

**А. Н. Кузнецов,**

Московский государственный университет имени М. В. Ломоносова,

**А. Б. Савинецкий,**

Национальный исследовательский ядерный университет «МИФИ», Москва

## ЭВОЛЮЦИЯ ПИКСЕЛОВ: ПРИШЕЛ, УВИДЕЛ, ПОБЕДИЛ

### Аннотация

Статья знакомит с впервые созданным интерактивным симулятором биологического видообразования: <http://evopix.soft-i.ru/>

**Ключевые слова:** эволюция, видообразование, искусственная жизнь, биоинформатика.

### Введение

Впервые создан интерактивный симулятор биологического видообразования: <http://evopix.soft-i.ru/><sup>\*</sup>. Архитектура биологической модели удовлетворяет одновременно научной достоверности и требованиям наглядности; визуальное отображение положено в самую ее основу — внедрен инновационный принцип изоморфного отображения. Разработаны быстродействующие вычислительные алгоритмы для основных блоков биологической модели: генетического, экологического (модель питания), репродуктивного, селективного и блока видообразования. Система подсчета очков с гандикапом превращает научную модель в увлекательную игру. Модель учит анализировать наблюдения в природе и переводить их на язык компьютерных экспериментов, направленных на подтверждение или опровержение эволюционных гипотез, и критически оценивать ограничения самих компьютерных моделей. Она учит ценить масштабы эволюционного времени и уникальность живых существ, а также красоту экспериментов и моделей. Эта игра дает практические навыки экспериментальной деятельности — как индивидуальной, так

и совместной, навыки действий в многопараметрической компьютерной среде, вырабатывает способность видеть, как эксперт, эволюционную подоплеку за биологическими явлениями.

Традиционное изучение эволюционной тематики всегда было чисто теоретическим (книжным), потому что природные эволюционные процессы имеют слишком большие масштабы во времени и пространстве для непосредственного восприятия человеческими органами чувств. Теперь компьютерные технологии позволяют моделировать и показывать эти процессы в реальном времени.

*A-life моделирование* — от английского *artificial life* (*искусственная жизнь*) — это интерактивные компьютерные симуляции различных аспектов жизни и эволюции биологических объектов, передовое, бурно развивающееся направление биологической информатики. Большинство моделей — клеточные автоматы разной степени сложности.

**Общий механизм эволюции в существующих моделях построен по следующему простому алгоритму:**

- 1) Пользователь создает, как будто он Бог, некий виртуальный организм — это может быть

<sup>\*</sup> В разработке программы принимали участие: Е. Н. Букварева (художник), И. В. Савельев (программист), С. Э. Сильвестров (веб-дизайнер) и многие другие (<http://evopix.soft-i.ru/druzya.html>). Первая версия продукта ([http://school-collection.edu.ru/catalog/rubr/2f2df4fb-6240-4c42-b476-1bf13538b678/?interface=catalog&class\[\]=49&subject\[\]=29](http://school-collection.edu.ru/catalog/rubr/2f2df4fb-6240-4c42-b476-1bf13538b678/?interface=catalog&class[]=49&subject[]=29)) выполнена при поддержке Национального фонда подготовки кадров (договоры № ELSP/A2/Gr/001-004-AUT-12/06, ELSP/A2/Gr/001-004-03/21/07). Вторая версия продукта (<http://evopix.soft-i.ru/>) выполнена при поддержке Фонда содействия развитию малых форм предприятий в научно-технической сфере (государственный контракт № 8521p/13771).

### Контактная информация

**Кузнецов Александр Николаевич**, доктор биол. наук, доцент кафедры зоологии позвоночных биологического факультета Московского государственного университета имени М. В. Ломоносова; *адрес:* 119234, г. Москва, Ленинские горы, д. 1, корп. 12; *телефон:* (495) 939-33-02; *e-mail:* sasakuzn@mail.ru

**A. N. Kuznetsov**,  
Lomonosov Moscow State University,

**A. B. Savinetsky**,  
National Research Nuclear University MEPhI (Moscow Engineering Physics Institute)

### PIXEL EVOLUTION: VENI, VIDI, VICI

#### Abstract

The article introduces the interactive emulator of biological speciation which was elaborated for the first time: <http://evopix.soft-i.ru/>

**Keywords:** evolution, speciation, artificial life, bioinformatics.

абстрактный программный код, группа ячеек в клеточном автомате или некий натуралистический монстр. Если участвуют несколько пользователей, то каждый создает свой организм и наделяет его различными свойствами (например, в натуралистической модели — ролью хищника или травоядного). Эти организмы и представляют собой стартовые виды.

