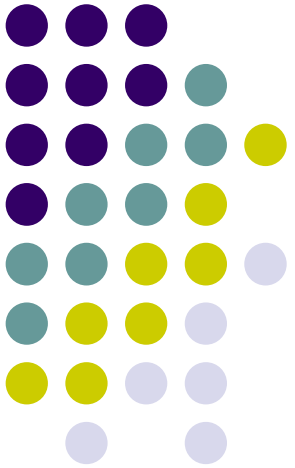
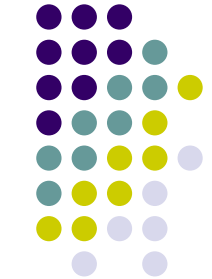
Composite Pattern Discovery for PCR Application

Stanislav Angelov
University of Pennsylvania, USA

Shunsuke Inenaga
Kyushu University, Japan
Japan Society for the Promotion of Science

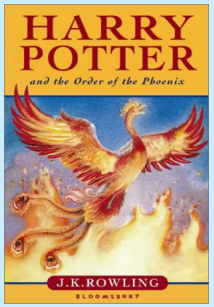


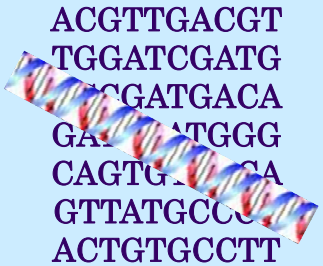
Pattern Discovery

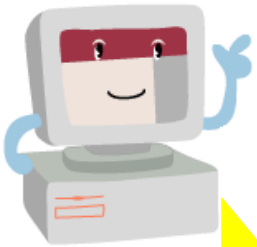


Input

text data







Output

pattern

knowledge

rule

Finding Missing Patterns

Input : text ***T*** and threshold α

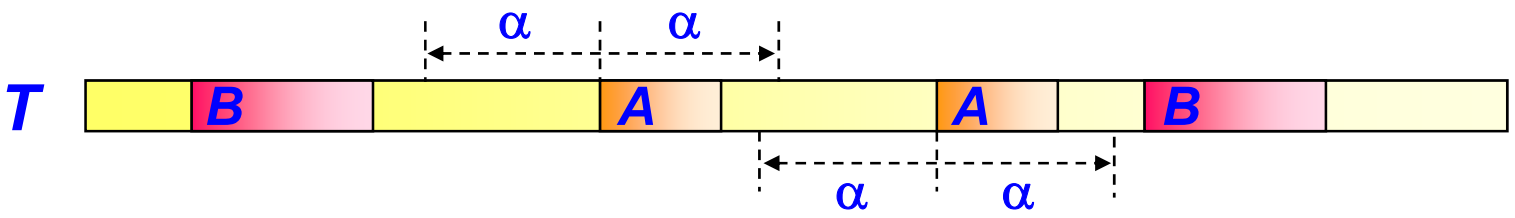
Output : Pattern pair (***A***,***B***) satisfying:

1. The distance between any occurrences of ***A*** and ***B*** in ***T*** is at least α,
2. |***A***| = |***B***|, and
3. |***A***| (=|***B***|) is shortest possible.

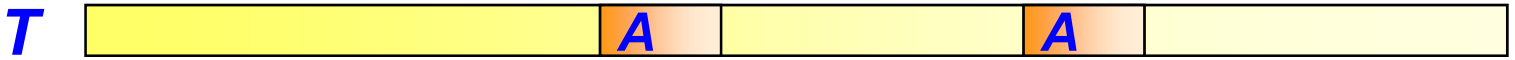
Finding Missing Patterns [cont.]

If *A* and *B* are non α-close,
(*A*,*B*) is said to be a **missing pair**.

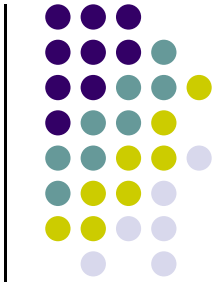
Case 2: non α-close



Case 3: non α-close



Application - PCR



PCR (Polymerase Chain Reaction)

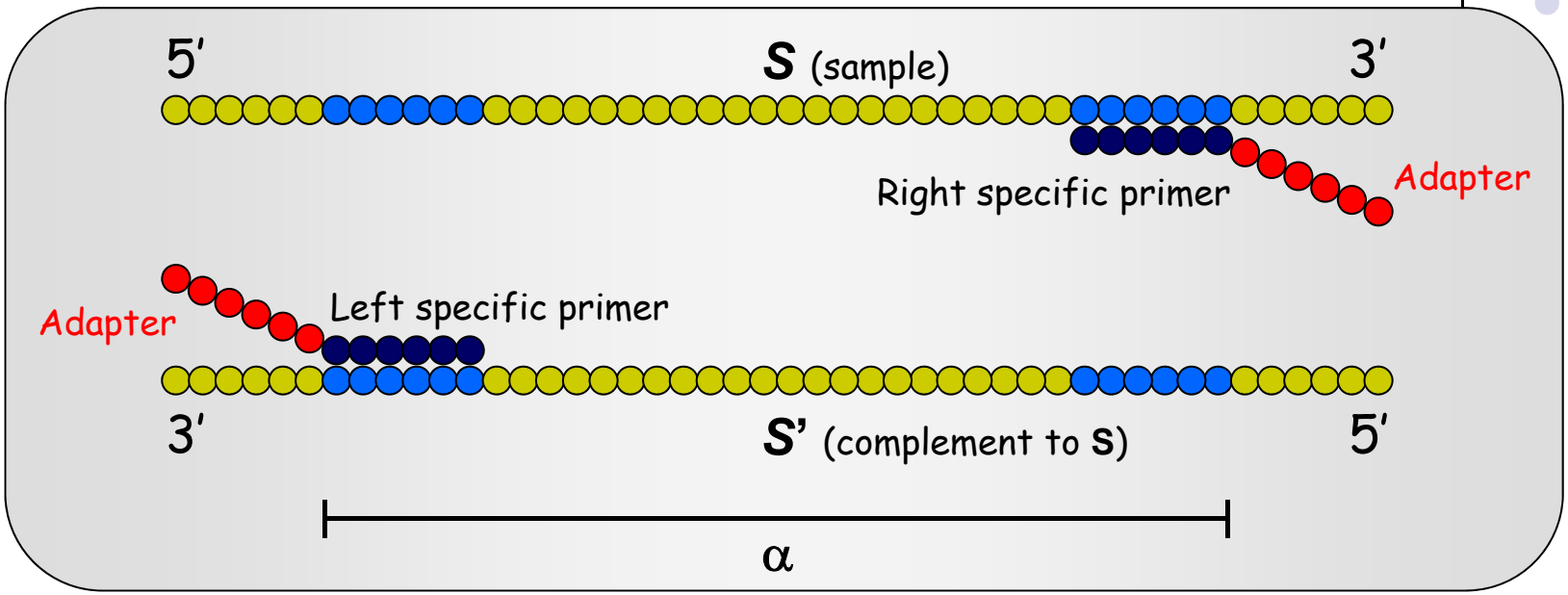
- Standard technique to produce many copies of a region of DNA (can be a tiny sample).
- In Medicine, to detect infections.
- In Forensic Science, to identify individuals.

