Approximate Cartesian Tree Matching: an Approach Using Swaps

Bastien Auvray, Julien David, Richard Groult, Thierry Lecroq

SeqBIM 2023, Lille, France, November 20th, 2023

Foreword

- It's about pattern matching in time series.
- Motivations:
  share prices in stock markets, temperatures, notes in music, GST data in bioinformatics...
- In recent years, new pattern matching problems such as Order-Preserving Matching and Cartesian Tree Matching have been introduced.
- To the best of our knowledge, no approximate pattern matching problem existed in the Cartesian tree framework.

Outline

Preliminaries

Prerequisites

- Sequences of integers
- A total order $<$
- All elements of x are distinct and numbered from 1 to n (the length of x)

Cartesian tree matching (1)

Cartesian tree [Vuillemin, 1980]

A sequence x of length n can be associated to its Cartesian tree $C(x)$ according to the following rules:

- if x is empty, then $C(x)$ is the empty tree;
- if $x[1 \dots n]$ is not empty and $x[i]$ is the smallest value of x, $C(x)$ is the Cartesian tree with:
  - i as its root,
  - $C(x[1 \dots i-1])$ as the left subtree,
  - $C(x[i+1 \dots n])$ as the right subtree.

Cartesian tree matching (1)

Cartesian tree [Vuillemin, 1980]

A sequence x of length n can be associated to its Cartesian tree $C(x)$ according to the following rules:

- if x is empty, then $C(x)$ is the empty tree;
- if $x[1 \ldots n]$ is not empty and $x[i]$ is the smallest value of x, $C(x)$ is the Cartesian tree with:
  - i as its root,
  - $C(x[1 \ldots i-1])$ as the left subtree,
  - $C(x[i+1 \ldots n])$ as the right subtree.

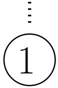
NB: In our examples, we will label the nodes with the values instead of the indices

Cartesian tree matching (1)

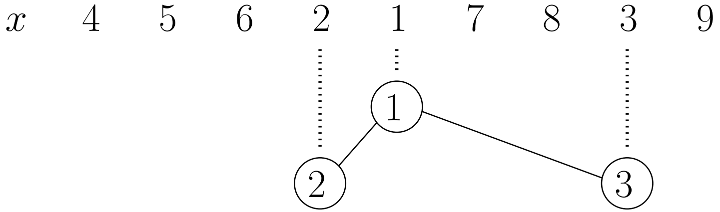
x	4	5	6	2	1	7	8	3	9

Cartesian tree matching (1)

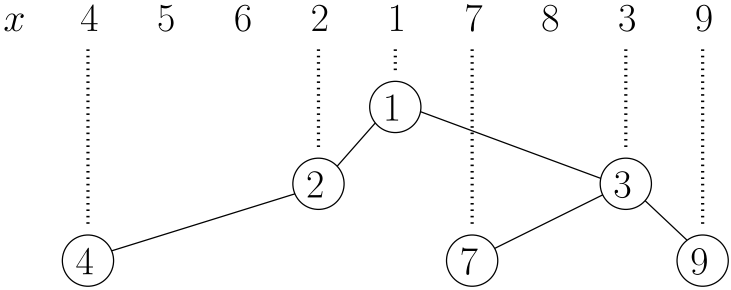
x	4	5	6	2	1	7	8	3	9



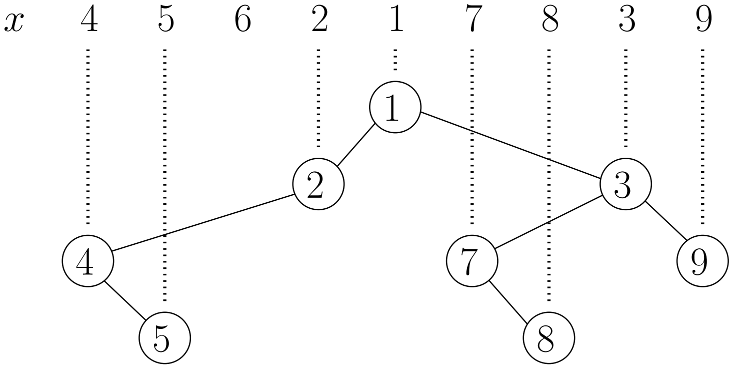
Cartesian tree matching (1)