- 2) Их помещают в соответствующую виртуальную среду — для кода это память компьютера, для клеточного автомата — поле клеток, для монстра — натуралистический ландшафт.
- 3) Затем запускается процесс размножения, от скорости которого зависит победа тех или иных организмов. В простейших клеточных автоматах размножение зависит только от конфигурации клеточных группировок. В более приближенных к реальности моделях организмы начинают потреблять соответствующие ресурсы среды и/или друг друга и используют полученные ресурсы для размножения — чем больше данный вид организмов потребляет ресурсов, тем быстрее он размножается. Возникает конкуренция за ресурсы и конкурентное вытеснение одних видов другими по принципу конкурентного исключения Гаузе либо прямое поедание хищниками жертв до полного уничтожения, если скорость размножения жертв ниже скорости размножения хищников. На этом этапе никакой эволюции еще не происходит, потому что созданные пользователем виды могут только исчезать, а не преумножаться.
- 4) Для эволюционного процесса вводятся мутации — случайно появляющиеся отклонения в конструкции виртуальных организмов, которые наследуются. Организм с мутацией (мутант) от рождения фактически считается новым видом. Получается: сколько мутаций, столько и новых видов (хотя обычно это не афишируется). Кто из них сильнее размножится и тем самым победит, зависит от того, повышает или понижает его мутация приспособленность, например скорость потребления ресурсов среды. Второй источник новых генотипов — скрещивание, благодаря которому, даже без мутаций, возникают новые комбинации наследственных единиц. В моделях, использующих скрещивание как механизм комбинаторики (необходимый элемент так называемых *генетических алгоритмов* поиска оптимальных решений различных задач), проблема видообразования вообще не поднимается.

**В действительности описанный «эволюционный» алгоритм в корне отличается от механизма, существующего в живой природе, по двум взаимосвязанным позициям:**

- 1) реальная эволюция происходит путем конкуренции и естественного отбора не между видами, а между особями внутри каждого вида, поскольку, как подчеркивал сам Дарвин, чем более сходны организмы, тем сильнее пересекаются их интересы;

- 2) в опубликованных виртуальных мирах все особи одного вида идентичны (фактически это не виды, а клоны), а в реальном мире каждый вид генетически неоднороден и видообразование происходит не при каждой мутации.

## Концепция

Исправление двух названных недостатков и стало нашим главным концептуальным нововведением. В свою очередь, **введение в модель внутривидового разнообразия потребовало постановки и решения следующих, ранее не ставившихся, проблем:**

- 1) Обеспечение загрузки стартовых поселений организмов, уже имеющих широкий генетический полиморфизм, причем с условием, чтобы пользователю было предельно легко этот полиморфизм задавать по собственному усмотрению.
- 2) Обеспечение разграничения внутривидового полиморфизма от межвидового в текущем времени (на каждом такте работы программы, соответствующем циклу размножения).
- 3) Отслеживание генеалогических связей виртуальных видов и построение соответствующего графа в режиме реального времени.

Дополнительным условием было максимально полное и адекватное отображение свойств виртуальных организмов (их генотипов, приспособленности и т. д.) на мониторе для того, чтобы пользователи легче вникали в непрерывно меняющуюся эволюционную ситуацию.

Для одновременного соблюдения всех этих разноречивых условий был усовершенствован подход, нередко встречающийся в A-life моделировании, а именно: в качестве виртуальной особи берется пиксел монитора. *Pixel* — от английского *picture element* — мельчайшая точка на мониторе или на электронной растровой картинке (например, фотографии в формате JPG).

**Было сделано два принципиальных нововведения:**

- 1) В качестве поля для жизнедеятельности и в качестве стартовых поселений организмов пользователь загружает произвольные файлы растровой графики — изображения с 24-битовой цветовой палитрой RGB (например, в формате BMP) — тем самым достигается максимальная простота в обеспечении стартового полиморфизма.
- 2) Введен общий *принцип изоморфного отображения* (рис. 1).

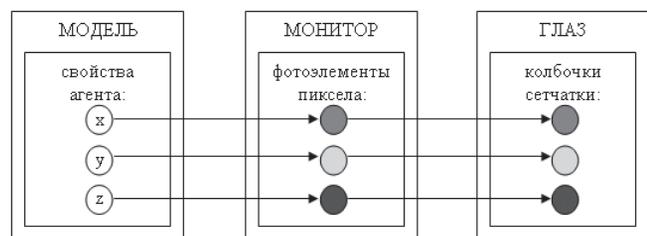


Рис. 1. Принцип изоморфного отображения

Глаз человека воспринимает цвет тремя типами колбочек сетчатки — одни более чувствительны к красной части спектра, другие — к зеленой, третьи — к синей. Монитор телевизора и компьютера уже сконструирован изоморфно глазу — имеет красные (R), зеленые (G) и синие (B) светящиеся элементы. Минимальная ячейка (клетка) изображения на мониторе — пиксел — состоит из тройки разноцветных элементов, от накала которых (т. е. от значений R, G, B) зависит, в каком цвете пиксел представится глазу. Принцип изоморфного отображения состоит в том, что модель тоже строится изоморфно монитору — каждый элементарный действующий агент модели отображается в виде одного пиксела; выбираются три существенных для наблюдателя свойства агента ( $x, y, z$ ), которые и преобразуются в RGB соответствующего пиксела. В целом диспозиция агентов модели предстает на мониторе в виде цветной картинке, легко, быстро и адекватно усваиваемой зрительной системой человека на интуитивном уровне.

