

УДК 519.876.2

**Д.А. Петросов**, канд. техн. наук, доцент, Белгородский государственный аграрный университет им. В.Я. Горина (Россия, 308503, Белгород, пос. Майский, ул. Вавилова, 1)  
(e-mail: scorpionss2002@mail.ru)

**Н.В. Петросова**, преподаватель, Белгородский государственный аграрный университет им. В.Я. Горина (Россия, 308503, Белгород, пос. Майский, ул. Вавилова, 1)  
(e-mail: petrosova.nat@mail.ru)

**А.Г. Бажанов**, канд. техн. наук, Белгородский государственный аграрный университет им. В.Я. Горина (Россия, 308503, Белгород, пос. Майский, ул. Вавилова, 1)  
(e-mail: bazhanov.ag@bstu.ru)

**О.И. Бажанова**, канд. техн. наук, доцент, Белгородский государственный аграрный университет им. В.Я. Горина (Россия, 308503, Белгород, пос. Майский, ул. Вавилова, 1)  
(e-mail: bazhanova.oi@bstu.ru)

### **ИМИТАЦИОННАЯ МОДЕЛЬ ГРАФИЧЕСКОГО МУЛЬТИПРОЦЕССОРА НА ОСНОВЕ ТЕОРИИ СЕТЕЙ ПЕТРИ**

*В работе рассматривается проблема создания имитационных моделей графического мультипроцессора для проведения вычислительных экспериментов с целью определения эффективности применения параллельных вычислений на основе технологии GPGPU (General-purpose computing for graphics processing units, неспециализированные вычисления на графических процессорах) в задачах структурно-параметрического синтеза больших дискретных систем на основе эволюционных процедур. В качестве математического аппарата предлагается использование теории сетей Петри, которая обладает свойством параллелизма и позволяет описывать дискретные процессы, протекающие как в генетическом алгоритме, так и в самой вычислительной системе. Разработка имитационной модели выполняется на основе архитектуры памяти графического мультипроцессорного модуля с учетом специфики ее работы, связанной с возможностью чтения, записи и передачи данных. Кроме того, рассмотрена особенность работы арифметико-логических устройств, которые способны выполнять одновременно одну команду над множеством данных. При построении модели учтена особенность графических мультипроцессоров, которая позволяет получить больший эффект от применения параллельных вычислений избегая срабатывания блоков ветвлений и управления, которые замедляют работу мультипроцессора (так как их количество меньше, чем количество вычислителей), образуя тем самым «узкие» места. Предложенная имитационная модель мультипроцессорного блока на основе выбранного инструментального средства, реализованная с помощью специализированного программного средства имитационного моделирования на базе теории сетей Петри – PIPE 5. Данное программное средство распространяется бесплатно и обладает широким спектром инструментальных и аналитических средств, что в значительной мере упрощает как процесс моделирования, так и процесс анализа полученных моделей. Полученная в результате исследования модель даст возможность оценить эффективность применения параллельных вычислений на основе технологии GPGPU при решении задачи повышения быстродействия интеллектуальных информационных систем поддержки принятия решений на базе адаптированных к предметной области генетических алгоритмов.*

**Ключевые слова:** имитационное моделирование; интеллектуальные системы; эволюционные методы; генетические алгоритмы; параллельные вычисления; графический мультипроцессор.

**DOI:** 10.21869/2223-1560-2018-22-5-127-135

**Ссылка для цитирования:** Имитационная модель графического мультипроцессора на основе теории сетей Петри / Д.А. Петросов, Н.В. Петросова, А.Г. Бажанов, О.И. Бажанова // Известия Юго-Западного государственного университета. 2018. Т. 22, № 5(80). С. 127-135.

\*\*\*

#### **Введение**

В настоящее время существует проблема повышения быстродействия ин-

теллектуальных информационных систем поддержки принятия решений. Для решения данной проблемы предлагаются

различные подходы, которые основываются как на повышении вычислительных возможностей аппаратной части, так и со стороны программной реализации. В свою очередь, программная реализация может использовать новые математические аппараты, модели и методы алгоритмизации, которые направлены на повышение скорости обработки данных для синтеза решений, а также возможно применение новых технологий при программировании систем искусственного интеллекта. Одним из таких подходов является технология GPGPU (General-purpose computing for graphics processing units, неспециализированные вычисления на графических процессорах) [1].

