

Aula 15

Análise Amortizada

Prof. Marco Aurélio Stefanos
marco em dct.ufms.br
www.dct.ufms.br/~marco

Aula 15 – p. 1

Análise Amortizada

- O tempo exigido para realizar um conjunto de operações em uma estrutura de dados é ponderado sobre todas as operações executadas.
- Não há probabilidade envolvida
- Análise amortizada dá o tempo médio de cada operação no **pior caso**
- Há três métodos para fazer Análise amortizada: Agregado, Contagem e Potencial.

Aula 15 – p. 2

Método Agregado

Numa sequência de n operações teremos tempo de pior caso $T(n)$ no total.

O Custo amortizado, por operação é $T(n)/n$.

Operações em pilhas : $|S| = s$

- Empilha(S, x) : coloca x em S - Tempo $\Theta(1)$
- Desempilha(S, x) : retira o topo de S - Tempo $\Theta(1)$
- Multi_desempilha(S, k) : realiza o Desempilha k vezes - Tempo $\Theta(\min\{k, s\})$

Custo de n operações Empilha e Multi_desempilha partindo de uma pilha vazia?

Aula 15 – p. 3

Método Agregado

- No pior caso, Multi_desempilha é tempo $\Theta(n)$.
 - Com n operações temos $O(n^2)$
- Agora, note que
 - No. de vezes que o Desempilha é executado: no máximo o no. vezes que o Empilha é executado.
 - Empilha é executado no máximo n vezes
 - Portanto, o custo total das n operações no pior caso é $O(n)$
 - Custo amortizado (1) por operação (Distribui o custo total por todas as operações)

Aula 15 – p. 4

Método Agregado

- Contador Binário
- Vetor $A[0..k-1]$ de bits ($A[i] \in \{0, 1\}$)

$$x = \sum_{i=0}^{k-1} A[i]2^i$$

- Contador começa com $x = 0$ e é incrementado de uma unidade n vezes

Algoritmo Incrementa(A)

- 1: $i = 0$
- 2: **while** $A[i] = 1$ **do**
- 3: $A[i] = 0$
- 4: $i = i + 1$
- 5: $A[i] = 1$

Aula 15 – p. 5

Método Agregado

- Custo de cada incremento no pior caso é $\Theta(k)$
- Com n incrementos temos $O(nk)$
- Note que
 - $A[0]$ muda a cada incremento
 - $A[1]$ muda a cada 2 incrementos
 - $A[i]$ muda a cada 2^i incrementos

Total de vezes que os bits são alterados

$$\sum_{i=0}^{\lceil \lg n \rceil} \left\lceil \frac{n}{2^i} \right\rceil < n \sum_{i=0}^{\infty} \frac{1}{2^i} = 2n$$

Portanto: seq. de n incrementos tem custo $O(n)$. O Custo amortizado por operação é $O(1)$

Aula 15 – p. 6

Método de Contagem

Operações com pilhas - Custo real

- Empilha(S, x) : 1
- Desempilha(S, x) : 1
- Multi_desempilha(S, k) : $\min\{k, s\}$

Custo Amortizado - diferente para cada operação

- Empilha(S, x) : 2
- Desempilha(S, x) : 0
- Multi_desempilha(S, k) : 0

Aula 15 – p. 7

Análise por Contagem

- Quando um elemento é empilhado o custo é 1 real de gasto e sobra 1 real de crédito.
- Cada elemento na pilha tem crédito de 1
- Quando o elemento é desempilhado, a operação é paga com o crédito do elemento. Tanto no Desempilha quanto no Multi_Desempilha
- Cada desempilha é pago com antecedência. O total de crédito nunca é negativo
- Para uma seq de n operações o total do custo amortizado é um limitante superior do custo real
- Como o custo total é $O(n)$, o custo real é $O(n)$

Aula 15 – p. 8

Contador Binário - Método de Contagem

- alterar bit de 0 para 1 **1 Custo Real**
- alterar bit de 1 para 0 **1**
- alterar bit de 0 para 1 **2 Custo Amortizado**
- alterar bit de 1 para 0 **0**
- Cada bit com valor 1 tem um crédito de 1 associado. Cada chamada do Incrementa tem custo amortizado 2, pois somente um bit é alterado de 0 para 1
- O Crédito total para qualquer configuração é não negativo, Portanto, o custo amortizado é $O(n)$ e limita o custo real total.