Принцип изоморфного отображения сродни известному интерфейсному принципу WYSIWYG (от английского *What you see is what you get*) и может найти применение в компьютерном моделировании самых разных процессов в тех случаях, когда требуется максимально быстрая экспертная оценка сложных ситуаций по монитору.

В данной конкретной разработке параметрам  $x, y, z$  на рисунке 1 соответствуют генетические параметры виртуального организма (пиксела). На модели RGB\* основаны все алгоритмы разработанного симулятора эволюции — генетика, питание, размножение, отбор и видообразование в виртуальном мире пикселов.

## Модель

По распространенной классификации разработанная модель характеризуется как:

- individual-based (основана на особях);
- spatially explicit (с явно заданным пространством обитания);
- grid-like space (пространство разбито на клетки);
- immobile agents (особи оседлы).

Охарактеризуем вкратце основные блоки биологической модели.

**Модель среды обитания\*\*.** Каждый тур использования программы начинается с создания среды обитания, т. е. с загрузки поля (фона), поверх которого затем загружаются стартовые поселения организмов. Последние загружаются в четыре угла поля (не обязательно одновременно), так что в эволюционном процессе могут участвовать до четырех пользователей (игроков). В качестве этого поля, как и поселений, пользователь может брать достаточно

произвольные (с некоторыми техническими ограничениями) картинки растровой графики. Цвет среды обитания со временем меняется в связи с выеданием (см. «Модель питания»). Вместе с тем идет восстановление среды к исходному состоянию; оно происходит по экспоненциальному закону (скорость восстановления тем выше, чем больше отличие текущего цвета данной фоновой клетки от ее исходного состояния), причем скорость восстановления (крутизна экспоненты) назначается пользователем.

**Модель питания\*\*\*.** Пищей для организмов служит цвет клеток (пикселов) среды обитания (см. «Модель среды обитания»). Выедание цвета происходит покомпонентно. Конкретный характер выедания зависит от соотношения цветов пиксела-организма и клетки его проживания, а также от пользовательских настроек, которыми задается прожорливости и пищевая специализация, т. е. то, какой цвет воспринимается организмом как изобилие пищи или как ее отсутствие. При равном уровне маскировки более прожорливые организмы имеют преимущество в естественном отборе (см. «Модель приспособленности»).

**Модель размножения\*\*\*\*.** Особи могут размножаться тремя способами, в зависимости от пользовательских настроек:

- 1) бесполом путем — клонирование;
- 2) гермафродитно — случайное образование пар с перекрестным оплодотворением;
- 3) раздельнополым путем — самки выбирают самцов по цвету (пользователю предлагается несколько вариантов цветовых предпочтений самок), в связи с чем возникает половой отбор; возможны гаремы.

Пары образуются только между ближайшими соседями, потомство расселяется тоже только на ближайшие клетки вокруг материнского организма. После размножения возникает избыток численности организмов и возникает борьба за существование\*\*\*\*.

**Модель приспособленности\*\*\*\*\*.** Приспособленность каждой особи поселения измеряется цветовой разницей между этим пикселом-организмом и пикселом среды, на котором он живет или только еще претендует (рис. 2). Из нескольких претендентов на клетку выживает наиболее похожий на нее организм. Таким образом, происходит естественный отбор по маскировочной окраске. Разработан *метод визуализации уровня маскировки особи в градациях серого цвета*.

**Генетическая модель.** RGB-значения пиксела-организма принимаются за его генотип, представленный тремя генами R, G и B в трех хромосомах. Таким образом, организмы считаются гаплоидными, а три гена, располагаясь в разных хромосомах, передаются потомкам независимо друг от друга, то есть, например, унаследовав от матери аллель (вариант

\* Модель RGB: <http://evopix.soft-i.ru/rules/kak-ono-ustroeno/model-rgb.html>

\*\* Модель среды обитания (Восстановление поля): <http://evopix.soft-i.ru/rules/kak-ono-ustroeno/bocstanovlenie-polya.html>

\*\*\* Модель питания (Питание пикселов): <http://evopix.soft-i.ru/rules/kak-ono-ustroeno/pitanie-pikselov.html>

