

## РАЗРАБОТКА СИСТЕМЫ ДИНАМИЧЕСКОГО ИЗМЕНЕНИЯ КОНТЕНТА ВИДЕОИГР НА ОСНОВЕ ЭВОЛЮЦИОННОГО МОДЕЛИРОВАНИЯ

*М.Г. Меженин*

*Темой данного исследования является применение методов эволюционного моделирования (ЭМ) для автоматического создания и динамического изменения мультимедийных данных в рамках процедурной генерации контента (ПГК) видеоигр. Использование таких методов ЭМ, как генетические алгоритмы, позволяет решить проблемы персонализации и адаптируемости контента, имеющую большую значимость в индустрии видеоигр. В работе приведен обзор существующих систем ПГК на основе ЭМ и предложен новый универсальный метод представления персонажей в подобных системах. В рамках исследования реализована система эволюционного моделирования, основанная на предложенном методе; в отличие от существующих решений, разработанная система является универсальной и может использоваться в различных коммерческих и научных проектах.*

*Ключевые слова: процедурная генерация контента, эволюционное моделирование, персонализация контента.*

### Введение

В современном мире индустрия электронных развлечений является одной из наиболее быстроразвивающихся в глобальной экономике; по различным оценкам, объем рынка видеоигр составляет от 50 до 70 миллиардов долларов. Все большой интерес к этой области появляется и у научного сообщества: создаются учебные направления, научные ассоциации, конференции и журналы. Одним из наиболее перспективных направлений исследований в этой сфере является процедурная генерация контента (ПГК), под которой понимают автоматическое алгоритмическое создание и динамическое изменение различных составляющих частей игр [11], а также правил и целей самого игрового процесса [3].

Использование ПГК позволяет не только значительно понизить стоимость создания контента, но и решает проблему персонализации, приобретающую большую значимость в связи с увеличением количества потенциальных игроков. Однако, в большинстве систем ПГК используются специализированные алгоритмы, применение которых ограничено параметрами одной конкретной игры [9]. Эту проблему можно эффективно решить с помощью методов эволюционного моделирования (ЭМ) [11, 12]. Таким образом, актуальной является задача разработки эволюционных моделей игрового контента различных видов и реализации на их основе универсальных систем ПГК.

В научной среде интерес к проблеме процедурной генерации контента в видеоиграх появился достаточно недавно. Теоретической базой данного исследования являются работы зарубежных ученых Д. Тогелиуса [11], Г. Яннакакиса [12] и М. Хендрикса [4] по классификации подходов к ПГК в видеоиграх.

В данной статье описывается разработка универсальной системы динамического изменения виртуальных персонажей на основе ЭМ для использования в различных научно-исследовательских и коммерческих игровых проектах и предложен новый метод представления персонажей игр в подобных системах.

Статья организована следующим образом. В первом разделе представлены историческая справка и обзор существующих подходов к ПГК в различных коммерческих и научно-исследовательских игровых проектах. Во втором разделе описан разработанный метод представления игровых персонажей в системах ПГК. В третьем разделе приведены детали проектирования и реализации системы динамического изменения контента на основе данного метода. В четвертом разделе описана апробация полученных результатов на основе разработанного прототипа видеоигры. В заключении суммируются основные результаты работы и рассматриваются направления дальнейших исследований.

## 1. Процедурная генерация контента на основе эволюционного моделирования

Под процедурной генерацией контента (ПГК) в видеоиграх понимается автоматическое создание и динамическое изменение различного игрового контента с помощью некоторого алгоритма [11]. Среди преимуществ ПГК отмечаются экономия средств, затраченных на производство, качественное повышение разнообразия и продолжительности игр, уменьшение требований к памяти устройств, а также возможность создания нового и необычного контента и даже совершенно новых игровых жанров, основанных на ПГК.

В той или иной мере, ПГК используется в видеоиграх с 1980-х годов; среди первых игр, важную роль в которых играл именно динамически генерируемый контент, можно отметить проекты *Rogue* (Toy and Wichma, 1980) и *Elite* (Braben and Bell, 1984). В настоящее время ПГК повсеместно используется в крупных коммерческих игровых проектах различных жанров, среди которых можно отметить стратегическую серию *Civilization* (Firaxis Games, 1991-2012), серию шутеров *Borderlands* (Gearbox Software, 2010-2012) и серию ролевых игр *Diablo* (Blizzard Entertainment, 1996-2013). Кроме того, ПГК используется в многочисленных играх от независимых разработчиков, например, в крайне популярной игре с открытым миром *Minecraft* (Mojang, 2009).

