(cc) BY

Жукович Ярослав Петрович
инженер, аспирант
Белорусский государственный университет
информатики и радиоэлектроники
г. Минск, Республика Беларусь

ГЕНЕТИЧЕСКИЕ АЛГОРИТМЫ

Аннотация: *в данной статье проводится анализ материалов по генетическим алгоритмам: рассматривается суть алгоритма, приводится подробное описание каждого этапа выполнения алгоритма, выполняется обзор различных модификаций генетических алгоритмов, в том числе параллельные модификации, а также определяются возможные направления дальнейших исследований.*

Ключевые слова: *генетические алгоритмы, модификации генетических алгоритмов, параллельные генетические алгоритмы.*

Генетические алгоритмы (ГА) относятся к числу универсальных методов оптимизации, позволяющих решать задачи различных типов (комбинаторные, общие задачи с ограничениями и без ограничений) и различной степени сложности. При этом ГА характеризуются возможностью как однокритериального, так и многокритериального поиска в большом пространстве, ландшафт которого является негладким.

В последние годы резко возросло число работ, прежде всего зарубежных ученых, посвященных развитию теории ГА и вопросам их практического использования. Результаты данных исследований показывают, в частности, что ГА могут получить более широкое распространение при интеграции с другими методами и технологиями. Появились работы, в которых доказывается эффективность интеграции ГА и методов теории нечеткости, а также нейронных вычислений и систем [1].

ГА относят к области мягких вычислений. Термин «мягкие вычисления» введен Лофти Заде в 1994 году. Это понятие объединяет такие области, как не-

Content is licensed under the Creative Commons Attribution 4.0 license (CC-BY 4.0)

четкая логика, нейронные сети, вероятностные рассуждения, сети доверия и эволюционные алгоритмы, которые дополняют друг друга и используются в различных комбинациях или самостоятельно для создания гибридных интеллектуальных систем.

Общее представление ГА. В общем случае ГА состоит из шести последовательных шагов, два их которых являются подготовительными, а остальные представляют собой непосредственно сам алгоритм.

Шаг 0. Подготовка данных.

Шаг 1. Создание начальной популяции.

Шаг 2. Оценка приспособленности.

Шаг 3. Оператор отбора (селекция и выбор родителей).

Шаг 4. Оператор скрещивания (рекомбинация).

Шаг 5. Оператор мутации.

Шаг 6. Повтор шагов 2–5 до достижения критерия сходимости.

Шаг 0. Подготовка данных. Задачу необходимо формализовать таким образом, чтобы ее решение могло быть закодировано в виде вектора (*хромосомы*) *генов*, где каждый ген может быть битом, числом или неким другим объектом. В классических реализациях ГА предполагается, что хромосомы имеют фиксированную длину, однако существуют вариации ГА, в которых это ограничение отсутствует.

Самым популярным методом кодирования хромосом является двоичное кодирование, то есть *аллели* всех генов в хромосоме равны 0 или 1. Для большинства функций ГА будут работать лучше при кодировании параметров кодом Грея, а не прямым бинарным кодом. Это связано с тем, что расстояние Хэмминга между битовыми представлениями данных может не отражать близость в привычном смысле. Код Грея характеризуется тем, что двоичные последовательности, соответствующие двум последовательным целым числам, отличаются только одним битом.

Другим вариантом кодирования служит логарифмическое кодирование, которое применяется в ГА для уменьшения длины хромосом. Оно используется,

Содержимое доступно по лицензии Creative Commons Attribution 4.0 license (CC-BY 4.0)

главным образом, в задачах многомерной оптимизации с большими пространствами поиска решений.

Шаг 1. Создание начальной популяции. На данном шаге некоторым образом, обычно случайным, создается начальная популяция.

Шаг 2. Оценка приспособленности текущей популяции. На данном шаге выполняется вычисление фитнесс-функции для каждой хромосомы. Фитнесс-функция – функция пригодности, которая определяет меру точности решения или меру удовлетворения решению задачи. Определение соответствующей фитнесс-функции является решающим для корректной работы ГА. В частности, вид фитнесс-функции может зависеть от накладываемых ограничений при решении оптимизационных задач.

