быть получена на основании анализа его динамических режимов, когда объект проявляет себя наиболее полно.

2. Увеличение расхода пара на обдув ($\Delta G_{об}$) вызывает активную теплопередачу от дымовых газов поверхностям нагрева в результате их очистки, поэтому происходит активное понижение температуры выходящих дымовых газов. При уменьшении расхода пара на обдув увеличивается степень загрязнения конвективных поверхностей нагрева, что снижает теплоотдачу, а следовательно, увеличивается и инерционность объекта.

3. Физическая интерпретация структурных параметров, определенных в результате идентификации и характеризующих динамику температуры дымовых газов, в прагматическом аспекте может быть получена на основании результатов обследования степени и характера загрязнения поверхностей нагрева.

4. Для уменьшения относительной погрешности определения структурных параметров τ_1, T_1, τ_2, T_2 желательно, чтобы относительная погрешность идентификации в окрестностях точек максимальных величин коэффициентов влияния была минимальной.

5. Динамика температуры дымовых газов, как показали исследования по идентификации их переходными функциями, может быть с достаточной для инженерной практики точностью описана линейными дифференциальными уравнениями.

Литература

1. Программа по вторичной идентификации (VtorId v1.00): программа для ЭВМ / Патрусова А. М., Колтыгин Д. С., Лузгин В. В. Св. № 2003612203 ; зарег. в реестре программ 26.09.2003.

2. Программа по идентификации передаточной функции с запаздыванием (Time-DelayId v.1.00): программа для ЭВМ / Панасов В. В., Колтыгин Д.С., Лузгин В. В. Св. ГР № 2003612203; зарег. в реестре программ 21.01. 2008.

УДК 681.3,681.5

Ю.А. Шичкина

ПРИМЕНЕНИЕ СПИСКОВ СЛЕДОВАНИЯ ДЛЯ ОПТИМИЗАЦИИ ИНФОРМАЦИОННОГО ГРАФА ПО ВЫСОТЕ

В процессе длительного использования последовательных компьютеров был накоплен и тщательно отработан огромный багаж численных методов и программ. Попытка разработать специальные параллельные методы, в частности методы малой высоты, оказалась на практике несостоятельной. Поэтому одним из решений проблемы создания эффективных численных методов для компьютеров с параллельной архитектурой является построение по последовательному алгоритму информационного графа и его преобразование с помощью разреженных матриц к параллельной форме с достаточной шириной ярусов и с последующей реализацией на параллельном компьютере. В данной статье показано, что, применяя матричный аппарат, можно не только формализовать этот процесс, но и найти оптимальное решение задачи распараллеливания алгоритма с учетом таких параметров, как количество процессоров, время вычислений и плотность вычислений на единицу времени.

Ключевые слова: параллельный алгоритм, информационный граф, ширина графа, разреженная матрица.

Введение.

В процессе длительного использования последовательных компьютеров был накоплен и тщательно отработан огромный багаж численных методов и программ. Попытка разработать специальные параллельные методы, в частности методы малой высоты, оказалась на практике несостоятельной. Если по алгоритму, записанному в виде математических соотношений или последовательных программ, удалось построить граф алгоритма, и для этого графа обнаружена параллельная форма с достаточной шириной ярусов, то рассматриваемый алгоритм можно реализовать на параллельном компьютере.

Оптимальным решением при распараллеливании является получение алгоритма, выполняемого за минимальное время с применением минимального количества процессоров.

Сегодня разработан ряд методов по преобразованию информационного графа алгоритма с целью определения экстремальных значений его отдельных параметров [1]. Большинство этих алгоритмов основаны на матрице смежности или матрице следования.

Применение матричного аппарата при исследованиях информационного графа является естественным и удачным решением в силу, опять же, большого числа существующих численных методов матричной алгебры и столь же большому внутреннему параллелизму в этих методах.

Хранить и обрабатывать все элементы матрицы информационного графа, большинство из которых нулевые, нерационально. Метод, предложенный ниже, основан не на матрицах смежности и следования, а на их аналоге – списках, что значительно экономит память, уменьшает время преобразования графа, имеет большой запас внутреннего параллелизма.

