

РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТА ПО СОЗДАНИЮ ЭВОЛЮЦИОНИРУЮЩЕГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

В статье рассматриваются результаты эксперимента, подтверждающего возможность построения эволюционирующей программной системы на основе коллектива цифровых организмов. Каждый из цифровых организмов обладает генотипом, определяющим алгоритм поведения и подверженным воздействию генетических операций, что позволяет наделить программную систему способностью к адаптации и самоорганизации.

Введение

В настоящее время процесс создания программного обеспечения, как правило, сводится к компоновке и обеспечению взаимодействия уже существующих программных единиц и информационных ресурсов, являющихся результатом деятельности множества промышленных и научных коллективов, отдельных разработчиков. К этим программным единицам и ресурсам можно отнести библиотеки, модули, программно реализованные интерфейсы сопряжения, сетевые базы данных, средства операционной системы, программы-серверы, обеспечивающие приложения услугами, встраиваемые компоненты plug-in.

Очевидно, что в связи с развитием сетевых технологий и сложившейся тенденцией к переносу основной обработки информации в информационно-вычислительные сети [1] выявленные особенности процесса разработки программного обеспечения будут усиливаться. Следствием этого станет необходимость в решении концептуальных задач, которыми являются:

- разработка единого механизма управления программными единицами и информационными ресурсами, используемыми при построении программного обеспечения;

- разработка новой концепции построения программного обеспечения, учитывающей особенности процессов распределенной обработки информации в сети, главными из которых являются высокая степень неопределенности состояния вычислительного окружения и бесконечность множества его возможных состояний.

Предложенная в [2] модель биоценоза содержит решение перечисленных задач. В соответствии с этой моделью информационные и программные ресурсы организуются по единому принципу, предполагающему стратификацию по функциональному признаку и обеспечивающему уникальную идентификацию каждого отдельного ресурса и его разновидности. При этом иерархически организуются и становятся легко анализируемыми логические взаимосвязи между программными и информационными ресурсами. Концепция построения программного обеспечения на основе модели биоценоза такова, что позволяет учесть неопределенность и многообразие состояний вычислительного окружения и адаптироваться к этим особенностям благодаря способности программного обеспечения к эволюционированию.

1. Модель цифрового организма

Идея создания эволюционирующего программного обеспечения была выдвинута еще А. М. Тьюрингом [3]. Тьюринг первым предположил возмож-

ность разработки программ путем эволюции на основе принципов, аналогичных принципам биологической эволюции. Спустя десятилетия благодаря результатам теории генетических алгоритмов и эволюционных вычислений с появлением теории искусственной жизни идеи Тьюринга были развиты в рамках теории генетического программирования [4] в моделях теории искусственной жизни Tierra [5], Avida [6, 7], Darwinbots [8].

Однако ни в одной из перечисленных моделей и теорий не был предложен подход, который мог бы получить массовое практическое применение. Недостатком генетического программирования является ориентация исключительно на язык программирования Лисп и сложность выполнения генетических операций над хромосомами древовидной структуры. Модели Tierra, Avida и Darwinbots носят академический характер и не применяются для разработки утилитарного семантически значимого программного обеспечения. В этих моделях эволюционирующие программы представляют собой произвольные самовоспроизводящиеся наборы инструкций на специализированных версиях ассемблера, в которых эволюционным изменениям подвергаются или отдельные команды (в моделях Tierra и Avida), или группы команд (в модели Darwinbots).

В модели биоценоза [2] структура эволюционирующей программы сходна с моделями Tierra, Avida, Darwinbots и также соответствует понятию цифрового организма, введенному для программ-особей этих моделей. Цифровые организмы, в которых программный код одновременно является исходными данными для генетических операций, можно рассматривать как очередной этап развития концепции соединения структур данных и программного кода в единый активный интеллектуальный объект (рис. 1).

