

УДК 004.8, 519.85

ВЫЧИСЛИТЕЛЬНЫЙ КЛАСТЕР ДЛЯ РЕШЕНИЯ ЗАДАЧ ОБЕСПЕЧЕНИЯ УСТОЙЧИВОСТИ ФУНКЦИОНИРОВАНИЯ РАСПРЕДЕЛЕННЫХ ИНФОРМАЦИОННЫХ СИСТЕМ

А. Н. Ивутин, к.т.н., доцент, заведующий кафедрой ВТ, ТулГУ; alexey.ivutin@gmail.com
Д. О. Есиков, аспирант, ТулГУ; mcgeen4@gmail.com

Предложено для решения задач обеспечения устойчивости функционирования распределенных информационных систем применять островной генетический алгоритм. Приведено описание предложенного островного генетического алгоритма для решения задач обеспечения устойчивости функционирования распределенных информационных систем. Для получения рационального (квази-оптимального) решения в условиях жестких временных ограничений предложена к использованию кластерная вычислительная система. Определены структура кластера, состав и параметры программно-аппаратных средств. Приведены таблицы с характеристиками вычислительных модулей, коммутаторов и средств коммуникации. Предложены инструкции для настройки каждого вычислительного модуля в кластерной вычислительной системе, а также для запуска программ на каждом модуле.

Ключевые слова: островной генетический алгоритм, кластер, распределенная информационная система, структура кластера, параметры кластера, многоагентный алгоритм, вычислительные модули, кластерная вычислительная система.

DOI: 10.21667/1995-4565-2016-57-3-63-67

Введение

Одной из сложных и остро стоящих задач в области обеспечения бесперебойной обработки данных является задача обеспечения устойчивости функционирования распределенных информационных систем.

Для обеспечения устойчивости функционирования распределенных информационных систем предлагается решить следующие задачи [1]:

1. Оптимизация распределения элементов программного обеспечения функциональных задач по узлам сети.

2. Оптимизация распределения информационных ресурсов по центрам хранения и обработки данных.

3. Определения рационального уровня расходов на формирование комплекса средств хранения данных в центрах хранения и обработки информации.

4. Оптимизация состава технических средств системы хранения и обработки данных.

5. Оптимизация распределения резерва информационных ресурсов по центрам хранения и обработки данных.

Данные задачи относятся к классу задач комбинаторной оптимизации [1], а именно целочисленного дискретного программирования с булевыми переменными [1, 2].

Теоретическая часть

Для получения точного решения существует достаточно большое количество методов дискретной оптимизации [2, 3]. Однако с увеличением размерности задач время решения точными методами возрастает экспоненциально.

В условиях жестких временных ограничений целесообразным является применение для решения указанных задач приближенных методов, имеющих алгоритмы, обладающие невысокой вычислительной сложностью и хорошо поддающиеся распараллеливанию. При этом необходимо обеспечить управление точностью получаемого решения. Это позволит обеспечить в распределенных информационных системах оперативное решение задач управления процессами хранения и обработки данных.

Этим критериям отвечают многоагентные популяционные алгоритмы [2]. Параллельные популяционные алгоритмы, построенные на основе островной модели (islandmodel) параллелизма, в некоторых публикациях называют миграционными алгоритмами или алгоритмами коммутирующей мультипопуляции. Суть алгоритмов этого класса состоит в следующем. Создается мультипопуляция $S = S_1 \cup S_2 \dots \cup S_{|P|}$, состоящая из числа субпопуляций (островов) S_i ,

равного числу используемых рабочих процессоров $|P|$ параллельной ЭВМ (вычислительных модулей кластерной системы). Каждый остров обрабатывает свой процессор системы. Обмен данными между островами выполняется после каждой независимых итераций (сезонов) в соответствии с используемой топологией соседства островов [6].

Островная модель использует статическую балансировку загрузки параллельной ЭВМ (кластерной системы) и обеспечивает, как правило, высокую производительность при выполнении следующих условий: вычислительная система является гомогенной; размеры субпопуляций одинаковы и достаточно велики; условием завершения итераций каждой из субпопуляций S_i является достижение одного и того же числа итераций.

