Московский физико-технический институт
Физтех-школа прикладной математики и информатики

ОСНОВЫ КОМБИНАТОРИКИ И ТЕОРИИ ЧИСЕЛ

II СЕМЕСТР

Лектор: *Андрей Михайлович Райгородский*

Автор: *Киселев Николай*
Репозиторий на Github

весна 2025

Содержание

«««< HEAD

0.1 Двусторонний алгоритм Дейкстры

<u>*Задача:*</u> хотим найти минимальный путь при условии, что все веса неотрицательные.

Идея решения: такая же, что и в двустороннем bfs - наращивать слои до их пересечения.

$$dist(s,t) = min_m(d_s[m] + d_t[m])$$

Пусть запущены два алгоритма Дейкстры. Будем каждый раз раскрывать вершины с $min\ d$. Завершаем алгоритм, когда какая-то вершина раскрыта с обеих сторон.

1 Алгоритм A^*

1. Заведем функцию $h(v)$ - эвристика - оценка на $dist(v,t)$

   Например, $h(v)$ - евклидово расстояние между v и t.

   Естественно, стоит брать как можно более точную оценку для эвристики, чтобы алгоритм работал эффективнее.

2. Пусть $g[v]$ - текущая min длина пути от s до v.

   $$f(v) = g[v] + h(v)$$

3. Применяем алгоритм Дейкстры на f:

```
g = {+\infty, +\infty,..., +\infty}; g[s] = 0
// Всегда поддерживаем f(v) = g[v] + h(v)
Заводим кучу, кладем в нее s;
while (куча не пуста) {
  Достаем $v$ - вершину из кучи с минимальным значением $f$
  Удаляем ее из кучи; if(v == t) break;
  Раскрываем v; обновляем соседей;
}
```

Определение 1.1. Эвристика h называется допустимой, если $\forall v: \ h(v) \leqslant dist(v,t)$

Определение 1.2. Эвристика h называется монотонной, если $h(t) = 0, \ h$удовлетворяет неравенству треугольника:

$$h(u) \leqslant h(v) + x$$

Теорема 1.1.

1. *Если h - монотонная эвристика, то в A^* каждая вершина раскрывается не больше 1 раза, причем все g находятся корректно.*

   Это в частности означает, что A^* ведет себя не хуже, чем алгоритм Дейкстры

2. *Если h - допустимая эвристика, то алго A^* может раскрывать вершины по несколько раз ($exp(n)$), но все g найдутся корректно.*

3. *Если h не является допустимой, то ничего не гарантируется. Но обычно приближается не очень плохо* ***(???)***

Лемма 1.1. *Пусть k - монотонная. Тогда вершины в A^* раскрываются в порядке неубывания f*

Доказательство. $f[v] - min$ в куче.

1. $f[u]$ не изменяется, тогда $f[u] \geqslant f[v]$, так как v - минимальная в куче

2. $f[u]$ становится равным $\cancel{g[v]} + x + h(u) \geqslant^? f(v) = \cancel{g[v]} + h(v)$, что верно по неравенству треугольника.

□

Теперь докажем утверждения теоремы:

Доказательство. По лемме f не убывает $\Longrightarrow$ вершина v не может извлечься из кучи больше 1 раза.

Почему g находится корректно?

Пусть u - первая из s нераскрытая вершина.

$f[v] = g[v] + h(v) > f[u]$?

$$f[u] = g[u] + h[u] = dist(s, u) + h(u)$$

Пусть l - размер пути от u до v

$$h(u) \leqslant h(v) + l$$

$$g[v] > dist(s, u) + l$$

$f[v] = g[v] + h(v) \geqslant dist(s, u) + l + h(v) \geqslant dist(s, u) + l + h(u) - l = f(u)$. Противоречие □

Утверждение 1.1. *$dist(s, t) = -\infty$ тогда и только тогда, когда существует цикл отрицательного веса.*

Доказательство. $\Longleftarrow$ Очевидно

$\Longrightarrow$ Пусть M - ограничение сверху на абсолютное значение всех весов, то есть $|w(e)| \leqslant M$. Рассмотрим путь из s до t веса меньше, чем $-Mn$, где n - количество ребер. Тогда, очевидно, он зацикливается. Если цикл имеет неотрицательный вес, то отбросим его, получив более короткий путь. Действуя таким образом, получим либо искомый цикл, лиюо противоречие.

□

2 Алгоритм Форда-Беллмана

$w : E \to R$. Хотим найти $dist(s, v) \forall v$

1. Заведем $dp[k][v]$ - минимальная стоимость пути от s до v, который использует не более k ребер.

