

# РЕШЕНИЕ ОПТИМИЗАЦИОННЫХ ЗАДАЧ ЭВОЛЮЦИОННО-ГЕНЕТИЧЕСКИМИ АЛГОРИТМАМИ

## Основные понятия генетических алгоритмов

При описании генетических алгоритмов используются определения, заимствованные из генетики. Например, речь идет о *популяции особей*, а в качестве базовых понятий применяются *ген*, *хромосома*, *генотип*, *фенотип*, *аллель*. Также используются соответствующие этим терминам определения из технического лексикона, в частности, *цепь*, *двоичная последовательность*, *структура*.

**Популяция** - это конечное множество *особей*.

**Особь**, входящие в популяцию, в генетических алгоритмах представляются хромосомами с закодированным в них множествами *параметров задачи*, т.е. решений, которые иначе называются *точками в пространстве поиска* (*search points*). В некоторых работах особи называются *организмами*.

**Хромосомы** (другие названия - *цепочки* или *кодированные последовательности*) - это упорядоченные *последовательности генов*.

**Ген** (также называемый *свойством*, *знаком* или *детектором*) - это атомарный элемент *генотипа*, в частности, хромосомы.

**Генотип** или **структура** - это набор хромосом данной особи. Следовательно, особями популяции могут быть генотипы либо единичные хромосомы (в довольно распространенном случае, когда генотип состоит из одной хромосомы).

**Фенотип** - это набор значений, соответствующих данному генотипу, т.е. *декодированная структура* или множество *параметров задачи* {*решение*, *точка пространства поиска*).

**Аллель** - это значение конкретного гена, также определяемое как *значение свойства* или *вариант свойства*.

**Локус** или *позиция* указывает место размещения данного гена в хромосоме (цепочке). Множество позиций генов - это *локи*.

Очень важным понятием в генетических алгоритмах считается *функция приспособленности* (*fitness function*), иначе называемая *функцией оценки*. Она представляет меру приспособленности данной особи в популяции. Эта функция играет важнейшую роль, поскольку позволяет оценить степень приспособленности конкретных особей в популяции и выбрать из них наиболее приспособленные (т.е. имеющие наибольшие значения функции приспособленности) в соответствии с эволюционным принципом выживания «сильнейших» (лучше всего приспособившихся). Функция приспособленности также получила свое название непосредственно из генетики. Она

оказывает сильное влияние на функционирование генетических алгоритмов и должна иметь точное и корректное определение. В задачах оптимизации функция приспособленности, как правило, оптимизируется (точнее говоря, максимизируется) и называется *целевой функцией*. В задачах минимизации целевая функция преобразуется, и проблема сводится к максимизации. В теории управления функция приспособленности может принимать вид *функции погрешности*, а в теории игр - *стоимостной функции*. На каждой итерации генетического алгоритма приспособленность каждой особи данной популяции оценивается при помощи функции приспособленности, и на этой основе создается следующая популяция особей, составляющих множество потенциальных решений проблемы, например, задачи оптимизации.

Очередная популяция в генетическом алгоритме называется *поколением*, а к вновь создаваемой популяции особей применяется термин «новое поколение» или «поколение потомков».

### Пример 1

Рассмотрим функцию

$$f(x) = 2x^2 + 1 \quad (1)$$

и допустим, что  $x$  принимает целые значения из интервала от 0 до 15. Задача оптимизации этой функции заключается в перемещении по пространству, состоящему из 16 точек со значениями 0, 1, ..., 15 для обнаружения той точки, в которой функция принимает максимальное (или минимальное) значение.

В этом случае в качестве *параметра задачи* выступает переменная  $x$ . Множество  $\{0, 1, \dots, 15\}$  составляет *пространство поиска* и одновременно - множество потенциальных решений задачи. Каждое из 16 чисел, принадлежащих к этому множеству, называется *точкой пространства поиска*, *решением*, *значением параметра*, *фенотипом*. Следует отметить, что решение, оптимизирующее функцию, называется *наилучшим* или *оптимальным* решением. Значения параметра  $x$  от 0 до 15 можно закодировать следующим образом:

0000 0001 0010 0011 0100 0101 0110 0111

1000 1001 1010 1011 1100 1101 1110 1111

Это широко известный способ двоичного кодирования, связанный с записью десятичных цифр в двоичной системе. Представленные *кодовые последовательности* также называются *цепями* или *хромосомами*. В рассматриваемом примере они выступают и в роли *генотипов*. Каждая из хромосом состоит из 4 генов (иначе можно сказать, что двоичные последовательности состоят из 4 битов). Значение гена в конкретной позиции называется аллелью, принимающей в данном случае значения 0

или 1. *Популяция* состоит из *особей*, выбираемых среди этих 16 *хромосом*. Примером популяции с численностью, равной 6, может быть, например, множество хромосом {0010, 0101, 0111, 1001, 1100, 1110}, представляющих собой закодированную форму следующих фенотипов: {2, 5, 7, 9, 12, 14}. *Функция приспособленности* в этом примере задается выражением (1). Приспособленность отдельных хромосом в популяции определяется значением этой функции для значений  $x$ , соответствующих этим хромосомам, т.е. для *фенотипов*, соответствующих определенным *генотипам*.

## Пример 2

Рассмотрим нейронную сеть, представленную на рис. 1, для которой необходимо подобрать оптимальные веса  $w_{11}, w_{12}, w_{21}, w_{22}, w_{31}, w_{32}$  и  $w_{10}, w_{20}, w_{30}$ , минимизирующих значение погрешности

$$Q = \frac{1}{4} \sum_{i=1}^4 (d_i - y_i)^2.$$

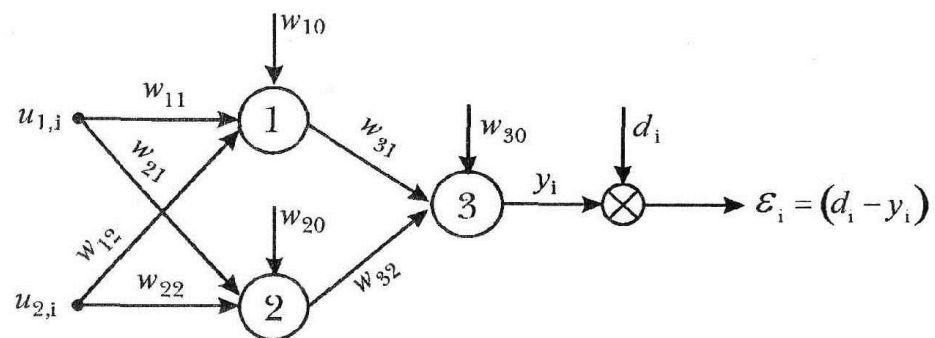


Рис.1. Нейронная сеть, реализующая операцию XOR

Это сеть, реализующая систему XOR, поэтому значения  $u_{1,i}$  и  $u_{2,i}$ , а также  $d_i$  для  $i = 1, \dots, 4$  принимают значения, приведенные в табл. 1.