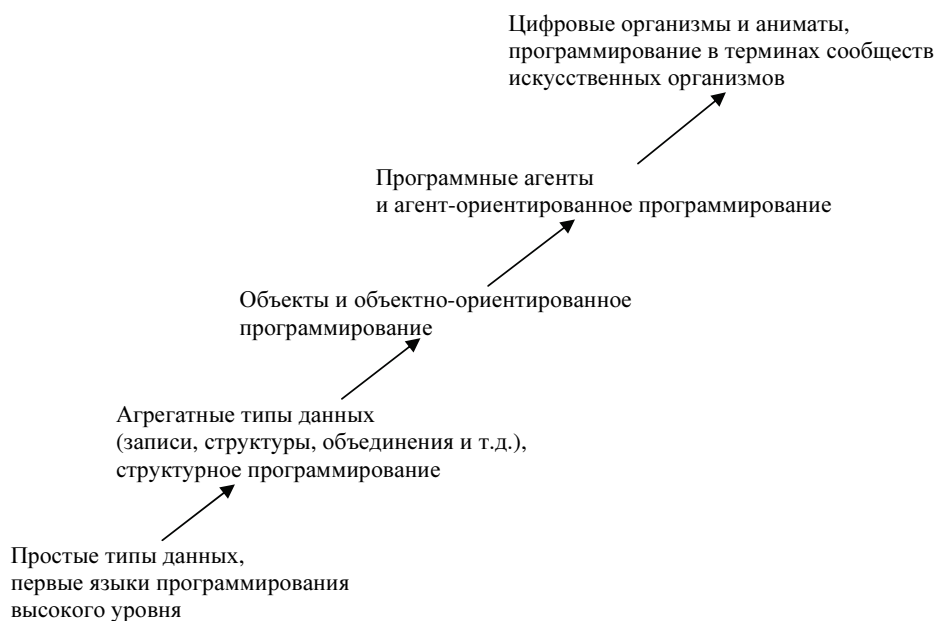


Рис. 1 Эволюция структур данных и парадигм программирования

Однако, в отличие от моделей Tierra, Avida и Darwinbots, эволюционирующие программы в модели биоценоза могут использоваться в практических целях для построения сложных многоуровневых программных систем.

Структура эволюционирующей программы в модели биоценоза описывается суперхромосомой, представляющей собой линейную последовательность нумерованных элементов, называемых локусами. С каждым номером локуса однозначно связана определенная семантическая интерпретация, поэтому у двух различных особей локусы с равными номерами определяют одни и те же атрибуты, хотя значения этих атрибутов могут быть различными. Каждый из локусов содержит либо особое «пустое» значение, либо целочисленный элемент индексного множества. Значение локуса может определять номер программной реализации элемента алгоритма поведения особи или индекс значения переменной в заданном домене. Следовательно, структура генотипа цифрового организма универсально представляет и алгоритм поведения, и значения переменных состояния.

В результате эволюция программ не нарушает общей логики их работы и не лишает их осмысленности. Это обстоятельство позволяет создать на основе предложенной концепции технологию разработки программного обеспечения массового применения. Эволюционные изменения способствуют адаптации программного обеспечения к изменяющимся условиям функционирования и особенностям вычислительной среды. Более того, линейная структура генотипа обеспечивает прозрачность логики работы эволюционирующей программы, делает возможным выявление и удаление цифровых организмов с нежелательными свойствами, позволяет накладывать ограничения на возможность изменения значений локусов и на множество допустимых значений локусов.

В отличие от цифровых организмов моделей Tierra и Avida, цифровые организмы в модели биоценоза могут обладать свойством мобильности, приближаясь в этом смысле к мобильным программным агентам и еще одной разновидности эволюционирующих программ-аниматов, обладающим нейросетью, развивающейся в процессе эволюции.

2. Описание эксперимента

Принципы построения эволюционирующего программного обеспечения, рассмотренные выше, стали основой для реализации программной платформы Padme, созданной автором. Данная платформа реализуется в виде сервера, который осуществляет трансляцию генотипа цифрового организма в фенотип, т.е. в программный код и набор переменных состояния, проверяет соответствие программного кода заданным ограничениям, а также осуществляет запуск программ-особей на выполнение. Платформа может функционировать как автономно на отдельной ЭВМ, так и на базе вычислительной сети. В последнем случае серверы Padme, установленные на разных узлах сети, соединяются друг с другом с помощью логических каналов, определяя тем самым топологию пространства искусственного мира.