Применение данной технологии может позволить в значительной мере повысить быстродействие систем поддержки принятия решений, которые базируются на эволюционных процедурах поиска решений [2, 3], таких как генетические алгоритмы и искусственные нейронные сети. Как в генетических алгоритмах [4], так и в нейронных сетях [5] заложено свойство параллелизма, которое может быть использовано совместно с технологией GPGPU. В этом случае возникает вопрос в описании данных методов с помощью единого математического аппарата, который позволит создать имитационную модель для проведения вычислительных экспериментов с целью выявления эффективности применения параллельных вычислений в области структурно-параметрического синтеза больших дискретных систем с заданным поведением. В качестве основного математического аппарата предлагается использовать

теорию сетей Петри [6]. В работе [2] была представлена имитационная модель генетического алгоритма на основе вложенных сетей Петри. Данная модель позволяла проводить процедуру синтеза моделей сетей Петри, описывающих поведение больших дискретных систем в соответствии с заданным поведением (способностью обрабатывать заданный входной вектор в эталонный выходной, проходя через ряд заданных состояний). В том числе данный подход можно применять для построения моделей дискретных процессов при создании управляющих автоматов [7, 8]. Для получения данных об эффективности применения технологии GPGPU целесообразно разработать имитационную модель мультипроцессора графического вычислителя, с использованием того же математического аппарата, и оценить повышение быстродействия.

Технология GPGPU позволяет распределить вычисления на множество процессоров, но существует проблема с работой памяти устройства, а также наличия «узких» мест [9], связанных с обработкой ветвлений и управления. Данная проблема связана с тем, что блоков управления и ветвления в мультипроцессоре меньше, чем процессоров, в которых, для повышения быстродействия, требуется обрабатывать одну команду на различных данных (архитектура памяти мультипроцессора показана на рисунке 1).

Особенности представленной архитектуры следует учесть при создании имитационной модели на основе теории сетей Петри.

