

МОДЕЛЬ РАЦИОНАЛИЗАЦИИ МАРШРУТОВ МОРСКОЙ ТРАНСПОРТИРОВКИ НА ОСНОВЕ МЕТОДА ГЕНЕТИЧЕСКИХ ХИМЕР

Г. Б. Попов, А. Л. Кузнецов (Санкт-Петербург)

Введение

В указанной математической задаче оптимальность решения в строгом значении понимается в смысле конкретного выбранного для оценки критерия. Таким критерием может быть длина маршрута, время в пути, стоимость перевозки и т.д. Классические (точные и результативные) методы поиска оптимального решения в основном представлены методами полного перебора и математического программирования, однако размер пространства поиска обычно не позволяет использовать эти инструменты. Кроме того, на практике действует жесткий принцип естественного экономического отбора, благодаря которому в реальной транспортной действительности решение относительно быстро оптимизируется самостоятельно. Как следствие, на установившихся направлениях перевозок глобальных транспортных сетей задача оптимизации маршрутов является давно решенной. Иначе дело обстоит на этапах масштабного освоения новых территорий и проектирования соответствующей сети морских перевозок для их транспортного обслуживания. Такой территорией сегодня, в частности, является зона Арктики и Северного морского пути. Из-за неустоявшихся грузопотоков, обусловленных разнородностью транспортных задач региона, проблема поиска наименее затратного кругового рейса с заходом в конкретные порты с конкретным грузом при перевозках в Арктическом регионе является актуальной.

Целью данного исследования является разработка инструмента, способного решать задачи оптимизации круговых рейсов, основываясь на критерии длины маршрута. Для достижения цели разработаны прототип модели на MS Excel и модель в специализированной среде моделирования AnyLogic. Адекватность обеих моделей доказана совпадением результатов моделирования с решениями, получаемыми методами полного перебора на примерах малой размерности. Адекватность специализированной модели прототипу доказывается совпадением результатов моделирования на репрезентативных примерах.

Описание метода и разработка прототипа модели

Известно, что рассматриваемая задача легко сводится к задаче коммивояжера (Travelling Salesman Problem – TSP), заключающейся в поиске оптимального маршрута через n пунктов, с заходом в пункты по одному разу и возвратом в исходный пункт [2]. Количество возможных маршрутов N , удовлетворяющих указанному условию, определяется выражением $N = n!$, где n – количество пунктов захода. С точки зрения теории вычислительной сложности, данная задача является трансвычислительной, так как при значениях $n \geq 67$ требуется обработка более 10^{94} вариантов решений методом полного перебора, что превышает предел Бремерманна, равный 10^{93} [1]. Для решения такого рода задач необходимо разрабатывать методы, не сводящиеся к схемам полного перебора и линейного программирования. Одним из наиболее эффективных методов в настоящее время представляется эволюционное программирование, а именно, метод генетических алгоритмов [3].

При использовании генетического алгоритма циклически повторяется определенный набор шагов. На каждом из этих шагов рассматривается лишь часть пространства поиска, с целью выявления наиболее рациональных решений из этой части. Рассматриваемая часть пространства поиска называется популяцией. Рациональность решений оценивается некоторой специальной функцией, называемой «функцией приспособ-

собственности». Решения, не удовлетворяющие условию отбора с помощью этой целевой функции, уничтожаются. Решения, удовлетворяющие условию отбора, должны быть скрещены между собой, чтобы на следующем шаге значение целевой функции (выражающей критерий оптимизации) было улучшено. Однако такое циклическое движение алгоритма может привести к вырождению системы и фиксации алгоритма на локальном экстремуме пространства поиска. Во избежание таких ситуаций в алгоритм добавляется механизм мутаций, позволяющий совершать случайные изменения отдельных частей решения (генов) с определенной частотой.

Как следствие, любая модель на основе генетического алгоритма состоит из следующих этапов:

1. инициализация – создание исходной популяции;
2. селекция – выбор решений, удовлетворяющих условию целевой функции;
3. применение генетических операторов – скрещивание и мутации;
4. остановка алгоритма – удовлетворение условию остановки.

