

Транспортная задача с открытой моделью

Демилович Евгений Михайлович, Змеева Юлия Викторовна,
Шульдова Светлана Георгиевна,

Минский инновационный университет, г. Минск, Беларусь
Минский инновационный университет, г. Минск, Беларусь
Белорусский государственный университет информатики и радиоэлектроники, г. Минск,
Беларусь

Аннотация

Статья посвящена вопросу решения транспортной задачи с открытой транспортной моделью. Метод, реализованный в программе, позволяет за конечное число шагов решать и вырожденные транспортные задачи. Программа разработана на языке С#.

Ключевые слова: транспортная задача, вырожденность, открытая модель.

Транспортная задача наиболее часто встречается в практических приложениях линейного программирования. Данная задача формулируется следующим образом. Имеется m пунктов производства некоторого товара и n пунктов его потребления. Известны транспортные расходы перевозки единицы товара из каждого пункта производства в каждый пункт потребления. Необходимо определить наиболее экономный план перевозок.

В случае замкнутой транспортной модели суммарный объем производства равен суммарному объему потребления. При нарушении этого условия задача является открытой транспортной моделью.

При разработке программы в данной работе использовался венгерский метод. В отличие от метода потенциалов венгерский метод нечувствителен к явлению вырожденности, кроме того он позволяет более экономно строить цепочки переходов. Также важным преимуществом венгерского метода является то, что он позволяет не только найти оптимальный план, но и оценить близость плана текущей итерации к оптимальному плану, что очень важно при решении задач с большим количеством пунктов производства и потребления.

В программе реализован контроль за последовательностью действий пользователя при решении задачи. В случае ошибочных операций выводится соответствующее сообщение и выводится подсказка, что пользователь должен выполнить. Предусмотрена возможность проверки правильности введенных данных и их корректировки. Программа находит оптимальный план перевозок как в случае замкнутой транспортной модели, так и в случае открытой транспортной модели. Если постановка задачи соответствует открытой транспортной модели, то программа запрашивает согласие пользователя на поиск решения в такой постановке. Программа реализована на языке С#, который способствует более компактному объявлению классов программы и реализации алгоритма решения задачи. Ниже приведены класс производителя товара и класс потребителя с соответствующими комментариями.

```
class Proiz // Класс производителя товара.
{
    public string Firm_Pro { get; set; } // Наименование фирмы производителя.
    public double Kol_Tov_Pro { get; set; } // Количество производимого товара.
    // Исходное количество товара у данного производителя.
    public double isx_Kol_Tov_Pro { get; set; }
    public double Kol_Tov_Ostavsh { get; set; } // Количество лишнего товара.
    double[] x; // Кому, сколько перевезти товара.
    public void memory_Potr(int n_potr)
    { // Выделение памяти под строку отправляемого товара.
        x=new double[n_potr];
        for (int i = 0; i < n_potr; i++) x[i] = new double();
    }
    public double this[int i] // Индексатор.
    {
        get { return x[i]; }
        set { x[i] = value; }
    }
}
```

```

class Potr // Класс потребителя товара.
{
    public string Firm_Pot { get; set; } // Наименование фирмы потребителя.
    public double Kol_Tov_Pot { get; set; } // Количество потребляемого товара.
    // Исходное количество товара, необходимого данному потребителю.
    public double isx_Kol_Tov_Pot { get; set; }
    // Количество недостающего товара.
    public double Kol_Tov_Nedostat { get; set; }
    double[] x; // Кто, сколько поставляет товара.
    public void memory_Proiz(int n_proiz)
    { // Выделение памяти под столбец доставляемого товара.
        x = new double[n_proiz];
        for (int i = 0; i < n_proiz; i++) x[i] = new double();
    }
    public double this[int i] // Индексатор.
    {
        get { return x[i]; }
        set { x[i] = value; }
    }
}

```

Весь код программы занимает более шестисот строк текста.

Заключение

Разработанная на языке C# программа находит решение за конечное число шагов, решает вырожденные транспортные задачи. В программе не накладываются ограничения на количество поставщиков и количество потребителей. Необходимая память для хранения исходных данных и матрицы перевозок выделяется динамически. Задача может быть поставлена как с замкнутой транспортной моделью, так и с открытой транспортной моделью. В обоих случаях программа находит оптимальный план перевозок. Данную программу после некоторой модификации можно использовать для решения транспортных задач других типов.

Литература

1. Гольштейн, Е.Г. Задачи линейного программирования транспортного типа / Е.Г. Гольштейн, Д.Б. Юдин. – М.: Издательство «НАУКА», Главная редакция физико-математической литературы, 1969. – 382 с.
2. Том Уильям. Структуры данных в C++: Пер. с англ. / Уильям Том, Уильям Форд. – М.: ЗАО «Издательство БИНОМ», 1999. – 816 с.: ил.
3. C# для профессионалов: в 2-х т. / С. Робинсон [и др.] // Программист – программисту. – М.: Лори, 2003. – 478 с.
4. Демидович, В.Е. C#. Консольные приложения: справочник / В.Е. Демидович, Е.М. Демидович. – Минск: Бестпринт, 2017. – 658 с.