Таблица 1

| $u_1$ | $u_2$ | $d = XOR(u_1, u_2)$ |
|-------|-------|---------------------|
| +1    | +1    | -1                  |
| +1    | -1    | +1                  |
| -1    | +1    | +1                  |
| -1    | -1    | -1                  |

В качестве параметров рассматриваемой задачи выступают перечисленные выше веса, т.е. задача имеет 9 параметров. В данном примере набор из этих 9 параметров определяет точку пространства поиска и, следовательно, представляет собой

возможное решение. Допустим, что веса могут принимать значения из интервала  $[-10, 10]$ . Тогда каждая хромосома будет комбинацией из 9 двоичных последовательностей, кодирующих конкретные веса. Это, очевидно, и есть генотипы. Соответствующие им фенотипы представлены значениями отдельных весов, т.е. множествами соответствующих действительных чисел из интервала  $[-10, 10]$ .

В приведенных примерах хромосомы и генотипы обозначают одно и то же - фенотипы особей популяции, закодированные в форме упорядоченных последовательностей генов со значениями (аллелями), равными 0 или 1.

В генетике генотип задает генетическую структуру особи, которая может включать более одной хромосомы. Например, клетки человека содержат 46 хромосом. В генетических алгоритмах генотип определяется аналогичным образом, однако чаще всего он состоит всего из одной хромосомы, которая и выступает в роли особи популяции.

Длина хромосом зависит от условий задачи. Следует заметить, что в естественных организмах хромосома может состоять из нескольких сотен и даже тысяч генов. У человека имеется около 100 000 генов, хотя их точное количество до сих пор неизвестно.

### **Классический генетический алгоритм**

Основной (классический) генетический алгоритм (также называемый элементарным или простым генетическим алгоритмом) состоит из следующих шагов:

- 1) инициализация, или выбор исходной популяции хромосом;
- 2) оценка приспособленности хромосом в популяции;
- 3) проверка условия остановки алгоритма;
- 4) селекция хромосом;
- 5) применение генетических операторов;
- 6) формирование новой популяции;
- 7) выбор «наилучшей» хромосомы.

Блок-схема основного генетического алгоритма изображена на рис.2. Рассмотрим конкретные этапы этого алгоритма более подробно с использованием дополнительных подробностей, представленных на рис. 3.

**Инициализация**, т.е. формирование исходной популяции, заключается в случайном выборе заданного количества хромосом (особей), представляемых двоичными последовательностями фиксированной длины.

**Оценивание приспособленности** хромосом в популяции состоит в расчете функции приспособленности для каждой хромосомы этой популяции. Чем больше значение этой функции, тем выше «качество» хромосомы. Форма функции приспособленности зависит от характера решаемой задачи. Предполагается, что функция приспособленности всегда принимает неотрицательные значения и, кроме того, что для решения оптимизационной задачи требуется максимизировать эту функцию. Если исходная форма функции приспособленности не удовлетворяет этим условиям, то выполняется соответствующее преобразование (например, задачу минимизации функции можно легко свести к задаче максимизации).

**Проверка условия остановки алгоритма.** Определение условия остановки генетического алгоритма зависит от его конкретного применения. В оптимизационных задачах, если известно максимальное (или минимальное) значение функции приспособленности, то остановка алгоритма может произойти после достижения ожидаемого оптимального значения, возможно - с заданной точностью. Остановка алгоритма также может произойти в случае, когда его выполнение не приводит к улучшению уже достигнутого значения. Алгоритм может быть остановлен по истечении определенного времени выполнения либо после выполнения заданного количества итераций. Если условие остановки выполнено, то производится переход к завершающему этапу выбора «наилучшей» хромосомы. В противном случае на следующем шаге выполняется селекция.



Рис.2. Блок-схема генетического алгоритма

**Селекция хромосом** заключается в выборе (по рассчитанным на втором этапе значениям функции приспособленности) тех хромосом, которые будут участвовать в создании потомков для следующей популяции, т.е. для очередного поколения. Такой выбор производится согласно принципу естественного отбора, по которому наибольшие шансы на участие в создании новых особей имеют хромосомы с наибольшими значениями функции приспособленности. Существуют различные методы селекции. Наиболее популярным считается так называемый *метод рулетки (roulette wheel selection)*, который свое название получил по аналогии с известной азартной игрой. Каждой хромосоме может быть сопоставлен сектор колеса рулетки, величина которого устанавливается пропорциональной значению функции приспособленности данной хромосомы. Поэтому чем больше значение функции приспособленности, тем больше сектор на колесе рулетки. Все колесо рулетки соответствует сумме значений функции приспособленности всех хромосом рассматриваемой популяции. Каждой хромосоме, обозначаемой  $ch_i$  для  $i = 1, 2, \dots, N$  (где  $N$  обозначает численность популяции) соответствует сектор колеса  $v(ch_i)$ , выраженный в процентах согласно формуле

$$v(ch_i) = p_s(ch_i)100\% \quad (2)$$

где

$$p_s(ch_i) = \frac{F(ch_i)}{\sum_{i=1}^N F(ch_i)}, \quad (3)$$

причем  $F(ch_i)$  - значение функции приспособленности хромосомы  $ch_i$ , а  $p_s(ch_i)$  - *вероятность селекции* хромосомы  $ch_i$ . Селекция хромосомы может быть представлена как результат поворота колеса рулетки, поскольку «выигравшая» (т.е. выбранная) хромосома относится к выпавшему сектору этого колеса. Очевидно, что чем больше сектор, тем больше вероятность «победы» соответствующей хромосомы. Поэтому вероятность выбора данной хромосомы оказывается пропорциональной значению ее функции приспособленности. Если всю окружность колеса рулетки представить в виде цифрового интервала  $[0, 100]$ , то выбор хромосомы можно отождествить с выбором числа из интервала  $[a, b]$ , где  $a$  и  $b$  обозначают соответственно начало и окончание фрагмента окружности, соответствующего этому сектору колеса; очевидно, что  $0 < a < b < 100$ . В этом случае выбор с помощью колеса рулетки сводится к выбору числа из интервала  $[0, 100]$ , которое соответствует конкретной точке на окружности колеса.

