แก้ไขครั้งล่าสุดเมื่อวันที่ 22 เมษายน 2568

Chapter 11
5G

CHAPTER 17

WIRELESS TRANSMISSION TECHNIQUES

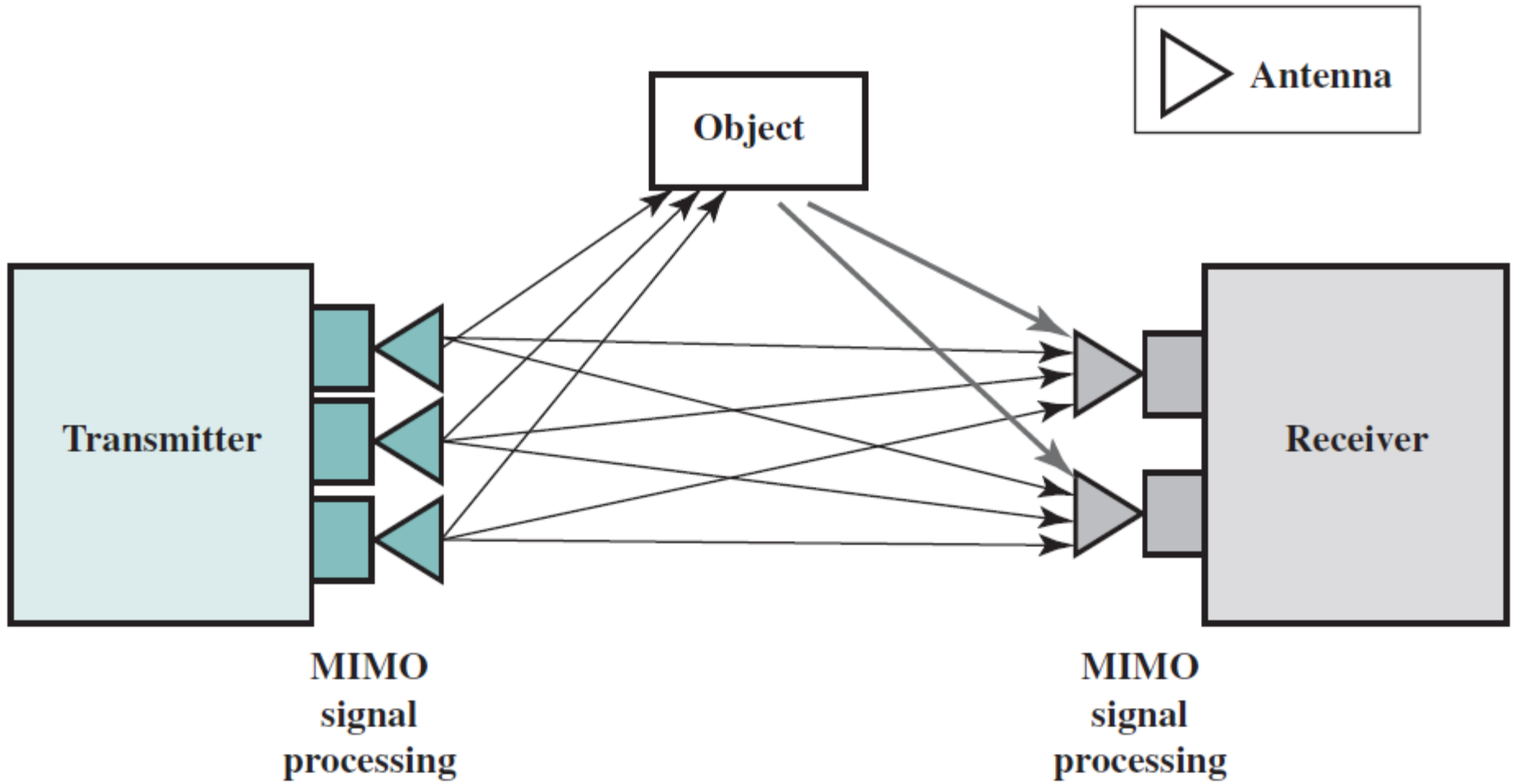
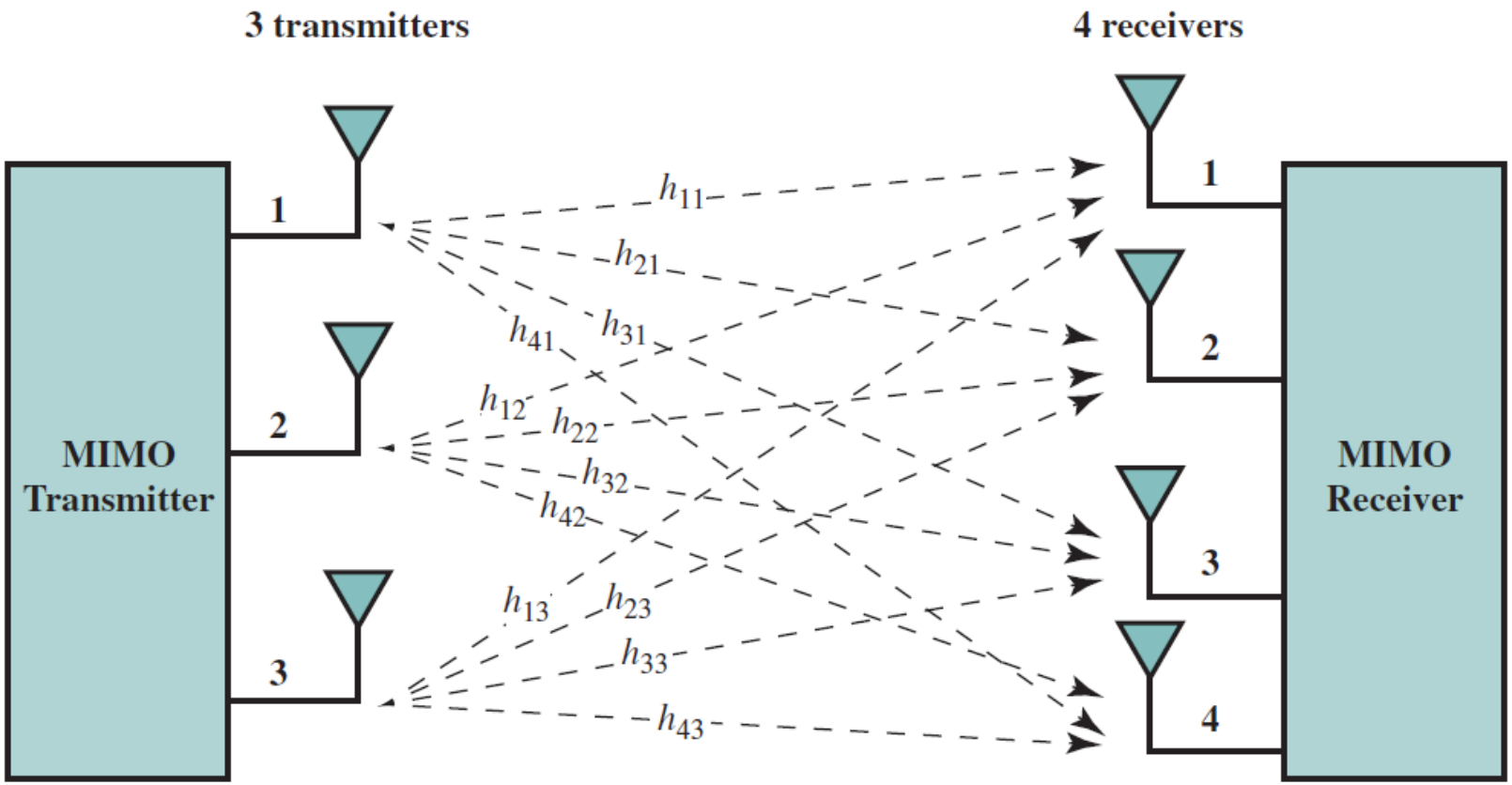


Figure 17.1 MIMO Scheme

- **Spatial diversity:** The same data is coded and transmitted through multiple antennas, which effectively increases the power in the channel proportional to the number of transmitting antennas. This improves signal-to-noise (SNR) for cell edge performance. Further, diverse multipath fading offers multiple "views" of the transmitted data at the receiver, thus increasing robustness. In a multipath scenario where each receiving antenna would experience a different interference environment, there is a high probability that if one antenna is suffering a high level of fading, another antenna has sufficient signal level. ท เสาอากาศ ส่งข้อมูลชุดเดียวกัน (redundant)
- **Spatial multiplexing:** A source data stream is divided among the transmitting antennas. The gain in channel capacity is proportional to the available number of antennas at the transmitter or receiver, whichever is less. Spatial multiplexing can be used when transmitting conditions are favorable and for relatively short distances compared to spatial diversity. The receiver must do considerable signal processing to sort out the incoming substreams, all of which are transmitting in the same frequency channel, and to recover the individual data streams. ท เสาอากาศ ส่งข้อมูลต่างกัน (parallel) ทำได้ในบางโอกาสเท่านั้น



3 transmit antennas

$$\mathrm{H} = \overbrace{\begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \\ h_{41} & h_{42} & h_{43} \end{bmatrix}}^{\text{4 receive antennas}}$$