\*\*\*\* Модель размножения (Брачные игры): <http://evopix.soft-i.ru/rules/kak-ono-ustroeno/bpachnyie-igpyi.html>

\*\*\*\*\* Размножение, конкуренция, возраст: <http://evopix.soft-i.ru/rules/kak-ono-ustroeno/pazmnozhenie,-konkypencziya,-vozpact.html>

\*\*\*\*\* Модель приспособленности (Естественный отбор): <http://evopix.soft-i.ru/rules/kak-ono-ustroeno/ectectvennyij-otbor.html>

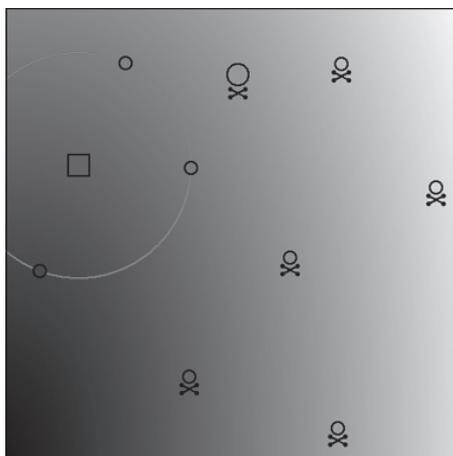


Рис. 2. Дарвиновский принцип выживания наиболее приспособленного организма в пространстве RGB (здесь для наглядности показано одно из его RG-сечений). Квадратик обозначает цвет клетки среды обитания, за которую идет борьба. Большой кружок — это пиксел-хозяин, выживший на данной клетке с прошлого года, а маленькие кружочки — это его конкуренты — претенденты из нового поколения. Трое лучших оказались по окраске на одинаковом, наименьшем из всех удалении от цвета клетки, т. е. одинаково хорошо приспособлены к жизни на ней. Победителя среди них определит «Его Величество Случай». Все остальные безоговорочно отбрасываются естественным отбором как менее приспособленные (более удаленные от идеальной приспособленности к клетке по уровню маскировки). В число проигравших в данном примере попал и старый хозяин. Если бы он оказался равен по приспособленности тройке лидеров, то предпочтение было бы отдано ему — как владельцу территории

гена R, потомок может с равной вероятностью унаследовать аллель гена G от матери или от отца. Конкретные целочисленные значения R, G и B каждого пиксела, которые в рамках 24-битовой палитры RGB могут принимать значения от 0 до 255, и являются аллелями этих генов, изоморфно отображающимися

в красный, зеленый и синий компоненты его цвета (фенотипа) (рис. 3). В общей сложности в пределах 24-битовой палитры RGB может быть реализовано до 256 аллелей каждого из трех генов и  $256^3$  фенотипов. Случайные изменения аллелей у потомков по сравнению с родителями — мутации — происходят с заданной пользователем частотой. Мутация считается летальной, когда численное значение аллеля оказывается меньше 0 или больше 255, т. е. за пределами 24-битовой палитры RGB. Разработан метод визуализации аллельных частот в виде гистограмм.

**Модель видообразования\***. Видовая принадлежность определяется по потенциальной скрещиваемости сосуществующих организмов путем полного перебора всех возможных пар вне зависимости от взаимной удаленности и половой принадлежности. Скрещиваемость измеряется потенциальным количеством потомков у данной пары. Она ступенчато падает при значительном несовпадении аллелей RGB у партнеров (порог несовпадения, препятствующий конъюгации виртуальных хромосом, задается пользователем). Вид определяется как связанное множество ныне живущих (сосуществующих) особей, внутри которого для каждой особи найдется хотя бы один партнер, при браке с которым мог бы получиться полный размер выводка. Разработан метод визуализации видообразования в виде графа генеалогических связей.

Один такт вычисления в программе фактически соответствует целому году реальной жизни\*\*. События, которые происходят за это время, выстроены в очередь, которую для наглядности удобно соотнести с привычными нам временами года:

- 1) Конец весны — начало лета. Восстановление пищевой базы.
- 2) Конец лета — начало осени. Питание.
- 3) Конец осени — начало зимы. Спаривание.
- 4) Конец зимы — начало весны. Рождение потомства, выживание наиболее приспособленных (замаскированных).

