Advanced Algorithmics

Strategies for Tackling Hard Problems

Sebastian Wild

Markus Nebel

Lecture 7

2017-05-11

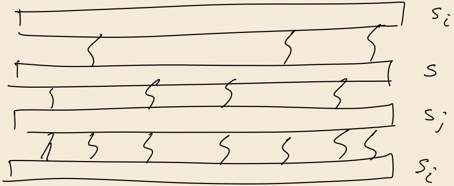
Depth-Bounded Search for Closest String

closestStringFpt(s_1, k)

procedure closestStringFpt(s, d):
    if $d < 0$ **then return** "not found"
    if $d_H(s, s_i) > k + d$ **for** an $i \in \{1, \ldots, m\}$ **then**
        return "not found"
    if $d_H(s, s_i) \le k$ **for** all $i = 1, \ldots, m$ **then return** s
    Choose $i \in \{1, \ldots, m\}$ arbitrarily with $d_H(s, s_i) > k$
        $P := \{p : s[p] \neq s_i[p]\}$
        Choose arbitrary $P' \subseteq P$ with $|P'| = k + 1$
        for p **in** P' **do**
            $s' := s$
            $s'[p] := s_i[p]$
            $s_{ret} :=$ closestStringFpt$(s', d - 1)$
            if $s_{ret} \neq$ "not found" **then return** s_{ret}
    return "not found"

Lemma 3.42 (Pair Too Different → No)

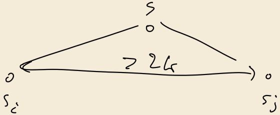
Let $S = \{s_1, s_2, \ldots, s_m\}$ a set of strings and $k \in \mathbb{N}$. If there are $i, j \in \{1, \ldots, m\}$ with $d_H(s_i, s_j) > 2k$, then there is no string s with $\max_{1 \le i \le m} d_H(s, s_i) \le k$.

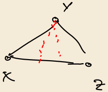


$k = 3$

d_H is a metric.

- $d_H(x,y) \geq 0$ ✓
- $d_H(x,x) = 0$ ✓
- Δ-ineq. $d_H(x,z) \leq d_H(x,y) + d_H(y,z)$





$\square$

Theorem 3.43 (Search Tree for Closest String)

There is a search tree of size $\mathcal{O}(k^k)$ for problem p-CLOSEST-STRING. & solves the problem.

```
procedure closestStringFpt(s, d):
    if d < 0 then return "not found"
    if d_H(s, s_i) > k + d for an i ∈ {1, ..., m} then
        return "not found"
    if d_H(s, s_i) ≤ k for all i = 1, ..., m then return s
    Choose i ∈ {1, ..., m} arbitrarily with d_H(s, s_i) > k
        P := {p : s[p] ≠ s_i[p]}
        Choose arbitrary P' ⊆ P with |P'| = k + 1
        for p in P' do
            s' := s
            s'[p] := s_i[p]
            s_ret := closestStringFpt(s', d − 1)
            if s_ret ≠ "not found" then return s_ret
    return "not found"
```

Fact: s can differ at at most k positions from any input.

If any candidate s has too many differences to any s_i $\leadsto$ ↯

W.l.o.g. $L \leq m \cdot k$ and all columns dirty.

Size of Search Tree:

depth $\leq k$

fan-out $\leq k+1$

$\Rightarrow$ search space $\leq (k+1)^k = O(k^k)$

$$\frac{(k+1)^k}{k^k} = \left(1 + \frac{1}{k}\right)^k < e$$

Correctness:

if we return a string ✓

if we abort : to show $\nexists$ consensus

Assume s_1 is not directly a consensus, but $\exists$ consensus $\hat{s}$

$\Rightarrow \quad \forall i \quad d_H(s_1, s_i) \leq 2k$

L. 3.42

s_i with $d_H(s_i, s_1) > k \qquad P := \{ j \; ; \; s_1[j] \neq s_i[j] \}$

$|P| \leq 2k$

$|P'| = k + 1 \qquad P' \subseteq P$

We call j correct if $j \in P_1 = \{ p \; ; \; s_1[p] \neq s_i[p] \wedge s_i[p] = \hat{s}[p] \}$

otherwise $\quad j \in P_2 = \{ p \in P \; ; \; s_i[p] \neq \hat{s}[p] \}$

$P = P_1 \,\dot{\cup}\, P_2 \qquad d_H(\hat{s}, s_i) \leq k \rightsquigarrow |P_2| \leq k$

$\Rightarrow$ at least one j in P' is correct. $\quad (P' \cap P_1 \neq \emptyset)$

$\Rightarrow$ There is a path of modification to $\hat{s}$.

Corollary 3.44 (Closest String is FPT)

$\propto$ recursive calls

p-CLOSEST-STRING can be solved in time $\mathcal{O}(\underbrace{mL}_{\text{"preprocessing"}} + \underbrace{mk^2}_{} \cdot \overbrace{k^k}^{})$.