Эффективность параллельного алгоритма зависит от многих параметров. Одним из таких параметров является равномерная загрузка процессоров вычислительной системы. При этом важно, сохранив высоту параллельного алгоритма, минимизировать его ширину. Другим параметром оптимального алгоритма является время его выполнения. Если предполагать, что время выполнения всех процессов, отраженных в информационном графе в виде отдельных вершин, одинаково, тогда ориентированный ациклический информационный граф является наиболее подходящим инструментом для построения оптимального по высоте и ширине параллельного алгоритма. Но на практике такие условия – крайняя редкость. Следовательно, из-за неравномерности выполнения отдельных процессов часть процессоров будет простаивать в ожидании результатов от своих предшественников. Поэтому сама собой напрашивается замена ориентированного ациклического информационного графа его взвешенным аналогом, в котором в качестве весов будет выступать время выполнения процессов.

Применение списков следования.

Временной список следования – совокупность элементов, полученных на основе матрицы следования по правилу: если на пересечении i-й строки и j-го столбца матрицы смежности стоит единица, то номера этих строки и столбца образуют в списке элемент (v_i, v_j, t_j), где t_j – время выполнения операции v_j.

Алгоритм поиска оптимального по высоте информационного графа по списку следования заключается в разбиении совокупности вершин на группы (построение параллельной формы) и перемещение вершин между группами с целью уменьшения числа ярусов и достижения максимальной плотности по каждому ярусу.

Алгоритм:

1. Рассчитать значение теоретической минимальной ширины d информационного графа [2].

2. Добавить в граф вершину. Все выходные вершины соединить с новой вершиной. Таким образом, в графе будет всего одна выходная *n+1*-я вершина.

3. Составить матрицу следования информационного графа и ее замыкания А. Вместо единиц в матрице следования будут стоять значения времени выполнения *j*-й операции.

4. Построить соответствующий информационному графу временной список следования $V_0 = (V_1, V_2, t)$.

5. Построить множество выходных вершин V_k.

6. Найти множество $V = V_2 - V_1$. Вершины, вошедшие в это множество, составят группу вершин M_i, принадлежащих одному i-му ярусу.

– Если число вершин множества $V = V_2 - V_1$ превышает *d*, то возможны следующие варианты:

a) во множестве M_i существует ровно *d* вершин с одинаковым временем выполнения. В этом случае эти *d* вершин с одинаковым временем выполнения составят множество M_i;

b) во множестве M_i существует больше чем d вершин с одинаковым временем выполнения. В этом случае любые *d* вершин с одинаковым временем выполнения составят множество M_i.

c) во множестве M_i существует меньше *d* вершин с одинаковым временем выполнения. В этом случае множество M_i составят *d* вершин с минимальным временем выполнения.

Остальные вершины необходимо включить в следующую группу M_{i+1}. При повторном проходе этого шага в группу M_{i+1} будут добавлены вершины:

– если число вершин множества $V = V_2 - V_1$ меньше числа *d*, то во множество M_i следует перенести вершины из множества $V_к$, удовлетворяющие правилу: $V_{kj} \cap V_1 = \varnothing$, т. е. вершины из множества $V_к$, отсутствующие во множестве V_1. Эти вершины могут быть взяты в произвольном порядке из множества: $V_k - V_1$.

7. Удалить из списка смежности все пары, конечная вершина которых (V_2) совпадает с одной из вершин множества V и построить тем самым список V_i.

8. Если *i*=1, то вернуться на шаг 6.

9. Во множестве M_{i-1} найти вершину с минимальным временем выполнения m_{i-1k}. Если все вершины имеют одинаковое время выполнения, то перейти к шагу 16.

10. Составить множества:

- $S_{m_{i-1k}M_i}$ – совокупность вершин из множества M_i, связанных с вершиной m_{i-1k} одним ребром и являющихся для данного ребра конечными вершинами.
- $S_{(M_{i-1}-m_{i-1k})M_i}$ – совокупность вершин из множества M_i, связанных с вершинами множества M_{i-1} (за исключением вершины m_{i-1k}) одним ребром и являющихся для данного ребра конечными вершинами.

Внимание! Если во множестве M_{i-1} есть укрупненные операции, то рассматривается всегда только последняя добавленная при укрупнении вершина.