В современных коммерческих видеоиграх процедурная генерация используется для динамического создания различных видов контента: уровней, карт и ландшафтов, противников, снаряжения, а также игровых заданий. Несмотря на это, подход к генерации контента практически не изменялся; большинство реализаций основано на генераторе случайных чисел и не учитывает предпочтения пользователей или какие-либо другие факторы [5].

В качестве альтернатив классическому подходу исследователями предлагаются различные методы [4], однако только эволюционное моделирование (ЭМ) является достаточно гибким для представления различных видов игрового контента. Кроме того, ЭМ обеспечивает не только генерацию, но и адаптируемость контента к различным игрокам за счет эволюционного подхода. Общая схема работы алгоритмов ЭМ изображена на рис. 1.



Рис. 1. Схема работы алгоритмов эволюционного моделирования

В рамках ЭМ, эволюция может быть описана как многоступенчатый итерационный процесс, состоящий из случайных изменений в наборе изначально случайных решений («особей»), комбинирования («скрещивания») полученных результатов и последующего отбора лучших решений в новый набор («популяцию») на основе значений некоторой заданной функции оценки.

Использование методов ЭМ в системах ПГК отличается от классических алгоритмов следующими двумя ключевыми особенностями:

1) в системах ПГК используется интерактивная функция оценки, которая напрямую или опосредованно оценивает некоторый процесс взаимодействия с пользователями, а не сами особи;

2) целью ПГК в играх является не поиск некоторого единственного «оптимального» решения, а создание нового, разнообразного контента, способного увлекать игроков, в виду чего процесс генерации особей не ограничен по времени или итерациям.

Одним из главных факторов успешности эволюционного моделирования является выбор адекватного способа представления моделируемого контента, поскольку именно от него во многом зависит реализация алгоритма процедурной генерации, а также разнообразие и характеристики генерируемого контента [12]. Существуют различные способы представления контента в эволюционных алгоритмах, для каждого из которых характерны определенные типы генерируемого контента.

Наиболее распространенными и хорошо изученными являются различные способы представления контента на основе *векторов значений*, которые чаще всего используются для генерации игровых уровней [2]. В данном подходе каждое значение (представленное двоичным, целым или числом с плавающей запятой) соответствует некоторому параметру процедурного генератора.

Для представления контента в эволюционной модели могут использоваться различные виды *формальных грамматик*, которые задают алгоритм генерации контента и изменяются в процессе эволюции. Данный метод также используется для генерации игровых уровней [8].

Эволюционное моделирование на основе *искусственных нейронных сетей* является достаточно новым подходом и чаще всего используется для реализации адаптирующегося ИИ [10], однако существуют экспериментальные реализации данного метода для генерации графики и эффектов [5, 7]. Каждая особь в этом подходе представлена искусственной нейронной сетью, характеристики и структура которой изменяется в процессе эволюции.

Представление контента в виде *дерева* позволяет эволюционным моделям работать со сложными иерархическими видами контента, но вместе с тем значительно усложняет реализацию скрещивания из-за необходимости учета совместимости вершин в деревьях различной структуры. На основе данного метода была реализована система генерации трехмерных зданий [6].

Отличительным для игровой области является представление контента с помощью *структурных элементов*. Под структурными элементами понимаются небольшие логические компоненты, из которых состоит экземпляр контента. Каждый структурный элемент представляет собой кортеж значений разных типов, которые характеризуют данный компонент. Данный метод был предложен в рамках разработки универсального

подхода к процедурной генерации игровых уровней [9] и не использовался для представления других видов контента.

Таким образом, существующие решения в области процедурной генерации игрового контента на основе эволюционного моделирования можно разделить на пять основных видов в соответствии с используемым в них типом представления контента. Наиболее распространенным из них является класс представлений с помощью вектора значений, в то время как одним из самых перспективных является класс объектно-ориентированных представлений на основе структурных элементов.