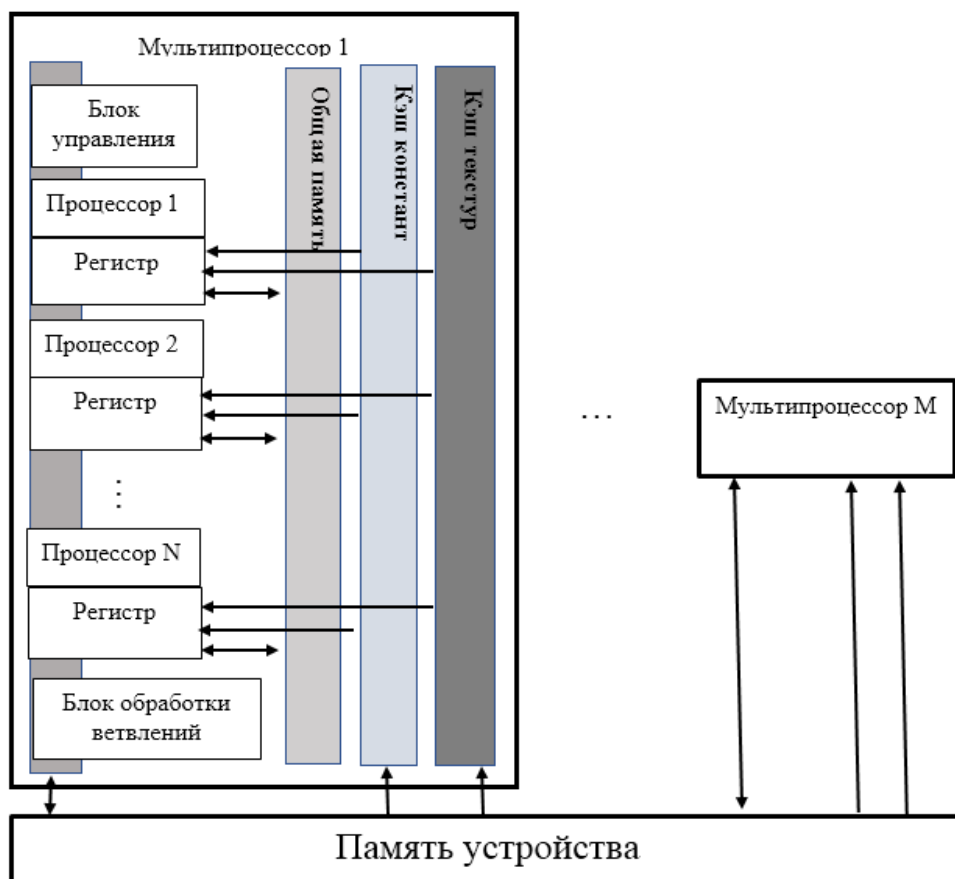


Рис. 1. Архитектура памяти мультипроцессора в графическом вычислителе

### Постановка задачи

В рамках проводимого исследования требуется построить имитационную модель мультипроцессора графического вычислителя  $M_{mult}$ , которая сможет отразить специфику обработки данных. Для этого следует учесть:

- работу всех моделей памяти вычислительного устройства;
- количество вычислителей и его регистры;
- взаимосвязь вычислителей и модулей памяти с учетом записи и чтения данных;
- взаимосвязь структуры компонентов графического мультипроцессора;
- работу «узких» мест мультипроцессора: блока управления и блока ветвления;

– запись начальных данных и параметров функционирования генетических алгоритмов.

### Методы, предложенные в статье

Таким образом разрабатываемую модель мультипроцессора в графическом вычислителе можно представить в следующем виде:

$$M_{mult} = (P, K_{str}, K_{const}, Rem_{all}, Rem_{dev}, Reg_{proc}, B_{contr}, B_{if}, C),$$

где  $M_{mult}$  – модель мультипроцессора;

$P$  – модель процессора и его регистра;

$K_{str}$  – кэш текстур;

$K_{const}$  – кэш констант;

$Rem_{all}$  – общая память;

$Rem_{dev}$  – память устройства;

$Reg_{proc}$  – регистры процессоров;

$B_{contr}$  – блок управления;

$B_{if}$  – блок ветвлений;

$C$  – связи между элементами.

Для каждого компонента модели требуется разработать модели на основе сетей Петри, тогда представленный кортеж примет следующий вид:

$M_{mult} = (PN_{proc}, PN_{str}, PN_{const}, PN_{rem_{all}}, PN_{rem_{dev}}, PN_{reg_{proc}}, PN_{contr}, PN_{if}, T, F)$ ,

где  $PN_{proc}$  – множество  $\{PN_{proc_i}\}_{i=1}^{M_r}$  моделей процессоров с регистром на основе сетей Петри;

$PN_{str}$  – модель кэша текстур на основе сетей Петри (в качестве хранения данных используются позиции, количество позиций соответствует размеру блока памяти);

$PN_{const}$  – модель кэша констант на основе сетей Петри (в качестве хранения данных используются позиции, количество позиций соответствует размеру блока памяти);

$PN_{rem_{all}}$  – модель общей памяти мультипроцессоров на основе сетей Петри (в качестве хранения данных используются позиции, количество позиций соответствует размеру блока памяти);

$PN_{rem_{dev}}$  – модель памяти устройств на основе сетей Петри (в качестве хранения данных используются позиции, количество позиций соответствует размеру блока памяти);

$PN_{reg_{proc}}$  – модель регистров процессоров на основе сетей Петри (в качестве хранения данных используются позиции, количество позиций соответствует размеру блока памяти);

$PN_{contr}$  – модель блока управления на основе сетей Петри;

$PN_{if}$  – модель блока разветвления на основе сетей Петри;

множество переходов  $T$  и соответствий  $F$  определяет структуру модели  $M_{mult}$ .