Application – PCR [cont.]

Nested PCR

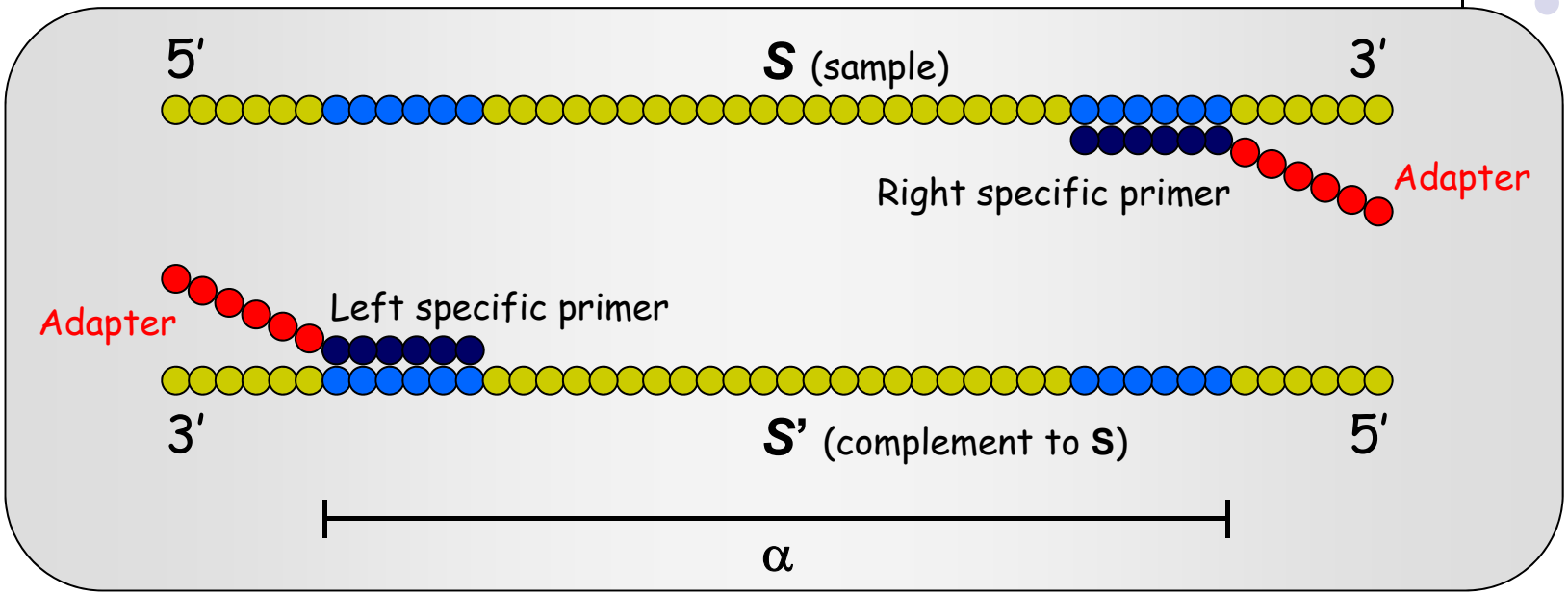
- Repeated PCR with nested primers
- Achieving ultra-sensitive detection
- Good adapter primers for nested PCR:
  bind only to the adapters, and
  amplify nothing directly from the samples!

Application – PCR [cont.]



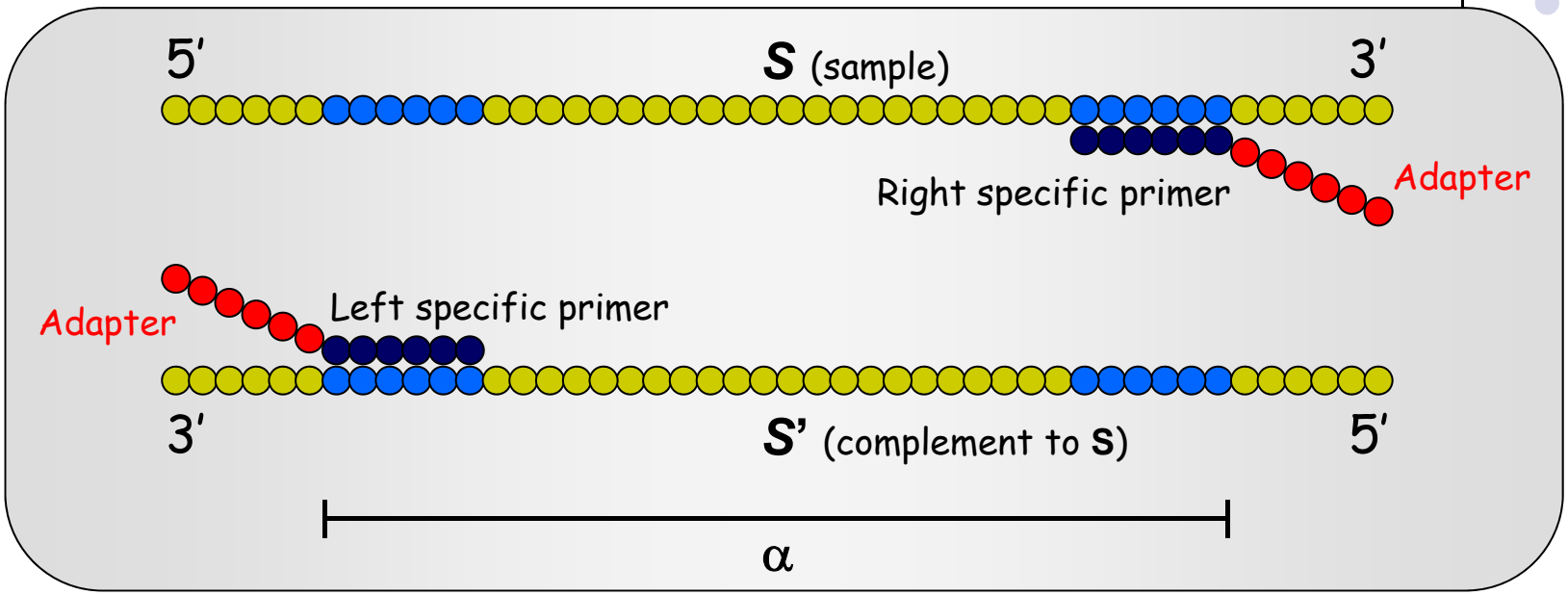
- We want a pair of good **adapter primers** which amplify nothing directly from ***S*** or ***S'***.
  (Adapter primers are complements to adapters.)