В ходе эксперимента был создан искусственный мир с тороидальной топологией двумерного клеточного пространства из 8 клеток, $x \in \{0, 1\}$, $y \in \{0, 1, 2, 3\}$. Искусственный мир населяли цифровые организмы, выполнявшие обработку пассивных данных, организованных в виде файлов, а затем уничтожавшие исходные файлы. За обработку каждого файла и его уничтожение цифровому организму начислялось десять баллов, что является аналогом питания. Организм, набравший свыше 200 баллов, временно прекращал

«питаться» и мог приступать к воспроизводству. В модели имитировалось размножение от двух разнополых родителей с использованием одноточечного кроссинговера. Особи пола n_1 сохраняли свою генетическую информацию в служебных файлах, из которой особи пола n_2 с использованием своей генетической информации формировали новые особи, равновероятно принадлежащие к полу n_1 или n_2 . Новорожденная особь получала 20 баллов, каждый из родителей терял при воспроизводстве десять баллов. При перемещении на одну клетку пространства особь теряла один балл. Особь, имеющая нулевое количество баллов, «умирала», т.е. прекращалось выполнение соответствующей программы. Суперхромосома цифрового организма включала в себя семь локусов, определяющих программную реализацию алгоритма поведения, в том числе и локус, определяющий алгоритм перемещения особи в пространстве. Допускались хромосомные мутации, изменяющие значения локусов, а также мутации делеции, дефишенсы, удлинения хромосомы.

Целью проведенного эксперимента было установить способность коллектива эволюционирующих программ к самоорганизации в условиях различного пространственного распределения источников обрабатываемых данных и статистических характеристик этих источников. В данном случае самоорганизация означает нарушение симметрии на множестве используемых особями алгоритмов поведения в сторону обеспечивающих лучшую приспособленность своим обладателям. Решить поставленную задачу аналитическим путем невозможно.

Эксперимент состоял из двух независимых частей. В первой части эксперимента источники исходных файлов данных располагались в клетках виртуального пространства с координатами $x = 0$. В каждой клетке располагался один источник, который с интервалом в одну секунду генерировал пять файлов исходных данных. Во второй части эксперимента в каждой клетке пространства находилось по одному источнику, каждый из которых генерировал от одного до десяти исходных файлов данных с интервалом от одной до пяти секунд. Количество сгенерированных файлов и величина паузы определялись как случайные числа. В обеих частях эксперимента генерация данных источниками производилась в течение 120 мин. В начале каждой из частей эксперимента в модель вводились две родительские особи, имеющие, за исключением половых различий, идентичный генотип. Этот генотип является исходным (или «диким») и предполагает перемещение особи в соответствии с алгоритмом *vant* (двумерный вариант машины Тьюринга). Другими возможными алгоритмами перемещения, реализации которых хранились в библиотеках серверов Padme, являлись алгоритмы перемещения по спирали и алгоритм прямолинейного движения при $x = \text{const}$ или $y = \text{const}$. Условием прекращения эксперимента являлась «смерть» всех цифровых организмов, т.е. прекращение выполнения всех прикладных процессов, запущенных серверами Padme.

3. Результаты эксперимента

Результаты эксперимента (рис. 2–5) позволяют убедиться в том, что статистические характеристики потоков исходных данных, поступавших в модель, и пространственное распределение источников данных оказывают влияние на выбор коллективом цифровых организмов оптимального алгоритма поведения.

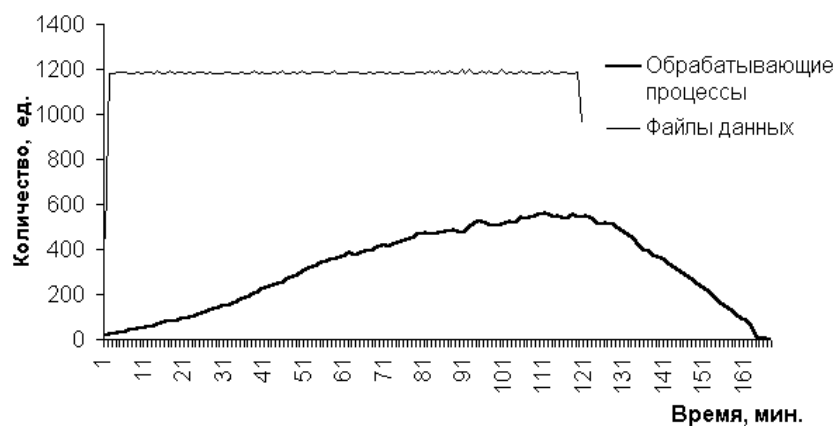


Рис. 2 Изменение количества файлов данных и обрабатывающих процессов (первая часть эксперимента)

