Quantum Fourier Transform

Introduction to Quantum Computing

Jothishwaran C A

Department of Electronics and Communication Engineering
Indian Institute of Technology Roorkee

The Discrete Fourier Transform

- Consider complex valued sequences of length N given by:

$$\boldsymbol{X} = x_0 x_1 \dots x_{N-1} \quad ; \; x_i \in \mathbb{C}$$

- This sequence can be treated as a vector belonging to the vector space $\mathbb{C}^n$.

- This space has the standard basis defined as follows:

$$\begin{aligned}
\boldsymbol{v}_0 &= (1 \quad 0 \quad \dots \quad 0 \quad 0) \\
\boldsymbol{v}_1 &= (0 \quad 1 \quad \dots \quad 0 \quad 0) \\
&\quad\quad\quad\quad \vdots \\
\boldsymbol{v}_{N-2} &= (0 \quad 0 \quad \dots \quad 1 \quad 0) \\
\boldsymbol{v}_{N-1} &= (0 \quad 0 \quad \dots \quad 0 \quad 1)
\end{aligned}$$

- Therefore the vector may be represented as, $\boldsymbol{X} = \sum_{k=0}^{N-1} x_k \boldsymbol{v}_k$.

The Discrete Fourier Transform

- It is also possible to represent the same sequence as a sum of the following periodic functions:

$$\boldsymbol{b}_k[n] = e^{\frac{2\pi\, i}{N}kn}\,, n \in [0, N-1] \qquad i^2 = -1$$

- To show this is true, we shall consider expanding the standard basis in terms of the new basis:

$$\boldsymbol{v}_j = \sum_{k=0}^{N-1} b_j^*[k]\, b_k[n]$$

- We shall see some examples for the same:

$$N = 4$$

$$\vec{X} = x_0\, x_1\, x_2\, x_3$$

$$v_0 = [1 \quad 0 \quad 0 \quad 0]$$

$$v_1 = [0 \quad 1 \quad 0 \quad 0]$$

$$v_2 = [0 \quad 0 \quad 1 \quad 0]$$

$$v_3 = [0 \quad 0 \quad 0 \quad 1]$$

$$b_k[n] = e^{i \frac{2\pi}{4} k n} \quad , \; n = 0, 1, 2, 3$$

$$= e^{i \pi/2 \cdot k n} = \left(\cos\frac{\pi}{2} + i \sin\frac{\pi}{2} \right)$$

$$= i^{kn}$$

$$\left.\begin{array}{c|l} k = & \quad n=0 \quad\; 1 \quad\; 2 \quad\; 3 \\ 0 & b_0 = [1 \quad 1 \quad 1 \quad 1] \\ 1 & b_1 = [1 \quad i \quad -1 \quad -i] \\ 2 & b_2 = [1 \quad -1 \quad 1 \quad -1] \\ 3 & b_3 = [1 \quad -i \quad -1 \quad i] \end{array}\right] \rightarrow \text{New basis}$$

$$v_0 = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} = \frac{\begin{bmatrix} b_0 + b_1 + b_2 + b_3 \end{bmatrix}}{4} = \begin{bmatrix} 4 & 0 & 0 & 0 \end{bmatrix} \times \frac{1}{4}$$

$$v_1 = \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix} = \frac{1}{4}\left[\underbrace{\begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix}}_{b_0} - i \underbrace{\begin{bmatrix} 1 & i & -1 & -i \end{bmatrix}}_{b_1} - \underbrace{\begin{bmatrix} 1 & -1 & 1 & -1 \end{bmatrix}}_{b_2} + i \underbrace{\begin{bmatrix} 1 & -i & -1 & i \end{bmatrix}}_{b_3} \right]$$

$$= \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix}$$

$$v_2 = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix} = \frac{\begin{bmatrix} b_0 - b_1 + b_2 - b_3 \end{bmatrix}}{4}$$

$$v_3 = [0 \quad 0 \quad 0 \quad 1] = \frac{[b_0 + i b_1 - b_2 - i b_3]}{4}$$

vector

$$v_j = \frac{1}{4} \sum_{k=0}^{3} \underbrace{b_j^*[k]}_{\text{single number}} \; b_k[n] \longrightarrow \text{vector}$$

complex conjugate

Generalizing

$$v_j = \frac{1}{N} \sum_{k=0}^{N-1} b_j^*[k] \cdot b_k[n]$$

$$\vec{X} = \sum_{j=0}^{N-1} x_j \tilde{v}_j \longrightarrow \text{Std. basis expansion}$$

$$= \frac{1}{N} \sum_{j=0}^{N-1} x_j \sum_{k=0}^{N-1} b_j^*[k] \, \tilde{b}_k$$

$$= \frac{1}{N} \sum_{k=0}^{N-1} \left[\sum_{j=0}^{N-1} x_j b_j^*[k] \right] \tilde{b}_k$$

