A Very Short Introduction to OTFS

(Orthogonal Time Frequency Spacing)

Weijia Zheng

Nov 26, 2024

IERG5110 Course Presentation

Dept. of IE, CUHK

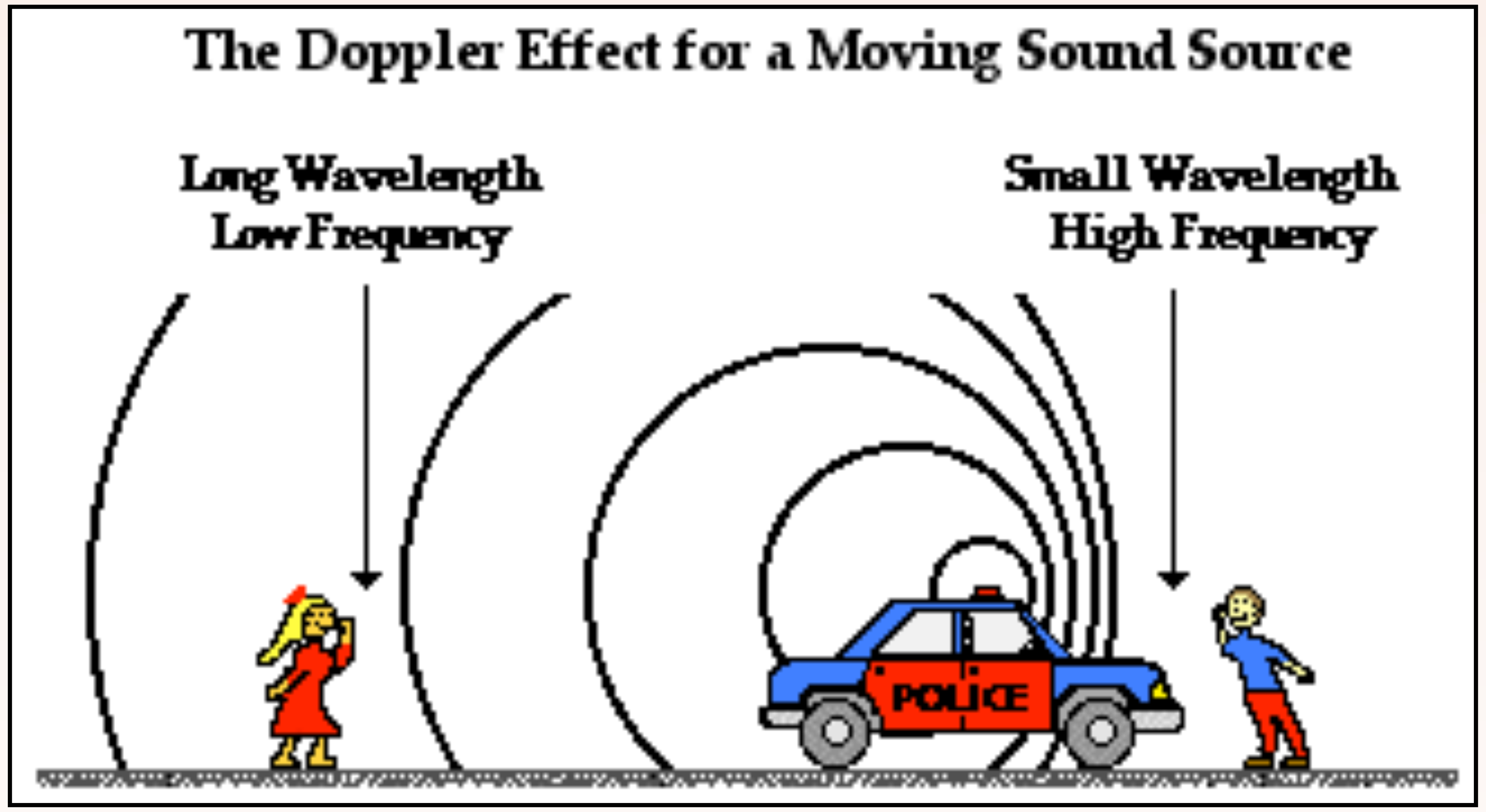
Content

- This < 1 hour talk will mainly focus on the basics of **OTFS (orthogoal time frequency spacing)**, and an inevitable wider concept: “**delay-Dopper communication/(de)modulation**”
  - Including: motivations, some derivations, the mathematical expressions, possibly with some examples.
- I tried to make this talk (and the slides as well) as simple & self-contained as they should be. So we can all learn something. Hopefully to develop some understanding.
- Also, as a humble learner on all the following topics, who knows little on this field. I have limited knowledge of the drawbacks and challenges associated with OTFS modulation.

- Specifically, we want to cover (not in the sequel):
  - **Motivation of OTFS (focus)**
  - **Must-know equations and expressions of OTFS (focus)**
  - **How to do channel estimation and message detection in DD domain (focus).**
  - Comparison between OTFS and OFDM
  - Some wonderful resources to learn OTFS
  - Code implementation for OTFS Tx-Rx and Mod/Demod (Toy example demo)

Delay and "Doppler" (OTFS modulation happens in the "delay-Doppler domain"/DD domain)

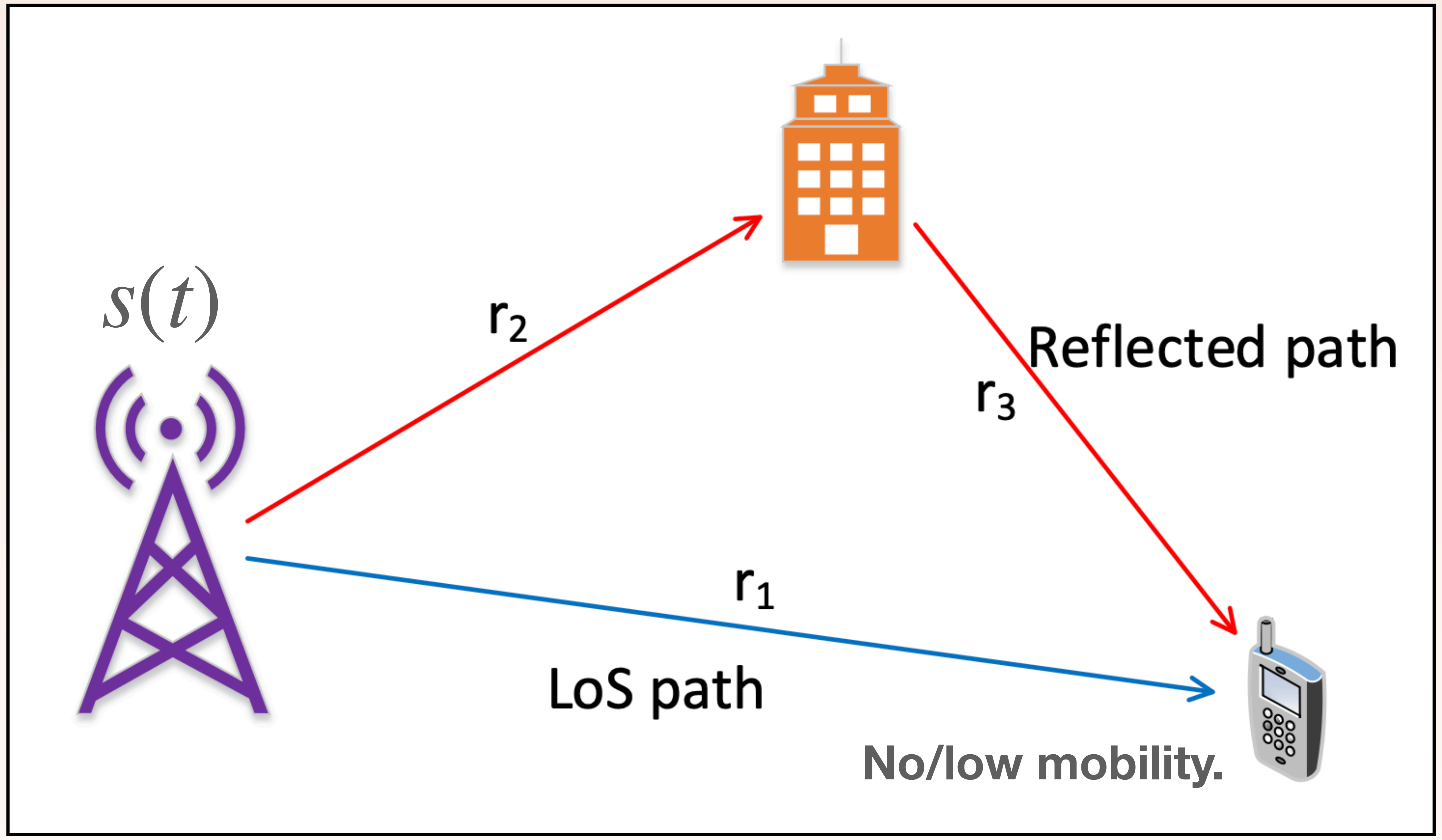
- What is delay? —A meeting is planned to start at 4pm, but eventually started at 5pm. This is called **delay**.
- In some sense, "delay" means the amount of inconsistency / shift in the time domain.



C. Doppler (1803 - 1853)

- What is "Doppler" ("Doppler shift")?
  - — Doppler is a person. :)
  - Something should be f_c Hz, but received as $f_c + f_d$ Hz. This additional amount f_d is termed as "**Doppler**".
  - *" A Doppler shift by f_d Hz is equivalent to modulating/upconverting the transmitted signal with $e^{j2\pi f_d t}$ "*.
  - "Doppler" can be interpreted as: the amount of inconsistency/ shift / difference in the frequency domain.
  - **What brings Doppler? —— Mobility!**

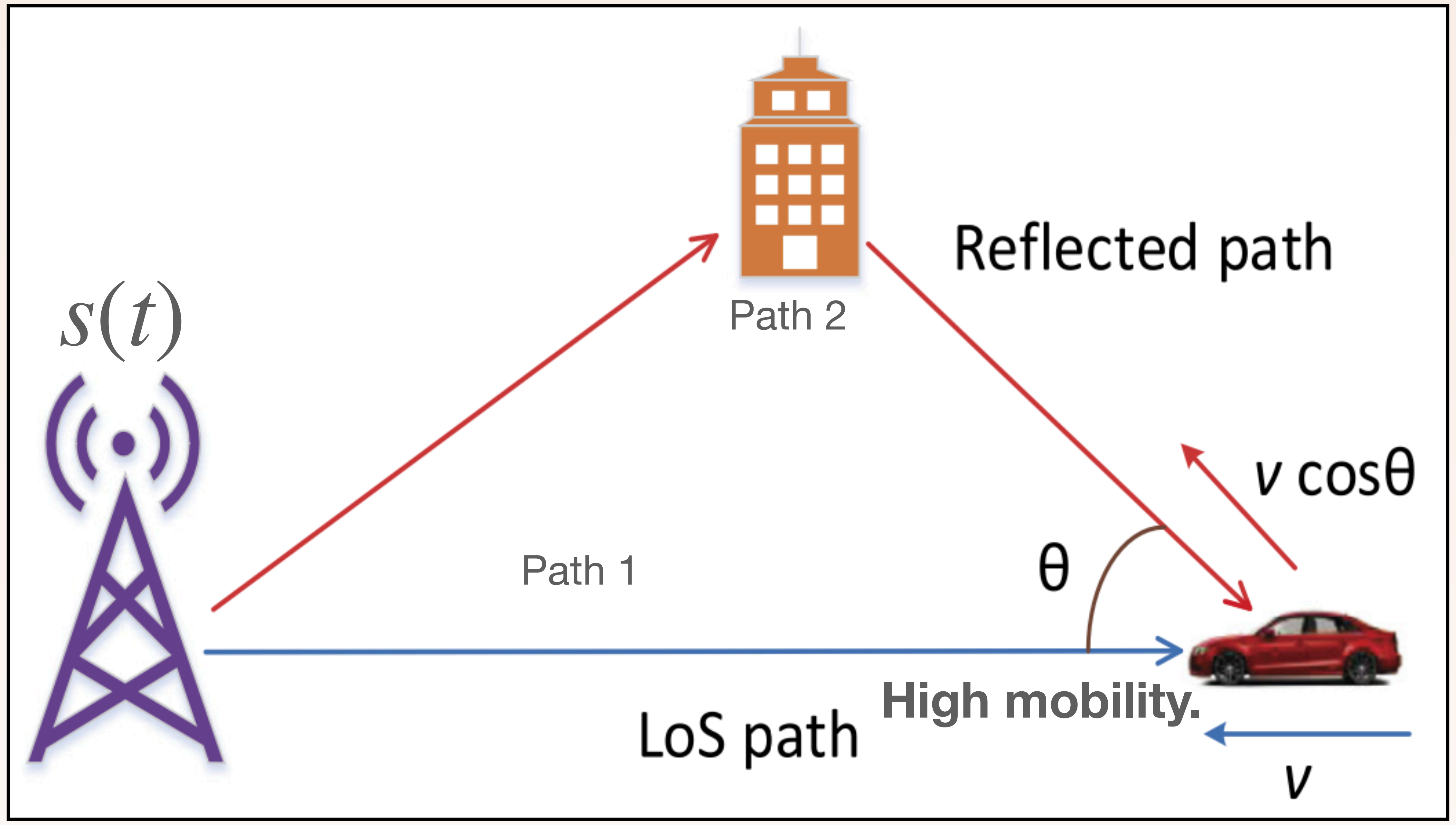
Low-mobility wireless channels



- Delay of LoS path: $\tau_1 = r_1/c$
- Delay of reflected path: $\tau_2 = (r_2 + r_3)/c$
- Delay spread: $\tau_2 - \tau_1$