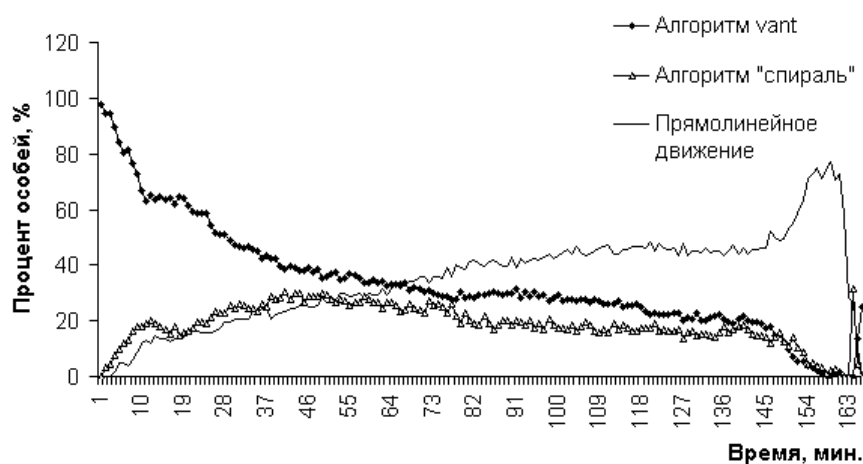


Рис. 3 Преобладающие генотипы (первая часть эксперимента)

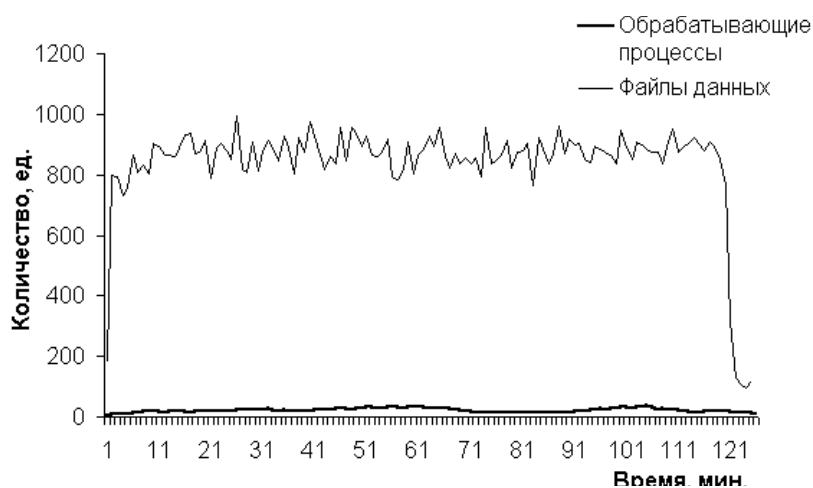


Рис. 4 Изменение количества файлов данных и обрабатывающих процессов (вторая часть эксперимента)

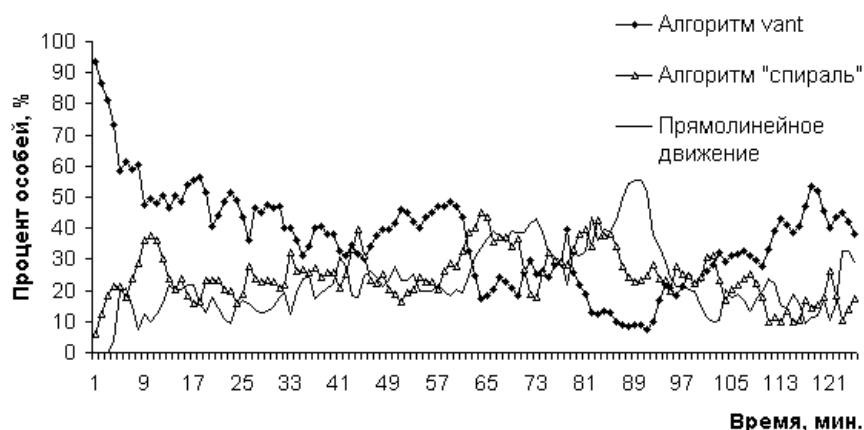


Рис. 5 Преобладающие генотипы (вторая часть эксперимента)

В случае первой части эксперимента (рис. 2, 3) данные характеристики потоков данных устойчивы во времени, распределению источников данных в пространстве присуща упорядоченность, что обуславливает эволюционные преимущества цифровых организмов с алгоритмом перемещения вдоль любой из осей координат. Особь с данным типом поведения гарантированно встречает на своем пути хотя бы один источник данных, а значит, находит «пищу». Количество подобных особей со временем неуклонно увеличивается и в модели начинает выделяться доминирующий генотип.