11. Найти разность:

$$S = S_{m_{i-1k}M_i} - S_{(M_{i-1}-m_{i-1k})M_i}.$$

- Если $S = \varnothing, S_{m_{i-1k}M_i} \neq \varnothing, S_{(M_{i-1}-m_{i-1k})M_i} \neq \varnothing$, то перейти к шагу 5.
- Если $S_{m_{i-1k}M_i} = \varnothing, S_{(M_{i-1}-m_{i-1k})M_i} = \varnothing$, то $S = M_i$.
- Если $S_{m_{i-1k}M_i} = \varnothing, S_{(M_{i-1}-m_{i-1k})M_i} \neq \varnothing$, то $S = M_i - S_{(M_{i-1}-m_{i-1k})M_i}$.

12. Найти во множестве вершину с минимальным временем выполнения $s_{\min}$. Удалить эту вершину из множества

M_i. Добавить эту вершину во множество M_{i-1}, укрупнив операцию m_{i-1k} путем ее слияния с операцией $s_{\min}$:

$$m_{i-1k} = m_{i-1k} \cup s_{\min},\ t_{i-1k} = t_{i-1k} + t_{s_{\min}}$$

13. Если $M_i = \varnothing$, то удалить множество $M_i = \varnothing$ и сдвинуть счетчик множеств $i = i - 1$. Перейти к шагу 5. Если $M_i \neq \varnothing$, то перейти к шагу 10.

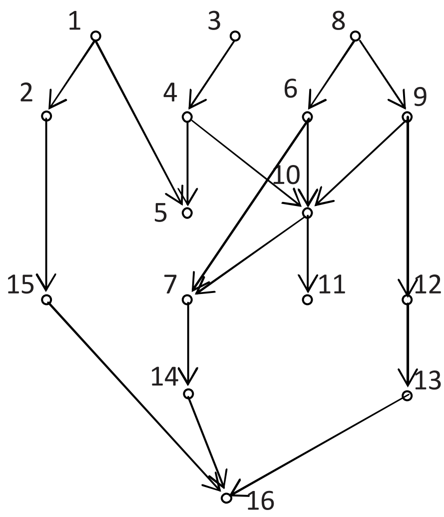
14. Если список не пустой, то вернуться на шаг 5.

15. Если в списке не осталось ни одной пары, то алгоритм окончен.

Общее время работы алгоритма составит $T = \sum_{i=1}^{k} t_{i,\max}$, где k – число групп, $t_{i,\max}$ – максимальное время среди операций в i-й группе.

Пример работы алгоритма.

Применим полученный алгоритм к информационному графу



Составим для заданного информационного графа матрицу следования и найдем замыкание:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0
5	2	0	3	2	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0
7	0	0	3	2	0	1	0	3	2	1	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0
10	0	0	3	2	0	1	0	3	2	0	0	0	0	0	0	0
11	0	0	3	2	0	1	0	3	2	1	0	0	0	0	0	0
12	0	0	0	0	0	0	0	3	2	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	3	2	0	0	3	0	0	0	0
14	0	0	3	2	0	1	2	3	2	1	0	0	0	0	0	0
15	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	2	1	3	2	0	1	2	3	2	1	0	3	1	2	3	0
17	2	1	3	2	2	1	2	3	2	1	3	3	1	2	3	1

Первоначальный список следования V_0, соответствующий данному информационному графу, имеет вид:

№	V_1	V_2	t
1	6	8	3
2	9	8	3
3	4	3	3
4	12	8	3
5	12	9	2
6	10	8	3
7	10	4	2
8	10	3	3
9	10	9	2
10	10	6	1
11	7	8	3
12	7	4	2
13	7	3	3
14	7	9	2
15	7	6	1
16	7	10	1
17	11	8	3
18	11	4	2
19	11	3	3
20	11	9	2
21	11	6	1
22	11	10	1

№	V_1	V_2	t
23	14	8	3
24	14	4	2
25	14	3	3
26	14	9	2
27	14	6	1
28	14	10	1
29	14	7	2
30	5	4	2
31	5	3	3
32	5	1	2
33	2	1	2
34	15	1	2
35	15	2	1
36	13	12	3
37	16	8	3
38	16	4	2
39	16	3	3
40	16	9	2
41	16	6	1
42	16	10	1
43	16	13	1
44	16	7	2

