

CHIMERICAL GENETIC ALGORITHM FOR SEA ROUTE RATIONALIZATION

A. L. Kuznetsov, A. V. Kirichenko, G. B. Popov

Admiral Makarov State University of Maritime and Inland Shipping,
St. Petersburg, Russian Federation

Solution for the route optimization problem is one of the basic problems of any transportation organization. Multi-criteria optimization is methodically impossible, but single criterion optimization may be a serious trouble, as the specified problem become transcomputational with initial data growth. At the same time, route optimization has important practical meaning, especially for such developing regions as Arctic region. Unlike developed regions, where most rational routes are already settled naturally (as a result of trials and errors), Arctic region has no such settled solution, which makes the optimal solution problem relevant. Geometric route length is selected as optimization criterion for problem description. Although this criterion may be drastically enhanced in practical implementation. Thus, the problem narrows down to the well-known travelling salesman problem (TSP). Evolutionary programming, based on genetic algorithms, appears to be a possible solution. It is indicated that genetic algorithms require modifications for adequate implementation for the problem. The authors suggest a modification, based on chimerical genetic method. The paper describes basics of the modified method, MS Excel prototype and a model, developed in a specialized modelling environment of AnyLogic 6. Prototype adequacy is proved by comparison with results of the brute force method for a small number of base points, which allows to make precise calculations. The model adequacy, in its turn, is proved by comparison with the tested prototype results. Specially planned testing experiments confirm both prototype and model adequacy. In conclusion, different port disposition examples are discussed.

Keywords: route rationalization, travelling salesman problem, genetic algorithms, adequacy proof.

For citation:

Kuznetsov, Aleksandr L., Aleksandr V. Kirichenko, and German B. Popov. "Chimerical genetic algorithm for sea route rationalization." *Vestnik Gosudarstvennogo universiteta morskogo i rechnogo flota imeni admirala S.O. Makarova* 9.3 (2017): 456–467. DOI: 10.21821/2309-5180-2017-9-3-456-467.

УДК 656.6

МЕТОД ГЕНЕТИЧЕСКИХ ХИМЕР ДЛЯ РЕШЕНИЯ ЗАДАЧИ РАЦИОНАЛИЗАЦИИ МАРШРУТОВ МОРСКОЙ ТРАНСПОРТИРОВКИ

А. Л. Кузнецов, А. В. Кириченко, Г. Б. Попов

ФГБОУ ВО «ГУМРФ имени адмирала С. О. Макарова», Санкт-Петербург,
Российская Федерация

Решение задачи оптимизации маршрутов является одной из основных проблем организации любой перевозки. Многокритериальная оптимизация невозможна методически, но и однокритериальная оптимизация может явиться серьезной проблемой, поскольку поставленная задача с увеличением объема исходных данных становится трансвычислительной. В то же время оптимизация маршрутов имеет важное практическое значение особенно для таких развивающихся регионов, как Арктика. В отличие от развитых регионов, где наиболее рациональные маршруты давно установились естественным путем (в результате проб и ошибок), в Арктическом регионе не существует устоявшегося решения, что делает задачу поиска оптимального решения крайне актуальной. Выбранным критерием оптимальности для изложения проблемы служит геометрическая длина маршрута, хотя в реальном применении понятие «расстояние» может быть существенно дополнено. Таким образом, задача сводится к известной задаче коммивояжера. В статье обсуждаются существующие методы поиска решения этой задачи. Возможным вариантом решения представляются методы эволюционного программирования, основывающегося на генетических алгоритмах. Показано, что генетические алгоритмы требуют модификации для корректного использования в решении поставленной задачи. Авторы статьи предлагают модификацию, основанную на методе генетических химер. В статье изложена основа модифицированного метода, описан реализованный в среде

MS Excel прототип алгоритма, а затем описана созданная в специализированной среде AnyLogic 6 рабочая модель. Адекватность прототипа доказана сравнением с результатами полного перебора для небольшого числа базовых точек, позволяющего точно вычислить правильное решение. Для рабочей модели, в свою очередь, устанавливается адекватность проверенному прототипу. Специально спланированные эксперименты по тестированию подтверждают адекватность прототипа и модели. В заключение рассмотрены различные примеры использования модели и расположения портов.

Ключевые слова: рационализация маршрутов, задача коммивояжера, генетические алгоритмы, доказательство адекватности.

Для цитирования:

Кузнецов А. Л. Метод генетических химер для решения задачи рационализации маршрутов морской транспортировки / А. Л. Кузнецов, А. В. Кириченко, Г. Б. Попов // Вестник Государственного университета морского и речного флота имени адмирала С. О. Макарова. — 2017. — Т. 9. — № 3. — С. 456–467. DOI: 10.21821/2309-5180-2017-9-3-456-467.

