всех уровней иерархии, а также вид (3).

Поскольку Grid-система на рис. 1 относится к классу СМИС [9], являясь иерархической и существенно-нелинейной, все аспекты ее функционирования сложно описать единой моделью. Поэтому выделим задачи идентификации и оценивания, в которых можно ограничить уровень абстракции модели с учетом существующей информации.

Формирование вектора состояния узла. Определим переменные, подлежащие оцениванию в процессе функционирования системы. Согласно (2), (3) ФЭ Grid-узла характеризуются векторами показателей. Вектор показателей $\mathbf{x}_{\text{proc}}$ для процессоров согласно (3) имеет вид

$$\mathbf{x}_{\text{proc}}^{\text{T}} = (x_{\text{arch}}, x_{\text{freq}}, x_{\text{load}})^{\text{T}},$$

где x_{arch} — архитектура процессора; x_{freq} — его частота; x_{load} — загрузка процессора (средняя за некоторый период времени). Первые две компоненты этого вектора не изменяются во времени и их значение определяется типом процессора. А загрузка процессора (компонента x_{load}) динамически изменяется и подлежит оцениванию при функционировании системы.

Для оперативной памяти вектор показателей имеет вид

$$\mathbf{x}_{\text{mem}}^{\text{T}} = (x_{\text{vol}}, x_{\text{vol\_total}}, x_{\text{rate}})^{\text{T}},$$

где x_{vol} — объем свободной памяти в данный момент времени; $x_{\text{vol\_total}}$ — общий объем памяти; x_{rate} — скорость доступа к памяти для чтения и записи. Для этого вектора последние две компоненты неизменны и определяются архитектурой модуля памяти, а объем свободной памяти x_{vol} динамически изменяется и подлежит оцениванию.

Для дисковой памяти (или хранилищ) вектор показателей имеет вид

$$\mathbf{x}_{\text{storage}}^{\text{T}} = (\mathbf{x}_{\text{seq}}^{\text{T}}, \mathbf{x}_{\text{par}}^{\text{T}}, x_{\text{vol}}, x_{\text{vol\_total}})^{\text{T}},$$

где $\mathbf{x}_{\text{seq}}$ — вектор, характеризующий зависимость пропускной способности хранилища от количества одновременных запросов для последовательного (потокового) доступа; $\mathbf{x}_{\text{par}}$ — вектор, характеризующий зависимость пропускной способности хранилища от количества одновременных запросов для параллельного доступа; x_{vol} — объем свободного пространства в хранилище в данный момент времени; $x_{\text{vol\_total}}$ — общий объем хранилища. Для вектора $\mathbf{x}_{\text{storage}}$ компонентой, подлежащей оцениванию в процессе функционирования системы, является x_{vol}. Оценки двух первых компонент $\mathbf{x}_{\text{seq}}$ и $\mathbf{x}_{\text{par}}$ являются табличными значениями от разработчиков аппаратного обеспечения.

Для управляющего узла используем вектор показателей

$$\mathbf{x}_{\text{control}}^{\mathrm{T}} = (\mathbf{x}_{\text{node\_queue}}^{\mathrm{T}}, x_{\text{node}}, \mathbf{x}_{\text{node\_load}}^{\mathrm{T}}, \mathbf{x}_{\text{queue}}^{\mathrm{T}})^{\mathrm{T}},$$

где $\mathbf{x}_{\text{node\_queue}} = (x_1, x_2, \ldots, x_k)$ — длина очереди задач на подконтрольных узлах; $\mathbf{x}_{\text{queue}}$ — длина очереди на управляющем узле; $x_{\text{node}} = k$ — количество подконтрольных узлов; $\mathbf{x}_{\text{node\_load}}$ — распределение загрузки подконтрольных узлов (среднее за некоторый период времени). Для данного вектора следует оценить векторы $\mathbf{x}_{\text{node\_queue}}$, $\mathbf{x}_{\text{queue}}$ и $\mathbf{x}_{\text{node\_load}}$.