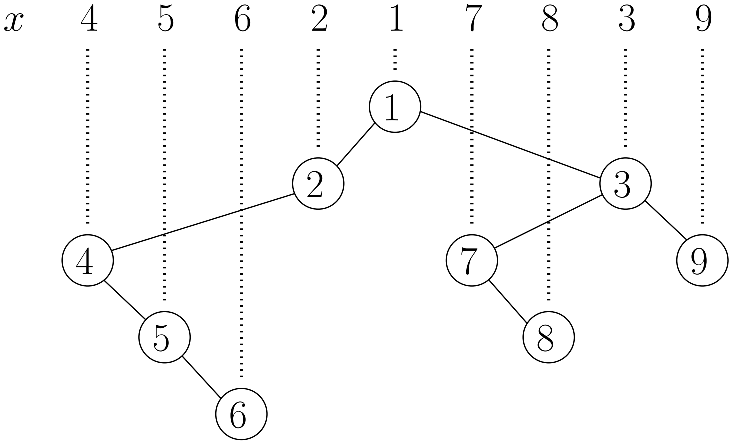
Cartesian tree matching (1)



Cartesian tree matching (1)



Cartesian tree matching (1)



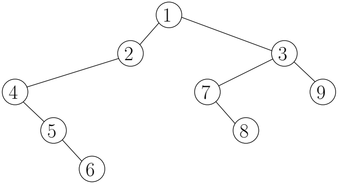
Cartesian tree matching (2)

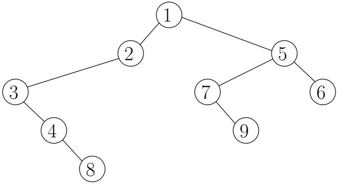
Similarity

Two sequences x and y are similar if they share the same Cartesian tree, and we note $x \approx_{CT} y$.

Cartesian tree matching (2)

$4 \quad 5 \quad 6 \quad 2 \quad 1 \quad 7 \quad 8 \quad 3 \quad 9 \quad \approx_{CT} \quad 3 \quad 4 \quad 8 \quad 2 \quad 1 \quad 7 \quad 9 \quad 5 \quad 6$





Cartesian tree matching (2)

Cartesian tree matching [Park, Amir, Landau and Park, 2019]

The Cartesian tree matching (CTM) problem is the following: Given a pattern p and a text t, find every factor f of t such that $f \approx_{CT} p$.

Cartesian tree matching (3)

Parent-distance [PALP19]

Given a sequence $x[1 \dots n]$, the parent-distance representation of x is an integer sequence $\overrightarrow{PD}_x[1 \dots n]$, which is defined as follows:

$$\overrightarrow{PD}_x[i] = \begin{cases} i - max_{1 \leq j < i}\{j \mid x[j] < x[i]\} & \text{if such } j \text{ exists} \\ 0 & \text{otherwise} \end{cases}$$

Cartesian tree matching (3)

x	1	4	7	9	5	8	6	2	3
$\overrightarrow{PD}_x$									

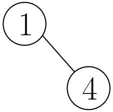
Cartesian tree matching (3)

x	1	4	7	9	5	8	6	2	3
$\overrightarrow{PD}_x$	0								

①

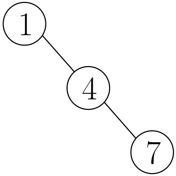
Cartesian tree matching (3)

x	1	4	7	9	5	8	6	2	3
$\overrightarrow{PD}_x$	0	1							



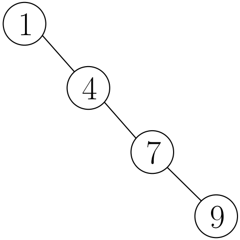
Cartesian tree matching (3)

x	1	4	7	9	5	8	6	2	3
$\overrightarrow{PD}_x$	0	1	1						



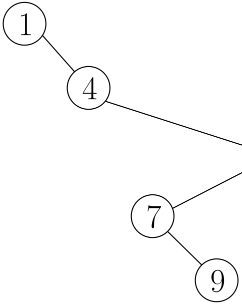
Cartesian tree matching (3)

x	1	4	7	9	5	8	6	2	3
$\overrightarrow{PD}_x$	0	1	1	1					



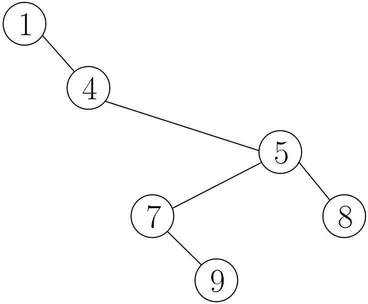
Cartesian tree matching (3)

x	1	4	7	9	5	8	6	2	3
$\overrightarrow{PD}_x$	0	1	1	1	3				



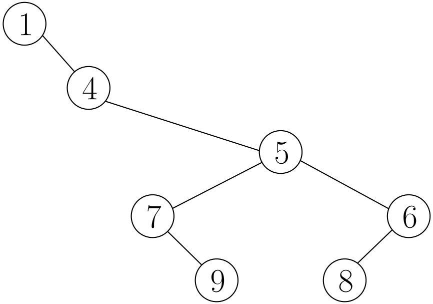
Cartesian tree matching (3)