Известны следующие рекомендации относительно предпочтительности использования островной модели [4]. Если пропускная способность коммуникационной сети ЭВМ является невысокой и мала вычислительная сложность фитнес-функции, то целесообразно выбирать островную модель. Также островную модель лучше использовать в случае популяции очень большого размера.

Варианты островной модели различаются топологиями связей островов, стратегиями миграции, режимами обмена агентами.

Метод получения рационального (квазиоптимального) решения в условиях жесткого ограничения времени решения, основанный на использовании островного генетического алгоритма, обеспечит за счет обоснованного выбора параметров алгоритма, в отличие от существующих приближенных методов, гарантированное получение рационального (квазиоптимального) решения (серии рациональных решений) за время, не превышающее заданное.

Сущность островного генетического алгоритма заключается в одновременной и независимой друг от друга работе на каждом вычислительном модуле кластера (процессоре параллельной ЭВМ) нескольких простых генетических алгоритмов (ГА) со своими популяциями. Каждый из них, вследствие случайной начальной инициализации исходной популяции, сходится к своему локальному экстремуму (острову). С определенной периодичностью (через определенное число поколений) по определенной схеме ГА обмениваются заданным числом лучших особей (осуществляется миграция особей), после чего процесс решения продолжается. Совокупность шагов алгоритма, завершающаяся обменом (миграцией особей) лучшими особями

между островами (вычислительными модулями), составляет одну итерацию островного генетического алгоритма (ОГА). Подобная схема позволяет обеспечить выход простых ГА в каждом из островов из области локального экстремума, большее разнообразие особей в популяциях и с большей вероятностью получить лучшее решение (в том числе достичь глобального экстремума). В качестве схемы миграции особей используются миграция с топологией полной сети, миграция с топологией кольца и их модификации [5,6].

Введем следующие обозначения: N^{Lnd} — число островов; N^{Step} — число итераций ОГА.

Островной ГА с миграцией особей с топологией кольца состоит из следующих шагов:

1. Для каждого k -го острова ($k=1,2,\dots,N^{Lnd}$) выполнить:

— инициализацию исходной популяции путем заполнения ее особями, сгенерированными случайным образом. В исходную популяцию не включаются нежизнеспособные особи. Повторное включение уже имеющейся особи не допускается;

— формирование элитной популяции объемом L^{el} особей.

2. $i = 0$.

3. Для каждого острова ($k=1,2,\dots,N^{Lnd}$) осуществляется последовательная эволюция поколений в количестве N^{gen} (выполнение простого ГА с заданной схемой репродукции).

4. Если $N^{Lnd} > 1$, для каждого k -го острова ($k=1,2,\dots,N^{Lnd}-1$) осуществляется перенос m ($m=L^{el}/2$) лучших особей из элитной популяции в исходную популяцию j -го ($j=k+1$) острова путем замещения худших особей. Для случая $k=N^{Lnd}$ перенос лучших особей осуществляется в 1-й остров.

5. $i = i + 1$.

6. Если $i < N^{Step}$, перейти п.3. В противном случае к п.7.

7. Сформировать результирующую популяцию из особей, составляющих элитные популяции каждого из островов.

8. Из результирующей популяции выбрать лучшую особь – результат решения.

9. Конец.

Выбор схемы репродукции оказывает существенное влияние на качество получаемого результата, однако эффективность схемы репродукции существенно зависит от результатов начальной инициализации исходной популяции. Неверный выбор схемы репродукции обычно приводит к существенному ухудшению качества получаемого решения. В связи с тем, что инициализация исходной популяции осуществляется

случайным образом, осуществить заранее эффективный выбор схемы репродукции не представляется возможным. Предлагается использовать адаптивную схему репродукции, для чего в каждую хромосому необходимо включить числовые метки, используемые для получения схемы отбора и варианта кроссинговера [6].

На каждом шаге алгоритма (при формировании нового поколения особей) определяется количество особей в родительской популяции, сформированных с использованием каждой схемы отбора и каждого вида кроссинговера, и рассчитывается частота (вероятность) использования схем отбора и вариантов кроссинговера. При начальной инициализации исходной популяции для всех хромосом данным меткам присваивается значение -1, а частоты применения схем отбора и вариантов кроссинговера принимают равновероятными. При этом схемы отбора особей и варианты кроссинговера используются случайным образом в соответствии с полученными частотами (вероятностями) их применения.

