search for j step to put the first jar

breaks: $MinT(j-1, k-1)$

not breaks: $MinT(n-j, k)$

max (adversary) worst case

want j with $Minim$ (overall)

with j $\left\{ \begin{array}{c} max \\ break/not \end{array} \right\}$ - - - $\left. \begin{array}{c} \\ \end{array} \right\} \right\}$

$q \geqslant \log n$ or   NOT POSSIBLE $\bigg| \begin{array}{l} k \leq \log n \\ \text{or} \\ \text{Binary Search} \end{array}$

$n \leq 2^q$

$q = T(n, k) =$ min # trials to search (worst case) ladder size n
with k jars

$n = R(k, q) =$ max ladder size that can be searched (worst case)
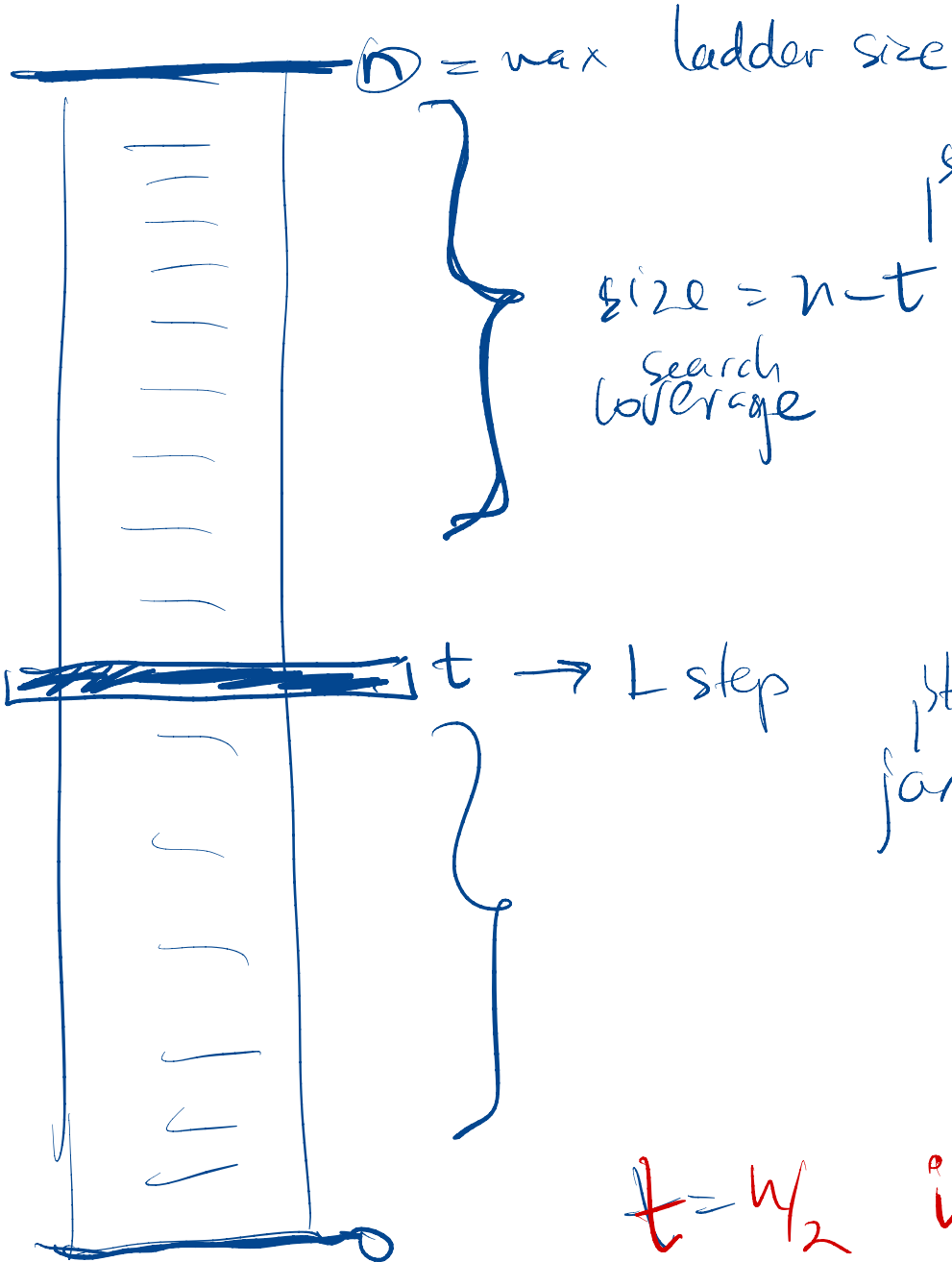with k jars and q trials

