

ЭВОЛЮЦИОННЫЙ МЕТОД УПРАВЛЕНИЯ АГЕНТАМИ В МНОГОАГЕНТНОЙ СИСТЕМЕ

Ю. И. Давидюк¹

¹ Магистр технических наук, старший преподаватель кафедры интеллектуальных информационных технологий Брестского государственного технического университета, Брест, Беларусь, e-mail : yuidavidyuk@g.bstu.by

Реферат

Предлагается эволюционный метод обучения нейронной сети. Приведен пример генетического алгоритма с его параметрами, предложенный для обучения нейронной сети, контролирующей поведение агента в многоагентной системе. Представлены архитектура нейронной сети, генетический алгоритм с функцией приспособленности и тестовая среда с результатами тестирования различных функций активации нейронной сети.

Ключевые слова: генетический алгоритм, нейронная сеть, эволюционное программирование, многоагентные системы.

EVOLUTIONAL METHOD OF AGENT CONTROL IN MULTI AGENT SYSTEM

Yu. I. Davidyuk

Abstract

Evolutional method of training of the neural network is proposed. The sample of the genetic algorithm with its parameters proposed to train neural network which controls behavior of the agent in multi agent system. Architecture of the neural network, genetic algorithm with fitness function and test environment presented with results of testing for different activation functions of the neural network.

Keywords: neural network, genetic algorithm, evolutionary programming, multi-agent system.

Введение

В классическом понятии искусственного интеллекта решение поставленной задачи сводится к созданию одной интеллектуальной системы. Такая система, получая необходимые данные, обладая знаниями и ресурсами, может решить некоторую глобальную задачу, для которой она и была создана. Можно сказать, что у нас есть объект, который решает большую задачу самостоятельно. Назовем этот объект агентом.

В теории многоагентных систем, или мультиагентных систем, используется противоположный подход. Теперь у нас есть объект, агент, который не знает о глобальной задаче, а имеет всего лишь частичное представление о глобальной проблеме. То есть агент не видит состояние всей среды, а только то, что находится рядом с ним. Так общая задача может быть разделена на множество агентов, которые, в свою очередь, делят среду между собой. Появляется новая задача: как организовать работу всех агентов так, чтобы они эффективно взаимодействовали и находили оптимальное решение глобальной задач. Таким образом создается многоагентная система.

В многоагентных системах одну задачу можно разделить на части. Каждому агенту назначается роль, которой соответствует некоторое задание из общей задачи. Так каждый агент становится частью группы. Для организации процесса распределения задачи в многоагентных системах создается либо система распределенного решения проблемы, либо децентрализованный искусственный интеллект. В первом варианте производится процесс декомпозиции глобальной задачи. Далее управляющий центр собирает информацию о найденных решениях и производит их композицию. В таком способе организации многоагентная система проектируется строго сверху вниз, исходя из ролей, определенных для агентов, и результатов разбиения глобальной задачи на подзадачи. В случае использования децентрализованного искусственного ин-

теллекта, распределение заданий происходит в процессе взаимодействия агентов и носит больше спонтанный характер. Нередко это приводит к появлению в многоагентных системах резонансных, синергетических эффектов [1].

Проблема коллективного поведения и группового управления не является новой и исследуется уже в течение нескольких десятилетий [2–4]. Ее актуальность возрастает каждый год. В настоящее время проблема группового управления ставится уже по-новому, а для ее решения требуются создания новых технологий проектирования и программной реализации.

В последнее время все чаще применяется машинное обучение для решения определенного класса задач. Так, например, для решения оптимизационных задач пробуют использовать генетические алгоритмы, для задач классификации – нейронные сети. Для решения методами машинного обучения необходимо проанализировать задачу и найти наиболее эффективный алгоритм решения и способ настройки параметров для выбранного алгоритма.