Введение (Introduction)

Среди множества логистических задач, рассматриваемых при организации любых перевозок, всегда в том или ином виде присутствует задача определения оптимального маршрута. Оптимальность в строгом математическом значении всегда понимается в смысле некоторого выбранного критерия. Им может быть наименьшая длина маршрута, минимальная стоимость перевозки, кратчайшее время в пути и т. д. Формальных алгоритмов многокритериальной оптимизации не существует, поэтому для принятия во внимание нескольких факторов обычно используются различные способы свертки линейных и нелинейных свертков — аддитивных или мультипликативных. Несмотря на все предпринимаемые усилия, в полученном интегральном критерии присутствует значительный субъективный фактор. Как результат, найденное решение оказывается половинчатым, далеким от оптимума по каждому из базовых критериев, и полученная оптимальность становится неочевидной. Эвристические «заплатки» в виде лексикографических методов, методов Парето и других способны улучшать решения в отдельных частных случаях, не решая проблему потери общности и универсальности. Кроме того, дискретный характер большинства переменных, входящих в критерии оптимизации, не позволяет применять эффективные и хорошо изученные классические методы функционального анализа, оставляя полный перебор вариантов единственным методом с доказуемой результативностью. В то же время размерность пространства поиска обычно исключает использование этого инструмента.

Как следствие, задача оптимизации в указанной постановке остается актуальной для математики как теоретической дисциплины, на практике в лучшем случае вытесняясь в сферу методов поддержки решений. Кроме того, при практическом применении действует жестокий принцип естественного экономического отбора, который позволяет быстро оптимизировать решение самостоятельно. В результате этого на установившихся направлениях, маршрутах и стрингах глобальных транспортных сетей задача оптимизации маршрутов давно решена самым убедительным образом, а возможные постановки в большинстве случаев сводятся к анализу чувствительности к отклонениям от сложившихся шаблонов.

В своей почти начальной постановке задача оптимизации маршрутов возникла на этапах масштабного освоения новых территорий и проектирования соответствующей сети морских перевозок для их транспортного обслуживания. Таким полигоном, в частности, сегодня является зона Арктики, Северного морского пути и прилегающих территорий. Ввиду неустановившегося характера грузопотоков, возникающих в ответ на многообразие разнообразных транспортных задач региона (обеспечение Северного завоза, организации вывоза добываемых природных ископаемых, обеспечение функций снабжения, поддержка экспедиционного завоза, оборонная деятельность и др.), данная задача приобретает новую актуальность [1] – [3]. В частности, возобновляется интерес и к самой простой и решенной в линейном судоходстве задаче: определению наименее

затратного кругового рейса с заходом в конкретные порты с конкретным грузом при перевозках в Арктическом регионе.

Целью данной статьи является создание алгоритмического инструмента, способного решать задачи оптимизации круговых судовых рейсов, базируясь на критерии длины маршрута. Для достижения данной цели необходимо решить следующие задачи: разработать прототип модели доступными неспециализированными средствами, доказать адекватность прототипа сверкой с результатами метода полного перебора, разработать модель в специализированной среде моделирования и доказать адекватность модели сверкой с результатами прототипирования и метода полного перебора.

Методы и материалы (Methods and Materials)

Известно, что рассматриваемую задачу можно свести к задаче коммивояжера (Travelling Salesman Problem — TSP), заключающейся в поиске наиболее выгодного маршрута, пролегающего через n пунктов, с заходом в пункты по одному разу и возвратом в исходный пункт [4], [5]. 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} бит информации методом полного перебора, что превышает предел Бремерманна, равный 10^{93} [6]. Этот факт позволяет отнести задачу к классу NP-сложных задач, что означает невозможность нахождения решения за полиномиальное время, т. е. время, полиномиально зависящее от объёма исходных данных [7]. Поэтому любые современные варианты поиска решения данной задачи основываются на использовании эвристических алгоритмов [8], [9] либо применении методов линейного программирования [10]. Одним из наиболее эффективных среди интенсивно исследуемых методов, по мнению авторов статьи, является эволюционное программирование, точнее, генетические алгоритмы.

Первые предположения касательно создания самообучающейся машины, базирующейся на принципах, схожих с принципами эволюционной теории, были высказаны А. Тьюрингом еще в 1950 г. Им были сформулированы основополагающие понятия эволюционного программирования: наследование, мутации, естественный отбор на основе целевой функции [11]. В дальнейшем ряд учёных развивали идеи, заложенные Тьюрингом, однако методологический базис оставался неизменным. Непосредственно генетические алгоритмы как часть эволюционного программирования получили распространение благодаря работам Джона Генри Холланда [12].

В основе использования любого генетического алгоритма лежит понятие *целевой функции* (Fitness) для каждого рассматриваемого варианта решения. Значение целевой функции, полученное для каждого варианта решения задачи, позволяет судить о том, насколько он подходит для ее решения.

В классическом варианте генетический алгоритм состоит из следующих шагов:

1. *Инициализация* (Initialization) — создание популяции возможных решений проблемы. Как правило, изначальные «особи» популяции, т. е. конкретные варианты решения, задаются случайно.