Figure 17.2 3 × 4 MIMO Scheme

iPhone 16 Pro

5G (sub-6 GHz) พร้อม MIMO แบบ 4x4

Gigabit LTE พร้อม MIMO แบบ 4x4

Wi-Fi 7 (มาตรฐาน 802.11be) พร้อม MIMO แบบ 2x2

For spatial multiplexing, there is a multilink channel that can be expressed as $\mathbf{y} = \mathbf{Hc} + \mathbf{n}$, where $\mathbf{y}$ is the vector of received signals, $\mathbf{c}$ is the vector of transmitted signals, $\mathbf{n}$ is an additive noise component, and $\mathbf{H} = [h_{ij}]$ is an $r \times t$ channel matrix, with r being the number of receiving antennas and t the number of transmitting antennas. The number of spatial data streams is $\min[r, t]$. For a channel with three transmitters and four receivers (Figure 17.2), the equation is:

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \\ h_{41} & h_{42} & h_{43} \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} + \begin{bmatrix} n_1 \\ n_2 \\ n_3 \\ n_4 \end{bmatrix}$$

Rx = Receiver

Environment

Tx = Transceiver

Noise

The h_{ij} are complex numbers $x + jz$ that represent both the amplitude attenuation (x) over the channel and the path dependent phase shift (z), and the n_i are additive noise components. The receiver measures the channel gains based on training fields containing known patterns in the packet preamble and can estimate the transmitted signal with the following equation:

$$\begin{bmatrix} \hat{c}_1 \\ \hat{c}_2 \\ \hat{c}_3 \end{bmatrix} = \mathbf{H}^{-1} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix}$$

ผู้ส่งส่ง known pattern มาก่อน
เพื่อให้ผู้รับคำนวณหา H หรือ H^{-1}
determinant = 0 จะหา H^{-1} ไม่ได้

Multiple-User MIMO

Multiple-user MIMO (MU-MIMO) extends the basic MIMO concept to multiple endpoints, each with multiple antennas. The advantage of MU-MIMO compared to single-user MIMO is that the available capacity can be shared to meet time-varying demands. MU-MIMO techniques are used in both Wi-Fi and 4G cellular networks.

There are two applications of MU-MIMO:

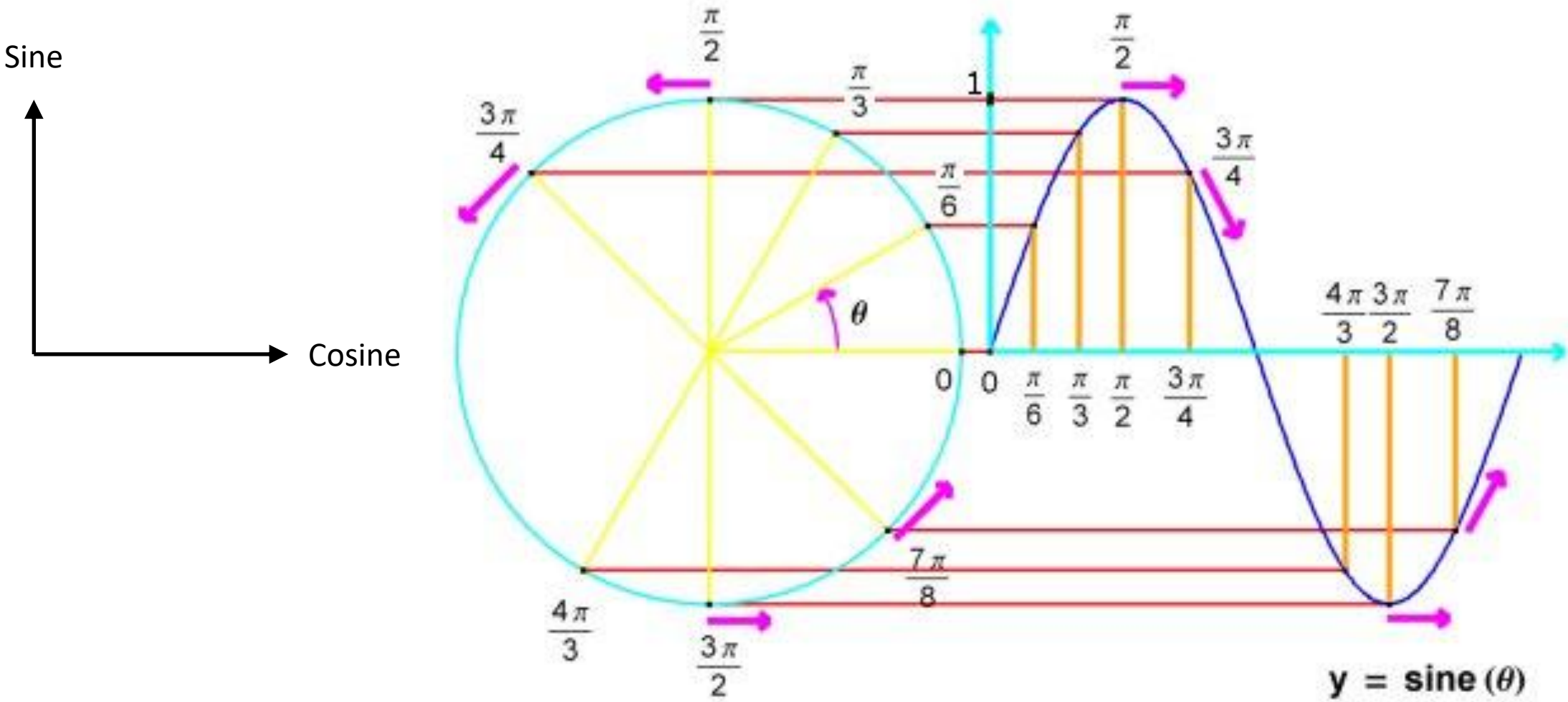
- **Uplink—Multiple Access Channel, MAC:** Multiple end users transmit simultaneously to a single base station.
- **Downlink—Broadcast Channel, BC:** The base station transmits separate data streams to multiple independent users.

Base station ต้องมีเสาอากาศจำนวนมาก เพื่อให้บริการผู้ใช้หลายคนได้พร้อม ๆ กัน
Base station กำหนดเสาอากาศให้ผู้ใช้แต่ละคน ตามความต้องการในขณะนั้น เช่น 1, 2, 4 เป็นต้น

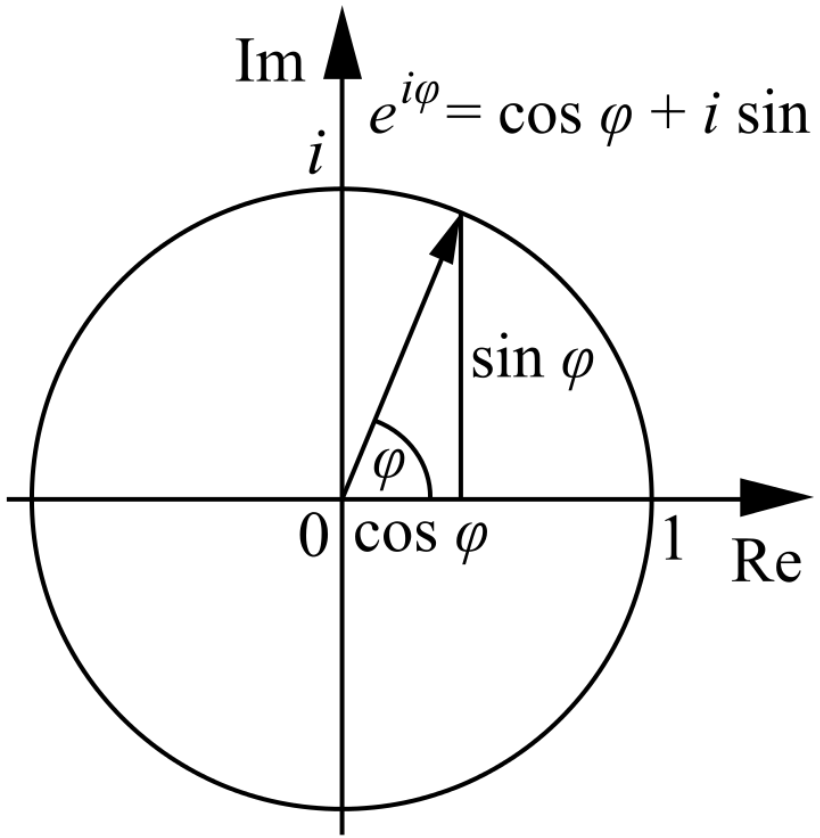
MIMO-MAC is used on the uplink channel to provide multiple access to subscriber stations. In general, MIMO-MAC systems outperform point-to-point MIMO, particularly if the number of receiver antennas is greater than the number of transmit antennas at each user. A variety of multiuser detection techniques are used to separate the signals transmitted by the users.

MIMO-BC is used on the downlink channel to enable the base station to transmit different data streams to multiple users over the same frequency band. MIMO-BC is more challenging to implement. The techniques employed involve processing of the data symbols at the transmitter to minimize interuser interference.