Постановка задачи

Смоделируем среду для поиска оптимального поведения группы агентов в реальном времени. Агенты будут находиться в конкурентной борьбе. Среда описывается шириной и высотой. Агенты не могут выходить за границы среды. В среде находится некоторое количество точек, которые описываются координатами x , y , и меньшее количество агентов, которые должны найти и собрать все точки. Для ориентирования в среде и взаимодействия с другими агентами у агентов имеются «сенсоры». Через сенсоры агенту поступают данные о наличии точки поблизости, расстояние до этой точки, косинус угла между вектором направления агента и вектором, направленным на точку, сигнал о наличии конкурирующих агентов рядом (см. рис. 1).

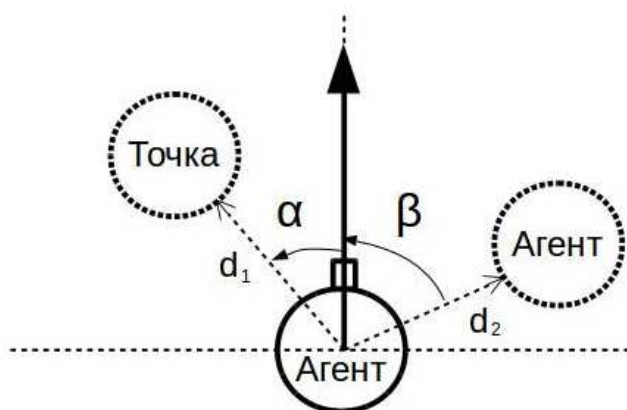


Рисунок 1 – Показатели сенсоров, обрабатываемые агентом

Каждый агент характеризуется положением в среде координатами x, y , скоростью и вектором направления движения. Агент взаимодействует со средой путем изменения собственного положения и направления (см. рис. 2).

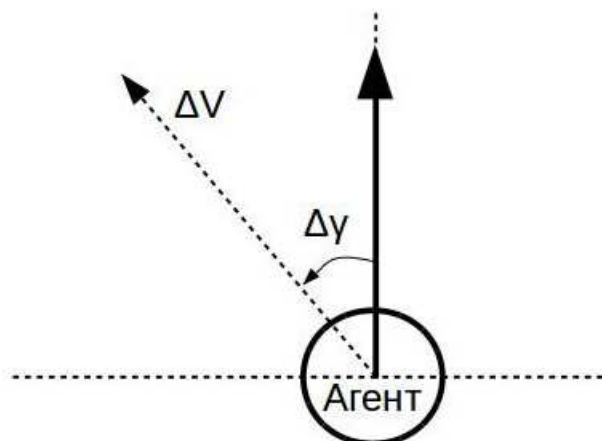


Рисунок 2 – Применение результатов работы нейронной сети к агенту

Задача считается решенной, если собраны все точки за кратчайшее время.

Для управления агентами будем использовать нейронную сеть. На вход распределительного слоя нейронной сети подаются показатели сенсоров, на выходе нейронной сети получаем значение угла поворота относительно текущего вектора движения агента, а также значение изменения скорости движения. Количество нейронов в скрытом слое можно подобрать экспериментально. Для решения поставленной задачи достаточно взять 4 нейрона на скрытом слое. Архитектура нейронной сети представлена на рисунке 3.

Выбор метода обучения нейронной сети

Классическим методом обучения данной нейронной сети является обучение с учителем с использованием алгоритма обратного распространения ошибки. Главным недостатком при использовании этого метода является отсутствие эталонных значений. Для того чтобы произвести обучение, необходимо сгенерировать обучающую выборку, для которой будут рассчитаны эталонные значения по изменению угла и скорости движения агента.