В результате процесса селекции создается *родительская популяция*, также называемая *родительским пулом (mating pool)* с численностью  $N$ , равной численности текущей популяции.

**Применение генетических операторов** к хромосомам, отобранным с помощью селекции, приводит к формированию новой популяции потомков от созданной на предыдущем шаге родительской популяции.

В классическом генетическом алгоритме применяются два основных генетических оператора: *оператор скрещивания (crossover)* и *оператор мутации (mutation)*. Однако следует отметить, что оператор мутации играет явно второстепенную роль по сравнению с оператором скрещивания. Это означает, что скрещивание в классическом генетическом алгоритме производится практически всегда, тогда как мутация - достаточно редко. Вероятность скрещивания, как правило, достаточно велика (обычно  $0,5 \leq p_c \leq 0,1$ ), тогда как вероятность мутации устанавливается весьма малой (чаще всего  $0 < p_m < 0,1$ ). Это следует из аналогии с миром живых организмов, где мутации происходят чрезвычайно редко.

В генетическом алгоритме мутация хромосом может выполняться на популяции родителей перед скрещиванием либо на популяции потомков, образованных в результате скрещивания.

**Оператор скрещивания.** На первом этапе скрещивания выбираются пары хромосом из родительской популяции (родительского пула). Это временная популяция, состоящая из хромосом, отобранных в результате селекции и предназначенных для дальнейших преобразований операторами скрещивания и мутации с целью формирования новой популяции потомков. На данном этапе хромосомы из родительской популяции объединяются в пары. Это производится случайным способом в соответствии с вероятностью скрещивания  $p_c$ . Далее для каждой пары отобранных таким образом родителей разыгрывается позиция гена (*локус*) в хромосоме, определяющая так называемую *точку скрещивания*. Если хромосома каждого из родителей состоит из  $L$  генов, то очевидно, что точка скрещивания  $l_k$  представляет собой натуральное число, меньшее  $L$ . Поэтому фиксация точки скрещивания сводится к случайному выбору числа из интервала  $[1, L-1]$ . В результате скрещивания пары родительских хромосом получается следующая пара потомков:

1) потомок, хромосома которого на позициях от 1 до  $l_k$  состоит из генов первого родителя, а на позициях от  $l_k + 1$  до  $L$  - из генов второго родителя;

2) потомок, хромосома которого на позициях от 1 до  $l_k$  состоит из генов второго родителя, а на позициях от  $l_k + 1$  до  $L$  - из генов первого родителя.

Действие оператора скрещивания проиллюстрировано примерами 3 и 4.

**Оператор мутации** с вероятностью  $p_m$  изменяет значение гена в хромосоме на противоположное (т.е. с 0 на 1 или обратно). Например, если в хромосоме [100110101010] мутации подвергается ген на позиции 7, то его значение, равное 1, изменяется на 0. что приводит к образованию хромосомы [100110001010]. Как уже упоминалось выше, вероятность мутации обычно очень мала, и именно от нее зависит, будет данный ген мутировать или нет. Вероятность  $p_m$  мутации может эмулироваться, например, случайным выбором числа из интервала [0, 1] для каждого гена и отбором для выполнения этой операции тех генов, для которых разыгранное число оказывается меньшим или равным значению  $p_m$ .

**Формирование новой популяции.** Хромосомы, полученные в результате применения генетических операторов к хромосомам временной родительской популяции, включаются в состав новой популяции. Она становится так называемой текущей популяцией для данной итерации генетического алгоритма. На каждой очередной итерации рассчитываются значения функции приспособленности для всех хромосом этой популяции, после чего проверяется условие остановки алгоритма и либо фиксируется результат в виде хромосомы с наибольшим значением функции приспособленности, либо осуществляется переход к следующему шагу генетического алгоритма, т.е. к селекции. В классическом генетическом алгоритме вся предшествующая популяция хромосом замещается новой популяцией потомков, имеющей ту же численность.

**Выбор «наилучшей» хромосомы.** Если условие остановки алгоритма выполнено, то следует вывести результат работы, т.е. представить искомое решение задачи. Лучшим решением считается хромосома с наибольшим значением функции приспособленности.

В завершение следует признать, что генетические алгоритмы унаследовали свойства естественного эволюционного процесса, состоящие в генетических изменениях популяций организмов с течением времени.

Главный фактор эволюции - это естественный отбор (т.е. природная селекция), который приводит к тому, что среди генетически различающихся особей одной и той же популяции выживают и оставляют потомство только наиболее приспособленные к окружающей среде. В генетических алгоритмах также выделяется этап селекции, на котором из текущей популяции выбираются и включаются в родительскую популяцию особи, имеющие наибольшие значения функции приспособленности. На следующем этапе, который иногда называется *эволюцией*, применяются генетические операторы скрещивания и мутации, выполняющие рекомбинацию генов в хромосомах.



Операция скрещивания заключается в обмене фрагментами цепочек между двумя родительскими хромосомами. Пары родителей для скрещивания выбираются из родительского пула случайным образом так, чтобы вероятность выбора конкретной хромосомы для скрещивания была равна вероятности  $p_c$ . Например, если в качестве родителей случайным образом выбираются две хромосомы из родительской популяции численностью  $N$ , то  $p_c = \frac{2}{N}$ . Аналогично, если из родительской популяции численностью  $N$  выбирается  $2z$  хромосом ( $z \leq N/2$ ), которые образуют  $z$  пар родителей, то  $p_c = 2z/N$ . Обратите внимание, что если все хромосомы текущей популяции объединены в пары до скрещивания, то  $p_c = 1$ . После операции скрещивания родители в родительской популяции замещаются их потомками.

Операция мутации изменяет значения генов в хромосомах с заданной вероятностью  $p_m$  способом, представленным при описании соответствующего оператора. Это приводит к инвертированию значений отобранных генов с 0 на 1 и обратно. Значение  $p_m$  как правило, очень мало, поэтому мутации подвергается лишь небольшое количество генов. Скрещивание - это ключевой оператор генетических алгоритмов, определяющий их возможности и эффективность. Мутация играет более ограниченную роль. Она вводит в популяцию некоторое разнообразие и предупреждает потери, которые могли бы произойти вследствие исключения какого-нибудь значимого гена в результате скрещивания.

Основной (классический) генетический алгоритм известен в литературе в качестве инструмента, в котором выделяются три вида операций: *репродукции*, *скрещивания* и *мутации*. Термины *селекция* и *репродукция* в данном контексте используются в качестве синонимов. При этом репродукция в данном случае связывается скорее с созданием копий хромосом родительского пула, тогда как более распространенное содержание этого понятия обозначает процесс формирования новых особей, происходящих от конкретных родителей. Если мы принимаем такое толкование, то операторы скрещивания и мутации могут считаться операторами репродукции, а селекция - отбором особей (хромосом) для репродукции.