Unit Circle to Sine Wave



Euler's formula

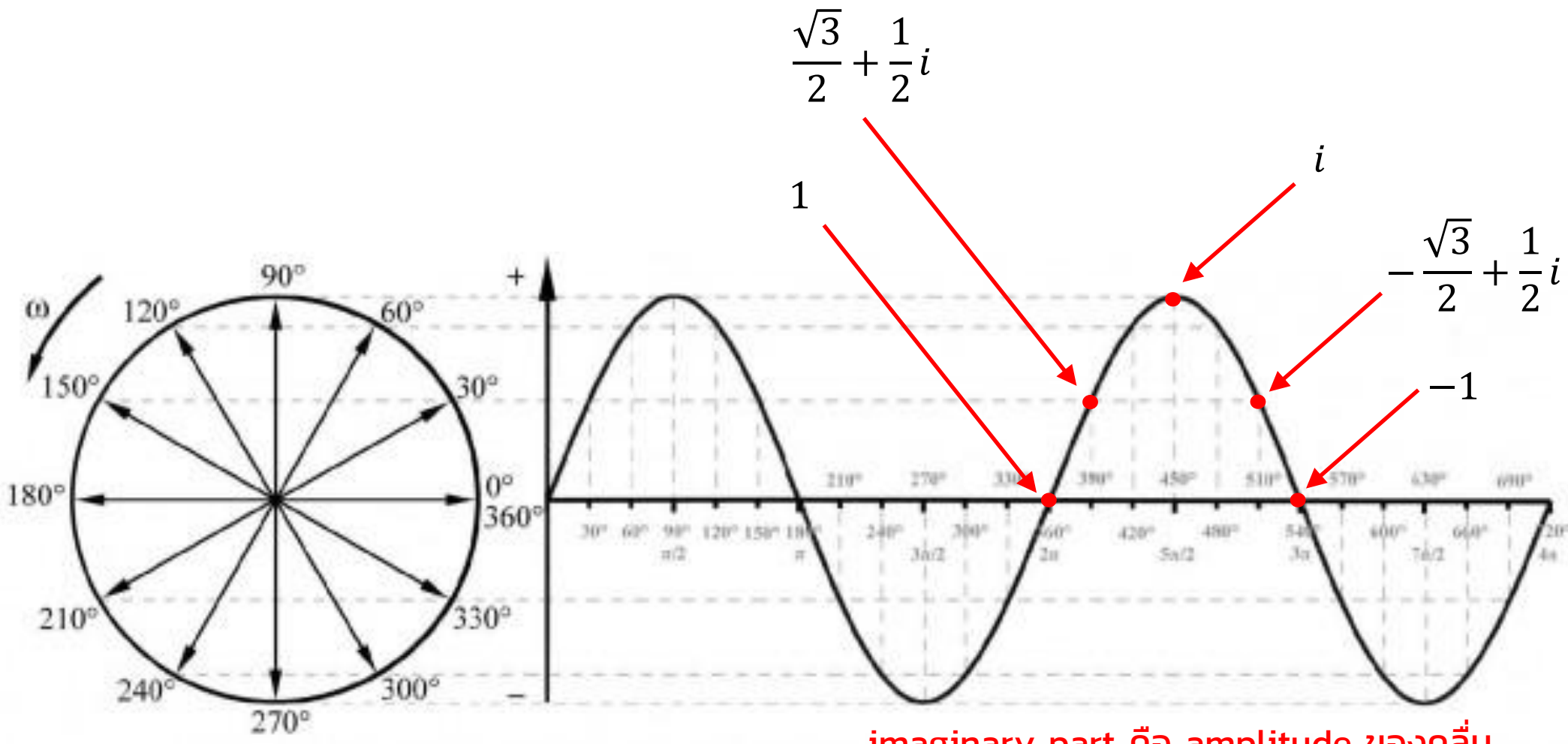


ถ้าจะเลื่อน phase จาก A ไปอีก B ก็คูณกันได้เลย

$$e^{iA}e^{iB} = e^{i(A+B)}$$

หรือคูณด้วย $e^{iB} = \cos B + i\sin B$

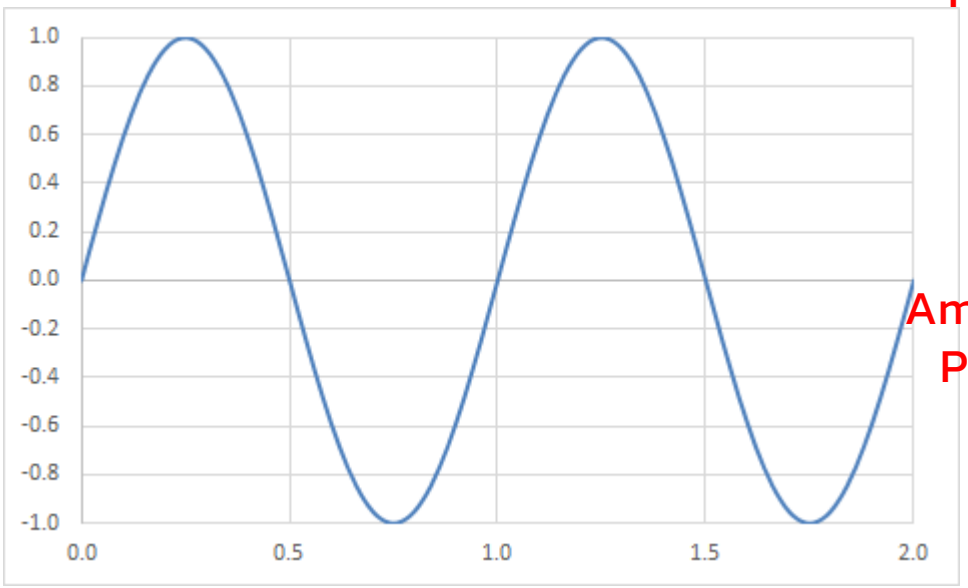
Note: ถ้า A + B > 2π ก็เอาแค่เศษที่เกิน 2π



