

- дународной научно-методической конференции «Университетское образование» (МКУО-2001). – Пенза, 2001, Часть 1, С.70.
6. *Васильев В.А.* Особенности оптических спектральных методов формирования информационных ресурсов твёрдотельных структур. В кн: XXVII Международная конференция «Информационные технологии в науке, образовании, телекоммуникации и бизнесе». Труды конференции. Ялта, Гурзуф, 2001, с.43-45.
 7. *Зингер Г.М., Ипатов И.П., Рыскин А.И.* Оптические свойства четверных твёрдых растворов на основе соединений A^3B^5 в области решёточных и плазменных колебаний. // ФТП.-1984.-Т.18, вып.1.-С.24-42.
 8. *Якобсон Р.-В* кн.: Физика тонких плёнок. Т.8/Под ред. Г. Хасса, М.Ф ранкомба и Р.Гофмана.-М.:Мир,1967.-С.91-151.

УДК 658.512:519.7:007.52

С.И. Родзин

ФОРМЫ РЕАЛИЗАЦИИ И ГРАНИЦЫ ПРИМЕНЕНИЯ ЭВОЛЮЦИОННЫХ АЛГОРИТМОВ

В настоящее время значение теории эволюции и самоорганизации не ограничивается только биологией. Идеи эволюции и самоорганизации привлекают внимание философов и психологов, специалистов в области организационного управления и финансового менеджмента, математиков и инженеров [1,2]. Принято различать четыре главных направления в моделировании эволюции: *генетические алгоритмы* (ГА), *генетическое программирование* (ГП), *эволюционные стратегии* (ЭС) и *эволюционное программирование* (ЭП), – которые объединяют одним общим термином – *эволюционные алгоритмы* (*Evolutionary computation*) (ЭА).

Высокий потенциал, которым обладают различные формы ЭА для решения трудных оптимизационных задач, известен давно. Диверсификация ЭА привела к значительному разнообразию их форм. Актуальной становится проблема сравнительного анализа различных форм ЭА, систематизации их особенностей и выработки рекомендаций по применению ЭА для решения оптимизационных задач.

Различные формы ЭА имитируют эволюционный процесс на различном уровне абстракции. ГА и ГП делают акцент на генетический механизм наследственности на уровне хромосом. ЭС, напротив, рассматривают ход эволюции на уровне фенотипа. Наконец, алгоритмы ЭП анализируют эволюцию на уровне популяции в целом. Далее, один стринг (хромосома) в ГА кодирует одно решение, т.е. пропорция между генотипом и фенотипом равна 1:1, что соответствует в биологическом смысле одному гаплоиду (особь с одинарным набором непарных хромосом). Между тем открытия в генетике, в частности, так называемые интроны (гены, не содержащие генетической информации), свидетельствуют о некоторой ограниченности подобного рода подхода к моделированию эволюции. В алгоритмах ЭС и ЭП оператор мутации применяется с частотой, соответствующей нормальному закону распределения, что является оправданным для выбранного уровня моделирования, в отличие от мутации на геномном уровне, где она проявляется как редкое событие и практически не влияет на фенотип. В ГА вероятность мутации отдельных генов и хромосом (инверсия) близка к природной. В алгоритмах ЭП мутация вообще является единственным оператором поиска. Оператор дискретной рекомбинации в ЭС со-

ответствует природному механизму доминирования родительских хромосом, в отличие от так называемой промежуточной рекомбинации, которая аналогична природной передаче некоторых признаков по наследству, что соответствует некоторому компромиссу между родительскими хромосомами. Наконец, оператор кроссинговера, который не применяется в алгоритмах ЭП, является основным алгоритмом для ГА и ГП.

Эти и многие другие методологические различия между разными формами ЭА тем не менее позволяют говорить о базовых постулатах, таких как *универсальность и фундаментальность*, присущих эволюции независимо от формы и уровня абстракции модели. Указанная общность может быть выражена в виде следующей схемы *абстрактного ЭА* [3]:

1. Установка параметров эволюции;
2. Инициализация начальной популяции $P(0)$;
3. $t:=0$;
4. Оценка решений, входящих в популяцию;
5. $t:=t+1$;
6. Селекция (отбор);
7. Репликация (повторение, копирование, аутоинтез);
8. Вариация (видоизменение);
9. Оценка решений-потомков;
10. Образование новой популяции $P(t)$;
11. Выполнение алгоритма до тех пор, пока параметр t не достигнет заданного значения t_{\max} либо не будут выполнены другие условия останова.
12. Вывод результатов и останов.

