УДК 004.738.5

Д.Б. Кузнецов, Т.В. Шадрина

Пермский национальный исследовательский
политехнический университет, Пермь

РАЗРАБОТКА МАТЕМАТИЧЕСКОЙ МОДЕЛИ ДЕЦЕНТРАЛИЗОВАННОГО УПРАВЛЕНИЯ ПОРТАЛОМ САЙТОВ

В статье описывается разработка математической модели децентрализованного управления порталом сайтов при помощи сети Петри и распределенных автоматов с применением теории множеств. Также приводится обоснование необходимости разработки такой модели.

Ключевые слова: модель, управление, портал, сеть Петри, распределенный автомат.

D.B. Kuznetsov, T.V. Shadrina

Perm National Research Polytechnic University, Perm

DEVELOPMENT OF A MATHEMATICAL MODEL FOR WEB-PORTAL MANAGEMENT

The article describes the development of a mathematical model of decentralized web-portal management using a Petri net and distributed machine using set theory. The rationale for the need to develop such a model is also provided.

Keywords: model, management, web-portal, Petri net, distributed machine.

В современном мире сайт является важным инструментом для привлечения клиентов, предоставления информации о продуктах или услугах, установления контакта с потенциальными покупателями, продвижения бренда и увеличения узнаваемости компании. При этом у организаций, имеющих схожую структуру и выполняющих однотипные функции, появляются похожие сайты, что предопределяет дальнейшее развитие разработок в виде объединения их в портальные решения в рамках платформы [1].

По данным Global Digital с каждым годом увеличивается количество пользователей в сети Интернет, на 2023 г. этот показатель составил 64,4 % от мирового населения. С ростом популярности веб-технологий увеличивается и число посетителей веб-порталов, вследствие чего увеличивается нагрузка на серверы, обрабатывающие запросы пользователей к сайтам, что, в свою очередь, повышает вероятность их выхода из строя. Последствиями такой ситуации могут быть как экономические, так и репутационные потери. Следовательно, возникает необходимость обеспечения надежности веб-портала, о чем также свидетельствует ГОСТ 28806-90, где надежность является одной из основных характеристик качества программного продукта, которая включает в себя отказоустойчивость как подхарактеристику.

В современных системах отказоустойчивость обеспечивается за счет избыточности (резервирования) и балансировки нагрузки. И балансировка нагрузки, и резервное копирование – сложные процессы, которыми необходимо каким-то образом управлять для обеспечения нормальной работы портала сайтов. Следовательно, возникает необходимость в формулировании некоторого набора правил для управления порталом сайтов. Такой набор правил формируется при применении различных моделей и методов управления порталом сайтов.

Результаты анализа существующих моделей и методов управления порталом сайтов приведены в таблице.

Подходы к управлению порталом сайтов

Наименование	Преимущества	Недостатки
Системы управления контентом	Широко применяется, высокая степень изученности	Ключевой недостаток следует из определения, CMS управляют контентом, а не серверами
Стохастические модели прогнозирования	Возможность предвиденья поведения как системы в целом, так и отдельных ее компонентов	Не предоставляет инструкций к действиям по устранения или минимизации последствий отказа
Системы оркестрации контейнеров (иерархическое управление)	Автоматизированное управление, следовательно, отсутствие сбоев, вызванных ошибками оператора	Централизованное управление, иерархическая структура, следовательно, наличие единой точки отказа
Бессерверные вычисления (децентрализованное управление)	Позволяет не задумываться об инфраструктуре и ее параметрах (масштабируемости, высокой доступности и т. п.)	Отсутствие единого универсального решения

На основании таблицы можно сделать вывод, что для управления порталом сайтов необходимо использовать бессерверные вычисления, т.е. модели и методы децентрализованного управления, но, так как разработанные технологии, которые позволяют использовать такие вычисления, являются коммерческой тайной, а также вследствие отсутствия единого универсального решения среди таких подходов, возникает необходимость в разработке собственной математической модели. В результате анализа методов моделирования порталов сайтов было принято решение об использовании сети Петри и распределенных автоматов. С формальной точки зрения взаимодействие агентов в децентрализованной системе управления порталом сайтов определяется упорядоченным конечным набором длинны $n = 3$, каждый из элементов которого принадлежит некоторому конечному множеству M_i $(1 \leq i \leq n)$ (аналогично для $n \neq 3$):

$$c = \langle X, Z, S \rangle, \quad (1)$$

где X – множество входных значений для агента, Z – множество выходных значений для агента, S – множество агентов.