Генетический алгоритм в описанной выше классической форме не удобен для решения задачи коммивояжера. Это обусловлено невозможностью «скрестить» гены двух родителей напрямую, т.к. имеется вероятность повторения отдельных пунктов захода и утраты иных в результирующем маршруте. Это противоречит постановке задачи, которая требует совершить заходы во все порты только один раз (рис. 1, а).

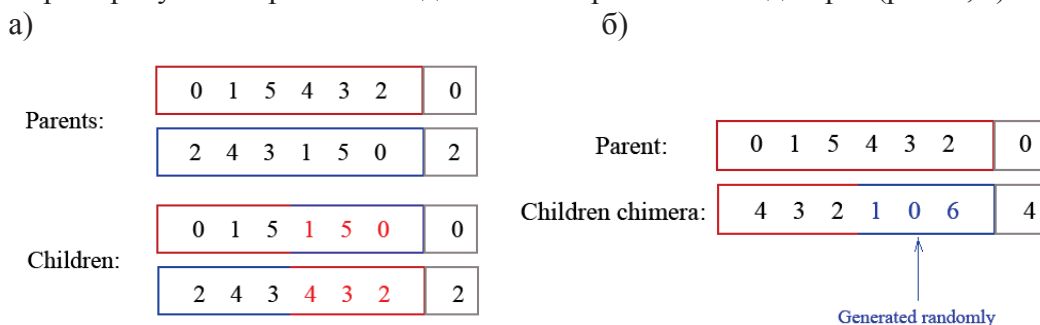


Рис. 1. Наследование генов: а – классический генетический алгоритм; б – метод генетических химер

Во избежание подобных случаев предлагается механизм создания т.н. химер, имеющих часть генома родителя, часть случайно сгенерированного генома (рис. 1, б). В таком случае от каждого родителя появится два наследующих решения: одно, повторяющее голову родителя и генерирующее случайный хвост, второе, повторяющее хвост родителя со случайной головой.

Прототип такого алгоритма реализован средствами MS Excel. Объяснением этому служит низкая трудоемкость создания прототипа, простота отладки и наглядность интерпретации получаемых результатов. Кроме того, сюда же можно отнести и стремление получить эталон, по которому можно будет провести отладку и верификацию модели, создаваемой в специализированной программной среде.

Работа полученной модели описывается следующим образом. Исходными данными являются координаты базовых точек – портов захода судна (рис. 2).

Управление			Таблица расстояний						
	x	y		1	2	3	4	5	6
ПОРТ 1	10	250	1	0	50	180	345	296	380
ПОРТ 2	15	300	2	50	0	228	335	305	378
ПОРТ 3	50	75	3	180	228	0	381	275	382
ПОРТ 4	350	310	4	345	335	381	0	130	72
ПОРТ 5	300	190	5	296	305	275	130	0	108
ПОРТ 6	390	250	6	380	378	382	72	108	0

Рис. 2. Задание координат портов и расчёт матрицы инцидентности

Следующим шагом создается начальная популяция случайных решений (рис. 3). В соответствии с постановкой задачи, конечный пункт любого маршрута совпадает с начальным, а поэтому в записи не указывается. В тестовом примере объем популяции выбран равным восьми. Для каждого варианта решения с помощью матрицы попарных расстояний рассчитывается длина соответствующего маршрута, на основании чего в созданной популяции отбирается половина наиболее рациональных решений.

			1	2	3	4	5	6				
	1		1	4	2	3	6	5	1695		1621,58	5
	2		1	4	2	5	3	6	2023		1655,97	3
	3		1	4	2	5	6	3	1656		1694,81	1
	4		1	4	2	6	3	5	2012		1736,67	7
	5		1	4	2	6	5	3	1622			
	6		1	4	3	2	5	6	1748			
	7		1	4	3	2	6	5	1737			
	8		1	4	3	5	2	6	2065			
			1	4	3	5	6	2				
1622	1677,26		1	4	2	6	5	3	1622		1	
			1	4	2	5	6	3	1656		2	
			1	4	2	3	6	5	1695		3	
			1	4	3	2	6	5	1737		4	

Рис. 3. Генерация случайных решений первого поколения и выбор 4-х лучших

Каждое следующее поколение генерируется аналогичным табличным блоком. Тиражирование такого блока на листе MS Excel позволяет создать любое количество шагов генетического алгоритма (рис. 4).

