Estruturas de Dados

Cristina Gomes Fernandes

ABB ótima

Problema: Dado n e uma distribuição de probabilidades (d.p.) $p_1, p_2, \ldots, p_n$, encontrar uma ABB ótima para esta d.p..

ABB ótima

Problema: Dado n e uma distribuição de probabilidades (d.p.) $p_1, p_2, \ldots, p_n$, encontrar uma ABB ótima para esta d.p..

Ou seja, encontrar uma ABB para as chaves $1, 2, \ldots, n$ cujo número esperado de comparações para uma busca com d.p. $p_1, p_2, \ldots, p_n$ é mínimo.

ABB ótima

Problema: Dado n e uma distribuição de probabilidades (d.p.) $p_1, p_2, \ldots, p_n$, encontrar uma ABB ótima para esta d.p..

Ou seja, encontrar uma ABB para as chaves $1, 2, \ldots, n$ cujo número esperado de comparações para uma busca com d.p. $p_1, p_2, \ldots, p_n$ é mínimo.

Seja $e[i, j]$ o número esperado mínimo de comparações para uma busca em uma ABB com as chaves $i, i+1, \ldots, j$, onde a busca segue a d.p. $p_i, p_{i+1}, \ldots, p_j$.

Vale a seguinte recorrência para $e[i, j]$.

Exemplo

$S = \{1, 2, 3\} \quad p_1 = 1/7 \quad p_2 = 2/7 \quad p_3 = 4/7$

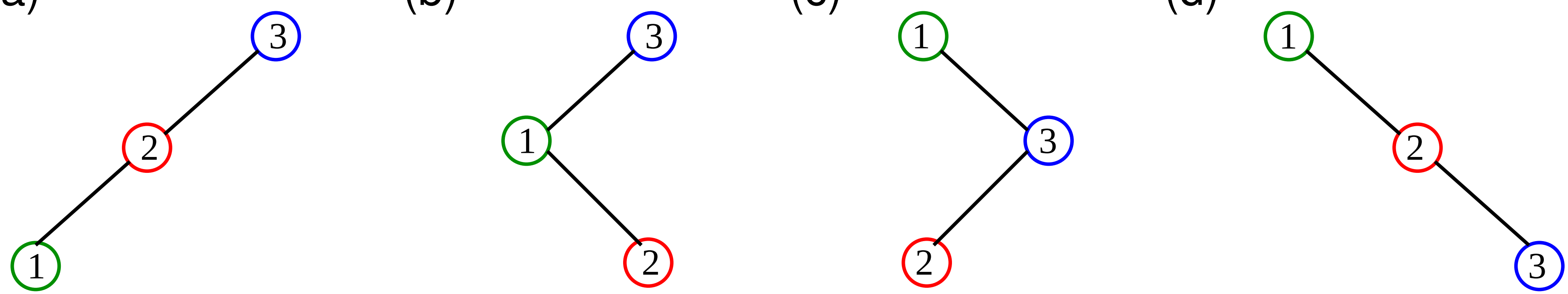
Uma outra ABB não seria melhor, dadas as freqüências?

Exemplo

$S = \{1, 2, 3\}$ $\quad p_1 = 1/7$ $\quad p_2 = 2/7$ $\quad p_3 = 4/7$

Uma outra ABB não seria melhor, dadas as freqüências?

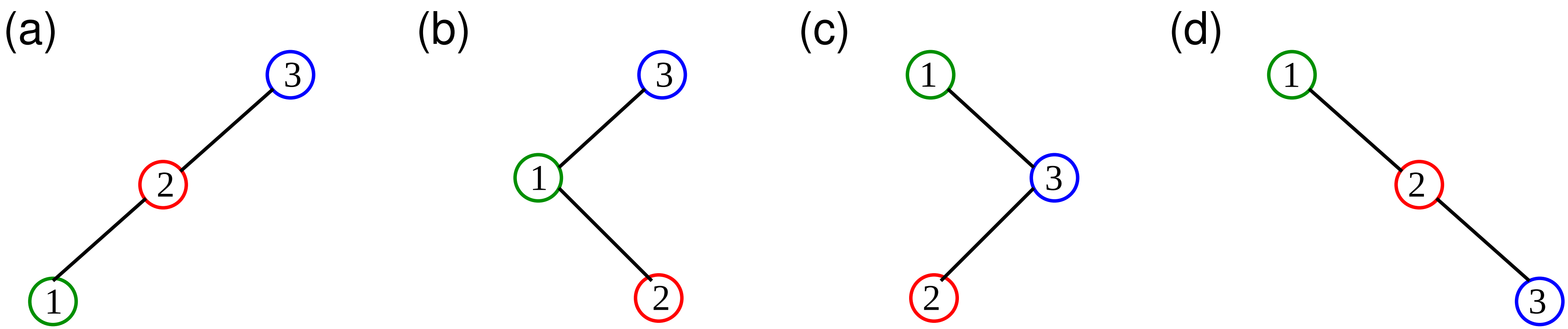
(a) (b) (c) (d)



Exemplo

$S = \{1, 2, 3\} \quad p_1 = 1/7 \quad p_2 = 2/7 \quad p_3 = 4/7$

Uma outra ABB não seria melhor, dadas as freqüências?