Согласно рис. 1 в состав вычислительного узла Grid-системы входит еще интерконнект. Поскольку при построении Grid-систем, во-первых, используются средства обеспечения межаппаратного взаимодействия (сетевые адаптеры, концентраторы, переключатели и т.д.), а во-вторых, параметры таких устройств не изменяются во времени, то при оценивании нагрузки на вычислительные и управляющие узлы Grid-системы соответствующие компоненты вектора их состояния являются постоянными. Следовательно, без потери общности их можно исключить из рассмотрения.

Все компоненты оцениваемых векторов измеряются в относительных единицах или в процентах и обеспечивают один диапазон допустимых значений, который для удобства можно пронормировать и привести к диапазону изменения от 0 до 1.

Отметим, что для отдельных компонент сформированных выше векторов состояния узлов Grid-системы характерны разные интервалы дискретизации для получения новых измерений (значит, и соответствующих оценок). Например, значения компоненты x_2 вектора $\mathbf{x}_{\text{node}}$ обычно изменяются в течение нескольких минут, а компонента x_3 изменяется гораздо медленнее. Поэтому общий интервал дискретизации для получения текущих оценок вектора состояния целесообразно выбирать из практических соображений. Например, можно учитывать продолжительность выполнения прикладных задач, которая может составлять несколько часов. В частности, известно, что задача получения численного прогноза погоды требует порядка 4 часов счета на 4-процессорном узле, задача подбора параметров метеорологической модели WRF — 1,5 часа и т.д. В качестве интервала дискретизации целесообразен интервал 5 мин, поскольку, с одной стороны, этот интервал гораздо меньше общего времени вычислений и позволяет динамически отслеживать изменение нагрузки на ресурсы. С другой стороны, такой интервал существенно превышает интервал дискретизации измерений (например, для центрального процессора в течение секунды выполняются миллионы инструкций).

Такой интервал дискретизации обусловлен также возможностями существующих программных средств, позволяющих реализовать механизм опроса ресурсов Grid-системы. Например, Spring [10] представляет требуемые оценки в фиксированные интервалы времени (1 раз за 5 мин). Следует учитывать также, что возможность получения измерительной информации сильно зависит от текущей нагрузки на сеть и другие ресурсы, т.е. при использовании стандартных средств для получения текущих оценок вектора состояния узла не гарантируется своевременность поступления требуемой информации. Для функционирования механизмов планирования распределения нагрузки необходимы модель и алгоритмы оценивания текущего значения вектора состояния.

В пользу модели и вычисления оценок вектора состояния узла Grid-системы свидетельствует также тот факт, что ресурсы этой системы принадлежат к разным административным доменам, в которых принятая политика безопасности и обеспечения целостности не допускает использования дополнительных средств тестирования и сбора статистики. В то же время стандартные программные средства диагностики и ведения системных журналов не могут предоставить всей необходимой информации. По сути оценивание текущих значений вектора состояния на основе модели — единственное средство получения данных. Итак, при оценивании нагрузки узла требуется оценить вектор состояния

$$\mathbf{x}_{\text{node}} = (\mathbf{x}_1^{\mathrm{T}}, \mathbf{x}_2^{\mathrm{T}}, \mathbf{x}_3^{\mathrm{T}})^{\mathrm{T}}, \tag{4}$$

где $\mathbf{x}_1 \in \mathbf{R}^n, \mathbf{x}_2 \in \mathbf{R}^m, \mathbf{x}_3 \in \mathbf{R}^k$, причем размерности векторов $\mathbf{x}_1, \mathbf{x}_2$ и $\mathbf{x}_3$ определяются конфигурацией аппаратных средств (число процессоров, блоков оперативной памяти и жестких дисков), а значения компонент этих векторов соответствуют значениям загрузки процессоров, оперативной памяти и жесткого диска соответственно.

