

Инварианты и симметрии в генетических алгоритмах

М.Ю. Богатырев

Тулский государственный университет
okkambo@mail.ru

Генетические алгоритмы вызывают большой интерес исследователей во всем мире на протяжении более чем двадцати пяти лет [1-3]. В последнее время этот интерес вновь увеличился в связи с ростом производительности вычислительных систем и появлением новых вычислительных технологий (параллельные и сетевые технологии). Благодаря им, возможности решения сложных, например, NP - полных задач оптимизации существенно расширились, что делает эволюционные вычисления и относящиеся к ним генетические алгоритмы, востребованным инструментом во многих областях.

Несмотря на относительно долгую историю и обилие публикаций, генетические алгоритмы до сих пор остаются довольно экзотическим средством решения задач оптимизации. До сих пор нет того, что можно было бы назвать *теорией генетических алгоритмов*. Мы можем говорить лишь о *моделях генетических алгоритмов*, понимая под этим создание специальных моделей, описывающих процесс эволюции популяции решений задачи путем привлечения для этого различных математических объектов.

Здесь следует отметить три основных направления моделирования: классический вероятностный подход [4], применение цепей Маркова [5] и алгебраический подход [6]. Последний, появившийся сравнительно недавно, является наиболее общим и, на наш взгляд, весьма перспективным.

Упомянутая выше теория генетических алгоритмов, по-видимому, должна стать теорией *применения генетических алгоритмов*, а в более общем плане – теорией применения эволюционных вычислений. Дело в том, что генетические алгоритмы обладают одним важным свойством: настройки алгоритма – выбор кодового представления и генетических операторов – определяются свойствами пространства поиска решаемой задачи.

Симметрия как раз является тем свойством, привлечение которого к исследованию генетических алгоритмов позволяет глубже понять их природу.

Алгебраическая модель генетического алгоритма

Рассмотрим алгебраическую модель генетического алгоритма, используя нотацию Воса [7]. Пусть X - множество решений, генерируемых алгоритмом, его *пространство*

поиска. Подмножество $P \subset X$ из r элементов составляет популяцию решений. Функция пригодности f есть отображение пространства поиска X на пространство R^+ положительных действительных чисел: $f: X \rightarrow R^+$.

Важной составляющей частью всякого генетического алгоритма является кодирование элементов популяции. К кодированию относятся: алфавит $A = \{a_i, i = 1, 2, \dots, d\}$ – конечный набор символов; хромосома – это l – местный кортеж, элементами которого являются символы из алфавитов A_j ; ген – элемент хромосомы, символ a_i из алфавита A_j , стоящий на i – м месте в хромосоме. Хромосомы образуют популяцию P как отношение вида: $P \subset A_1 \times A_2 \times \dots \times A_l$.

Генетический алгоритм начинает работать с исходной популяцией, P_0 . Очередная популяция (поколение) P_{t+1} генерируется из предыдущей популяции P_t действием операторов отбора, мутации и рекомбинации. Эволюция популяций к оптимальному решению приводит к тому, что популяция содержит хромосомы с максимальной величиной пригодности в смысле выбранного критерия. Если оптимальное решение единственно, то есть существует единственный элемент $x^* \in X$, такой, что $f(x^*) = \max_{x \in S} \{f(x)\}$, то финальная популяция содержит только хромосомы, соответствующие элементу x^* .

Каждой популяции P_t сопоставляется вектор $\mathbf{p}^{(t)}$ размерности n , где n – количество объектов в X , элементами которого являются числа $p_j^{(t)} = v/n$, где число v показывает, сколько раз элемент j встречается в k – й популяции. Очевидно, что $p_j^{(t)} \geq 0$ и $\sum_{j=1}^n p_j^{(t)} = 1$.

Величину $p_j^{(t)}$ можно трактовать как вероятность нахождения элемента j в популяции t . Здесь рассматриваются алгоритмы, в которых популяция не меняет свой размер от поколения к поколению. Поэтому эволюция популяции состоит в том, что она заполняется лучшими особями, которые могут быть представлены несколькими экземплярами.