DISCRETE FOURIER TRANSFORM (DFT)

Std. basis

$x[n] \; ; \; n \in [0, N-1]$

$\Big\updownarrow$ DFT

$\tilde{x}[n] \; ; \; n \in [0, N-1]$

$$\tilde{x}[k] = \frac{1}{N} \sum_{j=0}^{N-1} x_j \, b_j^*[k]$$

→ Discrete Fourier transform

$$x[n] = \sum_{k=0}^{N-1} \tilde{x}[k] \, b_k$$

Fourier Transformation

→ Represent any vector as a linear combination of vectors with Harmonic Coefficients (Sine & cosine)

$$e^{\frac{2\pi i}{N} k n} = \cos\left(\frac{2\pi kn}{N}\right) + i \sin\left(\frac{2\pi kn}{N}\right)$$

→ Continuous time (Fourier Transform)

Discrete time (DFT) ✓

+ other variants also exist

Why Fourier Transform?

Allows us to study the periodicity properties.

$$v_j' = \frac{1}{\sqrt{N}} v_j$$

$$b_k' = \frac{1}{\sqrt{N}} b_k$$

$$\tilde{X}[k] = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} x_j \left(b_j'[k]\right)^*$$

Basis Change Matrix for $N = 4$

$$B : \quad B v_j = b_j$$

$$B = k \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -1 \\ 1 & -i & -1 & i \end{bmatrix} \quad ; \quad B^{\dagger} = k \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -i & -1 & i \\ 1 & -1 & 1 & -1 \\ 1 & i & -1 & -i \end{bmatrix}$$

$B^{\dagger} \downarrow$ Transposed conjugate

$$B B^{\dagger} = \begin{bmatrix} 4 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 4 \end{bmatrix} = 4 \, \underline{I}$$

$$\tilde{B} = \frac{1}{2} B \Rightarrow \tilde{B}^{\dagger} = \frac{1}{2} B^{\dagger}$$

$$\rightarrow \quad k = \frac{1}{2}$$

The previous result is generalizes for all N

$\rightarrow$ $N = 2^n$; not be confused with the previous 'n'

then, the basis change matrix has dimensions $2^n \times 2^n$
and is unitary

$\Rightarrow$ there (potentially) exists a quantum circuit that does the same operation

How to perform QFT ?
(Discrete)