- Suppose the signal sent by the BS is $s(t)$. Then what is received at the user (the phone) can be expressed as:
- $$r(t) = g_1 s(t - \tau_1) + g_2 s(t - \tau_2),$$
- where g_1, g_2 are (baseband equivalent) complex channel gains. And $c = 3 \cdot 10^8 m/s$ is the speed of light.

High-mobility wireless channels



- Original signal: $s(t)$
- On path $i, i = 1{,}2$ considering the Doppler shift (an additional subcarrier): $s(t)e^{j2\pi\nu_i t}$
- In this example: $\nu_1 = \dfrac{v}{c} f_c$. And $\nu_2 = \dfrac{v\cos\theta}{c} f_c$.
- Consider delay: path i contribute $s(t-\tau_i)e^{j2\pi\nu_i(t-\tau_i)}$

- Hence, finally at the receiver, it receives (no noise yet)
- $$r(t) = g_1 e^{j2\pi\nu_1(t-\tau_1)} s(t-\tau_1) + g_2 e^{j2\pi\nu_2(t-\tau_2)} s(t-\tau_2)$$
- Note that, if there is no mobility $\Longrightarrow$ No Doppler $\Longleftrightarrow$ $\nu_i = 0 \Longrightarrow$
- $g_1 s(t-\tau_1) + g_2 s(t-\tau_2)$ (What we wrote on the last slide)

	$f_c = 2$ GHz	$f_c = 60$ GHz
$\upsilon = 1.5$ m/s $= 5.5$ km/h	$\nu_{\max} = 10$ Hz	$\nu_{\max} = 300$ Hz
$\upsilon = 3$ m/s $= 11$ km/h	$\nu_{\max} = 20$ Hz	$\nu_{\max} = 600$ Hz
$\upsilon = 30$ m/s $= 110$ km/h	$\nu_{\max} = 200$ Hz	$\nu_{\max} = 6$ kHz
$\upsilon = 150$ m/s $= 550$ km/h	$\nu_{\max} = 1$ kHz	$\nu_{\max} = 30$ kHz

Doppler spread for some typical wireless channels.

Let take the last row as an example:

$$\nu_{\max} = \frac{v}{c} f_c = \frac{150}{3\times10^8} 2\times10^9\text{Hz} = 10^3\text{Hz} = 1\text{kHz}$$

Why OFDM is not perfect in high-mobility scenarios

- OFDM stacks its info in frequency domain: $x \in \mathbb{C}^{M\times 1}$. After a $M-$point FFT, it becomes the time domian OFDM (symbol) vector $\tilde{x} = \mathbf{F}_M^{\dagger} x \in \mathbb{C}^{M\times 1}$. (For rectangular pulse-shaping waveforms), we equip the CP part as:

- $$s_{CP} = [\tilde{x}[M - l_{\max}], \ldots, \tilde{x}[M-1]]^T, \text{ and obtain } \underbrace{s' = [s_{CP}^T, \tilde{x}^T]}_{\text{To transmit, length is } M + l_{\max}}.$$

- Suppose the channel is **static**, $(\nu_i = 0)$, then: $h(t) = \sum_{i=1}^{P} g_i \delta(t - \tau_i)$. Discrete-time equivalent channel vector: $h' \triangleq [h_0, \ldots, h_{l_{\max}}]^T$. $h_l = g_i$ if $l = l_i, \forall i = 1,\ldots,P$

- The equivalent baseband signal (including CP) is given by: (In time domain, convolution happens)

- $$\underbrace{r' = h' * s' + w'}_{\text{To receive}} \in \mathbb{C}^{M + l_{\max}}.$$

- Take away the CP, we have $r = \underbrace{[r'[l_{\max}], \ldots, r'[M + l_{\max} - 1]]^T = h \circledast \tilde{x} + w}_{\text{Removed CP, length is } M}$. $h = [h_0, \ldots, h_{\max}, 0, \ldots, 0]^T \in \mathbb{C}^M$.

- Equivalently, we can express $r = \mathbf{H}\tilde{x} + w$ for some appropriate matrix $\mathbf{H}$.

$$\mathbf{H} = \underbrace{\begin{bmatrix} h_0 & 0 & \cdots & 0 & h_{l_{\max}} & h_{l_{\max}-1} & \cdots & h_1 \\ h_1 & h_0 & \cdots & 0 & 0 & h_{l_{\max}} & \cdots & h_2 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ h_{l_{\max}-1} & \ddots & \ddots & \ddots & \ddots & \ddots & 0 & h_{l_{\max}} \\ h_{l_{\max}} & h_{l_{\max}-1} & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & h_{l_{\max}} & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & h_0 & 0 \\ 0 & 0 & \cdots & h_{l_{\max}} & h_{l_{\max}-1} & \cdots & h_1 & h_0 \end{bmatrix}}_{M\times M \text{ Circulant matrix}},$$

$\mathbf{H}$ is a circulant matrix.

Why OFDM is not perfect in high-mobility scenarios

- Equivalently, we can express $r = \mathbf{H}\tilde{x} + w$

$$\mathbf{H} = \sum_{i=1}^{P} g_i \Pi^{l_i}\, \mathbf{D}^{k_i}, \qquad \mathbf{D} \triangleq \mathrm{diag}(1, \omega, \ldots, \omega^{M-1})$$

$$\mathbf{H} =$$

$$\mathbf{H} = \sum_{i=1}^{P} g_i \Pi^{l_i} \quad \underbrace{\begin{bmatrix} h_0 & 0 & \cdots & 0 & h_{l_{\max}} & h_{l_{\max}-1} & \cdots & h_1 \\ h_1 & h_0 & \cdots & 0 & 0 & h_{l_{\max}} & \cdots & h_2 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ h_{l_{\max}-1} & \ddots & \ddots & \ddots & \ddots & \ddots & 0 & h_{l_{\max}} \\ h_{l_{\max}} & h_{l_{\max}-1} & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & h_{l_{\max}} & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & h_0 & 0 \\ 0 & 0 & \cdots & h_{l_{\max}} & h_{l_{\max}-1} & \cdots & h_1 & h_0 \end{bmatrix}}_{M \times M \text{ Circulant matrix}},$$

Static channel (we saw on the last page)

- Note that on the LHS, **H** *is a circulant matrix: which can be diagonalized via Fourier transform matrices pairs* (DFT matrices form a orthonormal basis of circulant matrices)

- $\mathbf{H} = \mathbf{F}_M^{\dagger} \mathrm{diag}(\mathbf{F}_M h) \mathbf{F}_M.$

$$\Pi = \begin{bmatrix} 0 & \cdots & 0 & 1 \\ 1 & \ddots & 0 & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 1 & 0 \end{bmatrix}$$

- $y = \mathbf{F}_M r = \hat{h} \odot \tilde{x} + \hat{w}$

- Where $\odot$ denotes element-wise product. Such orthogonality enables us to use **single-tap equalizer** to estimate $\tilde{x}$ **entry-by-entry.**

- How to understand this? —— Each path $i \in [P]$. The signal is first "modulated" by a carrier k_i. Then shifted by Π^{l_i}. Then they composite together to make the received r, with noise.

$$= \begin{bmatrix} h_0 & 0 & \cdots & h_{l_{\max}}\omega^{k_P(M-l_{\max}))} & \cdots & h_1\omega^{k_1(M-1)} \\ \vdots & h_0\omega^{k_0} & \cdots & \ddots & \cdots & \vdots \\ \vdots & \ddots & \ddots & 0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & & h_{l_{\max}}\omega^{k_P(M-1)} \\ h_{l_{\max}} & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & h_{l_{\max}}\omega^{k_P} & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & h_{l_{\max}}\omega^{k_P(M-l_{\max}-1)} & \cdots & \cdots & h_0\omega^{k_0(M-1)} \end{bmatrix}$$

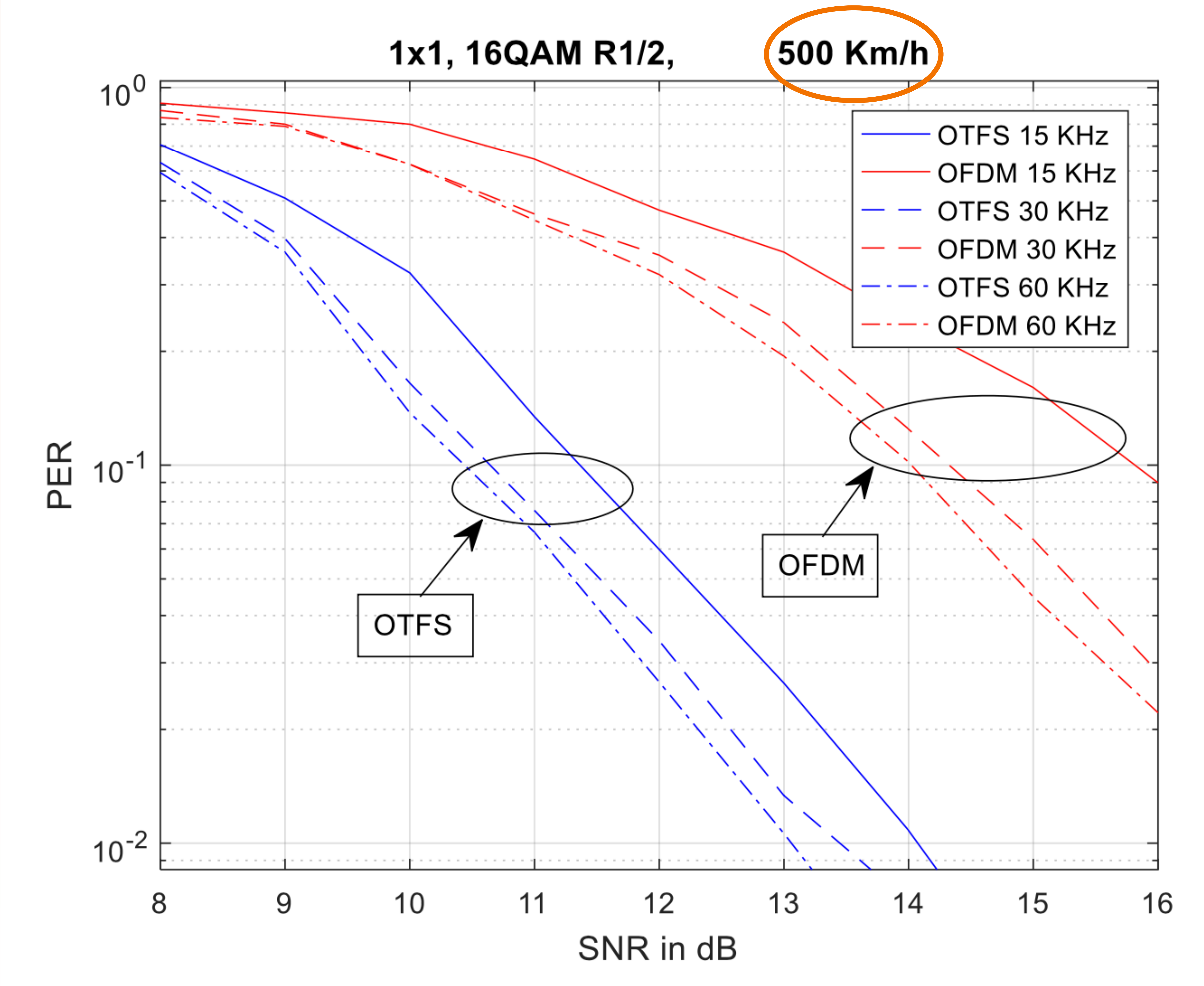
High-mobility channel

- However, on the RHS, there are some additional $\omega^{\cdots}$ factors, caused by Doppler shifts, makes this matrix no longer circulant. $\Longrightarrow$ cannot be eigen-decomposited via FFT matrices!!! (Where $\omega = e^{\frac{j2\pi}{M}}$)