Application – PCR [cont.]



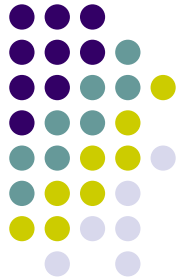
- If (***A***,***B***) is a missing pair in ***S*** and ***S'***, then (***A'***,***B***) is **not** a pair of binding sites for any region of length less than α.

Application – PCR [cont.]



- So (***A'***,***B***) satisfies a necessary condition of being a good adapter primer pair!!

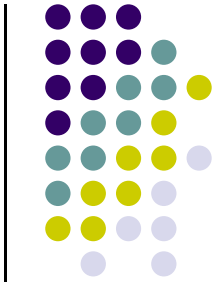
Previous Work



- Inenaga, Kivioja and Makinen. [WABI'04] proposed a bit-table based algorithm to find a missing pattern pair of the same length.

- We also gave a suffix tree based algorithm to solve a generalized problem where the patterns in the pair can be of different length.

Complexity Comparisons



Finding missing pattern pair of same length

	time	space
our algorithm	$O(\alpha n \log\log_\sigma n)$	$O(n)$
bit-table algorithm of inenaga et al. [WABI'04]	$O(\alpha n(\sigma+\log\log_\sigma n))$	$O(\alpha n)$

- σ is the alphabet size.
- α is typically 5000 (due to PCR application)!

Complexity Comparisons [cont.]

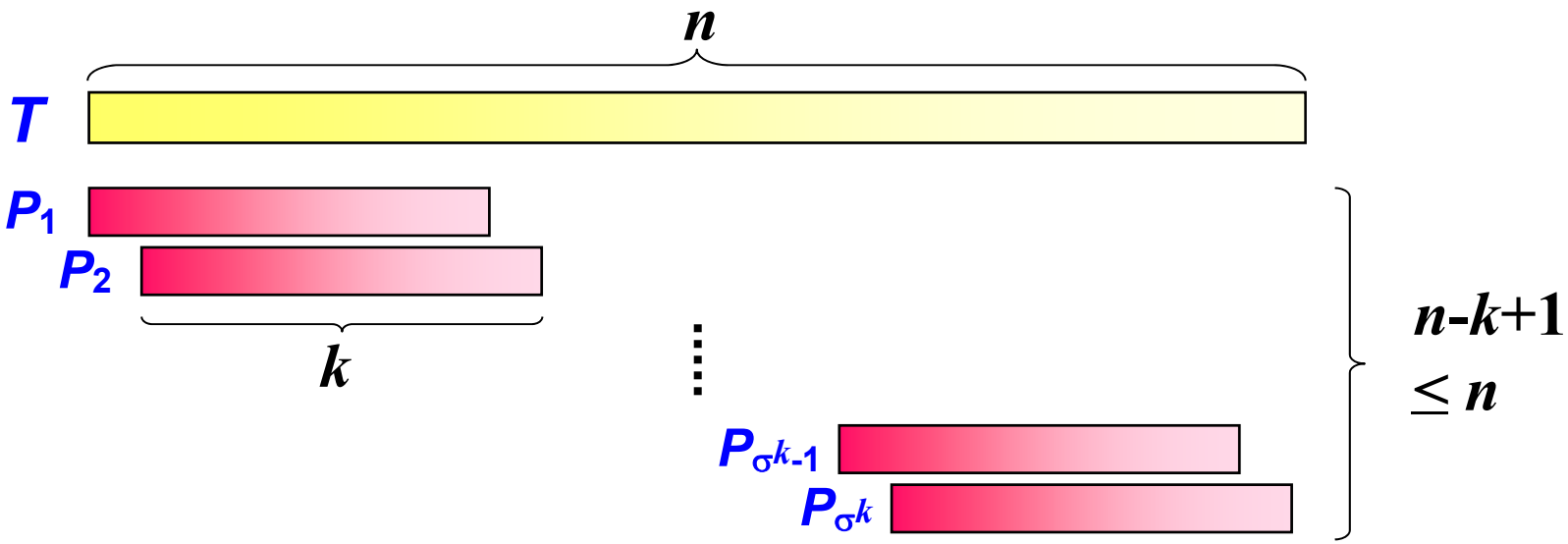
Finding missing pattern pair of different length

	time	space
our algorithm	$O(\alpha n \log_\sigma n)$	$O(n)$
suffix tree algorithm A of Inenaga et al. [WABI'04]	$O(n^2)$	$O(n)$
suffix tree algorithm B of Inenaga et al. [WABI'04]	$O(\alpha n \log_\sigma n)$	$O(n \log_\sigma n)$

- Our algorithm does not need a suffix tree – not only faster but also simpler.