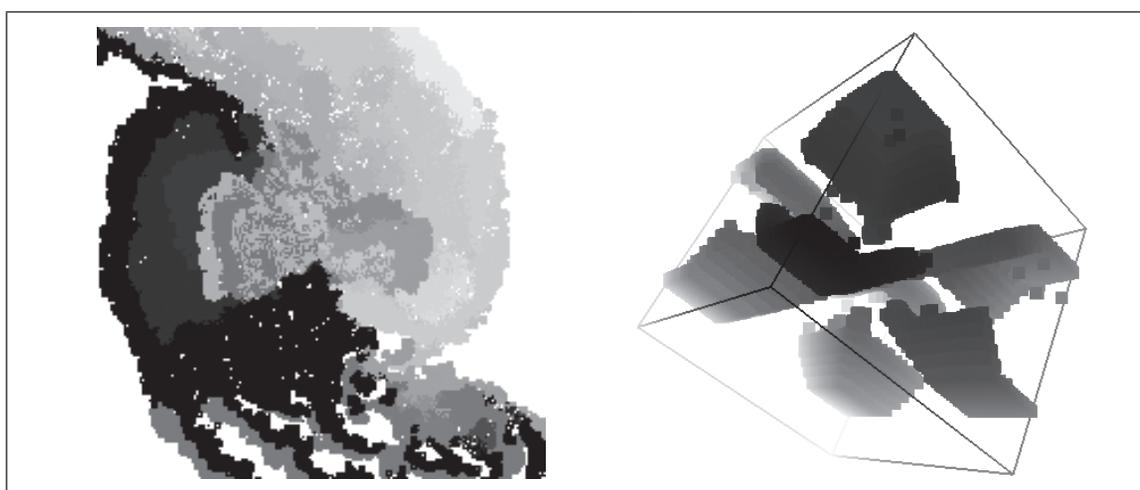


Рис. 3. Примеры отображения поселений пикселов на экране: естественное территориальное распределение (слева) и проекция в RGB-пространстве (справа)

\* Модель видообразования (Гены, степени бесплодия, виды): <http://evopix.soft-i.ru/rules/kak-ono-ustroeno/genyi,-ctepeni-becplodiya,-vidyi.html>

\*\* Годовой цикл: <http://evopix.soft-i.ru/rules/kak-ono-ustroeno/godovoj-czikl.html>

В этот момент программа определяет видовой состав выживших организмов и анализирует родственные связи получившихся видов с видами, жившими год назад. Затем годовой цикл повторяется. Особь может жить до 25 циклов; преждевременная смерть может наступить либо от голода, либо от проигрыша в борьбе за существование. Разработан метод визуализации возраста особей в градациях серого цвета.

## Игровые возможности

Как уже упоминалось, в виртуальном эволюционном процессе могут одновременно участвовать до **четырёх пользователей-игроков**, чьи виртуальные подопечные соперничают за территорию на общем поле, ибо в каждой его клетке может проживать только один организм. Таким образом, они взаимодействуют не как хищники и жертвы, а как конкуренты одного, как выражаются экологи, «трофического уровня», например как разные копытные — зебры, носороги, антилопы и жирафы африканских саванн.

Пользователям предоставляется весьма широкая свобода действий. Каждый из четверых может не только выбрать почти любую растровую картинку или фотографию в качестве своего стартового поселения, но и улучшить удобный момент, когда его «вбросить» на поле. Игроки не обязаны делать заселение одновременно, а в дальнейшем каждый совершенно волен решать, как часто ему стоит вмешиваться в естественный эволюционный процесс для вытеснения чужих поселений с территории. И вот здесь мы, для повышения азарта, устроили игрокам своего рода гандикап — каждому надлежит назначать себе «период невмешательства» — т. е. число виртуальных лет (тактов работы программы), в течение которых он лишает себя права менять биологические настройки своих подопечных. По истечении этого числа «лет» игрок может не только изменить настройки организмов, но и назначить новый период невмешательства.

С одной стороны, выгодно вмешиваться почаще, чтобы не терять контроль над эволюционным процессом. С другой стороны, за более долгое невмешательство начисляется больше очков при росте поселения и больше вычитается при его убыли — в этом и состоит гандикап\*. Фактически большой период невмешательства — это высокая игровая ставка. То, какую величину игрок назначает периоду невмешательства, отражает его азарт и самоуверенность в способности прогнозировать ситуацию, а то, какой от этого у него реально получается прирост очков или их убыль, помогает корректировать самооценку и исправлять величину периода невмешательства при следующем доступе к параметрам.

Для того чтобы эволюционный сеанс можно было переиграть или продолжить с любого места, разработан формат файлов для хранения возникающих эволюционных ситуаций (включая не только все параметры живущих на момент сохранения организмов, но и их полное эволюционное дерево). Повторно

загружая и запуская одну и ту же эволюционную ситуацию, можно быстро убедиться в роли случайности в эволюции: при идентичных стартовых условиях эволюционные эксперименты заметно различаются в своем дальнейшем развитии.

В компактном, но эффективном инструментарии, который мы разработали для управления эволюцией пикселей, наиболее мощным средством является редактор эволюционного древа\*\* (рис. 4). Редактор позволяет обращаться с эволюционным деревом буквально как садовнику, обрезая ненужные ветви и наделая другие (правда, строго по одной) новыми биологическими свойствами.