Модель процессора из множества  $PN_{proc}$  на основе сети Петри можно представить в следующем виде (рис. 2)  $PN_{proc_i} = \langle P, T, F, M_0 \rangle$ .

Позиции  $P = \{P_i\}_{i=1}^m$ , где  $m$  – количество ячеек памяти в регистре (на примере отображено четыре ячейки памяти  $P_0, P_1, P_2, P_3$ ), которые служат для хранения данных, над которыми следует провести вычисление.

Переход  $T_0$  моделирует выполнение операций над данными (работу вычислителя).

Для обработки ветвлений и работы блока управления целесообразно применить переходы с таймингом (рис. 3), таким образом получится учесть временную задержку, которая возникает в «узких» местах.

Переход  $T_0$  обладает таймингом, который выставляется на время срабатывания, данные для управления передаются из перехода  $T_1$  (для того, чтобы отделить данные, которые требуют использования данного блока, можно использовать или отдельные ячейки памяти или разделить их с помощью цвета, применяя цветные сети Петри) в позиции, соединённые с переходом  $T_0$  по входу, после обработки помещаются в позиции для дальнейшего участия в вычислениях.

Так как в соответствии с архитектурой мультипроцессора общая память имеет возможность записи и чтения из регистров процессора, это должно быть отражено в разрабатываемой модели, при этом данные для вычисления должны передаваться в свободные регистры. Для этого предлагается использовать ингибиторные дуги, которые позволяют сработать переходу при условии, что во входящей позиции нет меток (ячейки памяти свободны).