В частности для обычного однопроцессорного компьютера с одним жестким диском и единым пространством оперативной памяти вектор $\mathbf{x}_{\text{node}}$ имеет размерность 3. Для четырехпроцессорного вычислителя с общей памятью первая компонента $\mathbf{x}_1$ вектора $\mathbf{x}_{\text{node}}$ уже имеет размерность 4 (по числу процессоров). Тогда общая размерность $\mathbf{x}_{\text{node}}$ составляет 6. Если в качестве узла использован кластер, состоящий из $n = 127$ отдельных узлов (например, СКИТ-3 Института кибернетики им. В.М. Глушкова НАНУ), у каждого из которых есть свое адресное пространство и жесткий диск, то вектор $\mathbf{x}_{\text{node}}$ примет вид

$$\mathbf{x}_{\text{node}}^{\mathrm{T}} = (\mathbf{x}_1^{\mathrm{T}}, \mathbf{x}_2^{\mathrm{T}}, \mathbf{x}_3^{\mathrm{T}})^{\mathrm{T}}, \tag{5}$$

где $\mathbf{x}_1 = (x_{11}, x_{12}, \ldots, x_{1n})^{\mathrm{T}}$, $\mathbf{x}_2 = (x_{21}, x_{22}, \ldots, x_{2n})^{\mathrm{T}}$, $\mathbf{x}_3 = (x_{31}, x_{32}, \ldots, x_{3n})^{\mathrm{T}}$, причем $x_{1i} = x_{\text{load}\, i}$, $x_{2i} = x_{\text{vol}\, i}$ и $x_{3i} = x_{\text{vol}\, i}$ определяют текущее состояние процессора, оперативной памяти и жесткого диска i-го узла кластера соответственно. При этом размерность векторов $\mathbf{x}_1$, $\mathbf{x}_2$, $\mathbf{x}_3$ определяется количеством используемых вычислительных узлов.

Для управляющего узла оцениваемый вектор состояния имеет вид

$$\mathbf{x}_{\text{control}}^{\mathrm{T}} = (\mathbf{x}_1^{\mathrm{T}}, x_2^{\mathrm{T}}, \mathbf{x}_3^{\mathrm{T}}), \tag{6}$$

где $\mathbf{x}_1 = \mathbf{x}_{\text{node\_queue}}$, $x_2 = x_{\text{queue}}$, $\mathbf{x}_3 = \mathbf{x}_{\text{node\_load}}$.

Схема процесса оценивания загрузки Grid-узла. В немногочисленной литературе (скажем, в работах [1, 11]) нагрузка на вычислительные и информационные ресурсы одного узла Grid-системы оценивается на основе линейной модели объекта управления. В этих работах рассмотрены простые модели, не адекватные сложности задачи и оценивающие скалярную величину нагрузки на ресурс. Поскольку согласно (5) и (6) Grid-система характеризуется вектором показателей, для описания ее функционирования целесообразна более адекватная модель объекта управления в векторном пространстве состояний с неизвестными возмущениями, которая остается в классе линейных моделей. Для оценивания состояния такой модели в условиях нестатистически заданной неопределенности целесообразен теоретико-множественный подход [5], в частности метод эллипсоидального оценивания [3, 4]. При отсутствии достоверной априорной информации об оцениваемых величинах предлагается строить нечеткие множественные оценки неизвестных векторов [12].

При отсутствии уточняющей информации о структуре Grid-узла его загрузку целесообразно моделировать на основе информационного или кибернетического подхода, используя модель в виде «черного ящика» с измеряемыми входами и выходами. Для математического описания таких нелинейных моделей удобны нейронные сети, обеспечивающие адекватное представление моделей, как в терминах «вход-выход» (модель NARX), так и в пространстве состояний [7]. В [13] показано, что нейронной сетью персептронного типа с одним скрытым слоем можно аппрок-