Контроверзу Ч. Дарвина (1859 г.) о том, что в ходе естественного отбора побеждает сильнейший, необходимо дополнить: материал для эволюции дают побежденные. Между тем, в ЭА чаще всего применяются статические *fitness*-функции, что не позволяет учесть механизм коэволюции в живой природе. Популяция в природе, как правило, состоит из отдельных, более или менее изолированных локальных подпопуляций, допускающих миграцию отдельных индивидуумов. Большинство ЭА это обстоятельство игнорируется, алгоритмы работают с одной большой неструктурированной популяцией.

Тем не менее решающим обстоятельством для оценки практической пригодности ЭА является прежде всего то, насколько успешно с помощью ЭА может быть решена та или иная оптимизационная задача с точки зрения качества решения и вычислительной сложности.

Результаты сравнительного анализа ГА, ГП, ЭС и ЭП представлены в таблице.

В [4] была предпринята попытка сравнительного анализа конкурирующих алгоритмов оптимизации с точки зрения их результативности. Пусть A_1 и A_2 – конкурирующие алгоритмы, F – целевая функция оптимизации, H – гистограмма значений F , m – количество полученных решений. Обозначим через $p(H|F,m,A)$ вероятность получения с помощью A алгоритма поиска m различных решений, имеющих вид гистограммы H . Вольперт и Макрид сформулировали и доказали следующую теорему *NFL (No-Free-Lunch)*.

Таблица

Критерии сравнения Виды ЭА	Представле- ние решения	Целевая функция	Селекция	Мутация	Рекомбина- ция	Самоадаптив- ность
ГА	Двоичное, вещественное	Скаляр	Стохастическая, недискриминационная	Вспомогательный оператор	Основной оператор поиска	Малоисследована
ГП	Программное	Скаляр	Стохастическая, недискриминационная	Применяется неоднозначно	Основной оператор поиска	Малоисследована
ЭС	Вещественное	Вектор	Детерминированная, дискриминационная	Основной оператор поиска	Важна для самоадаптивности	Средне-квадратичное отклонение
ЭП	Вещественное	Вектор	Детерминированная, дискриминационная	Единственный оператор поиска	Не применяется	Средне-квадратичное отклонение

Теорема NFL. Для любой пары эвристических алгоритмов поисковой оптимизации (A_1 и A_2) справедливо следующее соотношение:

$$\sum_F p(H|F, m, A_1) = \sum_F H|F, m, A_2).$$

Иными словами, все алгоритмы поиска оптимума целевой функции имеют одинаковую результативность, если сравнение проводится по всем возможным представлениям F . Таким образом, если алгоритм A_1 при оптимизации некоторой целевой функции позволяет получить лучшие результаты, нежели алгоритм A_2 , то существуют оптимизационные задачи с другой целевой функцией, для которых лучшие результаты дает алгоритм A_2 . Несмотря на очевидную ограниченность *NFL*-теоремы (не учитывается время решения оптимизационной задачи конкурирующими алгоритмами), она позволяет подходить к оценке алгоритмов оптимизации с *единых методологических позиций* и имеет практически важные следствия:

1. Поиск эвристического алгоритма оптимизации, который превосходит все конкурирующие с ним алгоритмы, не имеет смысла без точного описания конкретных задач и целевых функций, при оптимизации которых разрабатываемый алгоритм имеет преимущество перед другими алгоритмами. Нельзя рассчитывать найти один алгоритм, который будет результативнее остальных для любых целевых функций оптимизации.
2. Чтобы найти хорошее решение для заданного класса задач оптимизации, необходимо вначале идентифицировать характеристические особенности класса задач и лишь затем на их основе искать подходящий алгоритм (примером здесь являются задачи линейного программирования и симплекс-метод).

На наш взгляд с этой точки зрения ЭА обладают следующими достоинствами и недостатками.