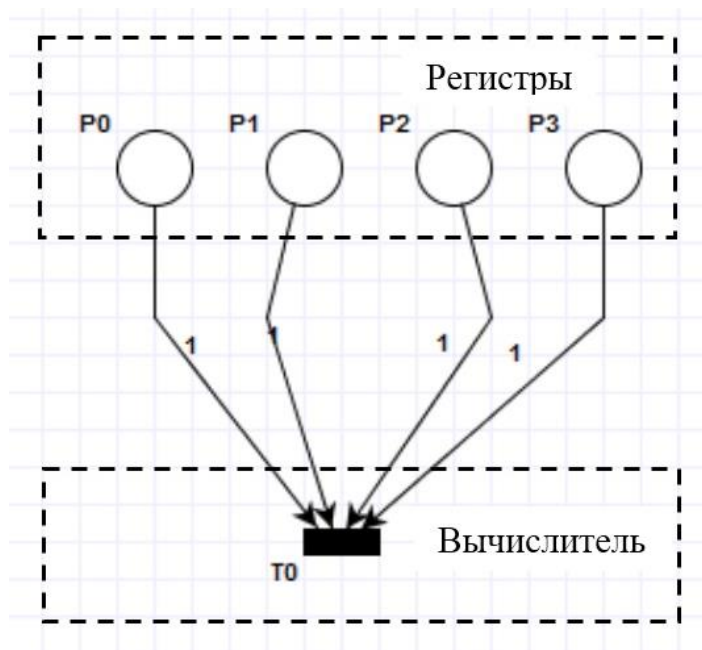


Рис. 2. Пример модели процессоров с регистром на основе сетей Петри

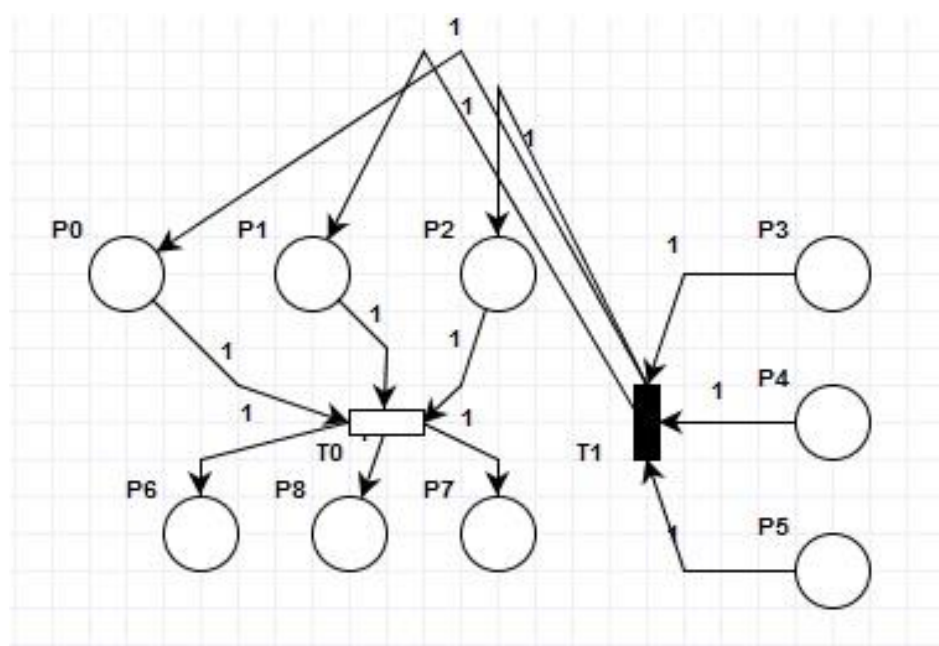


Рис. 3. Пример модели блока управления на сетях Петри

Регистры и кэши мультимикропроцессора можно моделировать с помощью позиций, соединённых с исполнительными модулями с помощью переходов.

Для моделирования модели мультимикропроцессорного модуля использовался

программный продукт PIPE 4.5. Данное программное средство поддерживает все расширения, которые требуются для построения имитационной модели. Полученная модель представлена на рисунке 4.