x	1	4	7	9	5	8	6	2	3
$\overrightarrow{PD}_x$	0	1	1	1	3	1			



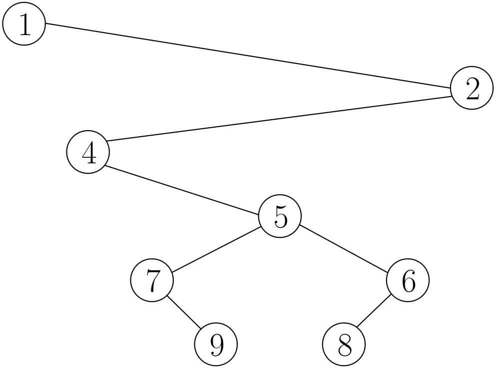
Cartesian tree matching (3)

x	1	4	7	9	5	8	6	2	3
$\overrightarrow{PD}_x$	0	1	1	1	3	1	2		



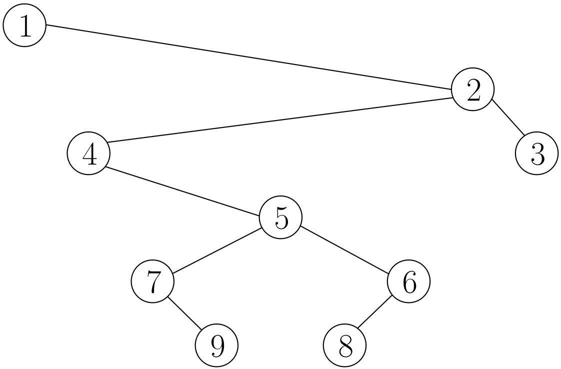
Cartesian tree matching (3)

x	1	4	7	9	5	8	6	2	3
$\overrightarrow{PD}_x$	0	1	1	1	3	1	2	7	



Cartesian tree matching (3)

x	1	4	7	9	5	8	6	2	3
$\overrightarrow{PD}_x$	0	1	1	1	3	1	2	7	1



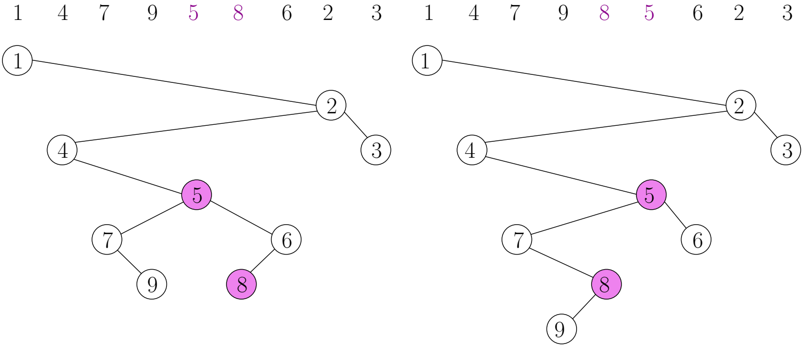
Approximate CTM (1)

Swap

Let x and y be two sequences of length n, and $i \in \{1, \ldots, n-1\}$, we denote $y = \tau(x, i)$ to describe a swap, that is:

$$y = \tau(x,i) \text{ if } \begin{cases} x[j] = y[j], \forall j \notin \{i, i+1\} \\ x[i] = y[i+1] \\ x[i+1] = y[i] \end{cases}$$

Approximate CTM (1)



Approximate CTM (2)

Approximate CTM

Let x and y be two sequences of length n, we have $x \overset{\tau}{\approx}_{CT} y$ if:

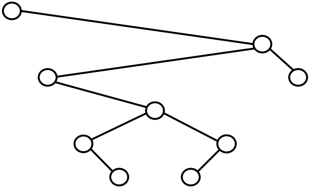
$$\begin{cases} x \approx_{CT} y \\ \text{or} \\ \exists\ x',\ y', \exists\ i \in \{1, \ldots, n-1\}, x' \approx_{CT} x, y' \approx_{CT} y, \\ x' = \tau(y', i) \text{ and } y' = \tau(x', i) \end{cases}$$

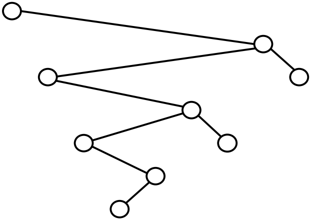
Approximate CTM (2)

$$1\ 4\ 7\ 9\ 5\ 8\ 6\ 2\ 3 \quad \overset{\tau}{\rightleftarrows} \quad 1\ 4\ 7\ 9\ 8\ 5\ 6\ 2\ 3$$

$$\approx_{CT} \qquad\qquad \approx_{CT}$$

$$1\ 3\ 7\ 9\ 5\ 8\ 6\ 2\ 4 \quad \overset{\tau}{\approx}_{CT} \quad 1\ 3\ 5\ 9\ 8\ 4\ 9\ 2\ 7$$