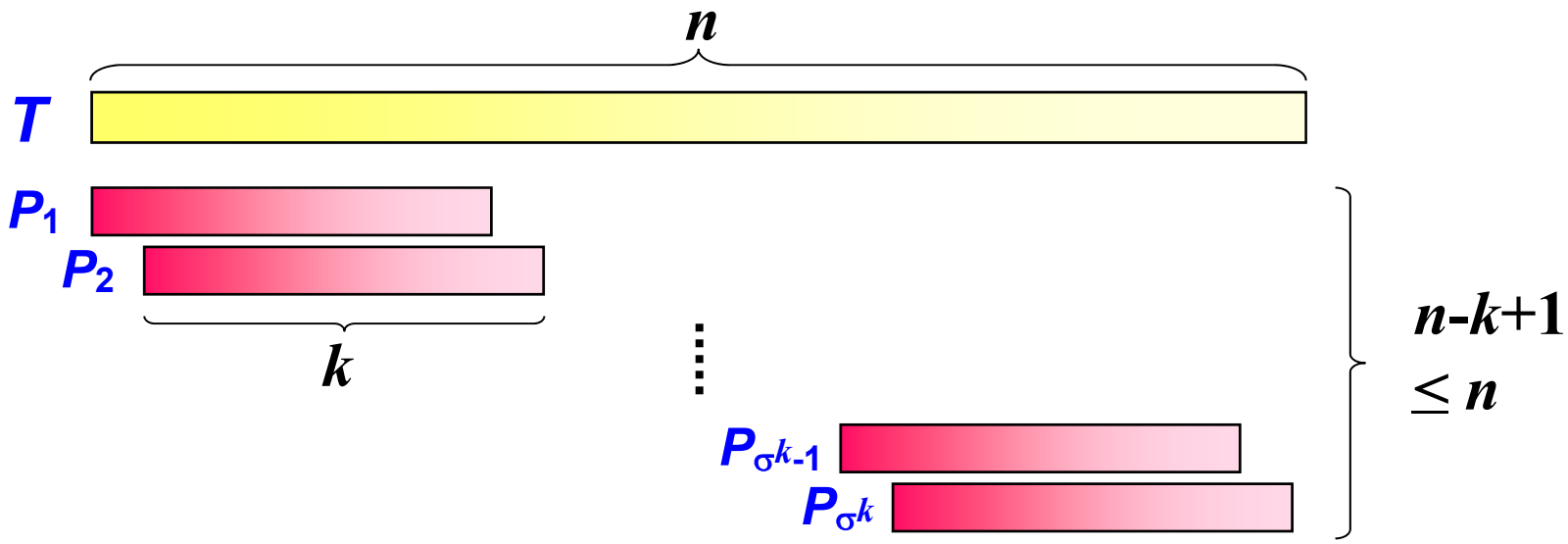
Single Missing Pattern

- We start with finding a single missing pattern.
- KEY: There are at most σ^k patterns of length k.



Single Missing Pattern [cont.]

- We have $k \leq \lfloor \log_\sigma n \rfloor$.
- If k is the largest integer for which all σ^k patterns of length k exist in T, then there is a missing pattern of length $\lceil \log_\sigma n \rceil$.

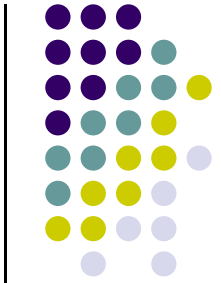


Single Missing Pattern [cont.]

- Compute a bit table of all patterns of length $\lfloor \log_\sigma n \rfloor$ using a bijective mapping f from patterns to integers. ($O(n)$ time, using e.g. Karp & Rabin algo.)

1) there exists a missing pattern of length $\lfloor \log_\sigma n \rfloor$
   ⇒ output it.
2) otherwise (all patterns of length $\lfloor \log_\sigma n \rfloor$ are present in T)
   ⇒ there is a missing pattern of length $\lceil \log_\sigma n \rceil$
   ⇒ compute and output it.

Missing Pair of Fixed Length

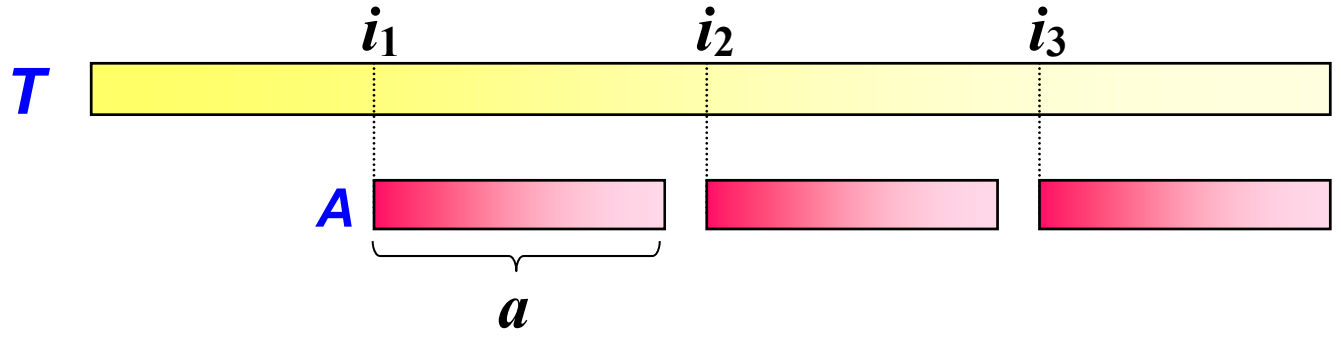


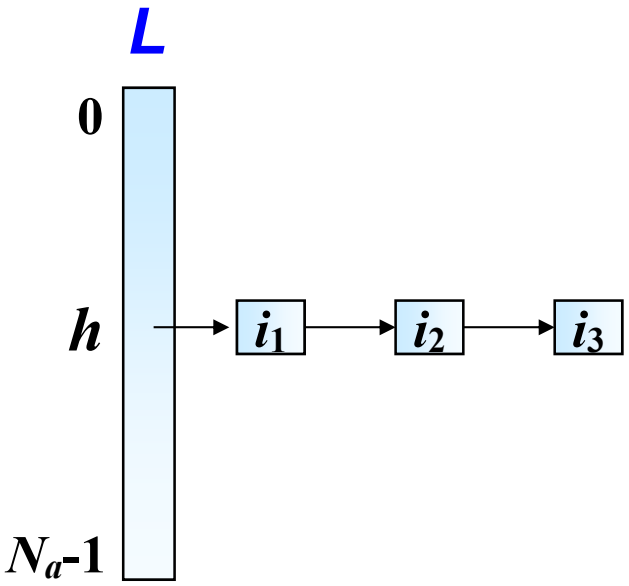
- Input: text T, threshold α,
  pattern lengths a and b

- Output: missing pattern pair (A, B)
  such that $|A| = a$ and $|B| = b$

  - Assume w.l.o.g. $a \geq b$.
  - We consider the case $a < m$, where m is the length of the shortest single missing pattern P in T.
    Or else P can be paired with any pattern of length b.
  - Let $N_a = \sigma^a$ and $N_b = \sigma^b$ (Note $n > N_a \geq N_b$).