Для реализации адаптивной схемы репродукции рассматривались [7]:

- схемы отбора особей:

- аутбридинг;
- инбридинг;
- случайный отбор;
- варианты кроссинговера:
 - одноточечный;
 - двухточечный;
 - универсальный;
 - триадный (с формированием маски кроссинговера из лучшей особи элитной популяции);
 - триадный с учетом схемы.

В связи с незначительным влиянием мутации особей на качество получаемого решения использовался только один вариант мутации – одноточечная мутация.

Экспериментальные исследования

Для оценки эффективности реализации параллельной схемы выполнения островного генетического алгоритма, а также для получения рационального (квазиоптимального) решения в условиях жестких временных ограничений предлагается использовать кластерную вычислительную систему, конфигурация, которой представлена на рисунке.

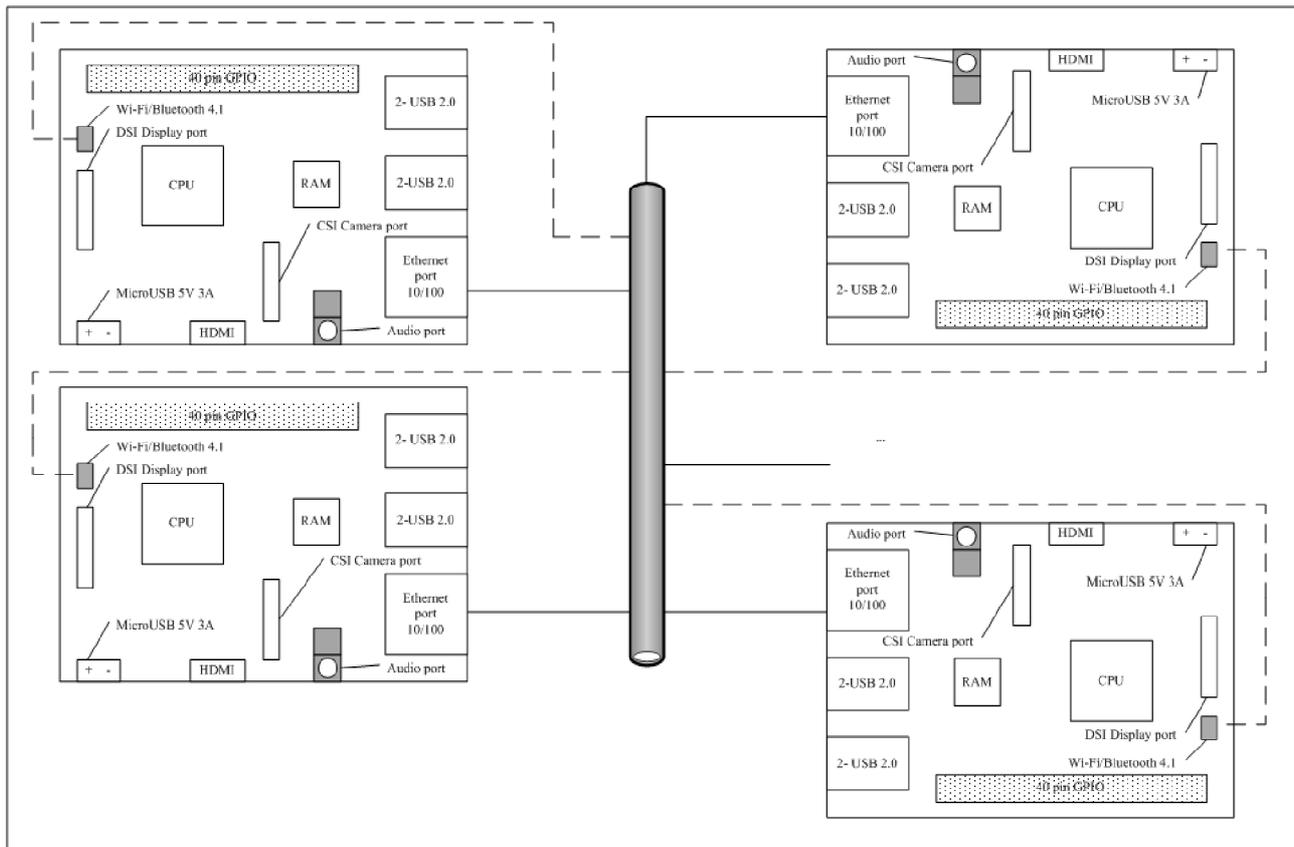


Схема кластерной вычислительной системы

Технические характеристики вычислительных модулей кластера, а также коммутатора и средств коммуникации представлены в таблице 1 и 2.

