УДК 519.7

ЭВОЛЮЦИОННЫЕ МЕТОДЫ ОПТИМИЗАЦИИ

Полякова А.С.
научный руководитель доктор техн. наук Семёнкин Е. С.
Сибирский Государственный Аэрокосмический Университет имени академика М.Ф. Решетнева, Россия, Красноярск

При решении задач оптимизации сложных систем часто встречаются ситуации, которые затрудняют или делают невозможным применение классических методов, такие как неопределенность, разношкальность, вычислительная сложность, высокая размерность и многоэкстремальность. Эволюционные алгоритмы способны преодолевать многие из возникающих трудностей. В частности, генетические алгоритмы реализуют прямой поиск, тем самым исключают проблему неопределенности, переменные решаемой задачи в генетических алгоритмах кодируются бинарными строками - решается проблема разношкальности. Как показывает практика применения эволюционных алгоритмов, высокая размерность задач, нелинейность и многоэкстремальность не создают для них дополнительных проблем.

Генетические алгоритмы (ГА) являются направлением более общей теории эволюционных алгоритмов, основанной на следующем принципе: «каждый биологический вид целенаправленно развивается и изменяется для того, чтобы наилучшим образом приспособиться к окружающей среде». На данный момент генетический алгоритм является одним из наиболее исследуемых и развивающихся алгоритмов глобальной оптимизации.

Один из недостатков классических эволюционных алгоритмов состоит в отсутствии механизма учета ограничений оптимизационной задачи. Но в практических задачах часто приходится иметь дело с ограничениями, наложенных на значения переменных. Поэтому представляет интерес изучение и сравнение методов учета этих ограничений, используемых в ГА.

Пусть решается следующая задача условной однокритериальной оптимизации:

$$f(x) \to extr$$

$$\begin{cases} g_j(x) \le 0,\ j = \overline{1,r} \\ h_j(x) = 0,\ j = \overline{r+1,m} \end{cases}$$

Одним из наиболее распространенных подходов к учету ограничений является метод штрафных функций, основная идея которого заключается в том, что пригодность индивида вычисляется не только в зависимости от соответствующего ему значения целевой функции, но и от меры нарушения ограничений:

$$fitness(x) = f(x) + \delta \cdot \lambda(t) \cdot \sum_{j=1}^{m} f_j^{\beta}(x)$$

Штрафные функции вычисляются по формуле:

$$f_j(x) = \begin{cases} \max\{0, g_j(x)\},\ j = \overline{1,r} \\ \left|h_j(x),\ j = \overline{r+1,m}\right| \end{cases}$$

Существуют следующие штрафные методы: метод «смертельных» штрафов, метод динамических штрафов, метод адаптивных штрафов, а также обсуждены гибридные методы, использующие механизм «лечения».

После детального анализа каждого метода было решено использовать метод динамических штрафов, метод смертельных штрафов и смертельных с лечением. В этой работе был проведен их сравнительный анализ, на множестве задач однокритериальной условной оптимизации. Целевые функции и ограничения в задачах представляют собой линейные и нелинейные функции нескольких переменных. Для исследования эффективности и работоспособности программной системы использовались тестовые задачи с ограничениями в виде равенств и неравенств.

В результате исследований было выяснено, какой из методов для учета ограничений является наиболее эффективным для выбранного множества задач. Для оценки эффективности алгоритма используем показатель надежности. Надежность – процент успешных запусков (решение найдено) к общему числу запусков алгоритма.

По результатам тестов алгоритма сделаны выводы о наилучших и наихудших настройках ГА для каждого метода штрафных функций.

На практике, реальные задачи являются многокритериальными, они не редко встречаются в практической деятельности людей и часто такие задачи имеют противоречивые критерии. Нахождение оптимального решения является сложной задачей и требует использования специализированных методов. Такими методами могут служить эволюционные алгоритмы, хорошо зарекомендовавшие себя в задачах однокритериального поиска.

В самом общем виде задача условной многокритериальной оптимизации включает набор из N параметров (переменных), множество K целевых функций от этих переменных и множество M ограничений. При решении многокритериальной задачи необходимо найти оптимум по совокупности K критериев.

В наиболее общей постановке задачи условной многокритериальной оптимизации от функций не требуется никаких дополнительных свойств, удобных с точки зрения оптимизации (выпуклость, дифференцируемость и т.д.). Функции могут быть заданы алгоритмически, а переменные могут быть непрерывными, дискретными, бинарными и даже смешанными.

Для проведения исследования была сделана программная реализация алгоритмов SPEA (Strength Pareto Evolutionary Algorithm), SPEA2 и NSGA-II (Non-dominated Sorting Genetic Algorithm). Сравнение алгоритмов проводилось на тестовых задачах для выявления надежности алгоритма.

Эффективность работы алгоритмов сравнивалась с использованием 3 метрик: *S, D, GD.*

$$GD = \frac{\sqrt{\sum_{i=1}^{n} d_i^2}}{n} \qquad S = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(\bar{d} - d_i)^2} \qquad D = \sqrt{\sum_{m=1}^{M}(\max_{i=1}^{d} f_m^i - \min_{i=1}^{d} f_m^i)^2}$$

Используемые метрики позволяют оценить равномерность распределения полученного недоминируемого фронта, разброс полученных векторов и близость полученного фронта к известному фронту Парето. Таким образом, использованные вместе, эти метрики дают объективную оценку качеству полученного решения, а, следовательно, позволяют сравнивать различные алгоритмы.