2. *Селекция* (Selection) — на каждом шаге выполнения алгоритма из популяции выбирается определённое количество «особей», лучше других удовлетворяющих условию целевой функции в данном поколении (на данном шаге). Только эти «особи» получают возможность скрещиваться между собой и давать потомство. В рассматриваемом случае это решения, несущие в себе части (хромосомы) родительских решений. Необходимо отметить, что количество «родителей» не обязательно равно двум. Некоторые исследователи считают, что увеличение количества родительских решений, предоставляющих свои хромосомы, благоприятно сказывается на качестве вычислений [13].

3. *Применение генетических операторов* (Genetic Operators) скрещивания (Crossover) и мутации (Mutations). «Особь», выбранная на этапе селекции, скрещивается между собой определенным образом, давая потомство, которое займёт место отмерших, т. е. наименее удовлетворяющих условиям целевой функции, «особей» в следующем поколении. Однако данный механизм может привести к так называемому «вырождению популяции», выражающемуся в невозможности преодоления локального минимума при помощи одних скрещиваний. Для предотвращения вырождения вводятся мутации — периодические случайные изменения «хромосом» в отдельных «особях».

4. *Остановка алгоритма* (Termination). Обычно условиями остановки (Termination Condition) выступают: нахождение решения, удовлетворяющего условиям задачи, достижение фиксированного количества шагов выполнения алгоритма, ручное вмешательство (Manual Inspection) и т. д.

Описанная общая схема алгоритма допускает модификации, позволяющие адаптировать его к потребностям конкретной задачи. Такие модификации рассмотрены далее в статье.

Предположим, что на некотором побережье имеются n портов, в которые необходимо совершить по одному судозаходу в рамках кругового рейса. Координаты портов в простейшем случае задаются в декартовой системе координат и измеряются в милях, т. е. расстояние между двумя любыми портами можно рассчитать по формуле

$$S_{i,j} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}, \quad (1)$$

где $S_{i,j}$ — расстояние между портом i и портом j ; x_i, x_j — координаты соответствующих портов по оси абсцисс; y_i, y_j — координаты соответствующих портов по оси ординат.

Используя формулу (1), можно построить матрицу расстояний между всеми портами, с помощью которой будет вычисляться длина каждого кругового рейса. Для простоты изложения не будем принимать во внимание факторы стоимости и общего времени рейса, тем самым полагая целевой функцией минимизацию длины маршрута рейса.

Каждый рейс состоит из последовательности портов, в которые необходимо совершить судозаход. Поскольку все рейсы круговые, первый и последний порты в последовательности всегда совпадают. Таким образом, для шести портов рейсы могут быть представлены комбинациями из шести неповторяющихся чисел от одного до шести. Поскольку для целевой функции безразлично, начиная с какого порта идет последовательность, количество комбинаций, в которое будут входить все уникальные варианты последовательности портов, определяется выражением $N = (n - 1)!$ При выбранном для демонстрационного примера числе портов $n = 6$ количество различных маршрутов составит 120. Найти маршрут наименьшей длины из такого количества комбинаций оказывается возможным при помощи *метода полного перебора*, что позволит проверить корректность работы алгоритма.

Необходимо отметить, что генетический алгоритм в классической форме не совсем удобен для решения задачи коммивояжера. Причина этого кроется в невозможности использования для скрещивания геномов двух «родителей» напрямую, поскольку «гены» (конкретные порты в маршруте) одного «родителя» могут быть уже использованы в «генах» другого, что противоречит условию захода во все порты только один раз (на рис. 1, *а* показаны повторяющиеся при наследовании «гены»). Для предотвращения подобных случаев предложен механизм создания «химер» — «особей» следующего поколения, наполовину повторяющих геном «родителя», наполовину генерирующих собственный геном из недостающих элементов (на рис. 1, *б* выделены «гены», сформированные случайным образом). От каждого «однополого родителя», таким образом, формируются две «особи»: одна с «головой родителя» и случайным «хвостом», другая — с «хвостом родителя» и случайной «головой», что и объясняет их название.

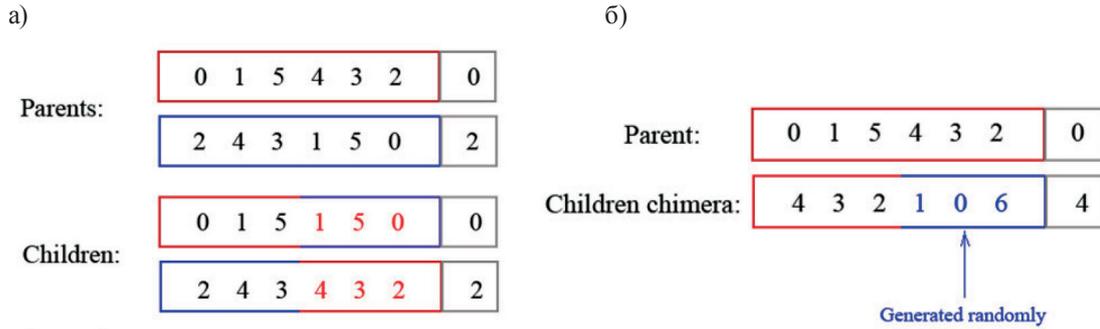


Рис. 1. Механизм наследования генов:
а — классический генетический алгоритм; б — метод генетических химер

Прототип указанной модификации генетического алгоритма можно реализовать средствами MS Excel. С этой целью необходимо задать координаты портов в декартовой системе и сформировать автоматически вычисляемую матрицу расстояний между ними (рис. 2).

