

Метод динамического программирования и его приложения к NP-трудным задачам

Краткий ознакомительный доклад

Е. А. Максименко

Южный федеральный университет

17 декабря 2007 г.

Пояснение

Эта презентация подготовлена для доклада на учебном семинаре «Методы решения NP-трудных задач» (руководитель семинара — М. Г. Адигеев).

Участникам семинара хорошо знакома идея метода динамического программирования (ДП), поэтому в презентации сразу приведены примеры практических задач, решаемых методом ДП, и затронута тема применения ДП к NP-трудным задачам. К сожалению, не рассказано о приближённом решении задачи о рюкзаке с помощью метода масштабирования.

Составитель презентации благодарен участникам семинара за внимание и поправки.

План доклада

1 Примеры задач из класса P , решаемых методом ДП

- Расстояние редактирования
- Другие примеры

2 Псевдополиномиальные алгоритмы

- Определение
- Поиск подмножества с заданной суммой
- Задача о рюкзаке
- Разбиение на классы похожести с минимизацией мусора

3 Задачи, NP-полные в сильном смысле

- Определение задач, NP-полных в сильном смысле
- Задача коммивояжёра
- 3-разбиение

План доклада

1 Примеры задач из класса P , решаемых методом ДП

- Расстояние редактирования
- Другие примеры

2 Псевдополиномиальные алгоритмы

- Определение
- Поиск подмножества с заданной суммой
- Задача о рюкзаке
- Разбиение на классы похожести с минимизацией мусора

3 Задачи, NP-полные в сильном смысле

- Определение задач, NP-полных в сильном смысле
- Задача коммивояжёра
- 3-разбиение

Расстояние редактирования

Элементарные операции редактирования строк

- Замена одного символа на другой:

парашу**т** \mapsto парашу**ют**, сон **н** \mapsto со**м**, культу**р** \mapsto му**ль**тур

- Вставка одного символа:

кеца**л**ь \mapsto кет**ц**аль, шик \mapsto п**ш**ик, бор \mapsto бор**щ**

- Удаление одного символа:

куз**д**ра \mapsto куз**р**а, ш**л**ак \mapsto ла**к**, медвед**ь** \mapsto медвед

Расстояние редактирования

Определение

Автор понятия

В. И. Левенштейн, Институт прикл. мат. им. М. В. Келдыша, 1965.

Определение

Расстояние редактирования между строками s_1 и s_2 — минимальное число элементарных операций редактирования, необходимое для превращения s_1 в s_2 .

Пример

$d(\text{пусто}, \text{окулист}) = 5$:

пусто \mapsto пуст \mapsto куст \mapsto окуст \mapsto окулст \mapsto окулист,
причём более короткой цепочки не существует.

Расстояние редактирования

Алгоритм вычисления методом ДП

Дано

Строка s длины m , строка t длины n .

Подзадачи

$$D[i, j] = d(s[1 \dots i], t[1 \dots j]) \quad (0 \leq i \leq m, 0 \leq j \leq n).$$

Начальные условия

$$D[i, 0] = i \quad (0 \leq i \leq m) \quad D[0, j] = j \quad (0 \leq j \leq n)$$

Рекуррентные соотношения

$$D[i, j] = \min(D[i - 1, j] + 1, \\ D[i, j - 1] + 1, \\ D[i - 1, j - 1] + (s[i] \neq t[j])).$$

Расстояние редактирования

Демонстрация работы алгоритма

Пример

$d(\text{ГРОМ}, \text{ГОРОД}) = ?$

Таблица ответов к подзадам

	∅	Г	О	Р	О	Д
∅						
Г						
Р						
О						
М						

Расстояние редактирования

Демонстрация работы алгоритма

Пример

$d(\text{ГРОМ}, \text{ГОРОД}) = ?$

Таблица ответов к подзадачам

	∅	Г	О	Р	О	Д
∅	0	→ 1	→ 2	→ 3	→ 4	→ 5
Г	1					
Р	2					
О	3					
М	4					

Расстояние редактирования

Демонстрация работы алгоритма

Пример

$d(\text{ГРОМ}, \text{ГОРОД}) = ?$

Таблица ответов к подзадачам

	∅	Г	О	Р	О	Д
∅	0	1	2	3	4	5
Г	1	0				
Р	2					
О	3					
М	4					

Diagram illustrating the dynamic programming table for edit distance between "ГРОМ" and "ГОРОД". The table shows the minimum number of operations (insertions, deletions, substitutions) required to transform one string into another. The values are: ∅ to ∅: 0; ∅ to Г: 1; ∅ to О: 2; ∅ to Р: 3; ∅ to О: 4; ∅ to Д: 5; Г to ∅: 1; Г to Г: 0; Р to ∅: 2; О to ∅: 3; М to ∅: 4. Red arrows indicate the path from (∅, ∅) to (Г, Г) via (∅, Г) and (Г, ∅).

Расстояние редактирования

Демонстрация работы алгоритма

Пример

$d(\text{ГРОМ}, \text{ГОРОД}) = ?$

Таблица ответов к подзадачам

	∅	Г	О	Р	О	Д
∅	0	1	2	3	4	5
Г	1	0	1			
Р	2					
О	3					
М	4					

Diagram illustrating the dynamic programming table for the edit distance between "ГРОМ" and "ГОРОД". The table shows the minimum number of operations (insertions, deletions, substitutions) required to transform the prefix of "ГРОМ" into the prefix of "ГОРОД". The values in the table are: (∅, ∅)=0, (Г, ∅)=1, (О, ∅)=2, (Р, ∅)=3, (О, ∅)=4, (Д, ∅)=5, (Г, Г)=0, (О, Г)=1, (Р, Г)=2, (О, Р)=3, (М, Р)=4. Red arrows indicate the path from (∅, ∅) to (Г, Г) via (∅, Г) and (∅, О), and from (∅, О) to (Г, О) via (∅, Г) and (∅, О).

Расстояние редактирования

Демонстрация работы алгоритма

Пример

$d(\text{ГРОМ}, \text{ГОРОД}) = ?$

Таблица ответов к подзадачам

	∅	Г	О	Р	О	Д
∅	0	→ 1	→ 2	→ 3	→ 4	→ 5
Г	1		→ 0	→ 1	→ 2	
Р	2					
О	3					
М	4					

Diagram illustrating the edit distance calculation between "ГРОМ" and "ГОРОД". The table shows the minimum number of operations (insertions, deletions, substitutions) required to transform one string into another. Red numbers and arrows highlight the path for the transformation: ∅ → Г → О → Р → О → Д.