---

$$R(k, q) = \boxed{k \geq q} \quad n = 2^q \quad \text{TRIVIAL}$$

Binary Search

$$k < q$$

- $k-1$ jars Bin Search linear $\boxed{2^{k-1}} \; (n = ?)$
- down to 1 jar $\Rightarrow$ remain trials <u>to be enough</u>

if all break $\boxed{q - (k-1)}$ linear coverage
jars

NOT OPTIMAL

$n = R(k, q)$

n = max ladder size

$n_1 t \in OPTSOL$

1st jar does not break

$size = n - t$

search coverage

$R(k, q-1)$

1st jar spot (OPTSOL)

$t \rightarrow L$ step

1st jar breaks

$R(k-1, q-1)$

$t = \frac{n}{2}$ if $k \geq \log n$

$k \geq q$

$n = 2^q$

For $n=1$, w_max
for $k=1$: k_max

$\underset{t}{\text{Min}}$ $\underset{\text{max}}{\Bigg\{}$ $\begin{cases} T(t-1, k-1) \\ T(n-t, k) \end{cases}$

$\Theta(n)$

$\Theta(n^2 k)$

Increasing with t

It would order
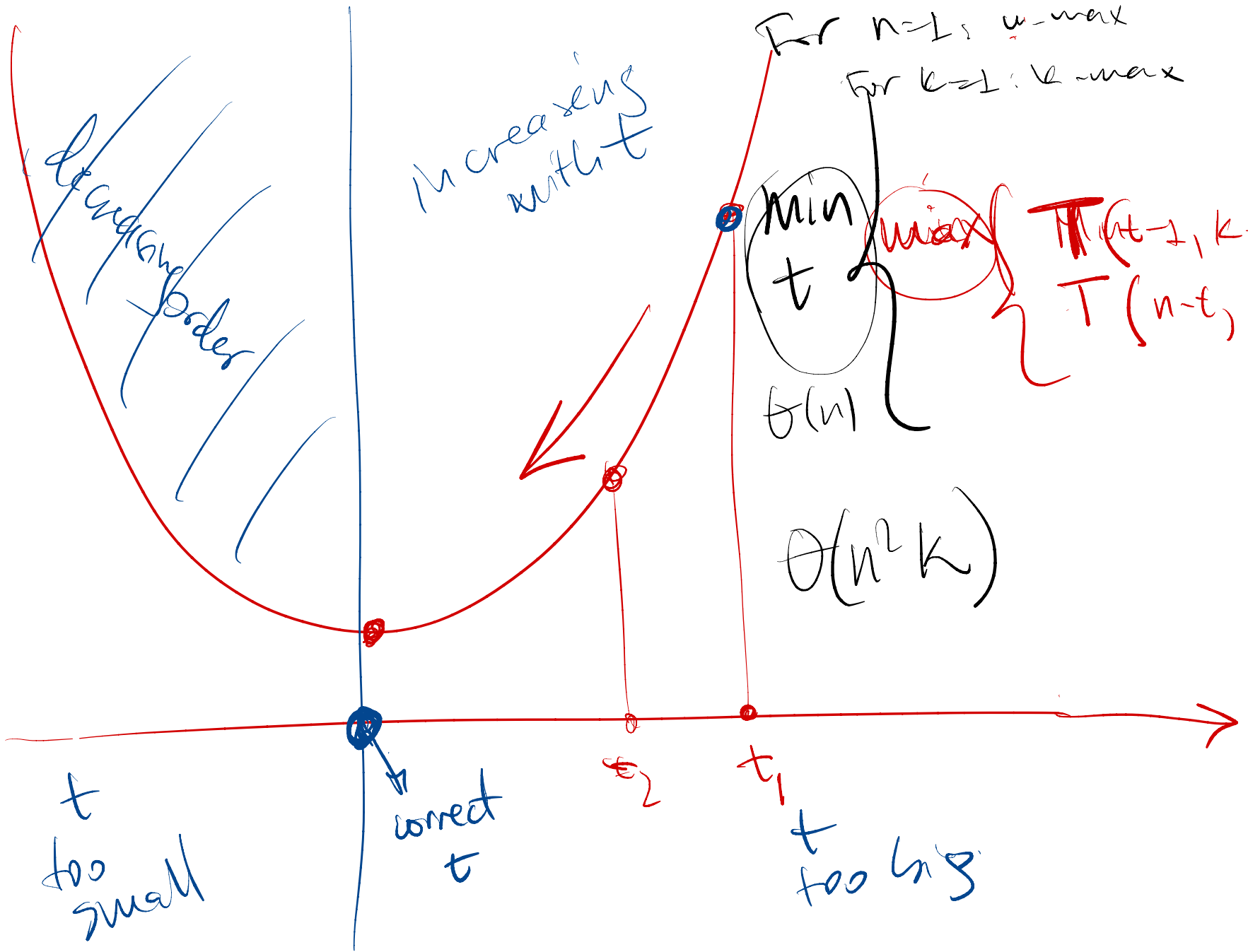
t too small

correct t

t_2

t_1

t too big
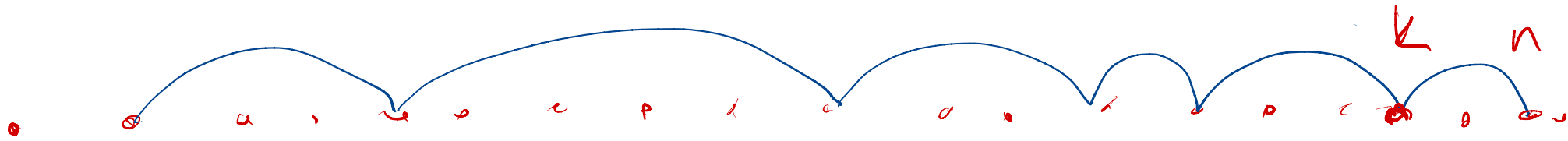
15.4-5

$c[u] =$ longest increase subseq ending in n

$=$ search for k $\}$ previous element $A[k] \le A[n]$