Шаг 3. Оператор отбора (*селекции*). В различных модификациях ГА могут использоваться разные методы *селекции*. Чаще всего селекция состоит в том, что родителями могут стать только те особи, значение приспособленности которых не меньше пороговой величины, например, среднего значения приспособленности по популяции. Некоторые из наиболее распространенных методов селекции: *отбор усечением*, *элитарный отбор*, *отбор вытеснением*, *метод Больцмана* [2].

По завершению селекции, из оставшихся особей, необходимо выбрать пары для проведения скрещивания на следующем шаге. Наиболее распространенными операторами выбора родителей являются следующие.

Панмиксия – самый простой оператор отбора. В соответствии с ним каждому члену популяции сопоставляется случайное целое число на отрезке [1, N], где N – количество хромосом в популяции.

Инбридинг – представляет собой такой метод, когда первый родитель выбирается случайным образом, а вторым родителем является член популяции ближайший к первому.

Аутбридинг – также используют понятие схожести хромосом, однако, теперь скрещиваемые пары формируют из максимально далеких особей.

Content is licensed under the Creative Commons Attribution 4.0 license (CC-BY 4.0)

Шаг 5. Оператор скрещивания (*рекомбинация*). Оператор скрещивания применяется сразу же после оператора отбора для получения новых особей-потомков. Смысл скрещивания заключается в том, что созданные потомки должны наследовать генную информацию от обоих родителей.

В зависимости от способа кодирования хромосом выделяют дискретную и бинарную рекомбинацию (кроссинговер).

Дискретная рекомбинация (discrete recombination) в основном применяется к хромосомам с вещественными генами. Основными способами дискретной рекомбинации являются собственно дискретная рекомбинация, промежуточная, линейная и расширенно линейная рекомбинации.

Рекомбинацию бинарных строк принято называть *кроссинговером* или скрещиванием. Кроссинговер также разделяют на *одноточечный* и *двухточечный* (или *n-точечный*, в общем виде). При одноточечном кроссинговере родительские хромосомы разделяются только в одной случайной точке. В двухточечном кроссинговере точек разрыва 2, а n-точечный кроссинговер является своеобразным обобщением одно- и двухточечного кроссинговеров для $n > 2$.

Кроме описанных типов кроссинговера есть ещё *однородный кроссинговер*. Его особенность заключается в том, что значение каждого бита в хромосоме потомка определяется случайным образом из соответствующих битов родителей. Для этого вводится некоторая величина $0 < p0 < 1$, и если случайное число больше p0, то на n-ю позицию первого потомка попадает n-й бит первого родителя, а на n-ю позицию второго – n-й бит второго родителя. В противном случае к первому потомку попадает бит второго родителя, а ко второму – первого. Такая операция проводится для всех битов хромосомы.

Среди других типов кроссинговера можно упомянуть *триадный кроссинговер*, *перетасовочный кроссинговер* и *кроссинговер с уменьшением замены* [2].

Шаг 6. Оператор мутации. Оператор мутации (mutation operator) необходим для «выбивания» популяции из локального экстремума и препятствует преждевременной сходимости. В общем случае это достигается за счет того, что

Содержимое доступно по лицензии Creative Commons Attribution 4.0 license (CC-BY 4.0)

изменяется случайно выбранный ген в хромосоме. Однако, так же как и кроссинговер, мутация может проводиться не только по одной случайной точке. Можно выбирать некоторое количество точек в хромосоме для инверсии, причем их число также может быть случайным. Также можно инвертировать сразу некоторую группу подряд идущих точек. Вероятность мутации значительно меньше вероятности кроссинговера и редко превышает 1%. Среди рекомендаций по выбору вероятности мутации нередко можно встретить варианты 1/L или 1/N, где L – длина хромосомы, N – размер популяции.

Шаг 7. Повторение шагов 2–5, до достижения критерия окончания. Процесс эволюции, вообще говоря, может продолжаться до бесконечности, поэтому необходимо задание критерия останова, в качестве которого может служить заданное количество поколений или *схождение (convergence)* популяции. Схождением называется состояние популяции, когда все хромосомы популяции находятся в области некоторого экстремума и почти одинаковы. То есть кроссинговер практически никак не изменяет популяции, а мутирующие особи склонны вымирать, так как менее приспособлены. Таким образом, схождение популяции означает, что достигнуто решение близкое к оптимальному. Итоговым решением задачи может служить наиболее приспособленная особь последнего поколения.

