

ПОСТАНОВКА ЛАБОРАТОРНОЙ РАБОТЫ ПО ВЫЧИСЛЕНИЮ МАКСИМАЛЬНОГО ВЗВЕШЕННОГО ПАРОСОЧЕТАНИЯ

¹Напрасников В. В., ²Кункевич Д. П., ³Ковалева И. Л.
¹Белорусский национальный технический университет,
 Минск, Беларусь, n_v_v@tut.by,
²Белорусский национальный технический университет,
 Минск, Беларусь, kunkevich@bntu.by,
³Белорусский национальный технический университет,
 Минск, Беларусь, ilkovaleva@bntu.by

Аннотация. В докладе рассматриваются особенности использования MATLAB при выполнении лабораторной работы по вычислению максимального парасочетания для слушателей-программистов на основе решения задачи целочисленного линейного программирования.

Задача о максимальном (взвешенном) парасочетании это одна из классических задач теории графов. Для ее решения создано множество алгоритмов, однако при проведении лабораторных работ особое значение имеет наглядность представления при постановке задачи и интерпретации полученных результатов.

Поэтому попытаемся создать наиболее простой программный код и воспользуемся тем, что эта задача может быть записана в терминах целочисленного линейного программирования (ЦЛП), и используем созданные эффективные процедуры `intlinprog` современной версии среды MATLAB.

Будем описывать граф двумя матрицами V и E . В матрице V задаем координаты вершин графа и их веса (если они заданы), а в матрице E задаем список ребер графа, соединяющих вершины и веса ребер (если они заданы). Например, если использовать для отображения графа процедуру `PlotGraph2023`, (является модификацией процедуры `PlotGraph`, предложенной в [1] с учетом внесенных нами изменений в версиях среды MATLAB, которые сделали процедуру `PlotGraph` из [1] неработоспособной), то после выполнения представленного ниже фрагмента:

```
clearall % очистили память
V=[0 0;1 2;2 1;1 -2]; % координаты вершин и их веса
E= [1 3 5;1 4 2;2 1 1;2 3 7;2 4 3;3 2 9;3 4 21;4 3 17];
% список ребер графа и их веса
PlotGraph2023(V,E,'g'); % рисуем граф
set(get(gcf,'CurrentAxes'),...
    'FontName','Times New Roman Cyr','FontSize',10) % установили шрифт
title('\bfИсходный граф со взвешенными ребрами')
```

Получим вид графа, представленный на рис. 1.

Исходный граф со взвешенными рѣбрами

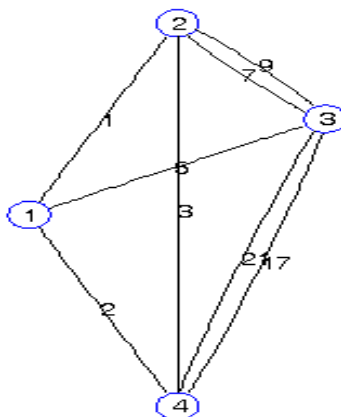


Рисунок 1 – Граф со взвешенными ребрами

В задаче о максимальном взвешенном паросочетании нужно максимизировать суммарный вес ребер. Введем в рассмотрение вектор-столбец e . Назовем этот вектор ассоциированным с ребрами графа E , т. к. каждая координата e_k вектора e будет характеризовать соответствующее ребро. Обозначим столбец весов ребер c . Если ребро e_k входит в паросочетание, то ассоциированная с ней переменная e_k будет принимать значение 1, а если нет, то 0.

Тогда будем иметь задачу с целевой функцией $z = (c, e)$:

$$\begin{cases} z = (c, e) \rightarrow \max; \\ Ae \leq 1; \\ e_k = 0 \vee 1; k = \overline{1, m}. \end{cases}$$

Здесь A матрица инцидентности, а условие $Ae \leq 1$ означает, что для каждой вершины v , сумма переменных e_k , ассоциированных с ребрами, инцидентными с v_i , не превышает единицы. Для решения задачи воспользуемся кодом, представленным ниже с использованием процедуры `intlinprog`.

```
% ===== Input data validation =====
if nargin<1,
error ('There are no input data!')
end
[m, n, E] = grValidation(E); % E data validation

% ===== Parameters of integer LP problem =====
A=zeros (n, m); % for incidence matrix
A(E(:,1:2) + repmat([1:m]'-1)*n,1,2)=1; % we fill the incidence matrix
b=ones(n,1); % Правые части ограничений (по количеству ВЕРШИН) Это
ЕДИНИЦЫ
f=-E(:,3); % При максимизации коэффициенты целевой функции с МИ-
НУСОМ

for i=1:m % Цикл по РЕБРАМ
```

```

intcon(1,i)=i; % Номера целочисленных переменных (по количеству РЕБЕР)
end;
Vlb=zeros(1,m);% Ограничения снизу на все переменные 0
Vub=ones(1,m);% Ограничения сверху на все переменные 1