Рассмотрим выполнение описанного в предыдущем разделе классического генетического алгоритма на простом примере. Проследим последовательность выполнения его этапов, соответствующих блок-схеме на рис. 2.

### Пример 3

Рассмотрим пример, состоящий в нахождении хромосомы с максимальным количеством единиц. Допустим, что хромосомы состоят из 12 генов, а популяция насчитывает 8

хромосом. Понятно, что наилучшей будет хромосома, состоящая из 12 единиц. Посмотрим, как протекает процесс решения этой весьма тривиальной задачи с помощью генетического алгоритма.

**Инициализация, или выбор исходной популяции хромосом.** Необходимо случайным образом сгенерировать 8 двоичных последовательностей длиной 12 битов. Это можно достигнуть, например, подбрасыванием монеты (96 раз, при выпадении «орла» приписывается значение 1, а в случае «решки» - 0). Таким образом можно сформировать исходную популяцию

$$\begin{array}{ll} ch_1 = [111001100101] & ch_5 = [010001100100] \\ ch_2 = [001100111010] & ch_6 = [010011000101] \\ ch_3 = [011101110011] & ch_7 = [101011011011] \\ ch_4 = [001000101000] & ch_8 = [000010111100] \end{array}$$

**Оценка приспособленности хромосом в популяции.** В рассматриваемом упрощенном примере решается задача нахождения такой хромосомы, которая содержит наибольшее количество единиц. Поэтому функция приспособленности определяет количество единиц в хромосоме. Обозначим функцию приспособленности символом  $F$ . Тогда ее значения для каждой хромосомы из исходной популяции будут такие:

$$\begin{array}{ll} F(ch_1) = 7 & F(ch_5) = 4 \\ F(ch_2) = 6 & F(ch_6) = 5 \\ F(ch_3) = 8 & F(ch_7) = 8 \\ F(ch_4) = 3 & F(ch_8) = 5 \end{array}$$

Хромосомы  $ch_3$  и  $ch_7$  характеризуются наибольшими значениями функции принадлежности. В этой популяции они считаются наилучшими кандидатами на решение задачи. Если в соответствии с блок—схемой генетического алгоритма (рис.2) условие остановки алгоритма не выполняется, то на следующем шаге производится селекция хромосом из текущей популяции.

**Селекция хромосом.** Селекция производится методом рулетки. На основании формул (2) и (3) для каждой из 8 хромосом текущей популяции (в нашем случае - исходной популяции, для которой  $N = 8$ ) получаем сектора колеса рулетки, выраженные в процентах (рис. 3).

$$\begin{array}{ll}
v(ch_1) = 15,22 & v(ch_5) = 8,70 \\
v(ch_2) = 13,04 & v(ch_6) = 10,87 \\
v(ch_3) = 17,39 & v(ch_7) = 17,39 \\
v(ch_4) = 6,52 & v(ch_8) = 10,87
\end{array}$$

Розыгрыш с помощью колеса рулетки сводится к случайному выбору числа из интервала  $[0, 100]$ , указывающего на соответствующий сектор на колесе, т.е. на конкретную хромосому. Допустим, что разыграны следующие 8 чисел:

79 44 9 74 44 86 48 23

Это означает выбор хромосом

$$ch_7 \ ch_3 \ ch_1 \ ch_7 \ ch_3 \ ch_7 \ ch_4 \ ch_2.$$

Как видно, хромосома  $ch_7$  была выбрана трижды, а хромосома  $ch_3$  – дважды. Заметим, что именно эти хромосомы имеют наибольшее значение функции приспособленности. Однако выбрана и хромосома  $ch_4$  с наименьшим значением функции приспособленности. Все выбранные таким образом хромосомы включаются в так называемый родительский пул.

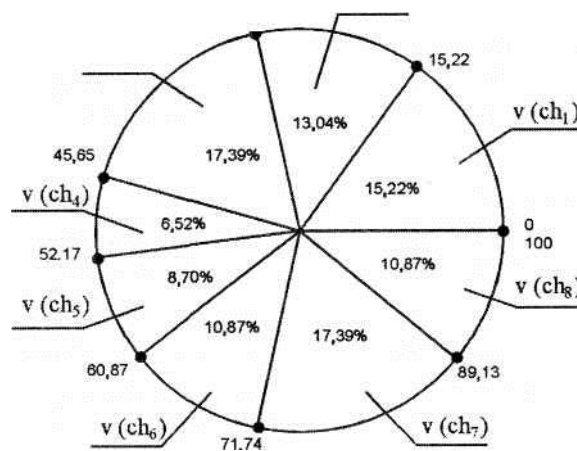


Рис.3. Колесо рулетки для селекции

**Применение генетических операторов.** Допустим, что ни одна из отобранных в процессе селекции хромосом не подвергается мутации, и все они составляют популяцию хромосом, предназначенных для скрещивания. Это означает, что вероятность скрещивания  $p_c = 1$ , а вероятность мутации  $p_m = 0$ . Допустим, что из этих хромосом случайным образом сформированы пары родителей

$$ch_2 \text{ и } ch_7, ch_1 \text{ и } ch_7, ch_3 \text{ и } ch_4, ch_3 \text{ и } ch_7.$$

Для первой пары случайным образом выбрана точка скрещивания  $l_k = 4$ , для второй  $l_k = 3$ , для третьей  $l_k = 11$ , для четвертой  $l_k = 5$ . При этом процесс скрещивания протекает так, как показано на рис. 4. В результате выполнения оператора скрещивания получаются 4 пары потомков.