Расстояние редактирования

Демонстрация работы алгоритма

Пример

$d(\text{ГРОМ}, \text{ГОРОД}) = ?$

Таблица ответов к подзадачам

	∅	Г	О	Р	О	Д
∅	0	→ 1	→ 2	→ 3	→ 4	→ 5
Г	1	0	→ 1	→ 2	→ 3	
Р	2					
О	3					
М	4					

Diagram illustrating the dynamic programming table for the edit distance between "ГРОМ" and "ГОРОД". The table shows the minimum number of operations (insertions, deletions, substitutions) required to transform the prefix of "ГРОМ" into the prefix of "ГОРОД". The values are: (∅, ∅)=0, (Г, ∅)=1, (О, ∅)=2, (Р, ∅)=3, (О, ∅)=4, (Д, ∅)=5, (Г, Г)=0, (О, Г)=1, (Р, Г)=2, (О, Г)=3. Red arrows indicate the path from (∅, ∅) to (Г, Г) via (∅, Г) and (О, Г). Dashed red arrows indicate the path from (Р, ∅) to (О, Г) via (О, Р) and (О, О).

Расстояние редактирования

Демонстрация работы алгоритма

Пример

$d(\text{ГРОМ}, \text{ГОРОД}) = ?$

Таблица ответов к подзадачам

	∅	Г	О	Р	О	Д
∅	0	→ 1	→ 2	→ 3	→ 4	→ 5
Г	1	0	→ 1	→ 2	→ 3	→ 4
Р	2					
О	3					
М	4					

Diagram illustrating the dynamic programming table for the edit distance between "ГРОМ" and "ГОРОД". The table shows the minimum number of operations (insertions, deletions, substitutions) required to transform the prefix of "ГРОМ" into the prefix of "ГОРОД". The values are: (∅, ∅)=0, (Г, ∅)=1, (О, ∅)=2, (Р, ∅)=3, (О, ∅)=4, (Д, ∅)=5, (Г, Г)=0, (О, Г)=1, (Р, Г)=2, (О, Г)=3, (Д, Г)=4, (Р, Р)=2, (О, Р)=3, (М, Р)=4, (О, О)=3, (М, О)=4, (М, М)=4. Red arrows highlight the path from (∅, ∅) to (Д, ∅) and then to (Д, Г).

Расстояние редактирования

Демонстрация работы алгоритма

Пример

$d(\text{ГРОМ}, \text{ГОРОД}) = ?$

Таблица ответов к подзадачам

	∅	Г	О	Р	О	Д
∅	0	→ 1	→ 2	→ 3	→ 4	→ 5
Г	1	0	→ 1	→ 2	→ 3	→ 4
Р	2	→ 1				
О	3					
М	4					

Diagram illustrating the dynamic programming table for the edit distance between "ГРОМ" and "ГОРОД". The table shows the minimum number of operations (insertions, deletions, substitutions) required to transform the prefix of "ГРОМ" into the prefix of "ГОРОД".

Key transitions shown with arrows:

- From (∅, ∅) to (Г, ∅): 0 → 1 (black arrow)
- From (Г, ∅) to (Г, Г): 1 → 0 (black arrow)
- From (Г, Г) to (Р, Г): 0 → 1 (red dashed arrow)
- From (Р, Г) to (Р, Р): 1 → 1 (red dashed arrow)
- From (∅, ∅) to (∅, О): 0 → 2 (black arrow)
- From (∅, ∅) to (∅, Р): 0 → 3 (black arrow)
- From (∅, ∅) to (∅, О): 0 → 4 (black arrow)
- From (∅, ∅) to (∅, Д): 0 → 5 (black arrow)
- From (Г, ∅) to (Г, О): 1 → 2 (black arrow)
- From (Г, ∅) to (Г, Р): 1 → 3 (black arrow)
- From (Г, ∅) to (Г, О): 1 → 4 (black arrow)
- From (Г, ∅) to (Г, Д): 1 → 5 (black arrow)
- From (Р, ∅) to (Р, О): 2 → 3 (black arrow)
- From (Р, ∅) to (Р, Д): 2 → 4 (black arrow)
- From (О, ∅) to (О, О): 3 → 4 (black arrow)
- From (О, ∅) to (О, Д): 3 → 5 (black arrow)
- From (М, ∅) to (М, О): 4 → 5 (black arrow)

Расстояние редактирования

Демонстрация работы алгоритма

Пример

$d(\text{ГРОМ}, \text{ГОРОД}) = ?$

Таблица ответов к подзадачам

	∅	Г	О	Р	О	Д
∅	0	1	2	3	4	5
Г	1	0	1	2	3	4
Р	2	1	1			
О	3					
М	4					

Diagram illustrating the dynamic programming table for edit distance between "ГРОМ" and "ГОРОД". The table shows the minimum number of operations (insertions, deletions, substitutions) required to transform the prefix of "ГРОМ" into the prefix of "ГОРОД".

Key transitions highlighted in red:

- From (Г, Г) to (Г, О): substitution (1 operation).
- From (Р, Г) to (Р, О): substitution (1 operation).
- From (Г, О) to (Р, О): insertion (1 operation).

Расстояние редактирования

Демонстрация работы алгоритма

Пример

$d(\text{ГРОМ}, \text{ГОРОД}) = ?$

Таблица ответов к подзадачам

	∅	Г	О	Р	О	Д
∅	0	1	2	3	4	5
Г	1	0	1	2	3	4
Р	2	1	1	1		
О	3					
М	4					

Diagram illustrating the dynamic programming table for edit distance between "ГРОМ" and "ГОРОД". The table shows the minimum number of operations (insertions, deletions, substitutions) required to transform one string into another. Red arrows indicate the path from the start (0,0) to the end (3,3), showing the sequence of operations: substitution (Г→О), insertion (Р), and deletion (О).

Расстояние редактирования

Демонстрация работы алгоритма

Пример

$d(\text{ГРОМ}, \text{ГОРОД}) = ?$

Таблица ответов к подзадачам

	∅	Г	О	Р	О	Д
∅	0	→ 1	→ 2	→ 3	→ 4	→ 5
Г	↓ 1	0	→ 1	→ 2	→ 3	→ 4
Р	↓ 2	↓ 1	↓ 1	1	→ 2	
О	↓ 3					
М	↓ 4					

Diagram illustrating the dynamic programming table for the edit distance between "ГРОМ" and "ГОРОД". The table shows the minimum number of operations (insertions, deletions, substitutions) required to transform the prefix of "ГРОМ" into the prefix of "ГОРОД". The values are: (∅, ∅)=0, (Г, ∅)=1, (О, ∅)=2, (Р, ∅)=3, (О, ∅)=4, (Д, ∅)=5, (Г, Г)=0, (О, Г)=1, (Р, Г)=2, (О, Г)=3, (Д, Г)=4, (Р, Р)=1, (О, Р)=1, (Д, Р)=2, (О, О)=3, (М, О)=4. Red dashed arrows indicate the path from (Г, Г) to (О, О) via (Р, Р).

