CMPSCI 711: More Advanced Algorithms

Section 4-1: Sequences

Andrew McGregor

Last Compiled: April 29, 2012

Outline

Sequences and Strings

Longest Increasing Subsequence

Bracket Matching

- Consider a length m stream of round and square brackets, e.g.

$$[\;] \; (\; [\; [\;] \;] \; (\; [\;] \;) \;)$$

- Say sequence is *well-balanced* if corresponding open and close brackets are of the same type.
- Can we recognize a well-balanced sequence in $O(\sqrt{n})$ space?

First Part of Result

- Say a sequence is *t-turn* if there are t open brackets immediately followed by a close bracket. E.g., following sequence is 3-turn:

$$[\ ]\ (\ [\ [\ ]\ ]\ (\ [\ ]\ )\ )$$

- Helpful to consider sequence in 2D representation where *height* corresponds to excess of open brackets amongst preceding brackets.

```
            [ ]       [ ]
          [     ] (       )
[ ] (                       )
```

- Using $O(t)$ memory, it's possible to determine height h of each new bracket and whether it's the 1st, 2nd, ... bracket at this height.
- As we read each symbol α, generate updates to $x, y \in \{0, 1, 2\}^{t \times m}$

$$x_{h,i} \leftarrow x_{h,i} + \begin{cases} 1 & \text{if } \alpha = \text{“(”} \\ 2 & \text{if } \alpha = \text{“[”} \end{cases} \qquad y_{h,i} \leftarrow y_{h,i} + \begin{cases} 1 & \text{if } \alpha = \text{“)”} \\ 2 & \text{if } \alpha = \text{“]”} \end{cases}$$

if α is the i-th bracket at height h. Then it suffices to check whether $x = y$ and this can be done by sketching or finger-printing.

Second Part of Result

- If we can ensure $t = O(\sqrt{n})$ the desired result follows.
- Buffer length-$\sqrt{n}$ segments and perform all cancelations within each segment before passing it to the t-turn algorithm. E.g.,

$$[\ ]\ (\ \mid\ [\ [\ ]\ \mid\ ]\ (\ [\ \mid\ ]\ )\ )$$

becomes

$$(\ \mid\ [\ \mid\ ]\ (\ [\ \mid\ ]\ )\ )$$

Outline

Sequences and Strings

Longest Increasing Subsequence

Length of Longest Increasing Subsequence

- *Goal:* Estimate the length of the longest increasing subsequence.
- *Idea:* For $\ell \geq 0$, maintain

  $$A[\ell] = \text{smallest value that ends a length } \ell \text{ increasing sequence so far}$$

  - Initially, $A[0] = -\infty$ and $A[\ell] = \infty$ for all $\ell > 0$
  - If the new element is v, find smallest ℓ such that $A[\ell] \leq v$ and set

    $$A[\ell + 1] \leftarrow v$$