Переход от одного поколения к другому описывается оператором \mathbf{A} , таким, что $\mathbf{A}(\mathbf{p})$ есть вектор, k – я компонента которого равна вероятности, что элемент $k \in X$ появится в следующем поколении.

Этот оператор состоит из двух компонент:

$$\mathbf{A} = \mathbf{M} \circ \mathfrak{R},$$

где \mathfrak{R} – схема отбора, \mathbf{M} – схема замещения, объединяющая рекомбинации и мутации.

Схема замещения – это составная часть модели генетического алгоритма в векторно-матричной форме относительно векторов $\mathbf{p}^{(t)}$. Схема замещения включает матрицу замещения \mathbf{M} такую, что для хромосом i, j ее элемент M_{ij} равен вероятности появления одной, конкретной хромосомы.

Выражения для матриц замещения, соответствующих различным вариантам операторов кроссовера и мутации, можно найти в [7-9]. Приведем важное для нас соотношение, в котором схема замещения \mathbf{M} , комбинирующая кроссовер и мутации, представляет собой вектор-функцию $\mathbf{M}(\mathbf{p})$, компонентами которой являются квадратичные формы:

$$\mathbf{M}(\mathbf{p}) = \left\langle \mathbf{p}^{(t)T} \mathbf{M} \mathbf{p}^{(t)}, [S(2)\mathbf{p}^{(t)}]^T \mathbf{M} S(2)\mathbf{p}^{(t)}, \dots, [S(k)\mathbf{p}^{(t)}]^T \mathbf{M} S(k)\mathbf{p}^{(t)}, \dots, [S(n)\mathbf{p}^{(t)}]^T \mathbf{M} S(n)\mathbf{p}^{(t)} \right\rangle \quad (1)$$

Здесь использован тот факт, что все операции, выполняемые генетическим алгоритмом с хромосомами, суть перестановки на множестве всех возможных хромосом, определяемом длиной строки хромосомы и алфавитом кодирования генов. Перестановки заданы матрицей перестановок $S(k)$, зависящей от k , элементы которой задаются, исходя

из соотношения: $s_{i,j} = \begin{cases} 1, & \text{если } j - i = k \\ 0, & \text{если } j - i \neq k \end{cases}$. Тогда для компоненты k вектора состояний имеем:

$$p_k^{(t+1)} = [S(k)\mathbf{p}^{(t)}]^T \mathbf{M} S(k)\mathbf{p}^{(t)} \quad (2)$$

Подчеркнем, что в модели используется единственная матрица замещения \mathbf{M} , зависящая от параметров кроссовера и мутации, с помощью которой, применяя отбор на каждом шаге, генерируется вся дискретная последовательность состояний алгоритма – его *эволюция*.

Главным недостатком модели (1)-(2) является ее высокая размерность (n – число всех возможных хромосом при выбранной их длине и алфавите кодирования!), впрочем этот недостаток становится все менее значимым. В целом, модель не решает проблему настройки генетических алгоритмов, но предлагает математический инструмент их исследования.

Определения инварианта и симметрии

Свойство симметрии тесно связано с понятием *инвариантности*. Согласно Г. Вейлю [10]: *объект является симметричным, если после применения к нему определенной операции он остается таким же, как до операции*. Такой объект считается *инвариантным* относительно данной операции, а сама операция называется *операцией*

симметрии объекта. Легко показать, что множество всех операций симметрии образуют *группу* [10], поэтому исследования симметрии привлекается аппарат теории групп. Основным объектом здесь является группа преобразований $G = \{g\}$ - множество всех преобразований g , замкнутое относительно композиции преобразований, включающее обратные g^{-1} и тождественное преобразование e .

Понятие инварианта органически связано с групповой структурой, определенной на произвольном множестве X . Если G – группа, действующая на множестве X , то это означает, что задано отображение $(g, x) \rightarrow gx: G \times X \rightarrow X$ со свойствами: $(g_1 g_2)x = g_1(g_2)x$ для всех $x \in X, g_1, g_2 \in G$, $ex = x$ для всех $x \in X$. Все преобразования группы вида $x \rightarrow gx$ определяют отображение $\mu_g: X \rightarrow X$. Элемент $x \in X$ называется *инвариантом*, если $\mu_g(x) = x$ для всех $g \in G$. Подмножество $F \subset X$ множества X называется G – инвариантным или просто инвариантным, если $\mu_g(F) \subseteq F$ для всех $g \in G$.