Расстояние редактирования

Демонстрация работы алгоритма

Пример

$d(\text{ГРОМ}, \text{ГОРОД}) = ?$

Таблица ответов к подзадачам

	∅	Г	О	Р	О	Д
∅	0	→ 1	→ 2	→ 3	→ 4	→ 5
Г	1	0	→ 1	→ 2	→ 3	→ 4
Р	2	1	1	1	→ 2	→ 3
О	3					
М	4					

Diagram illustrating the dynamic programming table for edit distance between "ГРОМ" and "ГОРОД". The table shows the minimum number of operations (insertions, deletions, substitutions) required to transform one string into another. The table is filled with values, and arrows indicate the path of the optimal solution. Red dashed arrows highlight the path from (Г, 3) to (Р, 2) and (Р, 3).

Расстояние редактирования

Демонстрация работы алгоритма

Пример

$d(\text{ГРОМ}, \text{ГОРОД}) = ?$

Таблица ответов к подзадачам

	∅	Г	О	Р	О	Д
∅	0 → 1 → 2 → 3 → 4 → 5					
Г	1	0 → 1 → 2 → 3 → 4				
Р	2	1	1	1 → 2 → 3		
О	3	2				
М	4					

Diagram illustrating the edit distance calculation between "ГРОМ" and "ГОРОД". The table shows the minimum number of operations (insertions, deletions, substitutions) required to transform the prefix of "ГРОМ" into the prefix of "ГОРОД". Red numbers and arrows highlight the path for the full transformation: 0 (∅) → 2 (Р) → 3 (О) → 2 (Г) → 1 (О) → 0 (Д).

Расстояние редактирования

Демонстрация работы алгоритма

Пример

$d(\text{ГРОМ}, \text{ГОРОД}) = ?$

Таблица ответов к подзадачам

	∅	Г	О	Р	О	Д
∅	0	1	2	3	4	5
Г	1	0	1	2	3	4
Р	2	1	1	1	2	3
О	3	2	1			
М	4					

Diagram illustrating the dynamic programming table for the edit distance between "ГРОМ" and "ГОРОД". The table shows the minimum number of operations (insertions, deletions, substitutions) required to transform one string into another. Red arrows indicate the optimal path from the start (∅, ∅) to the end (М, Д).

Расстояние редактирования

Демонстрация работы алгоритма

Пример

$d(\text{ГРОМ}, \text{ГОРОД}) = ?$

Таблица ответов к подзадачам

	∅	Г	О	Р	О	Д
∅	0	→ 1	→ 2	→ 3	→ 4	→ 5
Г	1	0	→ 1	→ 2	→ 3	→ 4
Р	2	1	→ 1	→ 1	→ 2	→ 3
О	3	2	→ 1	→ 2		
М	4					

Расстояние редактирования

Демонстрация работы алгоритма

Пример

$d(\text{ГРОМ}, \text{ГОРОД}) = ?$

Таблица ответов к подзадачам

	∅	Г	О	Р	О	Д
∅	0	→ 1	→ 2	→ 3	→ 4	→ 5
Г	1	0	→ 1	→ 2	→ 3	→ 4
Р	2	1	1	1	→ 2	→ 3
О	3	2	1	→ 2	→ 1	
М	4					

Diagram illustrating the dynamic programming table for the edit distance between "ГРОМ" and "ГОРОД". The table shows the minimum number of operations (insertions, deletions, substitutions) required to transform one string into another. The table is filled with values, and arrows indicate the path of the optimal solution. Red arrows highlight the path from (0,0) to (4,6), which corresponds to the edit distance of 4.

Расстояние редактирования

Демонстрация работы алгоритма

Пример

$d(\text{ГРОМ}, \text{ГОРОД}) = ?$

Таблица ответов к подзадачам

	∅	Г	О	Р	О	Д
∅	0	→ 1	→ 2	→ 3	→ 4	→ 5
Г	1	0	→ 1	→ 2	→ 3	→ 4
Р	2	1	1	1	→ 2	→ 3
О	3	2	1	→ 2	1	→ 2
М	4					

Diagram illustrating the dynamic programming table for the edit distance between "ГРОМ" and "ГОРОД". The table shows the minimum number of operations (insertions, deletions, substitutions) required to transform the prefix of "ГРОМ" into the prefix of "ГОРОД". The values are: (∅, ∅)=0, (Г, ∅)=1, (О, ∅)=2, (Р, ∅)=3, (О, ∅)=4, (Д, ∅)=5, (Г, Г)=0, (О, Г)=1, (Р, Г)=2, (О, Г)=3, (Д, Г)=4, (Р, Р)=1, (О, Р)=1, (Д, Р)=2, (О, О)=2, (Д, О)=1, (Д, М)=2. Red dashed arrows indicate the path from (Р, О)=2 to (О, О)=1 and then to (Д, О)=2.

Расстояние редактирования

Демонстрация работы алгоритма

Пример

$d(\text{ГРОМ}, \text{ГОРОД}) = ?$

Таблица ответов к подзадачам

	∅	Г	О	Р	О	Д
∅	0 → 1 → 2 → 3 → 4 → 5					
Г	1	0 → 1 → 2 → 3 → 4				
Р	2	1	1	1 → 2 → 3		
О	3	2	1	2	1 → 2	
М	4	3				

Diagram illustrating the dynamic programming table for edit distance between "ГРОМ" and "ГОРОД". The table shows the minimum number of operations (insertions, deletions, substitutions) required to transform the prefix of "ГРОМ" into the prefix of "ГОРОД". The values are: (∅, ∅)=0, (Г, ∅)=1, (О, ∅)=2, (Р, ∅)=3, (О, ∅)=4, (М, ∅)=5, (Г, Г)=0, (О, Г)=1, (Р, Г)=1, (О, Г)=2, (М, Г)=3, (Р, О)=1, (О, О)=2, (М, О)=3, (Р, Р)=1, (О, Р)=2, (М, Р)=3, (О, Д)=1, (М, Д)=3. Red numbers and dashed arrows highlight the path from (∅, ∅) to (М, Д): (∅, ∅) → (О, ∅) → (О, О) → (М, О) → (М, Д).

Расстояние редактирования

Демонстрация работы алгоритма

Пример

$d(\text{ГРОМ}, \text{ГОРОД}) = ?$

Таблица ответов к подзадачам

	∅	Г	О	Р	О	Д
∅	0 → 1 → 2 → 3 → 4 → 5					
Г	1	0 → 1 → 2 → 3 → 4				
Р	2	1	1	1 → 2 → 3		
О	3	2	1	2	1 → 2	
М	4	3	2	2		

Diagram illustrating the dynamic programming table for the edit distance between "ГРОМ" and "ГОРОД". The table shows the minimum number of operations (insertions, deletions, substitutions) required to transform the prefix of "ГРОМ" into the prefix of "ГОРОД". The values are shown in the cells, and arrows indicate the path of the optimal solution. Red arrows highlight the path from (0,0) to (5,5): (0,0) → (1,1) → (2,2) → (3,3) → (4,4) → (5,5). Dashed red arrows show alternative paths: (1,1) → (2,2) → (3,3) → (4,4) → (5,5) and (1,1) → (2,2) → (3,3) → (4,4) → (5,5).