  - Continuous case: $s(t) \to s(t)e^{j2\pi f_m t}$. Discrete case: $s[m] \to s[m]e^{j2\pi k_i m}$

- $\Longrightarrow$ Gives us performance degradation. If we do LHS decoding anyway $\Longrightarrow$ need to endure some ICI. If those $k_m = 0$, then RHS $\rightarrow$LHS.

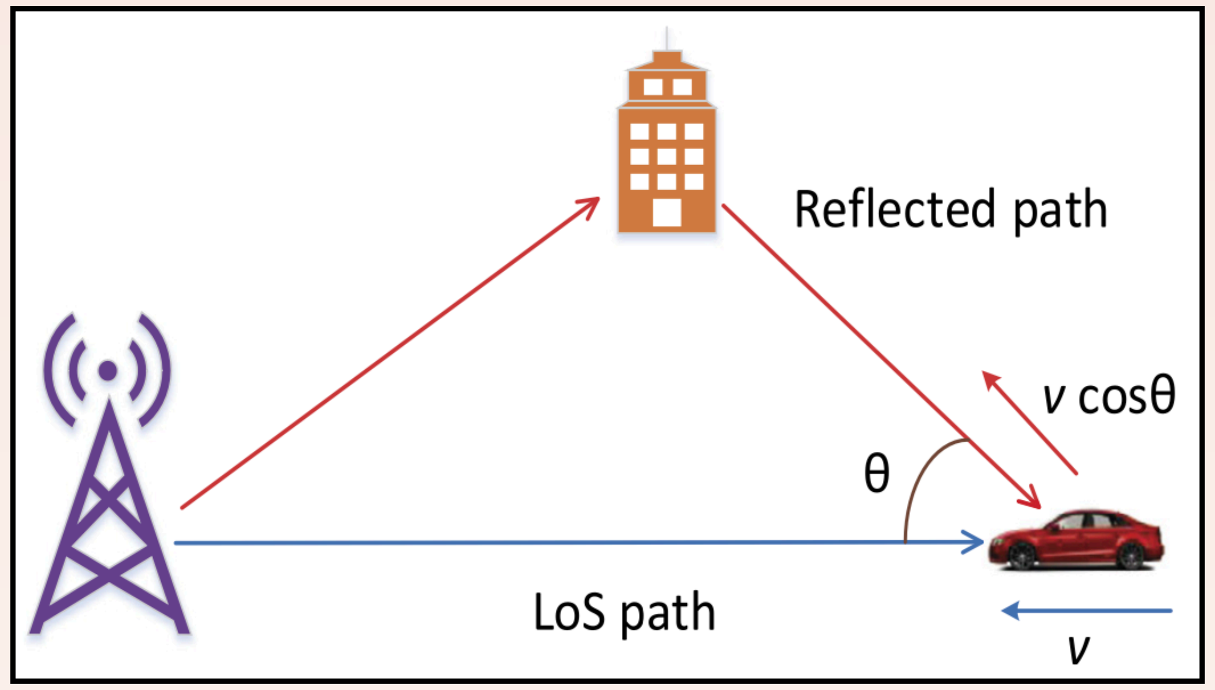
With Dopper, OTFS outperforms OFDM



This plot should be self-explanatory.

copied from page 20 of [2]

High-mobility wireless channels



$g_1 e^{j2\pi\nu_1(t-\tau_1)} s(t-\tau_1) + g_2 e^{j2\pi\nu_2(t-\tau_2)} s(t-\tau_2)$

- The 2-path result can be generalized for P paths as the following:
- $$r(t) = \sum_{i=1}^{P} g_i e^{j2\pi\nu_i(t-\tau_i)} s(t-\tau_i) = \sum_{i=1}^{P} g_i e^{j2\pi\nu_i t} s(t-\tau_i) e^{-j2\pi\nu_i\tau_i}$$
- We define the delay-Doppler channel representation as:
- $$h(\tau,\nu) \triangleq \sum_{i=1}^{P} g_i e^{-j2\pi\nu_i\tau_i} \delta(\tau-\tau_i)\delta(\nu-\nu_i)$$ [this function only has value on (τ_i, ν_i)]
- One can see the received signal $r(t)$ as:
- $$r(t) = \int\int h(\tau,\nu) e^{j2\pi\nu t} s(t-\tau) d\nu d\tau$$
- **The cool thing here is:** the channel $h(\tau,\nu)$ (in DD domain) is completely represented by these quantities: $(g_i, \tau_i, \nu_i), i = 1,2,\ldots,P$. The # of quantities you need is small $\Longrightarrow$ sparsity.
- We will relate the above new channel representation with something we knew.

Different (Linear time variant) channel representations

- $$h(\tau,\nu) \triangleq \sum_{i=1}^{P} g_i e^{-2\pi\nu_i\tau_i}\delta(\tau-\tau_i)\delta(\nu-\nu_i) \quad \text{(delay-Doppler)}$$
  This is the delay-time domain representation of the channel.

- If we go back one step: $r(t) = \sum_{i=1}^{P} g_i e^{j2\pi\nu_i(t-\tau_i)} s(t-\tau_i)$

- $$r(t) = \sum_{i=1}^{P} \underbrace{g_i e^{j2\pi\nu_i(t-\tau_i)}}_{g(\tau_i,t)} s(t-\tau_i) \implies \text{generalize} \int_0^\infty \underbrace{g(\tau,t)}_{\text{Delay-time domain}} s(t-\tau)d\tau.$$

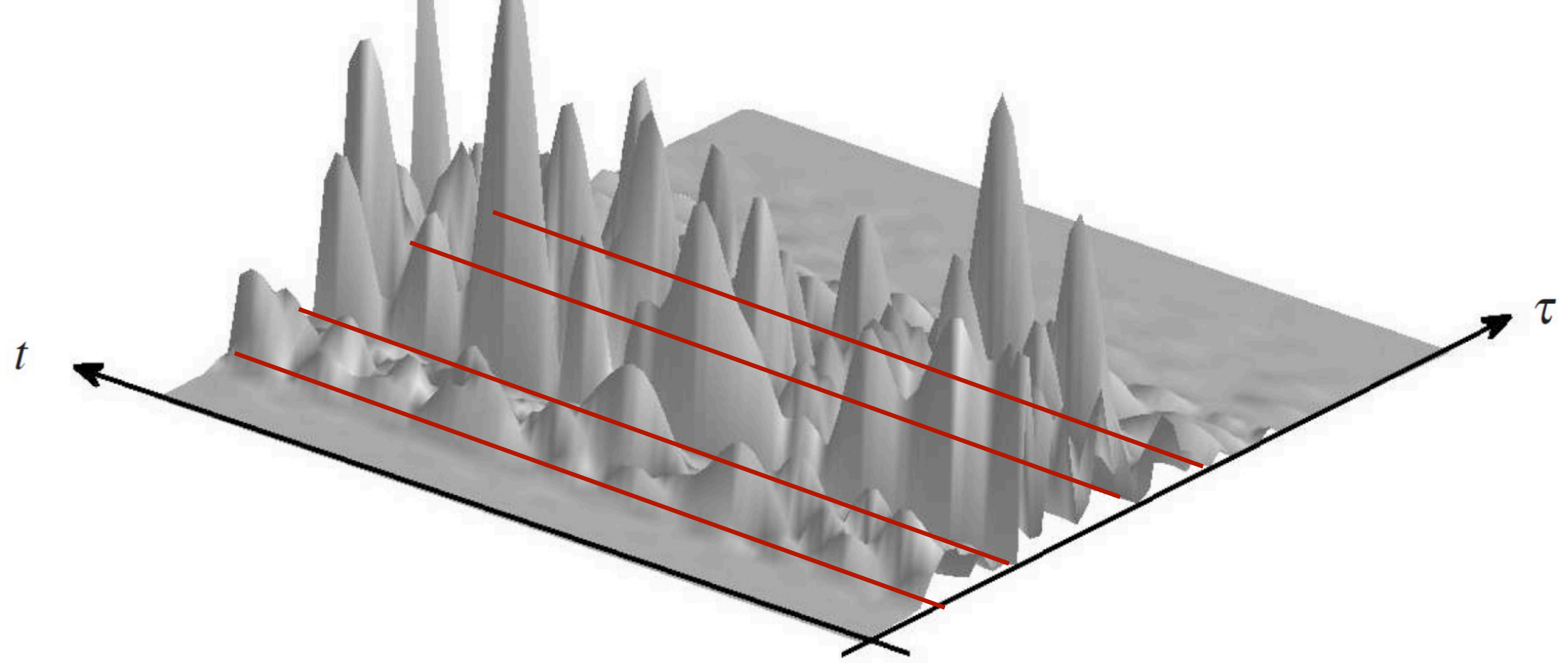
- Another immediate equivalent representation is the time-frequency representation:

- $$r(t) = \sum_{i=1}^{P} \underbrace{g_i e^{j2\pi\nu_i(t-\tau_i)}}_{g(\tau_i,t)} s(t-\tau_i) \implies \int_0^\infty g(\tau,t)s(t-\tau)d\tau = \int \underbrace{H(f,t)}_{\text{Time-frequency domain}} S(f)e^{j2\pi ft}df$$

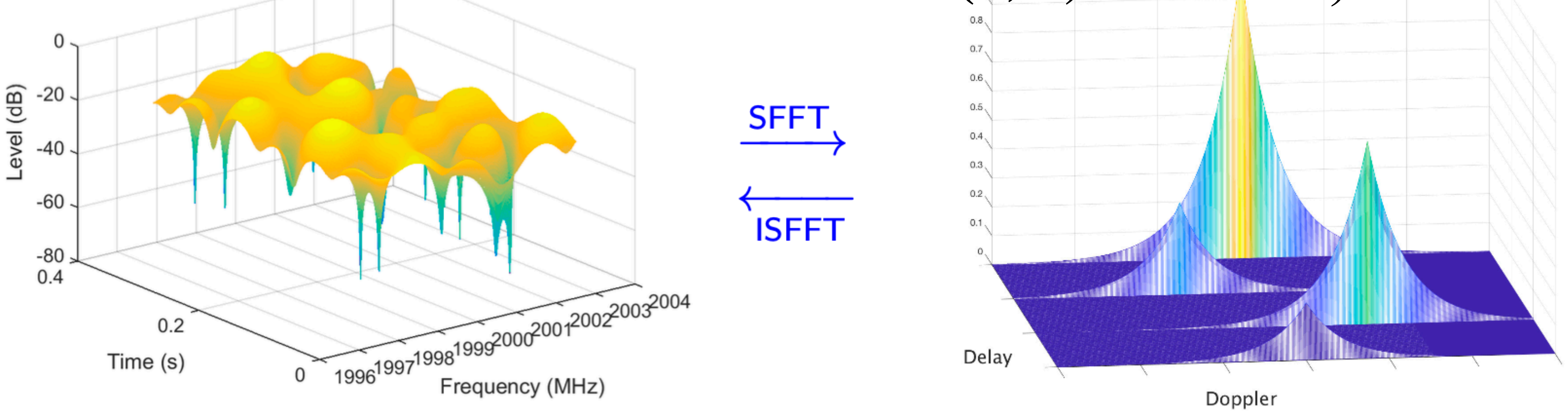
- The last step can be understood as consider product in frequency domain, then use IFT back to time domain.

