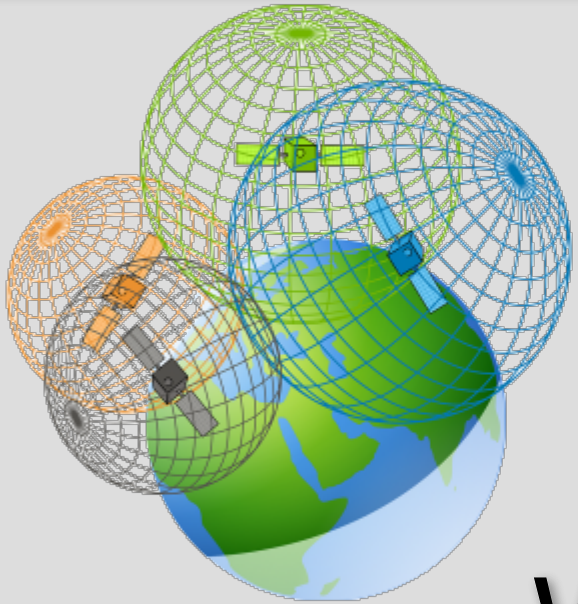
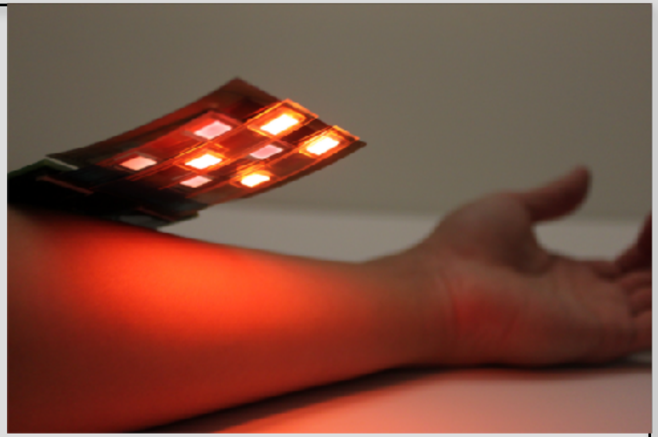




Welcome to EECS 16A!

Designing Information Devices and Systems I

Ana Arias and Miki Lustig

Fall 2021

Lecture 12A

Module 3: Correlation and Trilateration

Good morning!

Last time:

- Talked about GPS
  - Known position of satellites
  - Each satellite has its own signature
- Talked about inner product
  - Measure of similarity between vectors
  - When zero — orthogonal vectors
- Talked about using inner products for classification

Today:

- Computing delay with cross-correlation
- Finding position with multi-lateration

Announcements

Special Topic Joint EECS 16A & 16B Lecture by Jared Zerbe, Engineering Director at Apple

We are organizing a special topic lecture for EECS 16A and 16B about hardware design by one of Apple's chip design veterans. Please join us and ask Jared some tough questions.

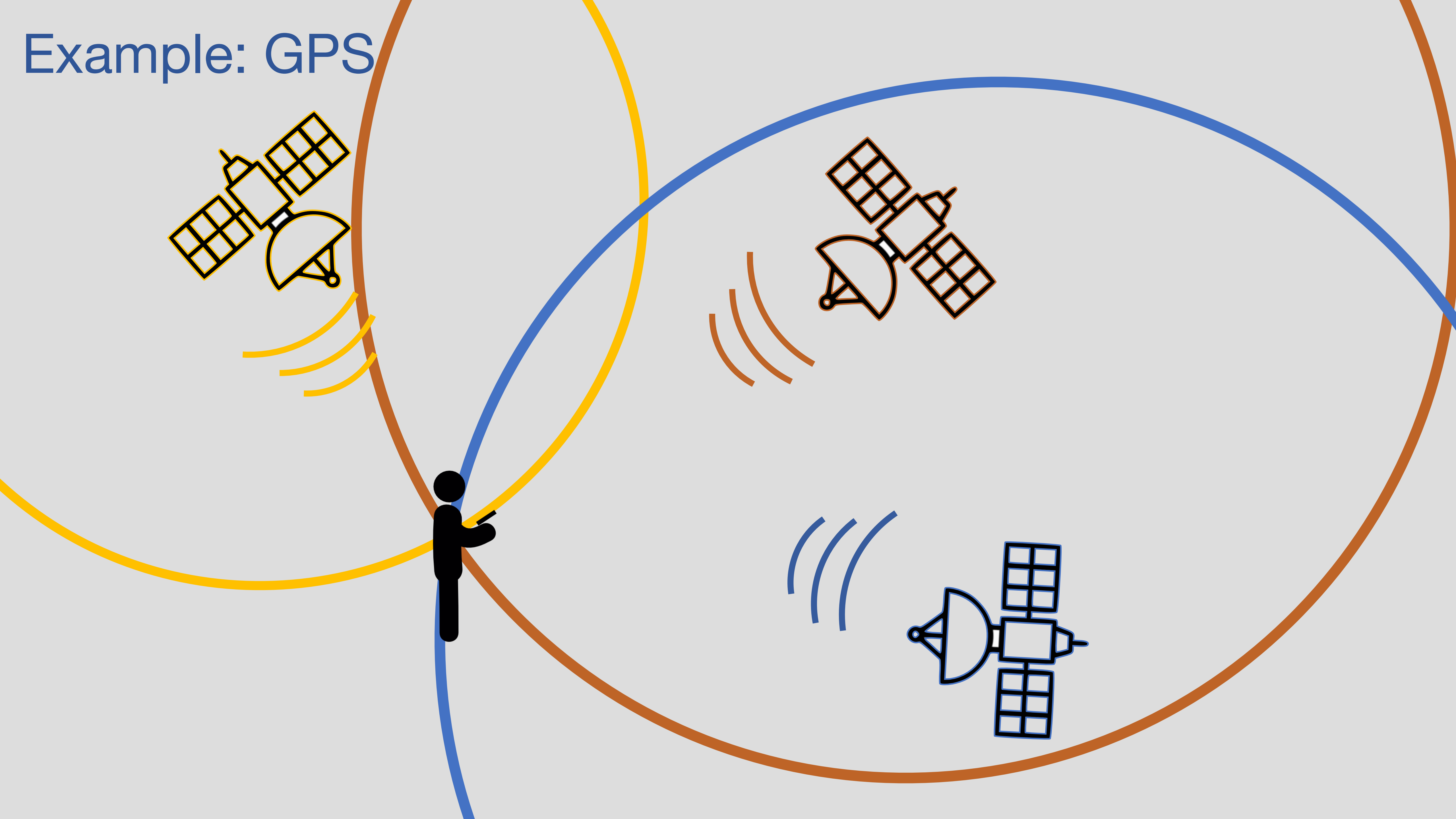
Special Topic Joint EECS 16A & 16B Lecture:
System-on-Chip Design - aka “what is in the future for me if I continue in hardware”
by Jared Zerbe, Engineering Director at Apple

Time/Date: 5-6:30pm on 11/17/2021
Zoom link: https://berkeley.zoom.us/j/98986312535

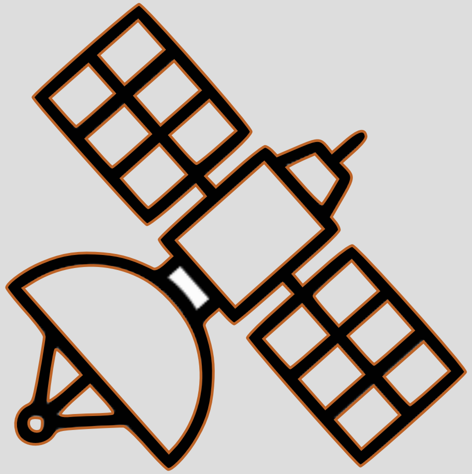
Abstract:
System on Chip “SoC” designs are complex systems built in silicon, and the heart of computing systems these days. In this talk we’ll go through some of the interesting challenges facing these designs, and how some of the tradeoffs are made in modern SoCs. This should be apropos for 16A/16B students wondering “what is in the future for me if I continue in hardware”. And you never know when your professors might sneak an exam question in from a guest lecture like this...

Bio:
Jared Zerbe received the B.S. degree in electrical engineering from Stanford University, Stanford, CA, in 1987. From 1987-1992 he worked at VLSI Technology and MIPS Computer Systems, where he designed high-performance floating-point units, and for over 20 years at Rambus he specialized in the design of high-speed I/O, PLL/DLL clock-recovery, and data-synchronization circuits. He has taught courses at both Berkeley and Stanford in high-speed I/O design and authored or co-authored over 40 IEEE conference and journal papers*, and is inventor of over 220 patents*. He served on the program committee for DesignCon and VLSI Circuits Symposium from 2010-2013 and was an associate editor for the Journal of Solid State Circuits from 2013-2014. In 2013 he joined Apple Inc. where since 2015 he has been Apple’s Exploratory Design group and is currently an Engineering Director. Jared was named an IEEE Fellow in 2019 for contributions to the development of high performance serial interfaces.

Example: GPS



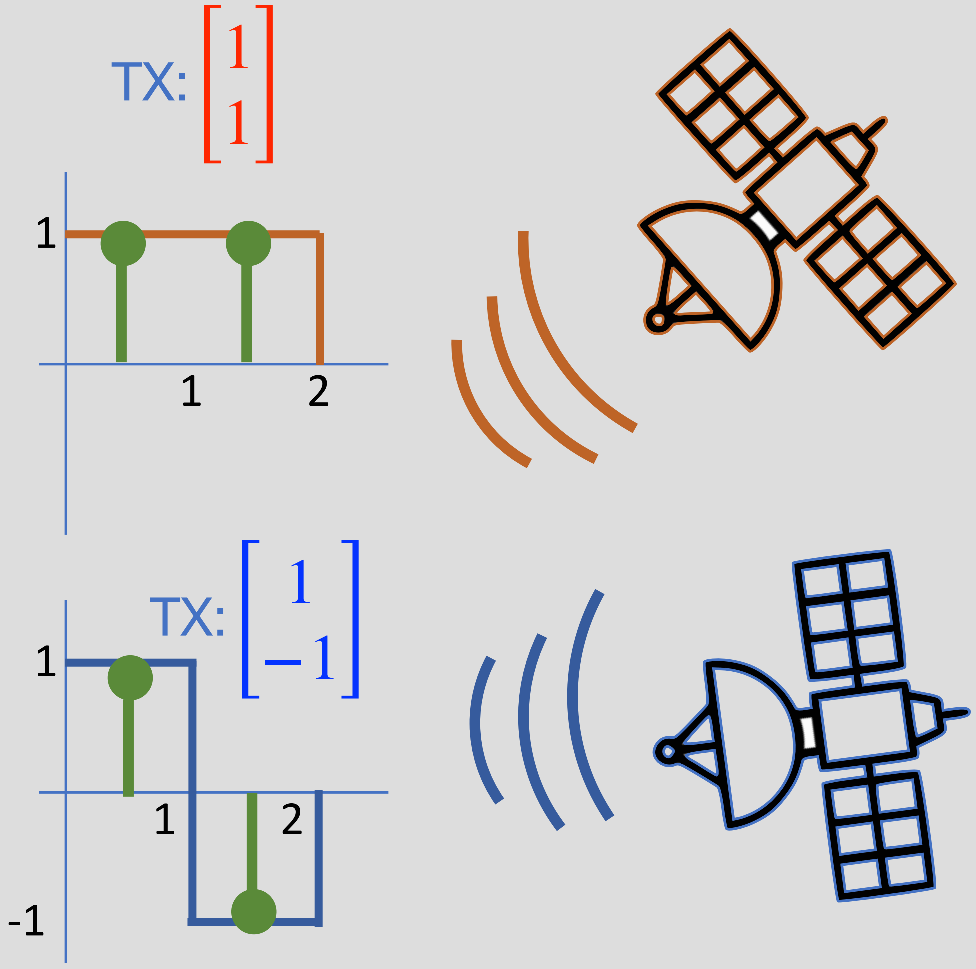
GPS



- 24 satellites
  - Known position
  - Time synchronized
  - 8 usually visible
- Problem:
  - Classify which satellite is transmitting
  - Estimate distance to GPS
  - Estimate position from noisy data
- Tools:
  - Inner product
  - Cross correlation
  - Least Squares

Localization

- Satellites transmit a unique code
  - Radio signal
- Signal is received and digitized by a receiver



Two problems:

1. Interference
2. Timing (Next!)

Interference

Possibility 1: Both sats are in TX

$$\vec{r} = \vec{s}_1 + \vec{s_2} + \vec{n}$$

Possibility 2: Only S1 is in Tx

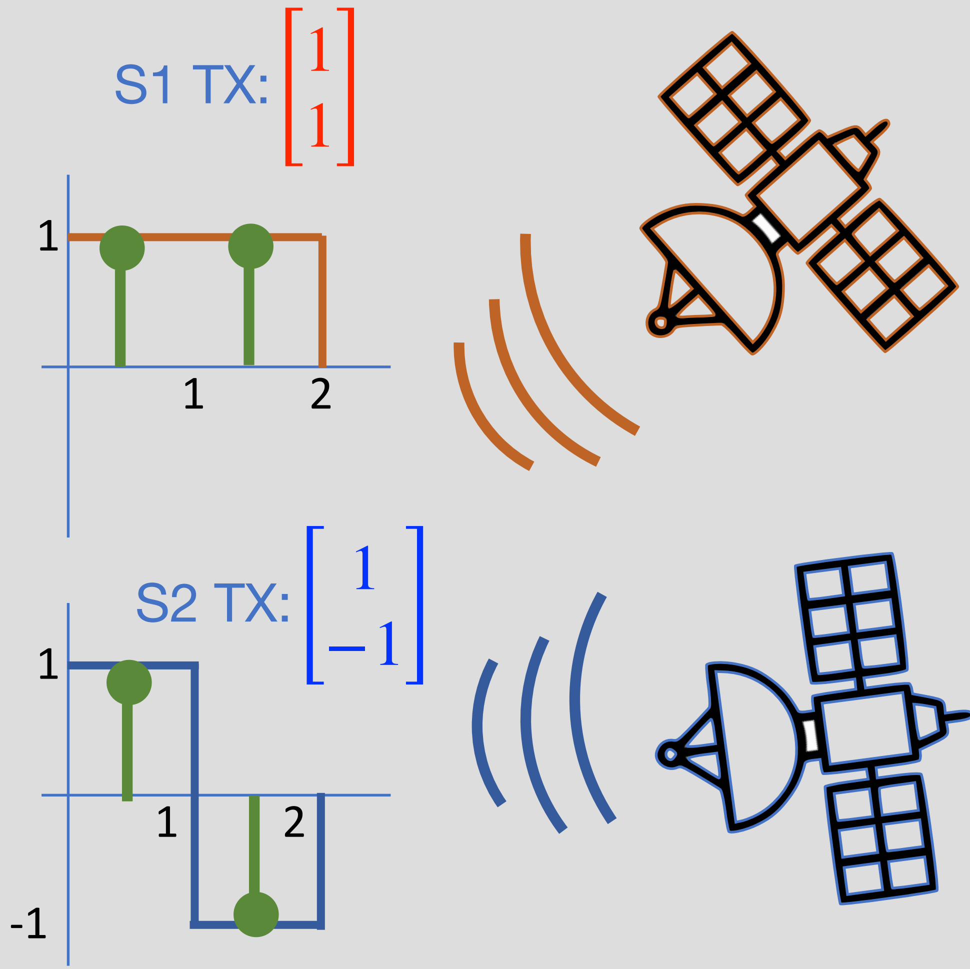
$$\vec{r} = \vec{s}_1 + \vec{n}$$

Possibility 3: Only S2 is in Tx

$$\vec{r} = \vec{s}_2 + \vec{n}$$

Possibility 4: None is in Tx

$$\vec{r} = \vec{n}$$



Interference

Possibility 1: Both sats are in TX

$$\vec{r} = \vec{s}_1 + \vec{s}_2 + \vec{n}$$

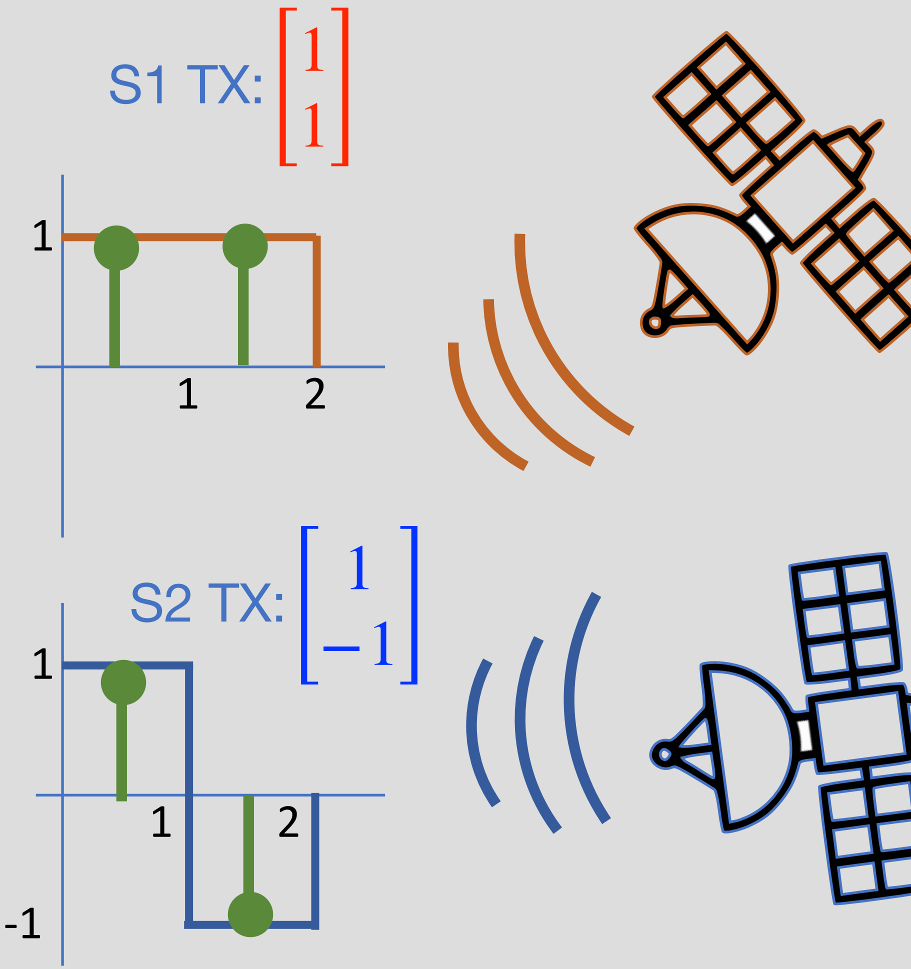
$$\langle \vec{r}, \vec{s}_1 \rangle = \langle \vec{s}_1 + \vec{s}_2 + \vec{n}, \vec{s}_1 \rangle$$