Approximate CTM (3)

Reverse parent-distance

Given a sequence $x[1 \dots n]$, the reverse parent-distance of x is an integer sequence $\overleftarrow{PD}_x[1 \dots n]$, which is defined as follows:

$$\overleftarrow{PD}_x[i] = \begin{cases} min_{i>j\geq n}\{j \mid x[i] > x[j]\} - i & \text{if such } j \text{ exists} \\ 0 & \text{otherwise} \end{cases}$$

Approximate CTM (3)

x	1	4	7	9	5	8	6	2	3
$\overleftarrow{PD}_x$									

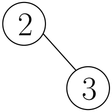
Approximate CTM (3)

x	1	4	7	9	5	8	6	2	3
$\overleftarrow{PD}_x$									0

(3)

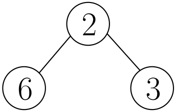
Approximate CTM (3)

x	1	4	7	9	5	8	6	2	3
$\overleftarrow{PD}_x$								0	0



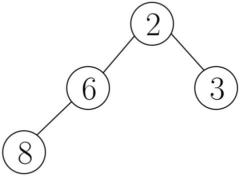
Approximate CTM (3)

x	1	4	7	9	5	8	6	2	3
$\overleftarrow{PD}_x$							1	0	0



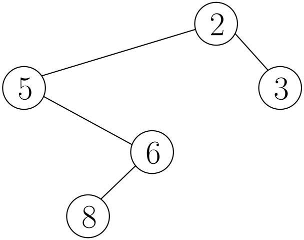
Approximate CTM (3)

x	1	4	7	9	5	8	6	2	3
$\overleftarrow{PD}_x$						1	1	0	0



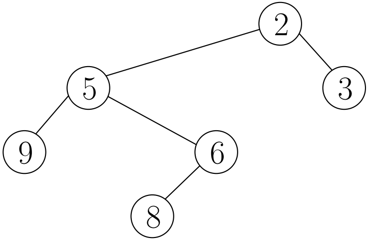
Approximate CTM (3)

x	1	4	7	9	5	8	6	2	3
$\overleftarrow{PD}_x$					3	1	1	0	0



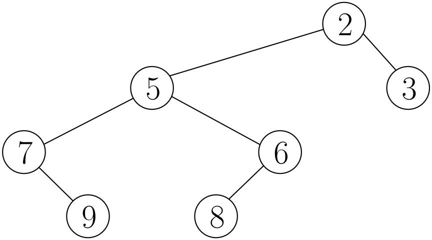
Approximate CTM (3)

x	1	4	7	9	5	8	6	2	3
$\overleftarrow{PD}_x$				1	3	1	1	0	0



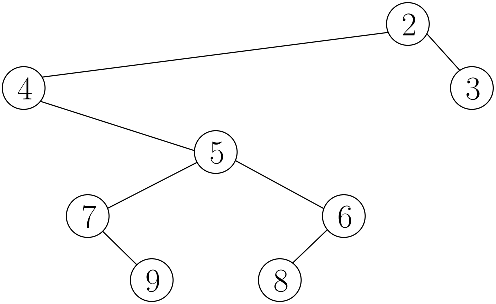
Approximate CTM (3)

x	1	4	7	9	5	8	6	2	3
$\overleftarrow{PD}_x$			2	1	3	1	1	0	0



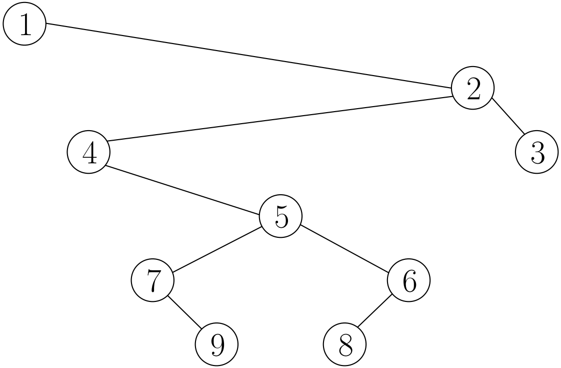
Approximate CTM (3)