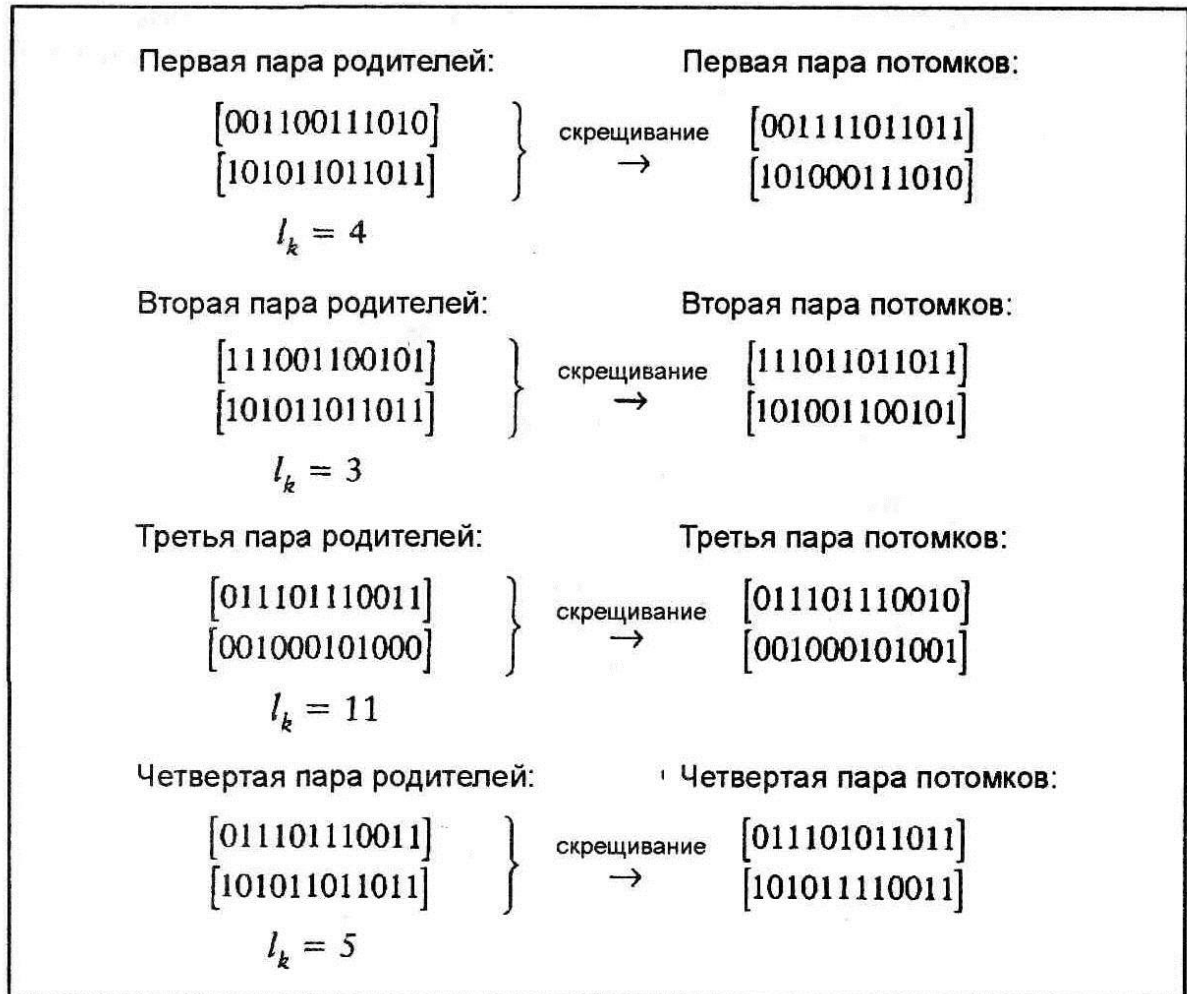


Рис.4. Процесс скрещивания хромосом

Если бы при случайном подборе пар хромосом для скрещивания были объединены, например,  $ch_3$  с  $ch_3$  и  $ch_4$  с  $ch_7$  вместо  $ch_3$  с  $ch_4$  и  $ch_3$  с  $ch_7$ , а другие пары остались без изменения, то скрещивание  $ch_3$  с  $ch_3$  дало бы две такие же хромосомы независимо от разыгранной точки скрещивания. Это означало бы получение двух потомков, идентичных своим родителям. Заметим, что такая ситуация наиболее вероятна для хромосом с

наибольшим значением функции приспособленности, т.е. именно такие хромосомы получают наибольшие шансы на переход в новую популяцию.

**Формирование новой популяции.** После выполнения операции скрещивания мы получаем (согласно рис. 4) следующую популяцию потомков:

$$\begin{array}{ll} Ch_1 = [001111011011] & Ch_5 = [011101110010] \\ Ch_2 = [101000111010] & Ch_6 = [001000101001] \\ Ch_3 = [111011011011] & Ch_7 = [011101011011] \\ Ch_4 = [101001100101] & Ch_8 = [101011110011] \end{array}$$

Для отличия от хромосом предыдущей популяции обозначения вновь сформированных хромосом начинаются с заглавной буквы  $C$ .

Согласно блок-схеме генетического алгоритма (рис. 2) производится возврат ко второму этапу, т.е. к оценке приспособленности хромосом из вновь сформированной популяции, которая становится текущей. Значения функций приспособленности хромосом этой популяции составляют

$$\begin{array}{ll} F(Ch_1) = 8 & F(Ch_5) = 7 \\ F(Ch_2) = 6 & F(Ch_6) = 4 \\ F(Ch_3) = 9 & F(Ch_7) = 8 \\ F(Ch_4) = 6 & F(Ch_8) = 8 \end{array}$$

Заметно, что популяция потомков характеризуется гораздо более высоким средним значением функции приспособленности, чем популяция родителей. Обратим внимание, что в результате скрещивания получена хромосома  $Ch_3$  с наибольшим значением функции приспособленности, которым не обладала ни одна хромосома из родительской популяции. Однако могло произойти и обратное, поскольку после скрещивания на первой итерации хромосома, которая в родительской популяции характеризовалась наибольшим значением функции приспособленности, могла просто «потеряться». Помимо этого «средняя» приспособленность новой популяции все равно оказалась бы выше предыдущей, а хромосомы с большими значениями функции приспособленности имели бы шансы появиться в следующих поколениях.

## Кодирование параметров задачи в генетическом алгоритме

Выбор исходной популяции связан с представлением параметров задачи в форме хромосом, т.е. с так называемым хромосомным представлением. Это представление определяется способом кодирования. В классическом генетическом алгоритме применяется двоичное кодирование, т.е. аллели всех генов в хромосоме равны 0 или 1. Длина хромосом зависит от условий задачи, точнее говоря - от количества точек в пространстве поиска.

### Пример 4.

Будем искать максимум функции, заданной формулой (1), но для переменной  $x$ , принимающей действительные значения из интервала  $[a, b]$ , где  $a = 0, b = 3,1$ . Допустим, что нас интересует решение с точностью до одного знака после запятой.

Поиск решения сводится к просмотру пространства, состоящего из 32 точек 0,0 0,1 ... 2,9 3,0 3,1. Эти точки (фенотипы) можно представить в виде хромосом (генотипов), если использовать бинарные пятизвенные цепочки, поскольку с помощью 5 битов можно получить  $2^5 = 32$  различных кодовых комбинации, причем хромосома [00000] будет соответствовать числу 0,0, хромосома [00001] - числу 0,1 и т.д., вплоть до хромосомы [11111], соответствующей числу 3,1.