Результаты второй части эксперимента позволяют говорить о том, что аperiodические колебания численности файлов исходных данных в модели приводят, во-первых, к существенному снижению количества цифровых организмов в модели по сравнению с полученной в первой части эксперимента; во-вторых, к невозможности формирования безусловно доминирующего генотипа, хотя наиболее вероятным кандидатом на эту роль оказывается исходный «дикий» тип с алгоритмом перемещения vant.

Следует отметить, что в обеих частях эксперимента преобладали «дикий» тип и две его вариации, отличающиеся только реализацией алгоритмов перемещения, хотя в эксперименте допускались и другие мутации. На рисунках 3 и 5 приведены графики именно для «дикого» типа и его преобладающих вариаций. Прочие генотипы были менее устойчивы, их появление чередовалось с полным исчезновением.

Хотя и в первой, и во второй частях эксперимента источники данных функционировали в течение 120 мин, продолжительность первой части эксперимента составила 167 мин, а продолжительность второй части составила 126 мин. Подобное различие объясняется тем, что в первой части эксперимента возникали очереди к выходным логическим каналам, и особи простаивали в очередях, не теряя баллы.

Заключение

Результаты эксперимента, рассмотренного в данной статье, доказывают наличие у мобильных эволюционирующих программ, созданных на основе модели биоценоза, следующих свойств:

- способность к адаптации поведения и самоорганизации;

– способность к сохранению семантической целостности и практической значимости даже в условиях выполнения генетических операций над программным кодом;

– управляемость поведения коллективов программ.

Все это свидетельствует о возможности применения эволюционирующих программ, созданных на основе модели биоценоза, в практических целях.

Список литературы

1. **Гофф, М. К.** Сетевые распределенные вычисления: достижения и проблемы / М. К. Гофф ; пер. с англ. – М. : КУДИЦ-ОБРАЗ, 2005. – 320 с.
2. **Кольчугина, Е. А.** Моделирование биоценозов / Е. А. Кольчугина // Новые информационные технологии и системы : труды VI Международной научно-технической конференции (г. Пенза, 17–19 июня 2004). – Пенза : Изд-во ПГУ, 2004. – С. 240–246.
3. **Turing, A. M.** Computing Machinery and Intelligence / A. M. Turing // Mind (A Quarterly Review Of Psychology And Philosophy). – 1950. – Vol. LIX. – № 236. – October. – P. 433–460.
4. **Koza, J. R.** Genetic Programming: Biologically Inspired Computation That Creatively Solves Non-Trivial Problems / J. R. Koza, F. H. Bennett, D. Andre, M. A. Keane // Evolution as Computation, DIMACS Workshop, Princeton, January 1999. – Heidelberg : Springer-Verlag, 2001. – P. 15–44.
5. **Thearling, K.** Evolving Multi-Cellular Artificial Life / K. Thearling, T. S. Ray // Artificial Life IV Conference Proceedings. – Cambridge : The MIT Press, 1994. – P. 283–288.
6. **Adami, C.** On Modelling Life / C. Adami // Artificial Life IV, Proceedings of the Forth International Workshop on Synthesis and Simulation of Living Systems / Eds. R. A. Brooks, P. Maes. – Cambridge, MA : MIT Press, 1994. – P. 269–276.
7. **Adami, C.** Evolutionary Learning in the 2d Artificial Life System “Avida” / C. Adami, T. Brown // Artificial Life IV, Proceedings of the Forth International Workshop on Synthesis and Simulation of Living Systems / Eds. R. A. Brooks, P. Maes. – Cambridge, MA : MIT Press, 1994. – P. 377–381.
8. The Darwinbots Internet Site // <http://www.darwinbots.com>.