Time-variant impulse response $g(t,\tau)$

- $g(\tau, t)$ be like :(

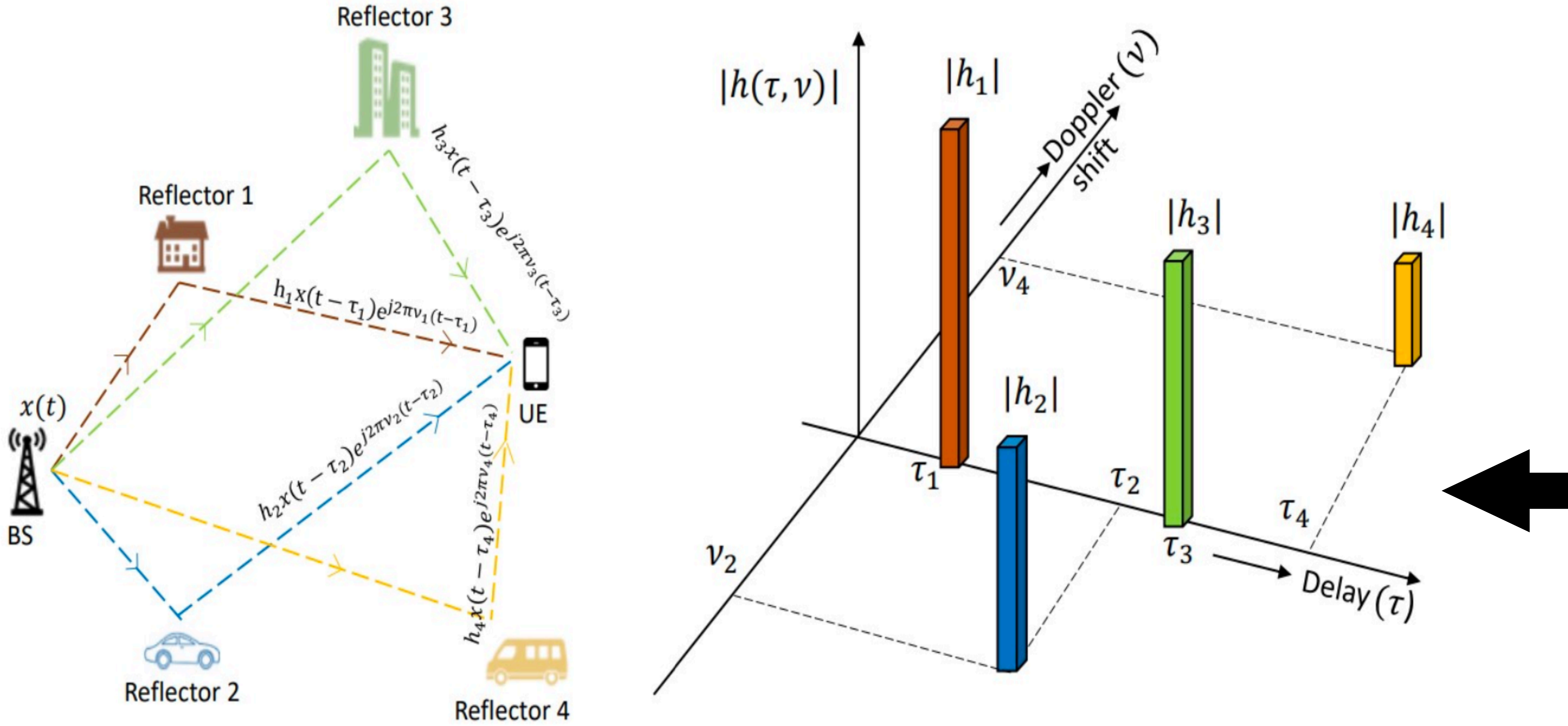


- $h(\tau, \nu)$ be like :)



- $H(t, f)$ be like :(

Channel in Time–frequency $H(t,f)$ and delay–Doppler $h(\tau,\nu)$

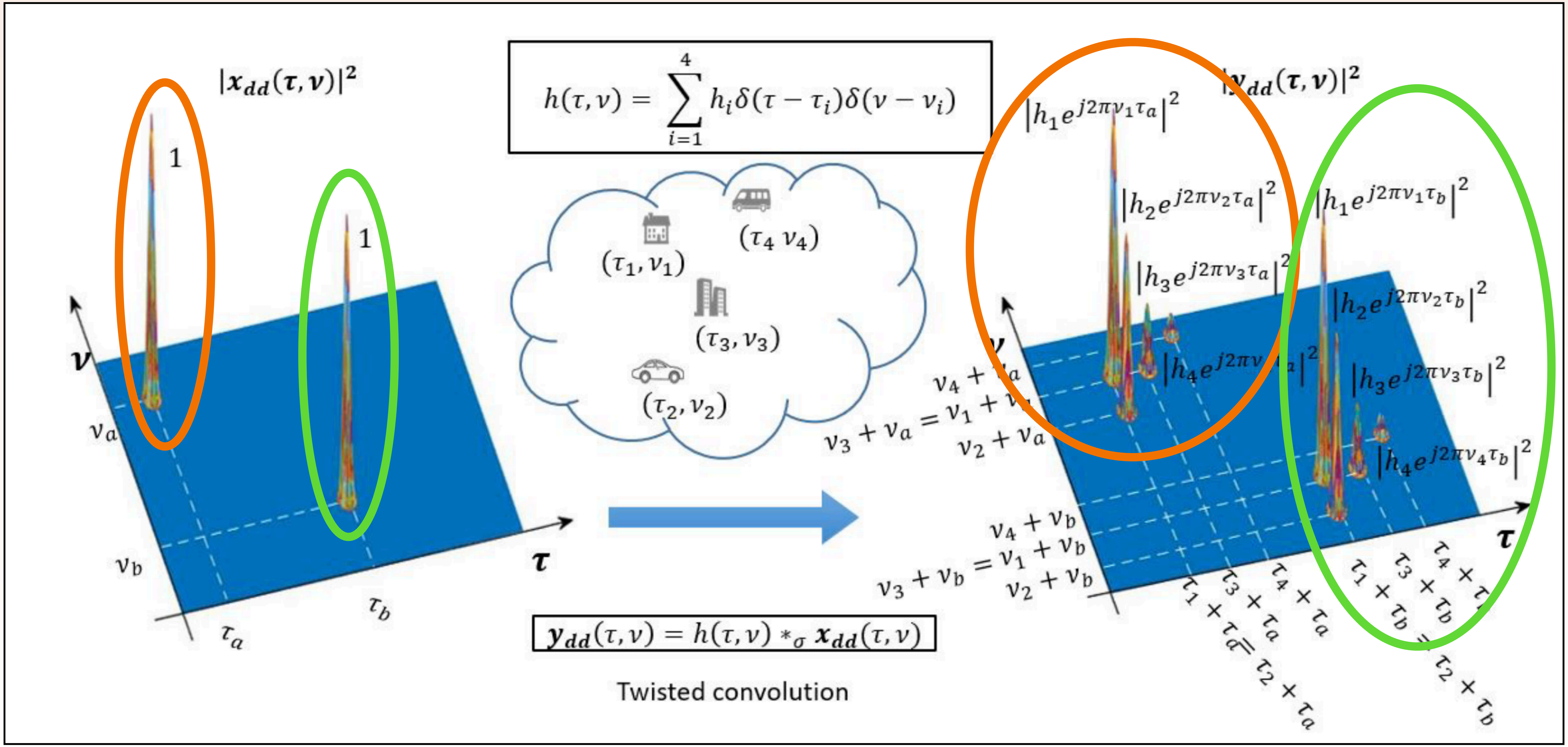


- DD domain notation sparsify the channel representation!!

- The channel representation should be simple as this!!!

TX and RX signal in *delay-Dopper domain*

- OK. But does the DD domain stuff ever make our lives easier?? —— Probabily, yes.



- We will come back to this later. Now just get some intuitive feeling of how signals will interact with the channel in DD domain.

Parameters and their choice in OTFS

- In an OTFS system, we need to specify a bunch of parameters.
- We assume the OTFS system operates on a P-path high mobility channel, with a bandwidth B.
- Maximum delay spread $\tau_{\max}$, maximum Doppler shift $\nu_{\max}$.
- Sampling frequency $f_s = B = \dfrac{1}{T_s}$, where T_s is referred as "sampling interval".
- Each OTFS frame contains NM samples divided into N time slots. M samples per block.
- OTFS frame duration $T_f = NMT_s = NT$. $T = MT_S$ denotes the duration of each block.
- $\Delta f = \dfrac{1}{T}$. This is the "space of spectrum samples".

OTFS modulation

- In OFDM, we store our information in frequency domain vectors. In OTFS, we place them in **"delay-Doppler"** domain matrix $\mathbf{X} \in \mathbb{C}^{M\times N}$, with entries $\mathbf{X}[m,n], m \in [M], n \in [N]$.

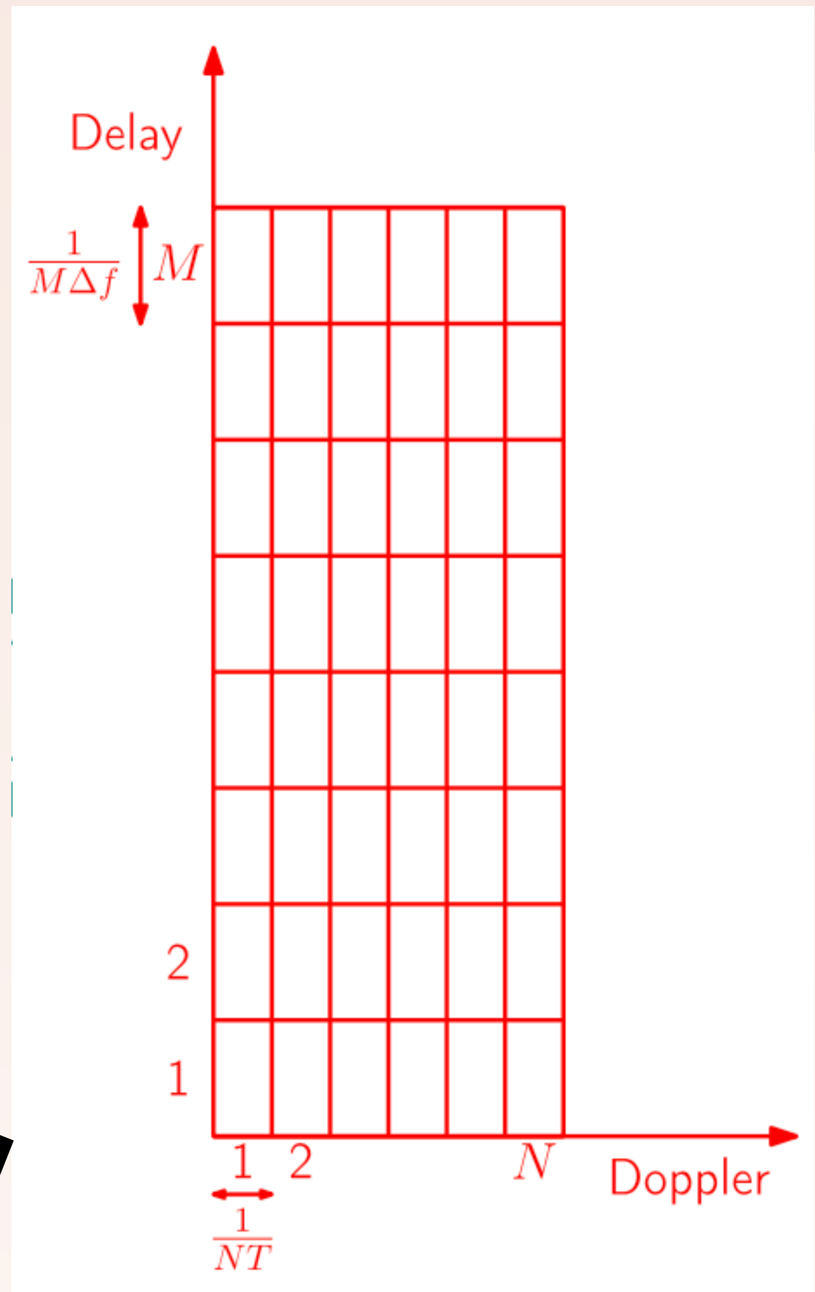
- We first convert $\mathbf{X}$ into frequency-time domain $\mathbf{X}_{ft} \in \mathbb{C}^{M\times N}$ by applying (N-pt) IFFT along the rows of $\mathbf{X}$, then apply (M-pt) FFT on columns of it. (Can we swap the order? —— yes) Put it in math, we have:

- $$\mathbf{X}_{ft} = \frac{1}{\sqrt{MN}} \sum_{n=0}^{N-1}\sum_{m=0}^{M-1} \mathbf{X}[m,n] e^{j2\pi(\frac{nk}{N}-\frac{ml}{M})} = \mathbf{F}_M \underbrace{\mathbf{X}}_{M\times N} \mathbf{F}_N^H$$ (this is called "ISFFT" inverse symplectic FFT)

- Then to convert the 2D $\mathbf{X}_{ft}$ to a time domain signal, we do the following:

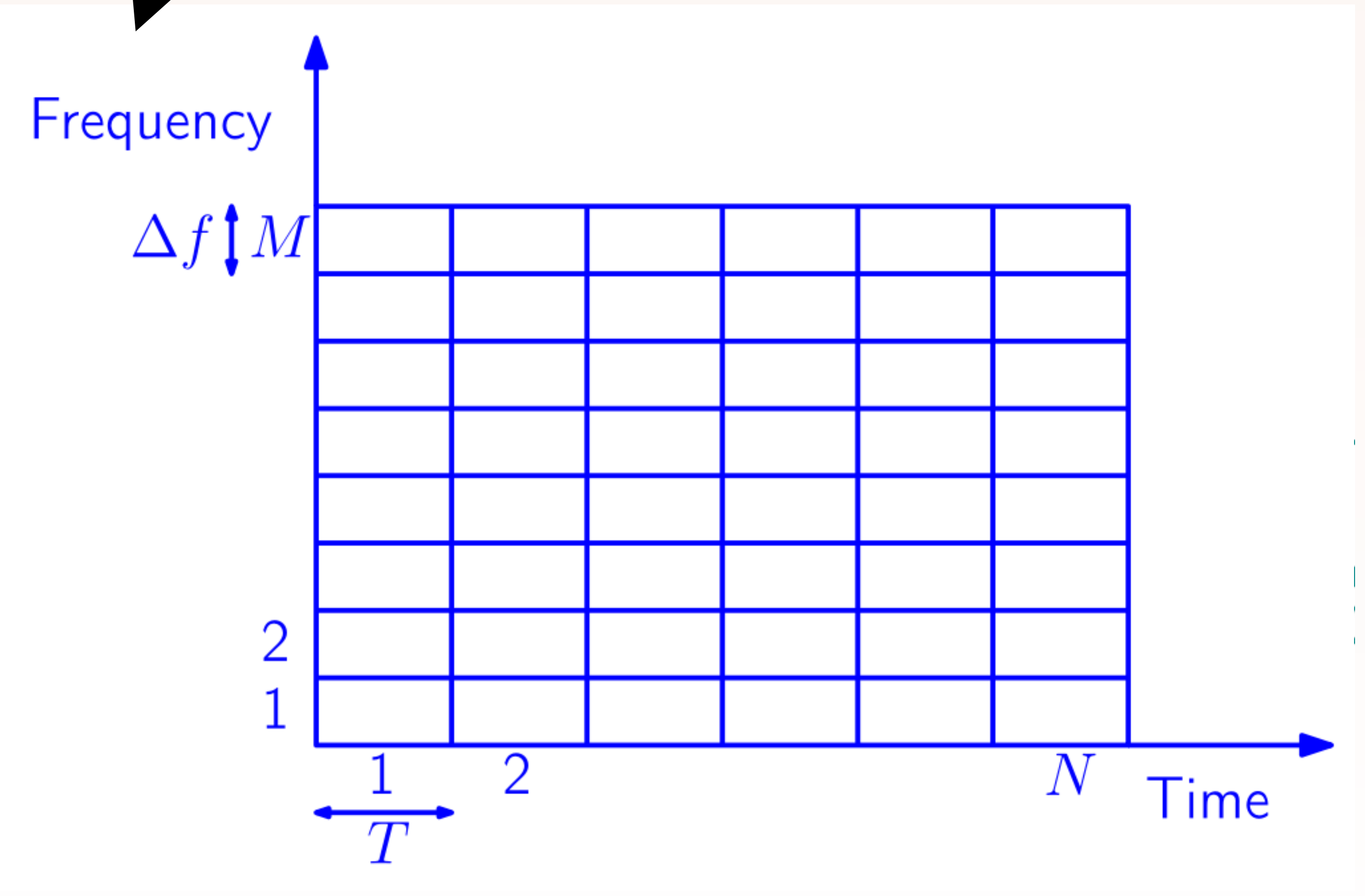
- $$s(t) = \sum_{k=0}^{N-1}\sum_{l=0}^{M-1} \mathbf{X}_{ft}[l,k] \underbrace{g_{tx}(t-kT)}_{\text{Some transmit waveform}} e^{j2\pi l\Delta f(t-kT)}$$ (this is called Heisenberg transform)

- The above $s(t)$ is ready to transmit.



delay-Doppler grid, where $\mathbf{X}$ lives.

ISFFT



Frequency-time grid, where $\mathbf{X}_{ft}$ lives in.

OTFS: channel (revisited)

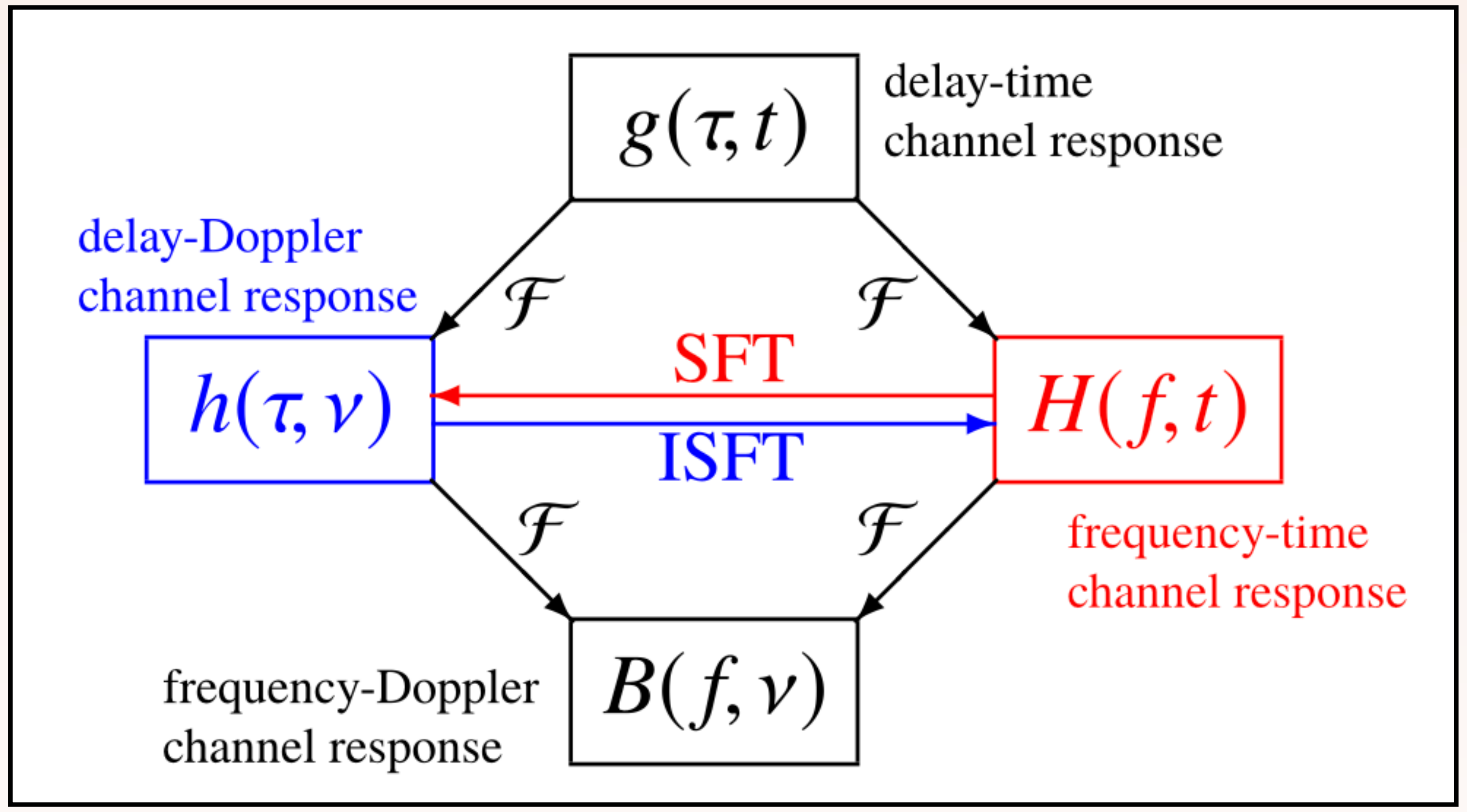
- $$s(t) = \sum_{k=0}^{N-1} \sum_{l=0}^{M-1} \mathbf{X}_{ft}[l, k] g_{tx}(t - kT) e^{j2\pi l \Delta f (t-kT)}$$ (what to transmit)

- Imparied by the channel, we can expressed the received signal as the following:

- $$r(t) = \int\int h(\tau, \nu) e^{j2\pi\nu t} s(t - \tau) d\nu d\tau$$ *(delay-Doppler channel).*

- $$= \int \underbrace{H(t, f)}_{\text{time-frequency response}} S(f) e^{j2\pi f t} df$$ *(Time-frequency channel).*

- $$= \int g(t, \tau) s(t - \tau) d\tau$$ *[delay - time expression, should be the most intuitive]*



Relations betwee different representations of LTV channels

OTFS: demodulation

- $s(t) = \sum_{k=0}^{N-1} \sum_{l=0}^{M-1} \mathbf{X}_{ft}[l,k] g_{tx}(t-kT) e^{j2\pi l \Delta f (t-kT)}, \; r(t) = \int\int h(\tau,\nu) e^{j2\pi\nu t} s(t-\tau) d\nu d\tau$ (other expressions available)
- With a matched filter, we process like the following:

Explanation needed

- $$Y(f,t) = A_{g_{rx},r}(f,t) \triangleq \int g_{rx}^*(t'-t) r(t') e^{-j2\pi f(t'-t)} dt'$$ (This is called "Wigner transform")
- "Cross-ambiguity function" between g_1, g_2 is defined as $A_{g_1,g_2}(f,t) \triangleq \int g_1(t') g_2^*(t'-t) e^{-j2\pi f(t'-t)} dt'$.
- Intuition? $A_{g_1,g_2}(f,t)$ defines the correlation between $g_1(t)$ and a delayed-by-t & frequency shifted by f version of $g_2(t)$.
- $$\mathbf{Y}_{ft}[l,k] = Y(f,t)\,|_{f=l\Delta f, t=kT} \in \mathbb{C}^{M\times N}$$ This lives in frequency-time domain
- Finally, we use the SFFT to process $\mathbf{Y}_{ft}[l,k]$, leading to a matrix $\mathbf{Y}$:
- $$\mathbf{Y}[m,n] = \frac{1}{\sqrt{NM}} \sum_{k=0}^{N-1} \sum_{l=0}^{M-1} \mathbf{Y}_{ft}[l,k] e^{-j2\pi(\frac{nk}{N} - \frac{ml}{M})}$$ This lives in DD domain
- *This finishes the "standard procedure" of OTFS modulation & demodulation.* (But understand little ☹️)