Множество входных значений – вектор бинарных значений длины $n = 6$:

$$X = \langle x_1, x_2, x_3, x_4, x_5, x_6 \rangle, \quad (2)$$

где x_1 – определяет отказал ли какой-нибудь из серверов, x_2 – можно ли считать данный сервер лучшим вариантом для размещения сайтов, x_3 – нужно ли выполнить резервное копирование, x_4 – совершил ли пользователь какое-нибудь действие (добавление или удаление блока или сайта), x_5 – произошла ли ошибка после выполнения действия пользователя, x_6 – нужно ли переместить часть сайтов с данного сервера на другой.

Множество выходных значений – это вектор бинарных значений длины $n = 4$:

$$Z = \langle z1, z2, z3, z4 \rangle, \quad (3)$$

где z1 – на данном сервере восстановлена работа сайтов с другого сервера, z2 – выполнено резервное копирование, z3 – исправлена пользовательская ошибка, z4 – выполнено перераспределение (часть сайтов с данного сервера размещены на другом).

Учитывая, что агенты в децентрализованной системе управления имеют одинаковый набор правил, можно провести моделирование только для одного из элементов множества S, с учетом уравнений (2) и (3). В таком случае графическое представление сети Петри будет выглядеть, как показано на рис. 1, где R – выполнение процесса восстановления работы сайтов, размещенных на вышедшем из строя сервере, B – процесс выполнение резервного копирования, C – процесс исправления ошибки, допущенной пользователем при внесении изменений на портале, D – процесс перераспределения сайтов по серверам, x – позиции для задания входного вектора, p – позиции, характеризующие промежуточные состояния, t – переходы между позициями, z – переходы между позициями, формирующие выходной вектор.

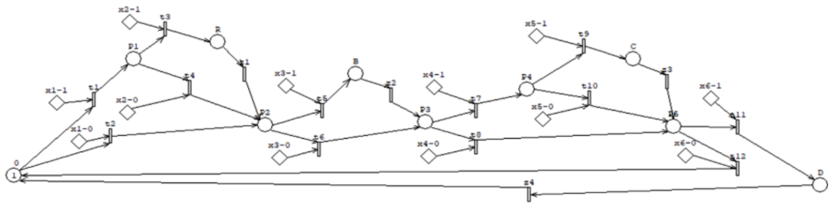


Рис. 1. Сервер портала как сеть Петри

Так как в основе анализа сетей Петри лежит анализ достижимости, проверим срабатывание выходных сигналов в зависимости от входного вектора при помощи имитационного моделирования. Для этого построим сеть Петри, представленную на рис. 1, в программе GRIN и запустим моделирование в автоматическом режиме.

На рис. 1 ромбами отмечены позиции, в которые будут устанавливаться фишки для задания конкретного значения входного вектора. Также фишка установлена в начальную позицию, отмеченную на рисунке нулем. Таким образом, задавая входной вектор при помощи расставления фишек в соответствующие позиции, на модели можно провести эксперементы.

Количество экспериментов можно посчитать по комбинаторной формуле, применимой для подсчета перестановок с повторениями:

$$\overline{A_n^k} = n^k = 2^6 = 64, \tag{4}$$

где n – количество значений переменной, k – количество переменных.

Результаты имитационного моделирования приведены на рис. 2.

На основании результатов, представленных на рис. 2, построенную модель можно считать применимой.

Для определения взаимодействия агентов в системе построим модель портала сайтов как распределенный автомат.

Автомат представляет собой систему:

$$A = \langle X, Z, Q, f, \phi, q_0 \rangle, \tag{5}$$

где $X = \{x_1, x_2, \dots x_n\}$ – множество входных сигналов, $Y = \{y_1, y_2, \dots y_m\}$ – множество выходных сигналов, $Q = \{q_1, q_2, \dots q_k\}$ – множество состояний автомата, $f(q, x)$ и $\phi(q, x)$ – функции переходов и выходов соответственно, q_0 – начальное состояние автомата [2].