(a) $\frac{1}{7} \cdot 3 + \frac{2}{7} \cdot 2 + \frac{4}{7} \cdot 1 \;=\; \frac{11}{7}$

(b) $\frac{1}{7} \cdot 2 + \frac{2}{7} \cdot 3 + \frac{4}{7} \cdot 1 \;=\; \frac{12}{7}$

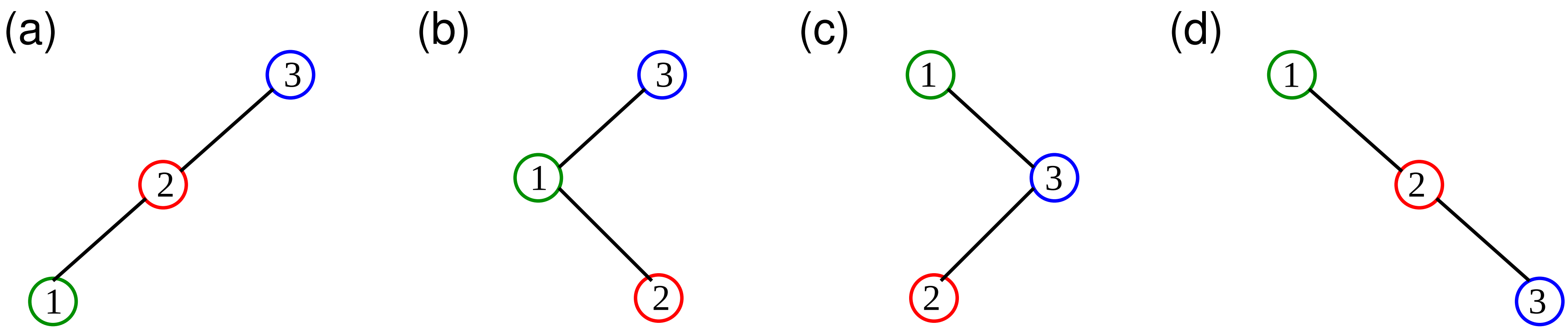
(c) $\frac{1}{7} \cdot 1 + \frac{2}{7} \cdot 3 + \frac{4}{7} \cdot 2 \;=\; \frac{15}{7}$

(d) $\frac{1}{7} \cdot 1 + \frac{2}{7} \cdot 2 + \frac{4}{7} \cdot 3 \;=\; \frac{17}{7}$

Exemplo

$S = \{1, 2, 3\}$ $\quad p_1 = 1/7 \quad p_2 = 2/7 \quad p_3 = 4/7$

Uma outra ABB não seria melhor, dadas as freqüências?



(a) $\frac{1}{7} \cdot 3 + \frac{2}{7} \cdot 2 + \frac{4}{7} \cdot 1 = \frac{11}{7}$ $\longleftarrow$ ABB ótima neste caso!

(b) $\frac{1}{7} \cdot 2 + \frac{2}{7} \cdot 3 + \frac{4}{7} \cdot 1 = \frac{12}{7}$

(c) $\frac{1}{7} \cdot 1 + \frac{2}{7} \cdot 3 + \frac{4}{7} \cdot 2 = \frac{15}{7}$

(d) $\frac{1}{7} \cdot 1 + \frac{2}{7} \cdot 2 + \frac{4}{7} \cdot 3 = \frac{17}{7}$

ABB ótima

Vale a seguinte recorrência para $e[i,j]$, o número de comparações esperado de uma ABB ótimas com as chaves de i a j para $p[i\,..\,j]$:

$$e[i,j] = \begin{cases} 0 & \text{se } i > j \\ \min_{i\le k\le j}\{e[i,k-1] + e[k+1,j]\} + w[i,j] & \text{se } i \le j \end{cases}$$

onde

$$w[i,j] \;=\; \begin{cases} 0 & \text{se } i > j \\ w[i,j-1] + p_j & \text{se } i \le j, \end{cases}$$

ou seja, $w[i,j] = \sum_{k=i}^{j} p_k$.

Podemos então usar PD para calcular $e[i,j]$ e $w[i,j]$.

ABB ótima

```
ABBÓTIMA (p, n)
para i ← 1 até n + 1 faça
    w[i, i − 1] ← 0
    e[i, i − 1] ← 0
para ℓ ← 1 até n faça
    para i ← 1 até n − ℓ + 1 faça
        j ← i + ℓ − 1
        w[i, j] ← w[i, j − 1] + p[j]
        e[i, j] ← ∞
        para k ← i até j faça
            aux ← e[i, k − 1] + e[k + 1, j] + w[i, j]
            se aux < e[i, j]
                então e[i, j] ← aux
                      r[i, j] ← k
devolva (e, r)
```

ABB ótima

Exercício: Escreva uma rotina MontaABB(r, n) que receba como parâmetro a matriz $r_{n \times n}$ devolvida pelo ABBÓTIMA(p, n) e devolva um apontador para a raiz de uma ABB ótima para $p[1 \mathinner{..} n]$.

ABB ótima: caso geral

No que segue, considere que
a 0-ésima chave é $-\infty$ e que a $(n+1)$-ésima chave é $+\infty$.

Entrada do caso geral: d.p. $q_0, p_1, q_1, \ldots, q_{n-1}, p_n, q_n$, onde
p_i é a probabilidade de se buscar a i-ésima chave, e
q_i é a probabilidade de se fazer uma busca mal-sucedida
por um x entre a i-ésima e a $(i+1)$-ésima chave.

Saída do caso geral: ABB ótima, levando em consideração na esperança as buscas mal-sucedidas.

Exercício: Resolva o caso geral da ABB ótima.

Dica: Redefina $e[i,j]$ como o valor de uma ABB ótima para buscas de valores estritamente maiores que a chave $i-1$ e estritamente menores que a chave $i+1$.
Ajuste a recorrência e o algoritmo adequadamente para essa nova definição.