Delay-Doppler domain analysis

- The delay-Doppler transmitted and received samples matrices are related to those in the frequency-time domian by SFFT: $\mathbf{Y} = \mathrm{SFFT}(\mathbf{Y}_{ft}), \quad \mathbf{X} = \mathrm{SFFT}(\mathbf{X}_{ft}).$

- Also, the delay-Doppler domain channel matrix (of size $M \times N$) is related to the frequency-time channel matrix as: $\mathbf{H}_{dd} = \mathrm{SFFT}(\mathbf{H}_{ft}).$

- *(In frequency-time domain, ideally)* $\mathbf{Y}_{ft}[l,k] = \mathbf{H}_{ft}[l,k]\mathbf{X}_{ft}[l,k]\,.$ ⬅️ Omit/fix the k, nothing but $Y(f) = H(f)S(f)$

- Then we can actually derive:

$$\begin{aligned}\mathbf{Y}[m,n] &= \mathbf{H}_{dd}[m,n] \circledast \mathbf{X}[m,n] \\ &= \sum_{m'} \sum_{n'} \mathbf{H}_{dd}[m',n']\mathbf{X}[[m-m']_M, [n-n']_N] \\ &= \sum_{i=1}^{P} g_i \mathbf{X}[[m-l_i]_M, [n-k_i]_N]\end{aligned}$$

- Lemma: $\mathbf{Z} = \mathbf{U} \circledast \mathbf{V}$ can be equivalently written as $\mathrm{vec}(\mathbf{Z}) = \mathbf{B}_{\mathrm{U}} \cdot \mathrm{vec}(\mathbf{V})$, where $\mathbf{B}_{\mathrm{U}} = \mathrm{circ}[\mathrm{circ}(U_1), \ldots, \mathrm{circ}(U_N)]$.

$$\mathrm{circ}(a_0, \ldots, a_{N-1}) = \begin{pmatrix} a_0 & a_{N-1} & \cdots & & a_1 \\ a_1 & a_0 & & & a_2 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ a_{N-2} & & & a_0 & a_{N-1} \\ a_{N-1} & a_{N-2} & \cdots & a_1 & a_0 \end{pmatrix}$$

Delay-Doppler domain analysis

- (In frequency-time domain, ideally) $\mathbf{Y}_{ft}[l,k] = \mathbf{H}_{ft}[l,k]\mathbf{X}_{ft}[l,k]$.

⬇️ want to derive the bottom

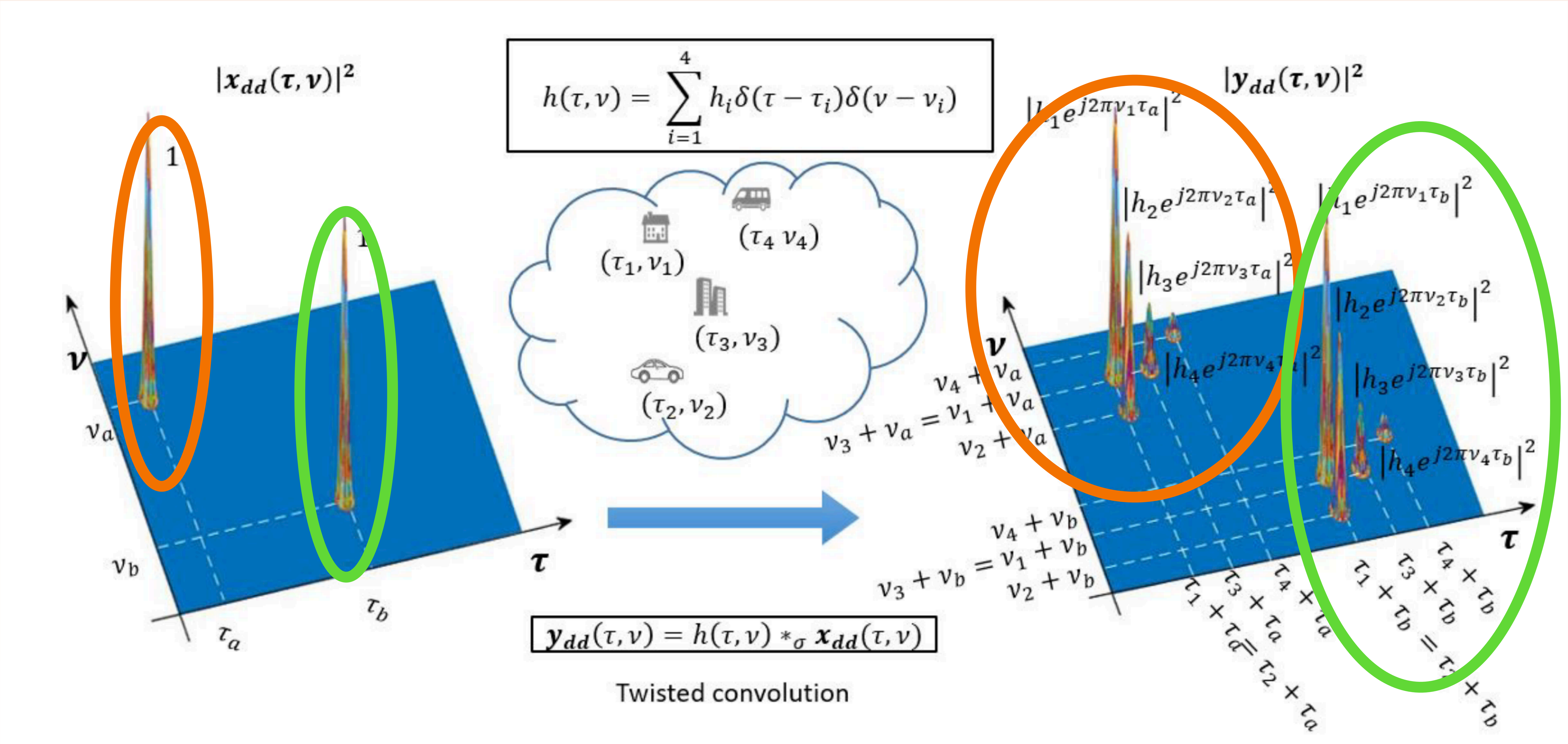
- $$\mathbf{Y}[m,n] = \mathbf{H}_{dd}[m,n] \circledast \mathbf{X}[m,n] \triangleq \sum_{m'} \sum_{n'} \mathbf{H}_{dd}[m',n']\mathbf{X}[[m-m']_M,[n-n']_N] .$$

- Proof sketch (details is quite tedious, if we do not use higher level theorems)
  - Suffice to prove $\mathrm{vec}(\mathbf{Y}) = \mathbf{B}_{\mathbf{H}_{dd}} \cdot \mathrm{vec}(\mathbf{X})$
  - $\mathbf{Y}_{ft}[l,k] = \mathbf{H}_{ft}[l,k]\mathbf{X}_{ft}[l,k] \iff \mathrm{vec}(\mathbf{Y}_{ft}) = \mathrm{vec}(\mathbf{H}_{ft}) \odot \mathrm{vec}(\mathbf{X}_{ft})$ $\quad\odot$ *element-wise / Hadarmard product*
  - $$\iff (\mathbf{F}_N \otimes \mathbf{F}_M^\dagger)^{-1}\underbrace{(\mathbf{F}_N \otimes \mathbf{F}_M^\dagger)\mathrm{vec}(\mathbf{Y}_{ft})}_{=\mathrm{vec}(\mathbf{Y})} = \mathrm{diag}(\mathrm{vec}(\mathbf{H}_{ft}))(\mathbf{F}_N \otimes \mathbf{F}_M^\dagger)^{-1}\underbrace{(\mathbf{F}_N \otimes \mathbf{F}_M^\dagger)\mathrm{vec}(\mathbf{X}_{ft})}_{=\mathrm{vec}(\mathbf{X})}$$

    $\otimes$ Kronecker product
  - $$\iff \mathrm{vec}(\mathbf{Y}) = \underbrace{(\mathbf{F}_N \otimes \mathbf{F}_M^\dagger)\ \mathrm{diag}(\mathrm{vec}(\mathbf{H}_{ft}))(\mathbf{F}_N \otimes \mathbf{F}_M^\dagger)^{-1}}_{\text{This, surprisingly, equals } \mathbf{B}_{\mathbf{H}_{dd}}}\ \mathrm{vec}(\mathbf{X})$$

TX and RX signal in DD domain (revisited)