$$= \langle \vec{s}_1, \vec{s}_1 \rangle + \langle \vec{s}_2, \vec{s}_1 \rangle + \langle \vec{n}, \vec{s}_1 \rangle$$

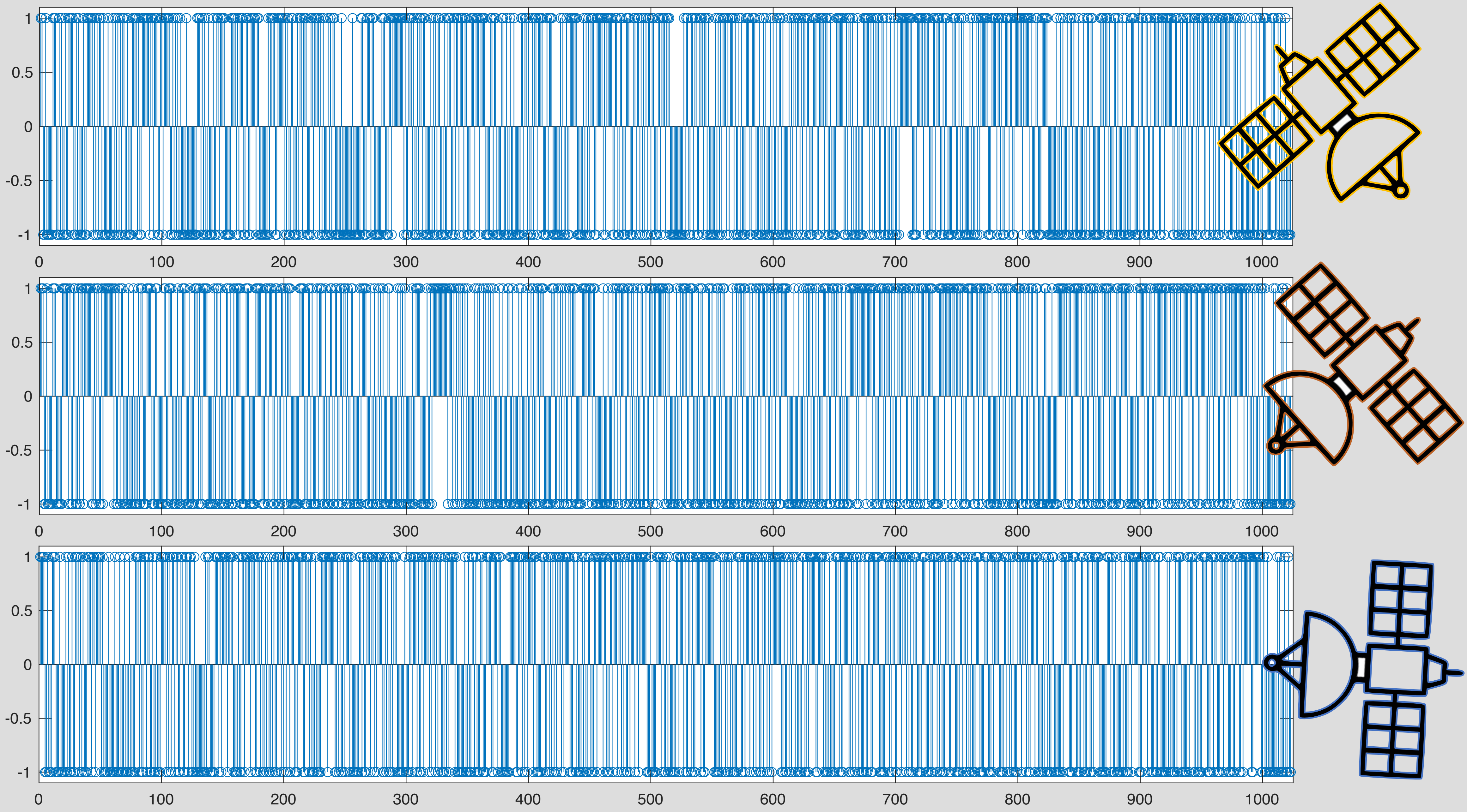
Desired Interference Small

Q: How to design codes that don't interfere?

A: Make them orthogonal! $\langle \vec{s}_2, \vec{s}_1 \rangle = 0$

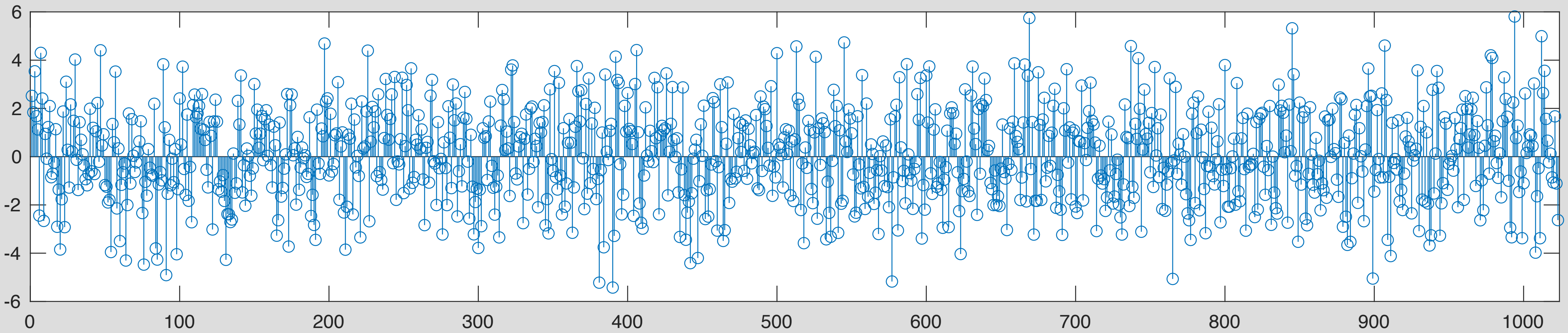


GPS Gold Codes

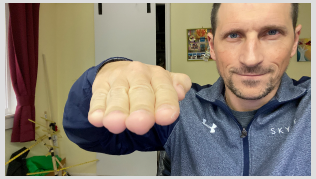


Example:

$\vec{r} =$



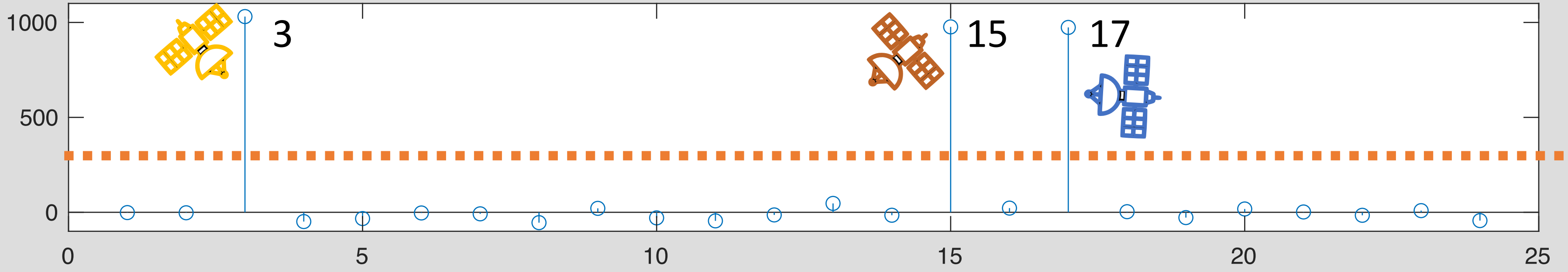
$$\langle \vec{r}, \vec{s}_i \rangle = \vec{r}^T \vec{s}_i$$



$\vec{r}^T$

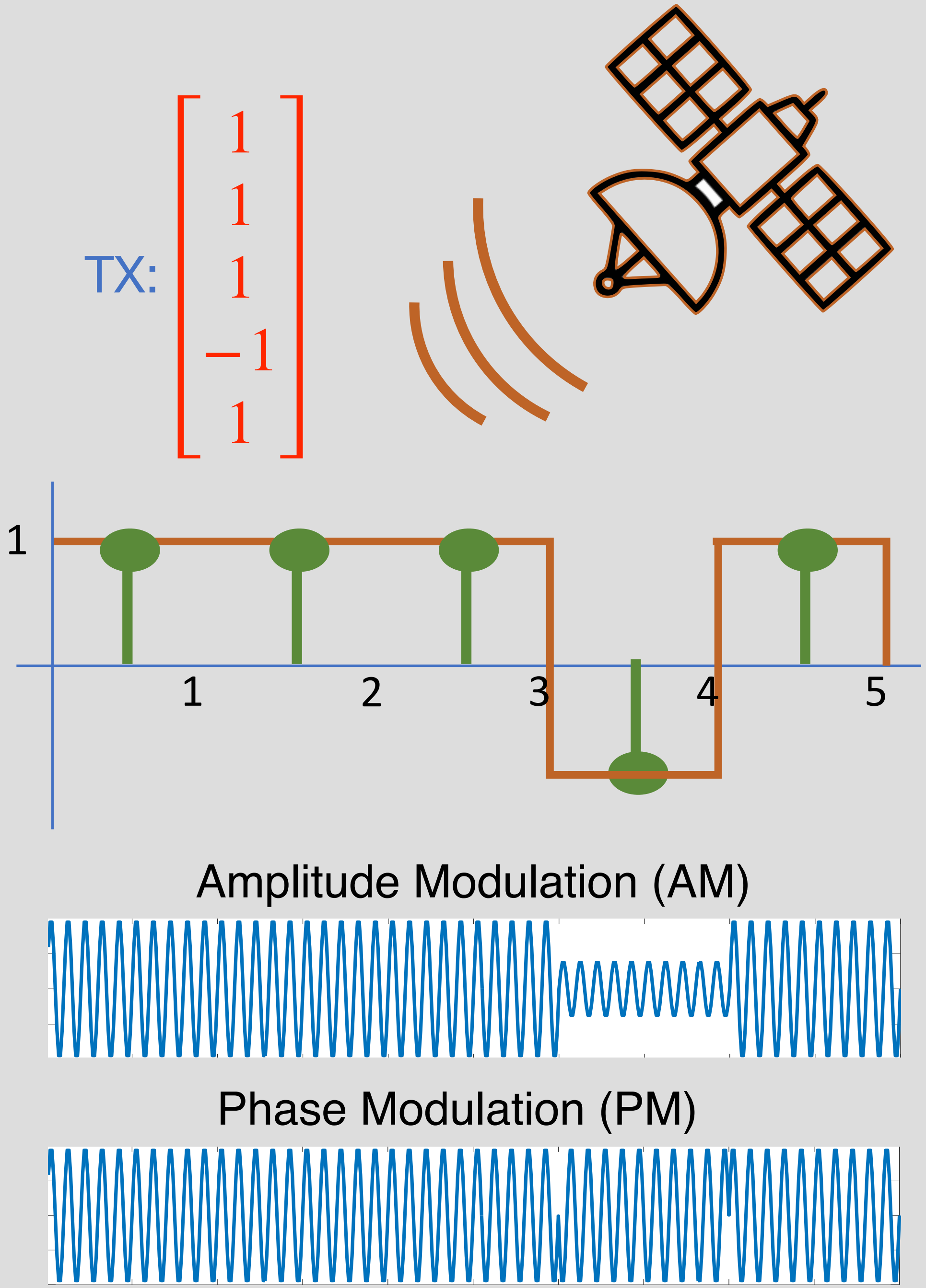
$\vec{s}_1$ $\vec{s}_2$ $\vec{s}_3$ ... $\vec{s}_{24}$ $=$

$\vec{r}^T\vec{s}_1$ $\vec{r}^T\vec{s}_2$ ... $\vec{r}^T\vec{s}_{24}$



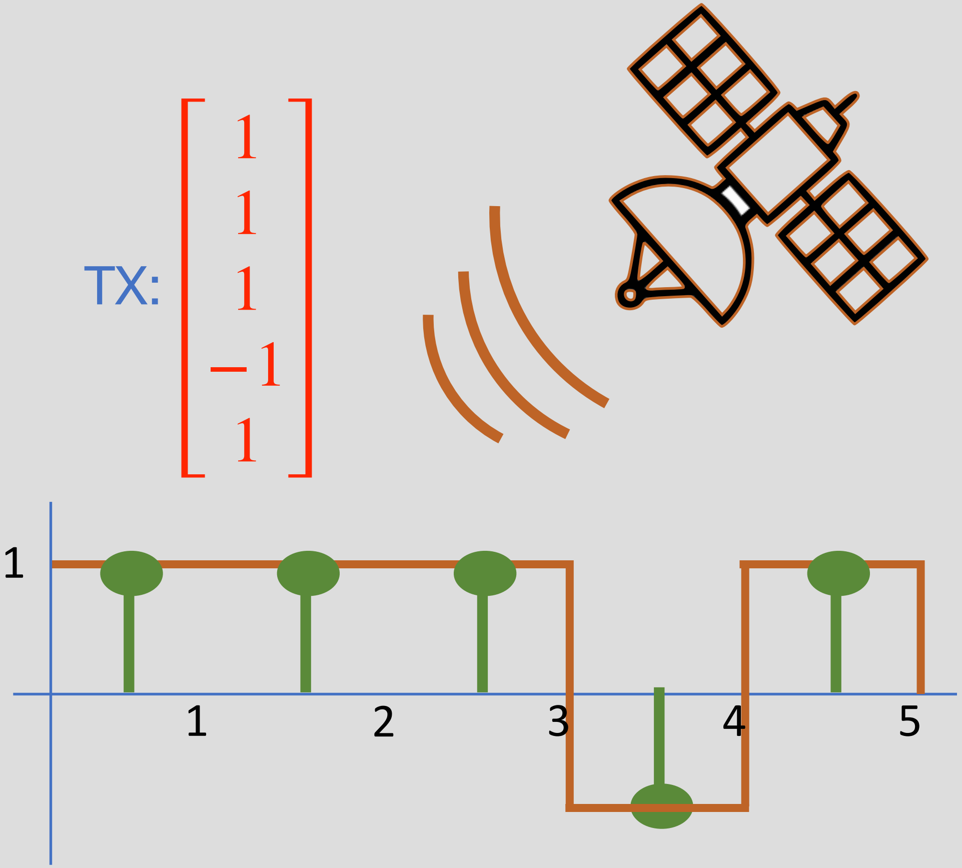
Timing….