С учетом выражения (5) зададим автомат следующим образом. Множество состояний автомата определим как:

$$Q = \{q_B, q_D, q_E, q_R\}, \tag{6}$$

где B – состояние при выполнении резервного копирования, D – состояние при перераспределении сайтов по серверам, E – состояние при возникновении ошибки, R – состояние при отказе одного из серверов.

№	x_1	x_2	x_3	x_4	x_5	x_6	z_1	z_2	z_3	z_4
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	1	0	0	0	1
3	0	0	0	0	1	0	0	0	0	0
4	0	0	0	0	1	1	0	0	0	1
5	0	0	0	1	0	0	0	0	0	0
6	0	0	0	1	0	1	0	0	0	1
7	0	0	0	1	1	0	0	0	1	0
8	0	0	0	1	1	1	0	0	1	1
9	0	0	1	0	0	0	0	1	0	0
10	0	0	1	0	0	1	0	1	0	1
11	0	0	1	0	1	0	0	1	0	0
12	0	0	1	0	1	1	0	1	0	1
13	0	0	1	1	0	0	0	1	0	0
14	0	0	1	1	0	1	0	1	0	1
15	0	0	1	1	1	0	0	1	1	0
16	0	0	1	1	1	1	0	1	1	1
17	0	1	0	0	0	0	0	0	0	0
18	0	1	0	0	0	1	0	0	0	1
19	0	1	0	0	1	0	0	0	0	0
20	0	1	0	0	1	1	0	0	0	1
21	0	1	0	1	0	0	0	0	0	0
22	0	1	0	1	0	1	0	0	0	1

№	x_1	x_2	x_3	x_4	x_5	x_6	z_1	z_2	z_3	z_4
23	0	1	0	1	1	0	0	0	1	0
24	0	1	0	1	1	1	0	0	1	1
25	0	1	1	0	0	0	0	1	0	0
26	0	1	1	0	0	1	0	1	0	1
27	0	1	1	0	1	0	0	1	0	0
28	0	1	1	0	1	1	0	1	0	1
29	0	1	1	1	0	0	0	1	0	0
30	0	1	1	1	0	1	0	1	0	1
31	0	1	1	1	1	0	0	1	1	0
32	0	1	1	1	1	1	0	1	1	1
33	1	0	0	0	0	0	0	0	0	0
34	1	0	0	0	0	1	0	0	0	1
35	1	0	0	0	1	0	0	0	0	0
36	1	0	0	0	1	1	0	0	0	1
37	1	0	0	1	0	0	0	0	0	0
38	1	0	0	1	0	1	0	0	0	1
39	1	0	0	1	1	0	0	0	1	0
40	1	0	0	1	1	1	0	0	1	1
41	1	0	1	0	0	0	0	1	0	0
42	1	0	1	0	0	1	0	1	0	1
43	1	0	1	0	1	0	0	1	0	0

№	x_1	x_2	x_3	x_4	x_5	x_6	z_1	z_2	z_3	z_4
44	1	0	1	0	1	1	0	1	0	1
45	1	0	1	1	0	0	0	1	0	0
46	1	0	1	1	0	1	0	1	0	1
47	1	0	1	1	1	0	0	1	1	0
48	1	0	1	1	1	1	0	1	1	1
49	1	1	0	0	0	0	1	0	0	0
50	1	1	0	0	0	1	1	0	0	1
51	1	1	0	0	1	0	1	0	0	0
52	1	1	0	0	1	1	1	0	0	1
53	1	1	0	1	0	0	1	0	0	0
54	1	1	0	1	0	1	1	0	0	1
55	1	1	0	1	1	0	1	0	1	0
56	1	1	0	1	1	1	1	0	1	1
57	1	1	1	0	0	0	1	0	0	0
58	1	1	1	0	0	1	1	1	0	1
59	1	1	1	0	1	0	1	1	0	0
60	1	1	1	0	1	1	1	1	0	1
61	1	1	1	1	0	0	1	1	0	0
62	1	1	1	1	0	1	1	1	0	1
63	1	1	1	1	1	0	1	1	1	0
64	1	1	1	1	1	1	1	1	1	1

Рис. 2. Результаты имитационного моделирования сети Петри

Множество выходных сигналов обозначим как:

$$Y = \{y_{OB}, y_{OD}, y_{OE}, y_{ER}, y_{BO}, y_{DO}, y_{EO}, y_{RO}\}. \tag{7}$$

Множество входных сигналов:

$$X = \{x_1, x_2, x_3, x_4, x_5\}, \tag{8}$$

где x_1 – любое внешнее воздействие, приводящее к ошибке, x_2 – ошибку нельзя исправить, и она критична, x_3 – команда на выполнение резервного копирования, x_4 – команда на перераспределение сайтов по серверам, x_5 – ошибка исправлена.