№	V_1	V_2	t
45	16	1	2
46	16	14	2
47	16	2	1
48	16	12	3
49	16	15	3
50	17	1	2
51	17	2	1
52	17	3	3
53	17	4	2
54	17	5	2
55	17	6	1
56	17	7	2
57	17	8	3
58	17	9	2
59	17	10	1
60	17	11	3
61	17	12	3
62	17	13	2
63	17	14	2
64	17	15	3
65	17	16	1

Найдем разность множеств $V = V_2 - V_1 = \{1, 3, 8\}$. Эта группа вершин составит первый ярус информационного графа: $M_1 = \{1, 3, 8\}$, $t(M_1) = \{2, 3, 3\}$.

Теоретическая минимальная ширина графа равна 3. Поэтому в эту группу пока не будем добавлять вершины из конечной группы V_k. Удалим из первоначального списка V_0 все пары, вторым элементом которых является одна из вершин группы M_1, получим новый список V_1 и найдем разность множеств конечных и начальных вершин ребер графа

$$M_2 = V_2 - V_1 = \{9, 4, 6, 2\},$$
$$t(M_2) = \{2, 2, 1, 1\}.$$

Во множестве M_1 найдем вершину с минимальным временем выполнения $m_{11} = 1, t_1 = 2$.

Составим множества S_{1M_2} (совокупность вершин из множества M_2, связанных с вершиной 1 одним ребром и являющихся для данного ребра конечными вершинами) и $S_{\{3,8\}M_2}$ (совокупность вершин из множества M_2, связанных с вершинами множества M_1 за исключением вершины 1 одним ребром и являющихся для данного ребра конечными вершинами). Найдем разность:

$$S = S_{1M_2} - S_{\{3,8\}M_2}.$$

S_{1M_2}	$S_{\{3,8\}M_2}$	$S = S_{1M_2} - S_{\{3,8\}M_2}$
4	4	2
6	9	
2	6	

Во множестве S одна вершина 2. Удалим эту вершину из множества M_2 и добавим ее во множество M_1, укрупнив тем самым операцию 1 путем ее слияния с операцией 2: *1'={1,2}*, $t_1 = t_1 + t_2 = 2 + 1 = 3$.

Скорректированные группы				
M_1			M_2	
i	t		j	t
1,2	3		9	2
3	3		4	2
8	3		6	1

В результате первая группа выровнена по времени. На этом ее фиксируем. Дальнейшим изменениям данная группа не подлежит.

Перейдем к составлению третьей группы. Удалим из списка V_1 все пары, первым элементом которых является одна из вершин группы M_2. В результате получим новый список V_2, $M_3 = V_2 - V_1 = \{10, 12, 15, 5\}$, $t(M_3) = \{1, 3, 3, 2\}$.

Во множестве M_2 найдем вершину с минимальным временем выполнения $m_{23} = 6, t_6 = 1$.

Составим множества S_{6M_3}, $S_{\{9,4\}M_3}$ и $S = S_{6M_3} - S_{\{9,4\}M_3}$.

S_{6M_3}	$S_{\{9,4\}M_3}$	$S = S_{6M_3} - S_{\{9,4\}M_3}$
10	10	15
15	12	
	5	

Во множестве S одна вершина 15. Удалим эту вершину из множества M_3 и добавим ее во множество M_2, укрупнив тем самым операцию 6 путем ее слияния с операцией 15: *6'={6,15}*, *t_6=t_6+t_{15}=1+3=4*.

Скорректированные группы				
M_2			M_3	
j	t		k	t
9	2		10	1
4	2		12	3
6,15	4		5	2

В результате во второй группе появились две вершины со временем выполнения меньше общего группового времени. Группа не выровнена по времени. Повторяем последние два шага для вершин с минимальным временем.

Во множестве M_2 найдем вершину с минимальным временем выполнения: $m_{21} = 9, m_{22} = 4, t_9 = t_4 = 2$. Выделим любую из них, например, 9.

Составим множества S_{9M_3}, $S_{\{4,15\}M_3}$ и $S = S_{9M_3} - S_{\{4,15\}M_3}$.

Внимание. Вершину 6 не рассматриваем, она перекрыта вершиной 15.