Нейронная сеть дает возможность построить модель зависимости выходных переменных от входных, не имея никаких сведений об истинной структуре зависимости. Использование генетических алгоритмов для настройки весов и её структуры ИНС называется нейроэволюцией или нейрогенезисом (Д.Уитли, 1995).

Применяя для обучения и настройки структуры сети генетический алгоритм, важной особенностью которого является способность к адаптации, можно получить самоорганизующуюся систему с достаточно большой степенью универсальности. Для работы этой системе будут необходимы только данные из внешнего окружения, причем их тип и отношение к решаемой задаче не играют большой роли. Таким образом, возможно решение поставленной проблемы о самоорганизующейся системе, способной решать широкий спектр задач.

Пусть $\overline{X} = (x_1, x_2, ...x_N)$ – вектор независимых (входных) переменных нейронной сети, $\overline{O} = (o_1, o_2, ...o_N)$ – вектор выходных переменных сети. Сеть состоит из $(T-2)$

скрытых слоев, одного входного слоя и выходного. В этой работе была создана нейронная сеть, в которой каждый нейрон одного слоя связан с каждым нейроном соседнего слоя. Нейроны слоев, не являющихся соседними, не связаны напрямую, т.е. выход одного нейрона не является входом другого. Скрытые слои, и выходной слой являются обрабатывающими. Входной слой не является таковым.

Выход каждого нейрона обрабатывающего слоя представляет собой некоторую функцию, называемую активационной, от взвешенной суммы выходов нейронов с предыдущего слоя, как показано на рисунке 2.

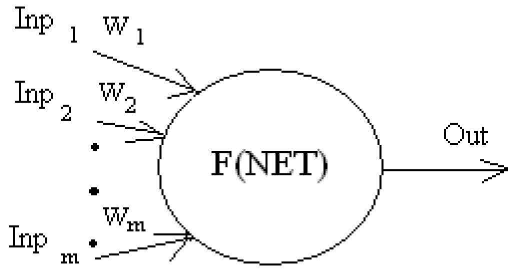

Рис. 1 Выход нейрона обрабатывающего слоя

Здесь m – число нейронов предыдущего слоя. Взвешенная сумма равна $\mathrm{NET} = \sum_{i=1}^{m} \mathrm{Inp}_i \cdot \mathrm{W}_i$. – это сумма произведений выходов нейронов с предыдущего слоя, умноженных на соответствующие веса. В работе была использована следующая активационная функция: $\mathrm{OUT} = \frac{1}{1+e^{-\mathrm{NET}}} = F(\mathrm{NET})$ (логистическая или «сигмоидальная» (S-образная) функция). Сигнал NET преобразуется активационной функцией F и дает выходной нейронный сигнал OUT.

В ходе работы реализована полносвязная нейронная сеть с произвольным числом скрытых слоев и нейронов на них. Нейроны на выходном и скрытых слоях имеют нейронные смещения (+1) .

Сеть обучается с помощью алгоритмов обратного распространения и обратного распространения с учетом предыдущего шага. Начальные веса могут быть выбраны случайно или с помощью генетического алгоритма.

Коэффициент ошибки для выходного слоя:

$$\delta_i^{(K+1)} = (D_i - Y_i)Y_i(1 - Y_i)$$

Коэффициент ошибки для остальных слоев:

$$\delta_i^{(k)} = Y_i^{(k)}\left(1 - Y_i^{(k)}\right)\sum_j \delta_j^{(k+1)} w_{ji}^{(k+1)} \quad (k = 1, \dots, K)$$

Корректировка весов осуществляется по формуле:

$$w_{ij}^{(k)}(t+1) = w_{ij}^{(k)}(t) + \eta\delta_i^{(k)}Y_j^{(k-1)} \ \ (1, \dots, K+1)$$

Использование ГА для формирования начальных весов помогает значительно улучшить качество получаемой модели. Однако это не гарантирует нахождение идеальных весов в связи с тем, что область поиска весов ограничена, и тем, что необходимо выбрать эффективные настройки и для самого алгоритма, что не является тривиальной задачей.

Было проведено исследование работы нейронной сети с применением настройки генетическим алгоритмом на различных тестовых функциях, так же было доказано, что применение Генетического алгоритма к настройке структуры и параметров сети оправдано и значительно улучшает результат построения такой модели даже по незначительной выборке.

Все алгоритмы тестировались с несколькими различными наборами параметров (размер популяции, точность, число поколений, типы скрещивания и селекции, вероятность мутации), каждый алгоритм независимо запускался с различными параметрами 50 раз. Полученные данные усреднялись для дальнейшего сравнения. Результаты и выводы будут представлены во время презентации.

Библиографические список

1. Ворожейкин А.Ю. Вероятностный генетический алгоритм решения задач условной оптимизации / А.Ю. Ворожейкин, Е.С. Семенкин // Компьютерные учебные программы и инновации №3, 2007.
2. Семенкин Е.С., Жукова М.Н., Жуков В.Г., Панфилов И.А., Тынченко В.В. Эволюционные методы моделирования и оптимизации сложных систем //Конспект лекций, СФУ, Красноярск, 2007.
3. Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach / Zitzler E., Thiele L.
4. Homogeneous Particle Swarm Optimizer for Multi-objective Optimization Problem / Seok K. Hwang, Kyungmo Koo, Jin S. Lee / 2005г. – 7 стр.