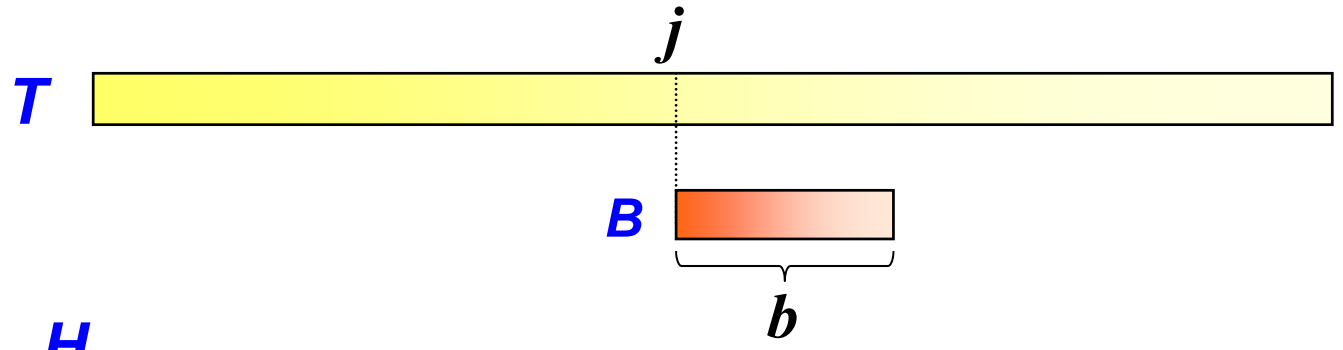
Missing Pair of Fixed Length [cont.]

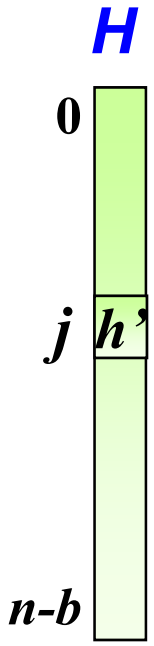




- Let $f(A) = h$.
- L : array of size N_a, where $L[h]$ is the list of occurrences of A in T.

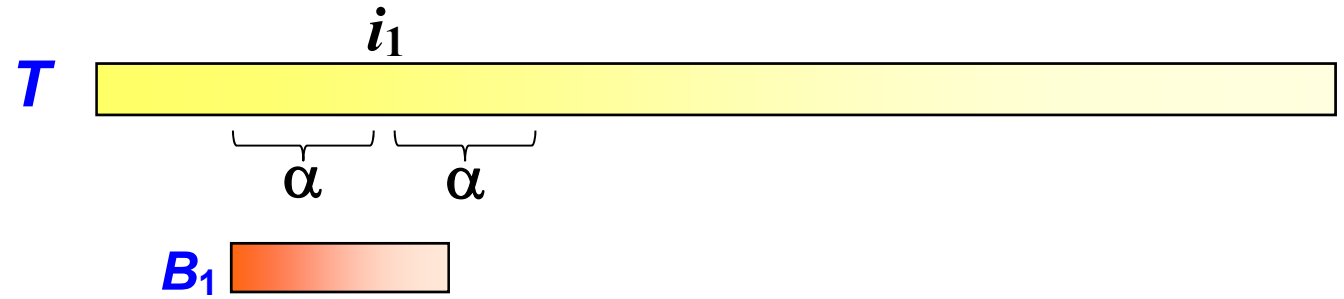
Missing Pair of Fixed Length [cont.]