S_{9M_3}	$S_{\{4,15\}M_3}$	$S = S_{9M_3} - S_{\{4,15\}M_3}$
10	10	12
12	5	

Во множестве S одна вершина 12. Удалим эту вершину из множества M_3 и добавим ее во множество M_2, укрупнив тем самым операцию 9 путем ее слияния с операцией 12: *9'={9,12}*, *t_9=t_9+t_{12}=2+3=5*.

Скорректированные группы				
M_2			M_3	
j	t		k	t
9,12	5		10	1
4	2		5	2
6,15	4			

Вторая группа не выровнена по времени, а в третьей группе есть еще вершины. Повторяем последние два шага для вершин с минимальным временем.

Во множестве M_2 найдем вершину с минимальным временем выполнения $m_{22} = 4, t_4 = 2$. Составим множества S_{4M_3}, $S_{\{12,15\}M_3}$ и $S = S_{4M_3} - S_{\{12,15\}M_3}$.

S_{4M_3}	$S_{\{12,15\}M_3}$	$S = S_{4M_3} - S_{\{12,15\}M_3}$
5	$\varnothing$	5
10		10

Во множестве S две вершины 5 и 10. Удалим из множества M_3 вершину с минимальным временем выполнения и добавим ее во множество M_2, укрупнив тем самым операцию 4 путем ее слияния с операцией 10: *4'={4,10}*, *t_4=t_4+t_{10}=2+1=3*.

Скорректированные группы				
M_2			M_3	
j	t		k	t
9,12	5		5	2
4,10	3			
6,15	4			

Вторая группа не выровнена по времени, а в третьей группе есть еще вершины. Снова во множестве M_2 найдем вершину с минимальным временем выполнения: $m_{22} = \{4{,}10\} == 10,\ t_{10} = 3$.

S_{10M_3}	$S_{\{12,15\}M_3}$	$S = S_{10M_3} - S_{\{12,15\}M_3}$
$\varnothing$	$\varnothing$	5

Так как оба множества, S_{10M_3} и $S_{\{12,15\}M_3}$ являются пустыми, то их разность тоже будет пустой. В этом случае любая вершина из множества M_3 может быть перенесена во множество M_2. Поэтому в графе $S = S_{10M_3} - S_{\{12,15\}M_3}$ отображена единственная вершина множества M_3, т. е. 5. Удалим эту вершину из множества M_3 и добавим ее во множество M_2, укрупнив тем самым операцию {4,10} путем ее слияния с операцией 5: *{4,10}'={4,10,5}*, $t_{4,10}=t_{4,10}+t_5=3+2=5$.

Скорректированные группы				
M_2			M_3	
j	*t*		*k*	*t*
9,12	5		$\varnothing$	
4,10,5	5			
6,15	4			

Вторая группа не выровнена по времени, но в третьей группе больше вершин нет. Удаляем третью группу. Считаем *i*=2 и переходим к созданию новой группы из списка V_2: $M_3 = V_2 - V_1 = \{7, 13, 11\}$, $t(M_3) = \{2, 1, 3\}$.

Во множестве M_2 найдем вершину с минимальным временем выполнения $m_{23} = \{6{,}15\} = 15,\ t_{15} = 4$.

S_{15M_3}	$S_{\{12,5\}M_3}$	$S = S_{15M_3} - S_{\{12,5\}M_3}$
$\varnothing$	13	7
		11

Множество S_{15M_3} пусто, $S_{\{12,10\}M_3}$ содержит вершину. Следовательно, во множество S войдут вершины, составляющие разность множеств: $S = M_3 - S_{\{12,5\}M_3} = \{7, 11\}$. Выберем из двух вершин множества S вершину с минимальным временем: 7. Удалим эту вершину из множества M_3 и добавим ее во множество M_2, укрупнив тем самым операцию {6,15}.