Начальное состояние q_0 – состояние, при котором обеспечивается нормальная работа портала. При построении распределенного автомата введем следующие допущения:

– каждый автомат имеет информацию о состоянии каждого другого автомата;

– два или более собственных переходов не могут произойти одновременно, т.е. переход в новое состояние инициируется единственным автоматом;

– при отказе сервер перестает принимать и получать идентификаторы состояний, т.е. «выпадает» из системы, следовательно, никак не влияет на поведение оставшихся.

С учетом этих допущений зададим распределенный асинхронный автомат для двух серверов графом, представленным на рис. 3.

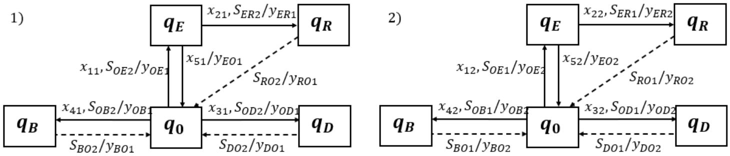
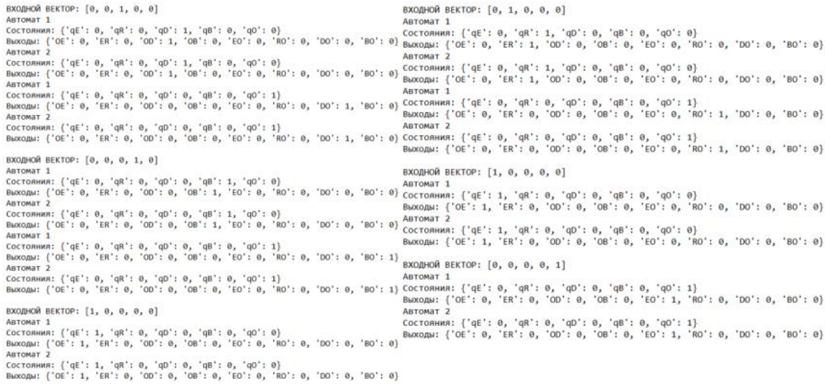