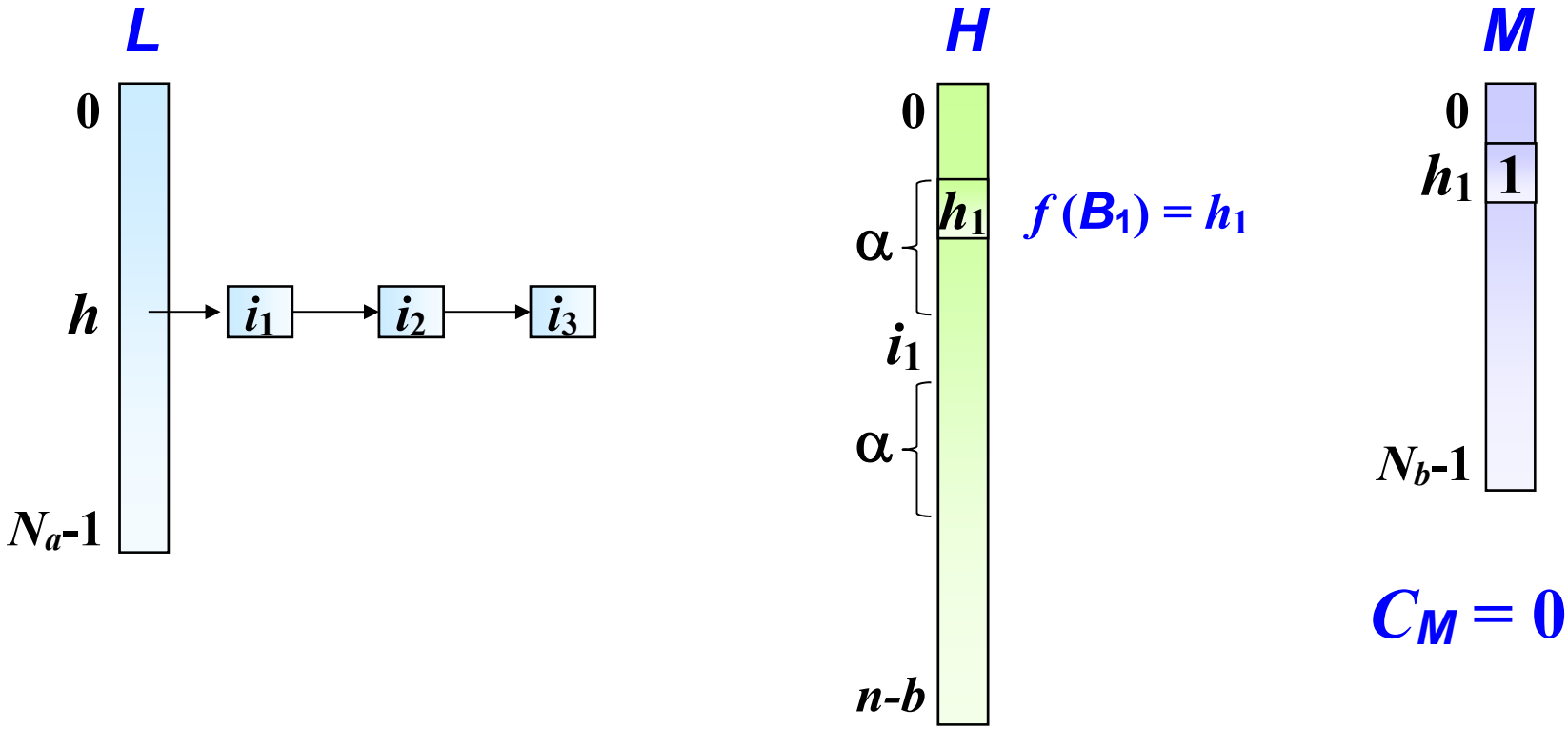




- Let $f(B) = h'$.
- H : array of size $n-b+1$, where $H[j] = h'$.

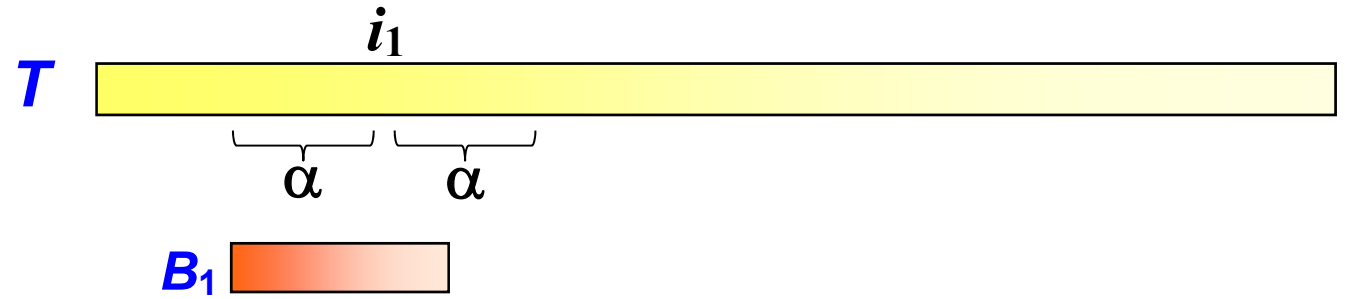
Missing Pair of Fixed Length [cont.]