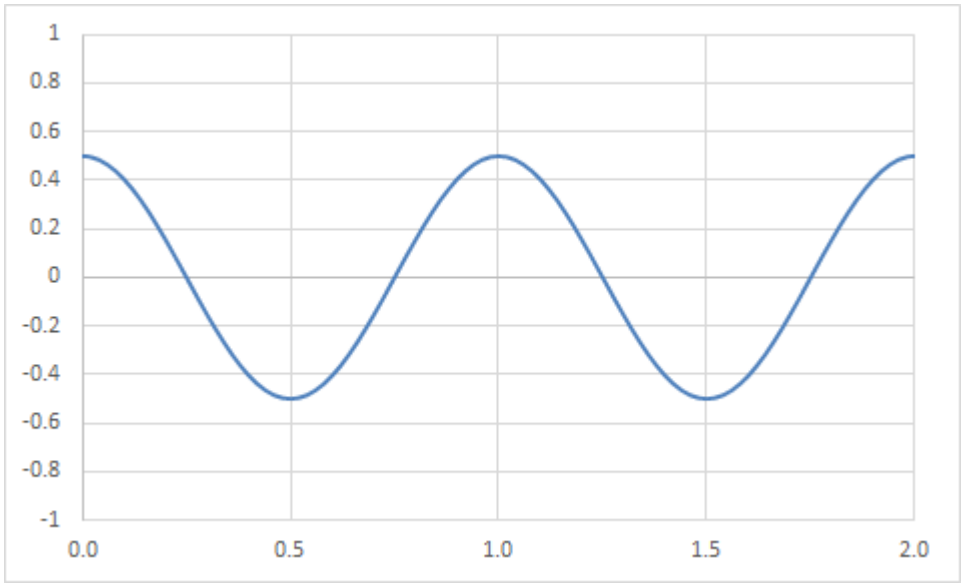
imaginary part คือ amplitude ของคลื่น

#Tx = 1

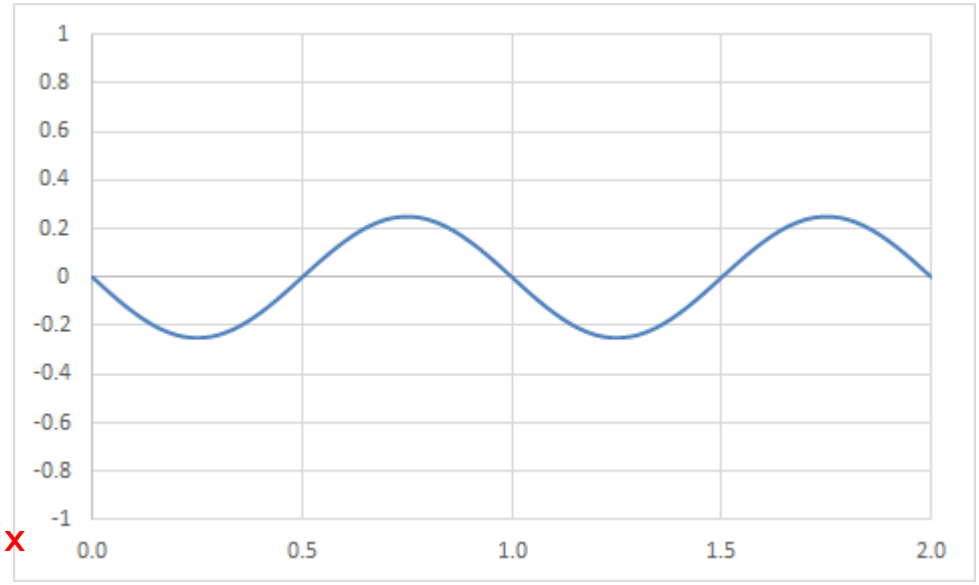
#Rx = 2



Amplitude ลด 2 เท่า
Phase shift +90



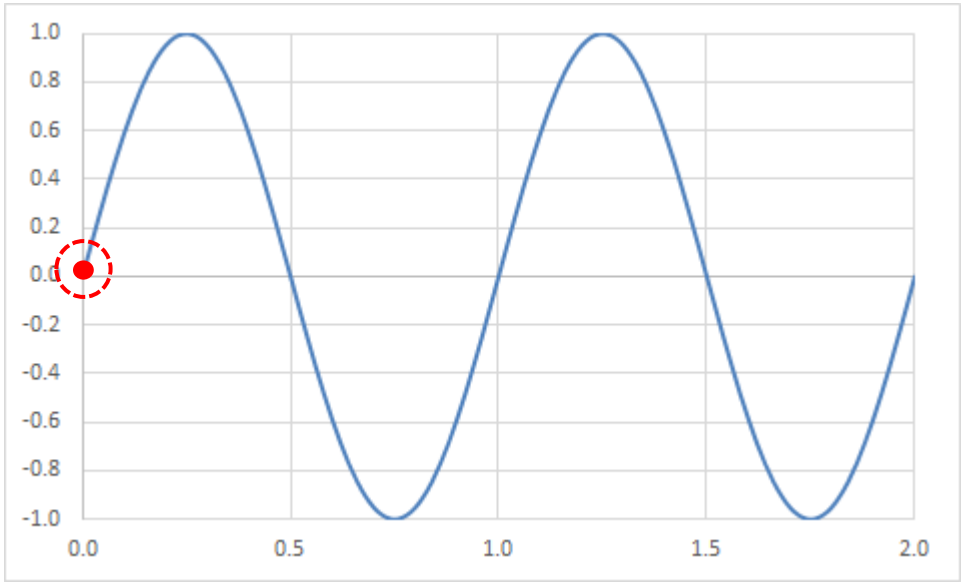
Amplitude ลด 4 เท่า
Phase shift +180

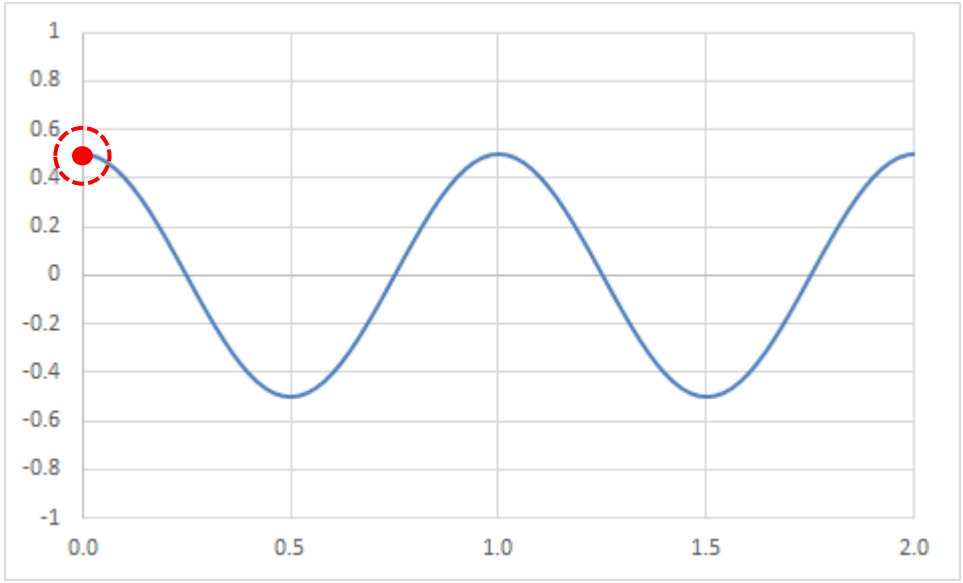


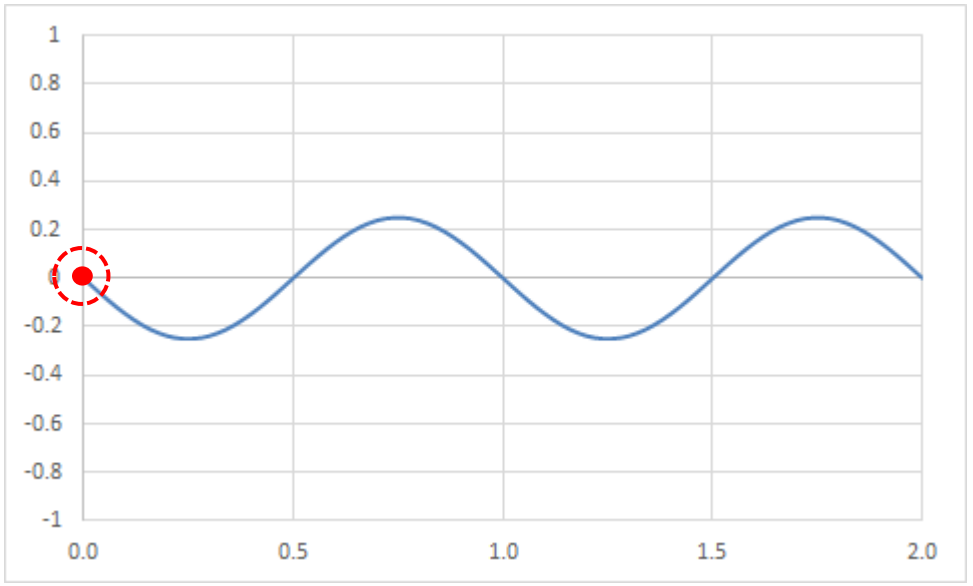
$$H = \begin{bmatrix} i/2 \\ -1/4 \end{bmatrix} \qquad H^{-1} = [-i \quad -2]$$

