

# Quantum Multi-string Matching

Allen Liu   Kevin Tong

University of Waterloo

ECE 405C Winter 2025

March 31, 2025

# Presentation Overview

## 1 Problem Definition

## 2 Classical Algorithms

- Existing Algorithms
- Polynomial Matching
- Polynomial Multiplication

## 3 Quantum Algorithms

- Polynomial Multiplication with QFT

# String Matching

- Text  $S[0 \dots n - 1]$
- Pattern (or key)  $P[0 \dots m - 1]$
- Strings from alphabet  $\Sigma$
- Find set of indices  $i$  such that  $S[i..i + m] = P$

Example:

- $T = \text{"GTAT GATC TC"}$  (ignore spaces)
- $P_1 = \text{"ATCT"}$
- $P_2 = \text{"TGAT"}$
- $P_3 = \text{"ACCC"}$
- $P_1$  matches at 5,  $P_2$  matches at 3,  $P_3$  has no matches
- Multi-string matching: search  $P_1, P_2, \dots$  in  $S$  simultaneously

# Classical Algorithms

- Brute Force  $\rightarrow O(mn)$
- Boyer-Moore  $\rightarrow O(n + m)$ ,  $O(mn)$  worst case
- Knuth-Morris-Pratt  $\rightarrow O(n + m)$  worst case
- Suffix Tree/Array  $\rightarrow O(m)/O(n \log n) + \text{preproc}$
- Karp-Rabin - rolling hash  $\rightarrow O(n + m)$  expected

G	T	A	T	G	A	T	C	T	C
hash-value 84									
	hash-value 194								
		hash-value 6							
			hash-value 18						
				hash-value 95					

- **Polynomial Matching**  $\rightarrow O(n \log n)$

# Polynomial Matching

Calculate a fingerprint for  $P$  and every  $m$  character sequence in  $S$ ;  
**matching fingerprints** suggest pattern match!

$$A \mapsto -3$$

$$C \mapsto 5$$

$$G \mapsto -7$$

$$T \mapsto 11$$

$$\text{"ATCT"} \longrightarrow 11x^3 + 5x^2 + 11x - 3$$

$$\text{"GTAT GATC TC"} \longrightarrow -7x^{15} + 11x^{14} - 3x^{13} + \dots + 11x^7 + 5x^6$$

Note that  $P$  encoding is reversed (similar to convolution)

# Polynomial Matching

Given polynomials

$$P(x) = \sum_{i=0}^m a_i x^i, S(x) = \sum_{j=0}^n b_j x^j$$

then

$$R(x) = \sum_{k=0}^{m+n} c_k x^k$$

where

$$c_k = \sum_{i=0}^k a_i b_{k-i}, \quad \text{for } 0 \leq k \leq m+n,$$

with the convention that  $a_i = 0$  for  $i > m$  and  $b_j = 0$  for  $j > n$ .

# Polynomial Matching

Coefficients of  $R(x)$  correspond to “dot products” between substrings.

$$P(x) = 11x^3 + 5x^2 + 11x - 3$$

$$S(x) = -7x^{15} + 11x^{14} - 3x^{13} + \dots + 11x^7 + 5x^6$$

$$R(x) = \dots + 71x^8 + 0x^9 + 276x^{10} + 98x^{11} - 4x^{12} + \dots$$

Degrees map to index:

15 yields index 0

14 yields index 1

...

11 yields index 4

10 yields index 5

9 yields index 6

# Polynomial Matching

Coefficients of  $R(x)$  correspond to “dot products” between substrings.

$$P(x) = 11x^3 + 5x^2 + 11x - 3$$

$$S(x) = -7x^{15} + 11x^{14} - 3x^{13} + \dots + 11x^7 + 5x^6$$

$$R(x) = \dots + 71x^8 + 0x^9 + 276x^{10} + 98x^{11} - 4x^{12} + \dots$$

Notice that  $\|P\|^2 = 11^2 + 5^2 + 11^2 + (-3)^2 = 276$

We get exact fingerprint match for single patterns. Let's extend to multiple patterns...



# Polynomial Matching

Add patterns together:

$$P(x) = P_1(x) + P_2(x)$$

$$S(x) = -7x^{15} + 11x^{14} - 3x^{13} + \dots + 11x^7 + 5x^6$$

$$R(x) = \dots + 94x^8 + 240x^9 + 272x^{10} + 64x^{11} + 296x^{12} - 60x^{13} \dots$$

No more exact matches, but higher values = likelier index.

# Polynomial Multiplication

For two polynomials of degree  $n$ ,

$$A(x) = a_0 + a_1x + a_2x^2 + \cdots + a_{n-1}x^{n-1}$$

$$B(x) = b_0 + b_1x + b_2x^2 + \cdots + b_{n-1}x^{n-1}$$

The goal is to simply to find their product:

$$C(x) = \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} a_k b_k x^{j+k}$$

- Naïve polynomial multiplication  $\rightarrow O(n^2)$
- Karatsuba Algorithm  $\rightarrow O(n^{\log_3 2}) = O(n^{1.59})$
- FFT  $\rightarrow O(n \log n)$

# Polynomial Multiplication

For two polynomials of degree  $n$ ,

$$A(x) = a_0 + a_1x + a_2x^2 + \cdots + a_{n-1}x^{n-1}$$

$$B(x) = b_0 + b_1x + b_2x^2 + \cdots + b_{n-1}x^{n-1}$$

The goal is to simply to find their product:

$$C(x) = \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} a_k b_k x^{j+k}$$

The naïve approach is to directly evaluate the summation (i.e. long multiplication/grade-school multiplication).

