

УДК 004.4'2

ОБЩИЙ МЕТОД РЕШЕНИЯ ТРАНСПОРТНОЙ ЗАДАЧИ В MAPLE

Доц. Шарстнев В.Л., ст. преп. Вардомацкая Е.Ю.

УО «Витебский государственный технологический университет»

Минимизация транспортных затрат – одна из важнейших задач современной экономики. Поэтому при принятии любого конкретного управленческого решения необходим предварительный системный анализ проблем логистики. Если еще два-три десятилетия назад проблемы оптимизации перевозок можно было решить методом перебора вариантов, то сегодня такой анализ невозможен без применения количественных математических методов.

Целью исследования является разработка обобщенного метода решения комплекса типовых оптимизационных задач о наиболее выгодном планировании транспортировки продукции от поставщиков к потребителям в среде системы компьютерной математики (СКМ) MAPLE. Задачи такого рода приходится решать практически каждому руководителю предприятия или его подразделения.

В качестве инструментария решения использована система компьютерной математики MAPLE.

Исходные данные (при условных значениях переменных, предполагающих задачу открытого типа) определены следующим образом:

- матрица стоимости перевозок единицы груза от производителя к потребителю:

```
>Стоимость:=Matrix([[1.5,2,1.75,2.25,2.25],[2.5,2,1.75,1,1.5],  
[2,1.5,1.5,1.75,1.75],[2,0.5,1.75,1.75,1.75]]);
```

- вектор производства продукции каждым производителем:

```
> Производство:=vector([200,250,225,175]);
```

- вектор потребности в продукции каждым потребителем:

```
> Потребление:=vector([100,200,50,250,150]);
```

- матрица неизвестных – объемы перевозок груза от производителя к потребителю:

```
>Перевозки:=Matrix(LinearAlgebra[RowDimension](Стоимость),  
LinearAlgebra[ColumnDimension](Стоимость),symbol=x);
```

Ключевыми аспектами программы, разработанной на встроенном языке ядра СКМ MAPLE, являются:

- Определение процедуры пользователя с формальными параметрами матрицами a и b для вычисления значения целевой функции – суммарных затрат на перевозку груза от потребителей к поставщикам:

```
> SumProduct:=proc(a,b)
```

```
> local i,j;
```

```
> if LinearAlgebra[RowDimension](a)<>LinearAlgebra[RowDimension](b)or  
LinearAlgebra[ColumnDimension](a)<>LinearAlgebra[ColumnDimension](b)
```

```
then error "Матрицы неодинаковой размерности"
```

```
> else add(add(a[i,j]*b[i,j],j=1..LinearAlgebra[ColumnDimension](b)),  
i=1..LinearAlgebra[RowDimension](b))
```

```
> end if
```

> end proc;

- Определение типа задачи, и в зависимости от типа:

а) вычисление потребности фиктивного потребителя или объема производства фиктивного поставщика, с добавлением этих значений в вектор НовоеПотребление или в вектор НовоеПроизводство соответственно;

б) формирование матриц НоваяСтоимость и НовыеПеревозки (объемы перевозок) путем добавления столбца с нулевой стоимостью и нулевыми объемами перевозок для фиктивного потребителя или строки с нулевой стоимостью и нулевыми объемами перевозок для фиктивного поставщика

> ТипЗадачи:="Закрытая транспортная задача";

> if СуммарноеПотребление<СуммарноеПроизводство then ТипЗадачи:="Открытая транспортная задача.Фиктивный потребитель";

>НовоеПотребление:=ArrayTools[Concatenate](2, Потребление, abs(СуммарноеПроизводство-СуммарноеПотребление));

>НоваяСтоимость:=convert(linalg[extend](Стоимость,0,1,0), Matrix);

>НовПер:=convert(linalg[extend](Перевозки,0,1,0), Matrix);

>НовыеПеревозки:=Matrix(LinearAlgebra[RowDimension]\(НовПер), LinearAlgebra[ColumnDimension](НовПер), symbol=x)

> end if;

> if (СуммарноеПроизводство<СуммарноеПотребление)
and(СуммарноеПроизводство<>СуммарноеПотребление)

then ТипЗадачи:="Открытая транспортная задача.Фиктивный производитель";

>НовоеПроизводство:=ArrayTools[Concatenate](2, Производство, abs(СуммарноеПроизводство-СуммарноеПотребление));