	A	B	C	D	E	F	G	H	I	J	K
1		x	y			1	2	3	4	5	6
2	ПОРТ 1	500	100		1	0	206	451	430	488	187
3	ПОРТ 2	450	300		2	206	0	459	341	444	79
4	ПОРТ 3	50	75		3	451	459	0	214	119	382
5	ПОРТ 4	110	280		4	430	341	214	0	127	281
6	ПОРТ 5	20	190		5	488	444	119	127	0	374
7	ПОРТ 6	389	250		6	187	79	382	281	374	0

Рис. 2. Задание координат портов
и расчёт матрицы расстояний между ними

Далее необходимо задать начальную популяцию, т. е. сформировать случайные начальные варианты маршрутов круговых рейсов (конечный пункт совпадает с первым, поэтому не указывается). В расчетном примере объем популяции выбран равным восьми, и начальными значениями могут служить любые неповторяющиеся перестановки чисел от двух до шести, которые являются «особями» нулевого поколения. Для каждой «особи популяции» с помощью матрицы попарных расстояний может быть рассчитана длина соответствующего маршрута. На основании этого из созданных «особей» отбирается половина наиболее приспособленных (выделяются четыре решения с минимальным значением целевой функции, т. е. с минимальной длиной маршрута). На рис. 3 показана начальная «популяция». Отбираемые «особи» представлены нижними цветными строками.

		1	2	3	4	5	6				
	1	1	4	2	3	6	5	2473		1471,307	8
	2	1	4	2	5	3	6	1901		1792,384	5
	3	1	4	2	5	6	3	2420		1837,714	4
	4	1	4	2	6	3	5	1838		1900,927	2
	5	1	4	2	6	5	3	1792			
	6	1	4	3	2	5	6	2106			
	7	1	4	3	2	6	5	2043			
	8	1	4	3	5	2	6	1471			
		1	4	3	5	6	2				
1471	1750,58	1	4	3	5	2	6	1471	1		
		1	4	2	6	5	3	1792	2		
		1	4	2	6	3	5	1838	3		
		1	4	2	5	3	6	1901	4		

Рис. 3. Генерация случайных «особей» первого поколения
и выбор четырех лучших из них

После этого каждое следующее поколение генерируется аналогичным табличным блоком, несущим в себе все функции по преобразованию. Тиражирование такого блока в листе MS Excel позволяет быстро и легко создать любое количество шагов генетического алгоритма. Такой «блок поколения» приведен на рис. 4.

	4	1	1	4	3	2	6	5	2043			1360,98	8
	1	2	1	4	3	5	2	6	1471			1471,307	2
	1	3	1	4	2	6	3	5	1838			1792,384	4
	1	4	1	4	2	6	5	3	1792			1837,714	3
	4	5	1	4	2	3	5	6	1908				
	1	6	1	4	2	6	3	5	1838				
	1	7	1	4	2	5	6	3	2420				
	2	8	1	2	4	5	3	6	1361				
1361	1627,36		1	2	4	5	3	6	1361				1
			1	4	3	5	2	6	1471				2
			1	4	2	6	5	3	1792				3
			1	3	2	6	4	5	1885				4

Рис. 4. «Блок поколения»

Как видно из рис. 4, «особи» соответствующих цветов, отбираемые из предыдущего поколения, расщепляются на «голову» и «хвост», а их недостающие части генерируются случайным образом. Генерация 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 «генов»

Номер протокола генерируется случайным образом (этот номер содержится во втором столбце блока рис. 4), что и реализует случайный механизм образования соответствующей недостающей части «химеры». Так же, как и в исходной популяции, в каждом блоке поколения отыскиваются четыре наименьших значения длины маршрута (второй справа столбец блока на рис. 4) и номера «особей», 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. Как видно из ранее изложенного, каждый блок поколения равен одному шагу генетического алгоритма.

Во избежание «вырождения» системы, возможность которого была указана ранее, используется 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 можно составить массив из 120 строк, включающий все уникальные 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

ма составило $k = 15$, а реальный минимум, который был найден, составил $k = 13$ шагов. Однако этот параметр зависит как от генерации случайных «особей» нулевого поколения, так и от генерации случайного значения протоколов перестановок. Результаты прототипирования были также сверены с результатами, полученными *методом ветвей и границ*, позволяющим отсекалть подмножества решений, заведомо не содержащие оптимальные маршруты.