2.
$$dp[0][k] = \begin{cases} 0 & v == s \\ +\infty & v \neq s \end{cases} \tag{2.1}$$

3. Переход:
$$dp[0][k] = min \begin{cases} min(dp[k-1][v]) \\ min_{(u,v)\in E} dp[k-1][u] + cost(u, v) \end{cases} \tag{2.2}$$

Замечание. Если в графе нет отрицательных циклов, то $dp[n-1][v] = dist(s, v) \forall v$

Асимптотика: $O(nm)$

Что делать в случае отрицательных циклов?

Утверждение 2.1. *1. Если C - отрицательный цикл, достижимый из S, то $\exists v \in C$: $dp[n][v] < dp[n-1][v]$*

2. Если для некоторого t : $d[n][t] < dp[n-1][t]$. то $\exists$ отрицательный цикл такой, что $S \to C \to t$.

Вывод: Чтобы найти все вершины с $dist(s, t) = -\infty$, достаточно запустить dfs от всех вершин v, для которых $dp[u][v] < dp[n-1][v]$. =======

2.1 Свойства, эквивалентные перечислимости

Далее будем считать, что наш алфавит — $\{0, 1\}$.

1. Можно выводить все элементы, но без повторов

2. Вычислима полухарактеристическая функция $\overline{\chi_A}(x) = \begin{cases} 1, x \in A \\ \text{не определено}, \ x \notin A \end{cases}$

3. A — область определения некоторой вычислимой функции

4. A — область значений некоторой вычислимой функции

5. $A = \varnothing$ или A — область значений всюду вычислимой функции.

6. A — проекция разрешимого множества пар $A = \{x | \exists y (x, y) \in B\}$, где $B \subset \{0, 1\}^* \times \{0, 1\}^*$

Утверждение 2.2. *A вычислимо $\Rightarrow$ 2)*

Доказательство.

```
chi_A(x) {
    fot i in A {
        if i == x { // если встретим x, то вернем 1
            return 1;
        }
    }
}
```

□

Утверждение 2.3. *2) $\Rightarrow$ 3)*

Доказательство. $A = Dom\ \overline{\chi_A(x)}$ □

Утверждение 2.4. *3) $\Rightarrow$ 4)*

Доказательство. Рассмотрим `f'(x)`:

```
f'(x) {
    f(x);
    return x;
}
```

Тогда $f'(x) = \begin{cases} x, x \in Dom\ f \\ \text{не определено, иначе} \end{cases}$ Заметим, что $Rad\ f' = Dom\ f$. □

Утверждение 2.5. *4) $\Rightarrow$ 5)*

Доказательство. Пусть $A = Ran\ f$. Если $A \neq \varnothing$, то положим a_0 — произвольный элемент в a_0. Положим $f' : \{0,1\}^* \times \mathbb{N} \to \{0,1\}^*$, так, что

$$f'(x,t) = \begin{cases} f(x), & \text{если } f(x) \text{ остановится за } t \text{ шагов} \\ a_0, & \text{иначе} \end{cases}$$

Заметим, что $Ran\ f = Ran\ f'$, а f' — вычислима. □

Утверждение 2.6. *5) $\Rightarrow$ 6)*

Доказательство. Пусть $A = Ran\ f$. Положим $B = \{(y,(x,t)) : f(x) = y \text{ за } t \text{ шагов}\}$. □

Утверждение 2.7. *6) $\Rightarrow$ A вычислимо*

Доказательство. Обойдем все пары (x,y), и, если $(x,y) \in B \Rightarrow$ печатаем x. □

2.2 Универсальная машина Тьюринга

Гарвардская архитектура машины — когда есть фиксированная программа и данные, с которыми она работает.

Принстонская архитектура машины — когда есть некоторый процессор, который может запускать различные программы, которые, в свою очередь, будут взаимодействовать с данными

Определение 2.1. Универсальная Машина Тьюринга — такая функция $U(M, x) = M(x)$ — по сути, машина, которая запускает машину M с вводом x.

Определение 2.2. Будем считать, что код Машины Тьюринга записан (как-то) последовательностью $0, 1$. Фунуция $U : \{0, 1\}^* \times \{0, 1\}^* \to \{0, 1\}^*$ называется универсальной вычислимой функцией, если

1. U вычислима как функция от двух аргументов
2. $\forall f : \{0, 1\}^* \to \{0, 1\}^*$, где f — вычислима, верно $\exists p \forall x\ U(p, x) = f(x)$

»»»> 416e11259fd86e9c5bb4c2fb1afb229c6b76f473