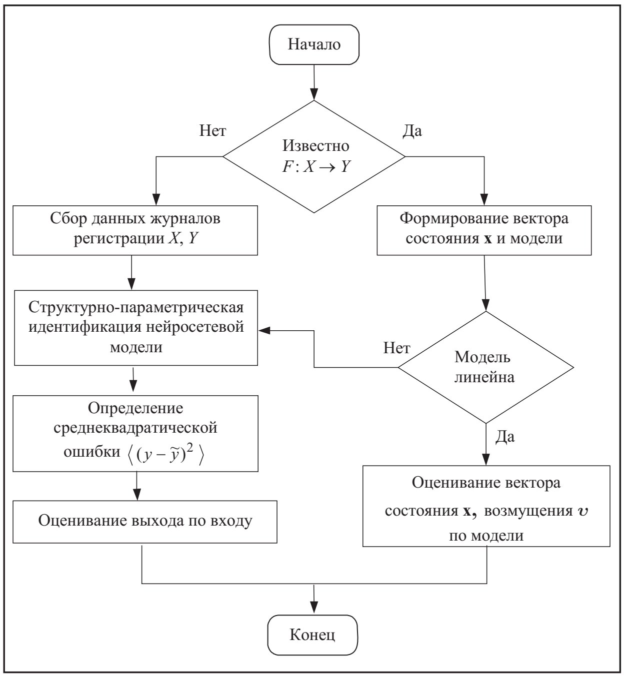

Рис. 2

симировать непрерывную функциональную зависимость любого порядка. Для преодоления недостатков традиционных алгоритмов настройки нейросетевых моделей, в частности для обеспечения адаптивности и робастности процесса идентификации матриц весовых коэффициентов, используют рекуррентное оценивание параметров этой модели [7].

Общая схема оценивания загрузки узла Grid-системы в зависимости от типа модели и степени неопределенности приведена на рис. 2.

Построение линейной модели. Пусть на узел Grid-системы направлена вычислительная задача. Положим, что в первом приближении изменение вектора показателей загрузки этого узла описывается линейной моделью объекта управления вида

$$\mathbf{x}_{k+1} = \bar{A}\mathbf{x}_k \,, \qquad (7)$$

где $\mathbf{x}_k \in \mathbf{R}^n$ — вектор состояния, семантика которого описывается соотношениями (4) и (6) для вычислительного и управляющего узлов соответственно, $\bar{A} = \mathrm{diag}\,\{a_1, \dots, a_n\}$ — невырожденная диагональная матрица размерности $n \times n$. Матрица является диагональной, поскольку загрузка каждого ФЭ (т.е. значение каждой компоненты вектора $\mathbf{x}_k$) зависит от значений его загрузки в предыдущие моменты времени и не зависит от загрузки остальных ФЭ (например, загрузка процессора не связана со степенью загрузки оперативной памяти и коэффициентом использования жесткого диска).

Правая часть приведенного соотношения описывает динамику изменения загрузки узла в процессе выполнения задачи без учета возможности поступления новых задач. Учитывая физический смысл переменных в модели, можно считать, что матрица $\bar{A}$ устойчива, т.е. норма $\|A\| < 1$. Действительно, если в систему не поступают новые задания, то загрузка ресурсов, определяемая векторами (4) и (6), монотонно снижается. Если в систему не поступают новые задачи, то в результате переходного процесса, связанного с выполнением текущей задачи, вектор состояния системы (показателей загрузка ее ФЭ) переходит в равновесное положение $\mathbf{x}^* = 0$.

Например, для тестовой задачи вычисления площадей затопленных территорий на однопроцессорном компьютере динамика изменения вектора состояния узла представлена на рис. 3. Такую временную зависимость можно аппроксимировать соотношением (7) при

$$\bar{A} = \begin{pmatrix} 0.33 & 0 & 0 \\ 0 & 0.30 & 0 \\ 0 & 0 & 0.30 \end{pmatrix}.$$

Отметим, что зачастую для периодического решения конкретной задачи выделяется отдельный Grid-узел [2]. Если на некотором узле Grid-сегмента периодически запускается одна и та же задача, для оценивания загрузки этого узла можно использовать линейную модель, предварительно идентифицировав ее параметры $\bar{A}$. Для оценивания матрицы $\bar{A}$ по текущим измерениям вектора состояния можно использовать множественный подход [5, 14]. Варианты практической реализации этого подхода в классе эллипсоидальных множественных оценок для различных критериев предложены в [15–17].