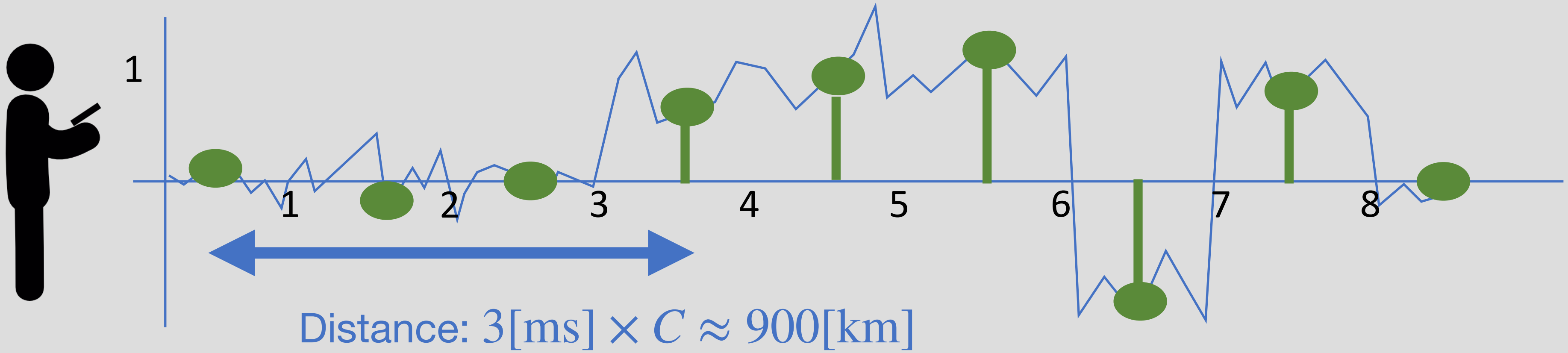
- Satellites transmit a (modulated) unique code
  - Radio signal
- Signal is received (demodulated) and digitized by a receiver



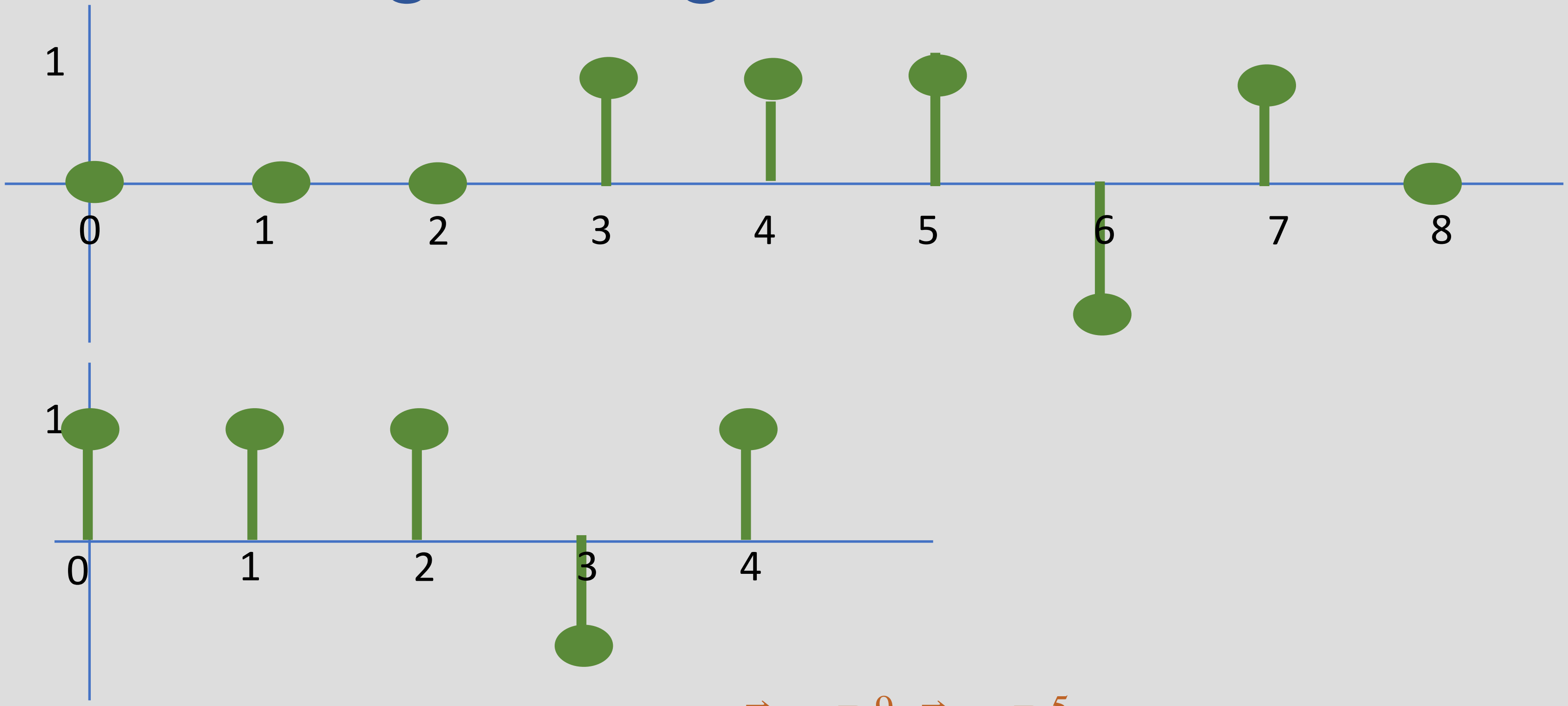
Timing….

- Satellites transmit a unique code
    - Radio signal
- Signal is received and digitized by a receiver





“Pattern Matching” of Signals



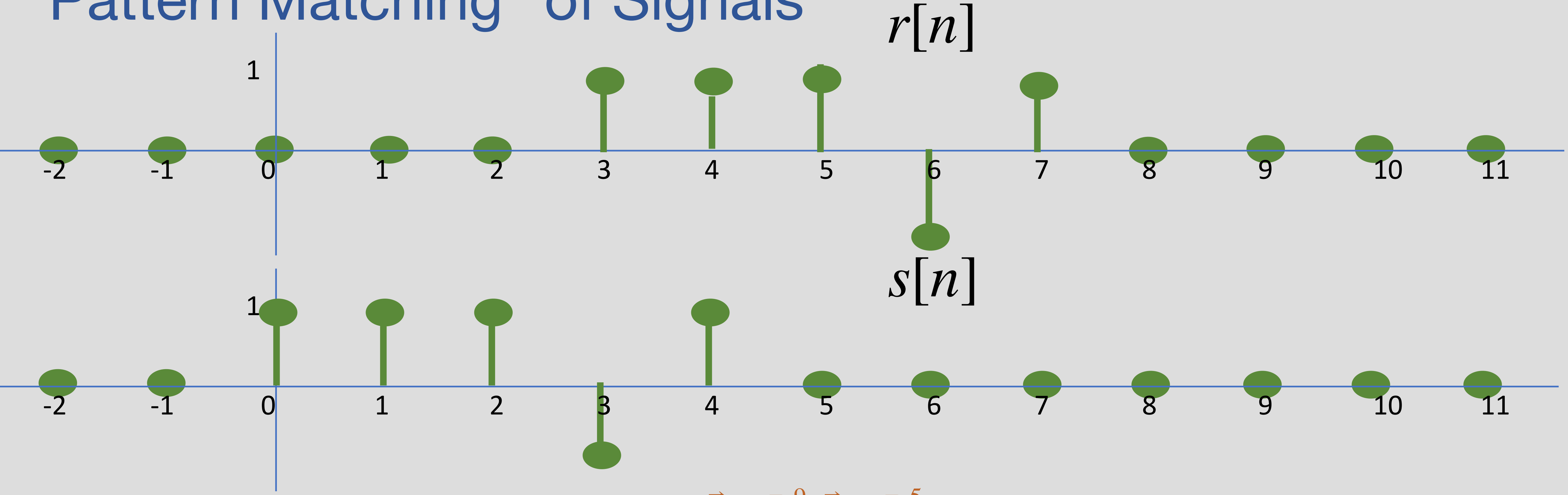
Problem: vectors (signals) not the same length… $\vec{r} \in \mathbb{R}^9$, $\vec{s} \in \mathbb{R}^5$

Solution: Define infinite signals $r[n], s[n]$ by zero-padding

$$\vec{r} = [r_0 \quad r_1 \quad r_2 \quad \cdots \quad r_8]^T \quad \Rightarrow r[n] = \begin{cases} r_n & 0 \le n \le 8 \\ 0 & \text{elsewhere} \end{cases}$$

$$\vec{s} = [s_0 \quad s_1 \quad s_2 \quad \cdots \quad s_4]^T \quad \Rightarrow s[n] = \begin{cases} s_n & 0 \le n \le 4 \\ 0 & \text{elsewhere} \end{cases}$$

"Pattern Matching" of Signals



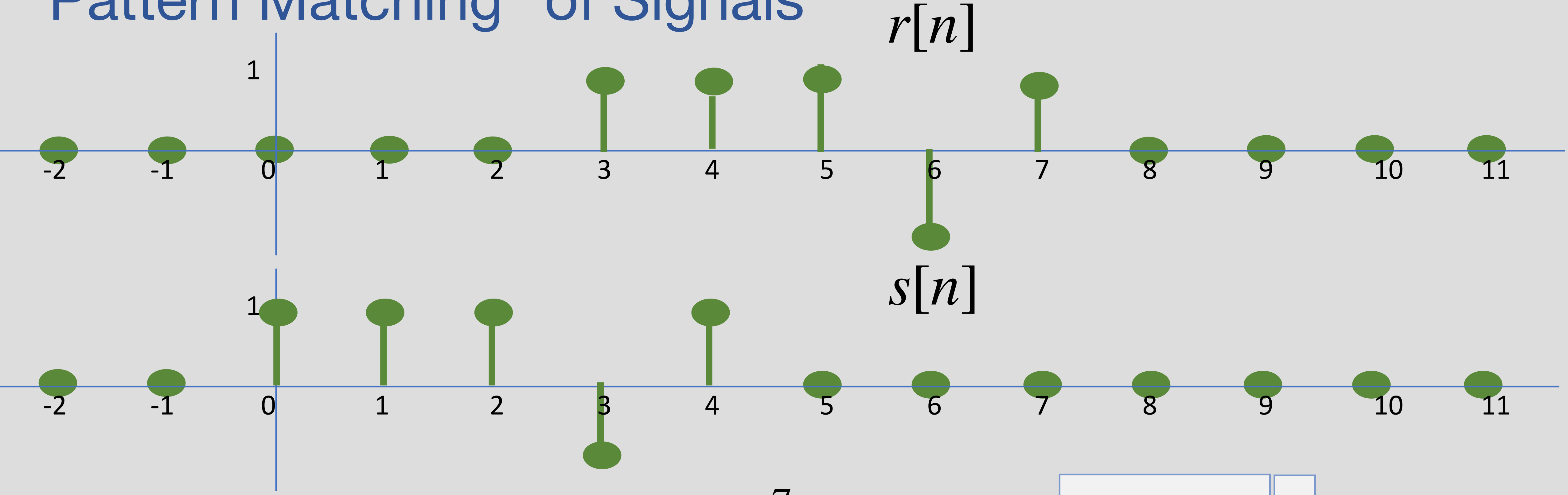
Problem: vectors (signals) not the same length... $\vec{r} \in \mathbb{R}^9$, $\vec{s} \in \mathbb{R}^5$

Solution: Define infinite signals $r[n], s[n]$ by zero-padding

$$\vec{r} = [r_0 \quad r_1 \quad r_2 \quad \cdots \quad r_8]^T \quad \Rightarrow r[n] = \begin{cases} r_n & 0 \le n \le 8 \\ 0 & \text{elsewhere} \end{cases}$$

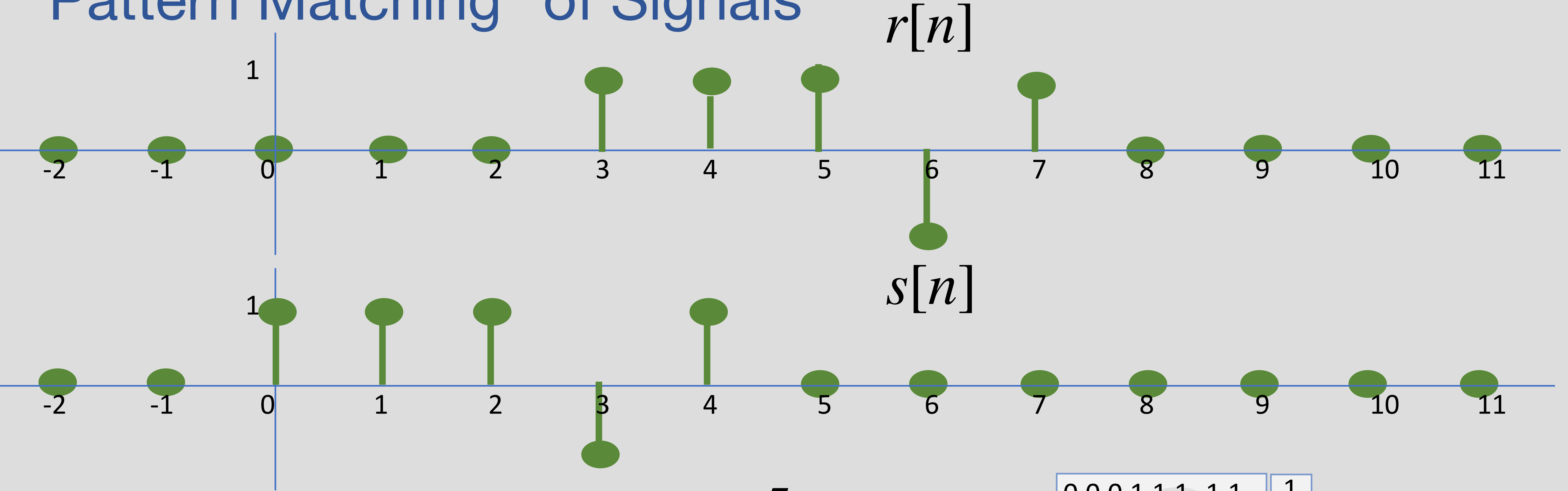
$$\vec{s} = [s_0 \quad s_1 \quad s_2 \quad \cdots \quad s_4]^T \quad \Rightarrow s[n] = \begin{cases} s_n & 0 \le n \le 4 \\ 0 & \text{elsewhere} \end{cases}$$

“Pattern Matching” of Signals



$$\langle r[n], s[n]\rangle = \sum_{n=-\infty}^{\infty} r[n]s[n] = \sum_{n=0}^{7} r[n]s[n] =$$

“Pattern Matching” of Signals

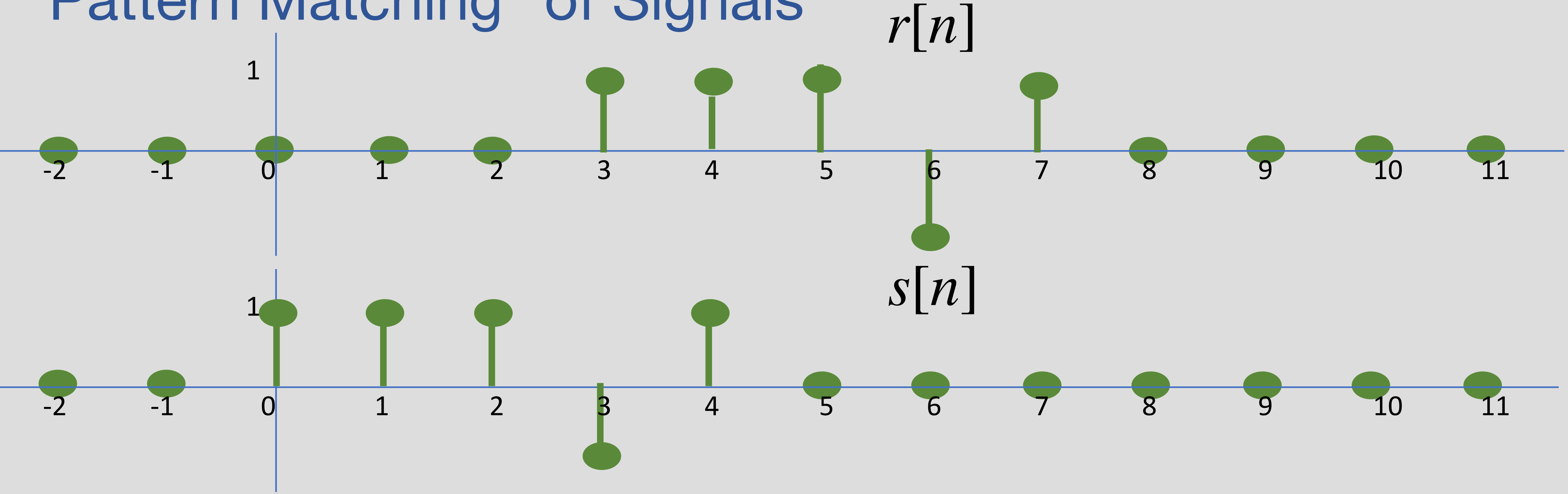


$$\langle r[n], s[n] \rangle = \sum_{n=-\infty}^{\infty} r[n]s[n] = \sum_{n=0}^{7} r[n]s[n] = \begin{bmatrix} 0,0,0,1,1,1,-1,1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ -1 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = 0$$

Q: How to match with shifted version?