Отметим, что в существующих научных работах не предпринималось попыток использовать системы ЭМ для моделирования характеристик игровых персонажей. По этой причине для данного вида контента необходима разработка соответствующего представления, учитывающего все параметры моделируемого контента. Проектирование такого представления описано в следующем разделе.

## 2. Разработка метода представления персонажей игр в системах ЭМ

Поиск наиболее подходящих способов представления различных видов контента является одной из главных проблем в области процедурной генерации [12]. Для определения требований к способу представления виртуальных персонажей в системе ЭМ их типичные характеристики были выявлены на основе анализа современных коммерческих видеоигр с динамической генерацией противников. В рассмотренных проектах персонажи чаще всего определяются тремя основными составляющими: числовыми характеристиками, способностями и категориями. Типичная структура, описывающая персонажей видеоигр, представлена на рис. 2.



Рис. 2. Структура виртуальных персонажей

На основе выявленных выше общих для виртуальных персонажей свойств можно сделать вывод, что данный вид игрового контента является комплексным и состоит из разнородных элементов. Ввиду этого факта целесообразным является разработка специализированного способа представления, который бы учитывал особенности структуры данного контента:

1) основные характеристики персонажей следует кодировать в виде вектора, для каждой из координат которого необходимо хранить название характеристики, используемый тип данных и область допустимых значений;

2) категорию персонажа нужно представлять в виде задаваемого пользователем перечислимого типа, значения которого интерпретируются в системе как целые числа;

3) для репрезентации способностей рационально использовать представление на основе структурных элементов, в которых тип взаимодействия, его объект и характеристика кодируются с помощью перечислимых типов, а значение и продолжительность изменения — с помощью числовых значений.

На рис. 3 приведен пример описания противника «Радскорпион» из игры Fallout 3 с помощью данного представления.

<p><b>Характеристики</b> (здоровье: 150, защита: 0.0, восприятие: 4, агрессивность: 2, уверенность: 4, взаимопомощь: 2, отравление: 0, ...)</p> <p><b>Категория</b> (тип: зверь)</p> <p><b>Способность</b> (тип: атака, объект: противники, характеристика: отравление, значение: 3, длительность: 5)</p>
---

Рис. 3. Пример представления противника из игры Fallout 3

Как видно из приведенного примера, разработанное представление может быть успешно использовано для репрезентации виртуальных персонажей различных видов в игровых системах ЭМ. Подобное представление позволяет разделять различные параметры персонажей и определять для каждого из них различные операторы скрещивания и мутации. Одновременно с этим, между ними сохраняется связь, которая обеспечивает простоту работы со всеми свойствами объекта целиком.

### 3. Разработка системы эволюционного моделирования

На основе описанного способа представления в рамках данной работы была разработана система эволюционного моделирования и динамического изменения виртуальных персонажей видеоигр. Общая схема работы разработанной системы изображена на рис. 4.