Данная кластерная система относится к классу систем с разделенной памятью [7].

Таблица 1 – Технические характеристики вычислительных модулей

| Технические характеристики | RaspberryPi 3 Model B |
|----------------------------|---|
| Процессор | Broadcom 2837 quad-core ARM Cortex-A53 64bit (1,2GHz) |
| Оперативная память | 1Gb |
| Видеовыход | HDMI |
| A/V выход | A/V выход 3.5мм jack 4 pin |
| USB порты | USB 2.0 x 4 |
| Сеть | WiFi 802.11n, 10/100Mb RJ45 Ethernet |
| Bluetooth | Bluetooth 4.1, BluetoothLowEnergy |
| Слот для карты памяти | Micro SD |
| GPIO | 40 |
| Операционная система | Raspbian Jessie (2016-05-27) |

На каждом вычислительном узле функционируют собственные копии операционной системы.

В этом случае единственно возможным механизмом взаимодействия между вычислительными модулями кластера является модель с передачей данных. При выполнении буферизированной передачи управление сразу же возвращается в вызывающую процедуру – сообщение при этом копируется в системный буфер и ожидает пересылки. В кластерной вычислительной системе представленной структуры и конфигурации, на каждый вычислительный модуль предлагается к установке операционная система Raspbian Jessie, основанная на Debian Jessie 8.5. Под управлением данной операционной системы выполняются параллельные ветви программы, реализующей ОГА.

Запуск программы может осуществляться как с внешнего компьютера, подключенного к кластеру, с помощью коммутатора или по *Wi-Fi* удаленно путем отправки команды запуска программы на главный вычислительный модуль, так и непосредственно с главного вычислительного модуля кластера. Для реализации программы решения задач обеспечения устойчивости функционирования распределенных информационных систем предлагается применение языка C++ с использованием библиотек MPI, поддерживаемых выбранной операционной системой.

Таблица 2 – Технические характеристики коммутатора и средств коммуникации

| Технические характеристики | Коммутатор D-LINK (DES-1005A/E) |
|--------------------------------------|---|
| Количество портов коммутатора | 5 x Ethernet 10/100 Мбит/с |
| Внутренняя пропускная способность | 1 Гбит/с |
| Интерфейсы среды передачи | Автоматическое определение MDI/MDIX на всех портах |
| Топология | Звезда |
| Скорость фильтрации/передачи пакетов | +Ethernet: 14,880 пакетов в секунду на порт +FastEthernet: 148,800 пакетов в секунду на порт |
| Буфер RAM | 49К на устройство |
| Скорость передачи данных | +Ethernet: -10 Мбит/с (полудуплекс) -20 Мбит/с (полный дуплекс) + FastEthernet: -100 Мбит/с (полудуплекс) -200 Мбит/с (полный дуплекс) |
| Средства коммуникации | Patch cord Hyperline RJ-45 category 6 U/UTP 0.3m -LSZH-ВК |

Заключение

Особенностями предложенной кластерной вычислительной системы являются хорошая масштабируемость, определяемая возможностями коммутационного оборудования, невысокая стоимость, возможность применения бесплатно распространяемого программного обеспечения.

Практическое использование кластерной вычислительной системы позволит высвободить вычислительные средства распределенной информационной системы от решения непрофильных задач при минимальных затратах.

Таким образом, предложенная кластерная вычислительная система в условиях жестких временных ограничений с помощью параллельной реализации островного генетического алгоритма позволит получать рациональное (квази-оптимальное) решение задач обеспечения устой-

чивости функционирования распределенных информационных систем за еще более короткие сроки, чем при традиционной реализации.