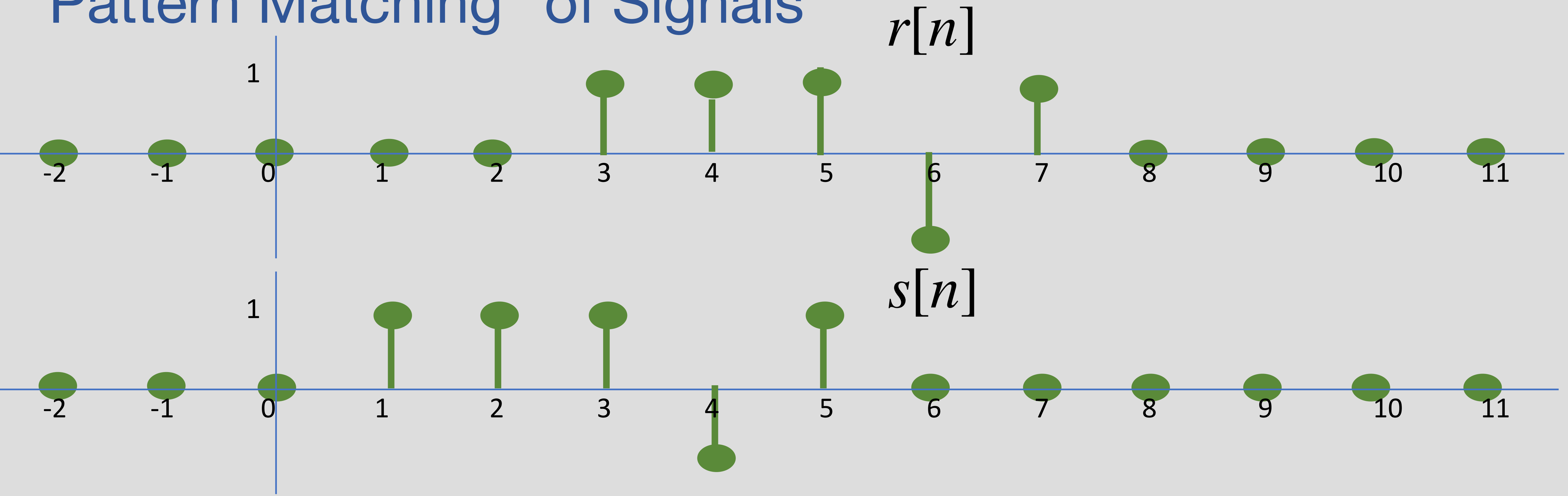
A: compute: $\langle r[n], s[n-1] \rangle = \sum_{n=-\infty}^{\infty} r[n]s[n-1]$

“Pattern Matching” of Signals



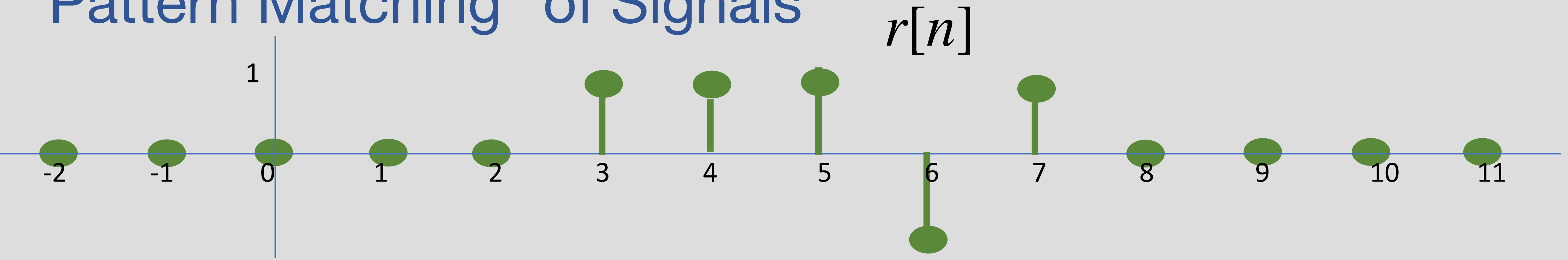
$$\langle r[n], s[n-1] \rangle = \sum_{n=-\infty}^{\infty} r[n]s[n-1]$$

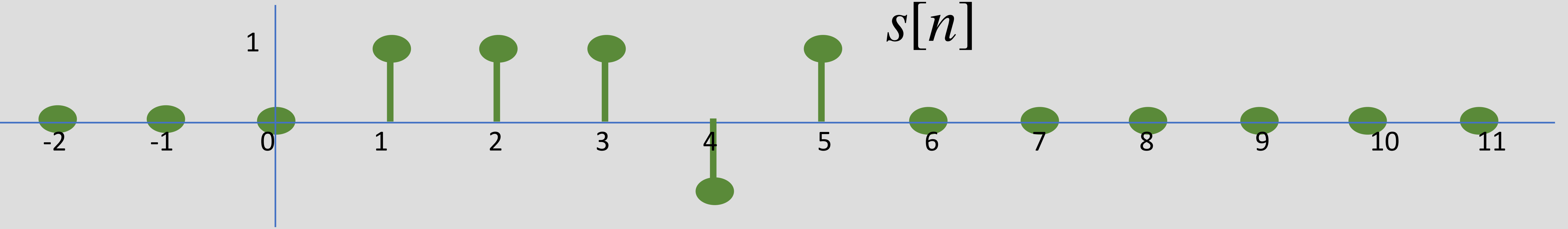
"Pattern Matching" of Signals



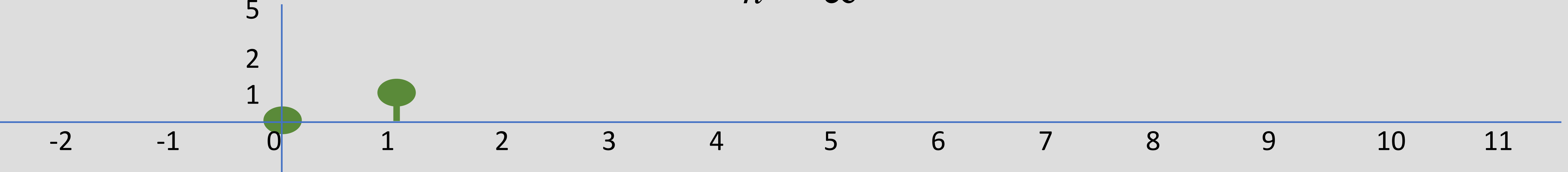
$$\mathrm{corr}_{\vec{r}}(\vec{s})[1] = \langle r[n], s[n-1] \rangle = \sum_{n=-\infty}^{\infty} r[n]s[n-1] = 1$$

“Pattern Matching” of Signals

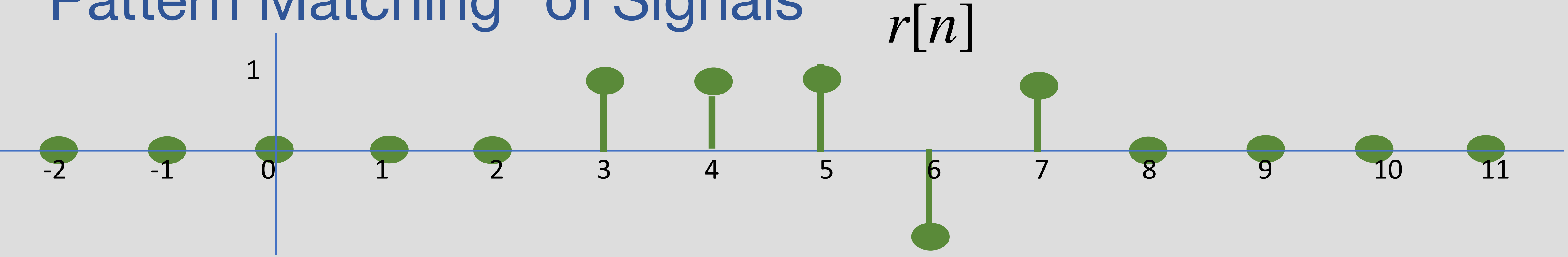


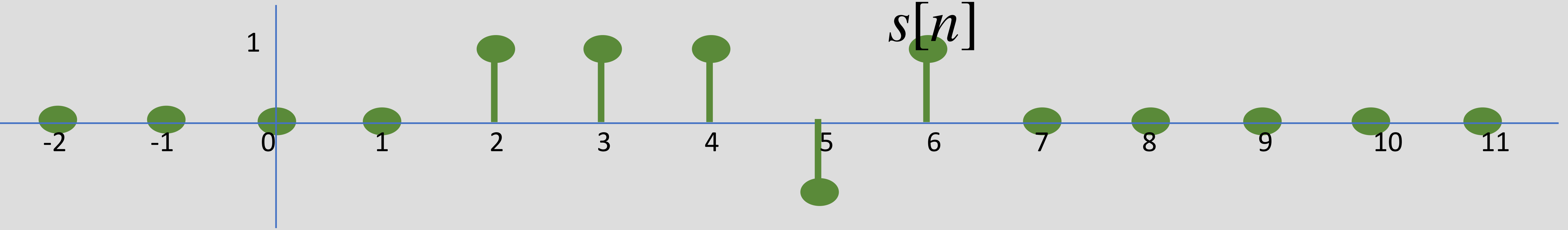


$$\text{corr}_{\vec{r}}(\vec{s})[1] = \langle r[n], s[n-1] \rangle = \sum_{n=-\infty}^{\infty} r[n]s[n-1] = 1$$

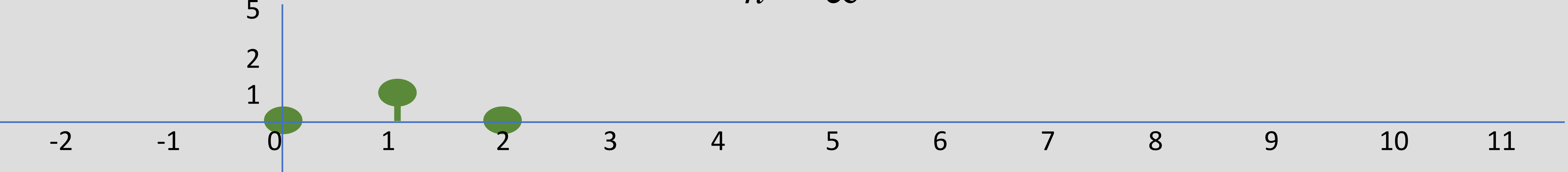


“Pattern Matching” of Signals

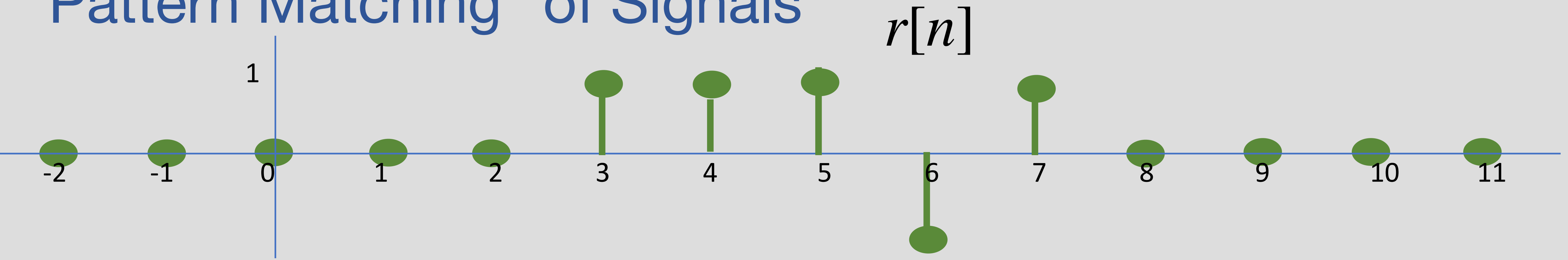


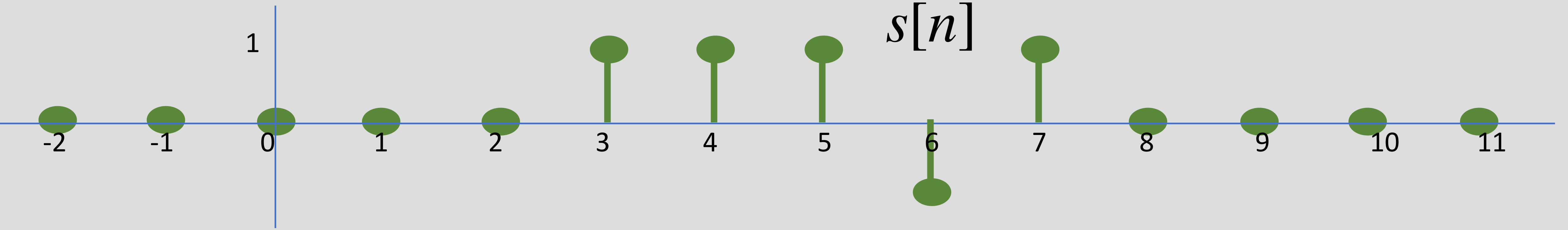


$$\mathrm{corr}_{\vec{r}}(\vec{s})[2] = \langle r[n], s[n-2] \rangle = \sum_{n=-\infty}^{\infty} r[n]s[n-2] = 0$$

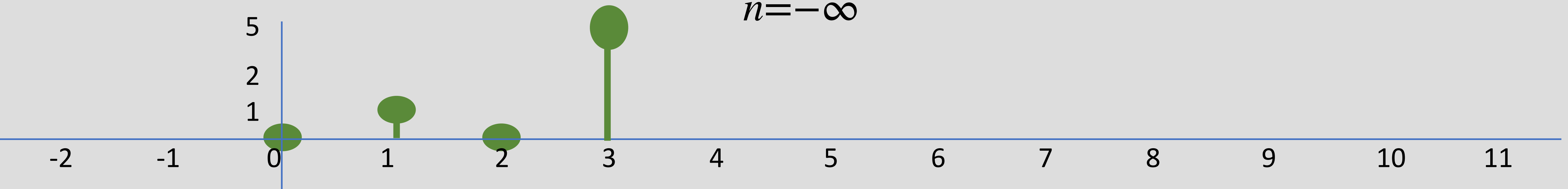


“Pattern Matching” of Signals

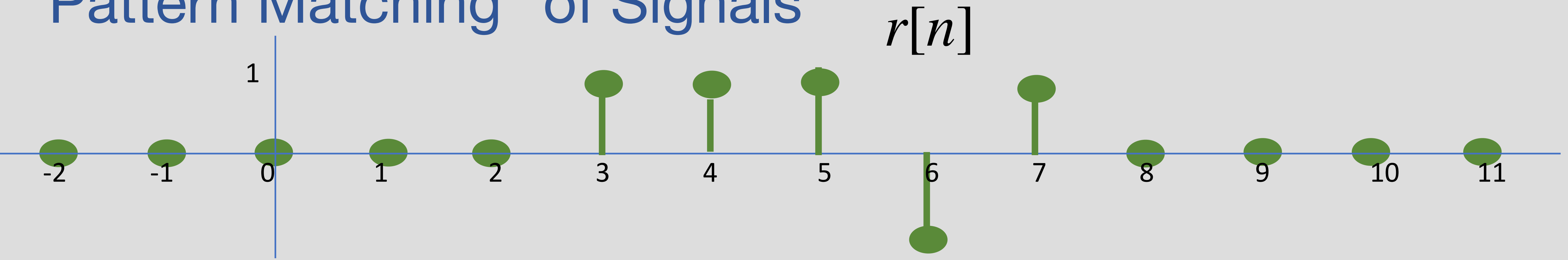


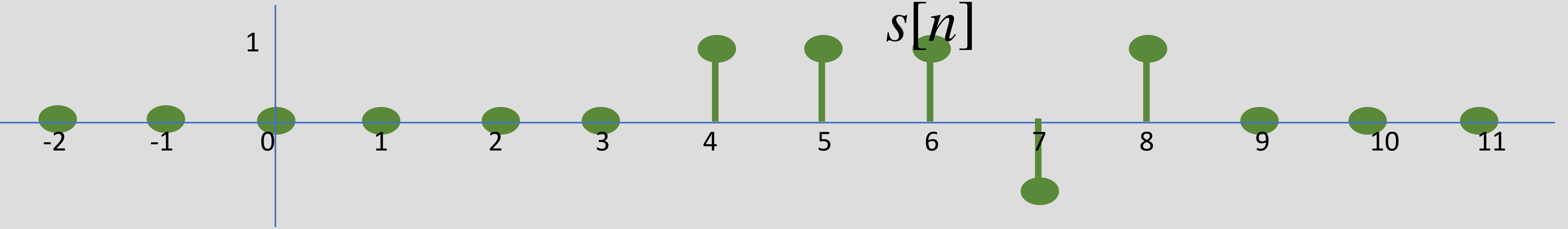


$$\text{corr}_{\vec{r}}(\vec{s})[3] = \langle r[n], s[n-3] \rangle = \sum_{n=-\infty}^{\infty} r[n]s[n-3] = 5$$