Скорректированные группы				
M_2			M_3	
j	*t*		r	t
9,12	5		13	1
4,10,5	5		11	3
6,15,7	6			

Вторая группа не выровнена по времени. Во множестве M_2 две вершины с минимальным временем выполнения:

$$m_{21} = \{9{,}12\} == 12,$$

$$m_{22} = \{4{,}10{,}5\} == 5,\ t_{12} = t_5 = 5.$$

Выберем любую из них, например 12, тогда:

S_{12M_3}	$S_{\{5,7\}M_3}$	$S = S_{12M_3} - S_{\{5,7\}M_3}$
13	$\varnothing$	13

Во множестве S одна вершина 13. Удалим эту вершину из множества M_3 и добавим ее во множество M_2, укрупнив тем самым операцию {9,12} путем ее слияния с операцией 13: *{9,12}'={9,12,13}*, $t_{9,12}=t_{9,12}+t_{13}=5+1=6$.

Скорректированные группы				
M_2			M_3	
j	t		k	t
9,12,13	6		11	3
4,10,5	5			
6,15,7	6			

Вторая группа по-прежнему не выровнена по времени. Снова во множестве M_2 найдем вершину с минимальным временем выполнения:

$$m_{22} = \{4{,}10{,}5\} == 5,\ t_5 = 5\,.$$

S_{5M_3}	$S_{\{13,7\}M_3}$	$S = S_{5M_3} - S_{\{13,7\}M_3}$
$\varnothing$	$\varnothing$	11

Удалим эту вершину из множества M_3 и добавим ее во множество M_2, укрупнив тем самым операцию {4,10,5} путем ее слияния с операцией 11: *{4,10,5}'={4,10,5,11}*, $t_{4,10,5}=t_{4,10,5}+t_{11}=5+3=8$.

Скорректированные группы				
M_2			M_3	
j	t		k	t
9,12,13	6		$\varnothing$	
4,10,5,11	8			
6,15,7	6			

Вторая группа не выровнена по времени, но в третьей группе больше вершин нет. Удаляем третью группу. Считаем *i*=2 и переходим к созданию новой группы из списка V_3: $M_3 = V_2 - V_1 = \{14\}$, $t(M_3) = \{2\}$.

Во множестве $M_2 = \{13{,}11{,}7\}$ найдем вершину с минимальным временем выполнения

$$m_{23} = \{9{,}12{,}13\} == 13,$$

$$m_{23} = \{6{,}15{,}7\} == 7,\ t_{13} = t_7 = 6.$$

Выберем из двух вершин, к примеру, 13.

Составим множества S_{13M_3} и $S_{\{7,11\}M_3}$. Найдем разность:

$$S = S_{13M_3} - S_{\{7,11\}M_3}\,.$$

S_{13M_3}	$S_{\{7,11\}M_3}$	$S = S_{13M_3} - S_{\{7,11\}M_3}$
$\varnothing$	$\varnothing$	14

Так как оба множества, S_{13M_3} и $S_{\{7,11\}M_3}$ являются пустыми, то их разность тоже будет пустой. В этом случае любая вершина из множества M_3 может быть перенесена во множество M_2. Поэтому в графе $S = S_{13M_3} - S_{\{7,11\}M_3}$ отображена единственная вершина множества M_3, т. е. 14. Удалим эту вершину из множества M_3 и добавим ее во множество M_2, укрупнив тем самым операцию {9,12,13} путем ее слияния с операцией 14: *{9,12,13}'={9,12,13,14},* $t_{9,12,13}=t_{9,12,13}+t_{14}=6+2=8$.

Скорректированные группы				
M_2			M_3	
j	t		k	t
9,12,13,14	8		$\varnothing$	
4,10,5,11	8			
6,15,7	6			

Вторая группа не выровнена по времени, но в третьей группе больше вершин нет. Удаляем третью группу. Считаем i=2 и переходим к созданию новой группы из списка V_4.

После удаления всех пар из списка V_4 с первым элементом из группы M_3 получим новый список V_5 и разность множеств конечных и начальных вершин ребер графа $V = V_2 - V_1 = \{16\}$.

M_3	
p	*t*
16	1

Во множестве $M_2 = \{14{,}11{,}7\}$ найдем вершину с минимальным временем выполнения $m_{23} = \{6{,}15{,}7\} == 7,\ t_7 = 6$.

Составим множества S_{7M_3} и $S_{\{14,11\}M_3}$. Найдем разность: $S = S_{7M_3} - S_{\{14,11\}M_3}$.