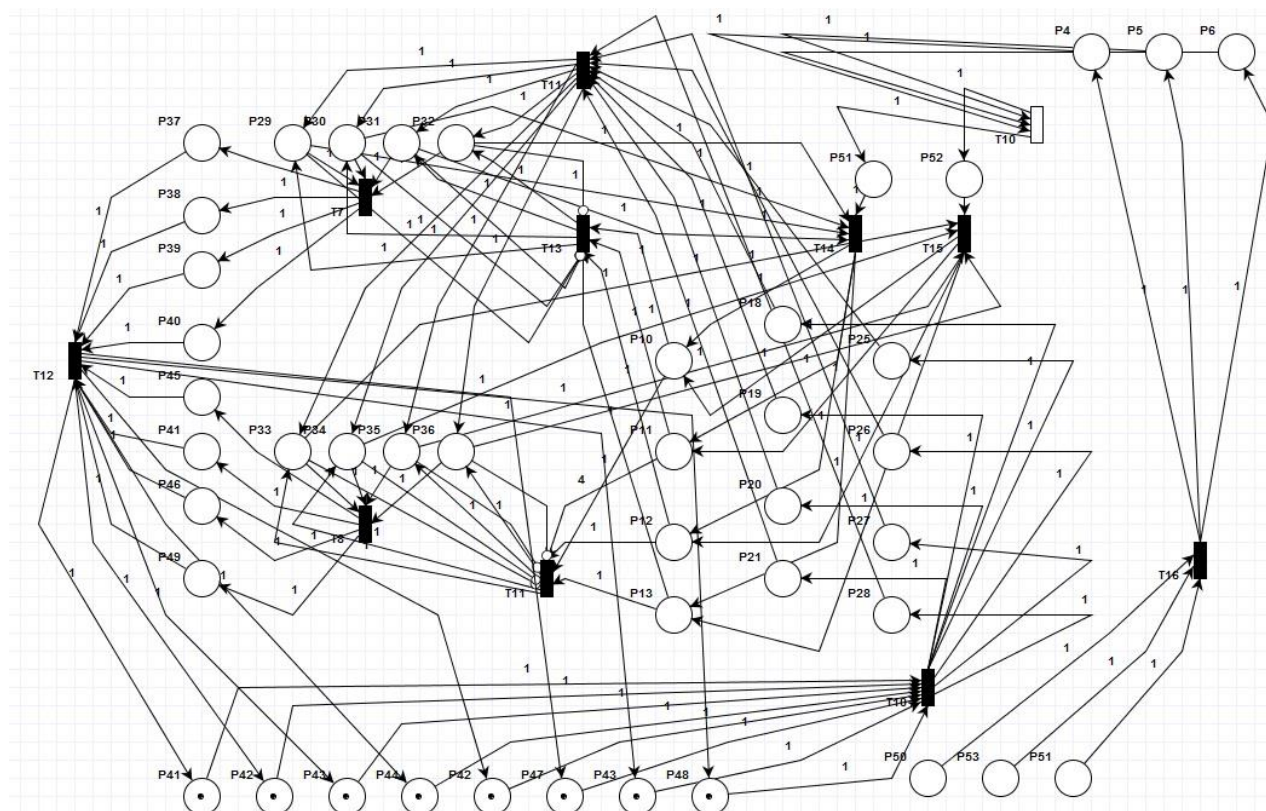


Рис. 4. Имитационная модель графического мультипроцессора на основе сетей Петри

### Вычислительный эксперимент

В ходе проверки адекватности полученной модели был проведен ряд вычислительных экспериментов по корректности обработки данных имитационной моделью мультипроцессора:

- без использования блока ветвления и блока управления (учет возможности полного распараллеливания процессов);
- с учетом использования блока ветвлений;
- с учетом использования блока управления;
- с учетом использования блока ветвлений и блока управления.

Адекватность полученных в результате работы модели данных была проверена с помощью деревьев достижимых маркировок.

### Обсуждение результатов и заключение

Предложенная модель иллюстрирует работу мультипроцессора при обработке популяций генетического алгоритма. В  $PN_{str}$  (кэше текстур) используется в  $M_{mult}$  для кэширования исходных данных задачи – популяция генетического алгоритма.  $PN_{const}$  (кэш констант) предлагается использовать для хранения неизменяемых в ходе вычисления данных для всех мультипроцессоров, для генетического алгоритма это могут быть параметры функционирования операторов, размер популяции, эталонные вектора и т.д. В  $PN_{rem\_all}$  (общей памяти мультипроцессоров) хранятся промежуточные значения и доступ к данному виду памяти имеет исключительно мультипроцессор, в котором проводятся вычисления. Данный вид памяти устройства доступен всем мульт

типпроцессорам и является внешней, следовательно, может использоваться для обмена данными между мультипроцессорами.

Предложенная модель будет использована для проведения имитационного моделирования работы генетического алгоритма при решении задачи структурно-параметрического синтеза больших дискретных систем с заданным поведением с применением средств параллельных вычислений в сравнении с однопроцессорной вычислительной системой. Что даст возможность сравнить полученные данные с расчетами по закону Амдала [10], которые показали увеличение быстродействия за счет увеличения количества вычислителей более чем в три раза. Также целесообразно проведение вычислительных экспериментов с применением программной реализации на основе программно-аппаратной архитектуры параллельных вычислений CUDA (Compute Unified Device Architecture) с использованием видеокарт компании Nvidia. Такого рода исследование позволит сравнить полученные в результате работы предложенной модели результаты не только с расчетами по закону Амдала, но и с реальными значениями.