# Efficient Polynomial Multiplication

Multiple algorithms for multiplication:

- Naïve polynomial multiplication  $\rightarrow O(n^2)$
- Karatsuba Algorithm  $\rightarrow O(n^{\log_3 2}) = O(n^{1.59})$
- FFT  $\rightarrow O(n \log n)$

# Polynomial Multiplication with FFT

- 1 Apply FFT to the coefficients of  $A(x)$  and  $B(x)$

With  $\vec{a} = [a_0 \ a_1 \ \dots \ a_{n-1}]^T$  and  $\vec{b} = [b_0 \ b_1 \ \dots \ b_{n-1}]^T$ , then

$$\vec{\alpha} = \text{FFT}(\vec{a}) \quad O(n \log n)$$

$$\vec{\beta} = \text{FFT}(\vec{b}) \quad O(n \log n)$$

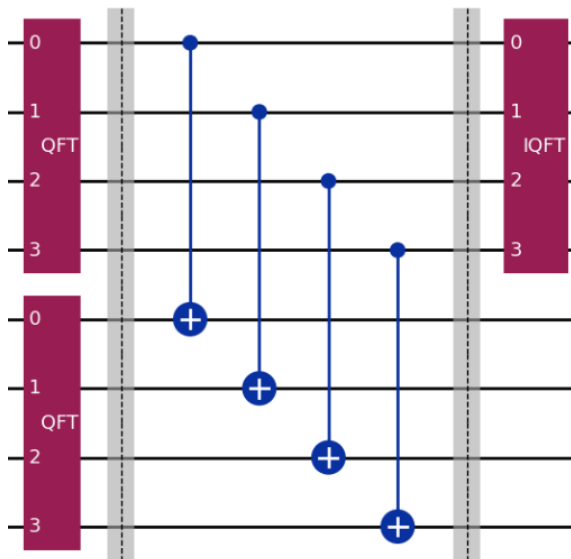
- 2 Perform the Hadamard (element-wise) product of coefficients

$$\begin{aligned} \vec{\gamma} &= \vec{\alpha} \odot \vec{\beta} \\ &= [\alpha_0 \beta_0 \ \alpha_1 \beta_1 \ \dots \ \alpha_{n-1} \beta_{n-1}]^T \quad O(n) \end{aligned}$$

- 3 Apply Inverse FFT

$$\vec{c} = \text{IFFT}(\vec{\gamma}) \quad O(n \log n)$$

# Polynomial Multiplication with QFT



# Lists

## Bullet Points and Numbered Lists

- Lorem ipsum dolor sit amet, consectetur adipiscing elit
  - Aliquam blandit faucibus nisi, sit amet dapibus enim tempus
    - Lorem ipsum dolor sit amet, consectetur adipiscing elit
    - Nam cursus est eget velit posuere pellentesque
  - Nulla commodo, erat quis gravida posuere, elit lacus lobortis est, quis porttitor odio mauris at libero
- 
- 1 Nam cursus est eget velit posuere pellentesque
  - 2 Vestibulum faucibus velit a augue condimentum quis convallis nulla gravida

# Blocks of Highlighted Text

## Block Title

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer lectus nisl, ultricies in feugiat rutrum, porttitor sit amet augue.

## Example Block Title

Aliquam ut tortor mauris. Sed volutpat ante purus, quis accumsan.

## Alert Block Title

Pellentesque sed tellus purus. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos.

Suspendisse tincidunt sagittis gravida. Curabitur condimentum, enim sed venenatis rutrum, ipsum neque consectetur orci.



# Multiple Columns

Subtitle

## Heading

- 1 Statement
- 2 Explanation
- 3 Example

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer lectus nisl, ultricies in feugiat rutrum, porttitor sit amet augue. Aliquam ut tortor mauris. Sed volutpat ante purus, quis accumsan dolor.

# Table

Subtitle

<b>Treatments</b>	<b>Response 1</b>	<b>Response 2</b>
Treatment 1	0.0003262	0.562
Treatment 2	0.0015681	0.910
Treatment 3	0.0009271	0.296

**Table:** Table caption

# Definitions & Examples

## Definition

A **prime number** is a number that has exactly two divisors.

## Example

- 2 is prime (two divisors: 1 and 2).
- 3 is prime (two divisors: 1 and 3).
- 4 is not prime (**three** divisors: 1, 2, and 4).

You can also use the theorem, lemma, proof and corollary environments.

# Theorem, Corollary & Proof

Theorem (Mass-energy equivalence)

$$E = mc^2$$

Corollary

$$x + y = y + x$$

Proof.

$$\omega + \phi = \epsilon$$



# Equation

$$\cos^3 \theta = \frac{1}{4} \cos \theta + \frac{3}{4} \cos 3\theta \quad (1)$$

## Example (Theorem Slide Code)

```
\begin{frame}  
\frametitle{Theorem}  
\begin{theorem}[Mass--energy equivalence]  
$E = mc^2$  
\end{theorem}  
\end{frame}
```

Slide without title.

# Citing References

An example of the `\cite` command to cite within the presentation:

This statement requires citation [Smith, 2022, Kennedy, 2023].



# References



John Smith (2022)

Publication title

*Journal Name* 12(3), 45 – 678.



Annabelle Kennedy (2023)

Publication title

*Journal Name* 12(3), 45 – 678.

# Acknowledgements

## Smith Lab

- Alice Smith
- Devon Brown

## Cook Lab

- Margaret
- Jennifer
- Yuan

## Funding

- British Royal Navy
- Norwegian Government

# The End

Questions? Comments?