Модификации ГА. В настоящее время ГА представляют собой целый класс алгоритмов, направленных на решение разнообразных задач. Как показывает анализ, модификации ГА отличаются, прежде всего, способом селекции хромосом. В основных модификациях ГА несколько способов селекции используется для достижения различных целей – упрощения формирования промежуточной популяции, распараллеливания работы алгоритма, возможности анализа и предсказания поведения ГА. Далее кратко рассмотрены некоторые из существующих модификаций.

Модификация под названием *генетический алгоритм с устойчивым состоянием (Steady State GA, SSGA)* затронула способ формирования промежуточной популяции, являющейся результатом отбора для формирования наследников с

Content is licensed under the Creative Commons Attribution 4.0 license (CC-BY 4.0)

помощью генетических операторов. SSGA не формирует промежуточную популяцию как классический ГА, а осуществляет последовательно выбор пары наилучших хромосом, применяя к ним генетические операторы, с целью формирования наследников, которые заменяют худшие хромосомы популяции.

В модели *Генитор (Genitor)* используется специфический способ отбора. На этапе отбора случайным образом выбираются две хромосомы и скрещиваются. В результате получается только один потомок, который оценивается и занимает место менее приспособленной хромосомы в популяции. После этого случайным образом выбираются две особи, потомок которых занимает место родительской хромосомы с более низкой приспособленностью. Процесс продолжается до тех пор, пока значения приспособленности всех хромосом не станут одинаковыми.

Среди других модификаций следует упомянуть: *метод прерывистого равновесия, гибридный алгоритм, «Cross generational elitist selection, Heterogenous recombination, Cataclysmic mutation» (CHC), ГА с нефиксированным размером популяции*. Более подробное описание этих модификаций изложено в работах [2; 3].

Параллельные генетические алгоритмы (*ПГА*). Суть распараллеливания любой программы заключается в разделении ее на части, которые затем можно выполнять на нескольких вычислительных ресурсах. Данный подход может быть применен и к ГА. Литература содержит множество примеров успешной параллельной реализации. Некоторые параллельные методы используют одну популяцию, другие разделяют основную популяцию на несколько относительно независимых подпопуляций. Некоторые методы лучше подходят для выполнения на одном многопроцессорном вычислительном узле, другие хорошо адаптированы для выполнения на кластере из многих вычислительных узлов.

Параллелизация на базе глобальной популяции. *Глобальная параллелизация мастер-рабочий (global master-slave parallelization)*. Вычислительные узлы разделяются на один мастер-узел и множество узлов-рабочих. Основная часть алгоритма выполняется на узлах-рабочих, которые отвечают за воспроизводство, мутацию и вычисление приспособленности. Все результаты работы узлов-рабочих

Содержимое доступно по лицензии Creative Commons Attribution 4.0 license (CC-BY 4.0)

пересылаются на мастер-узел, который проводит отбор особей в новую популяцию. Отобранные особи затем рассылаются на узлы-рабочие.

Мелкозернистая параллелизация (fine-grained algorithms) использует только одну популяцию, распределенную по некоторой пространственной структуре, которая ограничивает взаимодействие между индивидами. Индивиды могут соревноваться и скрещиваться только с их соседями, но так как соседние группы перекрываются, то успешные решения могут распространяться по всей популяции.

Параллелизация на базе подпопуляций. *Островная параллелизация* подразумевает использование нескольких относительно больших подпопуляций, развитие которых происходит независимо друг от друга с последующей *миграцией* особей между подпопуляциями.

Отдельные подпопуляции в данном типе параллелизации можно условно принять за вершины графа. В связи с этим можно рассматривать топологию графа миграционного ГА.

Наиболее распространенной топологией миграции является полный граф, при которой особи из любой подпопуляций могут мигрировать в любую другую подпопуляцию. Для каждой подпопуляции полное количество потенциальных иммигрантов строится на основе всех подпопуляций. Мигрирующая особь случайным образом выбирается из этого общего числа.