x	1	4	7	9	5	8	6	2	3
$\overleftarrow{PD}_x$		6	2	1	3	1	1	0	0

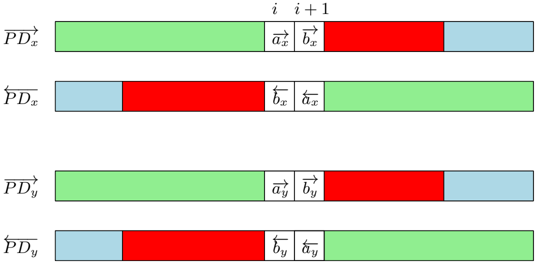


Approximate CTM (3)

x	1	4	7	9	5	8	6	2	3
$\overleftarrow{PD}_x$	0	6	2	1	3	1	1	0	0

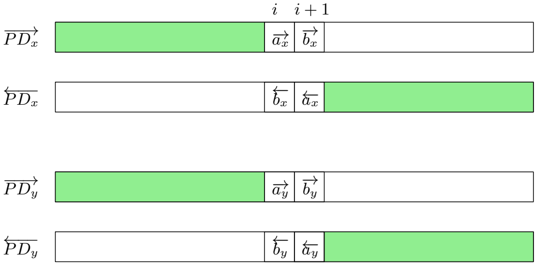


Overview



In the following, let us consider a simple example where $y = \tau(x, i)$ and $x[i] < x[i+1]$.

Green zones

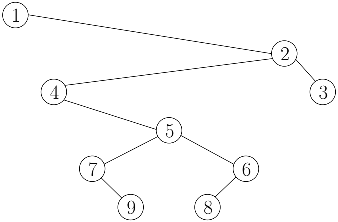


Green zones lemma

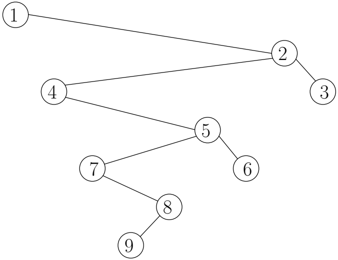
The green zones of $\overrightarrow{PD}_x$ and $\overrightarrow{PD}_y$ (resp. $\overleftarrow{PD}_x$ and $\overleftarrow{PD}_y$) are equal.

Green zones

x	1	4	7	9	5	8	6	2	3
$\overrightarrow{PD}_x$	0	1	1	1	3	1	2	7	1
$\overleftarrow{PD}_x$	0	6	2	1	3	1	1	0	0



y	1	4	7	9	8	5	6	2	3
$\overrightarrow{PD}_y$	0	1	1	1	2	4	1	7	1
$\overleftarrow{PD}_y$	0	6	3	1	1	2	1	0	0



Green zones

x	1	4	7	9	5	8	6	2	3
$\overrightarrow{PD}_x$	0	1	1	1	3	1	2	7	1
$\overleftarrow{PD}_x$	0	6	2	1	3	1	1	0	0

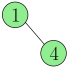
1

y	1	4	7	9	8	5	6	2	3
$\overrightarrow{PD}_y$	0	1	1	1	2	4	1	7	1
$\overleftarrow{PD}_y$	0	6	3	1	1	2	1	0	0

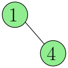
1

Green zones

x	1	4	7	9	5	8	6	2	3
$\overrightarrow{PD}_x$	0	1	1	1	3	1	2	7	1
$\overleftarrow{PD}_x$	0	6	2	1	3	1	1	0	0

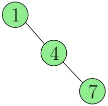


y	1	4	7	9	8	5	6	2	3
$\overrightarrow{PD}_y$	0	1	1	1	2	4	1	7	1
$\overleftarrow{PD}_y$	0	6	3	1	1	2	1	0	0

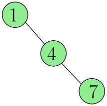


Green zones

x	1	4	7	9	5	8	6	2	3
$\overrightarrow{PD}_x$	0	1	1	1	3	1	2	7	1
$\overleftarrow{PD}_x$	0	6	2	1	3	1	1	0	0

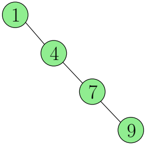


y	1	4	7	9	8	5	6	2	3
$\overrightarrow{PD}_y$	0	1	1	1	2	4	1	7	1
$\overleftarrow{PD}_y$	0	6	3	1	1	2	1	0	0

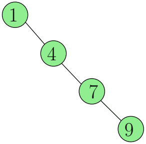


Green zones

x	1	4	7	9	5	8	6	2	3
$\overrightarrow{PD}_x$	0	1	1	1	3	1	2	7	1
$\overleftarrow{PD}_x$	0	6	2	1	3	1	1	0	0

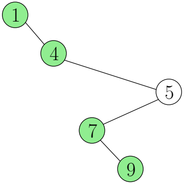


y	1	4	7	9	8	5	6	2	3
$\overrightarrow{PD}_y$	0	1	1	1	2	4	1	7	1
$\overleftarrow{PD}_y$	0	6	3	1	1	2	1	0	0

