

1. Модернизация ГА

Одной из серьезных проблем, возникающих при использовании генетических алгоритмов, является преждевременная сходимость. Не рекомендуется использовать классические ГА на маленьких популяциях, поскольку в популяциях с малым размером гены распространяются слишком быстро: все особи становятся похожими (популяция вырождается) еще до того, как найдено решение задачи. То есть новый генотип с лучшей оценкой быстро вытесняет менее хорошие комбинации генов, исключая тем самым возможность получения лучшего решения на их базе. Можно предложить три основных пути устранения преждевременной сходимости: увеличение размера популяции, применение самоадаптирующихся генетических операторов и создание «банка» заменяемых особей.

В первом случае, увеличивая размер популяции, можно надеяться на достижение многообразия генотипа в популяции. Но, с другой стороны, увеличение числа особей ведет к увеличению занимаемой памяти и времени работы алгоритма. Данный подход может быть эффективен или при параллельных вычислениях, или при наличии достаточно простой целевой функции.

Второй, и самый распространенный способ, — использование самоадаптирующихся алгоритмов — является более эффективным. Самоадаптация заключается в применении динамических мутаций. Динамические мутации в зависимости от скрещивающихся особей меняют значение вероятности мутации, тем самым становится возможным самоуправление алгоритма. В таких случаях выбирается малый размер популяции.

В третьем подходе создается массив для сохранения особей, генотип которых был утерян при формировании новых поколений, и временами эти особи добавляются в популяцию.

Ниже более подробно рассмотрим основные варианты динамических мутаций.

1.1. Самоадаптирующиеся алгоритмы

1.1.1. Неоднородная мутация

Одним из способов организации самоадаптирующегося алгоритма является применение неоднородной мутации [10]. Если мутирует ген y_i , то новое значение y_i' случайно генерируется на отрезке $[\min_i, \max_i]$:

$$y_i' = \begin{cases} y_i + (\max_i - y_i) \left(1 - r^{(1 - \frac{t}{T})^b}\right), & \text{если } q = 0, \\ y_i - (y_i - \min_i) \left(1 - r^{(1 - \frac{t}{T})^b}\right), & \text{если } q = 1, \end{cases}$$

где q случайным образом принимает значения 0 или 1; r — случайное число, принимающее значение из диапазона $[0;1]$; t — номер поколения; T — максимальное число поколений; b — некоторый параметр, обусловленный природой задачи; \min_i и \max_i — верхняя и нижняя границы для величины y_i .

1.1.2. Инцест

Стратегия инцеста используется как механизм самоадаптации оператора мутации. Она заключается в том, что «плотность мутации» (вероятность мутации каждого гена) определяется для каждого потомка на основании генетической близости его родителей. Например, это может быть отношение числа совпадающих генов родителей к общему числу генов хромосомы. В результате инцеста на начальных этапах алгоритма при высоком разнообразии генофонда популяции вероятность мутации будет предельно мала, т.е. практически будет происходить лишь скрещивание. При уменьшении разнообразия, возникающего в случае попадания алгоритма в локальный оптимум, вероятность мутации возрастет. Очевидно, что при полном схождении популяции алгоритм станет стохастическим, тем самым вероятность выхода популяции из локального оптимума возрастет.

Высокая вероятность мутаций гарантирует появление многообразия в популяции, но вполне возможно и разрушение хорошей особи при мутировании. Перьяк с соавторами предложили использовать помимо мутации инцеста метод *супериндивидуального приближения* для отбора особей в новую популяцию [12–15]. При супериндивидуальном приближении выбирается наилучшая особь среди числа родителей и их потомков — элитная хромосома. Все хромосомы новой популяции являются копиями этой элитной хромосомы. Поэтому в популяции будут происходить сильные мутации.

Инцест также известен как мутация, зависящая от расстояния.

1.1.3. Критерий расстояния

В качестве расстояния в случае представления особи в виде бинарной строки (код Грея) берется расстояние Хемминга. Перьяк и другие [12–15] предложили другой критерий расстояния, основанный на критерии минимальной площади. И вместо использования абсолютного расстояния, они взяли относительное расстояние, которое в расчетах границ для каждой переменной индивидуально. Пусть $A = (a_1, \dots, a_i, \dots, a_n)$ и $B = (b_1, \dots, b_i, \dots, b_n)$ — два родителя, \min_i и \max_i — верхняя и нижняя границы для i -й переменной:

$$\text{dist}(A, B) = \frac{1}{n} \sum_{i=1}^n \left| \frac{a_i - b_i}{\max_i - \min_i} \right|$$

где d — параметр, используемый для вычисления расстояния.

Пусть $M(A, B) = 1 - \text{dist}(A, B)$ — вероятность мутации, связанная с расстоянием. Тогда $\text{dist}(A, B) \in [0, 1]$ и $M(A, B) \in [0, 1]$. $M(A, B)$ достигает максимума при $\text{dist}(A, B) = 0$; при идентичных родителях A и B получаем $M(A, B) = 1$. Кроссинговер двух идентичных индивидуумов при $M(A, B) = 1$, вызывает обязательную мутацию для каждой переменной потомка. Очевидно, что такая мутация является очень большой, и поэтому авторы ввели множитель M_m , позволяющий скорректировать значение $M(A, B)$. Потом определяют эффективную вероятность мутации P_r :

$P_r(A, B) = M(A, B)M_m = (1 - \text{dist}(A, B))M_m$, где M_m — фиксированный параметр. В рассматриваемом классе задач оптимальными являются следующие значения параметров: $d = 0,2$ и $M_m = 0,9$.