$\mathbf{X}[m,n]$



$\mathbf{Y}[m,n]$

$$\mathbf{Y}[m,n] = \mathbf{H}_{dd}[m,n] \circledast \mathbf{X}[m,n]$$

OTFS: modulation & demodulation summary

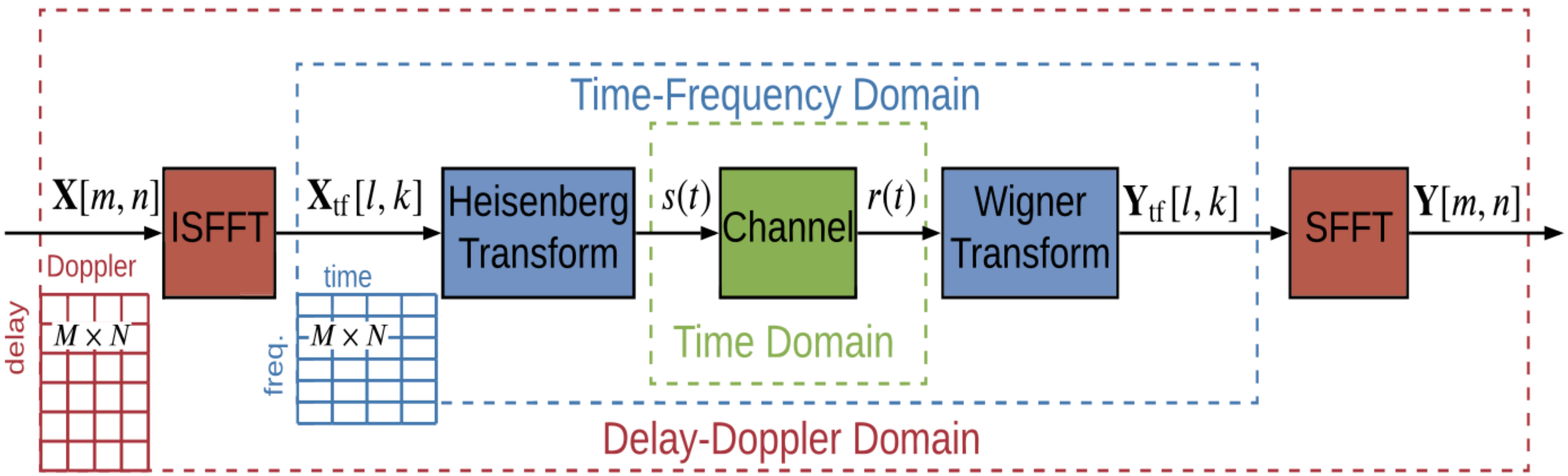


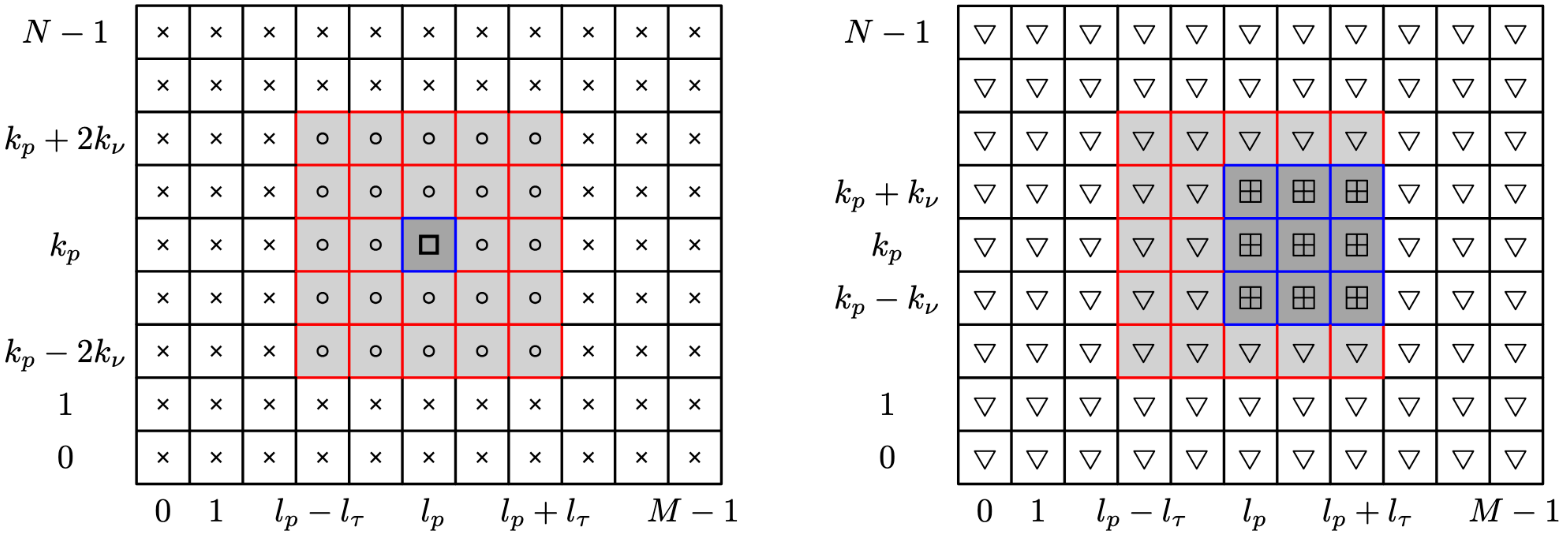
FIGURE 4.1 OTFS system diagram in its original form.

- As shown above, the OTFS modulator maps $\mathbf{X}[m,n]$ in the delay-Doppler domain to $\mathbf{X}_{ft}[l,k]$ in the time-frequency domain (using ISFFT). Then the Heisenberg transform is applied to $\mathbf{X}_{ft}[l,k]$ to generate time domain signal $s(t)$.

- At the receiver, $r(t)$ is transformed to the time-frequency domain by Wigner transform. Then to DD domian via SFFT.

Channel estimation (in delay-Doppler way)

- Esstimation in DD domain. (There are other domian-based channel estimation methods.)



(a) Tx symbol arrangement (□: pilot; ○: guard symbols; ×: data symbols)

Zero symbols

(b) Rx symbol pattern (▽: data detection, ⊞: channel estimation)

With channel estimation, how to decode?

- Suppose one have estimated the (delay-Doppler domain) channel matrix $\mathbf{H} \in \mathbb{C}^{NM \times NM}$, then we are facing (equivalently) a linear system with NM equations:
- $\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{w}$, (Comes from $\mathbf{Y} = \mathbf{H}_{dd} \circledast \mathbf{X}$, all $M \times N$ matices, using the previous lemma)
- Where where $\mathbf{y}, \mathbf{x}, \mathbf{w} \in \mathbb{C}^{NM \times 1}$ are $\mathbf{x}[k,l], \mathbf{y}[k,l], \mathbf{w}[k,l]$ in vectorized form, $k \in [N], l \in [M]$. $\mathbf{H}$ is treated as given.
- Given the sparse nature of $\mathbf{H}$ (each row sum=each column sum = P) we can solve it by message passing algorithm.

$$\widehat{\mathbf{x}} = \underset{\mathbf{x} \in \mathbb{A}^{NM \times 1}}{\arg\max} \Pr\left(\mathbf{x} \middle| \mathbf{y}, \mathbf{H}\right),$$

which has a complexity exponential in NM. Since the joint MAP detection can be intractable for practical values of N and M, we consider the symbol-by-symbol MAP detection rule for $c = 1, \ldots, NM$

$$\widehat{x}[c] = \underset{a_j \in \mathbb{A}}{\arg\max} \Pr\left(x[c] = a_j \middle| \mathbf{y}, \mathbf{H}\right) \tag{31}$$

$$= \underset{a_j \in \mathbb{A}}{\arg\max} \frac{1}{Q} \Pr\left(\mathbf{y} \middle| x[c] = a_j, \mathbf{H}\right) \tag{32}$$

$$\approx \underset{a_j \in \mathbb{A}}{\arg\max} \prod_{d \in \mathcal{J}_c} \Pr\left(y[d] \middle| x[c] = a_j, \mathbf{H}\right). \tag{33}$$

With channel estimation, how to decode?

- $$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{w},$$

Message passings from observation nodes $y[d]$ to variable nodes $x[c], c \in \mathcal{I}(d)$: The mean $\mu_{d,c}^{(i)}$ and variance $(\sigma_{d,c}^{(i)})^2$ of the interference, approximately modeled as a Gaussian random variable $\zeta_{d,c}^{(i)}$ defined as:

$$y[d] = x[c]H[d,c] + \underbrace{\sum_{e \in \mathcal{I}(d), e \neq c} x[e]H[d,e] + z[d]}_{\zeta_{d,c}^{(i)}}, \tag{34}$$

can be computed as:

$$\mu_{d,c}^{(i)} = \sum_{e \in \mathcal{I}(d), e \neq c} \sum_{j=1}^{Q} p_{e,d}^{(i-1)}(a_j) a_j H[d,e], \tag{35}$$

and

$$(\sigma_{d,c}^{(i)})^2 = \sum_{e \in \mathcal{I}(d), e \neq c} \left(\sum_{j=1}^{Q} p_{e,d}^{(i-1)}(a_j) |a_j|^2 |H[d,e]|^2 - \left| \sum_{j=1}^{Q} p_{e,d}^{(i-1)}(a_j) a_j H[d,e] \right|^2 \right) + \sigma^2. \tag{36}$$

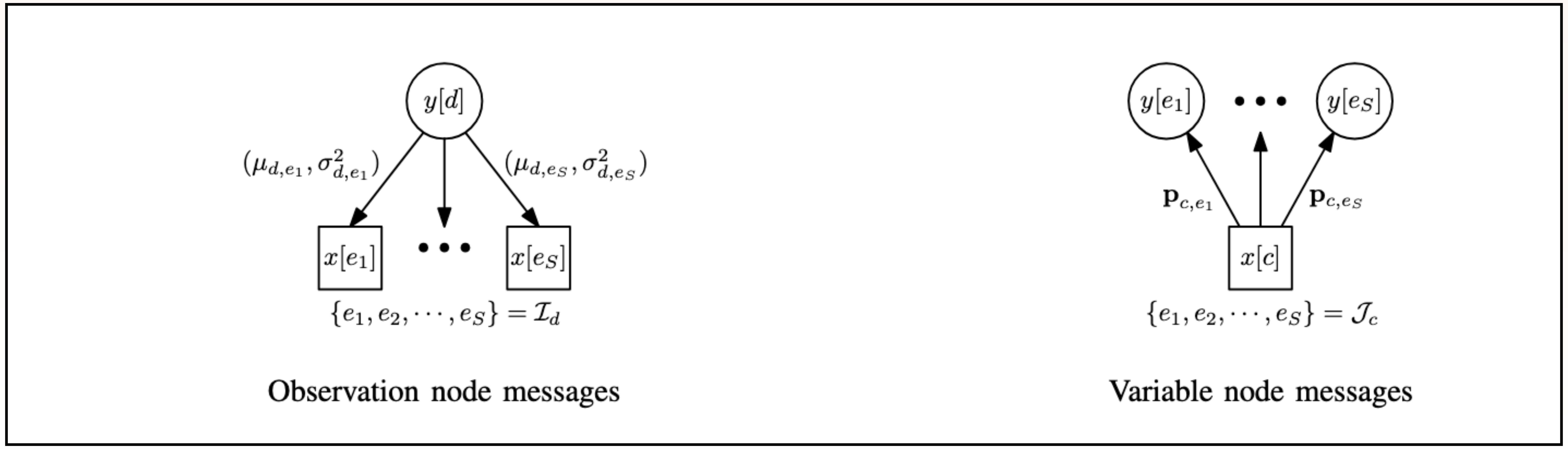
Message passings from variable nodes $x[c]$ to observation nodes $y[d], d \in \mathcal{J}(c)$: The pmf vector $\mathbf{p}_{c,d}^{(i)}$ can be updated as:

$$p_{c,d}^{(i)}(a_j) = \Delta \cdot \tilde{p}_{c,d}^{(i)}(a_j) + (1-\Delta) \cdot p_{c,d}^{(i-1)}(a_j), a_j \in \mathbb{A} \tag{37}$$

where $\Delta \in (0,1]$ is the *damping factor* used to improve the performance by controlling the convergence speed [18], and

$$\begin{aligned} \tilde{p}_{c,d}^{(i)}(a_j) &\propto \prod_{e \in \mathcal{J}(c), e \neq d} \Pr\left(y[e] \middle| x[c] = a_j, \mathbf{H} \right) \\ &= \prod_{e \in \mathcal{J}(c), e \neq d} \frac{\xi^{(i)}(e,c,j)}{\sum_{k=1}^{Q} \xi^{(i)}(e,c,k)}, \end{aligned} \tag{38}$$

where $\xi^{(i)}(e,c,k) = \exp\left(\frac{-\left| y[e] - \mu_{e,c}^{(i)} - H_{e,c} a_k \right|^2}{(\sigma_{e,c}^{(i)})^2} \right)$.



The message passing algorithm (copied from paper)

Algorithm 1 MP algorithm for OTFS symbol detection

Input: Received signal $\mathbf{y}$, channel matrix $\mathbf{H}$.

Initialization: pmf $\mathbf{p}_{c,d}^{(0)} = 1/Q, c = 1, \cdots, NM, d \in \mathcal{J}(c)$, iteration count $i = 1$.

repeat

- Observation nodes $y[d]$ compute the means $\mu_{d,c}^{(i)}$ and variances $(\sigma_{d,c}^{(i)})^2$ of Gaussian random variables $\zeta_{d,c}^{(i)}$ using $\mathbf{p}_{c,d}^{(i-1)}$ and pass them to variables nodes $x[c], c \in \mathcal{I}(d)$.
- Variable nodes $x[c]$ update $\mathbf{p}_{c,d}^{(i)}$ using $\mu_{d,c}^{(i)}$, $(\sigma_{d,c}^{(i)})^2$, and $\mathbf{p}_{c,d}^{(i-1)}$ and pass them to observation nodes $y[d], d \in \mathcal{J}(c)$.
- Compute convergence indicator $\eta^{(i)}$.
- Update the decision on the transmitted symbols $\widehat{x}[c], c = 1, \ldots, NM$ if needed.
- $i = i + 1$

until *Stopping criteria*;

Output: The decision on transmitted symbols $\widehat{x}[c]$.

With channel estimation, how to decode?

- Suppose one have estimated the (delay-Doppler domain) channel matrix $\mathbf{H} \in \mathbb{C}^{NM \times NM}$, then we are facing (equivalently) a linear system with NM equations:
- $$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{w},$$
- Where where $\mathbf{y}, \mathbf{x}, \mathbf{w}$ are $\mathbf{x}[k, l], \mathbf{y}[k, l], \mathbf{w}[k, l]$ in vectorized form, $k \in [N], l \in [M]$. $\mathbf{H}$ is treated given.