>НоваяСтоимость:=convert(linalg[extend](Стоимость,1,0,0), Matrix);

>НовПер:=convert(linalg[extend](Перевозки,1,0,0), Matrix);

>НовыеПеревозки:=Matrix(LinearAlgebra[RowDimension](НовПер), LinearAlgebra[ColumnDimension](НовПер), symbol=x)

> end if;

- Для поиска оптимального решения по критерию минимизации затрат (переменная Solution) при заданных ограничениях на производство и потребление использована подбиблиотека LPsolve (решение задач линейного программирования) библиотеки Optimization:

>ОграничениеПроизводство:=seq(add(НовыеПеревозки[i,j],
j=1..LinearAlgebra[ColumnDimension](НовыеПеревозки))=НовоеПроизводство[i], i=1..LinearAlgebra[RowDimension](НовыеПеревозки));

> ОграничениеПотребление:=seq(add(НовыеПеревозки[i,j],
i=1..LinearAlgebra[RowDimension](НовыеПеревозки))=НовоеПотребление[j], j=1..LinearAlgebra[ColumnDimension](НовыеПеревозки));

>Solution:=Optimization[LPsolve](SumProduct(НоваяСтоимость, НовыеПеревозки), Ограничения, assume=nonnegint);

Эта библиотека выдает единое решение: значение целевой функции - общих затрат на перевозку (в отладочном примере 862.50) и объемы перевозок ($X_{1,1}, X_{1,2} \dots X_{4,6}$) от поставщиков к потребителям. На экране результаты решения выглядят следующим образом:

$Solution := [862.50, [x_{1,1} = 100, x_{1,2} = 0, x_{1,3} = 0, x_{1,4} = 0, x_{1,5} = 0, x_{1,6} = 100, x_{2,1} = 0, x_{2,2} = 0, x_{2,3}$
 $= 0, x_{2,4} = 250, x_{2,5} = 0, x_{2,6} = 0, x_{3,1} = 0, x_{3,2} = 25, x_{3,3} = 50, x_{3,4} = 0, x_{3,5} = 150, x_{3,6} = 0, x_{4,1}$
 $= 0, x_{4,2} = 175, x_{4,3} = 0, x_{4,4} = 0, x_{4,5} = 0, x_{4,6} = 0]]$

В данном случае имеет место открытая транспортная задача с фиктивным потребителем (в решении присутствует шестой столбец при исходном векторе Потребление длиной 5).

Для представления на экране объемов перевозок ($X_{1,1}, X_{1,2} \dots X_{4,6}$) в виде матрицы первое решение (значение целевой функции) выделено из общего решения и с помощью оператора конкатенации `cat` соединено с полученными значениями матрицы Новыеперевозки:

```
> print(cat(ТипЗадачи, ".Решение--->"), Solution[1], Новыеперевозки);
```

"Открытая транспортная задача Фиктивный потребитель Решение ---> ", 862.50,

100	0	0	0	0	100
0	0	0	250	0	0
0	25	50	0	150	0
0	175	0	0	0	0

При изменении исходных данных анализ типа задачи и пересчет осуществляется автоматически.

Следует отметить, что к модели транспортной задачи сводится широкий круг задач оптимизации, таких как, например, задача о назначениях, задача о ранце, задача коммивояжера и некоторые другие. Предлагаемый алгоритм при незначительных изменениях может быть использован для поиска оптимальных решений и этих задач линейного программирования.

УДК 338.5

ЦЕНОВАЯ ПОЛИТИКА ОРГАНИЗАЦИИ НА ПРИМЕРЕ ФИТНЕС – КЛУБА «НОН – СТОП»

Студ. Якубкина Я.Р., доц. Прокофьева Н.Л.

УО «Витебский государственный технологический университет»

В развитых странах стремление к здоровому образу жизни подкреплено достаточно высокой покупательной способностью самых широких слоев населения. В Беларуси спрос на здоровый образ жизни и фитнес-услуги ограничен малым бюджетом и низкой готовностью заниматься спортом.

Сеть фитнес-залов «Нон-стоп» появилась в 2003 году. Фитнес-клуб «Нон-Стоп», расположенный по улице Кирова 11а – один из основных клубов сети «Нон-Стоп» в Витебске.

В своей практике фитнес-клуб «Нон-Стоп» использует дискриминационную политику второго типа, которая проявляется, прежде всего, в виде различных скидок с цены для определенных категорий потребителей, а также скидки при покупке