#### **Благодарности:**

*Работа выполнена при поддержке гранта РФФИ: № 18-07-00634-А*

#### **Список литературы**

1. Al-Mouhamed M., Khan A.H. Exploration of automatic optimisation for CUDA programming // International Journal of Parallel, Emergent and Distributed Systems. 2015. Vol. 30. Issue 4. Pp. 309 – 324. doi: 10.1080/17445760.2014.953158.
2. Петросов Д.А. Адаптация генетического алгоритма при моделировании вычислительной техники с изменяющейся структурой и набором компонентов на основе сетей Петри // Вопросы современной науки и практики. Университет им. В.И. Вернадского. 2009. № 6 (20). С. 151 – 160.
3. Петросов Д.А. Применение параллельных вычислений в интеллектуальных системах управления // Информационно-аналитические системы и технологии: материалы V международной конференции. 2018. С. 24 – 29.
4. Hart W. E., Baden S., Belew R. K., Kohn S. Analysis of the Numerical Effects of Parallelism on a Parallel Genetic Algorithm. In IEEE (ed.): CD-ROM IPPS97. 1997. 8p.
5. Al-Dabass D., Vindlacheruvu P., Evans D.J. Parallelism in neural nets // Parallel Algorithms and Applications. 1997. Is. 3-4. Pp. 169 – 185.
6. Lomazova I. A. Resource Equivalences in Petri Nets, in: Application and Theory of Petri Nets and Concurrency. 38th International Conference, PETRI NETS 2017, Zaragoza, Spain, June 25–30, 2017, Proceedings/ Ed. By W. van der Aalst E. Best. Vol. 10258: Lecture Notes in Computer Science. Switzerland: Springer, 2017. Pp. 19 – 34.
7. Подходы к построению дискретных моделей непрерывных технологических процессов для синтеза управляющих автоматов / В.З. Магергут, В.А. Игнатенко, А.Г., Бажанов В.Г. Шаптала // Вестник БГТУ им. В.Г. Шухова. 2013. № 2. С. 100 – 102.
8. Магергут В.З., Рубанов В.Г., Чуев А.С. Формализация и анализ дискретных организационно-технологических систем со структурированными агентами на ин-



дикаторных сетях: монография. Белгород: БГТУ, 2016. 149 с.

9. Басавин Д.А., Петросов Д.А., Игнатенко В.А. Применение технологии GPGPU в задачах создания интеллектуальных систем поддержки принятия решений // Высокие интеллектуальные технологии в науке и образовании: материалы IV Меж-

дународной научно-практической конференции. СПб., 2017. С. 63 – 65.

10. Amdahl, Gene M. Validity of the Single Processor Approach to Achieving Large-Scale Computing Capabilities // AFIPS Conference Proceedings. Pp. 483 – 485. doi:10.1145/1465482.1465560.

*Поступила в редакцию 30.07.18*

UDC 519.876.2

**D.A. Petrosov**, Candidate of Engineering Sciences, Associate Professor, Belgorod State Agricultural University named after V.Ya. Gorin (Russia, 308503, Belgorod, May Village, Vavilova Str., 1) (e-mail: scorpionss2002@mail.ru)

**N.V. Petrosova**, Lecturer, Belgorod State Agricultural University named after V.Ya. Gorin (Russia, 308503, Belgorod, May Village, Vavilova Str., 1) (e-mail: petrosova.nat@mail.ru)

**A.G. Bazhanov**, Candidate of Engineering Sciences, Associate Professor, Belgorod State Agricultural University named after V.Ya. Gorin (Russia, 308503, Belgorod, May Village, Vavilova Str., 1) (e-mail: bazhanov.ag@bstu.ru)

**O.I. Bazhanova**, Candidate of Engineering Sciences, Associate Professor, Belgorod State Agricultural University named after V.Ya. Gorin (Russia, 308503, Belgorod, May Village, Vavilova Str., 1) (e-mail: bazhanova.oi@bstu.ru)

## **SIMULATION MODEL OF GRAPHIC MULTIPROCESSOR BASED ON THE PETRI NETS THEORY**

*This work addresses the problem of creating the simulation models of a graphical multiprocessor for carrying out computational experiments to determine the efficiency of using the parallel computing based on GPGPU (General-purpose computing for graphics processing units, non-specialized computing on graphics processors) in tasks of structural-parametric synthesis of big discrete systems based on evolutionary procedures. We propose to use the Petri nets theory as a mathematical tool. It has the parallelism property and allows describing discrete processes occurring both in the genetic algorithm and in the computing system itself. The development of a simulation model is carried out on the basis of the graphic multiprocessor module memory architecture taking into account the specifics of its work related to the ability to read, write and transmit data. In addition, we describe the feature of the arithmetic logic devices work, which are able to simultaneously execute one command over a set of data. When building the model we take into account a feature of graphic multiprocessors which allows to get a greater effect from using the parallel computing avoiding the branching and control blocks operation that slow down the multiprocessor (since their number is less than the calculators number), thereby forming the “narrow” places. The proposed simulation model of a multiprocessor unit based on the selected tool is implemented using the specialized software for simulation based on the Petri nets theory – PIPE 5. This software is distributed free of charge and has a wide range of instrumental and analytical tools, which greatly simplifies both the modeling process and the process of analyzing the obtained models. The resulting model will provide an opportunity to evaluate the efficiency of using parallel computing based on GPGPU technology in solving the task of improving the performance of intelligent information decision support systems based on genetic algorithms adapted to the subject area.*



