Una semplice prova della $(2k-1)$-competitività di ω-WFA

29 novembre 2011

Vogliamo mostrare come l'ottimizzazione dell'algoritmo Work con una finestra di ricerca sia $(2k-1)$-competitivo esattamente come l'algoritmo originale modulo alcune assunzioni. In primo luogo considereremo uno spazio metrico finito, la cui distanza massima è Δ inoltre consideriamo una discretizzazione di questo spazio tale che la distanza minima tra le richieste si δ. In generale queste assunzioni non sono limitanti dal momento che le implementazioni effettive sul calcolatore hanno intrinsecamente queste limitazioni.

La prima scelta che operiamo è la dimensione della finestra. D'ora in avanti chiameremo tale dimensione ω e l'algoritmo derivante ω-WFA. Scegliamo $\omega \geq \Delta/\delta$ e $\omega > k$. Dove k è il numero di server a disposizione.

Il nostro algoritmo andrà a calcolare la funzione lavoro esattamente nello stesso modo in cui viene calcolata nell'algoritmo Work con la differenza che alla richiesta t-esima:

$$work(t, A_t) = \begin{cases} w_t(A_t) & \text{se } t < \omega \\ w_\omega(A_t) & \text{se } t \geq \omega \end{cases}$$

Considerate le premesse andremo a dimostrare il fatto che ω-WFA è $(2k-1)$-competitivo basandoci sull'assunzione che Work lo sia perciò qualsiasi risultato migliore venga provato per Work si propaga all'ottimizzazione in analisi.

Lemma 1. *Siano $\rho = r_1, ..., r_\omega$ richieste distinte. Allora*

$$C_{\omega-WFA}(\rho) \leq C_{OPT}(\rho) + cost$$

Dimostrazione. Ovvio dalla definizione di ω-WFA. □

Questo lemma ci dice che, se dividiamo in fasi lunghe ω la sequenza di richieste (supponiamo siano n) alla fine sarà valida la disuguaglianza:

$$C_{\omega-\text{WFA}}(\rho) \leq C_{\text{OPT}}(\rho) + cost$$

la costante in questo caso sarà $cost \leq \Delta kn$. Dobbiamo dimostrare che la costante non aumenta linearmente con il numero di fasi. In questo caso l'algoritmo è competitivo come Work.