Мы не сомневаемся, что студенты и школьники быстро научатся обыгрывать в нашу эволюционную игру своих профессоров и учителей, и немного надемся, что те в свою очередь убедятся в недостаточности книжных знаний об эволюционных процессах (рис. 5).

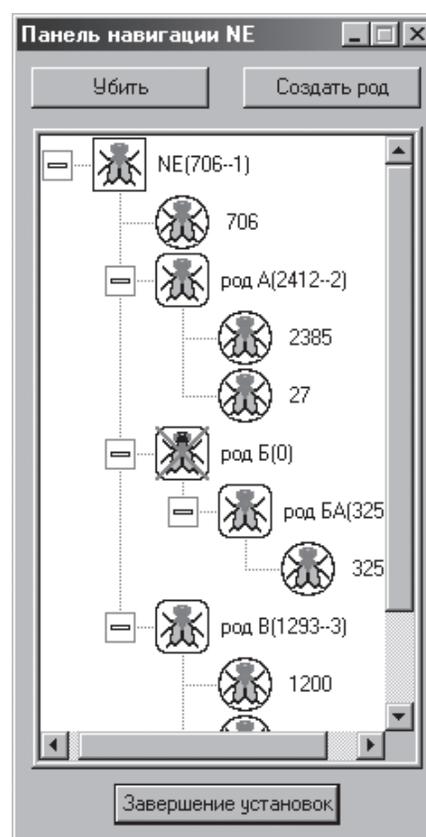


Рис. 4. Панель навигации и селекции служит для изучения того множества видов, которые на данный момент образовались в поселении, и для их искусственного отбора — выбраковки неудачных, с точки зрения пользователя, видов. Числа обозначают количество особей в каждом виде, а для родов — еще и количество входящих видов. Для каждого рода своего поселения пользователь может назначать особые биологические настройки, как то: прожорливость, соотношение самцов и самок, частоту мутаций и т. п. Кроме того, при каждом вмешательстве пользователь может присвоить какому-нибудь еще виду статус рода — т. е. сделать его новой полноценной фигурой на игровом поле эволюции

\* Счетчики: <http://evopix.soft-i.ru/rules/kak-etim-upravlyat/cchetchiki.html>

\*\* Навигация и селекция: <http://evopix.soft-i.ru/rules/kak-etim-upravlyat/panel-ypavleniya/havigaczziya-i-celekcziya.html>