Достоинства ЭА:

- ◆ Широкая область применения [5,6].
- ◆ Возможность проблемно-ориентированного кодирования решений, подбора начальной популяции, комбинирования ЭА с неэволюционными алгоритмами, продолжения процесса эволюции до тех пор, пока имеются необходимые ресурсы.
- ◆ Пригодность для поиска в сложном пространстве решений большой размерности.
- ◆ Отсутствие ограничений на вид целевой функции.
- ◆ Ясность схемы и базовых принципов ЭА.
- ◆ Интегрируемость ЭА с другими неклассическими парадигмами искусственного интеллекта, такими как искусственные нейросети и нечеткая логика [3].

Недостатки ЭА:

- ◆ Эвристический характер ЭА не гарантирует оптимальности полученного решения (правда, на практике, зачастую, важно за заданное время получить одно или несколько субоптимальных альтернативных решений, тем более, что исходные данные в задаче могут динамически меняться, быть неточными или неполными).
- ◆ Относительно высокая вычислительная трудоемкость, причинами которой является то обстоятельство, что в ходе моделирования эволюции многие решения отбрасываются как неперспективные (тем не менее многочисленные экспериментальные данные указывают на то, что временная сложность ЭА, в среднем, ниже, чем у лучших конкурирующих алгоритмов, но не более, чем на один порядок).
- ◆ Невысокая эффективность ЭА на заключительных фазах моделирования эволюции (в особенности это касается ГА); это объясняется тем, что операторы поиска в ЭА не ориентированы на быстрое попадание в локальный оптимум.
- ◆ Нерешенными остаются вопросы самоадаптации ЭА.

Преодоление перечисленных выше трудностей тесно связано с понятием *параллелизма* ЭА [7], что позволяет реалистично моделировать эволюцию с помощью параллельных вычислительных систем и использовать для эволюции популяции большой размерности.

Параллелизм ЭА. Различают следующие типы параллелизма ЭА[8]:

- ◆ Глобальный параллелизм.
- ◆ Миграционная модель.
- ◆ Диффузионная модель.

В первом случае речь идет о параллельном вычислении целевой функции и параллельном выполнении операторов ЭА. Целевая функция отдельного индивидуума в популяции, как правило, не зависит от других индивидуумов, поэтому вычисление целевых функций допускает распараллеливание. При этом популяция хранится в общей памяти, а отдельный процессор считывает хромосому из памяти и возвращает результат после вычисления целевой функции в виде *fitness*-значения. Процедура синхронизации здесь проводится лишь при формировании новой популяции. В случае, если вычислительная система имеет распределенную память, то популяция и *fitness*-значения отдельных индивидуумов хранятся в "мастер"-процессоре, а расчеты производятся на вспомогательных про-

цессорах. Некоторую трудность для распараллеливания представляет процедура селекции. Здесь преимущество в смысле распараллеливания получают те формы селекции, которые не используют глобальную статистику о популяции, например, "соревновательные" способы селекции. Легко распараллеливаемыми являются операторы мутации и кроссинговера, т.к. они выполняются независимо друг от друга и, в основном, случайным образом.

Другой тип параллелизма – миграционная модель – наиболее эффективно реализуется в вычислительных системах с SIMD-архитектурой путем разделения общей популяции на отдельные подпопуляции, эволюционное моделирование которых осуществляется на отдельных процессорах. Если между отдельными подпопуляциями установить правила миграции отдельных индивидуумов, то получится миграционная модель. Для построения миграционной модели требуется задать количество и размеры подпопуляций, топологию связей между ними, частоту и интервал миграции, а также стратегию эмиграции и иммиграции. Размеры и число подпопуляций определяются числом и быстродействием используемых процессоров или транспьютеров, а также сложностью решаемой оптимизационной проблемы. Что касается топологии связей в миграционной модели, то здесь различаются два варианта: "островная" модель, предусматривающая миграцию между любыми подпопуляциями, и миграционная модель, предусматривающая обмен индивидуумами между соседними подпопуляциями. Применение того или иного варианта миграционной модели зависит от архитектуры параллельного вычислителя (сетевая или кольцевая). Частота миграции определяется размерами подпопуляций. При высокой частоте подпопуляция может быстро потерять гетерогенность. Наоборот, при слабой интенсивности миграции и большом числе подпопуляций возрастает вероятность получения нескольких хороших решений, однако сходимость ЭА замедляется [8].