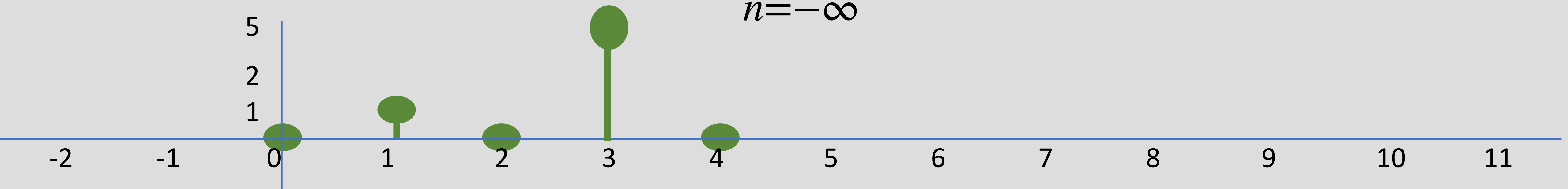


"Pattern Matching" of Signals

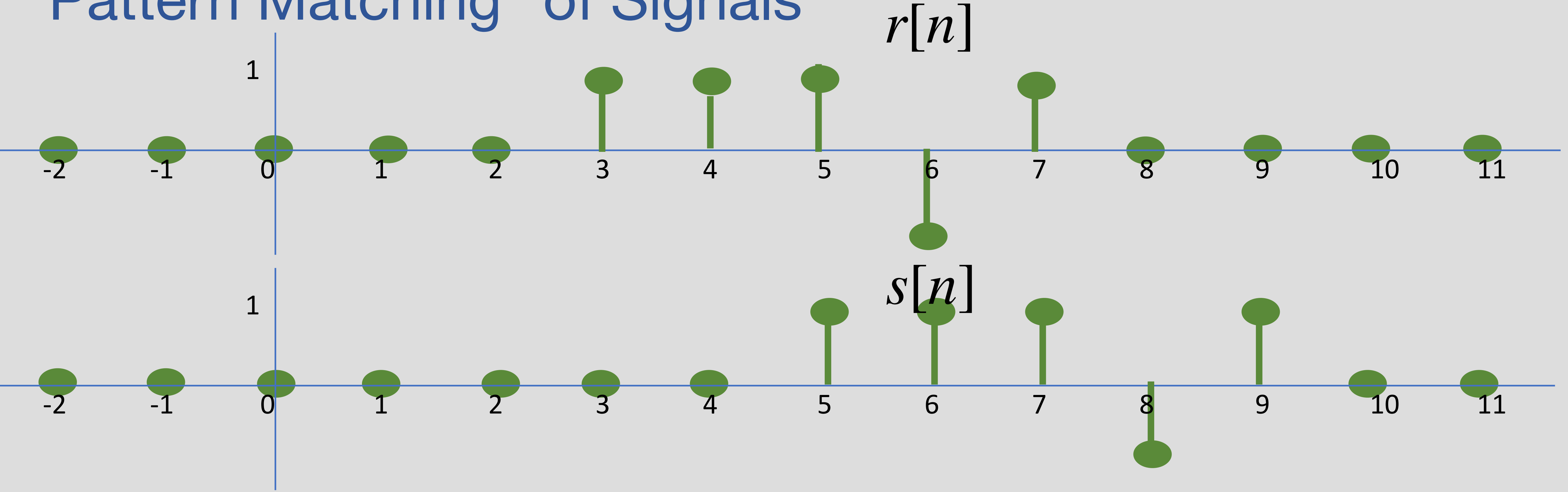




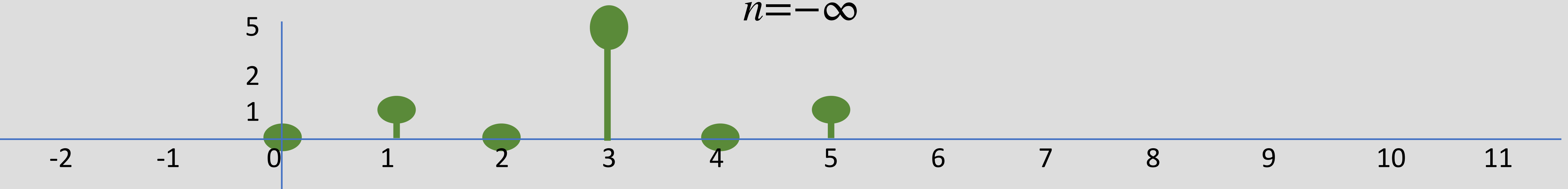
$$\mathrm{corr}_{\vec{r}}(\vec{s})[4] = \langle r[n], s[n-4] \rangle = \sum_{n=-\infty}^{\infty} r[n]s[n-4] = 0$$



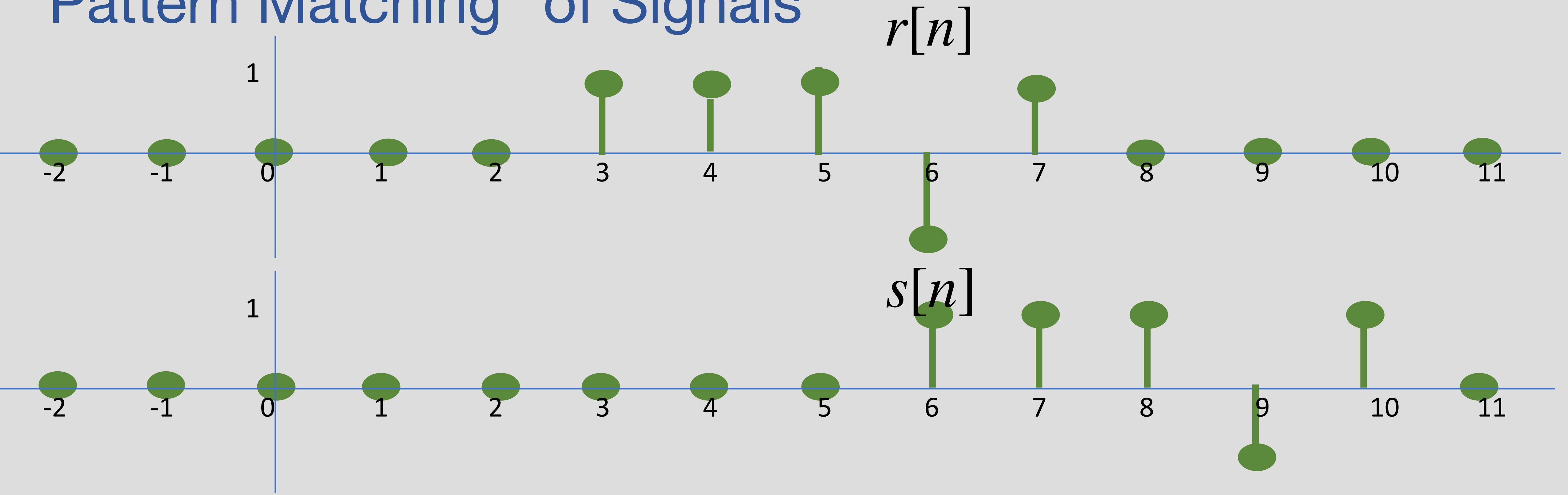
"Pattern Matching" of Signals



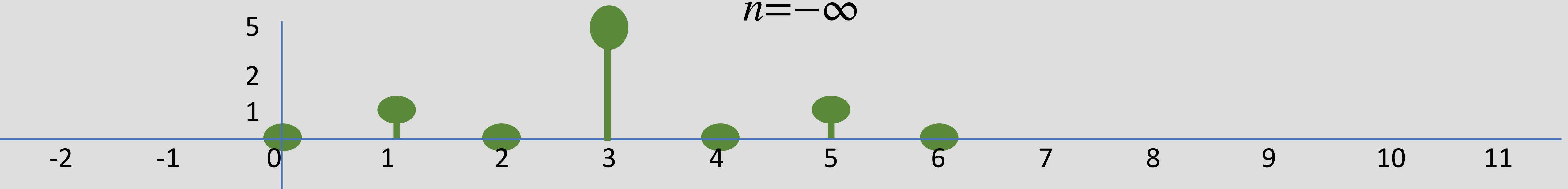
$$\mathrm{corr}_{\vec{r}}(\vec{s})[5] = \langle r[n], s[n-5] \rangle = \sum_{n=-\infty}^{\infty} r[n]s[n-5] = 1$$



"Pattern Matching" of Signals

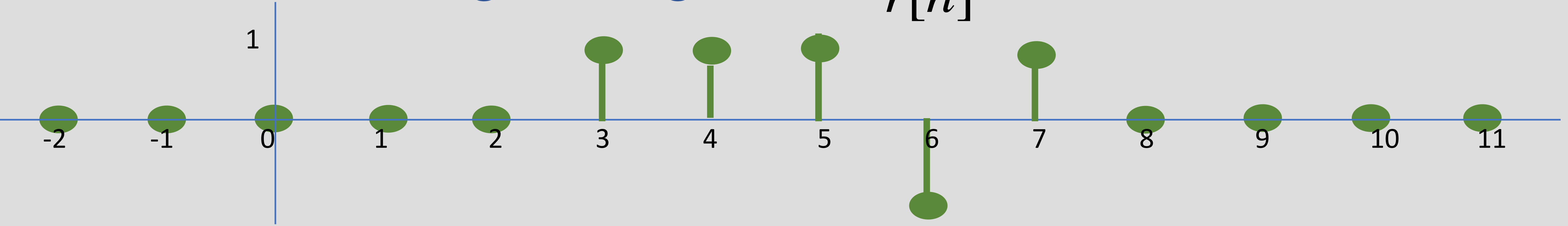


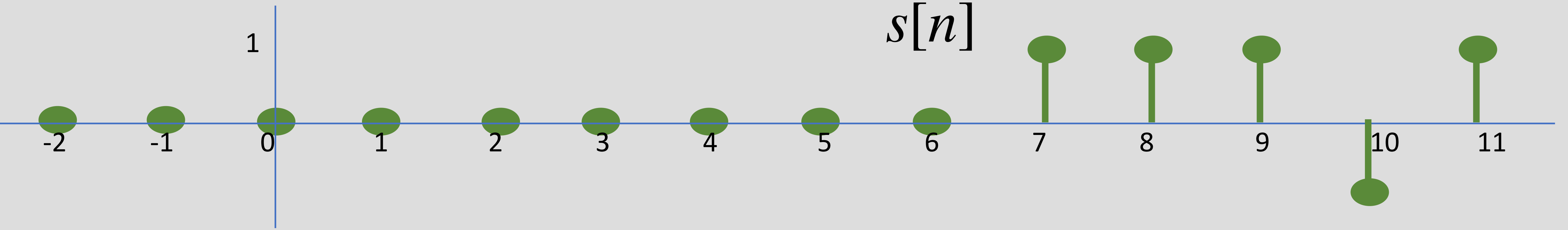
$$\mathrm{corr}_{\vec{r}}(\vec{s})[6] = \langle r[n], s[n-6] \rangle = \sum_{n=-\infty}^{\infty} r[n]s[n-6] = 0$$



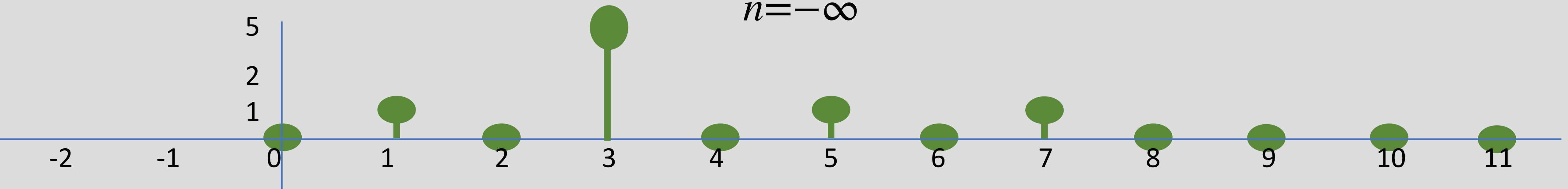
“Pattern Matching” of Signals

$r[n]$

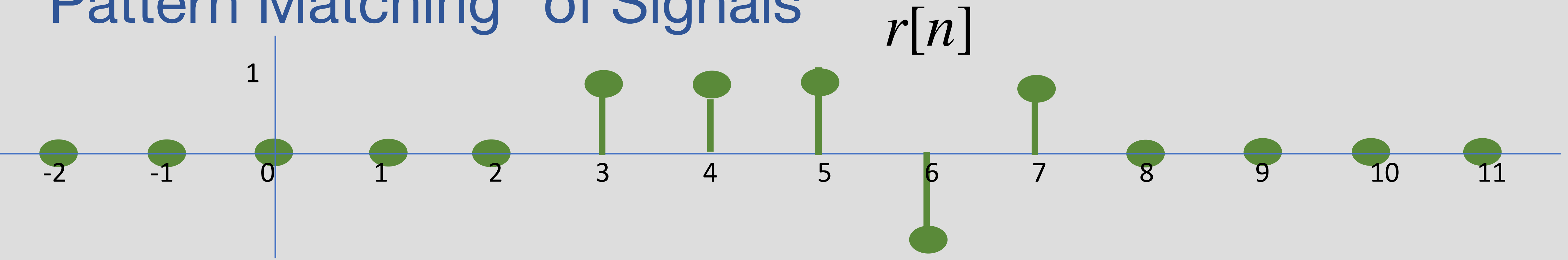


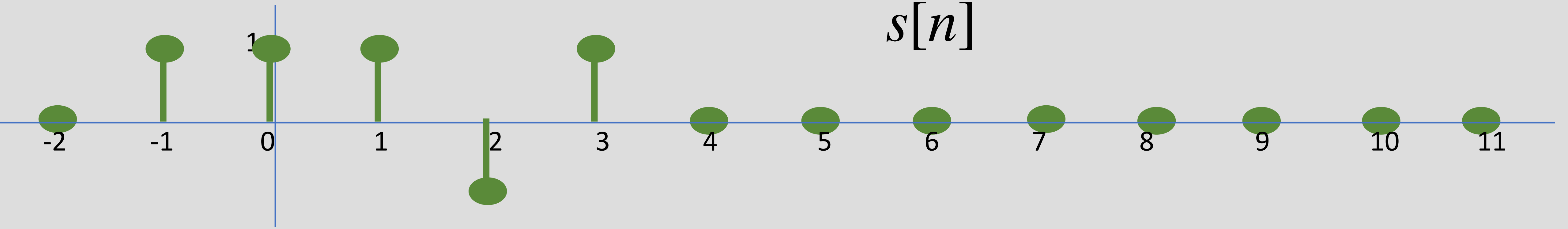


$$\text{corr}_{\vec{r}}(\vec{s})[7] = \langle r[n], s[n-7] \rangle = \sum_{n=-\infty}^{\infty} r[n]s[n-7] = 1$$

