CC4102 - Control 3

Prof. Gonzalo Navarro

2 de Diciembre de 2016

P1 (2 pt)

Considere un algoritmo online aleatorizado para el problema de arrendar los esquíes. Si el costo de arrendar por día es a y de comprar es c, este algoritmo decide comprar el día $\lfloor c/(2a) \rfloor + 1$ con probabilidad p, y sino compra el día $\lfloor c/a \rfloor + 1$ como el determinístico.

1. (1pt) Halle el costo esperado que paga el algoritmo aleatorizado hasta el día t, que es el día (desconocido) en el que se debe volver del resort. Separe en casos $t \leq c/(2a)$, $c/(2a) < t \leq c/a$, y $t > c/a$.
2. (1pt) Encuentre el valor de p que optimiza la competitividad esperada del algoritmo y muestre que ésta resulta ser menos que 2.
3. (Bono 0.5pt) Muestre que ningún algoritmo determinístico puede ser mejor que 2-competitivo.

P2 (2 pt)

Usted debe llenar un camión que soporta hasta n toneladas de peso, y va recibiendo los paquetes uno a uno, debiendo decidir en el momento si sube cada paquete al camión o no. La secuencia de paquetes es potencialmente infinita. El objetivo es llenar el camión lo más posible, y la competitividad se mide por el peso máximo que se puede poner en el camión eligiendo los paquetes en forma óptima dividido por el peso que logra poner un algoritmo online.

1. (1pt) Demuestre que, si los paquetes subidos al camión no se pueden bajar, entonces ninguna estrategia online puede ser c-competitiva para ninguna constante c (n no es una constante).
2. (1pt) Considere ahora el caso en que se le permite, al ver un nuevo paquete, bajar todos los que puso hasta ahora en el camión y subir ese nuevo paquete. Diseñe una estrategia 2-competitiva para esta variante.
3. (Bono 0.5pt) Para la segunda variante, demuestre que ninguna estrategia online puede ser $(2-\epsilon)$-competitiva, para ninguna constante $\epsilon > 0$.

P3 (2 pt)

Considere un árbol de altura h perfectamente balanceado, donde cada nodo interno tiene 3 hijos, por lo que hay $n = 3^h$ hojas. Cada hoja contiene un valor booleano, 0 ó 1. El valor booleano de cada nodo interno se calcula como el mayoritario entre sus 3 hijos.

1. (1pt) Muestre que cualquier algoritmo determinístico necesita en el peor caso examinar las $n = 3^h$ hojas para encontrar el valor de la raíz.
2. (1pt) Considere un algoritmo aleatorizado que elige dos hijos al azar, los evalúa recursivamente, y si dan el mismo valor evita calcular el tercero, de otro modo calcula el tercero para responder. Demuestre que el número esperado de hijos analizado por cada nodo es $8/3$.
3. (Bono 0.5pt) Analice el costo esperado del algoritmo aleatorizado y muestre que es $o(n)$.

Tiempo: 2.0 horas | Con una hoja de apuntes | Responder en hojas separadas