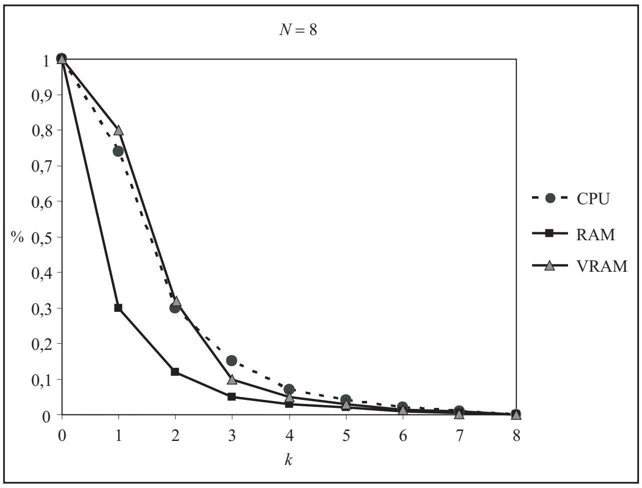

Рис. 3

Известны также алгоритмы решения задачи одновременного оценивания параметров и состояний, например, в [18, 19], однако точность получаемых оценок вектора состояний при достаточно грубых оценках параметров системы неудовлетворительна. Поэтому целесообразно разделить во времени процессы идентификации параметров и оценивания состояний, выполняя идентификацию параметров в строго отведенные периоды, когда владелец административного домена допускает проведение серии экспериментов по измерению состояния его ресурсов.

Поскольку матрица коэффициентов (7) имеет канонический вид $\overline{A} = \text{diag}\,\{a_1, \ldots, a_n\}$, то (7) сводится к набору из n скалярных соотношений $x^i_{k+1} = a_i x^i_k$, $i = \overline{1, n}$.

Следовательно, задача оценивания матрицы параметров вырождается в n независимых подзадач оценивания каждого из элементов a_i, $i = \overline{1, n}$. В рамках теоретико-множественной трактовки неопределенности можно считать, что $a_i \in [a_i^-, a_i^+]$, $a_i^- < a_i^+$, $a_i^-, a_i^+ \in R$. Следовательно, задача оценивания параметров сводится к тривиальной задаче определения границ и центра отрезка на вещественной прямой, как точечной оценки $\hat{a}_i$, $i = \overline{1, n}$.

Поступление новых задач в систему рассмотрим как неконтролируемое внешнее возмущение. Без потери общности считаем его ограниченным, например, физической длиной очереди заданий, поступающих планировщику. Изменение вектора состояния Grid-узла опишем модифицированным соотношением $\mathbf{x}_{k+1} = \overline{A}_k \mathbf{x}_k + B_k \boldsymbol{\upsilon}_k$, где $\mathbf{x}_k \in \mathbf{R}^n$ — вектор состояния, семантика которого описывается соотношениями (4) и (6) для вычислительного и управляющего узлов соответственно, $\boldsymbol{\upsilon}_k \in \mathbf{R}^l$ — вектор неконтролируемых возмущений размерности $l \le n$, $\overline{A}_k$ — невырожденная кусочно-постоянная матрица размерности $n \times n$, $\overline{A} = \text{diag}\,\{a_1, \ldots, a_n\}$, B_k — $(n \times l)$-матрица.

