

Санкт-Петербургский политехнический университет Петра Великого
Институт компьютерных наук и кибербезопасности
Высшая школа компьютерных технологий и информационных систем

Отчёт расчетной работе № 1

Дисциплина: Системный анализ и принятие решений.

Выполнил студент гр. 5130901/10101 _____ Д.Л. Симоновский
(подпись)

Руководитель _____ А.Г. Сиднев
(подпись)

“18” февраля 2023 г.

Санкт-Петербург

2024

Оглавление

1. Условие:	2
1.1. Вариант:	2
1.2. Условие задания:	2
2. Ход решения	3
2.1. Определить наиболее ранние моменты начала работ с использованием метода математического программирования:	3
2.2. Определить наиболее ранние моменты начала работ и их интенсивности, если длительность равна интенсивности выполнения работ, а суммарная интенсивность не превышает 75% от общего числа выполняемых работ.	4
3. Вывод	7
4. Ссылки	7
5. Приложение	7

1. Условие:

1.1. Вариант:

Вариант №10.

Граф №19.

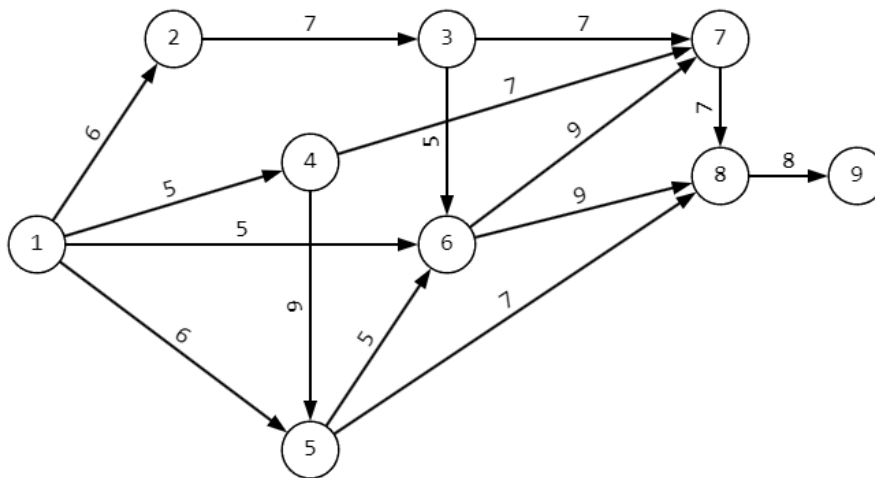


Рис. 1.1. Граф №19.

Число исполнителей 2.

Решающее правило: Короткие работы вперед.

1.2. Условие задания:

Выполнить следующие разделы:

1. Определить наиболее ранние моменты начала работ с использованием метода математического программирования.
2. Определить наиболее ранние моменты начала работ и их интенсивности, если длительность равна интенсивности выполнения работ, а суммарная интенсивность не превышает 75% от общего числа выполняемых работ.
3. Самостоятельно распределить работы между заданным числом исполнителей и сформулировать задачу математического программирования с бинарными индикаторными переменными. Определить число бинарных переменных и дополнительных ограничений в этой задаче и дать содержательную формулировку части ограничений с бинарными переменными.
 - 3.1. Изменить формулировку задачи так, чтобы число бинарных переменных не превышало 10. Решить полученную задачу с использованием команды **intlinprog**. Определить мощность множества бинарных переменных задачи и дать содержательную интерпретацию полученному решению.
4. Найти характеристики t_i^* , t_i^{**} и r_{ij} расписания выполнения комплекса работ с использованием метода динамического программирования. Привести соответствующие уравнения Беллмана. Определить критические пути на графе.
5. Найти те же характеристики t_i^* , t_i^{**} и r_{ij} расписания выполнения комплекса работ с использованием математического программирования.
6. Определить помимо полных резервов времени $F_n = r_{ij}$ работ ij резервы времени, относящиеся к событиям j сетевого графа, а именно $F_{нз1}$, F_c , $F_{нз2}$.
7. Рассмотреть вероятностную постановку задачи анализа расписания.

Считать СКО времен выполнения работ равными 5% от их длительностей. Предполагая неизменным критический путь (оценить справедливость этого предположения) найти вероятность того, что время выполнения комплекса работ не превысит найденного для детерминированной задачи в п.1 на 10%.

8. Представить пошаговую процедуру имитационного моделирования расписания по схеме событий с учетом числа исполнителей и решающего правила ранжирования работ из числа возможных. По результатам моделирования построить диаграмму Ганта.