```

```

% ===== We solve the integer LP problem =====
xMAX = intlinprog(f,intcon,A,b,[],[],Vlb,Vub);
nMM=find(round(xMAX)); % the answer – numbers of edges НОМЕРА РЕБЕР

```

Для отображения результатов воспользуемся следующим кодом.

```

fprintf('Количество ребер в максимальном взвешенном паросочетании =
%d\n',...
length(nMM));
disp('В максимальное взвешенное паросочетание включены ребра с номе-
рами:');
fprintf ('%d',nMM);
fprintf('\nОбщий вес = %d\n',sum(E(nMM,3)));

```

Результаты расчета представлены ниже:

```

Количество ребер в максимальном взвешенном паросочетании = 2
В максимальное взвешенное паросочетание включены ребра с номерами:
3 7
Общий вес = 22

```

Для визуализации результатов выполним следующий код.

```

PlotGraph2023(V,E(nMM, :),'g'); % рисуем максимальное взвешенное паросочетание
set(get(gcf,'CurrentAxes'),...
'FontName','Times New Roman Cyr','FontSize',10) % установили шрифт
title('\bfМаксимальное взвешенное паросочетание')

```

Результаты представлены на рис. 2.

Максимальное взвешенное паросочетание

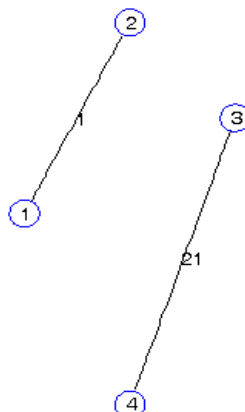


Рисунок 2 – Результаты расчета

Рассмотрим решение более сложного примера, предложенного в [1].

```
clearall % очистили память
V=[0 0;1 1;1 0;1 -1;2 1;2 0;2 -1;3 1;...
 3 0;3 -1;4 0]; % координаты вершин
E=[1 2 5;1 3 5;1 4 5;2 3 2;3 4 2;2 5 3;...
 2 6 2;3 6 5;3 7 2;4 7 3;5 6 1;6 7 1;...
 5 8 5;6 8 2;6 9 3;7 9 2;7 10 3;8 9 2;...
 9 10 2;8 11 5;9 11 4;10 11 4]; % список ребер графа и их веса
```

Результаты расчета представлены на рисунке 3 и совпадают с результатами из [1]. Описанное программное обеспечение используется в курсах, где предусмотрены разделы моделирования на графах.

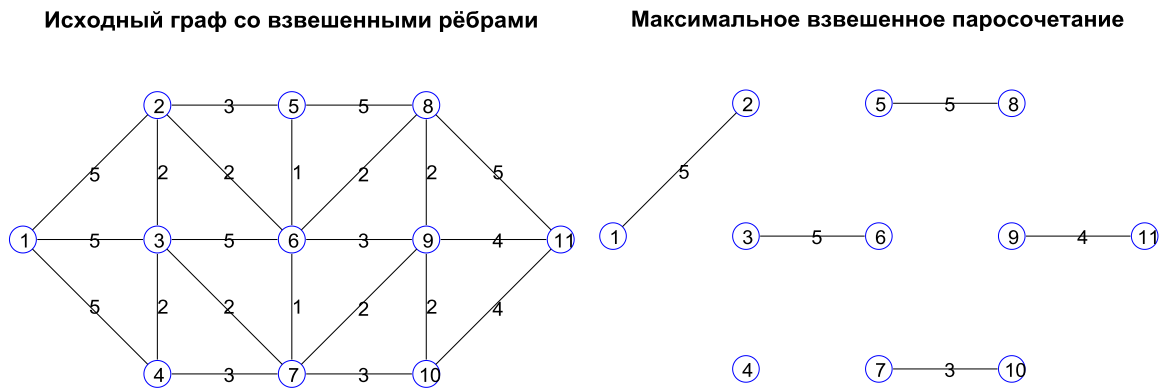


Рисунок 3 – Результаты расчета примера из [1].

Литература

1. Иглин, С. П. Математические расчеты на базе MATLAB / С. П. Иглин. – Серия: Учебное пособие. – СПб.: БХВ-Петербург, 2005. – 640 с.