Таким образом, мы можем воспроизвести последовательность этапов генетического алгоритма. Кодовые последовательности в рассматриваемой ситуации обозначают значения  $x$ , равные 0,0 0,1 ... 2,9 3,0 3,1. В результате выполнения этого алгоритма будет выбрано наилучшее решение, которое представляется хромосомой [11111] со значением фенотипа 3,1. Функция приспособленности этой хромосомы равна 20,22; это максимально возможное значение.

Заметим, что если бы нас интересовало решение с точностью, превышающей один знак после запятой, то интервал  $[0, 3,1]$  необходимо было бы разбить на большее количество подинтервалов, и для кодирования соответственно большего количества чисел потребовались бы более длинные хромосомы (с длиной, превышающей 5 битов). Аналогично, расширение области определения переменной  $x$  также потребует применения более длинных хромосом. Из этих наблюдений можно сделать вывод, что длина хромосом зависит от ширины области определения  $x$  и от требуемой точности решения.

Представим теперь задачу оптимизации в более общем виде. Допустим, что ищется максимум функции  $f(x_1, x_2, \dots, x_n) > 0$  для  $x_i \in [a_i, b_i] \subset R_i; i = 1, 2, \dots, n$ , и требуется найти решение с точностью до  $q$  знаков после запятой для каждой переменной  $x$ . В такой

ситуации необходимо разбить интервал  $[a_i, b_i]$  на  $(b_i - a_i) \cdot 10^q$  одинаковых подинтервалов. Это означает применение дискретизации с шагом  $r = 10^{-q}$ . Наименьшее натуральное число  $m_i$ , удовлетворяющее неравенству

$$(b_i - a_i) \cdot 10^q \leq 2^{m_i} - 1 \quad (4)$$

определяет необходимую и достаточную длину двоичной последовательности, требуемой для кодирования числа из интервала  $[a_i, b_i]$  с шагом  $r$ . Каждой такой двоичной последовательности соответствует десятичное значение числа, представляемого данным кодом (с учетом правил перевода десятичных чисел в двоичную форму). Пусть  $y_i$  обозначает десятичное значение двоичной последовательности, кодирующей число  $x_i$ . Значение  $x_i$ , можно представить выражением

$$x_i = a_i + y_i \frac{b_i - a_i}{2^{m_i} - 1}. \quad (5)$$

Таким способом задаются фенотипы, соответствующие кодовым последовательностям с длиной  $m_i$ . Рассмотренный пример - это частный случай задачи в данной постановке при условии, что  $i=1$  и  $q=1$ . Выражение (5) - это следствие из простого линейного отображения интервала  $[a_i, b_i]$  на интервал  $[0, 2^{m_i} - 1]$ , где  $2^{m_i}$  - десятичное число, закодированное двоичной последовательностью длиной  $m_i$  и составленной исключительно из единиц, а 0 - это, очевидно, десятичное значение двоичной последовательности длиной  $m_i$ , составленной только из нулей. Обратим внимание, что если  $a_i = -25$ ,  $b_i = 25$  и применяется шаг  $r = 0,05$ , то согласно формуле (4) получаем  $m_i = 10$ , а с помощью формулы (5) можно проверить значения фенотипов для генотипов, представленных в табл. 2.

**Таблица 2**

| Генотипы   |            | Фенотипы |        |
|------------|------------|----------|--------|
| 0000000000 | 0000000000 | -25,00   | -25,00 |
| 1010001001 | 0011001011 | 6,72     | -15,08 |
| 0110100010 | 1111010010 | -4,57    | 22,8.  |
| 1101101001 | 1110000111 | 17,67    | 19,13  |
| 0001101100 | 0000010001 | -19,72   | -24,17 |
| 0011000010 | 1011111010 | -15,52   | 12,24  |
| 1111111111 | 1111111111 | 25,00    | 25,00  |

## Решение комбинаторных задач с помощью генетических алгоритмов

При решении комбинаторных задач проблема заключается в поиске наилучшего решения среди возможных перестановок параметров задачи. В качестве примера можно назвать сортировку списка имен или задачу коммивояжера. Для решения комбинаторных задач применяются генетические операторы, определение которых несколько отличается от аналогичных операторов, ориентированных на оптимизационные задачи. В частности:

**Скрещивание** разбивается на следующие шаги:

- 1) случайным образом выбираются позиции у первого родителя; их количество зависит от показателя скрещивания;
- 2) находятся позиции с такими же значениями генов (аллелями) у второго родителя;
- 3) значения оставшихся позиций первого родителя копируются на оставшиеся позиции второго родителя в последовательности, в которой они записаны у первого родителя.

Описанный способ скрещивания иллюстрируется на рис.5.

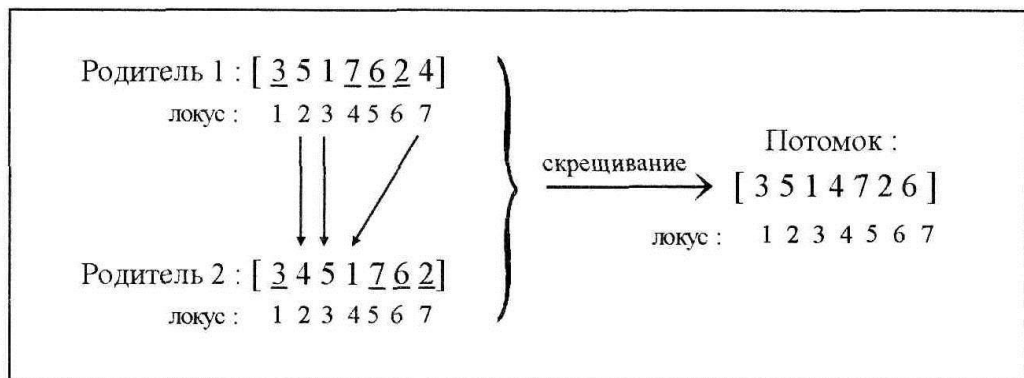


Рис.5. Скрещивание с сохранением порядка

На этом рисунке показаны две хромосомы родителей, состоящие из семи генов со значениями из интервала целых чисел от 1 до 7. Каждый ген в хромосоме характеризуется уникальным значением. Каждая хромосома представляет собой перестановку натуральных чисел от 1 до 7. Под каждым геном указан номер его позиции (locus). Допустим, что показатель скрещивания равен 0,5, и у первого родителя случайным образом выбраны позиции 1, 4, 5, 6, на которых находятся значения 3, 7, 6, 2 соответственно. У второго родителя эти значения находятся на позициях 1, 5, 6, 7. В результате копирования значений оставшихся позиций первого родителя (т.е. чисел 5, 1,



4 с позиций 2, 3, 7 соответственно) на оставшиеся позиции второго родителя т.е. на позиции 2, 3, 4) в последовательности, в которой они записаны у первого родителя, образуется потомок со значениями генов 3, 5, 1, 4, 7, 2, 6.