QUANTUM FOURIER TRANSFORM

→ What about $N=2$

$$b_k[j] = e^{i\frac{2\pi}{2} \times k \times j}$$

$$= e^{i\pi \times k \times j} = (-1)^{kj}$$

$$b_0 = [1 \;,\; 1]$$

$$b_1 = [1 \;,\; -1]$$

$$B = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = H$$

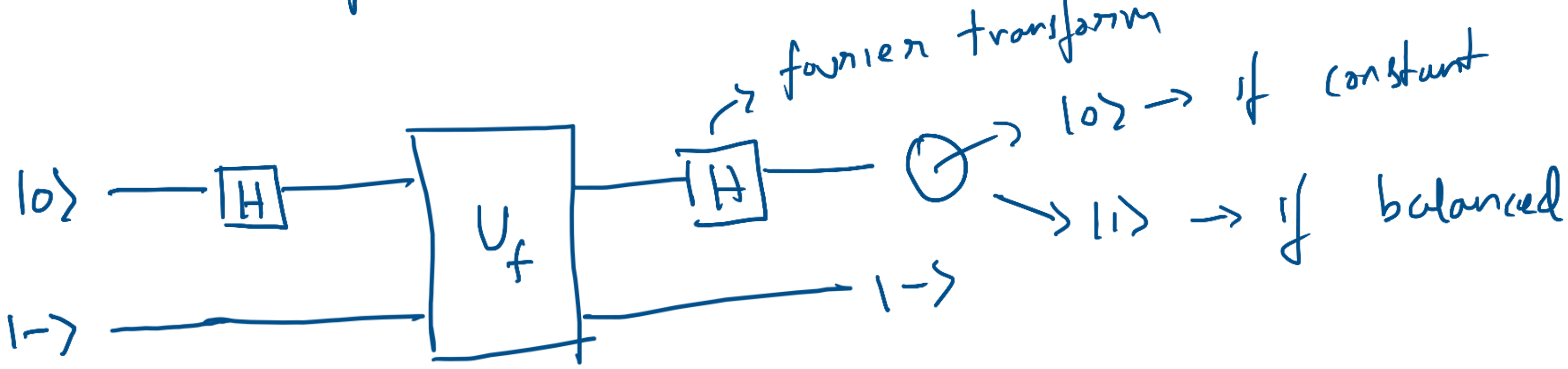
; QFT Matrix for $N = 2 = 2^1$

→ Deutsch Problem (QFT for $N=2$)

consider a single bit Boolean function, $f = f_0, f_1, f_2, f_3$

f_0, f_3 – constant functions

f_1, f_2 – balanced functions



QFT for $N=2$

$\rightarrow$ QFT for $N = 2^n$; # qubits $= \underline{\underline{n}}$

$N \downarrow$ length of the sequence

$$\begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \alpha|0\rangle + \beta|1\rangle \quad ; \quad \alpha, \beta \in \mathbb{C} \text{ and } |\alpha|^2 + |\beta|^2 = 1$$

How to represent $\vec{x}$ in a qubit state

$$\vec{x} = \sum_{j=0}^{N-1} x_j \hat{v}_j \quad ; \quad |\vec{x}| = 1 \quad \text{(any vector can be normalized)}$$

$$|\vec{x}\rangle = \sum_{j=0}^{N-1} x_j |j\rangle$$

; $|j\rangle$ is the binary vector corresponding to the integer 'j'

$|0\rangle = |0\rangle \otimes |0\rangle \otimes |0\rangle \otimes - - - \otimes |0\rangle$

n - qubits

$|2^n - 1\rangle$

$|N-1\rangle = |1\rangle \otimes |1\rangle \otimes |1\rangle \otimes - - - - - \otimes |1\rangle$

n - qubit

$|\vec{x}\rangle = \sum_{j=0}^{N-1} x_j |j\rangle$

$\vec{x} \in \mathbb{C}^{2^n} \rightarrow \mathbb{C} \times \mathbb{C} \times \mathbb{C} \times - - - \times \mathbb{C}$

2^n times

$|\vec{x}\rangle \in \mathbb{C}^2 \otimes \mathbb{C}^2 \otimes \mathbb{C}^2 \otimes - - - \otimes \mathbb{C}^2$

n - times

n - qubit state space.

In both cases the ordering is implied

→ Basis change matrix for $N = 4 \Rightarrow n = 2$

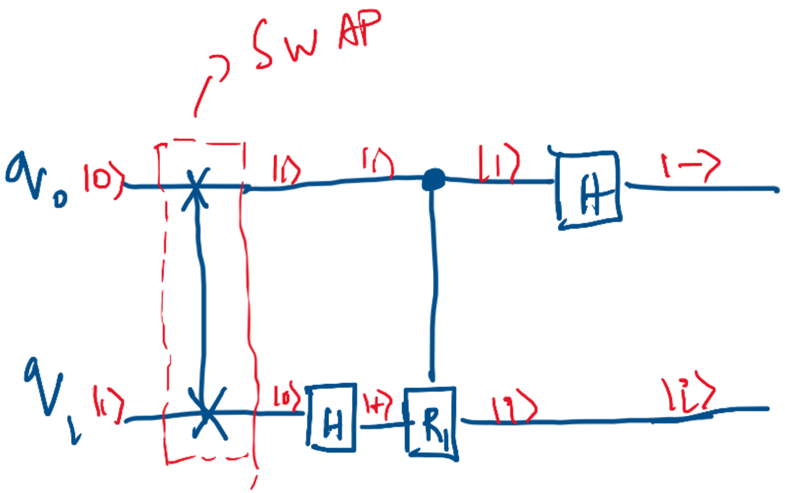
$$|0\rangle = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

