УДК 519.68

БИОНИЧЕСКИЕ АЛГОРИТМЫ КОМБИНАТРНОЙ ОПТИМИЗАЦИИ

Семенкина О.Е.,
научный руководитель д-р физ.-мат. наук Попов Е. А.
Сибирский государственный аэрокосмический университет имени М.Ф. Решетнева

Одной из часто встречающихся практических задач является задача оптимизации, в том числе при принятии решений. Стандартные методы, опирающиеся на то, что оптимизируемая функция обладает набором определенных качеств (гладкость, одноэкстремальность), зачастую не справляются с решением сложных прикладных задач. Поэтому в настоящее время для этих целей используются принципиально отличающиеся от них стохастические многоагентные алгоритмы. В связи с этим исследование таких алгоритмов является актуальной научной проблемой.

В данной работе рассматриваются несколько бионических алгоритмов - муравьиный алгоритм (Ant Colony Optimization (ACO)), алгоритм умных капель (Intelligent Water Drops (IWDs)), генетический алгоритм (Genetic Algorithm (GA или ГА)), адаптивный ГА (adaptive GA), а также эвристика Лин-Кернигана для решения одной из задач комбинаторной оптимизации – задачи коммивояжера (Travelling salesman problem (TSP)) [1]. Задача коммивояжера является обобщением задачи о гамильтоновых циклах в графах и относится к классу NP-полных задач. Она формулируется следующим образом: *Пусть задано множество из n городов. Требуется найти замкнутый обход минимальной длины, при условии, что каждый город должен быть посещен единственный раз.* Эта задача имеет широкое применение на практике, например, задачи маршрутизации, составления плана для одного станка или нарезки рулонного материала являются задачами коммивояжера. В более общем случае задачами на перестановках являются также задачи составления расписания с ограничениями, задачи игрового типа поиска оптимальной стратегии и т.д.

Одним из стандартных методов решения задач оптимизации является локальный спуск, в частности алгоритм 3-замены (3-opt) [2] (с мультистартом - эвристика Лина-Кернигана [3]). Суть алгоритма состоит в том, что относительно текущего решения, представленного циклическим графом f со значением целевой функции $c(f)$, рассматривается его окрестность, то есть множество циклических графов, которые можно получить из f удалением не более 3х ребер и заменой другими 3мя ребрами. Если в этом множестве существует граф g, целевая функция которого $c(g) \succ c(f)$, то g назначается текущим решением. Процедура повторяется до тех пор, пока текущее решение не перестанет изменяться.

Муравьиный алгоритм основывается на имитировании поведения и организации колонии муравьев в природе [4]. Не смотря на то, что муравьи являются практически слепыми животными, они находят кратчайший путь от гнезда до источника пищи. Для обмена информацией муравьи пользуются определенным ферментом (или феромоном), который они оставляют на пути, по которому двигаются. При выборе своего маршрута, каждый муравей с большей вероятностью пойдет по тому пути, на котором находится большее количество феромона. ACO имеет несколько важных параметров: коэффициент ρ такой, что $(1-\rho)$ - это *испарение* следа, α - относительна важность следа, β - относительная важность расстояния между городами.

Алгоритм умных капель [5] заимствует основную идею из законов, по которым естественные реки прокладывают свой маршрут, в том числе выбор самого легкого пути и стремление к низшей точке поверхности. Река представляет собой результат противоборства множеств капель воды друг с другом и, в тоже время, с

сопротивлением окружающей среды, которая представлена типом грунта в русле реки. Алгоритм IWDs имеет большое число настраиваемых параметров: коэффициенты обновления скорости a_v, b_v и c_v, коэффициенты обновления грунта a_s, b_s и c_s, локальный параметр обновления грунта ρ_n, глобальный параметр обновления грунта ρ_{IWD}, начальное количество грунта межу городами *InitSoil*, начальная скорость каждой капли *InitVelocity*.

Генетический алгоритм основывается на заимствовании из природы идеи естественной эволюции [6], [7]. Решение задачи представлено в виде хромосомы индивида в популяции (множество решений). В GA для задачи коммивояжера хромосома представляет собой перестановку из n чисел (номеров городов) в порядке их посещения. В своей работе GA использует операторы, следующие друг за другом, а именно селекция, рекомбинация и мутация. В связи с тем, что представление решения в виде хромосомы при решении задачи коммивояжера отличается от стандартной бинарной строки, то и стандартные операторы видоизменены. большое количество настраиваемых параметров, таких как вероятность мутации, тип селекции – турнирная (с выбором размера турнира), пропорциональная, ранговая с линейным и ранговая с экспоненциальным ранжированием (с выбором параметра $\lambda \in [0;1]$).

