

# ПОИСК ОПТИМАЛЬНОЙ СТРАТЕГИИ НА ОСНОВЕ ЭВОЛЮЦИОННЫХ АЛГОРИТМОВ

Довнар А. Д., Шилин Л. Ю.

Кафедра систем управления, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: andrevdovnar@gmail.com

*В работе рассматриваются подходы к улучшению и практическому применению эволюционных алгоритмов. Описаны проводимые эксперименты, полученные результаты и перспективы дальнейшего развития системы. Основная цель данной работы – проверить отображения методов биологической эволюции на технические системы, попытаться формализовать основные правила и методы, используемые в эволюционных алгоритмах, а также найти способы оптимизации данных методов.*

## ВВЕДЕНИЕ

В последнее время поднимаются вопросы о возможности воссоздания принципов работы человеческого мозга с использованием вычислительных мощностей современных компьютеров. Одним из таких вопросов является поиск универсального метода обучения, который может быть использован для решения совершенно различных задач из различных областей знания. В работе, на примере двухмерной игры, рассматриваются существующие методы проектирования эволюционных алгоритмов и попытки их улучшения, а также попытки совмещения различных методов машинного обучения, для решения задач с известной и, затем, с неизвестной оптимальной стратегией поведения.

### I. ОПИСАНИЕ ЭКСПЕРИМЕНТА

Эксперимент представляет собой игровое поле размером 4 на 4 клетки. На поле запускается агент – объект, обучение которого будет проводиться в дальнейшем. «Существо» имеет возможность передвигаться в 4 направлениях: вверх, вниз, влево, вправо или стоять на месте. Также агент обладает параметром сытости, который уменьшается после каждого игрового дня. За игровой день объект может совершить лишь одно действие движения. Также на поле появляется еда, которая, при поглощении агентом, полностью восполняет ему параметр сытости. При достижении параметром сытости значения 0, агент считается погибшим. На поле всегда присутствует одна единственная еда, вне зависимости от методов ее появления.

Сложность задачи определяется способом появления на карте агента и еды. Например, простой задачей считается обучить агента находить еду при условии, что оба игровых объекта появляются в строго определенных местах. Сложной же задачей является задача, в которой агент появляется в случайно точке карты, а еда появляется в одной из нескольких точек карты случайным образом.

Мозгом агента выступает нейронная сеть прямого распространения, а множество весов нейронной сети называется геномом. На вход нейронной сети подаются 5 кодированных величин: координаты X и Y обоих объектов и величина голода агента. Кодирование величин необходимо для того, чтобы привести входные данные к виду, удобному для обучения нейронной сети. Выходными параметрами являются соответственно направления движения. Таким образом решается задача классификации, в которой из множества возможных направлений движения выбирается приоритетное для конкретной ситуации.

### II. ОПИСАНИЕ ЭВОЛЮЦИОННЫХ АЛГОРИТМОВ

Эволюционные алгоритмы относятся к классу алгоритмов обучения с подкреплением. То есть в решаемых задачах не присутствует непосредственного примера поведения, но при этом существует функция, значение которой непосредственно характеризует успех агента при выполнении поставленной задачи. По принципу действия эволюционные алгоритмы являются отображением биологической эволюции, а математически относятся к методам случайного перебора значений.

Обучение разделено на поколения. Первое поколение получает случайные геномы. В простых задачах каждая особь проходит задачу единственный раз. В сложных задачах каждая особь проходит задачу несколько раз, для того, чтобы особи, решающие, к примеру, большую часть различных ситуаций, с большей вероятностью получили высокий рейтинг. Фитнес функцией данной задачи является число прожитых дней. То есть основная цель агентов в игре – прожить наибольшее возможное количество дней или найти способ жить вечно при данной сложности игры. После того, как подсчитан рейтинг всего поколения, происходит отбор особей, показавших себя лучше всех, то есть наиболее приемлемых для того, чтобы дать потомство.

