

Auxiliar 10 - Algoritmos Online y Paralelos

CC4102/CC53A - Diseño y Análisis de Algoritmos
Profesor: Pablo Barceló Auxiliar: Miguel Romero

29 de Mayo del 2014

1. Tiene usted $k\$$ y quiere gastarse la mayor cantidad posible en una feria a lo largo de una calle de un solo sentido (es decir, usted solo puede hacer una pasada por ella). El precio de cada producto varía entre $1\$$ y $k\$$. En cada momento usted puede devolver uno o más productos ya comprados y recuperar el dinero (pero no puede comprar algo por lo que ya pasó). Muestre un algoritmo 2-competitivo para este problema.
2. (Control 3 2012/02) Considere dos robots en una línea recta, y una secuencia de pedidos que implican que alguno de ellos se traslade a cierto punto de la recta. Se desea minimizar la distancia total recorrida por ambos robots. Si se conoce la secuencia de antemano, se puede construir un algoritmo óptimo (no importa cómo, para esta pregunta). Se busca un algoritmo competitivo que reciba cada pedido después de haber servido el anterior.
 - (a) Muestre que la estrategia de mover el robot más cercano al pedido no es competitiva.
 - (b) Muestre que modificando la estrategia para que, si el pedido cae entre los dos robots, se muevan *ambos* hacia el punto del pedido, hasta que uno de ellos llegue, es 2-competitiva. Para ello, llamemos s_1 y s_2 las posiciones de los robots en la estrategia óptima, y a_1 y a_2 en la aproximada, con $s_1 \leq s_2$ y $a_1 \leq a_2$. Considere la función potencial $\Phi = 2(|a_1 - s_1| + |a_2 - s_2|) + |a_1 - a_2|$. Muestre que (1) cuando el óptimo se mueve en x , $\Delta\Phi \leq 2x$, y (2) cuando el algoritmo online paga costo x , $\Delta\Phi \leq -x$. Muestre que eso implica que el algoritmo es 2-competitivo.
3. (Examen 2012/01) Considere paralelizar el problema de multiplicar dos matrices de $n \times n$. Considere que $T(n, 1) = O(n^3)$ (es decir, ignorando algoritmos mejores que el estándar).
 - (a) Proponga un algoritmo de tiempo $O(n)$ usando n^2 procesadores. Calcule $T(n)$, $W(n)$ y $E(n)$. ¿Qué modelo PRAM usa?
 - (b) Proponga un algoritmo de tiempo $O(\log n)$ usando n^3 procesadores. Calcule $T(n)$, $W(n)$ y $E(n)$.
 - (c) Mejore la eficiencia del algoritmo anterior para que sea $\Theta(1)$. ¿Qué tiempo obtiene y con cuantos procesadores?