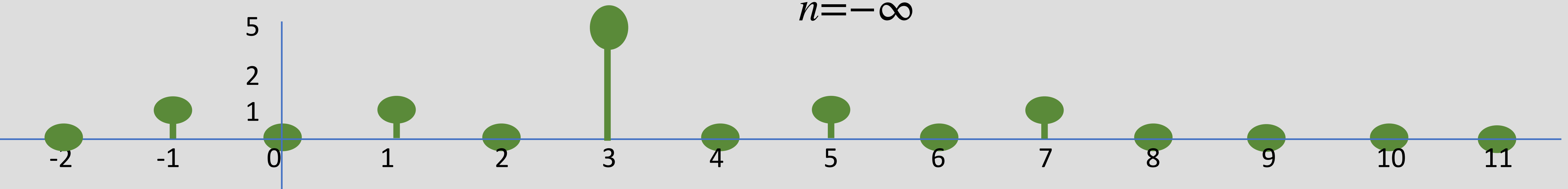


"Pattern Matching" of Signals



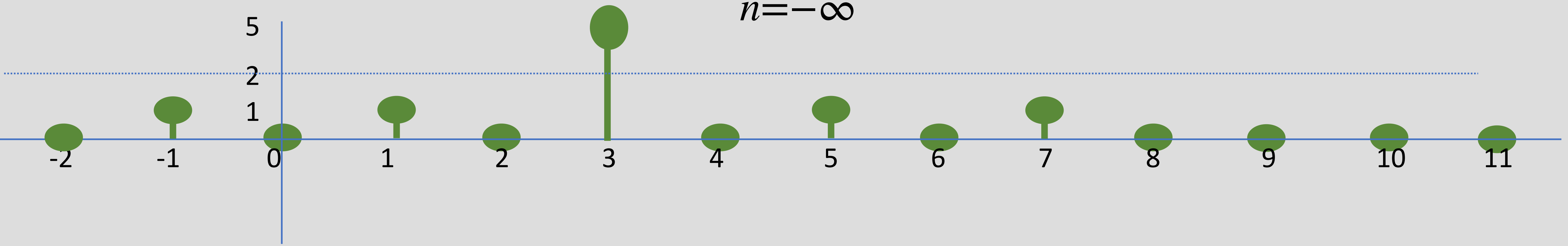


$$\mathrm{corr}_{\vec{r}}(\vec{s})[-1] = \langle r[n], s[n+1] \rangle = \sum_{n=-\infty}^{\infty} r[n]s[n+1] = 1$$



Cross Correlation

$$\text{corr}_{\vec{r}}(\vec{s})[k] = \langle r[n], s[n-k] \rangle = \sum_{n=-\infty}^{\infty} r[n]s[n-k]$$



$$k^* = \underset{k}{\text{argmax}} \ \text{corr}_{\vec{r}}(\vec{s})[k]$$

$$k^* = 3$$

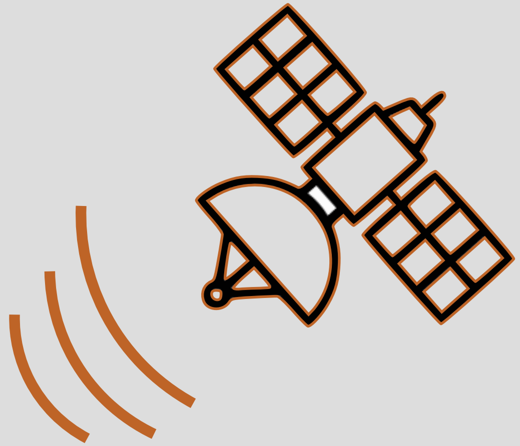
Cross Correlation Properties

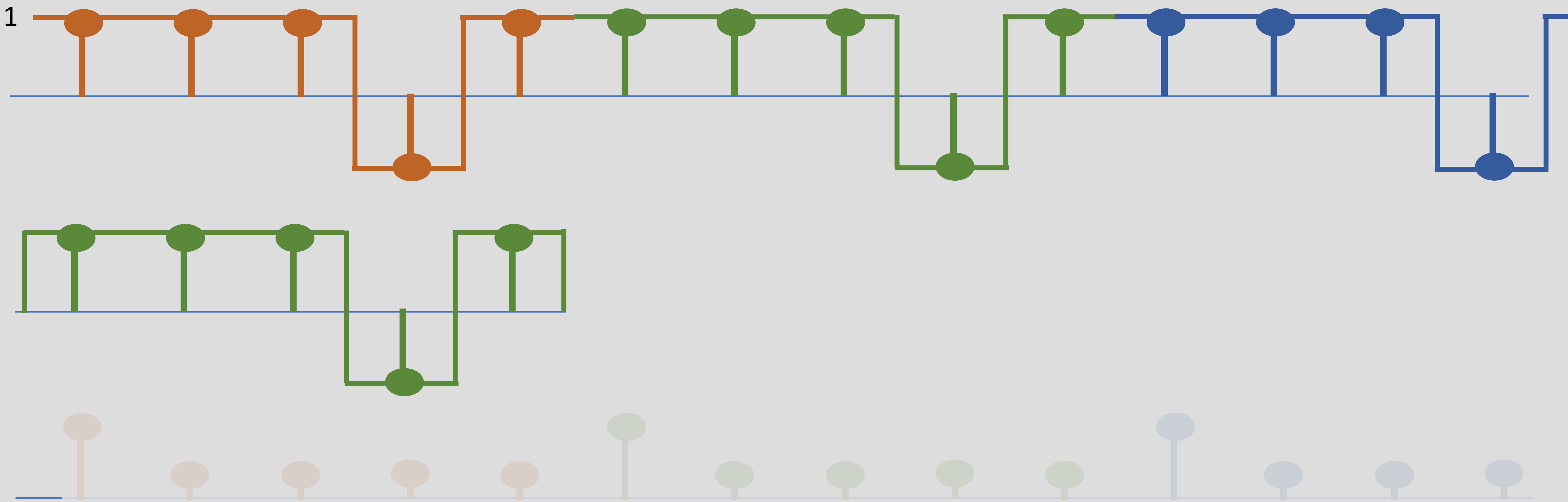
- If $\overrightarrow{x} \in \mathbb{R}^N$, and $\overrightarrow{y} \in \mathbb{R}^M$, then the length of $\mathrm{corr}_{\overrightarrow{x}}(\overrightarrow{y})$ is $N + M - 1$
- $\mathrm{corr}_{\overrightarrow{x}}(\overrightarrow{y}) \neq \mathrm{corr}_{\overrightarrow{y}}(\overrightarrow{x})$
- $\mathrm{corr}_{\overrightarrow{x}}(\overrightarrow{x})$ is called auto-correlation

Periodic Signals

- Satellites repeat the codes over and over
  - Cross correlation is "periodically expanded" instead of zero-padded
  - Result is periodic

TX: $\begin{bmatrix} 1 \\ 1 \\ 1 \\ -1 \\ 1 \end{bmatrix}$



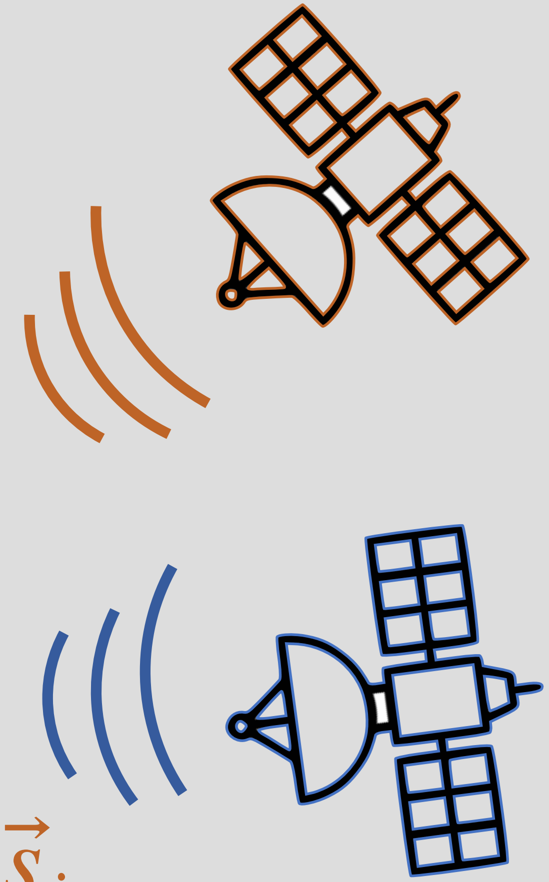


Localization

- Satellites transmit a unique code
  - Radio signal
- Signal is received and digitized by a receiver

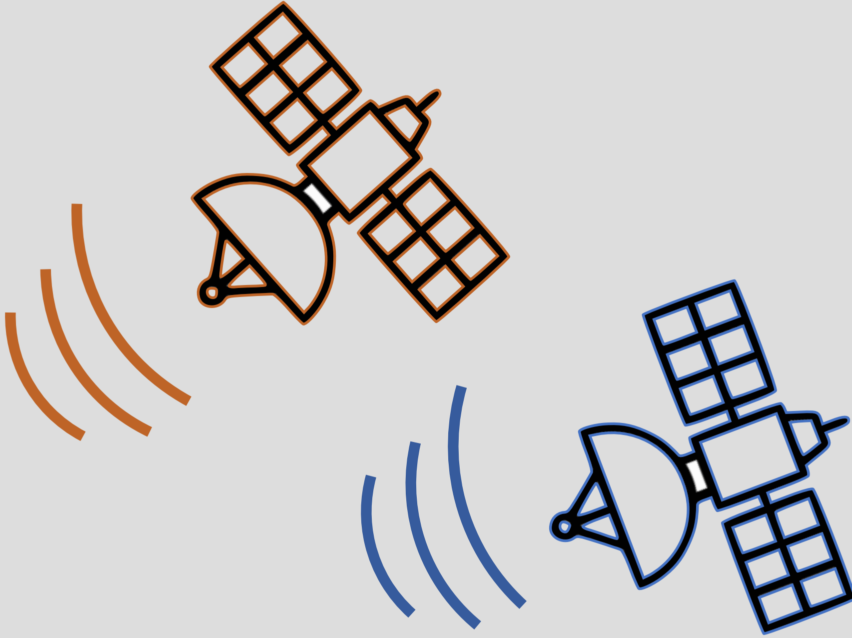
Two problems:

1. Interference
2. Timing



What are good properties for the codes $\vec{s}_i$

Received Signal

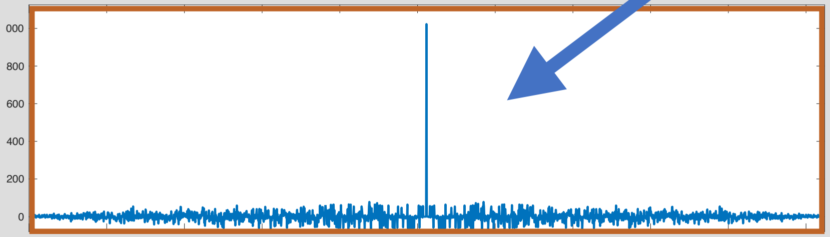


$$r[n] = s_1[n-\tau_1] + s_2[n-\tau_2] + w[n]$$

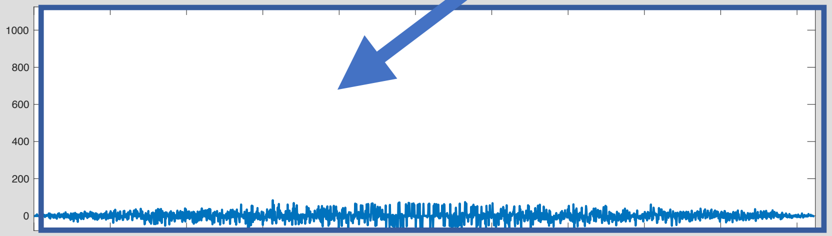
Correlate with $s_1[n]$:

$$\text{corr}_{\vec{r}}(\vec{s}_1)[k] = \langle r[n], s_1[n-k] \rangle$$

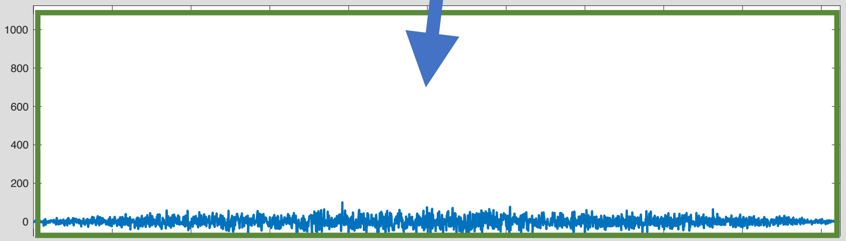
$$= \langle s_1[n-\tau_1], s_1[n-k] \rangle + \langle s_2[n-\tau_2], s_1[n-k] \rangle + \langle w[n], s_1[n-k] \rangle$$



Auto-correlation looks like an impulse



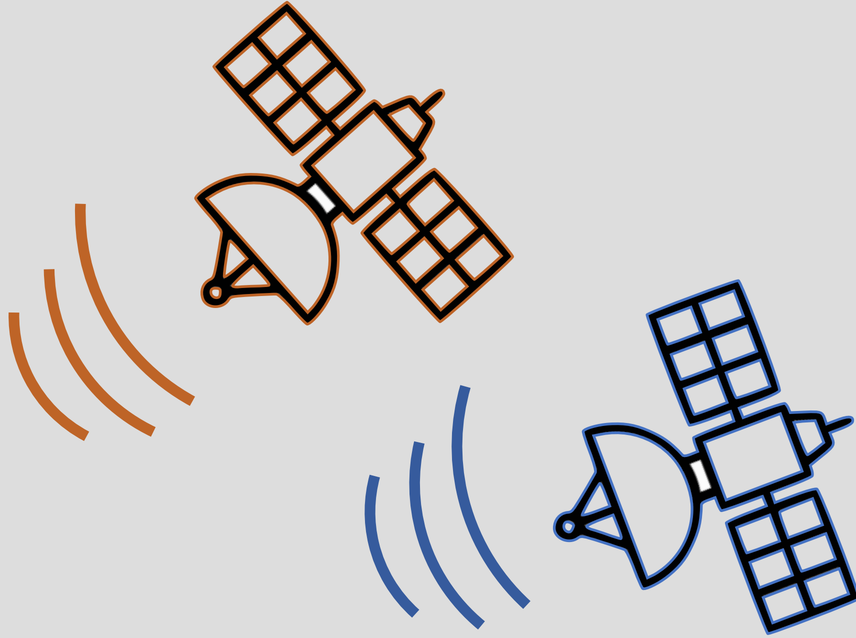
cross-correlation is small

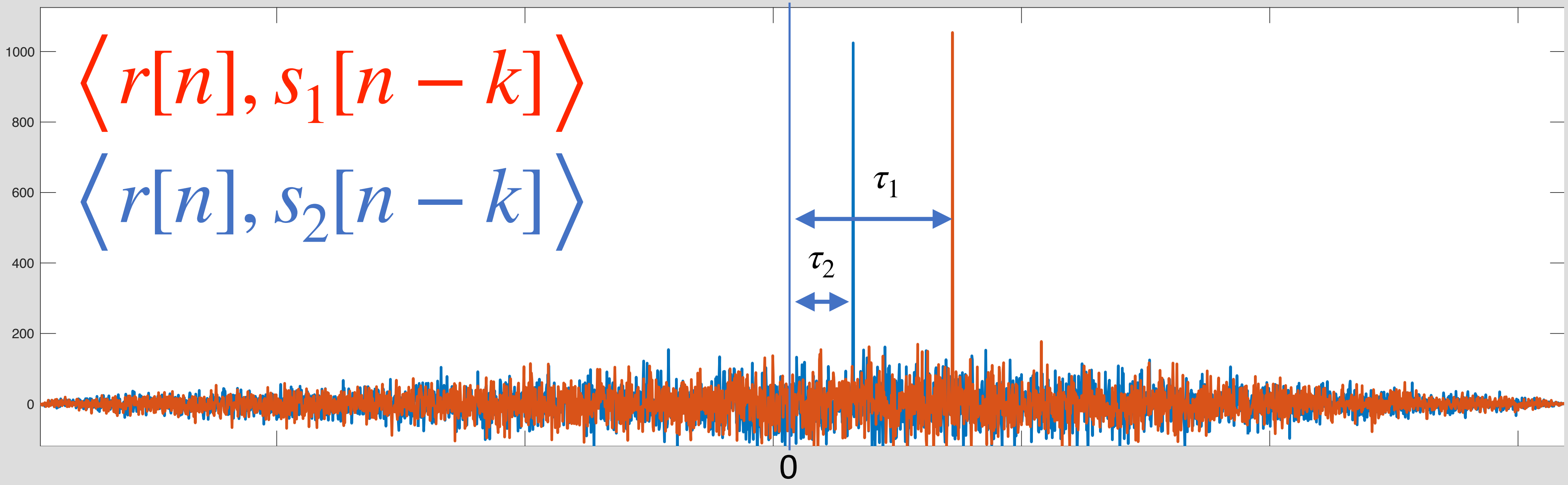


cross-correlation with noise is small
(always true)