วิธีปกติหา inverse ได้เฉพาะ square matrix

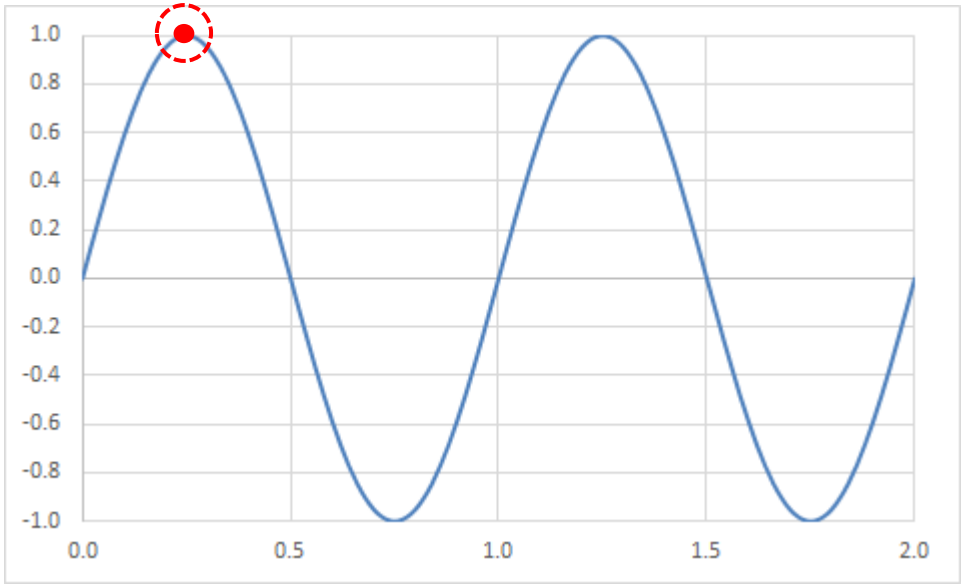
ผู้รับคำนวณ matrix H จาก training data

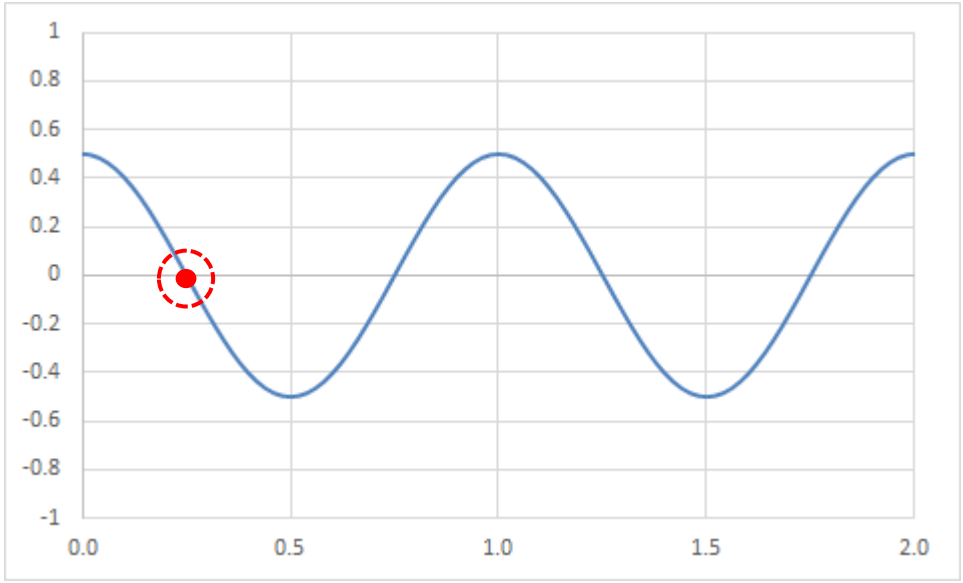


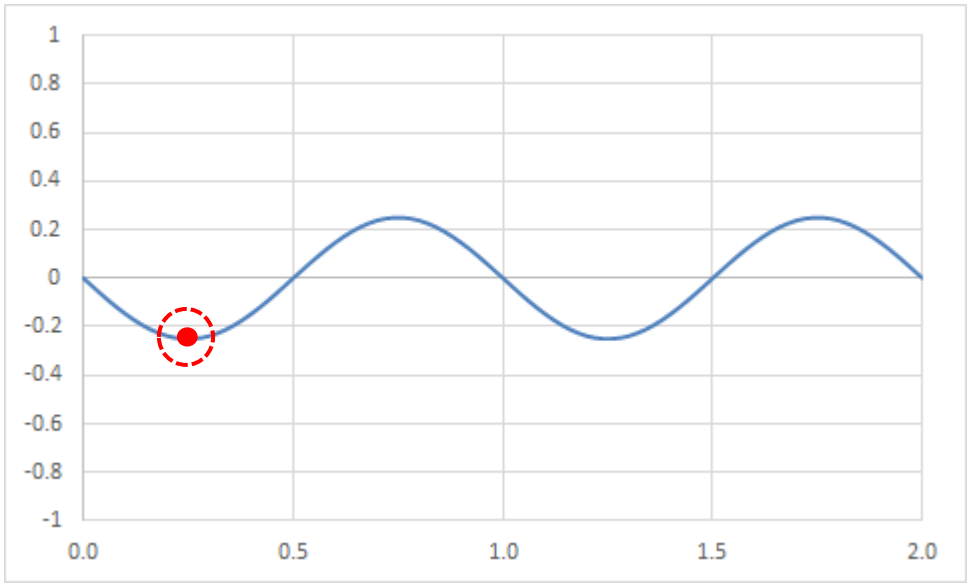




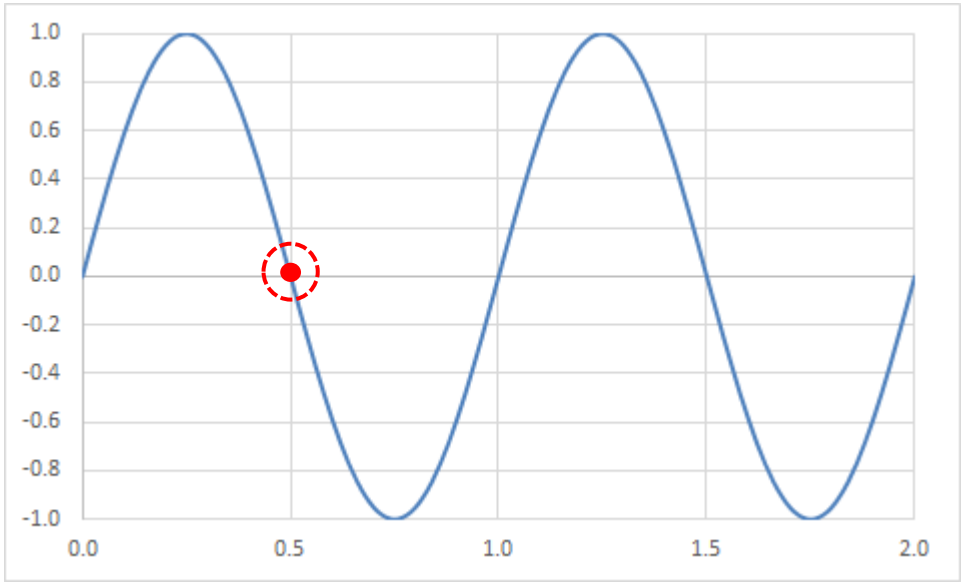
$$\begin{bmatrix} -i & -2 \end{bmatrix} \begin{bmatrix} i/2 \\ -1/4 \end{bmatrix} = 1$$

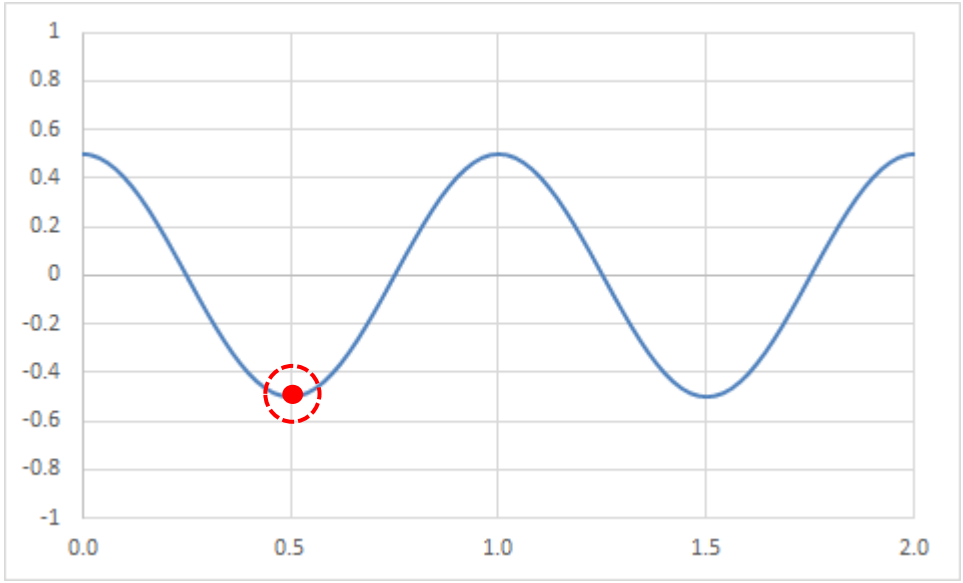


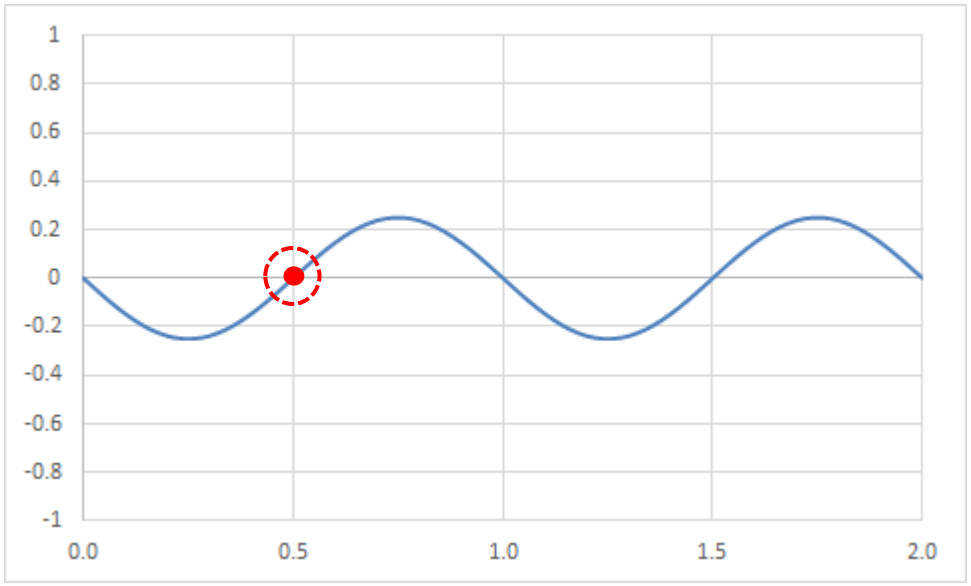




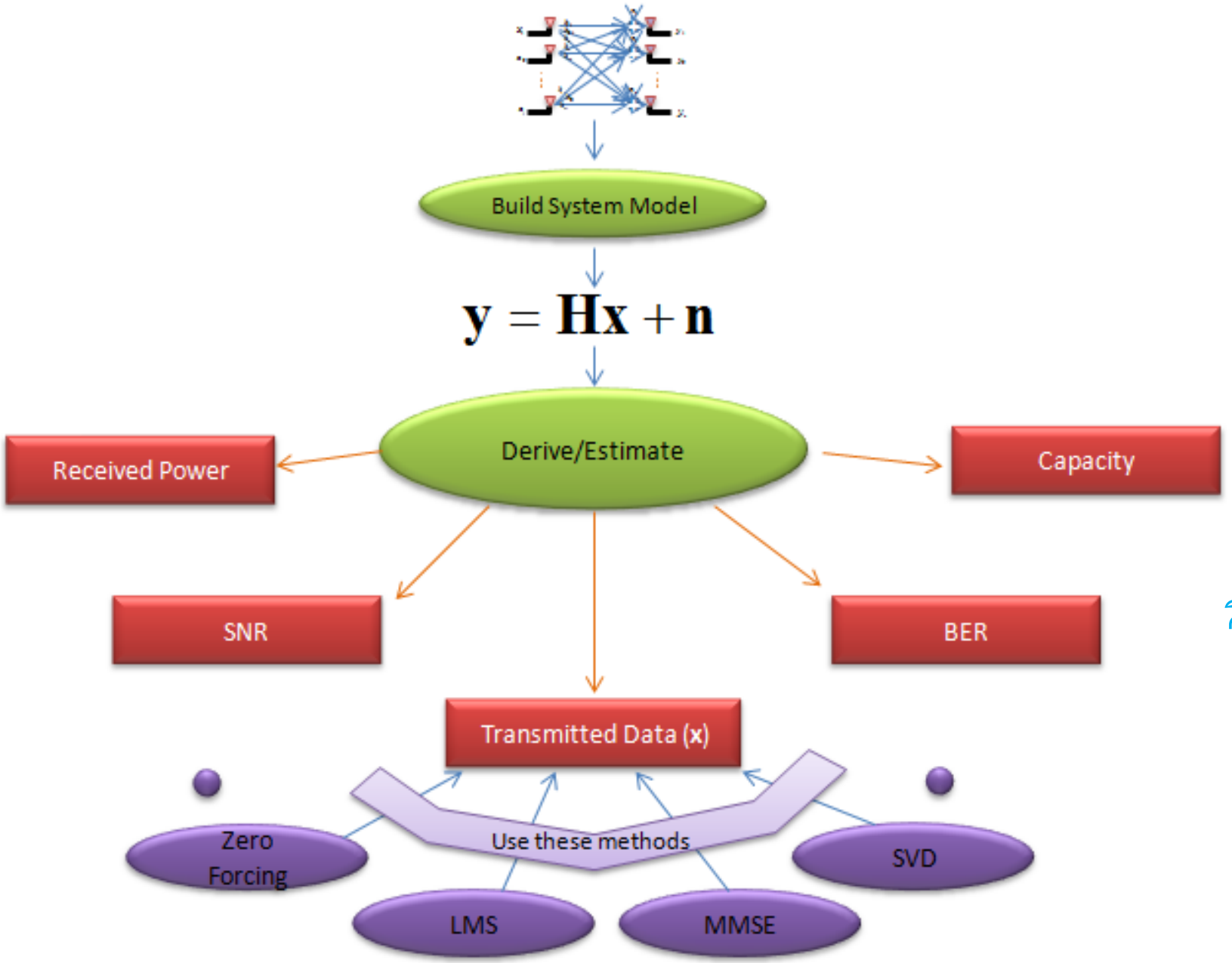
$$\begin{bmatrix} -i & -2 \end{bmatrix} \begin{bmatrix} -1/2 \\ -i/4 \end{bmatrix} = i$$