Наконец, третья разновидность параллелизма ЭА – диффузионная модель – наиболее эффективно реализуется в вычислительных системах с SIMD-архитектурой. В этом случае общая популяция разделяется на большое количество немногочисленных подпопуляций, а операторы кроссинговера и селекции применяются лишь в ограниченной области, определяемой отношением соседства. Это приводит к тому, что хорошие решения очень медленно распространяются в популяции (диффузия) и проблема попадания в "локальную яму" становится менее острой.

Выводы. Эволюционные алгоритмы *наименее подходят* для решения оптимизационных проблем, если:

- ◆ Требуется найти глобальный оптимум.
- ◆ Имеется эффективный неэволюционный алгоритм оптимизации.
- ◆ Переменные, от которых зависит решение, можно оптимизировать независимо.
- ◆ Для задачи характерна высокая степень эпистазии, т.е. взаимозависимость между переменными оптимизации такова, что одна переменная подавляет другую.
- ◆ Значения целевой функции во всех точках, за исключением оптимума, являются приблизительно одинаковыми.

Принципиально подходящими для решения с помощью ЭА являются следующие оптимизационные проблемы:

- ◆ Задачи многомерной оптимизации с мультимодальными целевыми функциями, для которых, по-существу, нет подходящих неэволюционных методов решения.
- ◆ Стохастические задачи.
- ◆ Динамические задачи с блуждающим оптимумом.
- ◆ Задачи комбинаторной оптимизации.
- ◆ Задачи прогнозирования и распознавания образов.

ЛИТЕРАТУРА

1. *Курейчик В.М.* Генетические алгоритмы. Обзор и состояние // Новости ИИ. 1998. №3. С.14-63.
2. *Nissen V.* Einfuhrung in Evolutionare Algorithmen. – Braunschweig: Vieweg. – 1997.
3. *Родзин С.И.* Гибридные интеллектуальные системы на основе алгоритмов эволюционного программирования // Новости ИИ. 1999. №3.
4. *Wolpert D.H., Macready W.G.* No Free Lunch Theorems for Search // Operations Research. – Santa Fe Institute, 1995.
5. *Курейчик В.М.* Генетические алгоритмы. Состояние. Проблемы. Перспективы // Известия РАН ТИСУ. 1999. №1.
6. *Angeline P.J.* Evolutionary Algorithms and Emergent Intelligence // Diss., Ohio St.Uni, Columbus. 1993.
7. *Koza J.R., Andre D.* Parallel Genetic Programming on a Network of Transputers // Technical Report, Stanford Uni. 1995.
8. *Cantu-Paz E.* A Summary of Research on Parallel Genetic Algorithms // IlliGAL Report 95007, Uni of Illinois, Urbana-Champaign. 1995.

УДК 658.512

Ю.В. Чернухин, М.А. Беляев

ОСОБЕННОСТИ ИСПОЛЬЗОВАНИЯ ГЕНЕТИЧЕСКИХ АЛГОРИТМОВ ПРИ ОБУЧЕНИИ ПЕРЦЕПТРОНОВ

Введение. Искусственные нейронные и нейропроцессорные сети способны решать трудноформализуемые задачи, относящиеся к области искусственного интеллекта [1]. Важной особенностью таких сетей является их способность к обучению, т.е. к автоматической настройке внутренних параметров в соответствии с решаемой задачей. В основе всех нейросетевых парадигм лежит принцип умножения подаваемых на входы отдельных нейроэлементов или нейропроцессоров сигналов на соответствующие данным входам синаптические веса. Суть обучения при этом заключается в нахождении набора требуемых значений синаптических весов при помощи тех или иных алгоритмов. Однако алгоритмы обучения, как правило, являются специфичными для каждой конкретной нейросетевой парадигмы и не всегда применимы к другим нейронным моделям. Кроме того, зачастую они обладают такими недостатками, как возможность попадания процедуры поиска синаптических весов в локальный экстремум ошибки или в область ее насыщения.

Один из методов решения данных проблем состоит в использовании генетических алгоритмов (ГА), представляющих собой механизм стохастического направленного поиска [2,3]. Преимуществом ГА является их слабая связь со структурой решаемой задачи, что дает возможность их