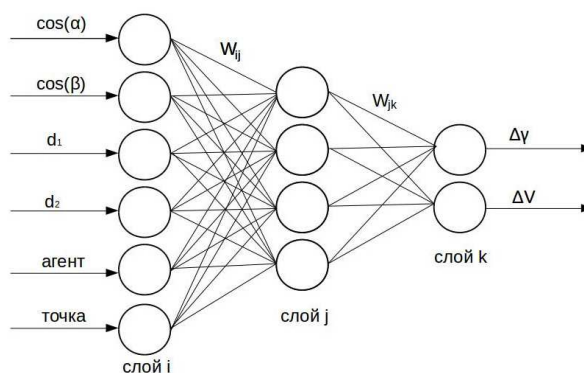


Рисунок 3 – Архитектура нейронной сети для управления агентами

Еще один из способов **машинного обучения**, в ходе которого испытываемая система (агент) обучается, взаимодействуя с некоторой средой. Откликом среды (а не специальной системы управления подкреплением, как это происходит в **обучении с учителем**) на принятые решения являются сигналы подкрепления, поэтому такое обучение является частным случаем **обучения с учителем**, но учителем является среда или её модель. Также нужно иметь в виду, что некоторые правила подкрепления базируются на неявных учителях, например, в случае **искусственной нейронной среды**, на одновременной активности формальных нейронов, из-за чего их можно отнести к обучению без учителя [5].

Для обучения нейронных сетей также можно применить генетические алгоритмы. Для решения задач, в которых сложно описать целевую функцию. Данный метод обучения рассмотрим подробнее.

Для удобства обучения с помощью генетического алгоритма необходимо нейронную сеть представить в линейном виде. Для этого все её параметры запишем в одномерный массив. Данный массив будет хромосомой. К параметрам нейронной сети относятся: весовые коэффициенты, пороговые значения, функция активации. Ген функции активации может отсутствовать, если мы не хотим менять ее во время обучения нейронной сети. Количество генов в хромосоме зависит от количества нейронных элементов в скрытом слое нейронной сети. Архитектуру нейронной сети представленной на рисунке 2 опишет хромосома с 34 генами.

Выбор родителей для скрещивания и мутации

Для выбора родителей для новой популяции будем использовать турнирный отбор. При турнирном отборе (tournament selection) из популяции, содержащей N особей, выбираются случайным образом t особей, и лучшая из них особь записывается в промежуточный массив. Эта операция повторяется N раз. Особи в полученном промежуточном массиве затем используются для скрещивания (также случайным образом). Размер группы строк, отбираемых для турнира, часто равен 2. В этом случае говорят о двоичном (парном) турнире. Вообще же t называют численностью турнира. Преимуществом данного способа является то, что он не требует дополнительных вычислений [6]. Для порождения новой популяции над хромосомами-родителями будем производить операции скрещивания и мутации.

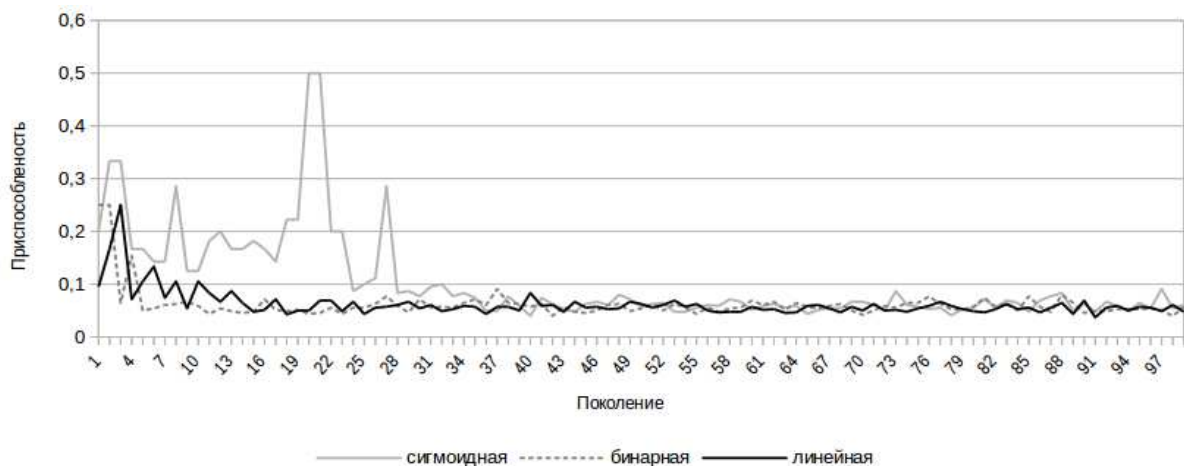


Рисунок 4 – Обучение нейронной сети с различными функциями активации

Сгенерируем модель со следующими параметрами: размеры среды 600 на 400, выбор хромосом-родителей будем проводить в среде с размерами 200 на 200 (для увеличения скорости обучения нейронной сети), количество агентов в среде – 15, количество частиц еды – 10. Произведем эксперимент с различными функциями активации. Результаты обучения отображены на рисунке 4.