Другая основная миграционная схема – это топология кольца. Здесь особи передаются между соседними (по направлению обхода) подпопуляциями. Таким образом, особи из одной подпопуляции могут мигрировать только в одну – соседнюю подпопуляцию.

Иерархические параллельные ГА. Во многих работах зарубежных исследователей можно встретить подход, при котором комбинируются несколько методов распараллеливания ГА. Такой подход можно назвать иерархические *параллельные ГА (hierarchical parallel GAs)*. В основном комбинируется два метода: например, на верхнем уровне используется глобальная параллелизация мастер-

Content is licensed under the Creative Commons Attribution 4.0 license (CC-BY 4.0)

рабочий, а на локальных уровнях мелкозернистая параллелизация. Более подробные материалы по этой теме можно найти в работе [4].

Кобминирование ГА с другими методами. Наиболее распространенным является использование ГА совместно с алгоритмом *k-ближних соседей (kNN)* и *нейронными сетями (НС)*. Подробнее в работах [5; 6].

Объединение генетических алгоритмов и нейронных сетей (НС) известно в литературе под аббревиатурой *COGANN (Combinations of Genetic Algorithms and Neural Networks)*. Такое объединение может быть вспомогательным или равноправным. Вспомогательное объединение двух методов означает, что они применяются последовательно один за другим, при этом один из них служит для подготовки данных для другого. При равноправном объединении оба метода применяются одновременно.

Заключение. Данная статья написана в рамках выполнения аналитического обзора литературных источников по диссертации на тему «Анализ и разработка методов «интеллектуального анализа данных» в распределенных системах». Так как тема дессертации является обширной и сочетает в себе широкий математический инструментарий (от классического статистического анализа до новых кибернетических методов) и последних достижений в сфере информационных технологий, то было принято решение ограничить обзор в данной статье анализом материалов по генетическим алгоритмам.

В рамках проведенного аналитического обзора по ГА определены возможные дальнейшие направления исследований, среди которых можно отметить: распараллеливание ГА и объединение ГА с другими методовами data mining. Несмотря на множество работ по данным направлениям остается большое количество вопросов, которые требуют дальнейшей теоретической и практической проработки. С течением времени появляются новые направления, в которых использование ГА может быть оправдано. Также разрабатываются новые методы и новые подходы к существующим методам в области data mining, которые могут дать лучшие результаты при объединении с ГА.

Содержимое доступно по лицензии Creative Commons Attribution 4.0 license (CC-BY 4.0)

Список литературы

1. Барсегян А.А. Технологии анализа данных: Data Mining, Visual Mining, Text Mining, OLAP / А.А. Барсегян, М.С. Куприянов, В.В. Степаненко, И.И. Холод. – 2-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2007. – 384 с.

2. Паначенко Т.В. Генетические алгоритмы [Текст]: Учебно-методическое пособие / Под ред. Ю.Ю. Тарасевича. – Астрахань: Издательский дом «Астраханский университет», 2007. – 87 с.

3. Darrel Whitley: A Genetic Algorithm Tutorial; November 10, 1993; Technical Report CS-93–103 (Revised); Department of Computer Science, Colorado State University, Fort Collins, US.

4. Erick Cantu-Paz: A survey of parallel genetic algorithms, IllGAL Report 97003, The University of Illinois, 1997.

5. Рутковская Д. Нейронные сети, генетические алгоритмы и нечеткие системы / Д. Рутковская, М. Пилиньский, Л. Рутковский; пер. с польск. И.Д. Рудинского. – М.: Горячая линия – Телеком, 2006. – 452 с.

6. A Novel Hybrid Classification Model of Genetic Algorithms, Modified k-Nearest Neighbor and Developed Backpropagation Neural Network [Электронный ресурс] / Nader Salari, Shamarina Shohaimi, Farid Najafi – National Center of Biotechnology Information, 2014 [Электронный ресурс]. – Режим доступа: https://www.ncbi.nlm.nih.gov/pmc/ articles/PMC4242540/ (дата обращения: 27.09.2016).

Content is licensed under the Creative Commons Attribution 4.0 license (CC-BY 4.0)