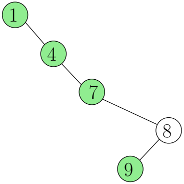


Green zones

x	1	4	7	9	5	8	6	2	3
$\overrightarrow{PD}_x$	0	1	1	1	3	1	2	7	1
$\overleftarrow{PD}_x$	0	6	2	1	3	1	1	0	0



y	1	4	7	9	8	5	6	2	3
$\overrightarrow{PD}_y$	0	1	1	1	2	4	1	7	1
$\overleftarrow{PD}_y$	0	6	3	1	1	2	1	0	0



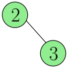
Green zones

x	1	4	7	9	5	8	6	2	3
$\overrightarrow{PD}_x$	0	1	1	1	3	1	2	7	1
$\overleftarrow{PD}_x$	0	6	2	1	3	1	1	0	0
									3

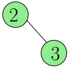
y	1	4	7	9	8	5	6	2	3
$\overrightarrow{PD}_y$	0	1	1	1	2	4	1	7	1
$\overleftarrow{PD}_y$	0	6	3	1	1	2	1	0	0
									3

Green zones

x	1	4	7	9	5	8	6	2	3
$\overrightarrow{PD}_x$	0	1	1	1	3	1	2	7	1
$\overleftarrow{PD}_x$	0	6	2	1	3	1	1	0	0

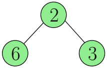


y	1	4	7	9	8	5	6	2	3
$\overrightarrow{PD}_y$	0	1	1	1	2	4	1	7	1
$\overleftarrow{PD}_y$	0	6	3	1	1	2	1	0	0

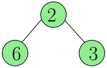


Green zones

x	1	4	7	9	5	8	6	2	3
$\overrightarrow{PD}_x$	0	1	1	1	3	1	2	7	1
$\overleftarrow{PD}_x$	0	6	2	1	3	1	1	0	0

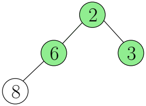


y	1	4	7	9	8	5	6	2	3
$\overrightarrow{PD}_y$	0	1	1	1	2	4	1	7	1
$\overleftarrow{PD}_y$	0	6	3	1	1	2	1	0	0

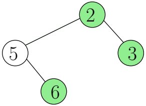


Green zones

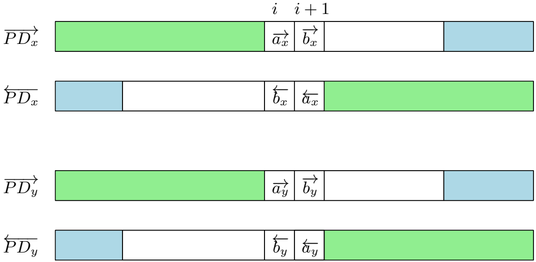
x	1	4	7	9	5	8	6	2	3
$\overrightarrow{PD}_x$	0	1	1	1	3	1	2	7	1
$\overleftarrow{PD}_x$	0	6	2	1	3	1	1	0	0



y	1	4	7	9	8	5	6	2	3
$\overrightarrow{PD}_y$	0	1	1	1	2	4	1	7	1
$\overleftarrow{PD}_y$	0	6	3	1	1	2	1	0	0



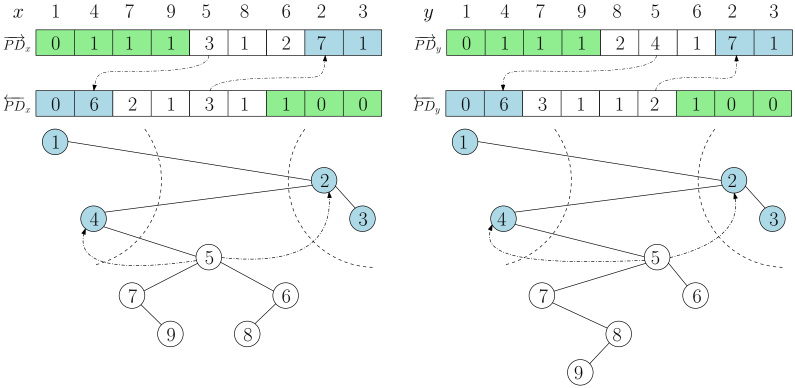
Blue zones



Blue zones lemma

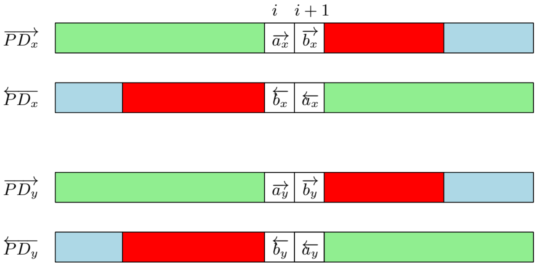
The blue zones of $\overrightarrow{PD}_x$ and $\overrightarrow{PD}_y$ (resp. $\overleftarrow{PD}_x$ and $\overleftarrow{PD}_y$) are equal.