Расстояние редактирования

Демонстрация работы алгоритма

Пример

$d(\text{ГРОМ}, \text{ГОРОД}) = ?$

Таблица ответов к подзадачам

	∅	Г	О	Р	О	Д
∅	0 → 1 → 2 → 3 → 4 → 5					
Г	1	0 → 1 → 2 → 3 → 4				
Р	2	1	1	1 → 2 → 3		
О	3	2	1	2	1 → 2	
М	4	3	2	2		

Diagram illustrating the dynamic programming table for the edit distance between "ГРОМ" and "ГОРОД". The table shows the minimum number of operations (insertions, deletions, substitutions) required to transform the prefix of "ГРОМ" into the prefix of "ГОРОД". The values are shown in the cells, and arrows indicate the path of the optimal solution. Red numbers and arrows highlight the path from (0,0) to (5,5): (0,0) → (1,1) → (2,2) → (3,3) → (4,4) → (5,5).

Расстояние редактирования

Демонстрация работы алгоритма

Пример

$d(\text{ГРОМ}, \text{ГОРОД}) = ?$

Таблица ответов к подзадачам

	∅	Г	О	Р	О	Д
∅	0	1	2	3	4	5
Г	1	0	1	2	3	4
Р	2	1	1	1	2	3
О	3	2	1	2	1	2
М	4	3	2	2	2	2

Diagram illustrating the dynamic programming table for edit distance between "ГРОМ" and "ГОРОД". The table shows the minimum number of operations (insertions, deletions, substitutions) required to transform one string into another. The path from (0,0) to (5,5) is highlighted with arrows, showing the sequence of operations: G, R, O, M, D.

Расстояние редактирования

Демонстрация работы алгоритма

Пример

$d(\text{ГРОМ}, \text{ГОРОД}) = ?$

Таблица ответов к подзадачам

	∅	Г	О	Р	О	Д
∅	0	1	2	3	4	5
Г	1	0	1	2	3	4
Р	2	1	1	1	2	3
О	3	2	1	2	1	2
М	4	3	2	2	2	2

Diagram illustrating the dynamic programming table for edit distance between "ГРОМ" and "ГОРОД". The table shows the minimum number of operations (insertions, deletions, substitutions) required to transform one string into another. The optimal path is highlighted in red: ∅ → Г → Р → О → М.

Расстояние редактирования

Демонстрация работы алгоритма

Пример

$d(\text{ГРОМ}, \text{ГОРОД}) = ?$

Ответ

$d(\text{ГРОМ}, \text{ГОРОД}) = 2.$

ГРОМ \mapsto

ГОРОМ \mapsto

ГОРОД

Таблица ответов к подзадачам

	\emptyset	Г	О	Р	О	Д
\emptyset	0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5					
Г	1	0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4				
Р	2	1	1	1 \rightarrow 2 \rightarrow 3		
О	3	2	1 \rightarrow 2	2	1 \rightarrow 2	
М	4	3	2	2	2	2

План доклада

1 Примеры задач из класса P , решаемых методом ДП

- Расстояние редактирования
- Другие примеры

2 Псевдополиномиальные алгоритмы

- Определение
- Поиск подмножества с заданной суммой
- Задача о рюкзаке
- Разбиение на классы схожести с минимизацией мусора

3 Задачи, NP-полные в сильном смысле

- Определение задач, NP-полных в сильном смысле
- Задача коммивояжёра
- 3-разбиение

Кратчайший путь в бесконтурном графе

Постановка задачи

Задача

Дано: оргграф $G = (V, E)$, в котором нет петель и контуров, весовая функция $w: E \rightarrow \mathbb{Z}^+$ («длины дуг»), вершины $s, f \in V$.

Найти: длину кратчайшего пути от s к f .

Кратчайший путь в бесконтурном графе

Постановка задачи

Задача

Дано: оргграф $G = (V, E)$, в котором нет петель и контуров, весовая функция $w: E \rightarrow \mathbb{Z}^+$ («длины дуг»), вершины $s, f \in V$.

Найти: длину кратчайшего пути от s к f .

Замечание

На множестве V можно ввести такую нумерацию $\{v_1, \dots, v_n\}$, что $(v_i, v_j) \notin E$ при $i > j$.

Кратчайший путь в бесконтурном графе

Постановка задачи

Задача

Дано: оргграф $G = (V, E)$, в котором нет петель и контуров, весовая функция $w: E \rightarrow \mathbb{Z}^+$ («длины дуг»), вершины $s, f \in V$.

Найти: длину кратчайшего пути от s к f .

Замечание

На множестве V можно ввести такую нумерацию $\{v_1, \dots, v_n\}$, что $(v_i, v_j) \notin E$ при $i > j$.

Замечание

Можно считать, что величина $w[i, j] = w(v_i, v_j)$ определена при всех $(i, j): i < j$, но может принимать значение $+\infty$.

Кратчайший путь в бесконтурном графе

Алгоритм решения методом ДП

Дано

$w[i, j]$ ($1 \leq i < j \leq n$), s, f ($1 \leq s < f \leq n$)

Кратчайший путь в бесконтурном графе

Алгоритм решения методом ДП

Дано

$w[i, j]$ ($1 \leq i < j \leq n$), s, f ($1 \leq s < f \leq n$)

Подзадачи

$D[j]$ — длина кратчайшего пути из s в j .

Кратчайший путь в бесконтурном графе

Алгоритм решения методом ДП

Дано

$w[i, j]$ ($1 \leq i < j \leq n$), s, f ($1 \leq s < f \leq n$)

Подзадачи

$D[j]$ — длина кратчайшего пути из s в j .

Начальные условия

$D[s] = 0$

Кратчайший путь в бесконтурном графе

Алгоритм решения методом ДП

Дано

$$w[i, j] \quad (1 \leq i < j \leq n), \quad s, f \quad (1 \leq s < f \leq n)$$

Подзадачи

$D[j]$ — длина кратчайшего пути из s в j .

Начальные условия

$$D[s] = 0$$

Рекуррентные соотношения

$$D[j] = \min_{s \leq i < j} (D[i] + w[i, j]) \quad (s < j \leq f)$$

Самая длинная общая подпоследовательность

Самая длинная общая подпоследовательность

Дано: $m \in \mathbb{Z}^+$, $n \in \mathbb{Z}^+$,
 $a_i \in \mathbb{Z}$ ($1 \leq i \leq m$), $b_i \in \mathbb{Z}$ ($1 \leq i \leq n$).

Найти: $k \in \mathbb{Z}$, $k \geq 0$, i_1, \dots, i_k , j_1, \dots, j_k :
 $1 \leq i_1 < i_2 < \dots < i_k \leq m$, $1 \leq j_1 < j_2 < \dots < j_k \leq n$,
 $a_{i_1} = b_{j_1}$, $a_{i_2} = b_{j_2}$, ..., $a_{i_k} = b_{j_k}$,
 $k \rightarrow \max$.