Первое слагаемое правой части соотношения описывает динамику изменения загрузки узла в процессе выполнения задачи без учета возможности поступления новых задач, а второе слагаемое позволяет учесть возрастание нагрузки на данный ФЭ с появлением новой задачи. Поскольку вновь поступающая задача может распределяться не на все ФЭ узла, размерность вектора возмущений $\boldsymbol{\upsilon}_k \in \mathbf{R}^l$ не превышает размерности вектора состояния системы $l \le n$. Вид матрицы B_k определяется алгоритмом планирования, реализованным для узла (в простейшем случае при отсутствии специального алгоритма планирования размерности векторов состояния и возмущения совпадают $l = n$, а матрица B_k представляет собой единичную матрицу размерности $n \times n$).

Поскольку не все показатели загрузки ФЭ Grid-узла доступны непосредственному измерению (например, при использовании многоядерных процессоров непосредственному измерению поддается общая (интегральная) загрузки процессоров (ядер), а не каждого ядра в отдельности), положим, что измеряемый выход системы в каждый момент дискретного времени описано соотношением $y_k = \overline{U}_k^{\mathrm{T}} \mathbf{x}_k$, где $y_k \in \mathbf{R}^3$ — наблюдаемый выход объекта, $\overline{U}_k$ — матрица размерности $(n+m+k) \times 3$ вида

$$U^{\mathrm{T}} = \begin{pmatrix} \mathbf{U}_{\mathrm{proc}}^{\mathrm{T}} & 0 & 0 \\ 0 & \mathbf{U}_{\mathrm{mem}}^{\mathrm{T}} & 0 \\ 0 & 0 & \mathbf{U}_{\mathrm{storage}}^{\mathrm{T}} \end{pmatrix},$$

где $\mathbf{U}_{\mathrm{proc}}^{\mathrm{T}}$, $\mathbf{U}_{\mathrm{mem}}^{\mathrm{T}}$, $\mathbf{U}_{\mathrm{storage}}^{\mathrm{T}}$ — векторы размерностей n, m и k соответственно; n, m, k — количество процессоров, блоков оперативной памяти и жестких дисков вычислительного узла соответственно. Векторы $\mathbf{U}_{\mathrm{proc}}^{\mathrm{T}}$, $\mathbf{U}_{\mathrm{mem}}^{\mathrm{T}}$, $\mathbf{U}_{\mathrm{storage}}^{\mathrm{T}}$ имеют следующую структуру:

$$\mathbf{U}_{\mathrm{proc}}^{\mathrm{T}} = (1/n, \ldots, 1/n)^{\mathrm{T}},$$

$$\mathbf{U}_{\mathrm{mem}}^{\mathrm{T}} = (1/m, \ldots, 1/m)^{\mathrm{T}},$$

$$\mathbf{U}_{\mathrm{storage}}^{\mathrm{T}} = (1/k, \ldots, 1/k)^{\mathrm{T}}.$$

Вид матрицы U определяется семантикой решаемой задачи.

Для управляющего узла матрица U имеет вид

$$U^{\mathrm{T}} = \begin{pmatrix} \mathbf{U}_{\mathrm{load\_queue}}^{\mathrm{T}} & 0 & 0 \\ 0 & u_{\mathrm{queue}} & 0 \\ 0 & 0 & \mathbf{U}_{\mathrm{node\_load}}^{\mathrm{T}} \end{pmatrix},$$

где $\mathbf{U}_{\mathrm{load\_queue}}^{\mathrm{T}}$, $\mathbf{U}_{\mathrm{node\_load}}^{\mathrm{T}}$ — векторы размерности k, k — количество подконтрольных узлов; $u_{\mathrm{queue}} \in \mathbf{R}^1$. Векторы $\mathbf{U}_{\mathrm{load\_queue}}^{\mathrm{T}} \in \mathbf{R}^k$, $\mathbf{U}_{\mathrm{node\_load}}^{\mathrm{T}} \in \mathbf{R}^k$ и u_{queue} имеют следующую структуру:

$$\mathbf{U}_{\mathrm{load\_queue}}^{\mathrm{T}} = (1, \ldots, 1)^{\mathrm{T}},$$

$$\mathbf{U}_{\mathrm{node\_load}}^{\mathrm{T}} = (1, \ldots, 1)^{\mathrm{T}},$$

$$u_{\mathrm{queue}} = 1.$$

Задачу оценивания нагрузки для узла можно рассматривать как задачу оценивания вектора состояния линейной системы управления с неконтролируемыми возмущениями. Описание задачи в терминах теории управления и метод ее решения см. в [20].