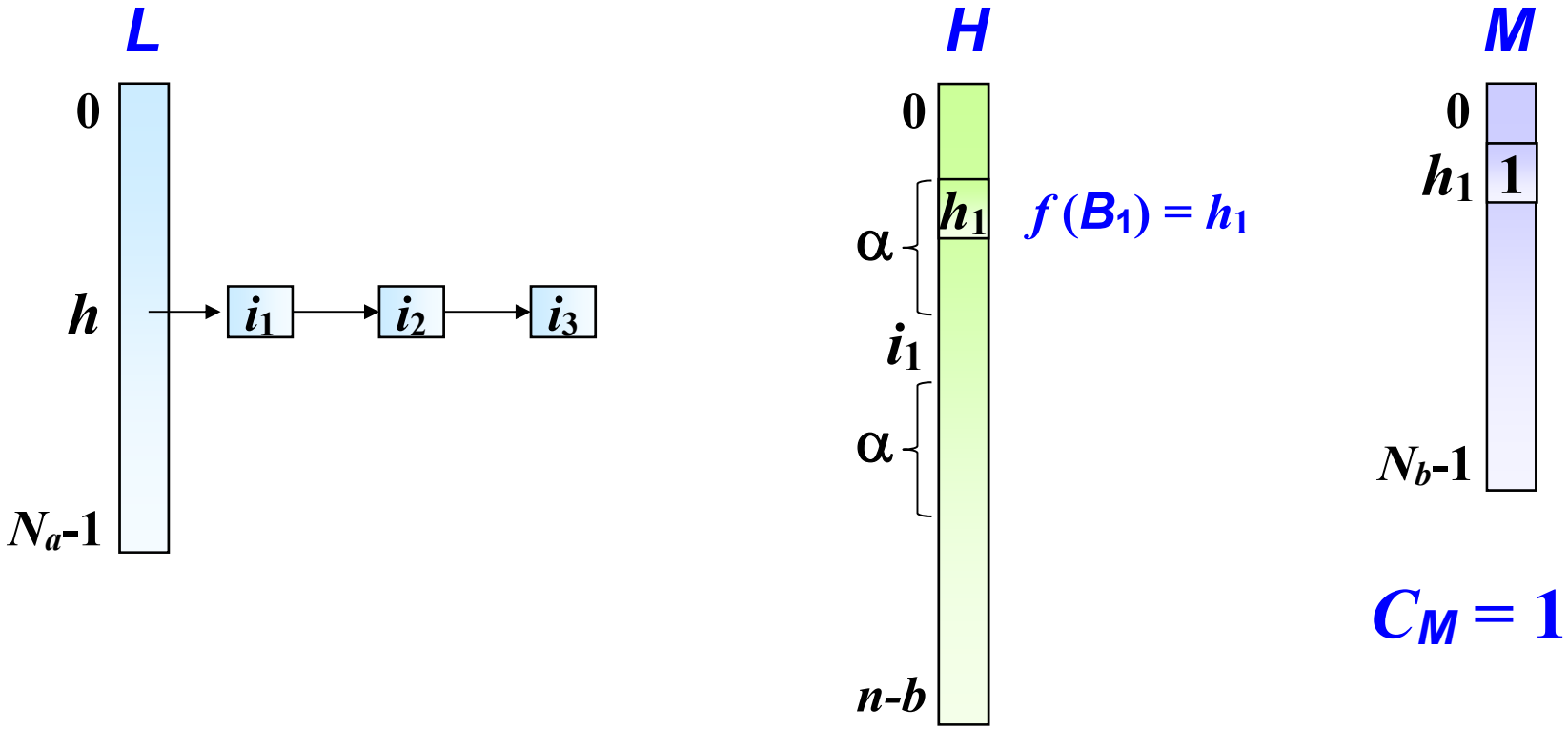




$C_M = 0$

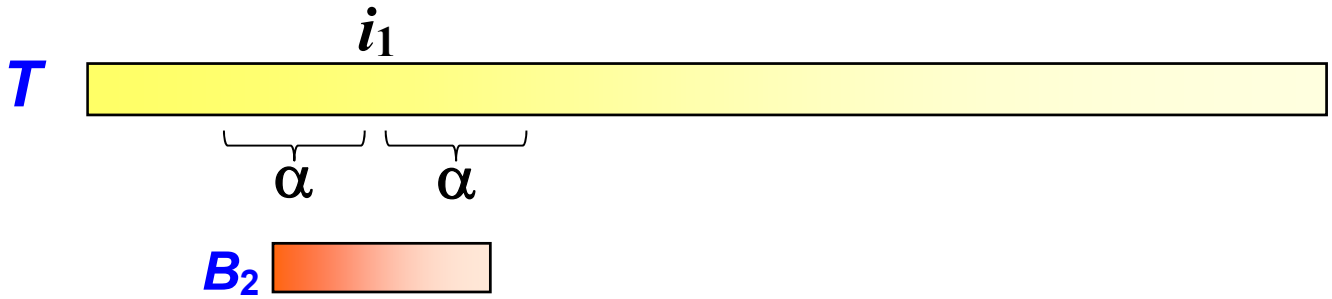
Missing Pair of Fixed Length [cont.]

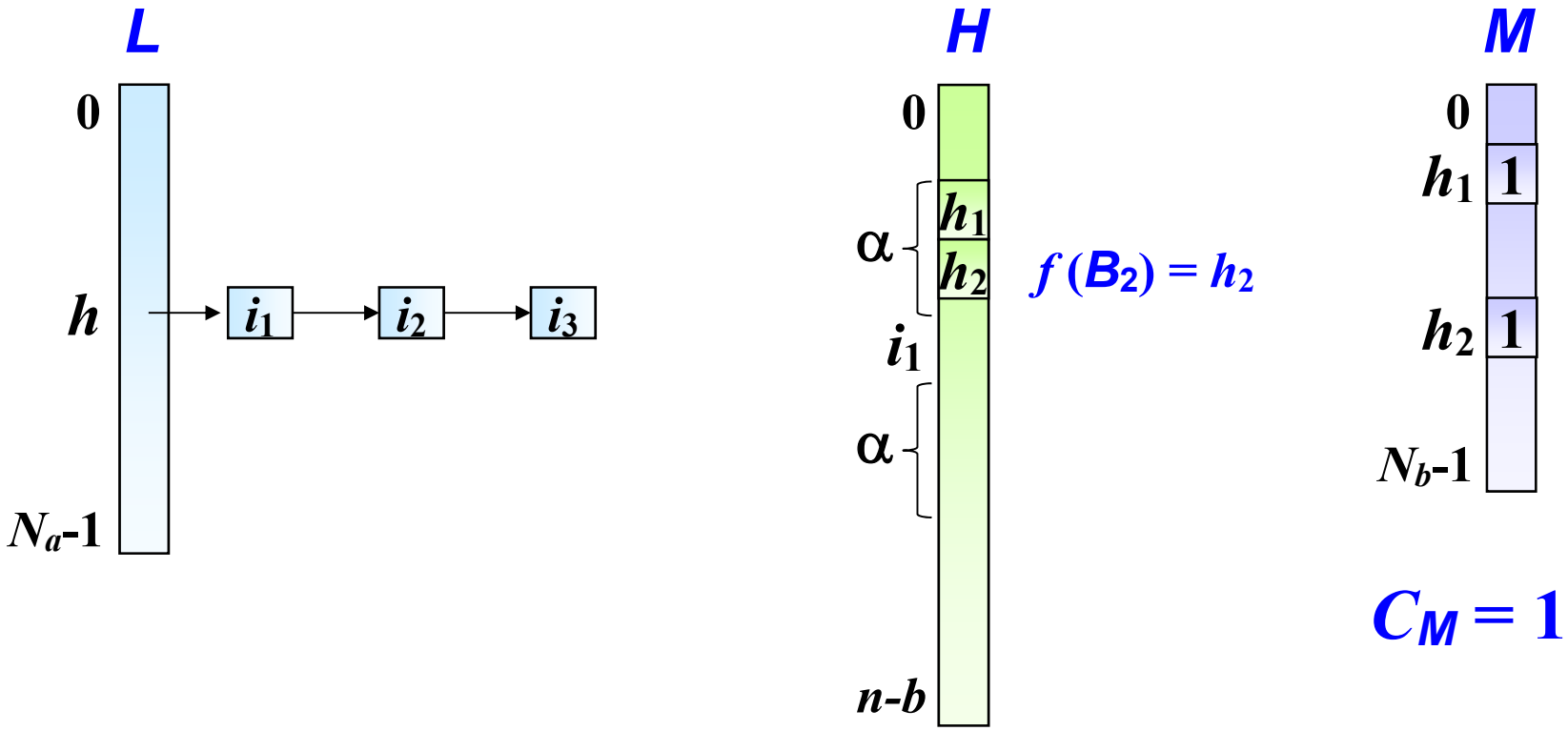




$C_M = 1$

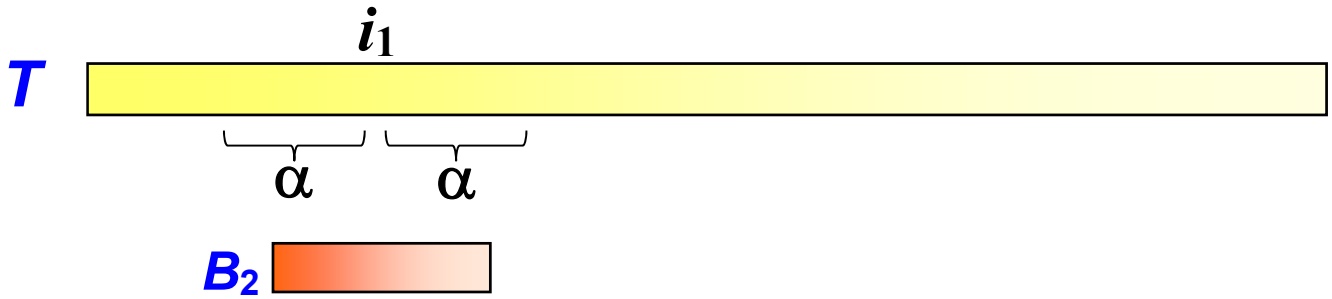
Missing Pair of Fixed Length [cont.]