Самая длинная общая подпоследовательность

Самая длинная общая подпоследовательность

Дано: $m \in \mathbb{Z}^+$, $n \in \mathbb{Z}^+$,
 $a_i \in \mathbb{Z}$ ($1 \leq i \leq m$), $b_i \in \mathbb{Z}$ ($1 \leq i \leq n$).

Найти: $k \in \mathbb{Z}$, $k \geq 0$, i_1, \dots, i_k , j_1, \dots, j_k :
 $1 \leq i_1 < i_2 < \dots < i_k \leq m$, $1 \leq j_1 < j_2 < \dots < j_k \leq n$,
 $a_{i_1} = b_{j_1}$, $a_{i_2} = b_{j_2}$, ..., $a_{i_k} = b_{j_k}$,
 $k \rightarrow \max$.

Пример

Дано: $m = 5$, $a = (1, 4, 2, 7, 2)$, $n = 4$, $b = (3, 4, 7, 2)$.

Ответ: $k = 3$: $(1, 4, 2, 7, 2)$, $(3, 4, 7, 2)$.

Глобальное соответствие между цепочками

Происхождение задачи

Практическая важность задачи

В биоинформатике важно оценивать близость цепочек нуклеотидов (ДНК) либо цепочек протеинов (пептиды, белки).

Глобальное соответствие между цепочками

Происхождение задачи

Практическая важность задачи

В биоинформатике важно оценивать близость цепочек нуклеотидов (ДНК) либо цепочек протеинов (пептиды, белки).

Авторы алгоритма

Saul Needleman, Christian Wunsch 1970.

Предложили алгоритм, основанный на методе ДП, для поиска наилучшего глобального соответствия между цепочками.

Глобальное соответствие между цепочками

Описание задачи

Матрица похожести нуклеотидов и штраф за разрыв

$$S = \left(\begin{array}{c|cccc} & A & G & C & T \\ \hline A & 10 & -1 & -3 & -4 \\ G & -1 & 7 & -5 & -3 \\ C & -3 & -5 & 9 & 0 \\ T & -4 & -3 & 0 & 8 \end{array} \right), \quad g = -5.$$

Глобальное соответствие между цепочками

Описание задачи

Матрица похожести нуклеотидов и штраф за разрыв

$$S = \left(\begin{array}{c|cccc} & A & G & C & T \\ \hline A & 10 & -1 & -3 & -4 \\ G & -1 & 7 & -5 & -3 \\ C & -3 & -5 & 9 & 0 \\ T & -4 & -3 & 0 & 8 \end{array} \right), \quad g = -5.$$

Пример оценки соответствия между двумя цепочками

- Цепочки: AGACTAGTTAC, CGAGACGT
- Соответствие:
AGACTAGTTAC
CGA---GACGT
- Оценка: $-3 + 7 + 10 - 3 \cdot 5 + 7 + (-4) + 0 + (-1) + 0 = 1.$

Глобальное соответствие между цепочками

Алгоритм Нидлмана-Вунша

Дано

$$S \in \mathbb{Z}^{4 \times 4}, g \in \mathbb{Z},$$

$$m \in \mathbb{Z}^+, A \in \{A, G, C, T\}^m, n \in \mathbb{Z}^+, B \in \{A, G, C, T\}^n.$$

Глобальное соответствие между цепочками

Алгоритм Нидлмана-Вунша

Дано

$S \in \mathbb{Z}^{4 \times 4}$, $g \in \mathbb{Z}$,

$m \in \mathbb{Z}^+$, $A \in \{A, G, C, T\}^m$, $n \in \mathbb{Z}^+$, $B \in \{A, G, C, T\}^n$.

Подзадачи

$F[i, j]$ — максимальная оценка среди соответствий
между $A[1, \dots, i]$ и $B[1, \dots, j]$ ($0 \leq i \leq m$, $0 \leq j \leq n$).

Глобальное соответствие между цепочками

Алгоритм Нидлмана-Вунша

Дано

$S \in \mathbb{Z}^{4 \times 4}$, $g \in \mathbb{Z}$,

$m \in \mathbb{Z}^+$, $A \in \{A, G, C, T\}^m$, $n \in \mathbb{Z}^+$, $B \in \{A, G, C, T\}^n$.

Подзадачи

$F[i, j]$ — максимальная оценка среди соответствий между $A[1, \dots, i]$ и $B[1, \dots, j]$ ($0 \leq i \leq m$, $0 \leq j \leq n$).

Начальные условия

$F[0, j] = g \cdot j$ ($0 \leq j \leq n$), $F[i, 0] = g \cdot i$ ($0 \leq i \leq m$).

Глобальное соответствие между цепочками

Алгоритм Нидлмана-Вунша

Дано

$$S \in \mathbb{Z}^{4 \times 4}, g \in \mathbb{Z},$$

$$m \in \mathbb{Z}^+, A \in \{A, G, C, T\}^m, n \in \mathbb{Z}^+, B \in \{A, G, C, T\}^n.$$

Подзадачи

$F[i, j]$ — максимальная оценка среди соответствий между $A[1, \dots, i]$ и $B[1, \dots, j]$ ($0 \leq i \leq m, 0 \leq j \leq n$).

Начальные условия

$$F[0, j] = g \cdot j \quad (0 \leq j \leq n), \quad F[i, 0] = g \cdot i \quad (0 \leq i \leq m).$$

Рекуррентные соотношения

$$F[i, j] = \max(F[i-1, j-1] + S[A[i], B[j]], F[i, j-1] + g, F[i-1, j] + g).$$

Разбиение текста на строки

Описание задачи

Простейший случай: слова нельзя переносить, все символы имеют одинаковую ширину. Даны длины слов: L_1, L_2, \dots, L_n и число M — максимальная ширина строки.

Ищется цепочка индексов: $1 = i_1 \leq i_2 \leq \dots \leq i_m = n + 1$.

В k -ю строку помещаются слова от i_k -го до $i_{k+1} - 1$.

s_k — ширина текста в k -й строке, учитывая пробелы:

$$s_k = i_{k+1} - i_k + \sum_{i=i_k}^{i_{k+1}-1} L_i.$$

Нужно выбрать индексы i_1, i_2, \dots, i_m так, чтобы

$$s_k \leq M \quad (1 \leq k \leq m), \quad \sum_{k=1}^{m-1} (M - s_k)^3 \rightarrow \min.$$

План доклада

1 Примеры задач из класса P , решаемых методом ДП

- Расстояние редактирования
- Другие примеры

2 Псевдополиномиальные алгоритмы

- Определение
- Поиск подмножества с заданной суммой
- Задача о рюкзаке
- Разбиение на классы похожести с минимизацией мусора

3 Задачи, NP-полные в сильном смысле

- Определение задач, NP-полных в сильном смысле
- Задача коммивояжёра
- 3-разбиение

Псевдополиномиальные алгоритмы: определение

Задачи с числовыми параметрами

Пусть Π — распознавательная задача с числовыми параметрами.