Библиографический список

1. **Есиков Д. О.** Задачи обеспечения устойчивости функционирования распределенных информационных систем // Международный научно-практический журнал «Программные продукты и системы». № 4 (112). 2015. Тверь: Научно-исследовательский институт «Центрпрограммсистем».

2. **Сергиенко И. В., Гуляницкий Л. Ф., Сиренко С. И.** Классификация прикладных методов комбинаторной оптимизации // Кибернетика и системный анализ. – 2009.

3. **Алексеев О. Г.** Комплексное применение ме-

тодов дискретной оптимизации. – М.: Наука, 1987. 248 с.

4. **Карпенко А. П.** Популяционные алгоритмы глобальной поисковой оптимизации. Обзор новых и малоизвестных алгоритмов. // Информационные технологии. 2012. №7. С. 13-15.

5. **Михалевич В. С., Волкович В. Л., Волошин А. Ф.** Метод последовательного анализа в задачах линейного программирования большого размера // Кибернетика.-1981.-№ 4.-с. 114-120.

6. **Holland J. H.** Adaptation in Natural and Artificial Systems: 2nd ed. Cambridge, MA: MIT Press, 1992.

7. **Букатов А. А., Дацюк В. Н., Жегуло А. И.** Программирование многопроцессорных вычислительных систем.–Ростов-на-Дону: Издательство ООО «ЦВВР». 2003. 208 с.

UDC 004.8, 519.85

COMPUTER CLUSTER FOR SOLVING TASKS TO ENSURE THE SUSTAINABILITY OF DISTRIBUTED INFORMATION SYSTEMS

A. N. Ivutin, Head of the Department, Associate Professor, TSU,Tula, PhD (technical sciences), alexey.ivutin@gmail.com.

D. O. Yesikov, post-graduate student, TSU,Tula, mcgeen4@gmail.com.

An island genetic algorithm is proposed to be used in solving sustainability problems of functioning of distributed information systems. A description of the proposed island genetic algorithm to solve the problems of ensuring the sustainability of distributed information systems is given. For rational (quasi) solutions under tight time constraints a cluster computing system is proposed to be used. Cluster structure, composition and parameters of the software and hardware are determined. The tables with the characteristics of computing modules, switches, and communication tools are given. The instructions for configuring each computer module in a clustered computer system, and for running the programs on each module are offered.

Key words: island genetic algorithm, cluster, distributed information system, cluster structure, cluster parameters, multi-agent algorithm, computational modules, a cluster computing system.

DOI: 10.21667/1995-4565-2016-57-3-63-67

References

1. **Esikov D. O.** Zadachi obespechenija ustojchivosti funkcionirovanija raspredelennyh informacionnyh system. Mezhdunarodnyj nauchno-prakticheskij zhurnal “Programmnye produkty i sistemy” no. 4 (112), 2015, Tver', Nauchno-issledovatel'skij institut «Centr-programmsistem» (In Russian).

2. **Sergienko I. V., Guljanickij L. F., Sirenko S. I.** Klassifikacija prikladnyh metodov kombinatornoj optimizacii. Kibernetika i sistemnyj analiz. 2009 (In Russian).

3. **Alekseev O. G.** Kompleksnoe primenenie metodov diskretnoj optimizacii. – Moscow: Nauka, 1987, 248 p. (In Russian).

4. **Karpenko A. P.** Populjacionnye algoritmy

global'noj poiskovoj optimizacii. Obzor novyh i maloizvestnyh algoritmov. Informacionnye tehnologii. 2012, no. 7, pp. 13-15. (In Russian).

5. **Mihalevich V. S., Volkovich V. L., Voloshin A. F.** Metodposledovatel'nogo analiza v zadachah linejnogoprogramirovanijabol'shgorazmera. Kibernetika. 1981. no. 4, pp. 114-120.(In Russian)

6. **Holland J. H.** Adaptation in Natural and Artificial Systems: 2nd ed. Cambridge, MA: MIT Press, 1992 (In English).

7. **Bukatov A. A., Dacjuk V. N., Zhegulo A. I.** Programirovanie mnogoprocessornyh vychislitel'nyh sistem.Rostov-na-Donu. ООО «CVVR», 2003, 208 p. (In Russian).