$$c[u] = \underbrace{\boxed{c[k]} + 1}_{max}$$

linear both
15.4-5
15.4-6

for $i = 1$ in

search for k - - -

to compute $c[n]$

$S[u] = k$ previous element in longest list ending in n

$\rightarrow$ 15.4-5 linear search $\Theta(n)$

15.4-6 binary search $\Theta(\log n)$

G line $\boxed{l_i \_ l_{i+1} \_ l_{i+2} \_ \qquad\qquad\qquad \_ l_j}$

words chunk   characters-sm $L_{ij} = \sum\limits_{t=i}^{j} \ell_t$

$i-j$ penalty G $\underbrace{(M - L_{ij} - j + i)}^{3}$

(#spaces)

current last word →

allow 1 space between words.

$C[j] =$

$= \min \left\{ \begin{array}{l} \sum \\ G \end{array} \right.$ penalty$_G$ $\rightarrow$ $\boxed{\text{extras}(i,j)}$

except last

$= \min \left\{ C[i-1] + \text{penalty}(i,j) \right.$ $\boxed{\text{penalty}}_{(i,j)} = \left\{ \begin{array}{ll} \infty & \text{if extras}(i,j) < 0 \\ & \text{(words don't fit on line)} \\ 0 & \text{last line} \\ \text{extras}(i,j)^3 & \text{other lines.} \end{array} \right.$

break ↙

first word on line