Существенный недостаток всех приведенных выше алгоритмов – большое число настраиваемых параметров, что затрудняет использование их на практике, так как в этом случае имеется серьезное ограничение на ресурсы, а узнать заранее какие настройки будут оптимальными на конкретной задаче невозможно. В связи с этим применяется метод адаптации [8], то есть настройка параметров алгоритма по ходу его работы. В данной работе был выбран способ адаптации, описанный в [9].

В Adaptive GA для каждого оператора выбор его варианта осуществляется отдельно. В начале работы алгоритма вероятности выбора всех вариантов оператора одинаковы, а на каждом следующем поколении производится оценка эффективности каждого варианта оператора. В данной работе Adaptive GA имел выбор из 8 вариантов селекции – турнирная с размером турнира 2, 4 и 8, ранговая с линейным ранжированием, ранговая с экспоненциальным ранжированием с параметром λ равным 0.95, 0.8 и 0.5 и пропорциональная, а также 5 вариантов мутации – очень низкая, низкая, средняя, высокая и очень высокая.

Программа, реализующая перечисленные выше алгоритмы была написана на языке программирования C++, в среде Embarcadero RAD Studio C++ Builder XE3. Окно программы приведено на Рис. 1. Данная программа позволяет просматривать информацию об алгоритмах в формате .pdf, загружать данные задачи в формате .txt, визуализировать их, а также находить и визуализировать решение, полученное одним из алгоритмов: 3-opt, ACO, IWDs, GA или Adaptive GA.

Эффективность работы рассматриваемых алгоритмов сравнивалась на двух задачах – Eil51 и Oliver30. При этом бионическим алгоритмам давалось столько ресурсов, сколько в среднем для своей работы требовал алгоритм 3-opt (52800 вычислений целевой функции для Oliver30 и 342210 для Eil51). Результаты работы оценивались по таким параметрам как надежность, длина и ошибка лучшего маршрута из всех прогонов, средние длина и ошибка маршрута по всем прогонам, а также среднеквадратичное отклонение и отклонение лучшего от худшего. Данные по задачам Oliver30 и Eil51 представлены в таблицах 1 и 2 соответственно.

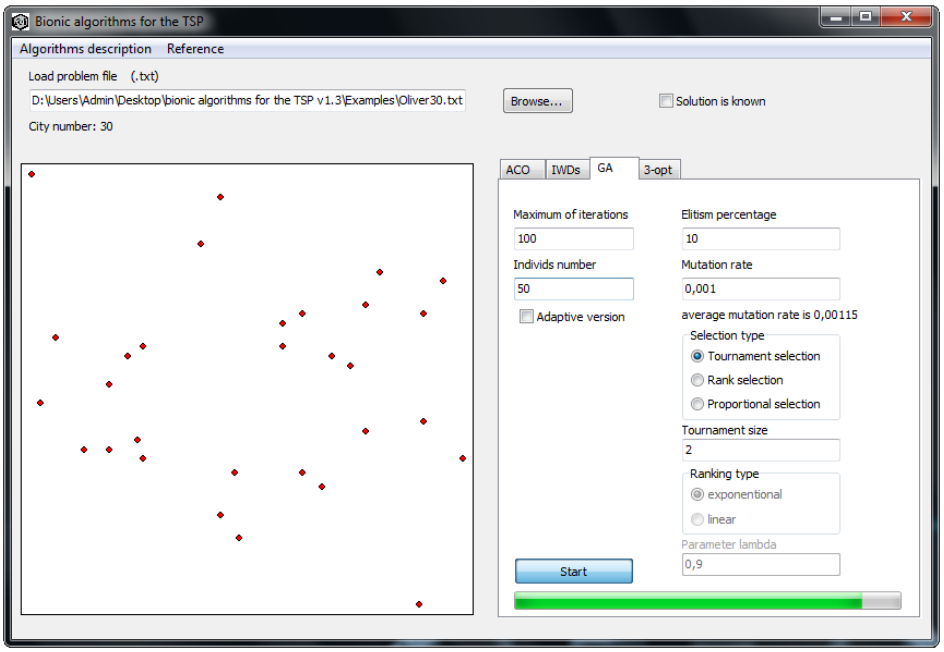

Рис. 1. Окно программы

Таблица 1. Надежность работы рассматриваемых алгоритмов на задаче Oliver30.