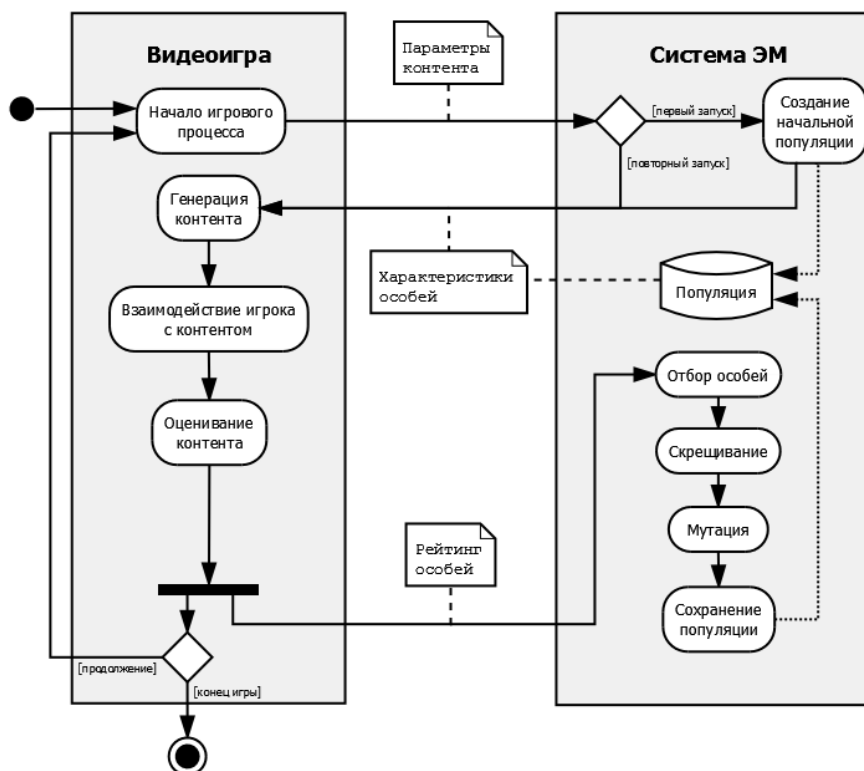


Рис. 4. Схема взаимодействия системы изменения контента с видеоигрой

Данная система изменения контента не зависит от реализации конкретной игры. Кроме того, она обеспечивает возможность гибкого задания параметров генерируемого контента во внешнем хранилище данных и может использоваться в качестве компонента различных игровых проектов.

По необходимости видеоигра запрашивает у системы ЭМ параметры текущей популяции, и на их основе генерирует виртуальных персонажей. Сгенерированный контент предоставляется игроку и, по результатам игровой сессии, каким-либо способом оценивается. Функция оценки на данном этапе может быть произвольной — например, пропорциональной нанесенному противниками урона игроку, некоторым образом интерпретирующей поведение игрока, и так далее.

Сведения об оценке каждого экземпляра популяции передаются обратно в систему ЭМ, в которой на основе полученных данных реализуются операции отбора, скрещивания и мутации. Результатом этого процесса становится новая популяция особей, которая может быть вновь передана видеоигре, либо сохранена во внешнее хранилище для последующей работы.

Ввиду использования в эволюционной модели нового способа представления контента, особый интерес представляет проектирование эволюционного алгоритма, а именно способ реализации в нем операций отбора, скрещивания и мутации особей.

Операция отбора реализована на основе метода турнирной селекции с копированием лучших особей в новую популяцию. В отличие от других методов, турнирная селекция не только гарантирует неубывание суммарной пригодности популяции, но и предотвращает преждевременное удаление потомков, способствуя разнообразию популяции [1]. Кроме того, поскольку на обучение игроков новым противникам тратится некоторое время, изменение всех особей популяции на каждом шаге нежелательно и целесообразным является использование частичной замены популяции.

Скрещивание особей происходит на основе попарного однородного кроссовера каждого из параметров родителей. Данный метод заключается в том, что значение каждой характеристики потомка с равной вероятностью копируется из характеристики одного из родителей. Способности потомка также выбираются случайным образом из способностей предков, общим числом не превышая максимального количества способностей предков.

Операция мутации для числовых характеристик основана на распределении Гаусса и заключается в изменении значения характеристики в соответствии со следующей формулой:

$$\begin{aligned} A' &= A + \Delta \times step, \quad \Delta \in \{-3; -2; -1; 1; 2; 3\}, \\ P(3) &= P(-3) = 0,021, \\ P(2) &= P(-2) = 0,136, \\ P(1) &= P(-1) = 0,341, \end{aligned} \tag{1}$$

где  $A$  — значение характеристики до мутации;  $A'$  — новое значение характеристики;  $step$  — минимальное изменение характеристики;  $\Delta$  — множитель, значения которого определяются в соответствии с плотностями вероятности нормального распределения.

#### 4. Интеграция и проверка адекватности системы

Для проверки адекватности разработанной системы и реализованных в ней алгоритмов, на ее основе был создан игровой прототип. Как уже было отмечено, разрабо-

танная система является универсальной и может быть интегрирована в существующие проекты.

Разработанный прототип представляет собой видеоигру, целью которой является набор очков, которые даются за убийство противников и снимаются при получении урона. Так как игровой процесс акцентирован именно на противниках, он хорошо подходит для тестирования системы.

Персонажи видеоигры характеризуются семью характеристиками. Один игровой уровень состоит из нескольких последовательных «волн» противников. Каждая волна представляет собой одну популяцию в эволюционной модели, и генерируется после уничтожения предыдущей. Размер популяции эволюционной модели зависит от конкретного уровня, количество особей в селекционном турнире равно 4, вероятность мутации — 30 %.