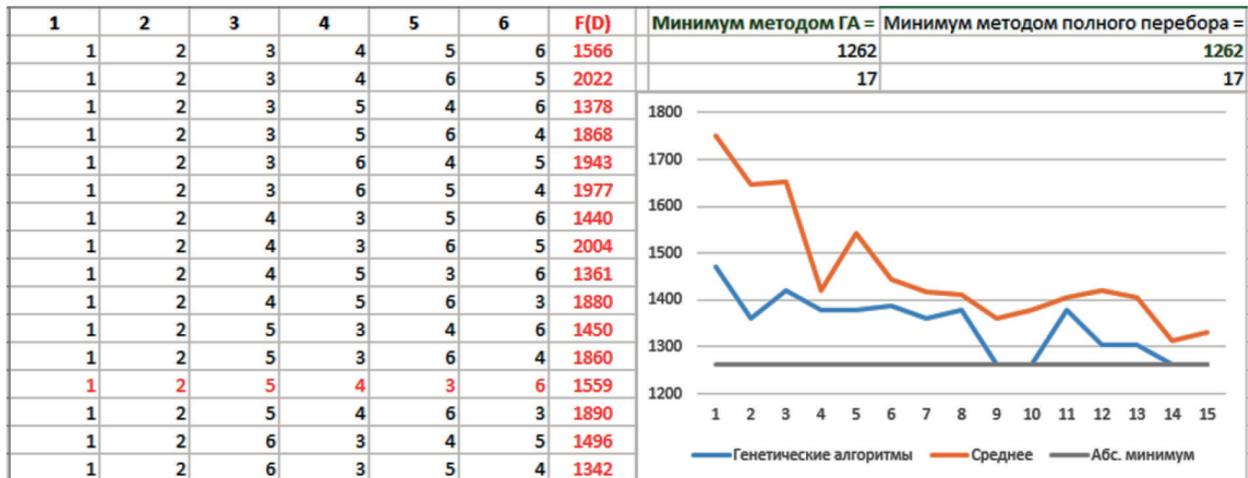


Рис. 6. Проверка методом полного перебора, сравнение эффективности методов

После прототипирования и проверки адекватности метода средствами MS Excel была разработана более мощная модель в специализированной среде моделирования AnyLogic 6 (информация о лицензии: USB-ключ #00559). Моделирование данной задачи в среде AnyLogic 6 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 каждой «особи», выбор лучших шести «особей» в поколении, разделение «особей» на «химеры», генерация случайных «хвостов химер», мутации, а также отслеживание минимального значения в течение всего времени прогона модели.

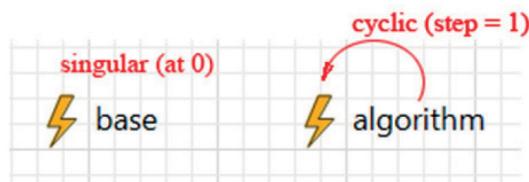


Рис. 7. Базовые события в AnyLogic 6

Для удобства сравнения результатов начальные данные в этой модели совпадают с начальными данными прототипа в MS Excel (см. рис. 2 и 8).

```
pts[0].setPos(500, 100);
pts[1].setPos(450, 300);
pts[2].setPos(50, 75);
pts[3].setPos(110, 280);
pts[4].setPos(20, 190);
pts[5].setPos(389, 250);
```

Рис. 8. Задание начальных координат портов в AnyLogic 6

Принцип работы модели (рис. 9) аналогичен принципу работы прототипа. Основным различием является практически неограниченное количество шагов при прогоне модели, что позволяет проводить более глубокие вычисления. Кроме того, нет необходимости копировать блоки поколений, чтобы увеличить количество шагов, что значительно упрощает моделирование.



Рис. 9. Общий алгоритм работы модели на AnyLogic 6

Мутации в модели представлены повторяемой перестановкой двух случайных «генов» в «особи». Число повторов выбрано равным $0,25n$, где n — число портов захода. Данное значение было найдено эмпирическим путем как благотворно влияющее на скорость выполнения модели. Количество «мутирующих особей» задаётся в виде доли от размера «популяции».

Результаты (Results)

На рис. 10 представлены результаты двух прогонов модели, в которых исходные данные взяты из прототипа модели (см. рис. 8) в первом случае (рис. 10, а) без мутаций, во втором (рис. 10, б) — с мутациями. В первом случае система не смогла найти значения меньше 1303, во втором случае было найдено значение 1263, являющееся абсолютным минимумом по результатам проверки *методом полного перебора*.