Received Signal

$$r[n] = s_1[n - \tau_1] + s_2[n - \tau_2] + w[n]$$



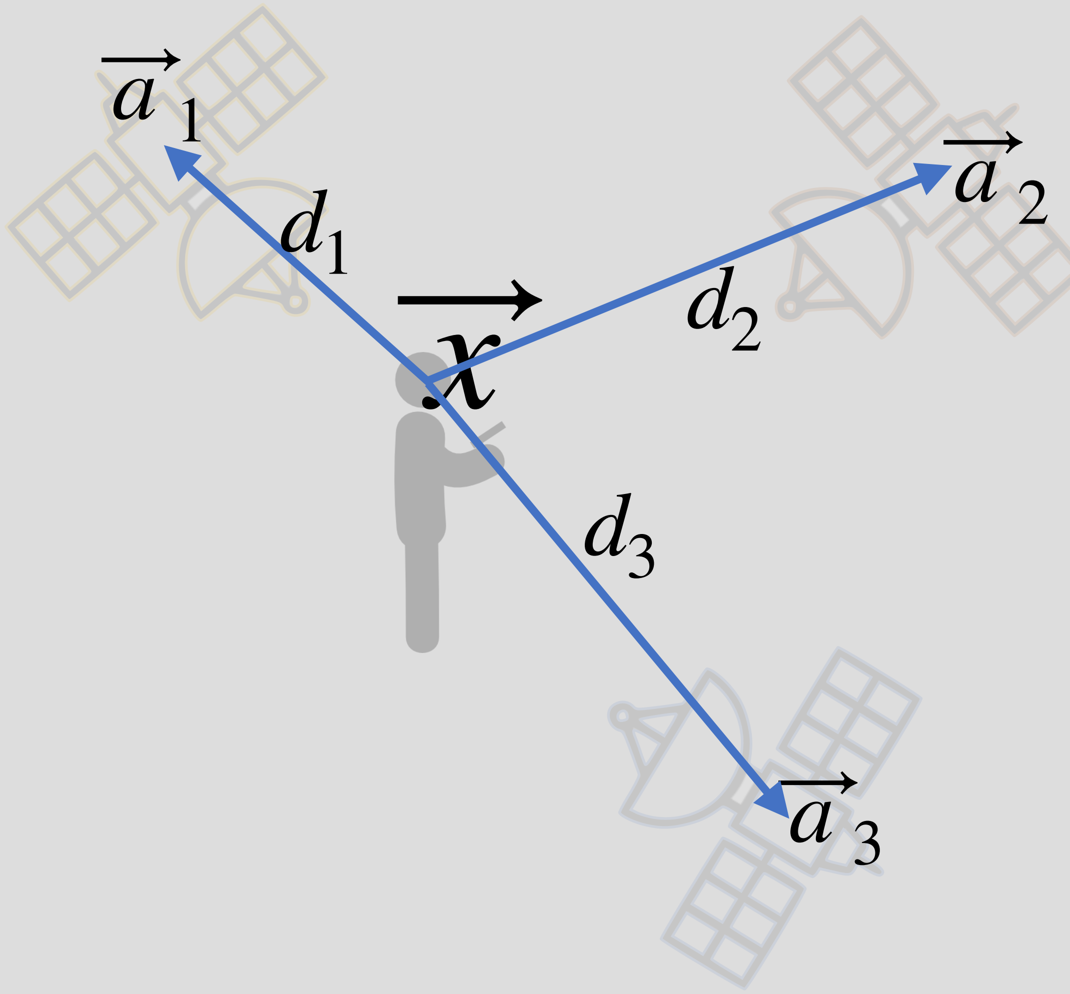


Trilateration

(1) $$\|\vec{x} - \vec{a}_1\|^2 = d_1^2$$

(2) $$\|\vec{x} - \vec{a}_2\|^2 = d_2^2$$

(3) $$\|\vec{x} - \vec{a}_3\|^2 = d_3^2$$



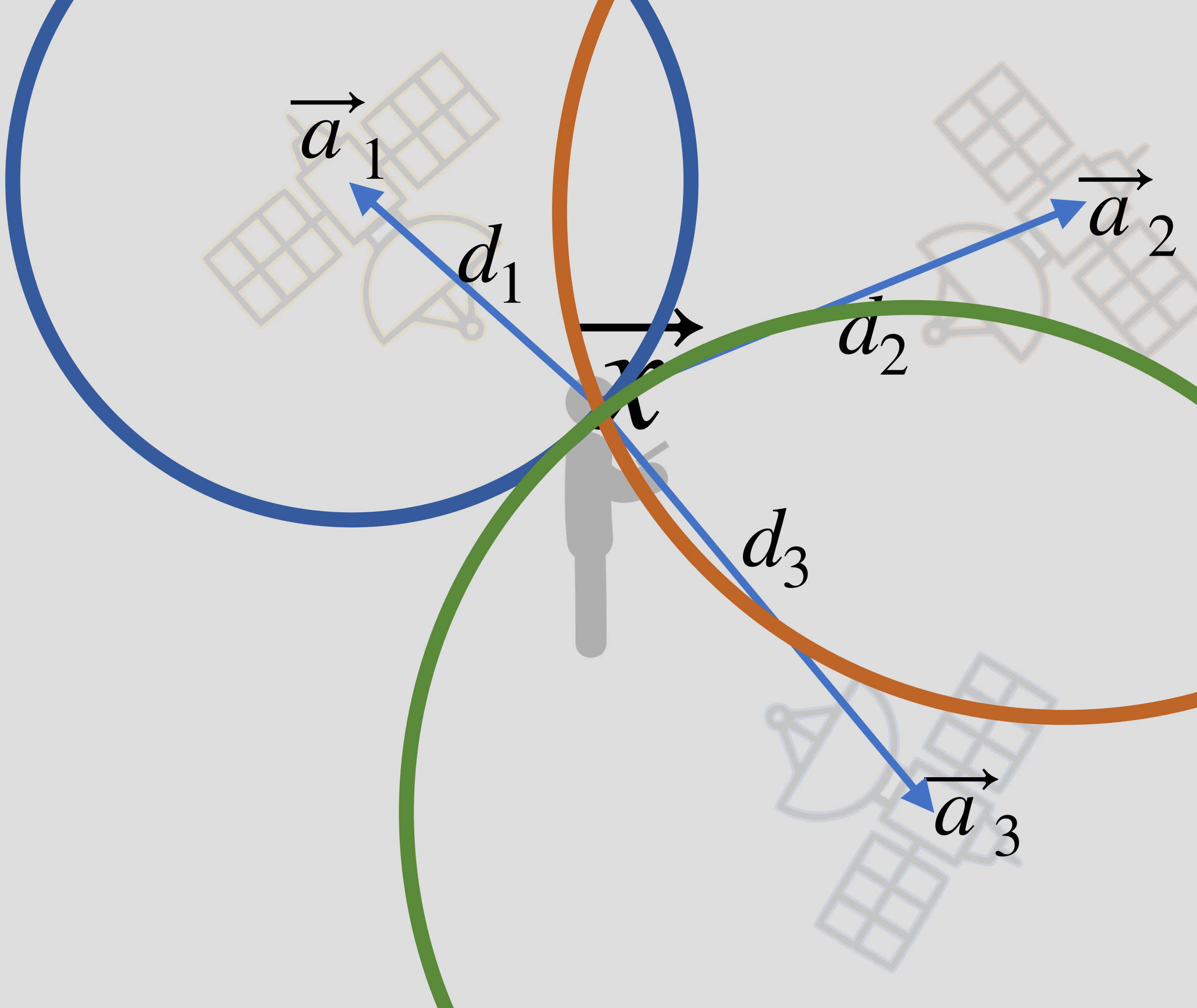
$$d_1 = \tau_1 C$$
$$d_2 = \tau_2 C$$
$$d_3 = \tau_3 C$$

Trilateration

(1) $\|\vec{x} - \vec{a}_1\|^2 = d_1^2$

(2) $\|\vec{x} - \vec{a}_2\|^2 = d_2^2$

(3) $\|\vec{x} - \vec{a}_3\|^2 = d_3^2$



$d_1 = \tau_1 C$

$d_2 = \tau_2 C$

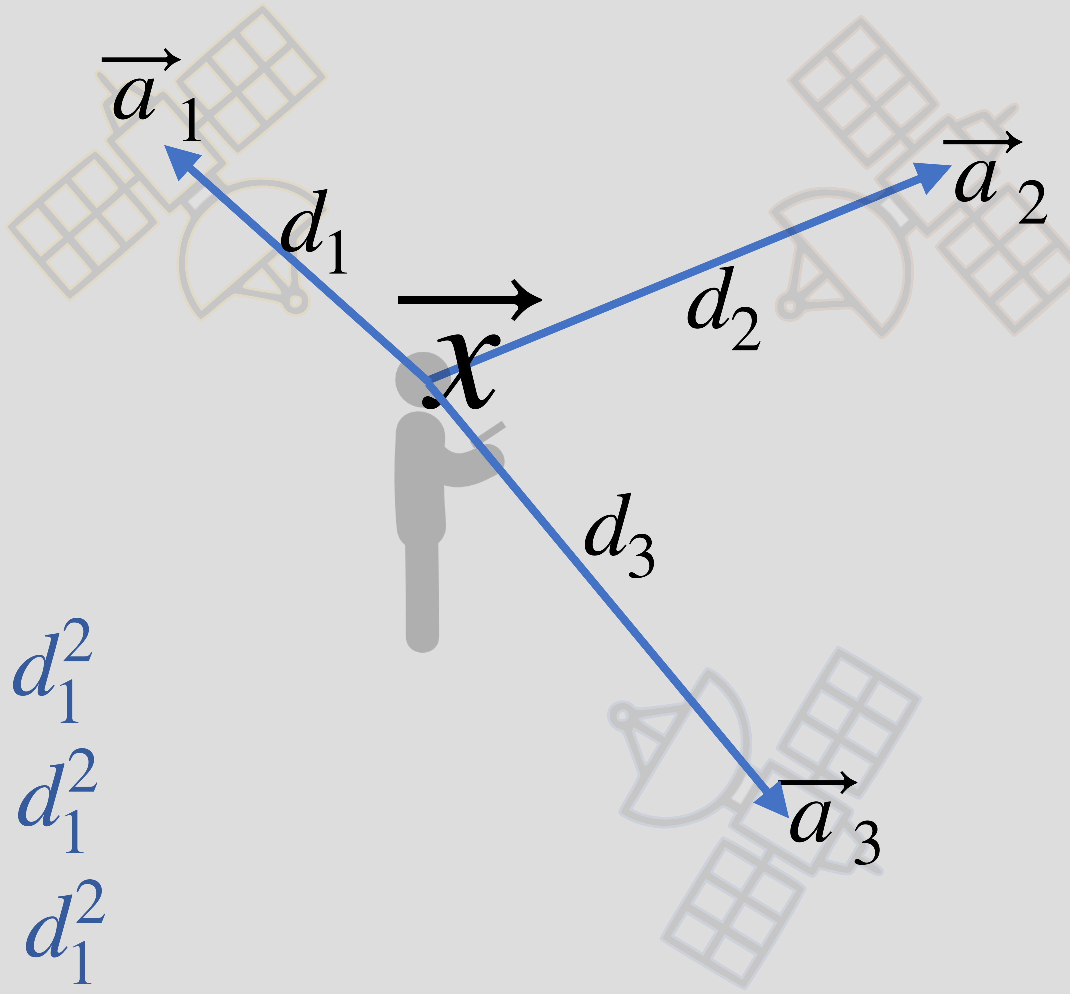
$d_3 = \tau_3 C$

Trilateration

(1) $$\|\vec{x} - \vec{a}_1\|^2 = d_1^2$$

(2) $$\|\vec{x} - \vec{a}_2\|^2 = d_2^2$$

(3) $$\|\vec{x} - \vec{a}_3\|^2 = d_3^2$$



$$\|\vec{x} - \vec{a}_1\|^2 = d_1^2$$

$$(\vec{x} - \vec{a}_1)^T(\vec{x} - \vec{a}_1) = d_1^2$$

$$\vec{x}^T\vec{x} - \vec{a}_1^T\vec{x} - \vec{x}^T\vec{a}_1 + \vec{a}_1^T\vec{a}_1 = d_1^2$$