1.1.4. Параметры вычислений

Для вычислений Перьяк и другие [12–15] использовали классические тестовые функции Де Йонга, широко применяемые для анализа эффективности ГА. Использовались следующие параметры:

- размер популяции $N = 20$;
- вероятность кроссинговера $P_c = 0,8$;
- вероятность мутации воспроизводства $P_m^{\text{rep}} = 0,01$;
- максимальная вероятность мутации $M_m = 0,9$.

Для каждой функции алгоритм запускался 100 раз.

Процесс работы генетического алгоритма представлен на рисунке 15.

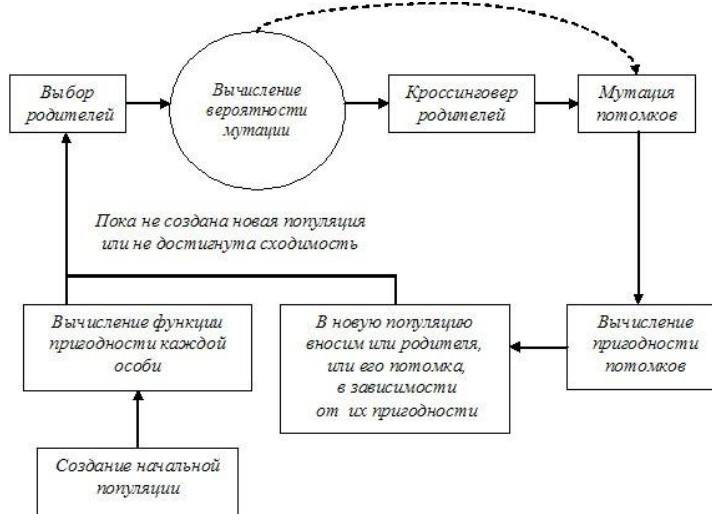


Рис. 15. ГА в работах Перьякса

Список литературы

1. Батищев, Д.И. Генетические алгоритмы решения экстремальных задач [Текст]/ Д.И. Батищев ; Нижегородский госуниверситет. — Нижний Новгород : 1995.с. — 62с.
2. Дарвин Ч. О происхождении видов путём естественного отбора или сохранении благоприятствуемых пород в борьбе за жизнь [Текст]/ Ч. Дарвин. — М.: АН СССР, 1939. — Т.3.
3. Гладков Л.А., Курейчик В.В., Курейчик В.М. Генетические алгоритмы [Текст]/ Под ред. В.М. Курейчика. — 2-е изд., испр. и доп. — М.: ФИЗМАТЛИТ, 2006. — 320 с. — ISBN 5-9221-0510-8.
4. 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
5. Fogel D. B. Evolutionary computation: towards a new philosophy of machine intelligence [Текст]/D. B. Fogel. — Piscatway: IEEE Press, 2000. — ISBN 0-7803-3481-7
6. Goldberg D. Genetic Algorithms in Search, Optimization, and Machine Learning [Текст]/ D. Goldberg. — Massachusetts: Addison-Wesley, 1989. — ISBN 0201157675
7. Hartmann A.K., Rieger H. Optimization Algorithms in Physics. — Berlin: Wiley-VCH, 2002. — 383 с. — ISBN 3527403078.
8. Holland J. H. Adaptation in Natural and Artificial Systems: An Introductory Analysis With Applications to Biology, Control, and Artificial Intelligence [Текст]/J. H. Holland. — The MIT Press, Cambridge, 1992. — ISBN 0262581116
9. Koza J. R. Genetic Programming [Текст]/J. R. Koza. — Cambridge: The MIT Press, 1998.— 609 с. — ISBN 0-262-11170-5
10. Michalewicz Z. Genetic algorithms + Data Structures = Evolution Programs [Текст]/Z. Michalewicz. — New York: Springer-Verlag, 1996. — 387 с. — ISBN 3540606769
11. Mitchell M. An Introduction to Genetic Algorithms [Текст]/M. Mitchell. — Cambridge: MIT Press, 1999 — 158 с. — ISBN 0-262-13316-4 (HB), 0-262-63185-7 (PB)
12. Periaux J., Sefrioui M., Ganascia J.-G. Fast Convergence Thanks to Diversity // Evolutionary computing. — San Diego, 1998. — 9 с.
13. Periaux J., Sefrioui M. Evolutionary computational methods for complex design in aerodynamics // AIAA-98-0222. — Reno, 1998. — 15 с.
14. Periaux J. Combining Game Theory and Genetic Algorithms with Application to DDM-Nozzle Optimization Problems // Proceedings of DDM. — Greenwich, 1998. — 17 с.
15. Periaux J. Genetic Algorithms for electromagnetic backscattering multiobjective optimization // Genetic algorithms for Electromagnetic Computation. — Ed: Erir. Mielchessen, 1998. — 30 с.
16. Schwefel H.-P. Evolution and Optimum Seeking [Текст]/H.-P. Schwefel. — New York: John Wiley & Sons, 1995.
17. Генетические алгоритмы, Исаев А. — <http://www.algolist.manual.ru>
18. Генетические алгоритмы на сайте Санкт-Петербургского государственного университета информационных технологий, механики и оптики. — <http://rain.ifmo.ru/cat>
19. Исследования по ГА в Мичиганском университете. — <http://garage.cps.msu.edu>
20. Исследования по ГА в университете штата Колорадо. — <http://www.cs.colostate.edu>
21. Организация по Генетическим Алгоритмам. — <http://www.geneticprogramming.org>
22. Ассоциация по Генетическим Алгоритмам университета Джорджа Мейсона. — <http://www.cs.gmu.edu/research/gag>