[please include normalization factors]

$$U|0\rangle = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} \; ; \quad U|1\rangle = \begin{pmatrix} 1 \\ i \\ -1 \\ -i \end{pmatrix} \; ; \quad U|2\rangle = \begin{pmatrix} 1 \\ -1 \\ 1 \\ -1 \end{pmatrix} \; ; \quad U|3\rangle = \begin{pmatrix} 1 \\ -i \\ -1 \\ i \end{pmatrix}$$

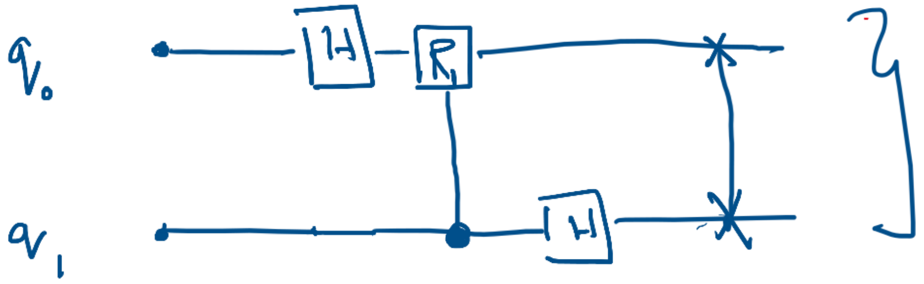
$$\begin{array}{cccc} |0\rangle|0\rangle & |0\rangle|1\rangle & |1\rangle|0\rangle & |1\rangle|1\rangle \\ q_0\ q_1 & q_0\ q_1 & q_0\ q_1 & q_0\ q_1 \end{array}$$

$= \underline{|+\rangle|+\rangle}$ $\qquad = \underline{|-\rangle}\ \underline{|i\rangle}$ $\qquad = |+\rangle\,|-\rangle$ $\qquad = |-\rangle|-i\rangle$



$$\begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \end{pmatrix}$$

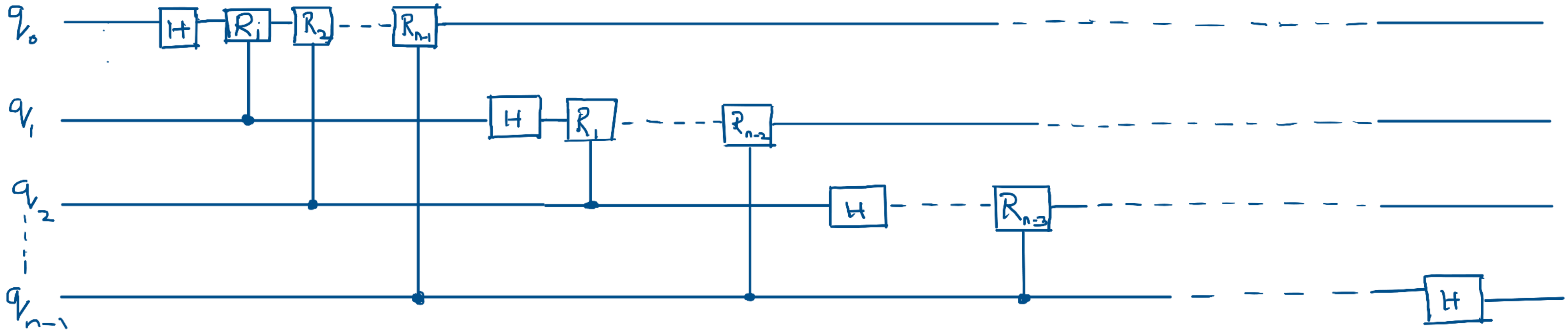
$$R_1 = \begin{bmatrix} 1 & & & 0 \\ & 1 & & \\ & & 1 & \\ 0 & & & i \end{bmatrix} \times \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ i \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes |i\rangle$$



Output will be the same as before other than the swap gate (the order of the qubits will be reversed)