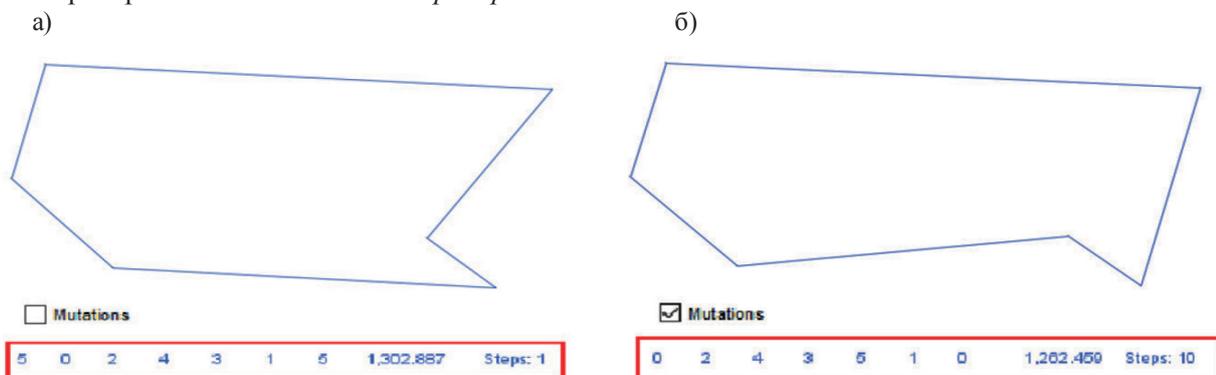


Рис. 10. Результаты прогона модели на AnyLogic 6:
 а — без «мутаций»; б — с «мутациями»

Помимо экспериментов, повторяющих поведение прототипа, были проведены эксперименты с большим числом портов захода. На рис. 11, а представлены результаты прогона модели

для 20 портов захода с их круговым расположением. Количество «особей в популяции» установлено вдвое большим, чем число портов захода. Мутации распространялись на 30 % «особей». Модели потребовалось 3 184 346 шагов для нахождения правильного решения для 20 точек. Однако в реальности маршруты круговых рейсов представляют из себя, скорее, ломаную линию, при этом порты могут быть расположены в виде кластеров на разных побережьях. В связи с этим был проведен эксперимент, представленный на рис. 11, б. Порты сгруппированы в два кластера со случайными отклонениями от заданных координат для каждого кластера таким образом, что их расположение формирует линию побережья. Количество «особей» и доля мутировавших «особей» такие же, как и в предыдущей серии экспериментов. Для нахождения оптимального решения модели потребовалось 1 538 140 шагов.

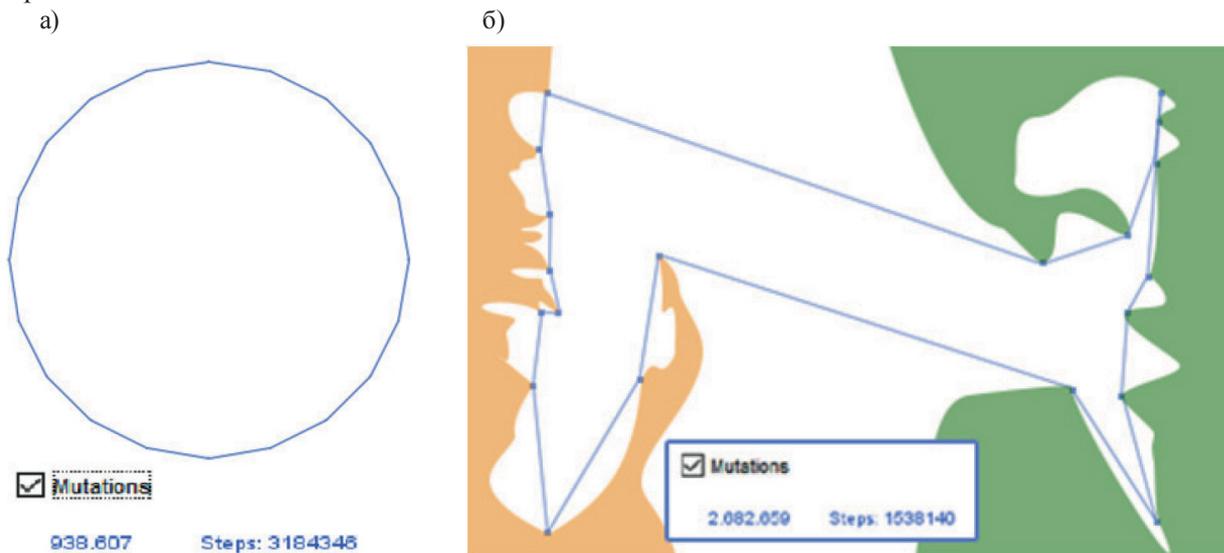


Рис. 11. Результаты прогона модели на AnyLogic 6 для 20 портов:
а — с круговым расположением; б — с кластерным расположением

Обсуждение (Discussion)

Результаты, приведенные на рис. 10, свидетельствуют о необходимости включения механизма мутаций в любую модель, основанную на генетических алгоритмах. Если данный механизм не проработан, велика вероятность того, что модель найдет лишь локальный минимум, как это видно на рис. 10, а. Однако влияние мутаций на работу модели на этом не заканчивается. Для более глубокого изучения влияния этого механизма на результаты прогона модели добавлена возможность его выключения во время прогона. В результате моделирования было выявлено, что постоянные мутации отрицательно влияют на скорость нахождения моделью абсолютного минимума. Периоды мутаций необходимо чередовать с периодами безмутационной эволюции при достаточно большом количестве портов захода. Кроме того, важным фактором является калибровка системы мутаций, а именно: доля популяции, на которую они влияют, частота мутаций, число мутирующих «генов» и т. д. Рис. 10 также демонстрирует, что результаты моделирования при помощи AnyLogic 6 совпадают с результатами прототипирования при помощи MS Excel, а также результатами, полученными методом полного перебора, что подтверждает адекватность модели.