2. Ход решения

2.1. Определить наиболее ранние моменты начала работ с использованием метода математического программирования:

Для графа, представленного на Рис. 1.1, составим систему неравенств для последующего решения с помощью методов линейного программирования. Обозначим за t_{ij} наиболее ранний момент начала работы ij , а за t_{end} – наиболее ранний момент окончания всех работ.

$$\begin{array}{lll}
 t_{23} \geq t_{12} + 6 & t_{58} \geq t_{15} + 6 & t_{68} \geq t_{56} + 5 \\
 t_{36} \geq t_{23} + 7 & t_{58} \geq t_{45} + 9 & t_{78} \geq t_{37} + 7 \\
 t_{37} \geq t_{23} + 7 & t_{67} \geq t_{16} + 5 & t_{78} \geq t_{47} + 7 \\
 t_{45} \geq t_{14} + 5 & t_{67} \geq t_{36} + 5 & t_{78} \geq t_{67} + 9 \\
 t_{47} \geq t_{14} + 5 & t_{67} \geq t_{56} + 5 & t_{89} \geq t_{78} + 7 \\
 t_{56} \geq t_{15} + 6 & t_{68} \geq t_{16} + 5 & t_{89} \geq t_{68} + 9 \\
 t_{56} \geq t_{45} + 9 & t_{68} \geq t_{36} + 5 & t_{89} \geq t_{58} + 7
 \end{array}$$

Задача оптимизации – минимизация следующей функции:

$$\min(\sum_{i,j} t_{i,j} + t_{end})$$

Решим эту задачу с помощью функции Matlab linprog. Для этого преобразуем полученные ранее ограничения в матрицы A и b :

```

1  %   t12 t14 t15 t16 t23 t36 t37 t45 t47 t56 t58 t67 t68 t78 t89
2  A = [1   0   0   0  -1   0   0   0   0   0   0   0   0   0   0; % 1
3       0   0   0   0   1  -1   0   0   0   0   0   0   0   0   0; % 2
4       0   0   0   0   1   0  -1   0   0   0   0   0   0   0   0; % 3
5       0   1   0   0   0   0   0  -1   0   0   0   0   0   0   0; % 4
6       0   1   0   0   0   0   0   0  -1   0   0   0   0   0   0; % 5
7       0   0   1   0   0   0   0   0   0  -1   0   0   0   0   0; % 6
8       0   0   0   0   0   0   0   0   1   0  -1   0   0   0   0; % 7
9       0   0   1   0   0   0   0   0   0   0  -1   0   0   0   0; % 8
10      0   0   0   0   0   0   0   0   1   0   0  -1   0   0   0; % 9
11      0   0   0   1   0   0   0   0   0   0   0   0  -1   0   0; % 10
12      0   0   0   0   0   1   0   0   0   0   0   0  -1   0   0; % 11
13      0   0   0   0   0   0   0   0   0   1   0  -1   0   0   0; % 12
14      0   0   0   1   0   0   0   0   0   0   0   0  -1   0   0; % 13
15      0   0   0   0   0   1   0   0   0   0   0   0  -1   0   0; % 14
16      0   0   0   0   0   0   0   0   0   1   0   0  -1   0   0; % 15
17      0   0   0   0   0   0   1   0   0   0   0   0   0  -1   0; % 16
18      0   0   0   0   0   0   0   1   0   0   0   0   0  -1   0; % 17
19      0   0   0   0   0   0   0   0   0   0   1   0  -1   0   0; % 18
20      0   0   0   0   0   0   0   0   0   0   0   0   1  -1   0; % 19
21      0   0   0   0   0   0   0   0   0   0   0   1   0  -1   0; % 20
22      0   0   0   0   0   0   0   0   0   1   0   0   0  -1   0; % 21
23  ];
24
25  b = [-6; 7; 7; 5; 5; 6; 9; 6; 9; 5; 5; 5; 5; 5; 7; 7; 9; 7; 9; 7];

```

После чего вызовем linprog:

```
1 f = ones(15, 1);
2 lb = zeros(15, 1);
3
4 linprog(f, A, b, [], [], lb, []);
```

Полученный результат выглядит следующим образом:

t_{12}	t_{14}	t_{15}	t_{16}	t_{23}	t_{36}	t_{37}	t_{45}	t_{47}	t_{56}	t_{58}	t_{67}	t_{68}	t_{78}	t_{89}
0	0	0	0	6	13	13	5	5	14	14	19	19	28	35

Табл. 2.1. Время начала всех работ.

Теперь мы знаем минимальное время начала каждой работы. Для получения информации о времени выполнения всех работ необходимо к времени начала работы t_{89} прибавить время её выполнения т.е. 8.

Итого время выполнения всех работ равно 43.

2.2. Определить наиболее ранние моменты начала работ и их интенсивности, если длительность равна интенсивности выполнения работ, а суммарная интенсивность не превышает 75% от общего числа выполняемых работ.