Для ранжирования сложности и интересности противников была разработана функция оценки, которую целесообразно использовать в играх рассматриваемого жанра. Данная функция оценивает противника тем выше, чем большую опасность он представлял (т.е. чем больше урона он нанес, а также насколько он успел приблизиться), а также негативно оценивает гибель игрока. Описанная функция имеет следующий вид:

$$f = \begin{cases} -r + k \times d, & \text{игрок выжил} \\ -r - k \times d, & \text{игрок погиб} \end{cases} \quad (2)$$

где  $r$  — расстояние от противника до игрока в момент гибели противника;  $d$  — нанесенный противником урон;  $k$  — коэффициент, задающий отношение между единицами измерения расстояния и урона.

Разработанная игра была предложена пятнадцати игрокам с различным уровнем умений. Для проверки возможности адаптации системы из различных начальных условий, каждому человеку было предложено пройти пять уровней различной сложности.

Нами было определено два основных критерия, по которым можно судить об адекватности разработанной системы. Продемонстрируем их выполнение на примерах.

По графику на рис. 5 видно, что на первом, легком уровне, данный игрок стабильно набирает очки, и сложность противников увеличивается. В то же время на четвертом уровне игрок теряет очки, и система снижает сложность противников. Таким образом, существует прямая зависимость между успешностью игры пользователей и сложностью генерируемых противников.

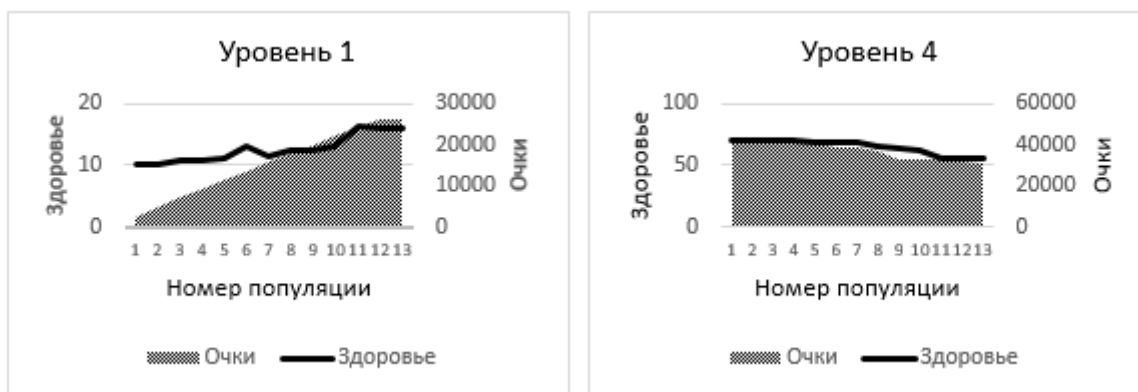


Рис. 5. Динамика изменения среднего значения характеристики «здоровье»

По графикам на рис. 6 видно, что в начале рассматриваемого уровня сложность обеспечивается высоким значением здоровья; в середине — низким здоровьем вместе с растущей силой; в конце — снова высоким здоровьем и снижающейся силой. Таким образом, мы показали наличие неоднородности и разнообразия изменений особей в различных популяциях ввиду наличия нескольких характеристик, влияющих на итоговую сложность персонажа.