Oliver30		3-замена	ACO	IWDs	GA	Adaptive GA
Надежность		0.29	0.91	0.27	0.28	0.17
Лучший	**значение**	423.741	423.741	423.741	423.741	423.741
	ошибка	0	0	0	0	0
Средний	**значение**	428.618	423.782	425.5	432.356	434.239
	ошибка	4.87654	0.040608	1.75938	8.61511	10.4981
Среднеквадратичное отклонение		7.47402	0.167533	3.7022	13.2365	13.6875
Отклонение худшего от лучшего, %		9.27723	0.22448	6.37278	14.3506	15.103

Таблица 2. Надежность работы рассматриваемых алгоритмов на задаче Eil51.

Eil51		3-замена	ACO	IWDs	GA	Adaptive GA
Надежность		0	0	0	0	0
Лучший	**значение**	428.872	429.484	428.872	431.953	429.118
	ошибка	2.87176	3.484	2.872	5.953	3.118
Средний	**значение**	438.598	432.732	437.279	447.943	449.938
	ошибка	12.5985	6.73212	11.279	21.943	23.9381
Среднеквадратичное отклонение		5.01605	3.11622	5.57137	7.5848	9.94143
Отклонение худшего от лучшего, %		5.02188	2.00496	5.24329	8.26178	10.1904

Как видно из приведенных таблиц, алгоритмы ACO и IDWs превосходят алгоритм 3-замена, который чувствителен к выбору начальной точки, однако GA и adaptive GA уступают им. Это объясняется тем, что генетический алгоритм универсален. GA применим к гораздо более широкому классу задач, в то время как ACO и IWDs изначально разработаны именно для комбинаторных задач. Более того, GA и IWDs начинают свою работу с решений, полученных случайным образом, а ACO уже на фазе инициализации ориентируется на расстояния до городов, поэтому в среднем ему требуется гораздо меньшее количество итераций для отыскания решения.

Стоит отметить, что существенным недостатком ACO, GA и IWDs является большое число настраиваемых параметров, так как выбор параметров является сложной задачей сам по себе даже для специалиста. Поэтому создание адаптивных

алгоритмов, подстраивающихся под задачу, существенно облегчает их применение, что является большим преимуществом Adaptive GA. Кроме того, хотя Adaptive GA и уступает остальным алгоритмам при лучших параметрах на задаче, он превосходит «средние» алгоритмы.

Таким образом, в данной статье были рассмотрены некоторые алгоритмы комбинаторной оптимизации, такие как муравьиный алгоритм, алгоритм умных капель, стандартный и адаптивный генетический алгоритмы, а также алгоритм 3-замена. Была исследована эффективность этих алгоритмов на задаче коммивояжера, проведен сравнительный анализ работы данных алгоритмов. Также рассмотрено понятие адаптации на примере генетического алгоритма, пояснена значимость адаптации и показана эффективность ее применения.

В качестве планов дальнейшей работы можно назвать создание адаптивных версий других алгоритмов, расширение круга решаемых ими задач, а также применение бионических алгоритмов для решения практических задач.

Литература:

1. *Мудров В.И.* Задача о коммивояжере: М.: изд-во «Знание», 1969. 62 с.
2. *Пападимитриу Х., Стайглиц К.* Комбинаторная оптимизация. Алгоритмы и сложность. Издательство «МИР», Москва, 1984.
3. *Lin S.* Computer Solutions of the Traveling Salesman Problem, BSTJ, 44, No. 10 (December 1965), p.2245-2269.
4. *Dorigo M.*, Gambardella L. M. Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem // IEEE Transactions on Evolutionary Computation, 1997, P. 53–66.
5. *Shah-Hosseini H.* Optimization with the Nature-Inspired Intelligent Water Drops Algorithm // Evolutionary Computation, Wellington Pinheiro dos Santos (Ed.), ISBN: 978-953-307-008-7, InTech, 2009.
6. *Holland J.H.* Adaptation in Natural and Artificial Systems // The University of Michigan Press, 1975.
7. *De Jong K.* An analysis of the behavior of a class of genetic adaptive systems // Unpublished PhD thesis, University of Michigan, Ann Arbor, 1975. (University Microfilms No. 76-9381).
8. *Schaefer, R., Cotta, C., Kołodziej, J., Rudolph, G.* Parallel Problem Solving from Nature – PPSN XI 11th International Conference, Kraków, Poland, September 11-15, 2010.
9. *Семенкин Е.С., Семенкина М.Е.* Проектирование ансамблей интеллектуальных информационных технологий самоконфигурируемым алгоритмом генетического программирования // Вестник Сибирского государственного аэрокосмического университета им. академика М.Ф. Решетнева. 2012. № 4. С. 89-96.