Blue zones



x 1 4 7 9 5 8 6 2 3
$\overrightarrow{PD}_x$ 0 1 1 1 3 1 2 7 1
$\overleftarrow{PD}_x$ 0 6 2 1 3 1 1 0 0
y 1 4 7 9 8 5 6 2 3
$\overrightarrow{PD}_y$ 0 1 1 1 2 4 1 7 1
$\overleftarrow{PD}_y$ 0 6 3 1 1 2 1 0 0

Red zones

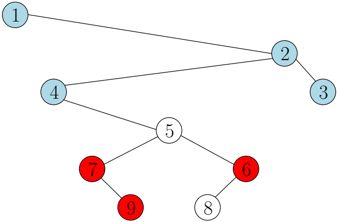


Red zones lemma

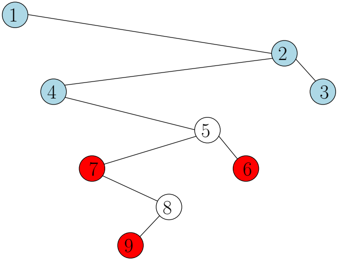
The red zones of $\overrightarrow{PD}_x$ and $\overrightarrow{PD}_y$ (resp. $\overleftarrow{PD}_x$ and $\overleftarrow{PD}_y$) differ by at most one.

Red zones

x	1	4	7	9	5	8	6	2	3
$\overrightarrow{PD}_x$	0	1	1	1	3	1	2	7	1
$\overleftarrow{PD}_x$	0	6	2	1	3	1	1	0	0



y	1	4	7	9	8	5	6	2	3
$\overrightarrow{PD}_y$	0	1	1	1	2	4	1	7	1
$\overleftarrow{PD}_y$	0	6	3	1	1	2	1	0	0



Local

x	1	4	7	9	5	8	6	2	3
$\overrightarrow{PD}_x$	0	1	1	1	3	1	2	7	1
$\overleftarrow{PD}_x$	0	6	2	1	3	1	1	0	0

y	1	4	7	9	8	5	6	2	3
$\overrightarrow{PD}_y$	0	1	1	1	2	4	1	7	1
$\overleftarrow{PD}_y$	0	6	3	1	1	2	1	0	0

Local lemma

1. $\overleftarrow{b_y} = 1$

Local

x	1	4	7	9	5	8	6	2	3
$\overrightarrow{PD}_x$	0	1	1	1	3	1	2	7	1
$\overleftarrow{PD}_x$	0	6	2	1	3	1	1	0	0

y	1	4	7	9	8	5	6	2	3
$\overrightarrow{PD}_y$	0	1	1	1	2	4	1	7	1
$\overleftarrow{PD}_y$	0	6	3	1	1	2	1	0	0

Local lemma

2. $$\overrightarrow{b_y} = \begin{cases} 0 & \text{if } \overrightarrow{a_x} = 0 \\ \overrightarrow{a_x} + 1 & \text{otherwise} \end{cases}$$

Local

x	1	4	7	9	5	8	6	2	3
$\overrightarrow{PD}_x$	0	1	1	1	3	1	2	7	1
$\overleftarrow{PD}_x$	0	6	2	1	3	1	1	0	0

y	1	4	7	9	8	5	6	2	3
$\overrightarrow{PD}_y$	0	1	1	1	2	4	1	7	1
$\overleftarrow{PD}_y$	0	6	3	1	1	2	1	0	0

Local lemma

3. $$\overleftarrow{a_y} = \begin{cases} 0 & \text{if } \overleftarrow{b_x} = 0 \\ \overleftarrow{b_x} - 1 & \text{otherwise} \end{cases}$$

Local

x	1	4	7	9	5	8	6	2	3
$\overrightarrow{PD}_x$	0	1	1	1	3	1	2	7	1
$\overleftarrow{PD}_x$	0	6	2	1	3	1	1	0	0

y	1	4	7	9	8	5	6	2	3
$\overrightarrow{PD}_y$	0	1	1	1	2	4	1	7	1
$\overleftarrow{PD}_y$	0	6	3	1	1	2	1	0	0

Local lemma

4. $$\overrightarrow{a_y} \leq \begin{cases} i-1 & \text{if } \overrightarrow{a_x} = 0 \\ \overrightarrow{a_x} & \text{otherwise} \end{cases}$$