$$\begin{bmatrix} -i & -2 \end{bmatrix} \begin{bmatrix} -i/2 \\ 1/4 \end{bmatrix} = -1$$

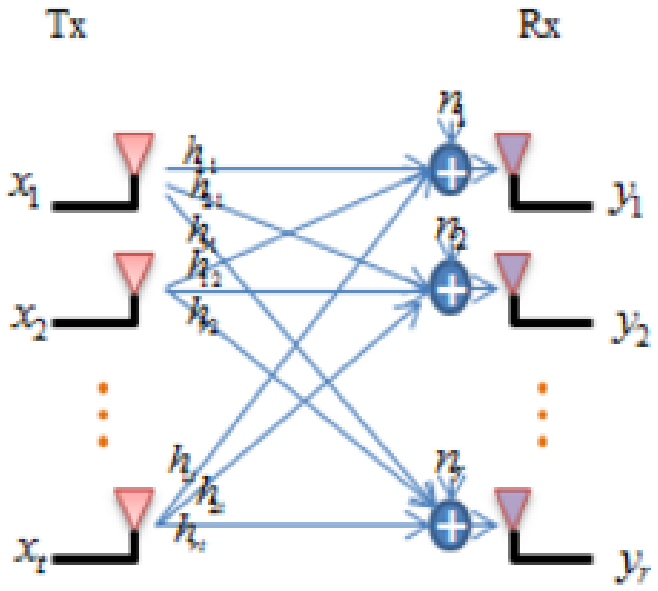
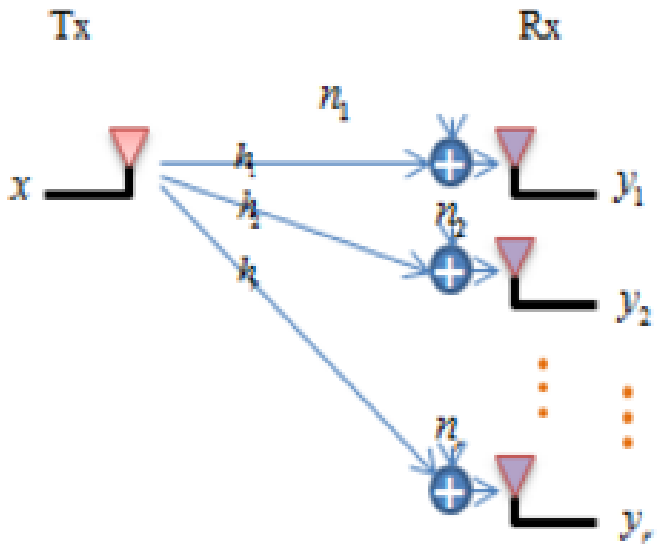
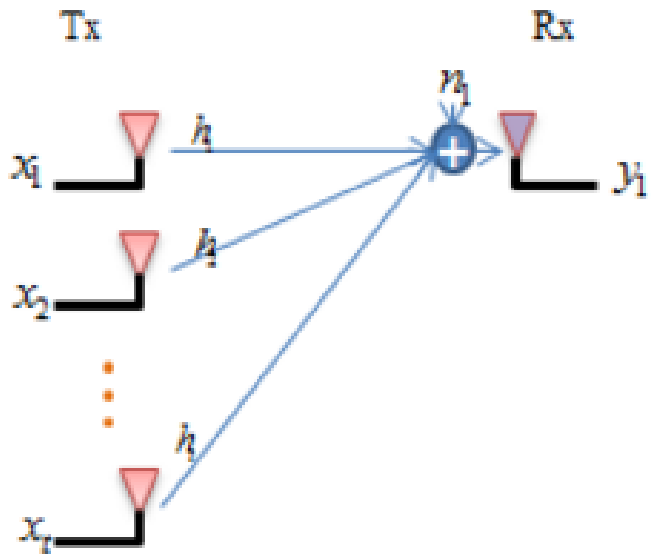


วิธีการคำนวณ

- Zero Forcing
- LMS
- MMSE
- SVD

http://www.sharetechnote.com/html/Communication_ChannelModel.html

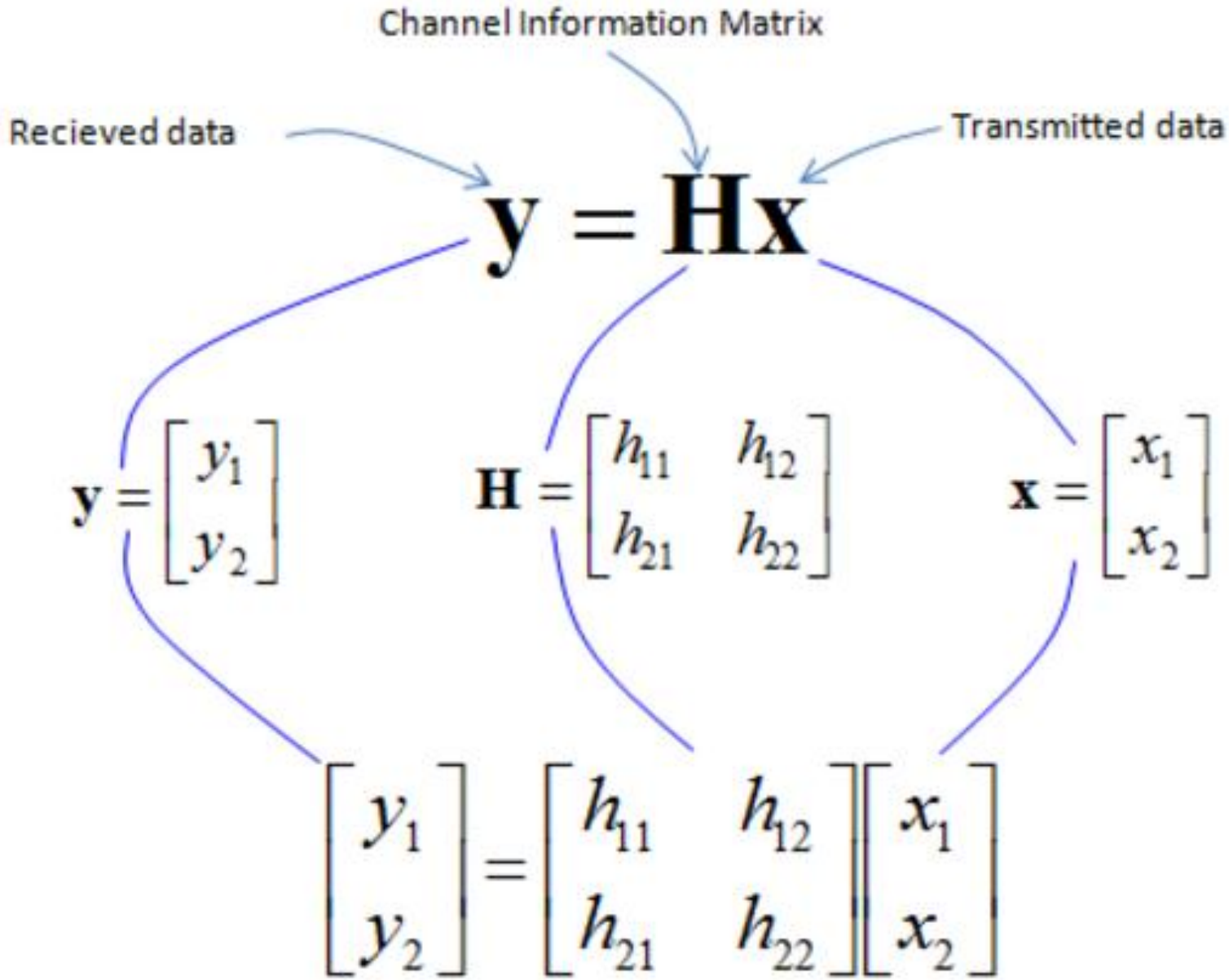
From the Generalized Mode, we can derive roughly three different typical cases as shown below. The details of each of the cases will be explained in separate pages linked below.