**Key words:** simulation; intelligent systems; evolutionary methods; genetic algorithms; parallel computing; graphic multiprocessor.

**DOI:** 10.21869/2223-1560-2018-22-5-127-135

**For citation:** Petrosov D.A., Petrosova N.V., Bazhanov A.G., Bazhanova O.I. Simulation Model of Graphic Multiprocessor Based on the Petri Nets Theory. Proceedings of the Southwest State University, 2018, vol. 22, no. 5(80), pp. 127-135 (in Russ.).

\*\*\*

## Reference

1. Al-Mouhamed M., Khan A.H. Exploration of automatic optimisation for CUDA programming. *International Journal of Parallel, Emergent and Distributed Systems*, 2015, vol. 30, is. 4, pp. 309 – 324.
2. Petrosov D.A. Adaptacija geneti-cheskogo algoritma pri modelirovanii vychislitel'noj tehniki s izmenjajushhejsja strukturoj i naborom komponentov na os-nove setej Petri. *Voprosy sovremennoj nauki i praktiki. Universitet im. V.I. Vernadskogo*, 2009, no. 6 (20), pp. 151 – 160.
3. Petrosov D.A. Primenenie paral-lel'nyh vychislenij v intel-lektual'nyh siste-mah upravlenija. *Informacionno-analitiche-skie sistemy i tehnologii. Materialy V mezh-dunarodnoj konferencii*, 2018, pp. 24 – 29.
4. Hart W. E., Baden S., Belew R. K., Kohn S. Analysis of the Numerical Effects of Parallelism on a Parallel Genetic Algo-rithm. In IEEE (ed.): CD-ROM IPPS97. 1997, 8 p.
5. Al-Dabass D., Vindlacheruvu P., Ev-ans D.J. Parallelism in neural nets. *Parallel Algorithms and Applications*, 1997, is. 3-4, pp. 169 – 185.
6. Lomazova I. A. Resource Equiva-lences in Petri Nets, in: Application and Theory of Petri Nets and Concurrency. 38th International Conference, PETRI NETS 2017, Zaragoza, Spain, June 25–30, 2017, Proceedings. Ed. By W. van der Aalst E. Best. Vol. 10258: Lecture Notes in Com-put-er Science. Switzerland: Springer, 2017, pp. 19 – 34.
7. Magergut V.Z., Ignatenko V.A., Ba-zhanov A.G., Shaptala V.G. Podhody k postroeniju diskretnyh modelej nepre-ryvnyh tehnologicheskikh processov dlja sin-teza upravljajushhih avtomatov. *Vestnik BGTU im. V.G. Shuhova*, 2013, no. 2, pp. 100 – 102.
8. Magergut V.Z., Rubanov V.G., Chuev A.S. Formalizacija i analiz diskretnyh organizacionno-tehnologicheskikh sistem so strukturirovannymi agentami na in-dikatornyh setjah. Belgorod, 2016, 149 p.
9. Basavin D.A., Petrosov D.A., Ig-natenko V.A. Primenenie tehnologii GPGPU v zadachah sozdaniya intellektual'nyh sistem podderzhki prinjatija reshenij. *Vysokie intel-lektual'nye tehnologii v nauke i obrazovanii. Materialy IV Mezhdunarodnoj nauchno-prakticheskoy konferencii*. Saint-Petersburg, 2017, pp. 63 – 65.
10. Amdahl Gene M. Validity of the Single Processor Approach to Achieving Large-Scale Computing Capabilities. *AFIPS Conference Proceedings*, pp. 483 – 485.