Описываемый способ подобен упорядоченному скрещиванию, показанному на рис.6. Номера позиций выбираются случайным образом. Далее значения генов с выбранных позиций одного из родителей переносятся на соответствующие позиции второго родителя. В результате скрещивания образуются два потомка.

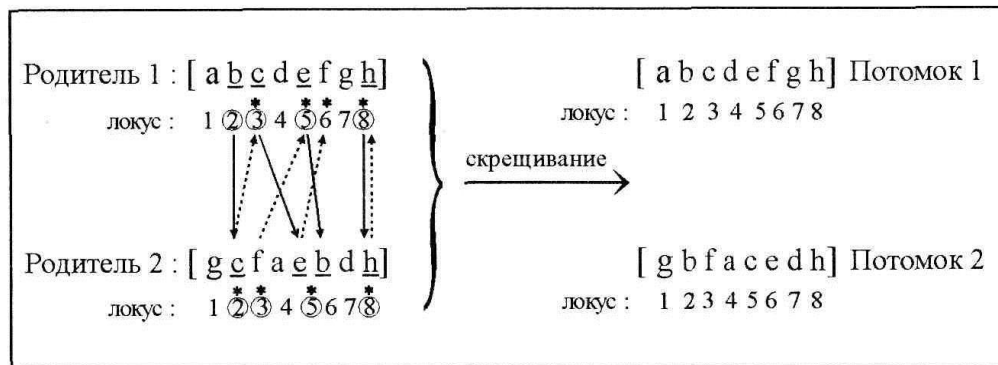


Рис.6. Упорядоченное скрещивание

**Мутация.** Оператор мутации реализует так называемую мутацию, основанную на упорядочении. Определенная таким образом мутация заключается в случайном выборе двух позиций в хромосоме и обмене значений генов на этих позициях. Например, после мутации хромосомы [3 5 1 4 7 2 6] на выбранных позициях 2 и 5 будет получена хромосома [3 7 1 4 5 2 6]. Количество обменов возрастает или снижается пропорционально увеличению или уменьшению показателя мутации.

Этот способ мутации отличается тем, что в результате ее выполнения формируется новая хромосома с измененной последовательностью генов. Такая мутация применяется для поиска наилучшей перестановки параметров задачи.

### Пример 5

Этот пример демонстрирует применение генетического алгоритма для сортировки списка имен в алфавитном порядке.

Для упрощения будем рассматривать очень короткий список из семи имен, начинающихся буквами J, M, B, R, S, H, F. Таким образом, наилучшее решение ищется в пространстве решений, состоящем из  $7! = 5040$  возможных перестановок семи элементов. Наилучшее решение очевидно - это B, F, H, J, M, R, S. На этом простейшем примере познакомимся с тем, как генетический алгоритм решает задачи того типа. Припишем каждому имени, включенному в исходный (несортированный) список, порядковый номер так, как это сделано в первом столбце на рис.7.

| Исходный список<br>(несортированный) |       | Одно из<br>возможных<br>решений |   |
|--------------------------------------|-------|---------------------------------|---|
| №                                    | Имена |                                 |   |
| 1                                    | J     | $P_1$                           | $\begin{bmatrix} 3 \\ 7 \\ 4 \\ 6 \\ 1 \\ 2 \\ 5 \end{bmatrix}$ B |
| 2                                    | M     | $P_2$                           | F   |
| 3                                    | B     | $P_3$                           | R   |
| 4                                    | R     | $P_4$                           | H   |
| 5                                    | S     | $P_5$                           | J   |
| 6                                    | H     | $P_6$                           | M   |
| 7                                    | F     | $P_7$                           | S   |

$$g_{21} = 0$$

$$g_{31} = 0, g_{32} = 0$$

$$g_{41} = 0, g_{42} = 0, g_{43} = 1$$

$$g_{51} = 0, g_{52} = 0, g_{53} = 1, g_{54} = 0$$

$$g_{61} = 0, g_{62} = 0, g_{63} = 1, g_{64} = 0, g_{65} = 0$$

$$g_{72} = 0, g_{73} = 0, g_{74} = 0, g_{75} = 0, g_{76} = 0$$

$$G = [0000010010001000000000]$$

Значение функции приспособленности = 3

Рис.7. Одно из допустимых решений

Допустимое решение, представленное на рисунке - это B, F, R, H, J, M, S. Приведенным первым буквам имен предшествуют соответствующие им порядковые номера из первого столбца. Последовательность этих номеров (на рисунке они вписаны в клетки) идентифицирует данное решение. На рис. 8 таким же образом представлено наилучшее решение, определяемое последовательностью 3, 7, 6, 1, 2, 4, 5.

| Исходный список<br>(несортированный) |       | Наилучшее<br>решение |   |
|--------------------------------------|-------|----------------------|---|
| №                                    | Имена |                      |   |
| 1                                    | J     | $P_1$                | $\begin{bmatrix} 3 \\ 7 \\ 6 \\ 1 \\ 2 \\ 4 \\ 5 \end{bmatrix}$ B |
| 2                                    | M     | $P_2$                | F   |
| 3                                    | B     | $P_3$                | H   |
| 4                                    | R     | $P_4$                | J   |
| 5                                    | S     | $P_5$                | M   |
| 6                                    | H     | $P_6$                | R   |
| 7                                    | F     | $P_7$                | S   |

$$g_{21} = 0$$

$$g_{31} = 0, g_{32} = 0$$

$$g_{41} = 0, g_{42} = 0, g_{43} = 0$$

$$g_{51} = 0, g_{52} = 0, g_{53} = 0, g_{54} = 0$$

$$g_{61} = 0, g_{62} = 0, g_{63} = 0, g_{64} = 0, g_{65} = 0$$

$$g_{71} = 0, g_{72} = 0, g_{73} = 0, g_{74} = 0, g_{75} = 0, g_{76} = 0$$

$$G = [0000000000000000000000]$$

Значение функции приспособленности = 0

Рис.8. Наилучшее решение задачи

Рассматриваемая задача имеет семь переменных (параметров задачи). Обозначим их  $P_1, P_2, \dots, P_7$ . Каждая из этих переменных может принимать целые значения от 1 до 7. На рис.7 показана одна из особей популяции, для которой значения конкретных переменных (параметров задачи) равны 3, 7, 4, 6, 1, 2, 5, а наилучшее решение на рис. 8 характеризуется значениями этих переменных 3, 7, 6, 1, 2, 4, 5. Приведенные последовательности чисел рассматриваются как аллели хромосом, представляющих соответствующие особи. Для каждой особи, входящей в популяцию, при выполнении генетического алгоритма рассчитывается значение функции приспособленности и на этой основе выбираются наилучшие и наихудшие особи.