С каждой индивидуальной задачей I из Π связаны два числа:

- $\text{Length}(I)$ — длина входных данных (число символов, используемых для записи входных данных);
- $\text{Max}(I)$ — максимальный по абсолютной величине числовой параметр среди входных данных задачи I .

Псевдополиномиальные алгоритмы: определение

Задачи с числовыми параметрами

Пусть Π — распознавательная задача с числовыми параметрами.

С каждой индивидуальной задачей I из Π связаны два числа:

- $\text{Length}(I)$ — длина входных данных (число символов, используемых для записи входных данных);
- $\text{Max}(I)$ — максимальный по абсолютной величине числовой параметр среди входных данных задачи I .

Определение

Пусть Π — задача с числовыми параметрами. Алгоритм A решения задачи Π называют *псевдополиномиальным* (по времени), если существует такой многочлен P от двух переменных, что для любой индивидуальной задачи I из Π

$$\text{Time}(A, I) \leq P(\text{Length}(I), \text{Max}(I)).$$

План доклада

1 Примеры задач из класса P , решаемых методом ДП

- Расстояние редактирования
- Другие примеры

2 Псевдополиномиальные алгоритмы

- Определение
- Поиск подмножества с заданной суммой
- Задача о рюкзаке
- Разбиение на классы схожести с минимизацией мусора

3 Задачи, NP-полные в сильном смысле

- Определение задач, NP-полных в сильном смысле
- Задача коммивояжёра
- 3-разбиение

Поиск подмножества с заданной суммой

Постановка задачи

Задача

Дано: $n \in \mathbb{Z}^+$, $a_i \in \mathbb{Z}^+$ ($1 \leq i \leq n$), $m \in \mathbb{Z}^+$.

Вопрос: $\exists I \subset \{1, \dots, n\}: \sum_{i \in I} a_i = m$?

Поиск подмножества с заданной суммой

Постановка задачи

Задача

Дано: $n \in \mathbb{Z}^+$, $a_i \in \mathbb{Z}^+$ ($1 \leq i \leq n$), $m \in \mathbb{Z}^+$.

Вопрос: $\exists I \subset \{1, \dots, n\}: \sum_{i \in I} a_i = m$?

Пример

Дано: $n = 6$, $a = (5, 2, 7, 5, 4, 3)$, $m = 15$.

Ответ: 1 (да).

Поиск подмножества с заданной суммой

Постановка задачи

Задача

Дано: $n \in \mathbb{Z}^+$, $a_i \in \mathbb{Z}^+$ ($1 \leq i \leq n$), $m \in \mathbb{Z}^+$.

Вопрос: $\exists I \subset \{1, \dots, n\}: \sum_{i \in I} a_i = m$?

Пример

Дано: $n = 6$, $a = (5, 2, 7, 5, 4, 3)$, $m = 15$.

Ответ: 1 (да).

Пример

Дано: $n = 5$, $a = (2, 3, 2, 10, 20)$, $m = 19$.

Ответ: 0 (нет).

Поиск подмножества с заданной суммой

Алгоритм решения методом ДП

Дано

$$n \in \mathbb{Z}^+, \quad a_i \in \mathbb{Z}^+ \quad (1 \leq i \leq n), \quad m \in \mathbb{Z}^+.$$

Поиск подмножества с заданной суммой

Алгоритм решения методом ДП

Дано

$$n \in \mathbb{Z}^+, \quad a_i \in \mathbb{Z}^+ \quad (1 \leq i \leq n), \quad m \in \mathbb{Z}^+.$$

Подзадачи

$$Q[k, s] = \left(\exists I \subset \{1, \dots, k\} : \sum_{i \in I} a_i = s \right) \quad (0 \leq k \leq n, \quad 0 \leq s \leq m)$$

Поиск подмножества с заданной суммой

Алгоритм решения методом ДП

Дано

$$n \in \mathbb{Z}^+, \quad a_i \in \mathbb{Z}^+ \quad (1 \leq i \leq n), \quad m \in \mathbb{Z}^+.$$

Подзадачи

$$Q[k, s] = \left(\exists I \subset \{1, \dots, k\} : \sum_{i \in I} a_i = s \right) \quad (0 \leq k \leq n, \quad 0 \leq s \leq m)$$

Начальные условия

$$Q[k, 0] = 1 \quad (0 \leq k \leq n) \quad Q[0, s] = 0 \quad (1 \leq s \leq m)$$

Поиск подмножества с заданной суммой

Алгоритм решения методом ДП

Дано

$$n \in \mathbb{Z}^+, \quad a_i \in \mathbb{Z}^+ \quad (1 \leq i \leq n), \quad m \in \mathbb{Z}^+.$$

Подзадачи

$$Q[k, s] = \left(\exists I \subset \{1, \dots, k\} : \sum_{i \in I} a_i = s \right) \quad (0 \leq k \leq n, \quad 0 \leq s \leq m)$$

Начальные условия

$$Q[k, 0] = 1 \quad (0 \leq k \leq n) \quad Q[0, s] = 0 \quad (1 \leq s \leq m)$$

Рекуррентные соотношения

$$Q[k, s] = Q[k-1, s] \vee (a[k] \leq s \wedge Q[k-1, s-a[k]])$$

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадам

s: 0 1 2 3 4 5 6 7

∅

3

6

2

3

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3								
6								
2								
3								

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1							
6								
2								
3								

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0						
6								
2								
3								

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0					
6								
2								
3								

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1				
6								
2								
3								

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1	0			
6								
2								
3								

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1	0	0		
6								
2								
3								

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1	0	0	0	
6								
2								
3								

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1	0	0	0	0
6								
2								
3								

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1	0	0	0	0
6	1							
2								
3								

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1	0	0	0	0
6	1	0						
2								
3								

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1	0	0	0	0
6	1	0	0					
2								
3								

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1	0	0	0	0
6	1	0	0	1				
2								
3								

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1	0	0	0	0
6	1	0	0	1	0			
2								
3								

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1	0	0	0	0
6	1	0	0	1	0	0		
2								
3								

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1	0	0	0	0
6	1	0	0	1	0	0	1	
2								
3								

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1	0	0	0	0
6	1	0	0	1	0	0	1	0
2								
3								

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1	0	0	0	0
6	1	0	0	1	0	0	1	0
2	1							
3								

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1	0	0	0	0
6	1	0	0	1	0	0	1	0
2	1	0						
3								

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1	0	0	0	0
6	1	0	0	1	0	0	1	0
2	1	0	1					
3								

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1	0	0	0	0
6	1	0	0	1	0	0	1	0
2	1	0	1	1				
3								

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1	0	0	0	0
6	1	0	0	1	0	0	1	0
2	1	0	1	1	0			
3								

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1	0	0	0	0
6	1	0	0	1	0	0	1	0
2	1	0	1	1	0	1		
3								