Для изучения поведения модели была проведена серия экспериментов с большим числом портов захода, которые для обеспечения наглядности были расположены на одной окружности, как показано на рис. 11, а. Очевидно, что при таком расположении портов кратчайший путь через все точки равен периметру фигуры, вписываемой в окружность, и не может быть

больше длины окружности. Следовательно, моделирование будет успешно проведено, когда модель найдёт путь, равный периметру обхода всех точек, как показано на рис. 11. Значительное увеличение количества шагов, затрачиваемых моделью на поиск оптимального решения по сравнению с рис. 10, объясняется природой задачи, рассмотренной ранее в данном исследовании.

На рис. 11, б приведен пример, более тесно связанный с реальностью, в котором некоторые 20 портов сформированы в два кластера по 10 портов на двух разных побережьях. Рис. 10 и 11 с матрицами расстояний между портами доступны для скачивания по ссылке: <https://journal.gumrf.ru/files/upload/art/9-3-1KKP.rar>.

Заключение (Conclusion)

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

Таким образом, на основании ранее изложенного можно сделать следующие выводы:

1. Предложен алгоритм решения задачи поиска наименьшего по длине пути, основанный на современных методах эволюционного программирования. Практический смысл задачи обусловлен ее применимостью для решения проблем 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.

СПИСОК ЛИТЕРАТУРЫ

1. Лебедев Г. В. Перспективы развития арктической морской транспортной системы / Г. В. Лебедев, Г. Е. Румянцев // Вестник Государственного университета морского и речного флота имени адмирала С. О. Макарова. — 2014. — № 3 (25). — С. 179–189.
2. Ботнарюк М. В. Анализ состояния и перспективы развития Северного морского пути / М. В. Ботнарюк // Вестник Государственного университета морского и речного флота имени адмирала С. О. Макарова. — 2013. — № 1 (20). — С. 186–194.
3. Давыденко А. А. Обоснование концепции создания транспортной системы совместного использования в арктическом регионе Российской Федерации / А. А. Давыденко, А. В. Кириченко, А. А. Кузнецов // Вестник Государственного университета морского и речного флота имени адмирала С. О. Макарова. — 2015. — № 3 (31). — С. 120–124. DOI: 10.21821/2309-5180-2015-7-3-120-124.
4. Schrijver A. On the history of combinatorial optimization (till 1960) / A. Schrijver // Handbooks in Operations Research and Management Science. — 2005. — Vol. 12. — Pp. 1–68. DOI: 10.1016/S0927-0507(05)12001-5.
5. García A. Polynomially solvable cases of the bipartite travelling salesman problem / A. García, J. Tejel // European Journal of Operational Research. — 2017. — Vol. 257. — Is. 2. — Pp. 429–438. DOI: 10.1016/j.ejor.2016.07.060.
6. Bremermann H. J. Optimization through evolution and recombination / H. J. Bremermann // Self-Organizing Systems. — USA, Washington DC, Spartan Books, 1962. — Pp. 93–106.

7. Knuth D. E. Postscript about NP-hard problems / D. E. Knuth // ACM SIGACT News. — 1974. — Vol. 6. — Is. 2. — Pp. 15–16. DOI: 10.1145/1008304.1008305.
8. 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. DOI: 10.1016/j.swevo.2013.11.001.
9. Nguyen H. D. Implementation of an effective hybrid GA for large-scale travelling salesman problems / H. D. Nguyen, I. Yoshihara, K. Yamamori, M. Yasunaga // IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics). — 2007. — Vol. 37. — Is. 1. — Pp. 92–99. DOI: 10.1109/TSMCB.2006.880136.
10. Morton G. A contribution to the “Travelling-salesman” Problem / G. Morton, A. H. Land // Journal of the Royal Statistics Society. Series B (Methodological). — 1955. — Vol. 17. — No. 2. — Pp. 185–203.
11. Turing A. M. Computer machinery and intelligence / A. M. Turing // Mind. — 1950. — Vol. 59. — No. 236. — Pp. 433–460. DOI: 10.1093/mind/LIX.236.433.
12. Holland J. H. Adaptation in Natural and Artificial Systems / J. H. Holland. — UK, London: The MIT Press, 1992. — 211 p.
13. Eiben A. E. Genetic algorithms with multi-parent recombination / A. E. Eiben, P.-E. Raue, Zs. Ruttkay // International Conference on Parallel Problem Solving from Nature. — Springer Berlin Heidelberg, 1994. — Pp. 78–87. DOI: 10.1007/3-540-58484-6_252.