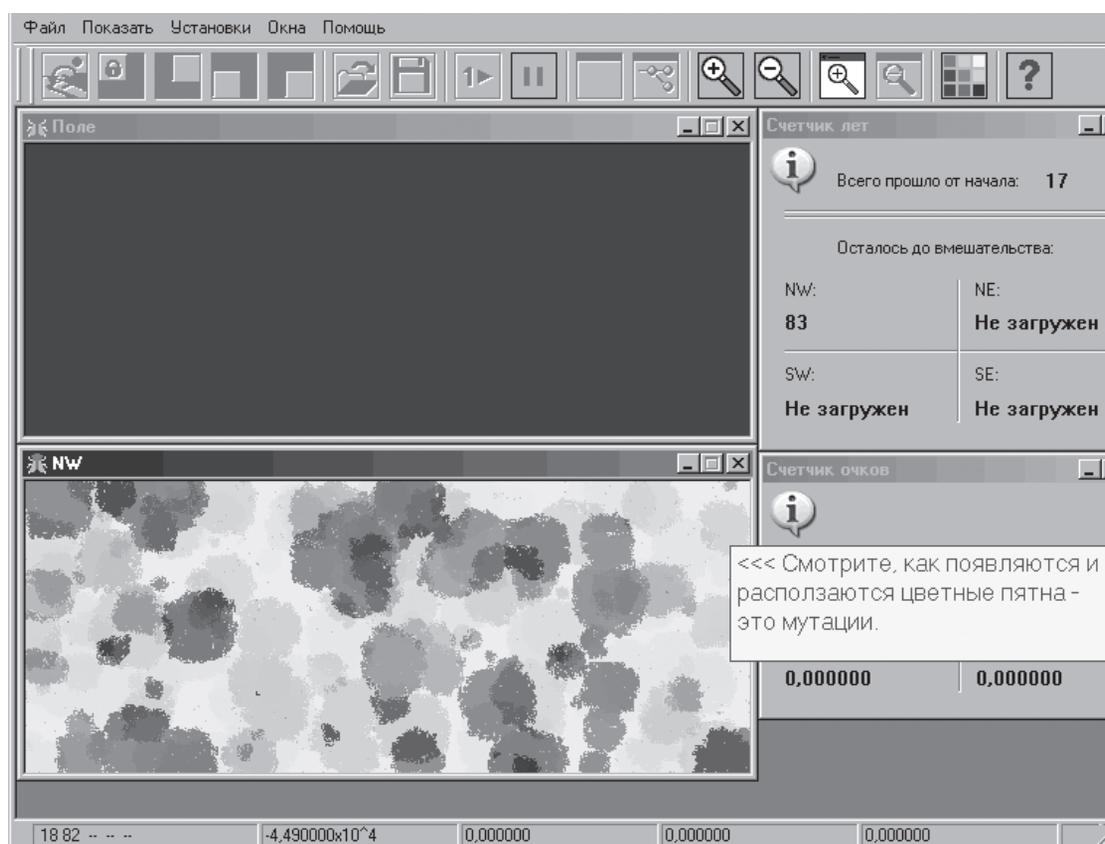


Рис. 5. Окно программы

## Методика

При реализации математической модели были разработаны методики обращения с большими массивами графической информации в реальном времени.

**Первая методика** связана с реализацией механизмов годового цикла жизнедеятельности виртуального поселения и представляет собой обработку массива пикселей с оптимизацией критических участков кода. Обработка осуществляется в два прохода. На первом проходе осуществляются все обработки, связанные с реализацией жизнедеятельности отдельной особи. На втором проходе осуществляется реализация конкуренции за клетку проживания между владельцем клетки и дочерними особями соседних родителей, на нее претендующими.

Выбор победителя осуществляется по уровню маскировки: кто из претендентов ближе всего по цвету к клетке, за которую идет борьба, тот ее и занимает. Цветовое сравнение между каждым претендентом и клеткой, строго говоря, должно измеряться по кратчайшей геометрической дистанции до нее в трехмерном пространстве RGB, т. е. по Евклидову расстоянию:

$$\rho_e(i, j) = \sqrt{(R_i - R_j)^2 + (G_i - G_j)^2 + (B_i - B_j)^2},$$

где:  $R_i, R_j$  — координаты  $i$ -го претендента и  $j$ -й клетки поля по оси  $R$ ;  $G_i, G_j$  — по оси  $G$ ;  $B_i, B_j$  — по оси  $B$ .

Поскольку в нашем случае имеет значение не точное значение  $\rho_e$  для каждого претендента, а лишь то, у кого из них оно минимально, можно обойтись

без извлечения каждый раз квадратного корня, что и было сделано для ускорения вычислений. В итоге у претендентов сразу сравниваются квадраты Евклидовых расстояний до клетки поля в пространстве RGB.

Если у претендентов разный уровень приспособленности, то результат детерминирован — клетку занимает, естественно, самый лучший. Если есть несколько лучших с равным уровнем приспособленности, то побеждает старый владелец клетки — по праву собственника. Если же владелец не относится к лучшим, то между лучшими претендентами происходит случайный розыгрыш (см. рис. 2).

Был применен прием, позволяющий уменьшить занятую память компьютера в случае равенства уровня приспособленности претендентов. Прямолинейное решение подразумевает накопление в памяти всех ( $N$ ) равных между собой наиболее приспособленных претендентов на клетку (со всеми их генетическими и прочими свойствами) и последующий равновероятный розыгрыш победителя между ними — случайный выбор одного из  $N$  с вероятностью  $1/N$  для каждого. Вместо такого накопления, для экономии расхода памяти, мы применяем попарное сравнение в очереди лучших претендентов. Для первой пары равных претендентов осуществляется случайный розыгрыш с вероятностями  $1/2$  для каждого. Факт такого розыгрыша записывается в дополнительную ячейку памяти как единица, и после каждого подобного розыгрыша туда прибавляется еще 1, а вся остальная информация о проигравшем сбрасывается из памяти (в этом и состоит экономия). После  $i$  подоб-

ных розыгрышей в дополнительной ячейке памяти оказывается число  $i$ , а розыгрыш между победителем последней дуэли и новым равным ему претендентом (порядковый номер которого получается  $i + 2$ ) осуществляется с вероятностями  $(i + 1)/(i + 2)$  и  $1/(i + 2)$  соответственно, что позволяет полностью учесть (с вероятностной точки зрения) предыдущие попарные розыгрыши и заменить тем самым прямолинейное решение (равновероятный случайный розыгрыш из всех накопленных претендентов сразу). Например, для последнего претендента очевидным образом  $i + 2 = N$ , и вероятность его победы  $1/(i + 2) = 1/N$ . Перемножив вероятности побед его противника в предыдущих дуэлях, нетрудно убедиться, что и для него в итоге тоже получится вероятность  $1/N$ .

После полного построчного прохода всех клеток и определения победителя на каждой из них осуществляется построение нового растрового изображения.