- Easier ways to do decoding?
- —MMSE! *(We use this in our MatLab code demo later)*
- $$\hat{\mathbf{x}} \triangleq (\mathbf{H}^{\dagger}\mathbf{H} + \lambda \mathbf{I})^{-1}\mathbf{H}^{\dagger}\mathbf{y}$$
- Note that: this involves a matrix inversion of a $NM \times NM$ matrix. Whose (in general) complexity is $\mathcal{O}(M^3N^3)$. Usually, when N, M are not too small. This will be huge…
- Side note: In fact, maybe we can leverage the property of $\mathbf{H}$ being sparse to accelarate it. But the result is shown not good as MP.

What if there is no Doppler?

IEEE WIRELESS COMMUNICATIONS LETTERS, VOL. 8, NO. 3, JUNE 2019

OTFS Performance on Static Multipath Channels

P. Raviteja, Emanuele Viterbo, and Yi Hong

- "…… In this letter, we **show that, in static multipath channels, the system structure of OTFS is equivalent to the asymmetric orthogonal frequency division multiplexing (A-OFDM, an OFDM-like scheme with unequal subcarrier spacing),** a scheme … bridging between cyclic prefix single carrier (CPSC) and traditional OFDM.

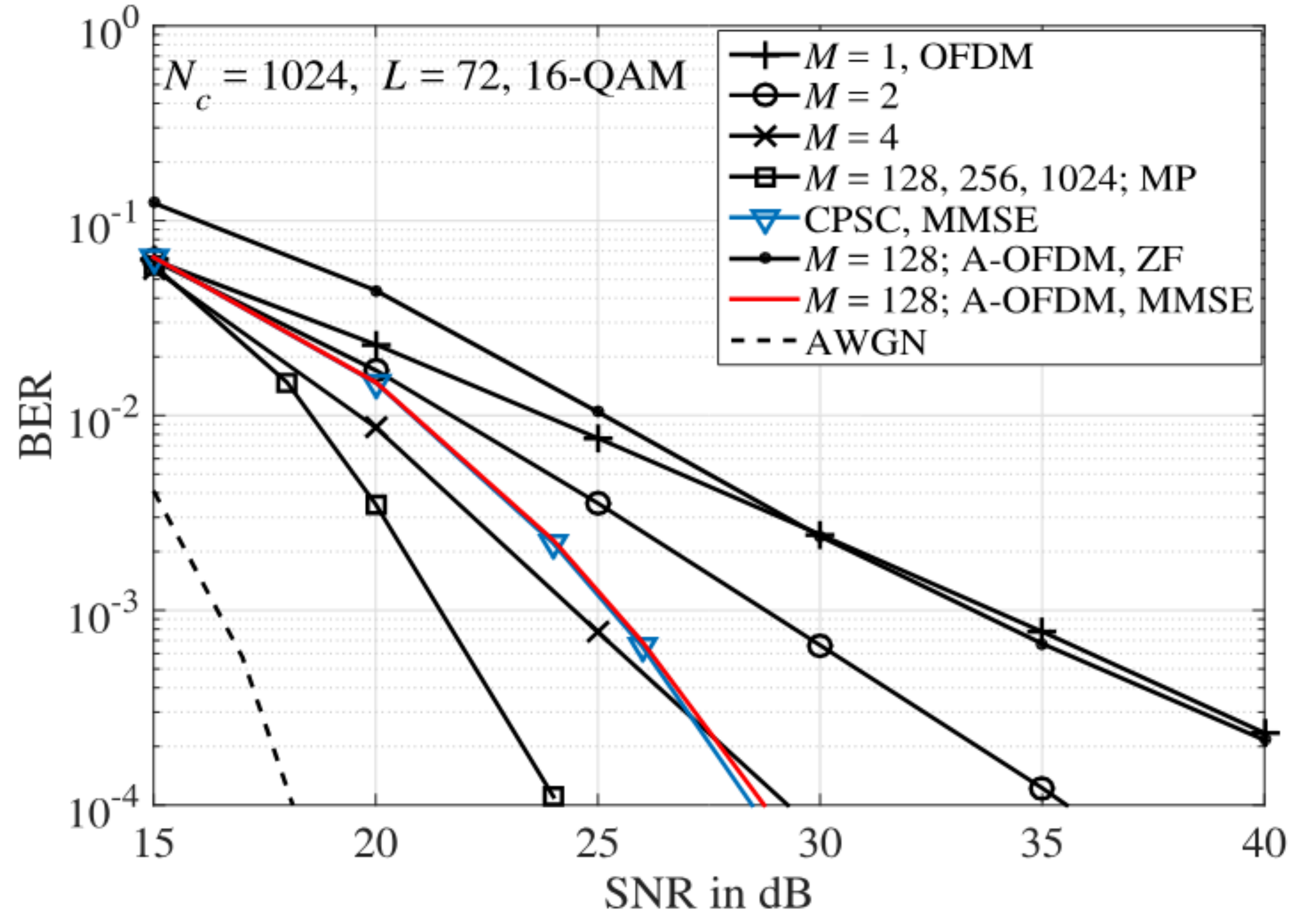


Fig. 3. BER of OTFS for different M with $N_c = 1024$, $L = 72$, and 16-QAM.

Matlab Code Demo

- Try some different SNRs and see the results.

- N = 16, M = 64. (The DD domain matrix is 16 x 64)
- Will output a number, indicating out of the 16x64 = 1024 data, how many are incorrect.

- 4QAM

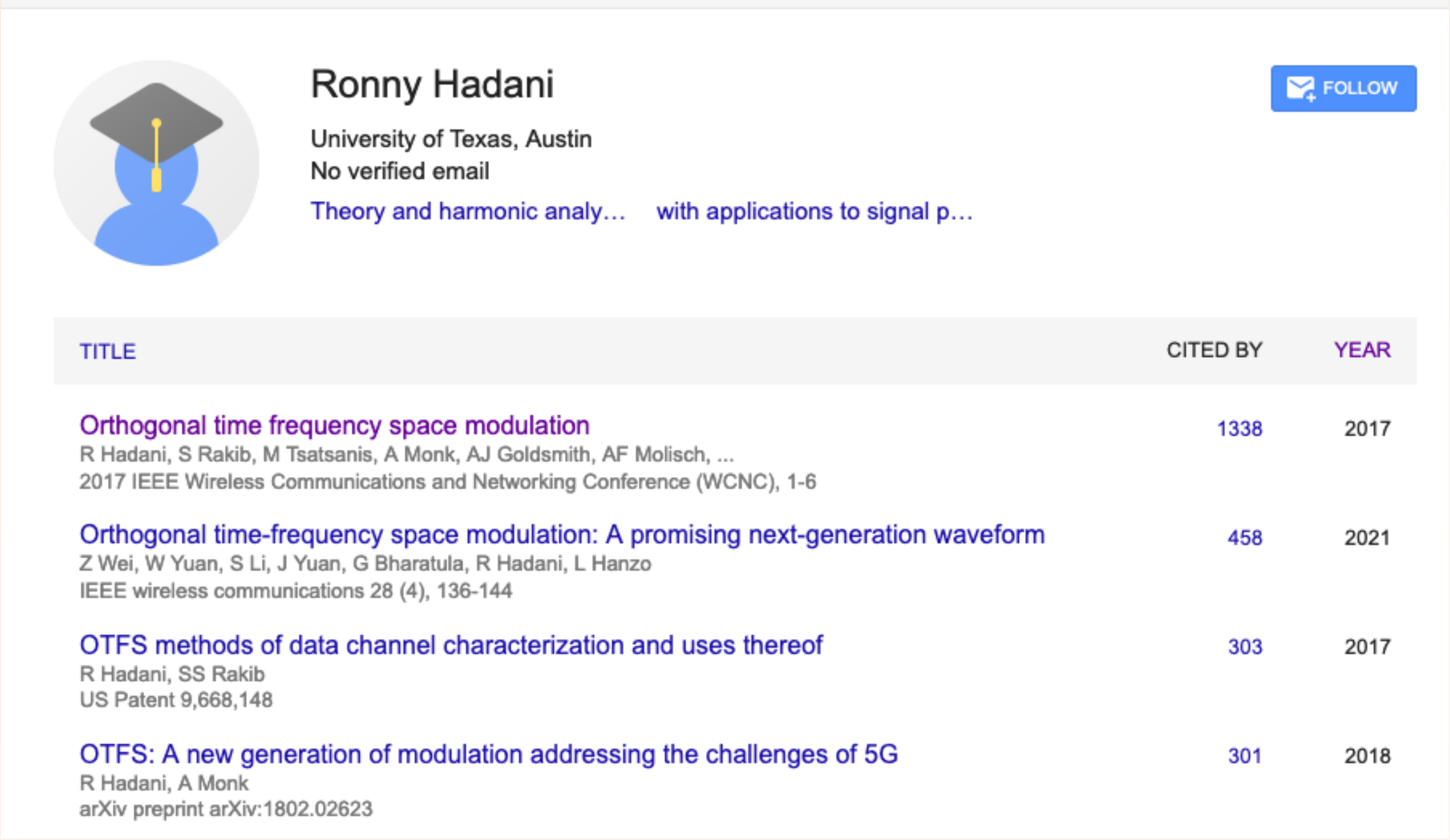
- About $\approx$ 9 taps.
- Rayleigh fading + AWGN
- **Assuming channel matrix is given**

Other concepts related to OTFS, but not mentioned

- Quasi-periodicity
- Pulsone: (the waveform) of OTFS
- Zak transform: a useful tool in simplifying the modulation / demodulation of OTFS
- …

OTFS community

- It was first presented in WCNC 2017, by Ronny Hadani.
- Since then, many (≥ 1000) works have been published.



- There is an official small community called “OTFS special interest group”. Members are experts and pioneers in OTFS. They host online webinar on a monthly basis.
- 

References

- Yi Hong et al., “Delay-Doppler Communications: Principles and Applications”, Elsevier, 2022 [Very comprehensive monograph on DD communication and OTFS]
- Saif Khan Mohammed, , “OTFS - A Mathematical Foundation for Communication and Radar Sensing in the Delay-Doppler Domain”, IEEE, 2022.
- Emanuele Viterbo, “An introduction to Orthogonal Time Frequency Space (OTFS) modulation for high mobility communications”, 2019. SIG on OTFS. [slides]
- Emanuele Viterbo, “OTFS Detection and Channel Estimation: a Low-Complexity Practical Solution”, 2021. SIG on OTFS. [slides]
- P. Raviteja, Khoa T. Phan, Yi Hong, and Emanuele Viterbo, “Interference Cancellation and Iterative Detection for Orthogonal Time Frequency Space Modulation” [The paper talked about message passing decoding of OTFS]
- Brian Olson, Steven Shaw, Chengzhi Shi, Christophe Pierre, Robert Parker, “Circulant Matrices and Their Application to Vibration Analysis”

OTFS: channel matrix representation

- Received signal in vector form in time domain (assuming noiseless)

$$\mathbf{r} = \mathbf{G}\mathbf{s}$$

- $\mathbf{G}$ is an $MN \times MN$ matrix of the following form

$$\mathbf{G} = \sum_{i=1}^{P} h_i' \mathbf{\Pi}^{l_i} \mathbf{\Delta}^{(k_i)},$$

where, $\mathbf{\Pi}$ is the permutation matrix (forward cyclic shift), and $\mathbf{\Delta}^{(k_i)}$ is the diagonal matrix

$$\underbrace{\mathbf{\Pi} = \begin{bmatrix} 0 & \cdots & 0 & 1 \\ 1 & \ddots & 0 & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 1 & 0 \end{bmatrix}_{MN \times MN}}_{\text{Delay (similar to OFDM)}}, \underbrace{\mathbf{\Delta}^{(k_i)} = \begin{bmatrix} e^{\frac{j2\pi k_i(0)}{MN}} & 0 & \cdots & 0 \\ 0 & e^{\frac{j2\pi k_i(1)}{MN}} & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & e^{\frac{j2\pi k_i(MN-1)}{MN}} \end{bmatrix}}_{\text{Doppler}}$$