Обсуждение линейной модели. Область применения предложенной модели состояния Grid-узла ограничивается ситуацией, когда известна иерархическая структура узла и изменение загрузки узла описывается линейной моделью.

В работах [21–23] для оценивания и прогнозирования нагрузки на ресурс Grid-системы используются линейные регрессионные модели, преимущественно первого порядка (с глубиной памяти 1). Такой выбор простейшей модели объясняется недостатком априорной информации. При наличии данных о структуре и порядке модели систему можно представлять в виде объекта управления в переменных «вход–выход», решая для него задачу параметрической идентификации одним из классических методов теории управления и идентификации [24]. Однако при оценивании состояния Grid-систем порядок модели точно неизвестен, поэтому приходится решать задачу не только параметрической, но и структурной идентификации [25]. Для этого предложены интеллектуальные методы поиска, в частности генетические алгоритмы [26].

В случае динамического распределения нагрузки на узел представить модель изменения загрузки узла в аналитической форме не представляется возможным. В этом случае целесообразно воспользоваться аппроксимацией (разложением по базовым функциям) и выбрать модель в форме «черного ящика». Согласно [13] универсальным аппроксиматором является нейронная сеть персептронного типа с одним скрытым слоем.

Заключение. В зависимости от постановки задачи и наличия информации о структуре и порядке Grid-системы рассмотрено три класса моделей: линейная модель переменных состояния с неизвестными возмущениями, модель переменных «вход–выход» и нейросетевая модель, применяемая для описания нелинейных объектов общего вида. Для оценки состояний и идентификации моделей применяются элементы интеллектуальных вычислений: нестатистическая неопределенность описана в классе нечетких множеств, для структурно-параметрической идентификации модели в терминах «вход–выход» использован генетический алгоритм.

В контексте исследования Grid-систем можно сформулировать следующие условия применимости рассмотренных моделей. При наличии всей информации, необходимой для распределения нагрузки по доступным ресурсам, можно применять строгие и четко определенные алгоритмы маршрутизации и планирования. Методы теории управления оказываются полезными при отсутствии информации о параметрах, показателях функционирования и текущей нагрузке системы. Такая ситуация характерна для большинства Grid-систем со сложными и динамически поступающими заданиями. На основе предложенного подхода можно идентифицировать структуру и параметры Grid-системы в целом или ее отдельного узла, оценить обобщенный вектор показателей, а затем применить его при реализации (оптимизации) политики распределения задач.