Прежде чем определить функцию приспособленности для рассматриваемой задачи, введем ряд обозначений. Пусть  $n(P_i)$  соответствует имени, определенному порядковым номером  $P_i, i=1,2,...,7$ , и пусть  $n(P_i) < n(P_j)$  означает, что имя с порядковым номером  $P_i$ , должно предшествовать имени с порядковым номером  $P_j$ , т.е. они упорядочиваются по алфавиту. Пусть  $G$  обозначает последовательность, составленную из элементов  $g_{km}, k=2,...,7, m=1,2,...,k-1$ , где

$$g_{km} = \begin{cases} 1, & \text{если } n(P_k) < n(P_m) \\ 0, & \text{в противном случае} \end{cases}$$

Очевидно, что если последовательность  $G$  состоит из одних нулей, то последовательность имен будет корректной. Поэтому наилучшая особь должна характеризоваться последовательностью  $G$ , все элементы которой равны нулю. Следовательно, функцию приспособленности можно определить как сумму элементов последовательности  $G$ , и нас, конечно, будет интересовать минимизация этой функции (которая может принимать целые значения в интервале от 0 до 21). Заметим, что если бы элемент  $g_{km}$  принимал нулевое значение при  $n(P_k) < n(P_m)$  и значение 1 в противном случае, то следовало бы максимизировать количество единиц в последовательности  $G$ . Теперь перейдем к интерпретации функции приспособленности.

Для задачи, сформулированной в примере 5, размерность популяции принята равной 10, показатель скрещивания равен 0,5, показатель мутации равен 0. При  $t = 10$ , что соответствует первой популяции классического генетического алгоритма, получена популяция, представленная на рис.9.

|    |    |   |   |   |   |   |   |   |
|----|----|---|---|---|---|---|---|---|
| 1  | 15 | 3 | 5 | 1 | 4 | 7 | 2 | 6 |
| 2  | 13 | 5 | 3 | 1 | 4 | 2 | 7 | 6 |
| 3  | 13 | 2 | 6 | 1 | 5 | 4 | 3 | 7 |
| 4  | 11 | 3 | 4 | 5 | 1 | 7 | 2 | 6 |
| 5  | 11 | 7 | 2 | 6 | 5 | 4 | 1 | 3 |
| 6  | 11 | 1 | 7 | 7 | 4 | 5 | 6 | 3 |
| 7  | 11 | 7 | 4 | 1 | 5 | 6 | 3 | 2 |
| 8  | 9  | 6 | 4 | 1 | 3 | 2 | 7 | 5 |
| 9  | 7  | 3 | 5 | 1 | 7 | 6 | 2 | 4 |
| 10 | 5  | 3 | 6 | 1 | 4 | 2 | 7 | 5 |

Рис.9. Популяция особей

Первый столбец определяет последовательность хромосом в популяции (от «наихудшей» к «наилучшей»). Во втором приведены значения функции

приспособленности каждой хромосомы. В третьем столбце записаны сами хромосомы, состоящие из семи генов. Каждая из этих хромосом представляет собой перестановку натуральных чисел от 1 до 7. Первое значение функции приспособленности, очерченное прямоугольником и равное 15, относится к хромосоме, исключенной из предыдущей популяции. Вместо нее вводится хромосома [3 5 1 4 7 2 6], полученная в результате скрещивания хромосом [3 5 1 7 6 2 4] и [3 4 5 1 7 2 6], присутствующих на рис.9. Последняя хромосома, имеющая функцию приспособленности, равную пяти, после 10 «тактов» является «наилучшей на данный момент». При продолжении выполнения алгоритма можно получить наилучшую хромосому [3 7 6 1 2 4 5] с функцией приспособленности, равной 0. Эта оптимальная хромосома представлена на рис.8. Заметим, что в рассматриваемом примере функция приспособленности интерпретируется как погрешность, которую следует минимизировать. Для наилучшего решения эта погрешность обязана быть равной 0!

Пример 5 легко обобщить на любые перечни имен или названий, подлежащих сортировке в алфавитном порядке (либо в соответствии с другим ключом). Таким же образом можно решить задачу о коммивояжере, имеющую гораздо более сложную структуру, чем рассмотренная задача из примера 5.

### Задания

1. Задача коммивояжера при числе городов  $N = 10$ .
2. Решить задачу раскраски карты четырьмя цветами с таким расчетом, чтобы ни одна пара соседних стран не была окрашена в одинаковый цвет. Число стран  $N = 10$ .
3. Задача разбиения графа. Требуется разбить множество  $N = 16$  вершин графа на две группы одинаковой мощности так, чтобы сумма весов ребер, имеющих концы в разных группах, была минимальной.
4. Задача о рюкзаке. Имеется рюкзак емкости  $V$  и  $m = 10$  предметов, каждый из которых имеет размер  $a_i$  и ценность  $c_i$ ,  $i = 1, 2, \dots, m$ . Требуется поместить в рюкзак предметы с наибольшей суммарной ценностью.
5. Задана матрица расстояний между  $N = 20$  вершинами графа. Найти 3 вершины-медианы графа такие, что сумма расстояний от остальных вершин графа до каждой из медиан минимальна.
6. Задана матрица расстояний между  $N = 20$  вершинами графа. Найти 3 вершины графа такие, что сумма расстояний от каждой такой вершины до всех остальных максимальна.
7. Квадратичная задача о назначениях. Заданы координаты  $N = 10$  городов и матрицы:

- $D = \|D_{ij}\|$  стоимостей перевозки тонны груза между пунктами  $i$  и  $j$ ;
- $B = \|B_{kl}\|$  объемов перевозки между заводами  $k$  и  $l$ .

Требуется разместить  $m = 10$  заводов по одному заводу в каждом городе, чтобы суммарная стоимость межзаводских перевозок была минимальной.

8. Задача кластеризации. Заданы координаты  $m = 15$  точек плоскости. Разбить точки на 4 группы (кластера) так, чтобы сумма по всем кластерам сумм расстояний от точек каждого кластера до его центра была минимальной среди всех возможных разбиений.