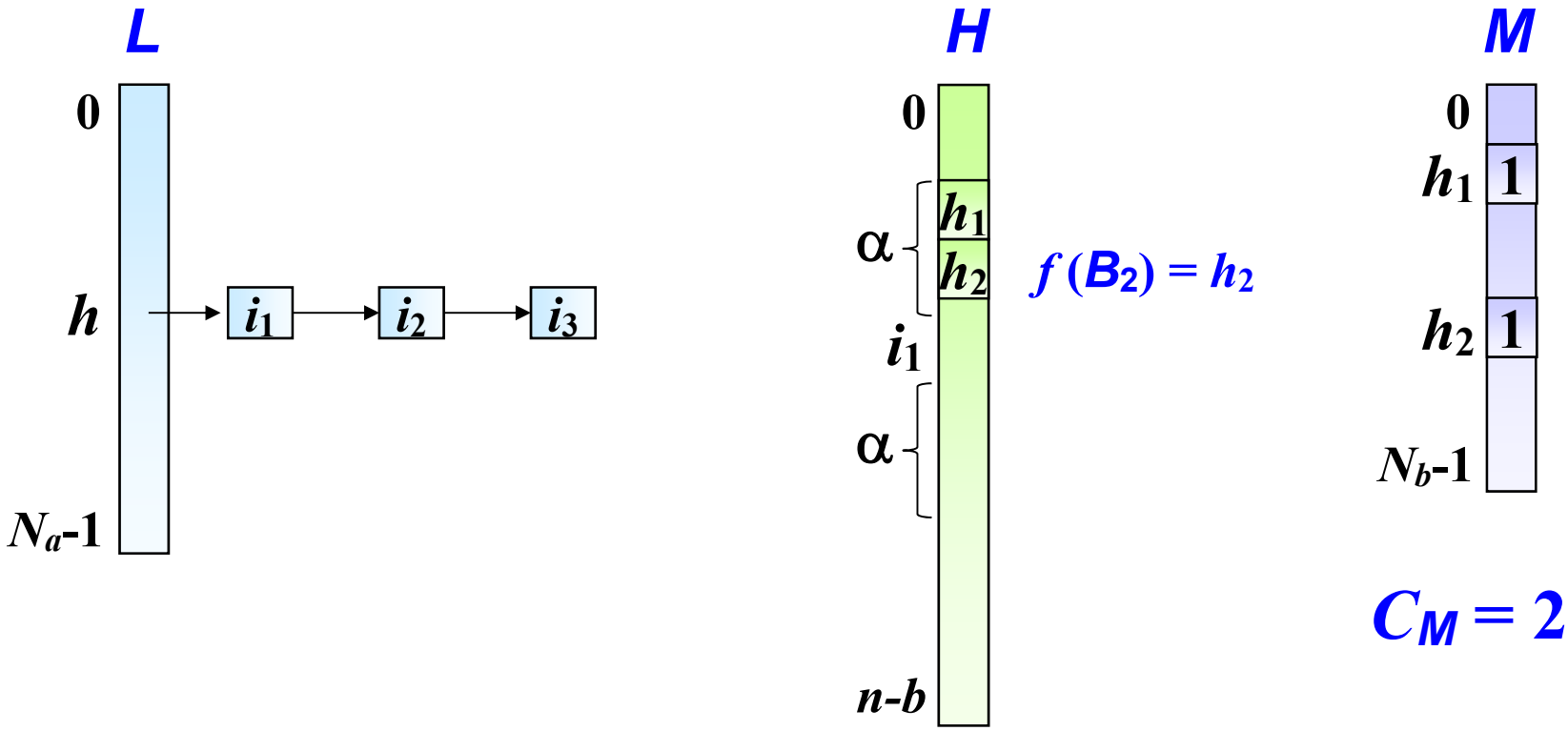




$C_M = 1$

Missing Pair of Fixed Length [cont.]





$C_M = 2$

Missing Pair of Fixed Length [cont.]

- The iteration ends
  - when $C_M = N_b$.
    This case, all patterns of length b are α-close to A.
  - or when all positions in $L[h]$ are processed.
    This case, scan M and find a missing pattern of length b. The algorithm outputs the missing pair.

- The algorithm runs in total of $O(\alpha n)$ time and $O(n)$ space.

Missing Pair of Same Length [cont.]

- Monotonicity property: If (A, B) is a missing pair, for any superstrings C, D of A, B resp., (C, D) is also a missing pair.

- By monotonicity property we can do a binary search on the length $1 \ldots \lceil \log_\sigma n \rceil$ of the patterns using the aforementioned algorithm, and find the shortest missing pair of same length.

  It takes $O(\alpha n \log\log_\sigma n)$ time and $O(n)$ space.

Missing Pair of Different Length

- It is not hard to extend the algorithm to the case where ***A*** and ***B*** do not necessarily have the same length.

- We can find such a missing pair in $O(\alpha n \log_\sigma n)$ time and $O(n)$ space.

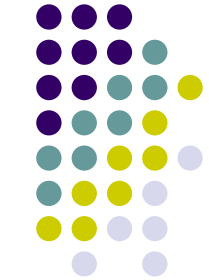
Experiments

- Linux on 1GHz CPU with 2GB RAM.
- In Java. http://www.cis.upenn.edu/~angelov

- Human genome (2.5GB) from ftp://ftp.ensembl.org/pub/current_human/

- $\alpha = 5000$.

Experiments [cont.]



- We found 238 pairs of missing patterns of length 8 for the human genome.
  - For the Baker's yeast genome, the patterns in the shortest missing pairs are also of length 8 ! [Inenaga et al. WABI'04]

- There are common missing pairs of patterns of length 8 for the human and yeast genomes.

Experiments [cont.]

Missing pattern pairs of length 8 for both the human and the yeast gemones. The reverse complements are also missing

missing pair	yeast α_{AB}	human α_{AB}
(AATCGACG,CGATCGGT)	5008	6458
(CCGATCGG,CCGTACGG)	5658	6839
(CGACCGTA,TACGGTCG)	13933	7585
(CGACCGTA,TCGCGTAC)	5494	5345
(CGAGTACG,GTCGATCG)	5903	8090
(CGATCGGA,GCGCGATA)	6432	6619

Conclusions

- We solved the missing pattern pair problem in $O(\alpha n \log\log_\sigma n)$ time for the same length case, and $O(\alpha n \log_\sigma n)$ time for the different length case. Both in $O(n)$ space.

- We also developed an alternative algorithm to solve this problem, and moreover solved extended problems (see the proceedings).