Рис. 3. Портал сайтов как распределенный автомат

На рис. 3 сплошной линией обозначены активные дуги (собственные переходы), а пунктирной – пассивные дуги (вынужденные переходы) петли, сохраняющие текущее состояние автомата, на рисунке опущены. Мультидуги показаны сплошной линией, помеченной разделенными запятой условием собственного перехода и обозначением нового состояния другого автомата, т.е. условием пассивного (вынужденного) перехода [3]. Идентификаторы вида S_{ijk} обозначают факт перехода k-го автомата (k = 1, 2) из i-го состояния (i = q_B, q_D, q_E, q_R) в j-е состояние (j = q_B, q_D, q_E, q_R). Определим портал сайтов как некоторое множество автоматов A длинны n>1. Тогда автомат $a_k \in A$, $1 \leq k \leq n$ и автомат $a_p \in A$, $1 \leq p \leq n$, причем $p \neq k$.



```
ВХОДНОЙ ВЕКТОР: [0, 0, 1, 0, 0]
Автомат 1
Состояния: {'qE': 0, 'qR': 0, 'qD': 1, 'qB': 0, 'qO': 0}
Выходы: {'OE': 0, 'ER': 0, 'OD': 1, 'OB': 0, 'EO': 0, 'RO': 0, 'DO': 0, 'BO': 0}
Автомат 2
Состояния: {'qE': 0, 'qR': 0, 'qD': 1, 'qB': 0, 'qO': 0}
Выходы: {'OE': 0, 'ER': 0, 'OD': 1, 'OB': 0, 'EO': 0, 'RO': 0, 'DO': 0, 'BO': 0}
Автомат 1
Состояния: {'qE': 0, 'qR': 0, 'qD': 0, 'qB': 0, 'qO': 1}
Выходы: {'OE': 0, 'ER': 0, 'OD': 0, 'OB': 0, 'EO': 0, 'RO': 0, 'DO': 1, 'BO': 0}
Автомат 2
Состояния: {'qE': 0, 'qR': 0, 'qD': 0, 'qB': 0, 'qO': 1}
Выходы: {'OE': 0, 'ER': 0, 'OD': 0, 'OB': 0, 'EO': 0, 'RO': 0, 'DO': 1, 'BO': 0}

ВХОДНОЙ ВЕКТОР: [0, 0, 0, 1, 0]
Автомат 1
Состояния: {'qE': 0, 'qR': 0, 'qD': 0, 'qB': 1, 'qO': 0}
Выходы: {'OE': 0, 'ER': 0, 'OD': 0, 'OB': 1, 'EO': 0, 'RO': 0, 'DO': 0, 'BO': 0}
Автомат 2
Состояния: {'qE': 0, 'qR': 0, 'qD': 0, 'qB': 1, 'qO': 0}
Выходы: {'OE': 0, 'ER': 0, 'OD': 0, 'OB': 1, 'EO': 0, 'RO': 0, 'DO': 0, 'BO': 0}
Автомат 1
Состояния: {'qE': 0, 'qR': 0, 'qD': 0, 'qB': 0, 'qO': 1}
Выходы: {'OE': 0, 'ER': 0, 'OD': 0, 'OB': 0, 'EO': 0, 'RO': 0, 'DO': 0, 'BO': 1}
Автомат 2
Состояния: {'qE': 0, 'qR': 0, 'qD': 0, 'qB': 0, 'qO': 1}
Выходы: {'OE': 0, 'ER': 0, 'OD': 0, 'OB': 0, 'EO': 0, 'RO': 0, 'DO': 0, 'BO': 1}

ВХОДНОЙ ВЕКТОР: [1, 0, 0, 0, 0]
Автомат 1
Состояния: {'qE': 1, 'qR': 0, 'qD': 0, 'qB': 0, 'qO': 0}
Выходы: {'OE': 1, 'ER': 0, 'OD': 0, 'OB': 0, 'EO': 0, 'RO': 0, 'DO': 0, 'BO': 0}
Автомат 2
Состояния: {'qE': 1, 'qR': 0, 'qD': 0, 'qB': 0, 'qO': 0}
Выходы: {'OE': 1, 'ER': 0, 'OD': 0, 'OB': 0, 'EO': 0, 'RO': 0, 'DO': 0, 'BO': 0}

ВХОДНОЙ ВЕКТОР: [0, 1, 0, 0, 0]
Автомат 1
Состояния: {'qE': 0, 'qR': 1, 'qD': 0, 'qB': 0, 'qO': 0}
Выходы: {'OE': 0, 'ER': 1, 'OD': 0, 'OB': 0, 'EO': 0, 'RO': 0, 'DO': 0, 'BO': 0}
Автомат 2
Состояния: {'qE': 0, 'qR': 1, 'qD': 0, 'qB': 0, 'qO': 0}
Выходы: {'OE': 0, 'ER': 1, 'OD': 0, 'OB': 0, 'EO': 0, 'RO': 0, 'DO': 0, 'BO': 0}
Автомат 1
Состояния: {'qE': 0, 'qR': 0, 'qD': 0, 'qB': 0, 'qO': 1}
Выходы: {'OE': 0, 'ER': 0, 'OD': 0, 'OB': 0, 'EO': 0, 'RO': 1, 'DO': 0, 'BO': 0}
Автомат 2
Состояния: {'qE': 0, 'qR': 0, 'qD': 0, 'qB': 0, 'qO': 1}
Выходы: {'OE': 0, 'ER': 0, 'OD': 0, 'OB': 0, 'EO': 0, 'RO': 1, 'DO': 0, 'BO': 0}

ВХОДНОЙ ВЕКТОР: [1, 0, 0, 0, 0]
Автомат 1
Состояния: {'qE': 1, 'qR': 0, 'qD': 0, 'qB': 0, 'qO': 0}
Выходы: {'OE': 1, 'ER': 0, 'OD': 0, 'OB': 0, 'EO': 0, 'RO': 0, 'DO': 0, 'BO': 0}
Автомат 2
Состояния: {'qE': 1, 'qR': 0, 'qD': 0, 'qB': 0, 'qO': 0}
Выходы: {'OE': 1, 'ER': 0, 'OD': 0, 'OB': 0, 'EO': 0, 'RO': 0, 'DO': 0, 'BO': 0}

ВХОДНОЙ ВЕКТОР: [0, 0, 0, 0, 1]
Автомат 1
Состояния: {'qE': 0, 'qR': 0, 'qD': 0, 'qB': 0, 'qO': 1}
Выходы: {'OE': 0, 'ER': 0, 'OD': 0, 'OB': 0, 'EO': 1, 'RO': 0, 'DO': 0, 'BO': 0}
Автомат 2
Состояния: {'qE': 0, 'qR': 0, 'qD': 0, 'qB': 0, 'qO': 1}
Выходы: {'OE': 0, 'ER': 0, 'OD': 0, 'OB': 0, 'EO': 1, 'RO': 0, 'DO': 0, 'BO': 0}
```