Мы сможем увеличить время выполнения всех работ за счет добавления интенсивностей работ, отличных от 1 – некоторые работы ускорим (интенсивность > 1), а некоторые замедлим (интенсивность < 1), если это потребуется.

Изменим исходную систему неравенств согласно правилу:

$$\min \left\{ \sum_{i,j} t_{ij} \right\}$$

$$\begin{cases} t_{ij} \geq \tau_{li} + \frac{Q_{li}}{m_{li}}, i = \overline{1, M-1}; l \in G^-(i) \\ \sum_{i,j} m_{ij} \leq 0,75 * 15, l \in G^-(M) \\ t_{ij} \geq 0 \end{cases}$$

где m_{ij} – интенсивность ij работы.

Таковыми условиями мы пытаемся минимизировать время начала всех работ, при интенсивности, не превышающей 75% от числа выполняемых работ т.е. 15.

Создадим набор всех «работ» т.е. ребер графа и массив троек, где закодируем систему неравенств, созданную ранее:

```

1  works = [12 14 15 16 23 36 37 45 47 56 58 67 68 78 89 99];
2  conds = [
3      23 12 6
4      36 23 7
5      37 23 7
6      45 14 5
7      47 14 5
8      56 15 6
9      56 45 9
10     58 15 6
11     58 45 9
12     67 16 5
13     67 36 5
14     67 56 5
15     68 16 5
16     68 36 5
17     68 56 5
18     78 37 7
19     78 47 7
20     78 67 9
21     89 78 7
22     89 68 9
23     89 58 7
24     99 89 8
25 ];

```

Стоит заметить, что появилась работа-фальшивка. Это необходимо, чтоб MATLAB оптимизировал также и путь из 8 в 9 вершину и выводил нам результат этой оптимизации. Создадим необходимые параметры для `fmincon`, а также функцию, которая распарсит заданные нам тройки в требуемые для `fmincon` значения и выведем результат выполнения на экран:

```

1  x0 = ones(length(works) * 2 - 1, 1);
2  lb = zeros(length(works) * 2 - 1, 1);
3
4  fun = @(x) sum(x(1:length(works)));
5
6  res = fmincon(fun, x0, [], [], [], [], lb, [], @funs);
7
8  function [c, ceq] = funs(x)
9      c = [];
10     for i = 1:length(conds)
11         t1 = find(works == conds(i, 1));
12         t2 = find(works == conds(i, 2));
13         q = conds(i, 3);
14         m = length(works) + t2;
15         c(end + 1) = -x(t1(1)) + x(t2(1)) + q / x(m);
16     end
17     c(end + 1) = sum(x(length(works) + 1:end)) - 0.75 * (length(works) - 1);
18     ceq = 0;
19 end
20
21 t_res = res(1:length(works));
22 m_res = res(length(works) + 1:end);
23 sum_m_res = sum(m_res);
24
25 disp(t_res);
26 disp(m_res);
27 disp(sum_m_res);

```

В результате выполнения получим следующие значения:

Моменты начала работ		Интенсивности	
t_{12}	0.00000	m_{12}	1.3138
t_{14}	0.00000	m_{14}	1.2574
t_{15}	0.00000	m_{15}	0.5536
t_{16}	0.00000	m_{16}	0.2890
t_{23}	4.56700	m_{23}	1.2735
t_{36}	10.0636	m_{36}	0.6906
t_{37}	10.0636	m_{37}	0.4130
t_{45}	3.97630	m_{45}	1.3114
t_{47}	3.97630	m_{47}	0.3039
t_{56}	10.8390	m_{56}	0.7734
t_{58}	10.8390	m_{58}	0.2723
t_{67}	17.3038	m_{67}	0.9269
t_{68}	17.3038	m_{68}	0.4677
t_{78}	27.0136	m_{78}	0.7344
t_{89}	36.5455	m_{89}	0.6692
t_{end}	48.5008		

Табл. 2.2. Результат выполнения программы.

Сумма интенсивностей равна 11.25, что составляет ровно 75% от числа исполняемых работ, как и требовалось в задании. Стоит отметить, что общее время работы возросло до 48.5 с 43 т.е. на 5.5 секунды (или в 1,13 раза). Это связано с тем, что при уменьшении интенсивности, некоторые работы стали работать дольше.

2.3. Самостоятельно распределить работы между заданным числом исполнителей и сформулировать задачу математического программирования с бинарными индикаторными переменными. Определить число бинарных переменных и дополнительных ограничений в этой задаче и дать содержательную формулировку части ограничений с бинарными переменными.

3. Вывод

4. Ссылки

5. Приложение

Листинг на github: https://github.com/DafterT/SADM_6_1