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1	0	0	0	0
6	1	0	0	1	0	0	1	0
2	1	0	1	1	0	1	1	
3								

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1	0	0	0	0
6	1	0	0	1	0	0	1	0
2	1	0	1	1	0	1	1	0
3								

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1	0	0	0	0
6	1	0	0	1	0	0	1	0
2	1	0	1	1	0	1	1	0
3	1							

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1	0	0	0	0
6	1	0	0	1	0	0	1	0
2	1	0	1	1	0	1	1	0
3	1	0						

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1	0	0	0	0
6	1	0	0	1	0	0	1	0
2	1	0	1	1	0	1	1	0
3	1	0	1					

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1	0	0	0	0
6	1	0	0	1	0	0	1	0
2	1	0	1	1	0	1	1	0
3	1	0	1	1				

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1	0	0	0	0
6	1	0	0	1	0	0	1	0
2	1	0	1	1	0	1	1	0
3	1	0	1	1	0			

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1	0	0	0	0
6	1	0	0	1	0	0	1	0
2	1	0	1	1	0	1	1	0
3	1	0	1	1	0	1		

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1	0	0	0	0
6	1	0	0	1	0	0	1	0
2	1	0	1	1	0	1	1	0
3	1	0	1	1	0	1	1	

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1	0	0	0	0
6	1	0	0	1	0	0	1	0
2	1	0	1	1	0	1	1	0
3	1	0	1	1	0	1	1	0

Поиск подмножества с заданной суммой

Демонстрация работы алгоритма

Пример

$$n = 4$$

$$a = (3, 6, 2, 3)$$

$$m = 7$$

Ответ

0 (нет)

Таблица ответов к подзадачам

s:	0	1	2	3	4	5	6	7
\emptyset	1	0	0	0	0	0	0	0
3	1	0	0	1	0	0	0	0
6	1	0	0	1	0	0	1	0
2	1	0	1	1	0	1	1	0
3	1	0	1	1	0	1	1	0

Поиск подмножества с заданной суммой

Псевдополиномиальность алгоритма, основанного на ДП

Дано

$$n \in \mathbb{Z}^+, \quad a_i \in \mathbb{Z}^+ \quad (1 \leq i \leq n), \quad m \in \mathbb{Z}^+.$$

Поиск подмножества с заданной суммой

Псевдополиномиальность алгоритма, основанного на ДП

Дано

$$n \in \mathbb{Z}^+, \quad a_i \in \mathbb{Z}^+ \quad (1 \leq i \leq n), \quad m \in \mathbb{Z}^+.$$

Величины Max и Length для данной задачи

$$\text{Max}(I) = \max(m, n, \max_{1 \leq i \leq n} a_i);$$

$$\text{Length}(I) = \lceil \log_2 n + 1 \rceil + \lceil \log_2 m + 1 \rceil + \sum_{i=1}^n \lceil \log_2 a_i + 1 \rceil.$$

Поиск подмножества с заданной суммой

Псевдополиномиальность алгоритма, основанного на ДП

Дано

$$n \in \mathbb{Z}^+, \quad a_i \in \mathbb{Z}^+ \quad (1 \leq i \leq n), \quad m \in \mathbb{Z}^+.$$

Величины Max и Length для данной задачи

$$\text{Max}(I) = \max(m, n, \max_{1 \leq i \leq n} a_i);$$

$$\text{Length}(I) = \lceil \log_2 n + 1 \rceil + \lceil \log_2 m + 1 \rceil + \sum_{i=1}^n \lceil \log_2 a_i + 1 \rceil.$$

Псевдополиномиальность, но не полиномиальность

Время работы алгоритма пропорционально nm .

$$nm \leq \text{Max}(I) \cdot \text{Length}(I).$$

nm нельзя ограничить сверху многочленом от $\text{Length}(I)$.

План доклада

1 Примеры задач из класса P , решаемых методом ДП

- Расстояние редактирования
- Другие примеры

2 Псевдополиномиальные алгоритмы

- Определение
- Поиск подмножества с заданной суммой
- **Задача о рюкзаке**
- Разбиение на классы схожести с минимизацией мусора

3 Задачи, NP-полные в сильном смысле

- Определение задач, NP-полных в сильном смысле
- Задача коммивояжёра
- 3-разбиение

Задача о рюкзаке

Задача

Дано: $n \in \mathbb{Z}^+$ — число предметов;
 $p_i \in \mathbb{Z}^+$ ($1 \leq i \leq n$) — стоимости предметов;
 $w_i \in \mathbb{Z}^+$ ($1 \leq i \leq n$) — веса предметов;
 $c \in \mathbb{Z}^+$ — ограничение на суммарный вес.

Найти: $I \subset \{1, \dots, n\}$: $\sum_{i \in I} w_i \leq c$, $\sum_{i \in I} p_i \rightarrow \max$.

План доклада

- 1 Примеры задач из класса P , решаемых методом ДП
 - Расстояние редактирования
 - Другие примеры
- 2 Псевдополиномиальные алгоритмы
 - Определение
 - Поиск подмножества с заданной суммой
 - Задача о рюкзаке
 - **Разбиение на классы похожести с минимизацией мусора**
- 3 Задачи, NP-полные в сильном смысле
 - Определение задач, NP-полных в сильном смысле
 - Задача коммивояжёра
 - 3-разбиение

Разбиение на классы похожести с минимизацией мусора

Отношение «похожести» целых чисел

Отношение r -похожести на множестве \mathbb{Z}

Пусть $r \in \mathbb{Z}^+$. Определим бинарное отношение \sim^r на \mathbb{Z} :

$$x \sim^r y \iff |x - y| \leq r.$$

Рефлексивно и симметрично. На всём \mathbb{Z} не является транзитивным:

$$0 \sim^r r, \quad r \sim^r 2r, \quad 0 \not\sim^r 2r.$$

Разбиение на классы похожести с минимизацией мусора

Отношение «похожести» целых чисел

Отношение r -похожести на множестве \mathbb{Z}

Пусть $r \in \mathbb{Z}^+$. Определим бинарное отношение \sim^r на \mathbb{Z} :

$$x \sim^r y \iff |x - y| \leq r.$$

Рефлексивно и симметрично. На всём \mathbb{Z} не является транзитивным:

$$0 \sim^r r, \quad r \sim^r 2r, \quad 0 \not\sim^r 2r.$$

Пример таких $r \in \mathbb{Z}^+$ и $A \subset \mathbb{Z}$, что \sim^r транзитивно на A

Отношение \sim^5 транзитивно на $\{3, 4, 10, 11, 15\}$.

Классы эквивалентности: $\{3, 4\}$ и $\{10, 11, 15\}$.

Разбиение на классы похожести с минимизацией мусора

Формулировка задачи

Задача

Дано: $r \in \mathbb{Z}^+$, $n \in \mathbb{Z}^+$, $a_i \in \mathbb{Z}$ ($1 \leq i \leq n$).