"preprocessing" deleting non-dirty columns and check $\leq m \cdot k$ columns remain

3.5 Interleaving

Up to now, considered two-phase algorithms

1. Reduction to problem kernel
2. Solve kernel by depth-bounded exhaustive search

Idea: Apply kernelization *in each recursive step*.

Setting for Interleaving

Assumptions: (more restrictive than general kernelization!)

- K kernelization that
  - produces *kernel of size* $\leq q(k)$ for q a *polynomial* (closest string ex+)
  - in time $\leq p(n)$ for p a polynomial
- Branch in depth-bounded search tree
  - into i subproblems with branching vector $\vec{d} = (d_1, \ldots, d_i)$ (i.e.parameter in subproblems $k - d_1, \ldots, k - d_i$)
  - Branching is computed in time $\leq r(n)$ for r a polynomial
- search space has size $\mathcal{O}(\alpha^k)$.

$\rightsquigarrow$ Running time of two-phase approach on input x with $n = |x|$ and $k = \kappa(x)$:

$$\mathcal{O}\Big(\overset{K(x)}{p(n)} + \underbrace{r(q(k))}_{\text{time per recursive call}} \cdot \underbrace{\alpha^k}_{\text{\# recursive calls}}\Big)$$

With Interleaving

Now replace splitting by: c tuning parameter

---

if $|I| > c \cdot q(k)$ **then**
    $(I, k) := (I', k')$ where (I', k') forms a problem kernel // *Conditional Reduction*
end;
replace (I, k) with $(I_1, k - d_1), (I_2, k - d_2), \ldots, (I_i, k - d_i)$. // *Branching*

$\rightsquigarrow$ Running time of interleaved approach on input x with $n = |x|$ and $k = \kappa(x)$ is at most T_k:

$$T_\ell = T_{\ell-d_1} + \cdots + T_{\ell-d_i} + \underbrace{p(q(\ell)) + r(q(\ell))}_{\text{polynomial in } \ell}$$

$$T_k \overset{\text{Thm 3.85}}{=} O(\alpha^k)$$

Compare to non-interleaved version:

$$T'_\ell = T'_{\ell-d_1} + \cdots + T'_{\ell-d_i} \rightsquigarrow T'_\ell = O(\alpha^\ell)$$

$$T_\ell = T_{\ell-d_1} + \cdots + T_{\ell-d_i} + r(q(k)) \qquad T_\ell = (r(q(k)) \cdot \alpha^k)$$

Here the inhomogeneous term is constant w.r.t. ℓ, but depends on k
$\rightsquigarrow$ cannot ignore constant factors

Theorem 3.45 (Linear Recurrences II)

Let $d_1, \ldots, d_i \in \mathbb{N}$ and $d = \max d_j$.
Consider the *inhomogeneous linear recurrence equation*

$$T_n = T_{n-d_1} + T_{n-d_2} + \cdots + T_{n-d_i} + f_n, \qquad (n \geq d)$$

with $(f_n)_{n \in \mathbb{R}_{>0}}$ a known sequence of positive numbers and d initial values $T_0, \ldots, T_{d-1} \in \mathbb{R}_{>0}$.

$f_n = O(n^c)$

Let z_0 be the root with largest absolute value of $z^d - \sum_{j=1}^{i} z^{d-d_j}$ and assume $f_n = \mathcal{O}((z_0 - \varepsilon)^n)$ for some fixed $\varepsilon > 0$.
Then $T_n = \mathcal{O}(T_n^0)$ where T_n^0 is defined as T_n with $f_n \equiv 0$.

$$T(z) = \sum_{n \geq 0} T_n z^n = \underbrace{T_0 z^0 + \cdots + T_{d-1} z^{d-1}}_{P(z)} + \sum_{j=1}^{i} \sum_{n \geq d_j} z^n T_{n-d_j} + \underbrace{\sum_{n \geq d} z^n f_n}_{F(z)}$$

$$= P(z) + F(z) + \left(\sum_{j=1}^{i} z^{d_j} \right) T(z) - Q(z)$$

$$T(z) = \frac{P(z) + Q(z) + F(z)}{1 - \sum_{j=1}^{i} z^{d_j}} \qquad \begin{array}{l} = N(z) \\ = D(z) \end{array}$$

$N(z) = \text{polynomial} + F(z)$

$F(z)$ is analytic $|z| < z_0$

$$\leadsto T_n = N(z_0) \cdot C / (\mu - 1)! \cdot z_0^{-n} n^{\mu-1} (1 \pm O(n^{-1}))$$