Рис. 4. Результаты имитационного моделирования распределенного автомата

Запишем логические функции переходов:

$$q_{Ek}(t+1) = q_{Ok}(t)(x_{1k}(t) \vee S_{OEp}(t)); \; q_{Rk}(t+1) = q_E(t)(x_{2k}(t) \vee S_{ERp}(t));$$

$$q_{Dk}(t+1) = q_{Ok}(t)(x_{3k}(t) \vee S_{ODp}(t)); \tag{9}$$

$$q_{Bk}(t+1) = q_{Ok}(t)(x_{4k}(t) \vee S_{OBp}(t));$$

$$q_O(t+1) = q_{Bk}(t)S_{BOp}(t) \vee q_{Dk}(t)S_{DOp}(t) \vee q_{Rk}(t)S_{ROp}(t) \vee q_{Ek}(t)x_{5k}(t).$$

Функции выходов:

$$y_{OE} = q_{Ok}(x_{1k} \vee S_{OEp});\ y_{ER} = q_{Ek}(x_{2k} \vee S_{ERp});\ y_{OD} = q_{Ok}(x_{3k} \vee S_{ODp});$$
$$y_{OB} = q_{Ok}(x_{4k} \vee S_{OBp}); \quad (10)$$
$$y_{EO} = q_{Ek}(x_{5k} \vee S_{EOp});\ y_{RO} = q_{Rk}S_{ROp};\ y_{DO} = q_{Dk}S_{DOp};\ y_{BO} = q_{Bk}S_{Bop}.$$

Для анализа достижимости состояний была построена имитационная модель на языке Python, результаты моделирования представлены на рис. 4.

На основании результатов, приведенных на рис. 4, можно утверждать о правильности функционирования разработанной модели.

Таким образом, в результате исследования разработана математическая модель децентрализованного управления порталом сайтов. Модель построена при помощи сети Петри и распределенного автомата Милли. Полученные функции переходов и выходов можно использовать для составления правил децентрализованного управления порталом сайтов, что является поводом для дальнейших исследований.

Библиографический список

1. Езова Н.С., Кузнецов Д.Б. Математическая модель платформы портального решения для построения системы сайтов // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2013. – № 8. – С. 23–30.

2. Викентьева О.Л., Соловьев А.Е., Файзрахманов Р.А. Дискретная математика: учеб. пособие. – Пермь: ПГТУ, 2009. – С. 112–120.

3. Кузнецов Б.П. Распределенные конечные автоматы // Приборы и системы. Управление, контроль, диагностика. – 2000. – № 2. – С. 9–12.

Сведения об авторах

Кузнецов Денис Борисович – старший преподаватель кафедры «Информационные технологии и автоматизированные системы» Пермского национального исследовательского политехнического университета, г. Пермь, e-mail: kdenisb@gmail.com

Шадрина Татьяна Владимировна – магистрант Пермского национального исследовательского политехнического университета, гр. РИС-23-1м, г. Пермь, e-mail: shanyat@mail.ru