Найти: $G \subset \{1, \dots, n\}$:

\sim^r транзитивно на $\{a_i: i \in \{1, \dots, n\} \setminus G\}$,

$|G| \rightarrow \min$.

Разбиение на классы похожести с минимизацией мусора

Формулировка задачи

Задача

Дано: $r \in \mathbb{Z}^+$, $n \in \mathbb{Z}^+$, $a_i \in \mathbb{Z}$ ($1 \leq i \leq n$).

Найти: $G \subset \{1, \dots, n\}$:

$\overset{r}{\sim}$ транзитивно на $\{a_i: i \in \{1, \dots, n\} \setminus G\}$,

$|G| \rightarrow \min$.

Пример

Дано: $r = 10$, $n = 8$, $a = (1, 1, 9, 11, 20, 22, 30, 40)$.

Ответ: $G = \{4, 6\}$

($\overset{10}{\sim}$ транзитивно на $\{1, 9, 20, 22, 40\}$).

План доклада

- 1 Примеры задач из класса P , решаемых методом ДП
 - Расстояние редактирования
 - Другие примеры
- 2 Псевдополиномиальные алгоритмы
 - Определение
 - Поиск подмножества с заданной суммой
 - Задача о рюкзаке
 - Разбиение на классы похожести с минимизацией мусора
- 3 Задачи, NP-полные в сильном смысле
 - Определение задач, NP-полных в сильном смысле
 - Задача коммивояжёра
 - 3-разбиение

Определение задач, NP-полных в сильном смысле

Подзадача с p -ограниченными числовыми параметрами

Пусть Π — задача с числовыми параметрами, $p \in \mathbb{Z}[x]$.

$$\Pi_p = \{I \in \Pi : \text{Max}(I) \leq p(\text{Length}(I))\}.$$

Определение задач, NP-полных в сильном смысле

Подзадача с p -ограниченными числовыми параметрами

Пусть Π — задача с числовыми параметрами, $p \in \mathbb{Z}[x]$.

$$\Pi_p = \{I \in \Pi : \text{Max}(I) \leq p(\text{Length}(I))\}.$$

Определение

Пусть Π — задача с числовыми параметрами.

Π называется *NP-полной в сильном смысле*,

если $\Pi \in \text{NP}$ и существует такой многочлен $p \in \mathbb{Z}[x]$,

что задача Π_p является NP-полной.

Определение задач, NP-полных в сильном смысле

Подзадача с p -ограниченными числовыми параметрами

Пусть Π — задача с числовыми параметрами, $p \in \mathbb{Z}[x]$.

$\Pi_p = \{I \in \Pi : \text{Max}(I) \leq p(\text{Length}(I))\}$.

Определение

Пусть Π — задача с числовыми параметрами.

Π называется *NP-полной в сильном смысле*,

если $\Pi \in \text{NP}$ и существует такой многочлен $p \in \mathbb{Z}[x]$,

что задача Π_p является NP-полной.

Связь с псевдополиномиальными алгоритмами

Если $\text{NP} \neq \text{P}$ и задача Π является NP-полной в сильном смысле, то задачу Π нельзя решить псевдополиномиальным алгоритмом.

План доклада

- 1 Примеры задач из класса P , решаемых методом ДП
 - Расстояние редактирования
 - Другие примеры
- 2 Псевдополиномиальные алгоритмы
 - Определение
 - Поиск подмножества с заданной суммой
 - Задача о рюкзаке
 - Разбиение на классы похожести с минимизацией мусора
- 3 Задачи, NP-полные в сильном смысле
 - Определение задач, NP-полных в сильном смысле
 - Задача коммивояжёра
 - 3-разбиение

Задача коммивояжёра

Формулировка задачи

Определение

Гамильтонов цикл — цикл, проходящий через каждую вершину графа ровно один раз.

Задача коммивояжёра

Формулировка задачи

Определение

Гамильтонов цикл — цикл, проходящий через каждую вершину графа ровно один раз.

Задача

Дано: полный неориентированный граф $G = (V, C_V^2)$
функция $w: C_V^2 \rightarrow \mathbb{Z}^+$ (длины рёбер),
 $s \in \mathbb{Z}^+$.

Вопрос: \exists в G гамильтонов цикл, длина которого $\leq s$?

Задача коммивояжёра

Формулировка задачи

Определение

Гамильтонов цикл — цикл, проходящий через каждую вершину графа ровно один раз.

Задача

Дано: полный неориентированный граф $G = (V, C_V^2)$
функция $w: C_V^2 \rightarrow \mathbb{Z}^+$ (длины рёбер),
 $s \in \mathbb{Z}^+$.

Вопрос: \exists в G гамильтонов цикл, длина которого $\leq s$?

NP-полнота в сильном смысле

Задача коммивояжёра NP-полная даже при ограничении $w_{i,j} \leq 2$.

План доклада

- 1 Примеры задач из класса P , решаемых методом ДП
 - Расстояние редактирования
 - Другие примеры
- 2 Псевдополиномиальные алгоритмы
 - Определение
 - Поиск подмножества с заданной суммой
 - Задача о рюкзаке
 - Разбиение на классы похожести с минимизацией мусора
- 3 Задачи, NP-полные в сильном смысле
 - Определение задач, NP-полных в сильном смысле
 - Задача коммивояжёра
 - 3-разбиение

3-разбиение

Задача

Дано: $m \in \mathbb{Z}^+$, $b \in \mathbb{Z}^+$,
 $a_i \in \mathbb{Z}^+$ ($1 \leq i \leq 3m$),

причём $\frac{b}{4} < a_i < \frac{b}{2} \quad \forall i \in \{1, \dots, 3m\}$ и $\sum_{i=1}^{3m} a_i = mb$.

Вопрос: \exists разбиение $P = \{P_1, \dots, P_m\}$ множества A :
 $\forall k \in \{1, \dots, m\} \quad \sum_{i \in P_k} a_i = b$?

3-разбиение

Задача

Дано: $m \in \mathbb{Z}^+$, $b \in \mathbb{Z}^+$,
 $a_i \in \mathbb{Z}^+$ ($1 \leq i \leq 3m$),

причём $\frac{b}{4} < a_i < \frac{b}{2} \quad \forall i \in \{1, \dots, 3m\}$ и $\sum_{i=1}^{3m} a_i = mb$.

Вопрос: \exists разбиение $P = \{P_1, \dots, P_m\}$ множества A :
 $\forall k \in \{1, \dots, m\} \quad \sum_{i \in P_k} a_i = b$?

NP-полнота в сильном смысле

3-разбиение является NP-полной задачей при ограничении $a_i \leq 12^4 m^4$.

Источники



Википедия, свободная энциклопедия.

<http://en.wikipedia.org> и <http://ru.wikipedia.org>.



ГЭРИ М., ДЖОНСОН Д. Вычислительные машины и труднорешаемые задачи. — М.: «Мир», 1982.