Aula 15 – p. 9

Método potencial

- Estrutura inicial D_0 , onde n operações são realizadas, resultando em uma seqüência $D_0, D_1, D_2, \dots, D_n$
- Seja C_i o custo real da i -ésima operação.
- Uma função potencial, atribui a cada estrutura D_i um no. real $\Phi(D_i)$
- O Custo amortizado da i -ésima operação é definida por:

$$\hat{C}_i = C_i + \Phi(D_i) - \Phi(D_{i-1})$$

- Custo amortizado total

$$\begin{aligned}\sum_{i=1}^n \hat{C}_i &= \sum_{i=1}^n (C_i + \Phi(D_i) - \Phi(D_{i-1})) \\ &= \sum_{i=1}^n C_i + \Phi(D_n) - \Phi(D_0)\end{aligned}$$

- Se $\Phi(D_n) - \Phi(D_0) \geq 0$, então $\sum_{i=1}^n \hat{C}_i$ é um limitante superior para o custo real.

Aula 15 – p. 10

Método potencial - Pilhas

- $\Phi(D_i) :=$ No. de elementos na pilha
- $\Phi(D_0) =$ No. de elementos no início
- $\Phi(D_n) =$ No. de elementos no final
- Neste caso, $\Phi(D_n) - \Phi(D_0) \geq 0$
- Custo amortizado do empilha: $\Phi(D_i) - \Phi(D_{i-1}) = 1$
 $\hat{C}_i = 1 + 1 = 2$
- Multidesempilha: $k' = \min\{k, s\}$ onde $s = |D_{i-1}|$
 $\Phi(D_i) - \Phi(D_{i-1}) = -k'$
 $\hat{C}_i = C_i + \Phi(D_i) - \Phi(D_{i-1}) = k' - k' = 0$
- O custo amortizado de cada operação é $O(1)$ e portanto o custo amortizado total é $O(n)$.

Aula 15 – p. 11

Método potencial - Contador Binário

- $\Phi(D_i) = b(i)$, o número de valores 1 no contador depois de i operações INCREMENTA
- Custo real do i -ésimo INCREMENTA: Se há t_i bits reinicializados, o custo é no máximo $t_i + 1$
- Custo amortizado do INCREMENTA
- Caso $b_i = 0$ temos $b_{i-1} = t_i = k$, k no. de bits de A
- Caso $b_i > 0$ temos $b_i = b_{i-1} - t_i + 1$
- Em ambos os caso: $b_i \leq b_{i-1} - t_i + 1$
- $\Phi(D_i) - \Phi(D_{i-1}) \leq (b_{i-1} - t_i + 1) - b_{i-1} = 1 - t_i$
- Então o custo amortizado é:
 $\hat{C}_i = C_i + \Phi(D_i) - \Phi(D_{i-1}) \leq (t_i + 1) + (1 - t_i) = 2$
- Como $\Phi(D_i) \geq 0$ e o custo amortizado por operação é $O(1)$, portanto o custo amortizado total é $O(n)$.

Aula 15 – p. 12

Método potencial - Ordenação

Def.: Uma seq. tem k elementos fora de ordem se a remoção de k elementos

Problema: Analisar o algoritmo de inserção numa seqüência de n elementos com k deles fora de ordem

- Seja $A = \langle a_1, a_2, \dots, a_n \rangle$ a seq. dada.
- $D_0 = A$, D_i a seq. depois da inserção no i -ésimo elemento
- Para cada seq. B define $s_i(B) = 1$
- $s_i(B) = |\{b_j \in B \mid (b_j > b_i \text{ e } j < i) \text{ ou } (b_j < b_i \text{ e } j > i)\}|$
- Potencial de B : $\phi(B) = \sum_{i=1}^{|B|} s_i(B)/2$
- $\Phi(D_n) = 0$: D_n é a seq. ordenada

Aula 15 – p. 13

Método potencial - Ordenação

- Custo amortizado $\hat{C}_i = c_i - \Phi(D_i) - \Phi(D_{i-1})$
 $= t_i + 1 - t_i = 1$

- Seque que

$$\begin{aligned}\sum_{i=1}^n \hat{C}_i &= \sum_{i=1}^n (C_i + \Phi(D_i) - \Phi(D_{i-1})) \\ &\geq \sum_{i=1}^n C_i - kn \Rightarrow \\ \sum_{i=1}^n C_i &\leq kn + n\end{aligned}$$

- Concluimos que o custo total do Insertion é $O(kn)$

Aula 15 – p. 15

Método potencial - Ordenação

- Se $i \mid B(i)$ está fora de ordem, $s_i(D_0) \leq n - 1$
- Se $i \mid B(i)$ não está fora de ordem, $s_i(D_0) \leq k$
- Existem k elementos fora de ordem e $n-k$ elementos em ordem, portanto
- $\Phi(D_0) = k(n - 1) + (n - k)k \leq kn$
- Vamos calcular $\Phi(D_i) - \Phi(D_{i-1})$
- Suponha que o elemento a_i anda t_i posições à esq até encontrar sua posição correta.
- Então $\Phi(D_i) - \Phi(D_{i-1}) = -t_i$

Aula 15 – p. 14