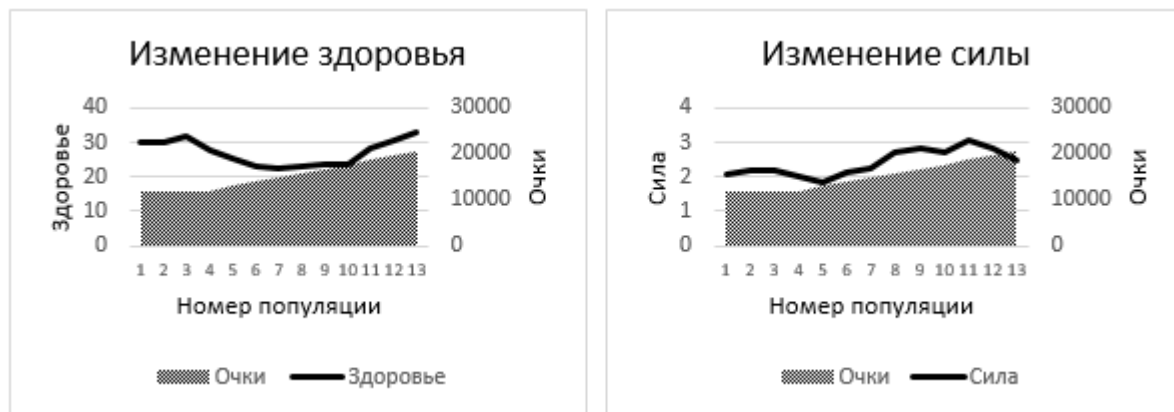


Рис. 6. Динамика изменения среднего значения характеристик «здоровье» и «сила»

На основе проведенного анализа результатов проверки адекватности можно утверждать, что разработанная система адекватно и корректно реализует требуемую функциональность. Характеристики персонажей успешно адаптируются к поведению пользователей, что подтверждается как статистическими данными, так и отзывами игроков. Использование эволюционного моделирования обеспечивает плавность изменения контента, одновременно с этим способствуя его разнообразию.

## Заключение

Данная статья посвящена исследованию возможности использования эволюционного моделирования для динамического адаптивного изменения виртуальных персонажей в видеоиграх на основе предпочтений и навыков пользователей.

В статье предложен способ представления игровых персонажей в системах эволюционного моделирования и описана разработанная на его основе универсальная система процедурной генерации контента. Для проверки корректности реализованной системы была проведена проверка ее адекватности, результаты которой продемонстрировали корректность решения рассматриваемой проблемы.

В дальнейших исследованиях планируется в следующих основных направлениях:

- 1) использование разработанного способа представления для кодирования в эволюционных моделях информации о других игровых объектах, в частности — предметах снаряжения, разделяющих с игровыми персонажами большинство свойств;
- 2) изучение возможности добавления в разработанную систему дополнительных средств задания ограничений на генерируемый контент;
- 3) реализация на основе данной работы многопользовательской игровой системы, агрегирующей результаты взаимодействия всех пользователей для генерации.



## Литература

1. Ashlock, D. Evolutionary Computation for Modeling and Optimization / D. Ashlock — New York: Springer, 2006. — 571 p.
2. Ashlock, D. Search-Based Procedural Generation of Maze-Like Levels / D. Ashlock // IEEE Transactions on Computational Intelligence and AI in Games. — Sept. 2011. — Vol. 3, No 3. — P. 260–273.
3. Cook, M. Multi-Faceted Evolution of Simple Arcade Games / M. Cook, S. Coulton // IEEE Conference on Computational Intelligence and Games. — Aug. 2011. — P. 289–296.
4. Hendrikx, M. Procedural Content Generation for Games: A Survey / M. Hendrikx, S. Meijer, J. Van Der Velden, A. Iosup // ACM Transactions on Multimedia Computing, Communications, and Applications. — Feb. 2013. — Vol. 9, No 1. — P. 1–22.
5. Hastings, E.J. Automatic Content Generation in the Galactic Arms Race Video Game / E.J. Hastings, R.K. Guha, K.O. Stanley // IEEE Transactions on Computational Intelligence and AI in Games. — Dec. 2009. — Vol. 1, No 4. — P. 245–263.
6. Martin, A. Evolving 3D Buildings for the Prototype Video Game Subversion / A. Martin, A. Lim, S. Colton, C. Browne // Proceedings of the 2010 International Conference on Applications of Evolutionary Computation. — Nov. 2010. — Vol. 6024. — P. 111–120.
7. Risi, S. Combining Search-based Procedural Content Generation and Social Gaming in the Petalz Video Game / S. Risi, J. Lehman, D.B. D'Ambrosio, R. Hall, K.O. Stanley // Proceedings of the 8th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment. — Oct. 2012. — P. 63–68.
8. Shaker, N. Evolving Levels for Super Mario Bros Using Grammatical Evolution / N. Shaker, G.N. Yannakakis, J. Togelius, M. Nicolau, M. O'Neill // IEEE Conference on Computational Intelligence and Games. — Sept. 2012. — P. 304–311.
9. Sorenson, N. A Generic Approach to Challenge Modeling for the Procedural Creation of Video Game Levels / N. Sorenson, P. Pasquier, S. DiPaola // IEEE Transactions on Computational Intelligence and AI in Games. — Sept. 2011. — Vol. 3, No 3. — P. 229–244.
10. Stanley, K.O. Real-Time Neuroevolution in the NERO Video Game / K.O. Stanley, B.D. Bryant, R. Miikkulainen // IEEE Transactions on Evolutionary Computation. — Dec. 2005. — Vol. 9, No 6. — P. 653–668.
11. Togelius, J. Search-Based Procedural Content Generation: A Taxonomy and Survey / J. Togelius, G.N. Yannakakis, K.O. Stanley, C. Browne // IEEE Transactions on Computational Intelligence and AI in Games. — Sept. 2011. — Vol. 3, No 3. — P. 172–186.
12. Yannakakis, G.N. Experience-Driven Procedural Content Generation / G.N. Yannakakis, J. Togelius // IEEE Transactions on Affective Computing. — July 2011. — Vol. 2, No 3. — P. 147–161.