**Вторая методика** — реализация видообразования — оказывается более тяжеловесной задачей. По сути дела на каждом шаге (такте) стоит задача кластеризации особей, произвольно расположенных в трехмерном кубе, представляющем собой RGB-пространство. Кластеризация — это разбиение элементов некоторого множества на группы, т. е. классификация. Для ее проведения необходимо ввести понятие сходства — близости объектов по интересующим переменным характеристикам (в нашем случае — координатам в признаковом пространстве RGB). В каждый кластер должны попасть объекты, имеющие сходные характеристики.

В кластерном анализе для количественной оценки близости вводится понятие *метрики*. Сходство и различие между классифицируемыми объектами устанавливается в зависимости от метрического расстояния между ними. Если каждый объект описывается  $k$  признаками, то он может быть представлен как точка в  $k$ -мерном пространстве (в нашем случае — трехмерном). Сходство с другим объектом будет определяться как расстояние между ними. Наиболее часто применяемым типом расстояния является все то же Евклидово расстояние, а его квадрат употребляют, когда требуется придать большие веса более отдаленным друг от друга объектам, чем близлежащим. Но мы для определения видовой принадлежности использовали покоординатные абсолютные дистанции в пространстве RGB. Два пиксела-организма  $i$  и  $j$  считаются принадлежащими к одному виду (кластеру), если для них соблюдается следующее тройное условие:

$$|R_i - R_j| \leq d, |G_i - G_j| \leq d, |B_i - B_j| \leq d,$$

где:  $R_i, R_j$  — координаты  $i$ -го  $j$ -го организмов по оси  $R$ ;  $G_i, G_j$  — по оси  $G$ ;  $B_i, B_j$  — по оси  $B$ ;  $d$  — пороговое значение близости, которое задается пользователем в пределах от 0 до 255 и названо в интерфейсе «порог конъюгации». Это название открывает биологический смысл такой метрики близости: если значения аллелей хотя бы одного гена у  $i$ -го и  $j$ -го организмов отличаются больше, чем предусматривает заданный порог, то соответствующие хромосомы не будут, как положено, конъюгировать при мейозе и попытка образования гаплоидного потомка из диплоидной

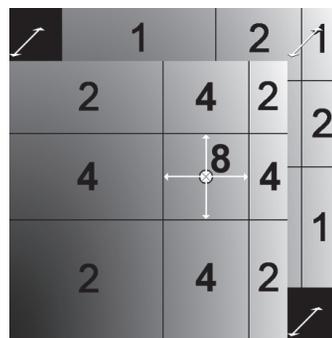


Рис. 6. Определение генетической совместимости пикселей при помощи покомпонентного сравнения их цветов.

Показаны два RG-сечения пространства RGB — одно за другим, со сдвигом; кружок на переднем из них обозначает «опорный» пиксел, с которым сравниваются все остальные; белые стрелки, параллельные координатным осям пространства RGB, представляют величину «порога конъюгации», заданного пользователем. Цифрами обозначено количество потомков, которое пиксели из данной области дадут при скрещивании с «опорным»: 8 — область пикселей, которые не отличаются от «опорного» ни по одной координате больше, чем на заданный порог, т. е. гарантированно относятся к тому же виду; 4 — области пикселей, которые отличаются от «опорного» больше, чем на заданный порог, только по одной координате; 2 — области пикселей с превышением порога по двум координатам; 1 — области пикселей с превышением порога по всем координатам

оплодотворенной яйцеклетки может не состояться — плодовитость пары  $i$ -го и  $j$ -го организмов будет ниже нормы (в мире пикселей она равна восьми потомкам), что и свидетельствует против их принадлежности к одному виду (рис. 6).

Алгоритмы кластерного анализа отличаются большим разнообразием. Это могут быть, например, алгоритмы, реализующие полный перебор сочетаний объектов, или более быстрые, состоящие из двух этапов. На первом этапе задается начальное (возможно, искусственное или даже произвольное) разбиение множества объектов на классы и определяется некоторый математический критерий качества автоматической классификации. Затем, во втором этапе, объекты переносятся из класса в класс до тех пор, пока значение критерия не перестанет улучшаться. Мы используем для ускорения вычислений (во избежание полного перебора бракосочетаний пикселей) восходящий алгоритм иерархической кластеризации, который состоит в следующем. Сначала кубическое признаковое пространство RGB подразделяется на кубики меньшего размера — их ребра равны заданному пользователем «порогу конъюгации», так что попадающие в один кубик пиксели-организмы заведомо относятся к одному виду. Разделенные по кубикам организмы формируют первичные кластеры. Затем ближайшие соседи из смежных кластеров сравниваются на предмет превышения того же «порога конъюгации», и, если ни по одной координате он не превышен, кластеры объединяются во все более и более крупные по цепочечному принципу.

**Интернет-источники**

1. <http://evopix.soft-i.ru/biblioteka.html>
2. <http://evopix.soft-i.ru/konkurentyi.html>