Таким образом, симметрия – это существование определенного инварианта в модели M , заданной на множестве X как отображение $M: X \rightarrow X$. Симметрия может проявляться также в виде инвариантного подмножества, которому принадлежат, например, решения уравнений модели.

Данные определения инварианта, инвариантного подмножества и симметрии обладают необходимой общностью для применения их в моделях различных математических форм. Поэтому под моделью мы можем понимать как математическую модель решаемой задачи (систему уравнений и выражения для критерия оптимальности), так и упомянутую выше модель генетического алгоритма.

Тем не менее, следует различать: симметрию задачи оптимизации, решаемой с помощью генетического алгоритма и симметрию как внутреннее свойство самого генетического алгоритма.

Структурно – инвариантный анализ генетических алгоритмов

Структурно – инвариантный анализ как метод опирается на две составные части: модель генетического алгоритма и групповые свойства пространств поиска и состояний популяции. Целью анализа является получение результатов, помогающих применять генетические алгоритмы на практике.

Рассмотрим некоторые характерные результаты.

Ниши генетических алгоритмов

Данный результат касается понижения размерности модели (1)-(2) путем ее декомпозиции.

Отметим, что групповая структура присуща пространству поиска по определению. Пространство поиска X , содержащее все возможные решения задачи оптимизации, идентифицируется как множество целых чисел $1, 2, \dots, n$. Соответственно, в нем действует полная группа перестановок S_n . Полная группа перестановок является максимальной группой на множестве X . Группа G в конкретном алгоритме может быть ее подгруппой и также может быть задана перестановками, то есть, изоморфна некоторой подгруппе полной группы перестановок S_n . Эта подгруппа задается матрицами $S(k)$ в модели (1)-(2). Данное свойство открывает возможность интерпретировать действие генетических операторов как групповые операции. Для этого выводятся необходимые условия коммутации операторов кроссовера и мутации с *представлением* группы G в виде матриц перестановок. Важным следствием этого является следующее утверждение [12]: матрица замещения M в модели (1)-(2) определяет схему замещения M если эта схема коммутирует с представлением группы G :

$$M \circ S(g) = S(g) \circ M \quad (3)$$

Условие (3) есть условие симметрии модели генетического алгоритма. Зная группу симметрии модели (1)-(2), можно декомпозировать модель на совокупность моделей меньшего порядка, чем порядок исходной модели. Для этого методами теории групп строится декомпозирующее преобразование [11] для матрицы замещения, в результате чего она приобретает блочно – диагональное строение. В результате декомпозиции схемы замещения она распадается на совокупность независимых схем замещения

$P_{t+1}^j = M^j P_t^j$, действующих в соответствующих *инвариантных подпространствах* пространства состояний алгоритма. Такие подпространства образуют так называемые *ниши генетического алгоритма* [4]. Ниши – это подмножества решений, из которых алгоритм не выходит во время своей работы. Зацикливаться в нишах может как вся популяция, так и ее подмножества. Таким образом, ниши определяются групповой структурой генетического алгоритма.

Отметим, что данный подход является обобщением Фурье – анализа генетических алгоритмов, предлагаемого в работах [5, 7]. Известные функции Уолша, применяемые для моделирования генетических алгоритмов [3], также являются частным случаем преобразований Фурье, когда алфавит кодирования бинарный.

Инвариантные шаблоны

Известно, что сходимость генетических алгоритмов и эффективность поиска ими экстремума определяются тем, насколько стабильны возникающие в процессе эволюции *инвариантные шаблоны* - участки хромосом, имеющий конкретный код, повторяющийся в других хромосомах. Участки хромосом вне инвариантного шаблона могут иметь коды, содержащие любые символы из алфавита, используемого в алгоритме. Коды инвариантных шаблонов содержат “генетическую информацию”, сохранение и накопление которой ведет к решению задачи оптимизации.