Новое поколение складывается из нескольких различных групп. В первую группу вводятся все особи прошлого поколения, получившие лучший рейтинг. Во вторую группу вводятся особи, являющиеся потомками первой группы. Геномы потомков получаются путем случайного перемешивания геномов родителей и возможных мутаций, то есть случайного переопределения некоторых генов [1]. Коэффициент мутации определяет вероятность того, что ген будет переопределен. Если остановиться только на двух группах агентов, то со временем, из-за маленькой вероятности появления в поколении новых генов, алгоритм упрется в локальный минимум. Для того, чтобы привнести разнообразие генов в новое поколение добавляется еще две немногочисленные группы агентов. В третью группу входят совершенно новые для экосистемы особи, то есть просто созданные случайным образом. В четвертую группу входят особи прошлого поколения, добавленные в новое поколение случайным образом вне зависимости от их рейтинга, при том эти особи так же, как и первая группа, участвуют в определении генома агентов второй группы. Каждое новое поколение по численности агентов должно быть точно таким же, как предыдущее.

### III. ОПРЕДЕЛЕНИЕ ПРОДВИЖЕНИЯ В РЕШЕНИИ ЗАДАЧИ

Эффективность решения задачи для всего поколения определяется по двум параметрам: средняя продолжительность жизни особи в поколении, количество особей, способных выживать бесконечно долго при данной сложности – золотые особи [2]. В целом, для того, чтобы задача считалась решенной, необходимо выполнение двух условий:

- Средняя продолжительность жизни поколения должна быть больше, чем половина максимальной продолжительности.
- Количество золотых особей не падает ниже количества агентов первой группы в течении нескольких десятков поколений.

Для определения золотых особей используется специальная метрика. Каждая особь, получившая максимальный возможный рейтинг в поколении, то есть сыгравшая все попытки на отлично, снова проводится через игру, но в данный раз не 2-5 раз, как в обучении, а 16 раз, со строго определёнными точками появления агента. Если после этого существо также получает максимальный возможный рейтинг, то причисляется к золотым, хотя такой подход и не доказывает, что конкретно данное существо решает задачу во всех возможных случаях, но учитывая то, что данная особь точно окажется в следующих поколениях и еще много раз пройдет через данную метрику, результат, посчитанный за несколько поколений, можно считать объективным.

### IV. ПРИМЕНЕННЫЕ МЕТОДЫ

Одной из проблем решения задач такого рода является выбор способа кодирования входного вектора. В эксперименте для кодирования были использованы: код Грея, двоичное кодирования и унитарный код. Из опробованных методов унитарный код показал наилучшую эффективность, что объясняется во многом его простотой для восприятия не только нейронной сетью, но и человеку, не знакомому с задачей.

Вторая основная проблема была в выборе оптимальной конфигурации нейронной сети. Практика показала, что конфигурации с большим количеством скрытых слоев и малым количеством нейронов в каждом слое справляется с задачей в общем случае точно так же, а зачастую и гораздо медленнее, чем конфигурация с единственным скрытым слоем и большим количеством нейронов в нем. Обе основные конфигурации по итогу пришли к одному и тому же конечному результату в решении задачи.

Основным предложенным способом улучшения работы эволюционных алгоритмов является способ дообучения. При таком подходе вначале решается простая задача. После успешного решения простой задачи золотые особи отбираются и попадают в первое поколение для решения более сложной задачи. Применение данного способа обучения в некоторых ситуациях дало более чем трехкратное увеличение скорости решения некоторых сложных задач, которые могли быть совершенно не решены, при использовании обучения «с нуля».

### ЗАКЛЮЧЕНИЕ

Эволюционные алгоритмы являются интересным методом решения различных задач по поиску стратегий, однако требуют очень много вычислительных мощностей. Предложенные методы дообучения поколений позволяют в несколько раз сократить время, необходимое для обучения агентов, однако приводят к повышению сложности самого алгоритма. Для дальнейшего улучшения работы эволюционных алгоритмов разумным шагом будет их комбинирование с другими методами машинного обучения для дообучения поколений и введения нелинейности в «мышление» агентов, а также достижения свойства «памяти».

### V. СПИСОК ЛИТЕРАТУРЫ

1. Нейронные сети для обработки информации /С. Осовский // Издательство: Финансы и статистика, 2004. – 334 с.
2. Нейронные сети, генетические алгоритмы и нечеткие системы /Д. Рутковская, М. Пилиньский, Л. Рутковский // Издательство: Горячая Линия - Телеком, 2007. – 452 с.