	5	1	1	4	2	3	5	6	1671		1185,47	4
	1	2	1	4	2	6	5	3	1622		1553,18	6
	3	3	1	4	2	3	5	6	1671		1621,58	2
	2	4	1	2	4	5	6	3	1185		1671,46	1
	2	5	1	4	2	6	3	5	2012			
	2	6	1	2	4	3	6	5	1553			
	2	7	1	4	3	6	2	5	2089			
	1	8	1	4	3	2	6	5	1737			
			1	2	4	5	6	3	1185		1	
1185	1505,69		1	2	4	3	6	5	1553		2	
			1	4	2	6	5	3	1622		3	
			1	5	2	3	4	6	1663		4	

Рис. 4. Блок поколения

Генерация недостающих частей решений средствами MS Excel сводится к перестановке элементов в соответствии с одним из пяти возможных протоколов. Схема перестановки по каждому протоколу представлена на рис. 5.

Protocol #	Permutation		
1	A	C	B
2	B	A	C
3	C	A	B
4	B	C	A
5	C	B	A

Рис. 5. Протоколы перестановки элементов решений

Номер протокола генерируется случайным образом, что, в свою очередь, реализует случайный механизм образования соответствующей недостающей части химеры. Так же, как и в исходной популяции, в каждом блоке формирования поколения выбираются четыре наименьших значения длины маршрута и номера решений, которые соответствуют этим значениям. Они будут отобраны для химерного размножения в следующем поколении (четыре последние строки). Минимум целевой функции, достигнутый в данном поколении, приводится в первом столбце, четвертой строке снизу на рис. 4. Один блок поколения равен одному шагу генетического алгоритма.

Во избежание вырождения системы, возможность которой указана ранее, в модель встраивается механизм мутаций. С этой целью два гена, произвольно выбранные из головы и хвоста одной из особей, меняются местами. Таким образом, в каждом поколении одна из четырех особей мутирует, предотвращая циклическое движение алгоритма вокруг локального минимума и создавая возможность для перехода к более перспективной локации.

Для проверки корректности работы вышеописанного алгоритма был составлен массив из 120 строк, включающий в себя все варианты решения (перестановки). Расчет длины маршрута для каждого из вариантов представляет собой метод полного перебора, на основании которого в данном массиве выбирается минимальное решение. Все решения, получаемые на каждом из k шагов прогона генетического алгоритма, для изучения эффективности сравниваются с этим минимальным значением (рис. 6). В приведенном на этом рисунке примере число шагов алгоритма составило $k = 15$, а реальный минимум был найден за $k = 4$ шага. Однако этот параметр зависит как от генерации случайных решений первого поколения, так и от генерации случайного значения протоколов перестановок. Результаты прототипирования были также сверены с результатами, полученными методом ветвей и границ, позволяющим отсекал подмножества решений, заведомо не содержащие оптимальные маршруты.