REFERENCES

1. Lebedev, G. V., and G. E. Rumyantsev. “The prospects of arctic marine transportation system’s development.” *Vestnik Gosudarstvennogo universiteta morskogo i rechnogo flota imeni admirala S.O. Makarova* 3(25) (2014): 179–189.
2. Botnariuk, M. V. “Analysis and perspectives of North sea way development.” *Vestnik Gosudarstvennogo universiteta morskogo i rechnogo flota imeni admirala S.O. Makarova* 1(20) (2013): 186–194.
3. Davydenko, A. A., A. V. Kirichenko, and A. L. Kuznetsov. “Theoretical justification for the development of multi-users transportation system in the Arctic region of Russian Federation.” *Vestnik Gosudarstvennogo universiteta morskogo i rechnogo flota imeni admirala S.O. Makarova* 3(31) (2015): 120–124. DOI: 10.21821/2309-5180-2015-7-3-120-124.
4. Schrijver, A. “On the history of combinatorial optimization (till 1960)” *Handbooks in Operations Research and Management Science* 12 (2005): 1–68. DOI: 10.1016/S0927-0507(05)12001-5
5. García, A., and J. Tejel. “Polynomially solvable cases of the bipartite travelling salesman problem.” *European Journal of Operational Research* 257.2 (2017): 429–438. DOI: 10.1016/j.ejor.2016.07.060.
6. Bremermann, H. J. “Optimization through evolution and recombination.” *Self-Organizing Systems*; edited by M. C. Yovits, G. T. Jacobi and G. D. Goldstine. Washington, DC: Spartan Books, 1962: 93–106.
7. Knuth, D. E. “Postscript about NP-hard problems.” *ACM SIGACT News* 6.2 (1974): 15–16. DOI: 10.1145/1008304.1008305.
8. Changdar, C., G. S. Mahapatra, and R. K. Pal. “An efficient genetic algorithm for multi-objective solid travelling salesman problem under fuzziness.” *Swarm and Evolutionary Computation* 15 (2014): 27–37. DOI: 10.1016/j.swevo.2013.11.001.
9. Nguyen, H. D., I. Yoshihara, K. Yamamori, and M. Yasunaga. “Implementation of an effective hybrid GA for large-scale travelling salesman problem.” *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 37.1 (2007): 92–99. DOI: 10.1109/TSMCB.2006.880136.
10. Morton, G., and A. H. Land. “A contribution to the “Travelling-Salesman” Problem.” *Journal of the Royal Statistics Society. Series B (methodological)* 17.2 (1955): 185–203.
11. Turing, A. M. “Computer machinery and intelligence.” *Mind* 59.236 (1950): 433–460. DOI: 10.1093/mind/LIX.236.433.
12. Holland, J. H. *Adaptation in Natural and Artificial Systems*. London: The MIT Press, 1992.
13. Eiben, Agoston E., P.-E. Raue, and Zs Ruttkay. “Genetic algorithms with multi-parent recombination.” *International Conference on Parallel Problem Solving from Nature*. Springer Berlin Heidelberg, 1994: 78–87. DOI: 10.1007/3-540-58484-6_252.

ИНФОРМАЦИЯ ОБ АВТОРАХ

Кузнецов Александр Львович —
доктор технических наук, профессор
ФГБОУ ВО «ГУМРФ имени
адмирала С. О. Макарова»
198035, Российская Федерация, г. Санкт-Петербург,
ул. Двинская, 5/7
e-mail: thunder1950@yandex.ru, kaf_pgt@gumrf.ru
Кириченко Александр Викторович —
доктор технических наук, профессор
ФГБОУ ВО «ГУМРФ имени
адмирала С. О. Макарова»
198035, Российская Федерация, г. Санкт-Петербург,
ул. Двинская, 5/7
e-mail: a.v.kirichenko@mail.ru,
KirichenkoAV@gumrf.ru
Попов Герман Борисович — аспирант
Научный руководитель:
Кузнецов Александр Львович
ФГБОУ ВО «ГУМРФ имени
адмирала С. О. Макарова»
198035, Российская Федерация, г. Санкт-Петербург,
ул. Двинская, 5/7
e-mail: german_bp@mail.ru

INFORMATION ABOUT THE AUTHORS

Kuznetsov, Aleksandr L. —
Dr. of Technical Sciences, professor
Admiral Makarov State University of Maritime
and Inland Shipping
5/7 Dvinskaya Str., St. Petersburg 198035,
Russian Federation
e-mail: thunder1950@yandex.ru, kaf_pgt@gumrf.ru
Kirichenko, Aleksandr V. —
Dr. of Technical Sciences, professor
Admiral Makarov State University of Maritime
and Inland Shipping
5/7 Dvinskaya Str., St. Petersburg 198035,
Russian Federation
e-mail: a.v.kirichenko@mail.ru,
KirichenkoAV@gumrf.ru
Popov, German B. — Postgraduate
Supervisor:
Kuznetsov, Aleksandr L.
Admiral Makarov state University of Maritime
and Inland Shipping
5/7 Dvinskaya Str, St. Petersburg, 198035,
Russian Federation
e-mail: german_bp@mail.ru

*Статья поступила в редакцию 27 апреля 2017 г.
Received: April 27, 2017.*