		
Tx > 1, Rx > 1 ◦ ZF (Zero Forcing) ◦ ZF by James Weng ◦ MMSE ◦ SVD	Tx = 1, Rx > 1 ◦ Rx Diversity	Tx > 1, Rx = 1 ◦ Tx Diversity ◦ Alamouti Code/OSTBC

Tx > 1 และ Rx > 1

- Same number of Tx and Rx Antenna without Noise
- Same number of Tx and Rx Antenna with Noise
- Different number of Tx and Rx Antenna without Noise
- Different number of Tx and Rx Antenna with Noise

http://www.shaetechnote.com/html/Communication_ChannelModel_ZF.html


Channel Information Matrix
Recieved data
Transmitted data
$\mathbf{y} = \mathbf{H}\mathbf{x}$
$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$
$\mathbf{H} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix}$
$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$
$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$

Channel Information Matrix

(It is assumed that We know this)

Recieved data

(We know this)

$$y = Hx$$

Transmitted data

(This is what we need to figure out)

รู้ H จาก training data

HOW ?

$$\mathbf{y} = \mathbf{Hx}$$

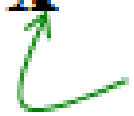
Multiply $\mathbf{H}^{-1}$ on both sides

$$\mathbf{H}^{-1}\mathbf{y} = \mathbf{H}^{-1}\mathbf{Hx}$$

This becomes Identity Matrix

$$\mathbf{H}^{-1}\mathbf{y} = \mathbf{Ix}$$

$$\mathbf{H}^{-1}\mathbf{y} = \mathbf{x}$$



This is the answer that we want to get

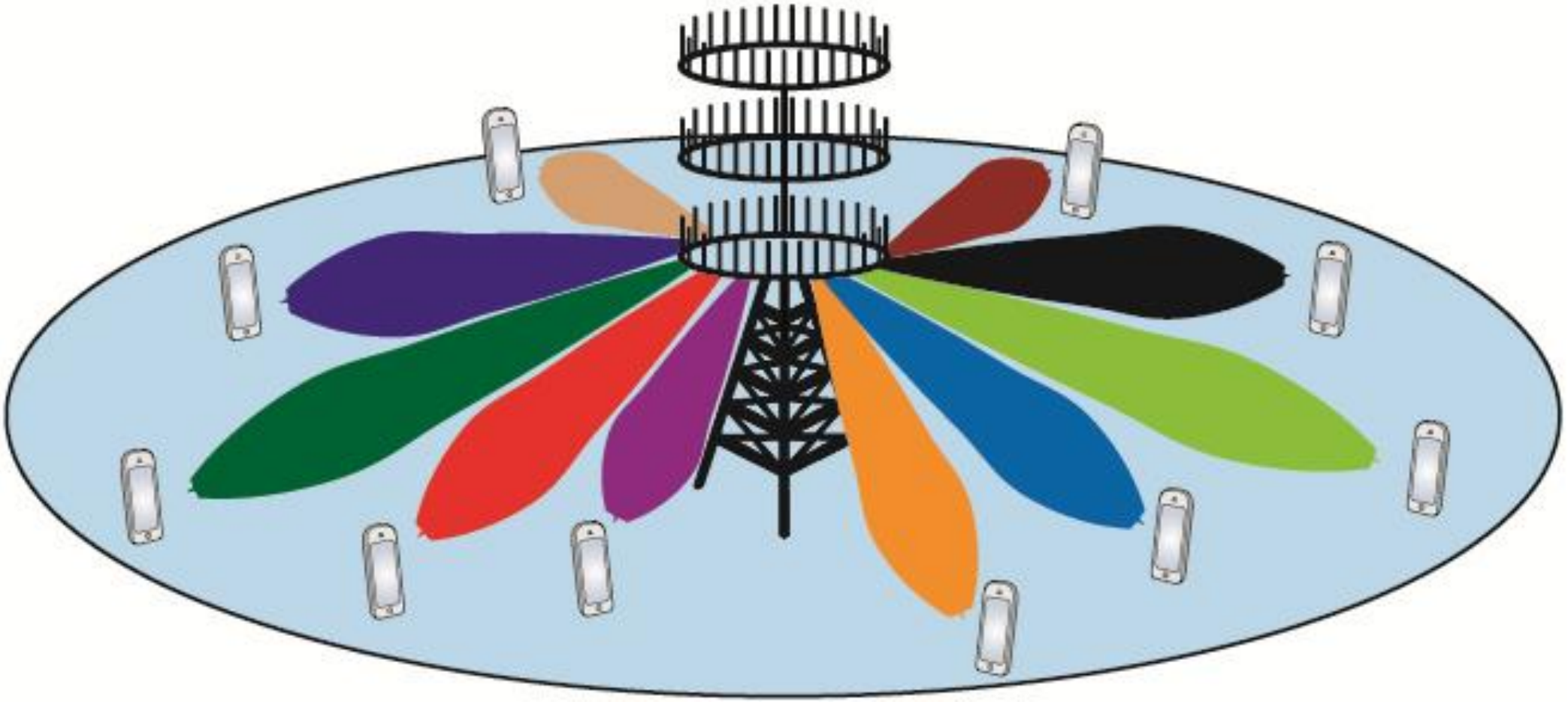
It seems very simple.
But would this always work ?

Unfortunately it is not always working.
It works only with following conditions

$$y = Hx$$

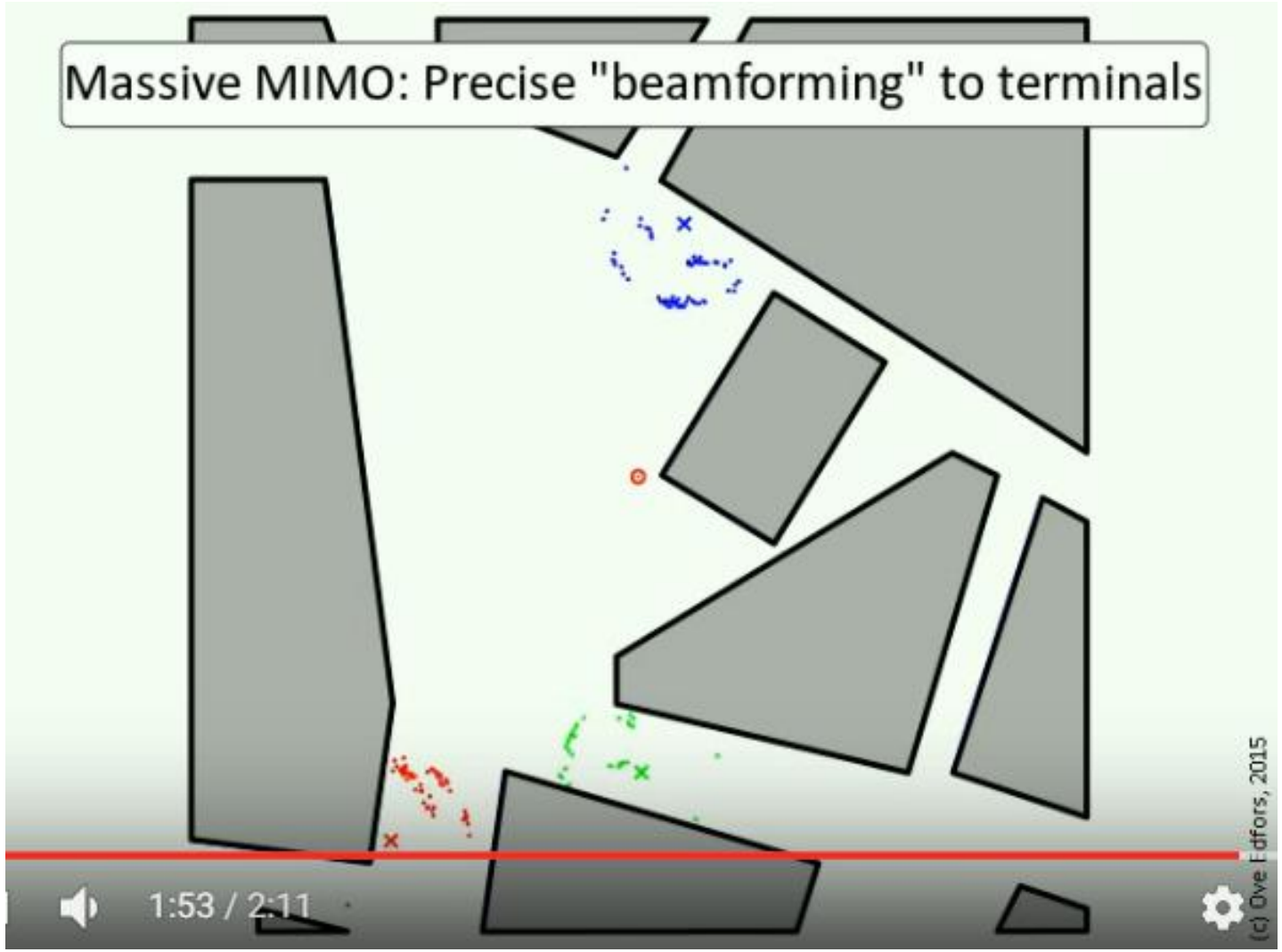
- This should be square matrix → It means the number of Tx antenna and Rx Antenna should be same
- This should be invertible → Determinant of the matrix should not be Zero
  - It should have full Rank (Rank idex is 'n' when it is n x n matrix)