Меженин Михаил Григорьевич, аспирант, лаборант кафедры системного программирования, Южно-Уральский государственный университет (Челябинск, Российская Федерация), m.mezhenin@gmail.com.

*Поступила в редакцию 25 декабря 2013 г.*

## EVOLUTIONARY-BASED PROCEDURAL CONTENT GENERATION FOR VIDEOGAMES

*M.G. Mezhenin*, South Ural State University (Chelyabinsk, Russian Federation)

*The focus of this research is applying evolutionary algorithms to automatically generating and dynamically changing multimedia data within the context of procedural content generation (PCG) for videogames. Evolutionary computing allows personalizing and adapting the generated content, which is an increasingly important issue in videogame industry. In this paper, we describe existing evolutionary-based PCG systems and propose a universal method for videogame characters representation in such systems. An evolutionary PCG system based on this method was implemented as a part of this research. The system is universal and is well suited for integration in commercial and scientific game projects.*

*Keywords: procedural content generation, evolutionary computation, content personalization.*

### References

1. Ashlock D. Evolutionary Computation for Modeling and Optimization. New York: Springer, 2006. 571 p.
2. Ashlock D. Search-Based Procedural Generation of Maze-Like Levels // IEEE Transactions on Computational Intelligence and AI in Games. Sept. 2011. Vol. 3, No 3. P. 260–273.
3. Cook M. Multi-Faceted Evolution of Simple Arcade Games // IEEE Conference on Computational Intelligence and Games. Aug. 2011. P. 289–296.
4. Hendrikx M. Procedural Content Generation for Games: A Survey // ACM Transactions on Multimedia Computing, Communications, and Applications. Feb. 2013. Vol. 9, No 1. P. 1–22.
5. Hastings E.J. Automatic Content Generation in the Galactic Arms Race Video Game // IEEE Transactions on Computational Intelligence and AI in Games. Dec. 2009. Vol. 1, No 4. P. 245–263.
6. Martin A. Evolving 3D Buildings for the Prototype Video Game Subversion // Proceedings of the 2010 International Conference on Applications of Evolutionary Computation. Nov. 2010. Vol. 6024. P. 111–120.
7. Risi S. Combining Search-based Procedural Content Generation and Social Gaming in the Petalz Video Game // Proceedings of the 8th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment. Oct. 2012. P. 63–68.
8. Shaker N. Evolving Levels for Super Mario Bros Using Grammatical Evolution // IEEE Conference on Computational Intelligence and Games. Sept. 2012. P. 304–311.
9. Sorenson N. A Generic Approach to Challenge Modeling for the Procedural Creation of Video Game Levels // IEEE Transactions on Computational Intelligence and AI in Games. Sept. 2011. Vol. 3, No 3. P. 229–244.

10. Stanley K.O. Real-Time Neuroevolution in the NERO Video Game // IEEE Transactions on Evolutionary Computation. Dec. 2005. Vol. 9, No 6. P. 653–668.
11. Togelius J. Search-Based Procedural Content Generation: A Taxonomy and Survey // IEEE Transactions on Computational Intelligence and AI in Games. Sept. 2011. Vol. 3, No 3. P. 172–186.
12. Yannakakis G.N. Experience-Driven Procedural Content Generation // IEEE Transactions on Affective Computing. July 2011. Vol. 2, No 3. P. 147–161.

*Received 25 December 2013*