Термин *инвариантный шаблон* или просто *шаблон* кажется нам более удачным, чем применяемый в русскоязычной литературе термин *шима*, являющийся транслитерацией английского слова *schema*, что переводится как *схема*. Однако, термином *схема генетического алгоритма* удобнее обозначать всю совокупность его параметров.

Структурно – инвариантный анализ, примененный к исследованию инвариантных шаблонов, позволил установить следующее.

Каждый инвариантный шаблон принадлежит инвариантной подгруппе группы, действующей в пространстве состояний, или является смежным классом этой подгруппы. Поэтому групповая структура пространства состояний определяет структуру множества инвариантных шаблонов. Инвариантность шаблона в поколениях популяции определяется свойствами рекомбинации. Для генетических алгоритмов с бинарным алфавитом так называемый *универсальный кроссовер* [3] обеспечивает инвариантность шаблонов. Если алфавит алгоритма небинарный, то доказываемое существование оператора кроссовера, обеспечивающего инвариантность шаблонов.

Данный результат важен потому, что позволяет при известной группе симметрии генетического алгоритма определить подпространства его инвариантных шаблонов как решение задачи разложения группы на классы и инвариантные подгруппы.

В качестве примера рассмотрим структуры, которые связаны с *шаблонами генетических алгоритмов*. Каждый шаблон $W(l, \alpha, \beta, s)$ представляет собой набор хромосом, имеющих одинаковые значения определенных генов. Параметрами шаблона являются: l – длина хромосомы, α – порядок шаблона, β – определяющая длина шаблона, s – код шаблона, строка символов шаблона из алфавита $\{A, *\}$, где A – некоторый набор символов, а знак $*$ соответствует любому символу. Шаблон $W(l, \alpha, \beta, s)$ называется *инвариантным*, если для любых удовлетворяющих ему хромосом x, y , выполняется равенство $\forall x, y \in W : F(x) = F(y)$, где F – функция пригодности, определенная для

данного алгоритма. Как следует из определения, код шаблона определяет пригодность всех хромосом, имеющих данный шаблон.

Например, групповая структура на множестве бинарных хромосом длины 3 с использованием инвариантных шаблонов:

$$\begin{array}{l}
 0^{**} = \{000, 001, 010, 011\} \text{ – подгруппа (группа Клейна)} \\
 1^{**} = \{100, 101, 110, 111\} \text{ – смежный класс} \\
 0^ = \{000, 001, 100, 101\} \text{ – подгруппа (группа Клейна)} \\
 1^ = \{010, 011, 110, 111\} \text{ – смежный класс} \\
 **0 = \{000, 010, 100, 1110\} \text{ – подгруппа (группа Клейна)} \\
 **1 = \{001, 011, 101, 111\} \text{ – смежный класс} \\
 00^* = \{000, 001\} \text{ – подгруппа } S_2 \equiv C_2 \\
 01^* = \{010, 011\} \text{ – смежный класс} \\
 10^* = \{100, 101\} \text{ – смежный класс} \\
 11^* = \{110, 111\} \text{ – смежный класс}
 \end{array}
 \left. \vphantom{\begin{array}{l} 0^{**} \\ 1^{**} \\ *0^* \\ *1^* \\ **0 \\ **1 \\ 00^* \\ 01^* \\ 10^* \\ 11^* \end{array}} \right\} \begin{array}{l} \text{факторгруппа } S_2 \equiv C_2 \\ \text{факторгруппа } S_2 \equiv C_2 \\ \text{факторгруппа } S_2 \equiv C_2 \\ \text{факторгруппа Клейна} \end{array}$$

Практические приложения

Рассмотренные здесь результаты позволяют строить конкретные варианты генетических алгоритмов для решения прикладных задач, используя свойство симметрии.