S_{7M_3}	$S_{\{14,11\}M_3}$	$S = S_{7M_3} - S_{\{14,11\}M_3}$
16	16	$\varnothing$

Множество S является пустым. Фиксируем множество M_2.

После удаления всех пар из списка V_5 с первым элементом из группы M_3 получим новый список V_6, который является

пустым. Фиксируем его. На этом алгоритм закончен.
В результате мы получили три группы:

M_1	
i	t
1,2	3
3	3
8	3

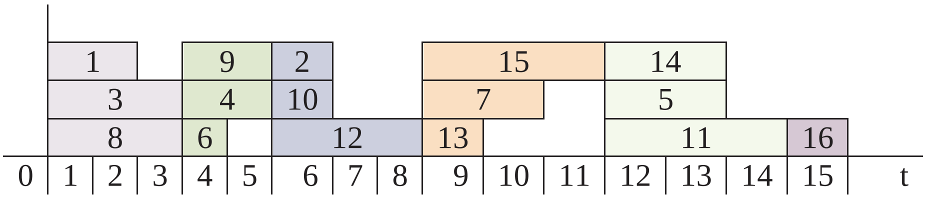
M_2	
j	t
9,12,13,14	8
4,10,5,11	8
6,15,7	6

M_3	
b	t
16	1

Общее время работы алгоритма составит $T = \sum_{i=1}^{k} t_{i,\max} = 3+8+1=12$.

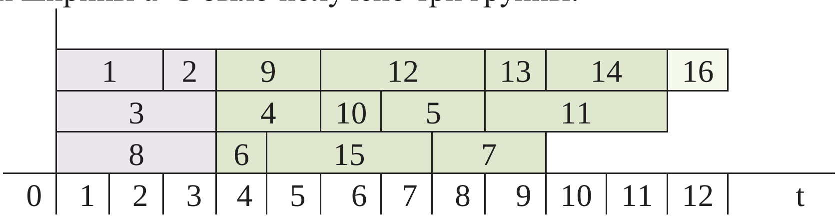
Проиллюстрируем результаты с помощью временных диаграмм.

До применения алгоритма оптимизации по времени на основе любого из методов оптимизации по ширине [1-2] были получены шесть групп операций для трех процессоров:



Время работы алгоритма равно 15 ед.

После применения метода оптимизации по времени с сохранением минимальной теоретической ширины d=3 было получено три группы:

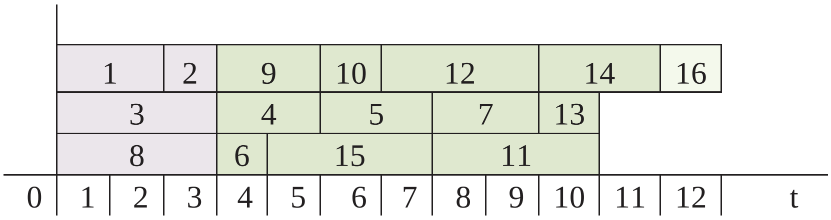


Время работы алгоритма равно 12 ед.

Заключение.

Анализ трудоемкости показал, что применение временных списков следования является более эффективным по сравнению с аналогичным алгоритмом, состоящим из двух частей [1] и основанным на матрице смежности. Метод оптимизации информационного графа по времени с применением временных списков следования позволяет значительно сократить время работы алгоритма с сохранением заданной ширины графа. При этом ширина графа может быть оговорена произвольным образом: либо на основании найденной теоретической минимальной ширины графа [1], либо в соответствии с особенностями вычислительной системы.

Следует заметить, что решение задачи минимизации алгоритма по времени вычисления может быть не единственным. Применение вышеизложенного метода оптимизации информационного графа по времени с применением временных списков следования позволяет свести время выполнения алгоритма к минимальному при заданной ширине и найти только одно из решений задачи минимизации. Так, в приведенном примере существует еще одно решение:



Время работы алгоритма равно 12 ед.

Литература

1. Шичкина Ю. А., Воробьев В. И. Оптимизация параллельного алгоритма по числу процессов // Вестн. гражданских инж. 2008. № 2 (15). С. 92-97.

2. Шичкина Ю. А. Сокращение высоты информационного графа параллельного алгоритма // Науч.-техн. ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2009. №3 (80). С. 148-152.