СПИСОК ЛИТЕРАТУРЫ

1. Tian Z., Liu L., Yang Y., Zhai Z. A stochastic control model for hierarchical grid service // NPC 2005, LNCS 3779. Ed. by H. Jin, D. Reed, W. Jiang. — 2005. — P. 72–79.
2. Куссуль Н.Н., Шелестов А.Ю. Grid-системы для задач исследования Земли. Архитектура, модели и технологии. — Киев: Наук. думка, 2008. — 452 с.
3. Бакан Г.М. Аналитический синтез алгоритмов гарантированного оценивания состояний динамических объектов // Проблемы управления и информатики. — 2003. — № 3. — С. 38–55.
4. Волосов В.В. Робастные алгоритмы эллипсоидального оценивания состояния многомерных нестационарных непрерывных динамических систем // Там же. — 1999. — № 1. — С. 38–52.
5. Кунцевич В.М., Лычак М.М. Синтез оптимальных и адаптивных систем управления. Игровой подход. — Киев: Наук. думка, 1985. — 286 с.
6. Куссуль Н.Н., Шелестов А.Ю. Оценивание фазового состояния линейных многомерных динамических объектов с использованием размытых эллипсоидальных множеств // Проблемы управления и информатики. — 1995. — № 1. — С. 53–63.
7. Куссуль Н.Н., Шелестов А.Ю. Идентификация нейросетевых моделей объектов управления в классе нечетких множественных оценок // Тр. Одес. Политехн. ун-та. — 2001. — Вып. 3(15). — С. 124–127.
8. Шелестов А.Ю. Структурно-функциональный анализ компонентов Grid-систем // Проблемы управления и информатики. — 2007. — № 5. — С. 119–132.
9. Згуровский М.З., Панкратова Н.Д. Системный анализ: проблемы, методология, приложения. — Киев: Наук. думка, 2005. — 744 с.
10. Johnson R., Hoeller J., Arendsen A., Sampaleanu C. The Spring framework — reference documentation. — http://springframework.org.
11. Prem H., Srinivasa Raghavan N.R. A support vector machine based approach for forecasting of network weather services // J. of Grid Comput. — 2006. — **4.** — P. 89–114.
12. Куссуль Н.Н., Шелестов А.Ю. Нечеткий эллипсоидальный наблюдатель состояния линейных динамических объектов с неизвестными возмущениями // Праці Міжнар. конф. з управління «Автоматика-2000», Львів, 2000. — С. 149–154.
13. Reed D., Marks J. II Neural smithing. Supervised learning in feedforward artificial neural networks. — Cambridge, Massachusetts, London: A Bradford Book, 1999. — 346 p.
14. Кунцевич В.М. Управление в условиях неопределенности: гарантированные результаты в задачах управления и идентификации. — Киев: Наук. думка, 2006. — 264 с.
15. Polyak B.T., Nazin S.A., Durieu C., Walter E. Ellipsoidal parameter or state estimation under model uncertainty // Automatica. — 2004. — **40.** — P. 1171–1179.
16. Волосов В.В., Одинцова Е.А., Храмов С.А. Алгоритмы эллипсоидального оценивания матрицы параметров линейного дискретного динамического объекта управления // Проблемы управления и информатики. — 1995. — № 1. — С. 63 –77.

17. C h e r n o u s k o F . L . Ellipsoidal state estimation for dynamical systems // Nonlinear Analysis. —2005. — **63,** N 5–7. — P. 872–879.
18. К у н ц е в и ч В . М . Определение гарантированных оценок векторов состояния и параметров линейных динамических систем при ограниченных возмущениях // ДАН СССР. — 1986. — **288,** № 3. — С. 567–570.
19. K u n t s e v i c h V . M . Set-valued estimation of state and parameter vectors within adaptive control systems // Bounding approaches to system identification / Eds: M. Milanese, J.Norton, H.-Piet-Lahanier, E. Walter. — New York; London: Plenum Press, 1996. — P. 239–259.
20. Ш е л е с т о в А . Ю . , К у с с у л ь Н . Н . Робастное оценивание состояния узла Grid-системы методом нечетких эллипсоидов // Кибернетика и системный анализ. — 2008. — № 6. — С. 67–74.
21. J a i n R . The art of computer systems performance analysis. — New York: John Wiley and Sons, 1991. — 685 p.
22. L e v i n e D . , R a m s e y P . , S m i d t R . Applied statistics for engineers and scientists: Using Microsoft Excel & Minitab — New York: Prentice Hall, 2001. — 714 p.
23. M a r t i n i c h J . Production and operations management: an applied modern approach. — New York: John Wiley and Sons, 1996. — 944 p.
24. С е й д ж Э . П . , У а й т Ч . С . Оптимальное управление системами. — М.: Радио и связь, 1982. — 392 с.
25. Ш е л е с т о в А . Ю . Структурная и параметрическая идентификация моделей объектов управления на основе генетического алгоритма // Управляющие системы и машины. — 2007. — № 5. — С. 50–60.
26. E b e r h a r t R . , S i m p s o n P . , D o b b i n s R . Computational Intelligence PC Tools. — New York: AP Professional, 1996. — 464 p.

Поступила 30.06.2009