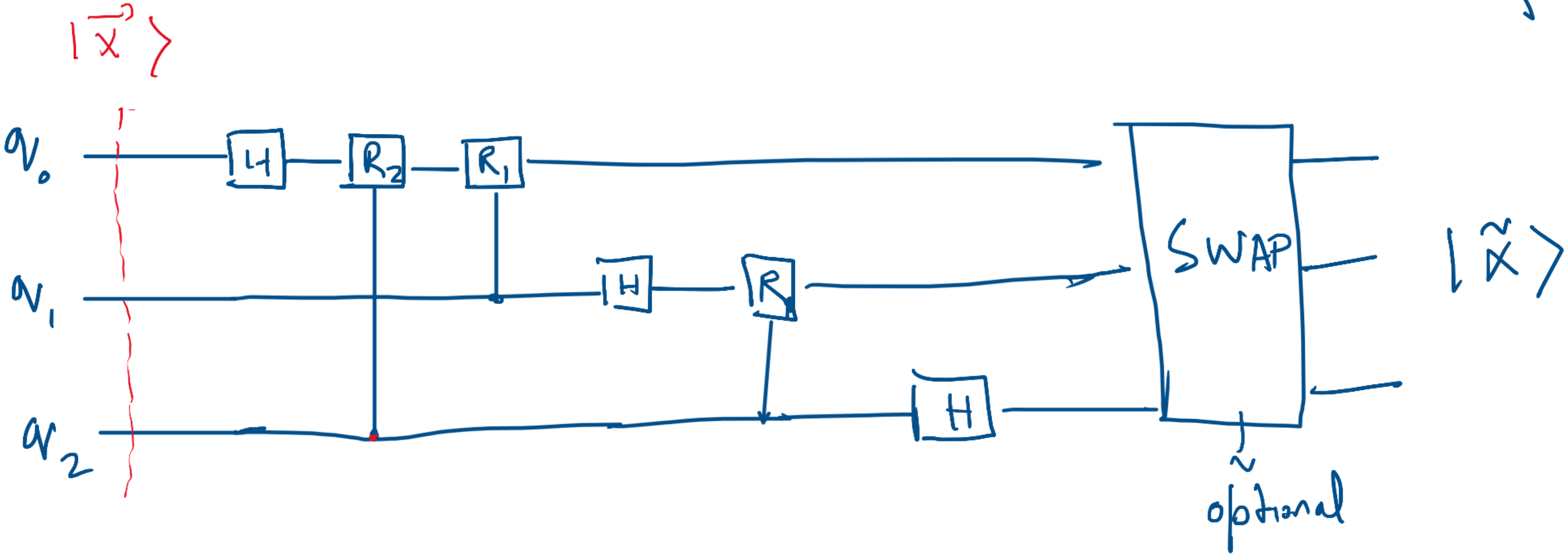
The SWAP gate may be applied before or after the Rotation gates. Typically the SWAP operations are performed after the rotations.

QFT for $N = 2^n$



$N = 8 \quad ; \quad n = 3$

$$R_j = \begin{bmatrix} 1 & & & 0 \\ & 1 & & \\ & & 1 & \\ 0 & & & e^{i\pi/2} \end{bmatrix}$$



R_j — is a controlled rotation matrix about the z-axis.

$$\omega = e^{i 2\pi/N} \Rightarrow R_j = \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \\ & & & \omega^j \end{bmatrix} \quad \forall\, j \neq 0 \quad ; \quad \text{if } j = 0 \;\; R_j = H$$

COMPLEXITY OF QFT

(Computational)

n - Hadamard gates

$$\# R \text{ gates} = (n-1) + (n-2) + \cdots + 1$$

$$= \frac{n(n-1)}{2} = O(n^2)$$

$\therefore$ QFT requires $O(n^2)$ quantum gates   exponential speed up

DFT requires $N \log N = n \underbrace{2^n}$ multiplications

Can every $\vec{x}$ be encoded efficiently into $|\vec{x}\rangle$?

$\rightarrow$ No !!!

Creating a general $|\vec{x}\rangle$ is an NP-Hard Problem

(Harder than Classical DFT)

$\therefore$ If we include the encoding & read-out complexities, the QFT is almost invariably slower than DFT

How to encode $\vec{X}$; $N = 4$; $n = 2$

$$\vec{X} = [1 \quad 2 \quad 3 \quad 4]$$

$$= 1 + 4 + 9 + 16 = 30$$

$$|\vec{X}\rangle = \frac{1}{\sqrt{30}}\left[\,|00\rangle + 2\,|01\rangle + 3\,|10\rangle + 4\,|11\rangle\,\right]$$