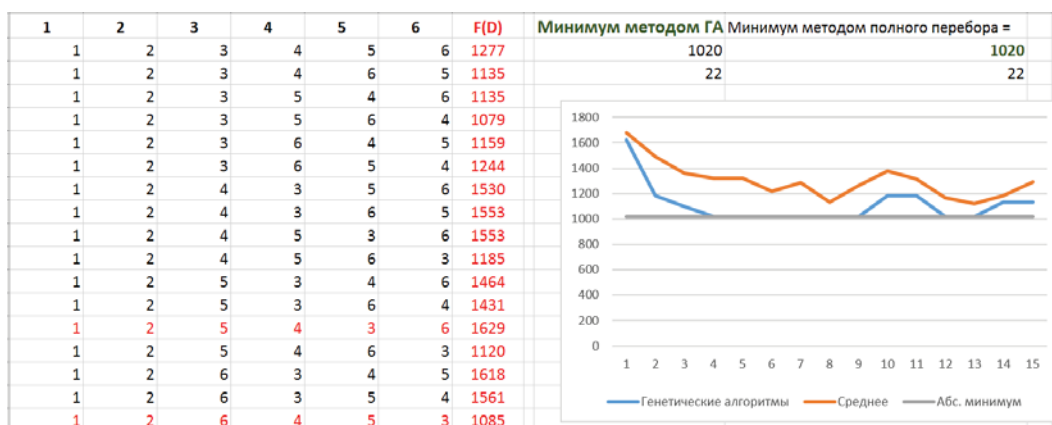


Рис. 6. Проверка методом полного перебора, сравнение эффективности методов

Моделирование в специализированной среде

После прототипирования и проверки адекватности метода средствами MS Excel была разработана более мощная модель в специализированной среде моделирования AnyLogic. Моделирование данной задачи в среде AnyLogic базируется на двух событиях, представленных на рис. 7. Событие *base* задаёт координаты для шести портов, рассчитывает матрицу дистанций между портами и генерирует нулевое поколение из 12 случайных маршрутов. Это событие выполняется один раз в момент времени, равный нулю. Второе событие – *algorithm* – является циклическим, выполняется каждую единицу времени (для удобства восприятия человеком) и служит основой генетического алгоритма модели. При выполнении этого события происходит подсчёт целевой функции для каждого варианта решения, выбор лучших шести вариантов в поколении, разделение химер, генерация случайных хвостов, мутации, а также отслеживание минимального значения за всё время прогона модели.

Результаты моделирования в среде AnyLogic проверены на адекватность сравнением с методом полного перебора и с результатами моделирования в MS Excel. При этом модель, построенная в специализированной среде, позволяет значительно расширить количество базовых точек и, самое главное, увеличить число шагов практически до бесконечности. Такие возможности позволяют проводить эксперименты, пример которых приведен на рис. 8. В этом примере использовались 20 портов захода, расположенных в виде кластеров на двух побережьях.

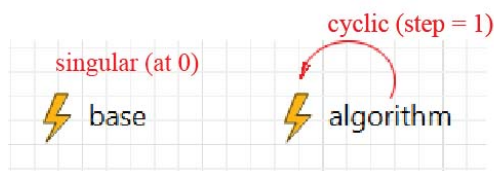


Рис.7. Базовые события в AnyLogic



Рис. 8. Результаты прогона модели для кластерного расположения 20-и портов

Выводы

Задача оптимизации играет важнейшую роль в регионах с развивающейся транспортной инфраструктурой и неустановившимися грузовыми потоками, обуславливая актуальность проведенного исследования. Одним из наиболее эффективных методов поиска решения является эвристическое программирование, а именно - использование генетических алгоритмов. На основе этого метода разработаны прототип и рабочая модель в специализированной среде моделирования и доказана их адекватность. Дальнейшим развитием в данной области представляется изучение возможностей рационализации перевозок между кластерами портов.

На основании ранее изложенного можно сделать следующие выводы:

1. предложен алгоритм решения задачи поиска наименьшего по длине пути, основанный на методах эволюционного программирования. Практический смысл задачи обусловлен ее применимостью для решения оптимизации в транспортной логистике;
2. адекватность алгоритма доказана на примере варианта постановки задачи, допускающего проверку нахождения оптимального решения методом полного перебора, т. е. реализации задачи коммивояжера для шести портов захода;
3. на основании исследованного прототипа в профессиональной среде моделирования создана модель, позволяющая изучать проблему для необходимых практически размеров пространства поиска.

Литература

1. **Bremermann H. J.** Optimization through evolution and recombination // Self-Organizing Systems. – USA, Washington DC, Spartan Books, 1962. – Pp. 93–106.
2. **Schrijver A.** On the history of combinatorial optimization (till 1960) // Handbooks in Operations Research and Management Science. – 2005. – Vol. 12. – Pp. 1–68. DOI: 10.1016/S0927-0507(05)12001-5.
3. **Holland J. H.** Adaptation in Natural and Artificial Systems / J. H. Holland. – UK, London: The MIT Press, 1992. – 211 p.
4. **Changdar C.** An efficient genetic algorithm for multi-objective solid travelling salesman problem under fuzziness/C. Changdar, G. S. Mahapatra, R. K. Pal // Swarm and Evolutionary Computation. – 2014. – Vol. 15. – Pp. 27–37.