Очевидная, *визуальная симметрия*, присуща многим объектам и задачам оптимизации. Другая, *скрытая симметрия*, есть следствие групповых свойств генетического алгоритма. Свойство симметрии генетических алгоритмов связано с известной проблемой поиска глобального экстремума мультимодальных функций, когда алгоритм “зависает” в областях локальных экстремумов. Такое поведение алгоритма может быть обусловлено как скрытой, так и визуальной симметрией. Например, данное явление характерно для симметричных ландшафтов, образованных значениями функции пригодности, неразличимыми в областях локальных экстремумов, имеющих одинаковую величину. Исследование симметрий ландшафтов генетических алгоритмов позволяет находить все локальные, совпадающие между собой экстремумы мультимодальной функции. Такие задачи возникают в ряде приложений, в частности, в задачах извлечения знаний [14], в задачах кластеризации [15] и т.д.

Практический интерес представляет решение следующей общей, достаточно сложной проблемы. Современный групповой анализ позволяет связать практически с каждой системой уравнений некоторую допускаемую ей группу преобразований и найти соответствующие инварианты, то есть связать симметрии (возможно, тривиальные) с

системой уравнений. Исследование того, каким образом эти симметрии индуцируют симметрии в эволюционных алгоритмах, позволит приблизиться к решению другой фундаментальной проблемы: как построить эффективный генетический алгоритм для конкретной задачи оптимизации.

Литература

1. Holland J.H. *Adaptation in Natural and Artificial Systems*, Ann Arbor: The University of Michigan Press. Reprinted by MIT, 1992.
2. Емельянов В.В., Курейчик В.В., Курейчик В.М. Теория и практика эволюционного моделирования. – М.: Физматлит, 2003 - 432 с.
3. Sivanandam S.N., Deepa S. N. *Introduction to Genetic Algorithms* – Springer Verlag, 2007.
4. Goldberg D. E. *Genetic Algorithms in search, optimization and machine learning*. - Addison-Wesley, 1989.
5. Whitley D. L., Vose, M. D. *Foundations of Genetic Algorithms*. - Morgan Kaufmann Publishers, San Francisco, 1995. – 86 p.
6. Radcliffe, N.J. The algebra of genetic algorithms. - *Annals of Mathematics and Artificial Intelligence*. – V. 10 p. 339 – 384.
7. Vose, M. D., Rowe, J. E. Random heuristic search: Applications to GAs and functions of unitation. // *Computer Methods in Applied Mechanics and Engineering*, 186(2-4), 2000. p.p. 195-220.
8. Богатырёв М.Ю. Структурно – инвариантный подход к исследованию генетических алгоритмов. - *Изв. ТулГУ. Сер. Математика. Механика. Информатика. Том 8, вып. 3. Информатика*. - Тула, 2002. – С. 128 - 136.
9. Богатырёв М.Ю. Генетические алгоритмы: принципы работы, моделирование, применение – Тула, ТулГУ, 2003. – 152 с.
10. Вейль Г. *Симметрия*. – М.: Наука, 1968. – 192 с.
11. M. Bogatyrev. *Modelling Systems With Symmetry* – In: *Proceedings of the 4 th International IMACS Symposium of Mathematical Modelling*. - Vienna, Austria, February 5-7, 2003.- ARGESIM-Verlag, Vienna, 2003. - pp. 270 - 275.
12. Богатырев М.Ю., Столбовская И.А.. Симметрии в генетических алгоритмах. // *Известия ТулГУ. Серия. Вычислительная техника. Информационные технологии. Системы управления. Т. 1. Вып. 2. Информационные технологии*. – Тула: ТулГУ, 2004. С. 51 – 57.
13. M. Bogatyrev, V. Latov, K. Avdeev. *Symmetry Based Decomposition and its Application in Evolutionary Modelling*. – *Applied Mathematica: Proc. of 8 th International Mathematica Symposium*. Avignon, France, 19-23 June, 2006.
14. Богатырев М.Ю., Ковалев Д.А., Евсюков В.В. Эволюционный подход к извлечению знаний из реляционных баз данных в корпоративных информационных системах – «Информационные технологии», 2004, № 9. – С. 19-27.
15. Богатырев М. Ю., Латов В. Е., Столбовская И. А., Тюхтин В. В. Эволюционный подход к задаче кластеризации на концептуальных графах и его применение в системах поддержки электронных библиотек. - *Математические методы распознавания образов. 13 Всероссийская конференция.. Сб. докладов*. – М.: МАКС Пресс, 2007. – 668 с.- С. 464-468.