A parent-distance based algorithm

Algorithm 1: $DoubleParentDistanceMethod(p, t)$

Input : A pattern p and a text t

Output: The occurrences that $\overset{\tau}{\approx}_{CT}\ p$ in t

$(\overrightarrow{PD}_p, \overleftarrow{PD}_p) \leftarrow$ Compute the parent-distance tables of p;

for $j \in \{1, \ldots, |t| - |p| + 1\}$ **do**

  $(\overrightarrow{PD}_x, \overleftarrow{PD}_x) \leftarrow$ Compute the parent-distance tables of $x = t[j \ldots j + p - 1]$;

  if $\overrightarrow{PD}_p = \overrightarrow{PD}_x$ **then**

    An occurrence has been found;

  else

    foreach Eligible position for a swap **do**

      if Lemmas Blue, Red and Local hold **then**

        An occurrence has been found;

A parent-distance based algorithm

Complexity

The parent-distance based algorithm has a worst-case time complexity of $\Theta(mn)$ and a $\Theta(m)$ space complexity (where m is the length of the pattern and n the length of the text).

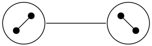
Swap graph

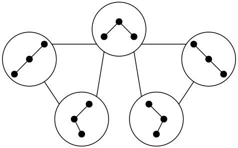
Definition

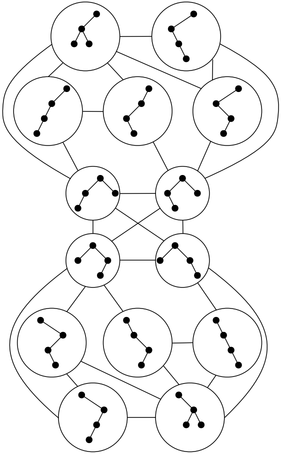
The swap graph of Cartesian trees for a given n is a graph where:

- The vertices are the Cartesian trees of size n
- There is an edge between two vertices T and T' if there exist 2 sequences x and y such that:
  $C(x) = T$, $C(y) = T'$ and $x \overset{\tau}{\approx}_{CT} y$

Swap graph







Lower bound

Number of Cartesian trees

The number of Cartesian tree T with n nodes is the n-th Catalan number:

$$C_n = \frac{1}{n+1}\binom{2n}{n} = O\left(\frac{4^n}{n^{3/2}}\right)$$

Neighbours lemma

The number of neighbours $|ng(T)|$ of a given Cartesian tree T is bounded, and we have:

$$n - 1 \leq |ng(T)| \leq \lceil 3(n-1) - 2(\log_2(n+1) - 1) \rceil$$

Lower bound

Number of Cartesian trees

The number of Cartesian tree T with n nodes is the n-th Catalan number:

$$C_n = \frac{1}{n+1}\binom{2n}{n} = O\left(\frac{4^n}{n^{3/2}}\right)$$

Neighbours lemma

The number of neighbours $|ng(T)|$ of a given Cartesian tree T is bounded, and we have:

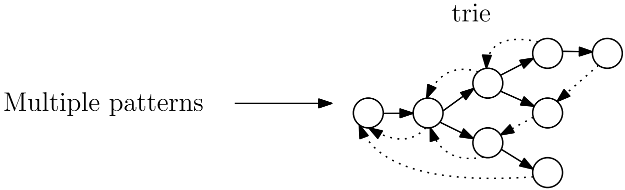
$$n - 1 \leq |ng(T)| \leq \lceil 3(n-1) - 2(\log_2(n+1) - 1) \rceil$$

A lower bound for the graph diameter

The diameter of the swap graph is $\Omega(\frac{n}{\ln n})$.

An Aho-Corasick based algorithm

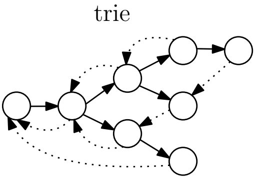
[Alfred V. Aho, Margaret J. Corasick 1975]



An Aho-Corasick based algorithm

[S. G. Park, A. Amir, G. M. Landau, K. Park 2019]

Multiple patterns ⟶ Parent-distances ⟶



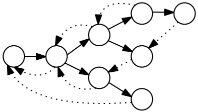
An Aho-Corasick based algorithm

Single pattern

↓

Cartesian tree + neighborhood ⟶ Parent-distances ⟶

trie



An Aho-Corasick based algorithm

Complexity

The Aho-Corasick based algorithm has an $O((m^2 + n) \log m)$ worst-case time complexity and an $O(m^2)$ space complexity (where m is the length of the pattern and n the length of the text).

Closing words

Perspectives

- Generalize our results
- Use another representation of CT
- Introduce new metrics for approximate CTM
- Filtration
- Average complexity

Thank you for your attention!