A Little Excursion: Singularity Analysis

$$f(z) = \frac{N(z)}{D(z)}$$

$D(z)$ polynomial

z_0 real root of $D(z)$ with smallest absolute value

and $N(z)$ analytic for all $|z| < z_0$

"smooth"

$$= N(z) \left(\text{partial fractions of } \frac{1}{D(z)} \right)$$

$$\sum_{z_\lambda} \sum_{j=1}^{\mu_\lambda} \frac{c_{\lambda,j}}{\left(1 - \frac{z}{z_\lambda}\right)^j}$$

z_0 is singularity of $f(z)$

(point where complex derivative)

approximate $f(z)$

for z near z_0

$$= \left[N(z_0) \pm O\left(\left(1 - \frac{z}{z_0}\right)^1\right)\right] \cdot \left(\frac{c_{0,\mu}}{\left(1 - \frac{z}{z_0}\right)^\mu} \pm O\left(\left(1 - \frac{z}{z_0}\right)^{-\mu+1}\right) \pm O(1) \right)$$

$(z \to z_0)$

$$= N(z_0)\, c_{0,\mu} \cdot \frac{1}{\left(1 - \frac{z}{z_0}\right)^\mu} \pm O\left(\left(1 - \frac{z}{z_0}\right)^{-\mu+1}\right) \quad (z \to z_0)$$

Taylor Expansion of f around z_0

analytic

$$f(z) = \sum_{i=0}^{\infty} \frac{f^{(i)}(z_0)}{i!} (z - z_0)^i$$

$$= f(z_0) + f'(z_0)(z - z_0) \pm O((z - z_0)^2) \quad (z \to z_0)$$

$$\left(1 - \frac{z}{z_0}\right)^a = o\left(\left(1 - \frac{z}{z_0}\right)^b\right) \quad a > b$$

$$\left| \frac{\left(1 - \frac{z}{z_0}\right)^a}{\left(1 - \frac{z}{z_0}\right)^b} = \left(1 - \frac{z}{z_0}\right)^{\overbrace{a-b}^{>0}} \underset{z \to z_0}{=} 0 \right.$$

$$|z - z_0|^a = \left|\frac{z}{z_0} - 1\right|^a z_0^{\,a} = z_0^{\,a} \left|1 - \frac{z}{z_0}\right|^a$$

$$= \Theta\left(\left(1 - \frac{z}{z_0}\right)^a\right)$$

Theorem 3.46 (Transfer-Theorem of Singularity Analysis)

Assume $f(z)$ is Δ-analytic and admits the singular expansion

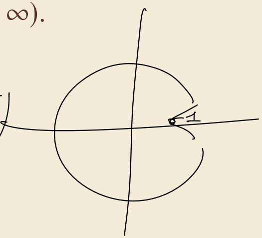
$$f(z) = g(z) \pm \mathcal{O}\big((1-z)^{-\alpha}\big) \qquad (z \to \underset{z_0}{1})$$

with $\alpha \in \mathbb{R}$. Then

f_n if $f(z) = \sum_n f_n z^n$

$$\underset{\displaystyle f_n \,/\!/}{[z^n]f(z)} = [z^n]g(z) \pm \mathcal{O}\big(n^{\alpha-1}\big) \qquad (n \to \infty).$$

$$f(z) = \underbrace{\underbrace{N(z_0)\cdot C_{0,\mu}}_{\text{constant}} \frac{1}{\left(1-\frac{z}{z_0}\right)^{\mu}}}_{g(z)} \pm O\left(\left(1-\frac{z}{z_0}\right)^{-\mu+1}\right)$$



$$[z^n]f(z) = N(z_0)\cdot C_{0,\mu}\left([z^n]\frac{1}{\left(1-\frac{z}{z_0}\right)^{\mu}}\right) \pm O\left(z_0^{-n} n^{\mu-1}\right)$$

$$\binom{n+\mu-1}{n} z_0^{-n} = \Theta\left(z_0^{-n} n^{\mu-1}\right)$$

$$\boxed{\begin{aligned} f(z) &= \sum_n f_n z^n \\ f(z z_0) &= \sum_n \underbrace{f_n z_0^n} z^n \end{aligned}}$$

$$(1-z)^{-\mu} = \sum_{n\geq 0} \binom{-\mu}{n} (-z)^n (1)^{-\mu-n} \qquad (a+b)^m = \sum_{n=0}^{m} \binom{n}{m} a^n b^{m-n}$$

$$\binom{-\mu}{n} = \frac{(-\mu)^{\underline{n}}}{n!}$$

$$[z^n] \qquad = \binom{-\mu}{n}(-1)^n \qquad \binom{r}{k} = \binom{k-r-1}{k}(-1)^k \quad k \in \mathbb{N}_0$$

$$= \binom{n+\mu-1}{n} \qquad \binom{n+\mu-1}{n} = \binom{\cancel{n-n}-\mu\cancel{+1-1}}{n}(-1)^n$$