Massive MIMO



Massive MIMO

Massive MIMO



https://www.youtube.com/watch?v=XBb481RNqGw

เสาส่งมี 32 ตัว ทำ 4 x 4 MIMO กับมือถือ (4 Gbit/s)

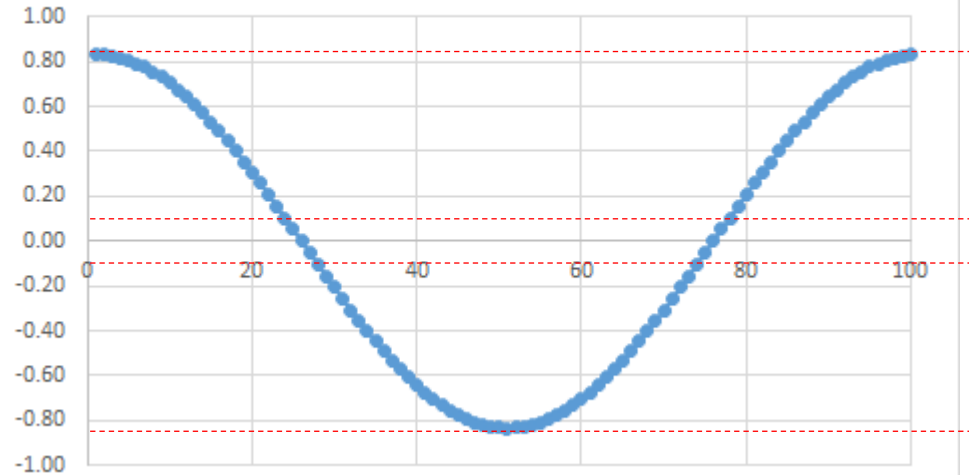
2017

ตัวอย่างข้อสอบ: จงหาว่า Tx1 และ Tx2 ส่งอะไรมา

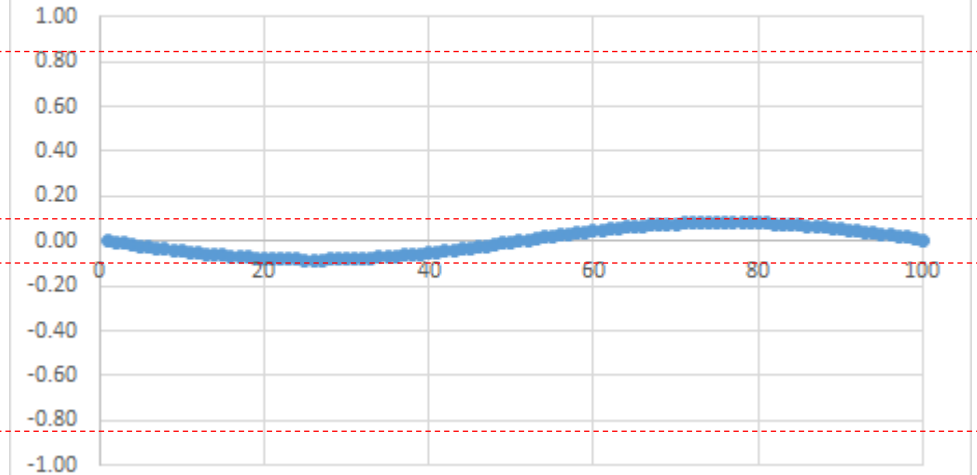
$$H = \begin{bmatrix} 1/2 & i/3 \\ i/4 & 1/6 \end{bmatrix} \qquad H^{-1} = \begin{bmatrix} 1 & -2i \\ -3i/2 & 3 \end{bmatrix}$$

หา inverse ให้แล้ว
ข้อสอบจริงให้แค่ H นะครับ

เฟสเริ่มต้นคือ 90°



เฟสเริ่มต้นคือ 180°



+0.83 หรือ 5/6

+0.083 หรือ 1/12

-0.083 หรือ -1/12

-0.833333 หรือ -5/6

เฉลย: Tx1 ส่ง cos และ Tx2 ส่ง sin ที่มี amplitude 1 หน่วย (ความถี่เท่ากัน)

Excel functions for complex numbers

- complex สร้าง complex number
- improduct หาผลคูณของ complex number
- imsum หาผลรวมของ complex number
- imreal เอาเฉพาะ real part ของ complex number
- Sumproduct หา sum of products
- imaginary เอาเฉพาะ imaginary part ของ complex number

Expo

npm install axios
npm install expo-secure-store
npm install @react-navigation/drawer
npx expo install @react-navigation/drawer react-native-gesture-handler react-native-reanimated

เติมเอพีไอหน้าล็อกอิน ขอ token เก็บไว้ใน secured storage มี toast ใส่ menu

App.js > App

```
import { NavigationContainer } from "@react-navigation/native"
import { createStackNavigator } from "@react-navigation/stack"
import { createDrawerNavigator, DrawerItem } from "@react-navigation/drawer"
import { View, Text } from "react-native"
import * as SecureStore from 'expo-secure-store'

import SignIn from "./screens/SignIn"
import Main from "./screens/Main"
import Credit from "./screens/Credit"

const Stack = createStackNavigator()
const Drawer = createDrawerNavigator()

function CustomDrawerContent(props) {

  const signOut = async () => {

    await SecureStore.deleteItemAsync('token')
    props.navigation.reset({
      index: 0,
      routes: [{ name: 'SignIn' }]
    });
  }

  return (
    <View style={{ paddingTop: 50 }}>
      <DrawerItem label="Main" onPress={() => props.navigation.navigate('Main')}/>
      <DrawerItem label="Credit" onPress={() => props.navigation.navigate('Credit')}/>
      <DrawerItem label="Sign Out" onPress={signOut} labelStyle={{ color: 'red' }}/>
    </View>
  )
}
```

```
  return (
    <View style={{ paddingTop: 50 }}>
      <DrawerItem label="Main" onPress={() => props.navigation.navigate('Main')}/>
      <DrawerItem label="Credit" onPress={() => props.navigation.navigate('Credit')}/>
      <DrawerItem label="Sign Out" onPress={signOut} labelStyle={{ color: 'red' }}/>
    </View>
  )
}

function MainDrawer() {
  return (
    <Drawer.Navigator drawerContent={(props) => <CustomDrawerContent {...props} />}>
      <Drawer.Screen name="Main" component={Main} />
      <Drawer.Screen name="Credit" component={Credit} />
    </Drawer.Navigator>
  )
}

export default function App() {
  return (
    <NavigationContainer>
      <Stack.Navigator>
        <Stack.Screen name="SignIn" component={SignIn} options={{ headerShown: false }}/>
        <Stack.Screen name="MainDrawer" component={MainDrawer} options={{ headerShown: false }}/>
      </Stack.Navigator>

    </NavigationContainer>
  )
}
```

screens > JS SignIn.js > default

```jsx
import React, { useState } from 'react'
import { View, Text, TextInput, Button, Alert } from 'react-native'
import { useNavigation } from "@react-navigation/native"
import axios from 'axios'
import * as SecureStore from 'expo-secure-store'

const SignIn = () => {

  const [nationalId, setNationalId] = useState('')
  const [password, setPassword] = useState('')
  const navigation = useNavigation()

  const submit = async () => {

    if (!nationalId || !password) {

      Alert.alert('Error', 'กรุณากรอกข้อมูลให้ครบถ้วน')
      return
    }

    try {

      const response = await axios.post('http://192.168.1.35:6666/tokens', { nationalId, password })
      await SecureStore.setItemAsync('token', response.data.token)
      navigation.navigate('MainDrawer')

    } catch (error) {

      Alert.alert('Error', error.response.status.toString())
    }
  }
```

```
33    return (
34
35      <View style={{ padding: 20 }}>
36        <Text style={{ fontSize: 24, marginBottom: 20 }}>เข้าสู่ระบบ</Text>
37        <TextInput placeholder="เลขประจำตัวประชาชน" value={nationalId} onChangeText={setNationalId} style={{ borderWidth: 1, padding: 10, marginBottom: 10, borderRadius: 5 }}/>
38        <TextInput placeholder="รหัสผ่าน" value={password} onChangeText={setPassword} secureTextEntry style={{ borderWidth: 1, padding: 10, marginBottom: 10, borderRadius: 5 }}/>
39        <Button title="เข้าสู่ระบบ" onPress={submit} color="#007bff"/>
40      </View>
41    )
42  }
43
44  export default SignIn
```

Main.js ใช้ไฟล์เดิม ไม่ต้องแก้
Credit.js ใช้ไฟล์เดิม ไม่ต้องแก้

The Traditional Approach: npm install

`npm install` is the standard package installation command that most JavaScript developers are familiar with. It:

- Installs the latest version of a package that satisfies your package.json requirements
- Adds the dependency to your package.json file
- Updates your package-lock.json with exact versions
- Works with any Node.js project, not just Expo

Example Usage:

```
npm install  @react-navigation/material-top-tabs
```

The Expo Way: npx expo install

`npx expo install` is Expo's recommended way to install packages in Expo projects. It offers several advantages:

- Automatically selects versions compatible with your Expo SDK version
- Handles peer dependencies more gracefully
- Manages platform-specific dependencies appropriately
- Reduces version conflicts between Expo packages

Example Usage:

```
npx expo install @react-navigation/material-top-tabs
```