Приемлемое поведение агентов происходит уже на 20 поколении, что достаточно быстро и совсем не нагружает вычислительную машину. От функции активации мало что зависит при большом количестве поколений. Если рассматривать оптимальный вариант, то стоит использовать линейную функцию активации.

Сгенерировать одинаковые условия не представляется возможным, в виду случайной инициализации значений генов у хромосом.

Заключение

В результате обучения нейронная сеть достаточно часто начинает менять поведения агента таким образом, что агент оглядывается по сторонам. Такое естественное поведение является эффективным, поскольку всегда есть вероятность появления точки ближе, чем текущая цель агента.

Для приближения смоделированной среды к реальным условиям можно использовать штрафные функции и ограничивать агентов в ресурсах.

Существуют и некоторые сложности: среда, в которой развиваются агенты, не является абсолютно просматриваемой, то есть в нашей задаче агент видит только перед собой и немного вокруг, а также среда является недетерминированной (координаты точек генерируются случайным образом).

Список цитированных источников

1. Абросимов, В. К. Групповое движение интеллектуальных летательных аппаратов в антагонистической среде / В. К. Абросимов – М. : Наука, 2013. – 250 с.
2. Варшавский, В. И. Коллективное поведение автоматов / В. И. Варшавский – М. : Наука, 1973. – 408 с.

3. Варшавский, В. И. Оркестр играет без дирижера / В. И. Варшавский, Д. А. Поспелов. – М. : Наука, 1984. – 208 с.
4. Цетлин, М. Л. Исследования по теории автоматов и моделированию биологических систем / М. Л. Цетлин. – М. : Наука, 1969. – 316 с.
5. Барто, Э. Обучение с подкреплением / Э. Барто, Р. Саттон. – М. : ДМК, 2020. – 552 с.
6. Рассел, Ст. Искусственный интеллект. Современный подход / Ст. Рассел, П. Норвиг. – М. : И. Д. Вильямс, 2017. – 1408 с.
7. Панченко, Т. В. Генетические алгоритмы : учебно-методическое пособие / Т. В. Панченко; под ред. Ю. Ю. Тарасевича. – Астрахань : Издательский дом «Астраханский университет», 2007. – 87 с.

References

1. Abrosimov, V. K. Gruppovoe dvizhenie intellektual'nyh letatel'nyh apparatov v antagonistskoj srede / V. K. Abrosimov – M. : Nauka, 2013. – 250 s.
2. Varshavskij, V. I. Kollektivnoe povedenie avtomatov / Varshavskij V.I. – M : Nauka, 1973. – 408 s.
3. Varshavskij, V. I. Orkestr igraet bez dirizhera / V. I. Varshavskij, D. A. Pospelov. – M : Nauka, 1984. – 208 s.
4. Cetlin, M. L. Issledovaniya po teorii avtomatov i modelirovaniyu biologicheskikh sistem / M. L. Cetlin. – M. : Nauka, 1969. – 316 s.
5. Barto, E. Obuchenie s podkrepleniem / E. Barto, R. Satton. – M. : DMK, 2020. – 552 s.
6. Rassel, St. Iskusstvennyj intellekt. Sovremennij pod-hod / St. Rassel, P. Norvig. – M. : I. D. Vil'yams, 2017. – 1408 s.
7. Panchenko, T. V. Geneticheskie algoritmy : uchebno-metodicheskoe posobie / T. V. Panchenko; pod red. YU. YU. Tarasevicha. – Astrahan' : Izdatel'skij dom «Astrahanskij universi-tet», 2007. – 87 s.

Материал поступил в редакцию 20.10.2020