$$\|\vec{x}\|^2 - 2\vec{a}_1^T\vec{x} + \|\vec{a}_1\|^2 = d_1^2$$

$$\|\vec{x}\|^2 - 2\vec{a}_1^T\vec{x} + \|\vec{a}_1\|^2 = C^2\tau_1^2$$

$$d_1 = \tau_1 C$$

$$d_2 = \tau_2 C$$

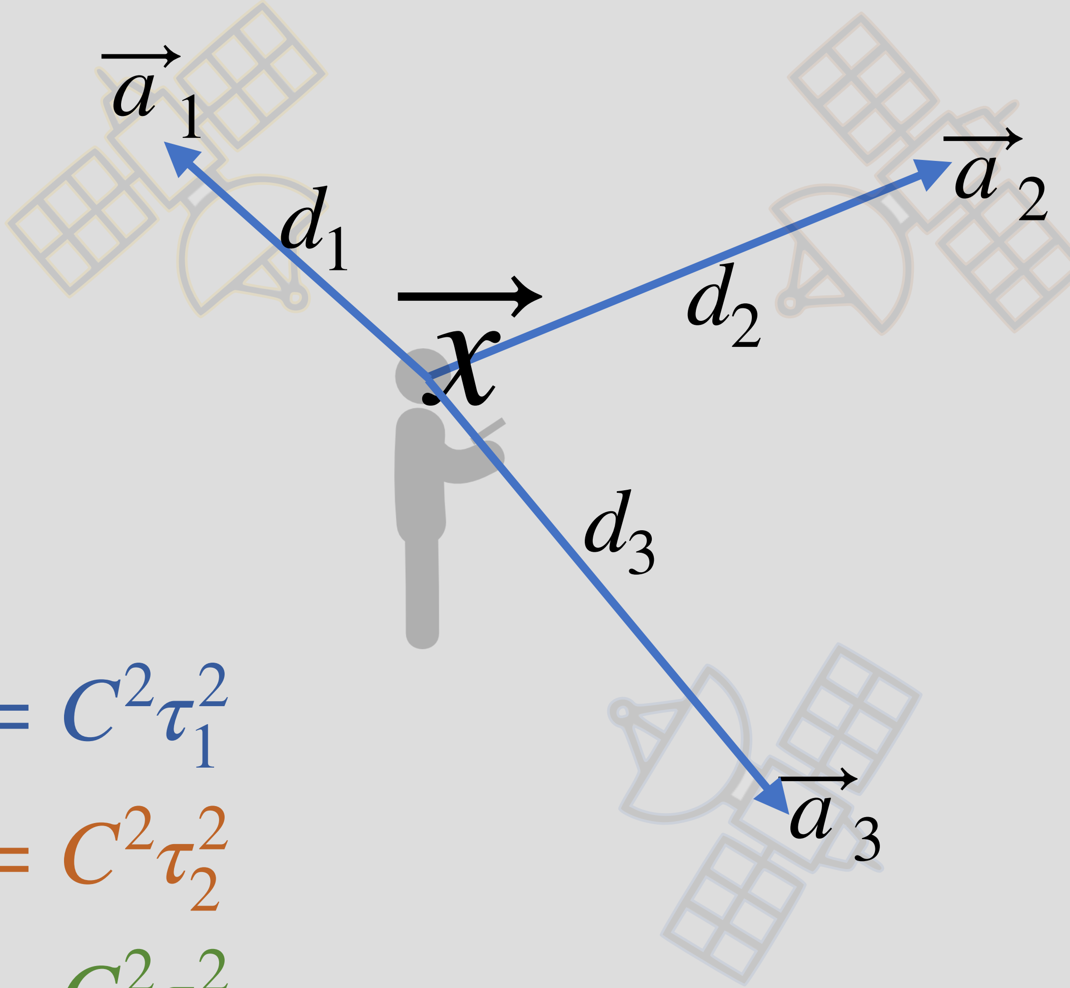
$$d_3 = \tau_3 C$$

Trilateration

(1) $$\|\vec{x} - \vec{a}_1\|^2 = d_1^2$$

(2) $$\|\vec{x} - \vec{a}_2\|^2 = d_2^2$$

(3) $$\|\vec{x} - \vec{a}_3\|^2 = d_3^2$$



(1) $$\|\vec{x}\|^2 - 2\vec{a}_1^T\vec{x} + \|\vec{a}_1\|^2 = C^2\tau_1^2$$

(2) $$\|\vec{x}\|^2 - 2\vec{a}_2^T\vec{x} + \|\vec{a}_2\|^2 = C^2\tau_2^2$$

(3) $$\|\vec{x}\|^2 - 2\vec{a}_3^T\vec{x} + \|\vec{a}_3\|^2 = C^2\tau_3^2$$

$$d_1 = \tau_1 C$$
$$d_2 = \tau_2 C$$
$$d_3 = \tau_3 C$$

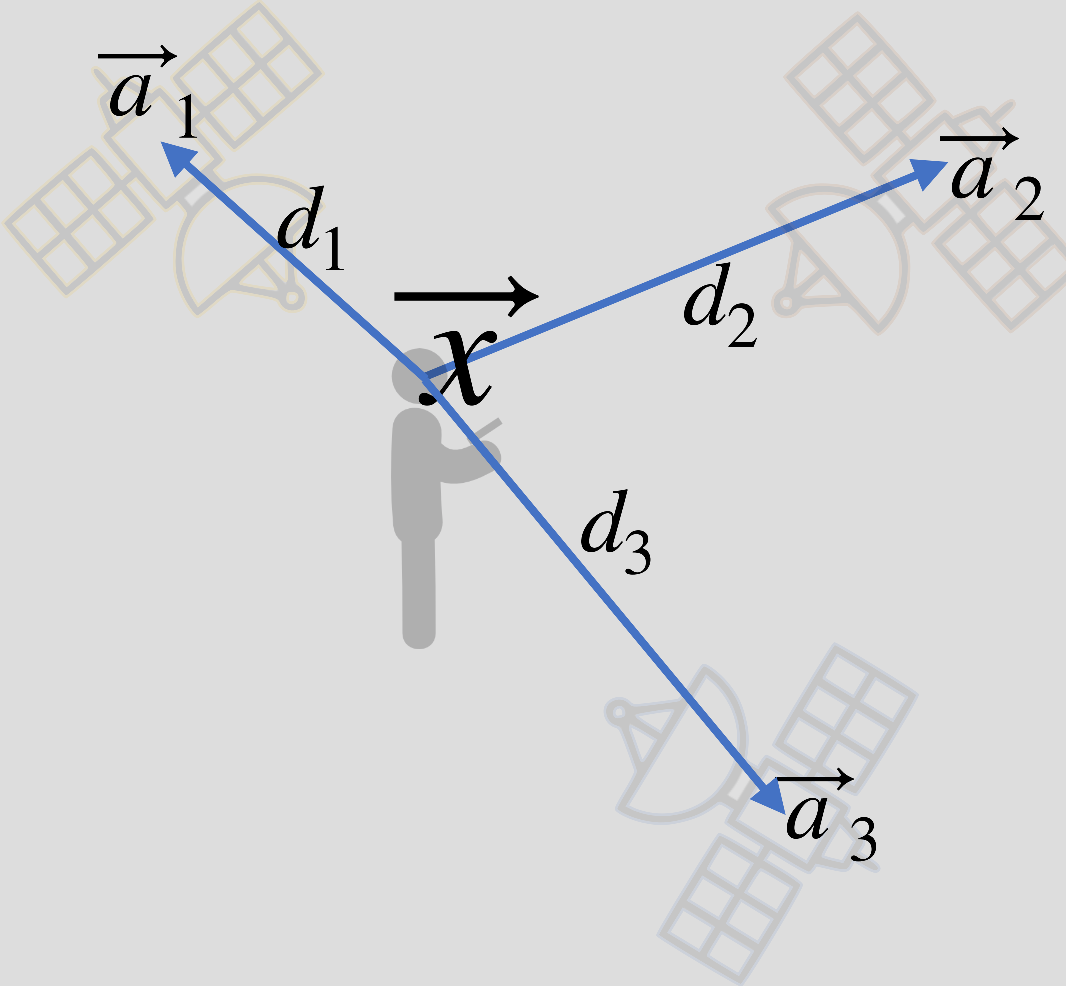
Trilateration

(1) $\|\vec{x}\|^2 - 2\vec{a}_1^T\vec{x} + \|\vec{a}_1\|^2 = C^2\tau_1^2$

(2) $\|\vec{x}\|^2 - 2\vec{a}_2^T\vec{x} + \|\vec{a}_2\|^2 = C^2\tau_2^2$

(3) $\|\vec{x}\|^2 - 2\vec{a}_3^T\vec{x} + \|\vec{a}_3\|^2 = C^2\tau_3^2$

(2) - (1)

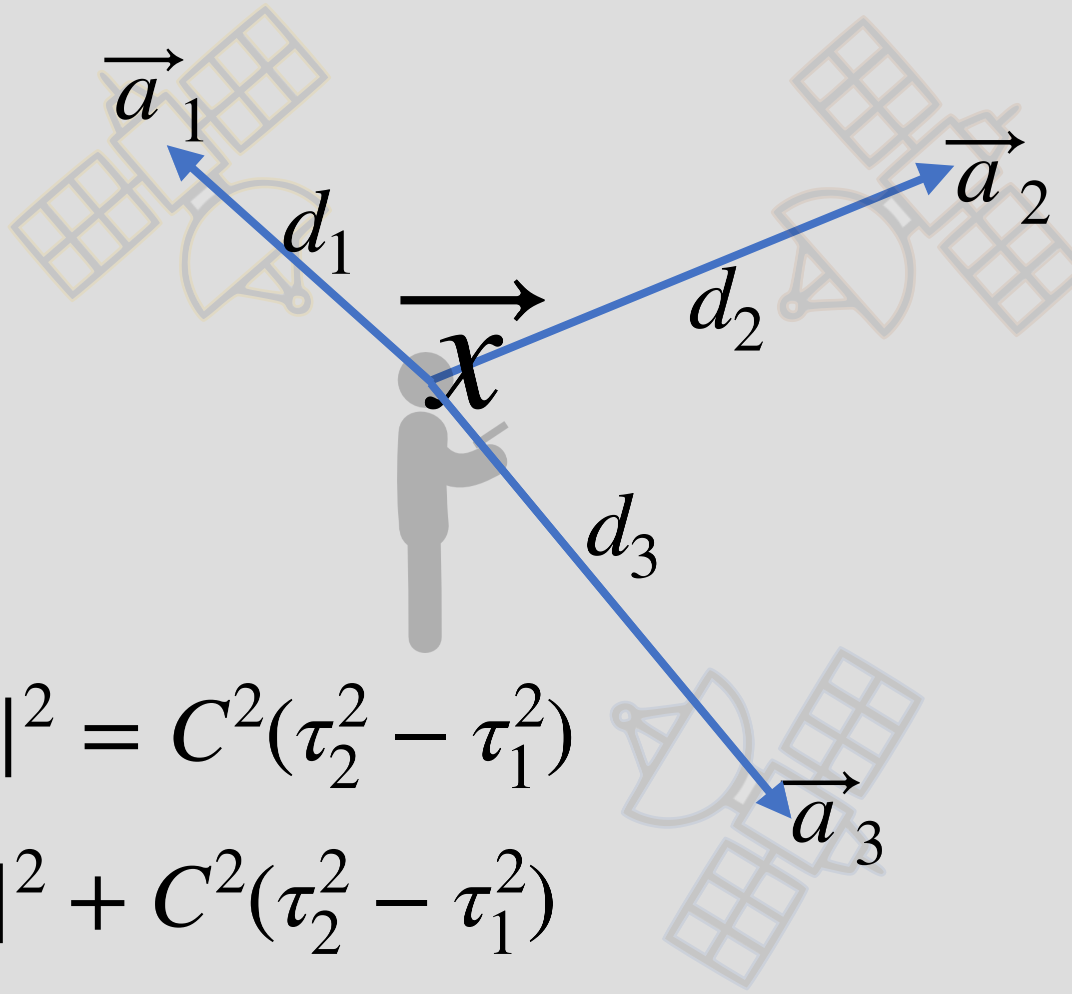


Trilateration

(1) $\|\vec{x}\|^2 - 2\vec{a}_1^T\vec{x} + \|\vec{a}_1\|^2 = C^2\tau_1^2$

(2) $\|\vec{x}\|^2 - 2\vec{a}_2^T\vec{x} + \|\vec{a}_2\|^2 = C^2\tau_2^2$

(3) $\|\vec{x}\|^2 - 2\vec{a}_3^T\vec{x} + \|\vec{a}_3\|^2 = C^2\tau_3^2$



(2) - (1)

$$-2\vec{a}_2^T\vec{x} + 2\vec{a}_1^T\vec{x} + \|\vec{a}_2\|^2 - \|\vec{a}_1\|^2 = C^2(\tau_2^2 - \tau_1^2)$$

$$2(\vec{a}_1 - \vec{a}_2)^T\vec{x} = \|\vec{a}_1\|^2 - \|\vec{a}_2\|^2 + C^2(\tau_2^2 - \tau_1^2)$$

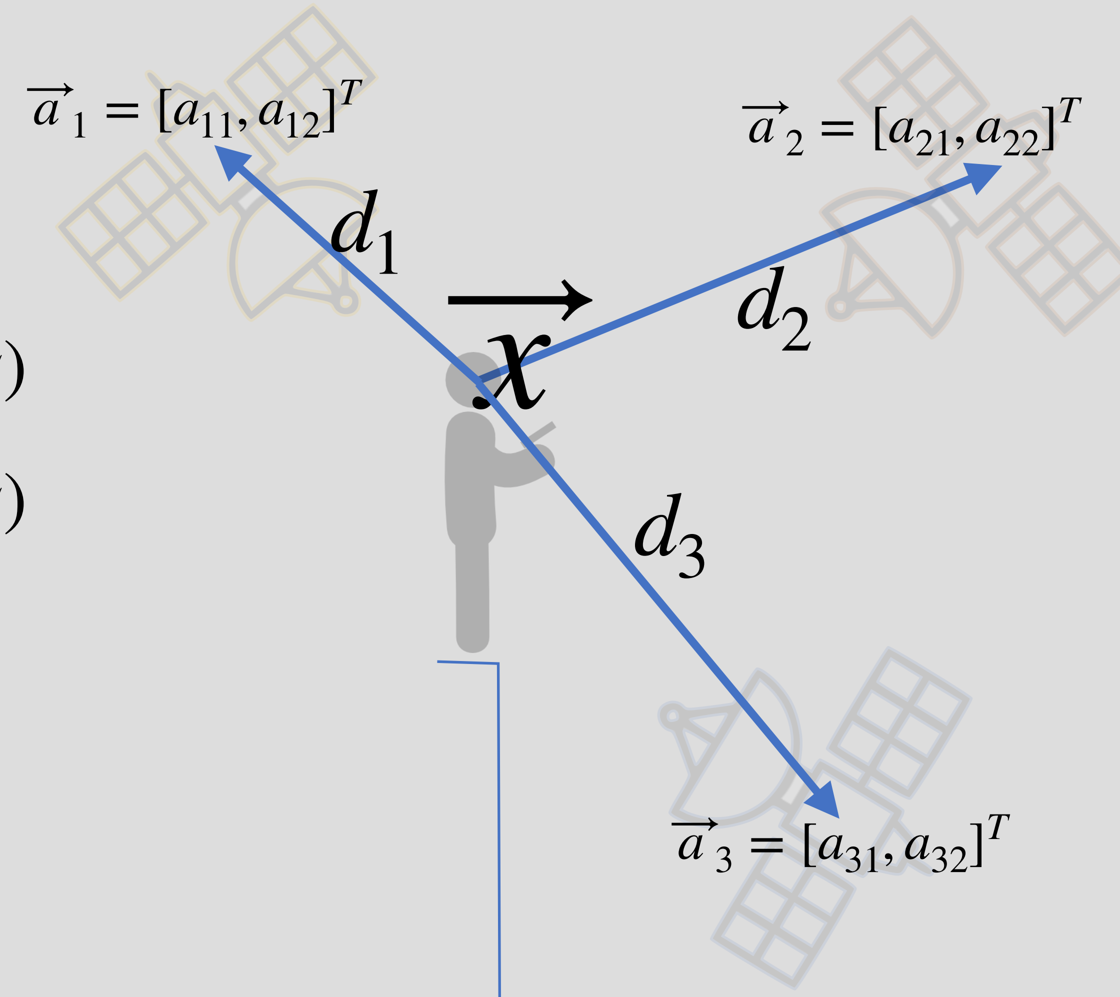
(3) - (1)

$$2(\vec{a}_1 - \vec{a}_3)^T\vec{x} = \|\vec{a}_1\|^2 - \|\vec{a}_3\|^2 + C^2(\tau_3^2 - \tau_1^2)$$

Trilateration

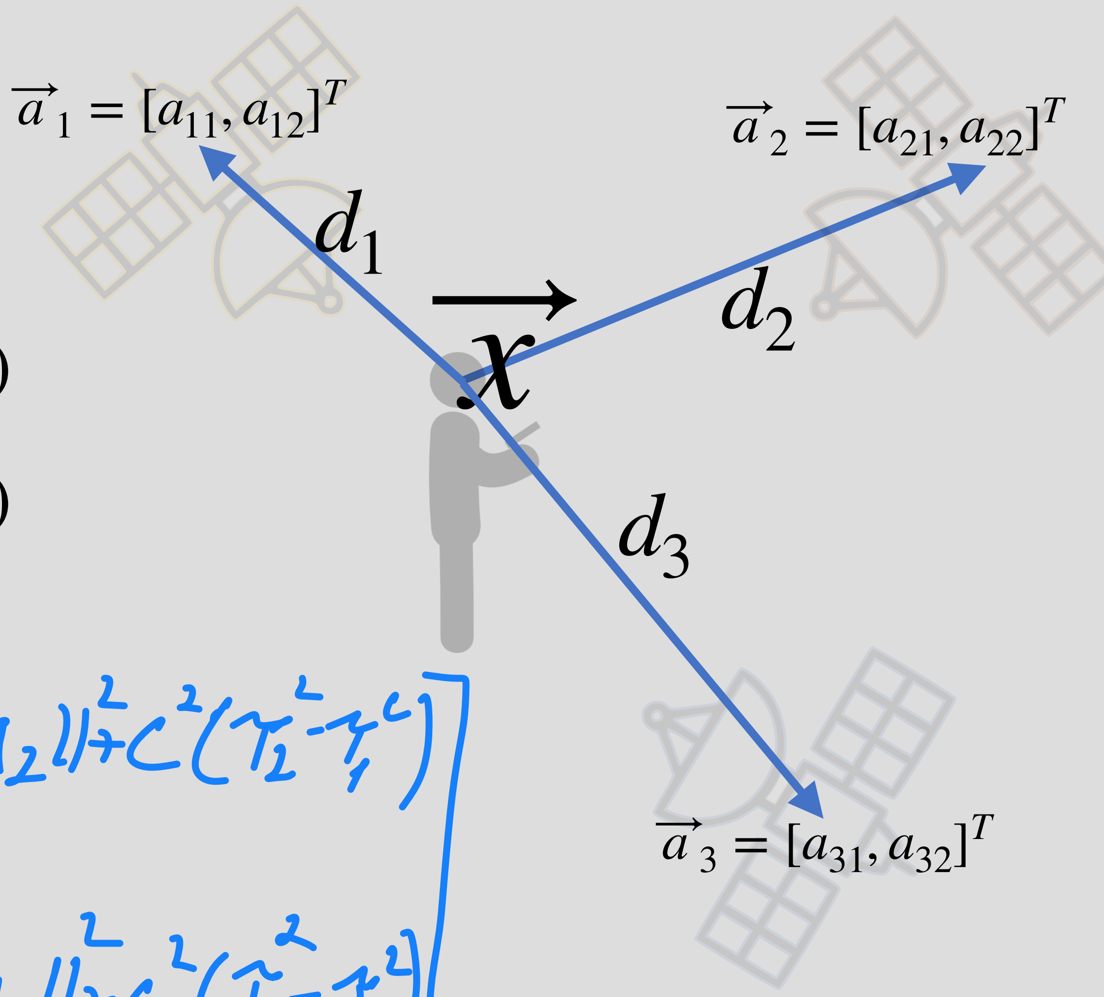
$$2(\vec{a}_1 - \vec{a}_2)^T \vec{x} = \|\vec{a}_1\|^2 - \|\vec{a}_2\|^2 + C^2(\tau_2^2 - \tau_1^2)$$

$$2(\vec{a}_1 - \vec{a}_3)^T \vec{x} = \|\vec{a}_1\|^2 - \|\vec{a}_3\|^2 + C^2(\tau_3^2 - \tau_1^2)$$



=

Trilateration



$$2(\vec{a}_1 - \vec{a}_2)^T \vec{x} = \|\vec{a}_1\|^2 - \|\vec{a}_2\|^2 + C^2(\tau_2^2 - \tau_1^2)$$

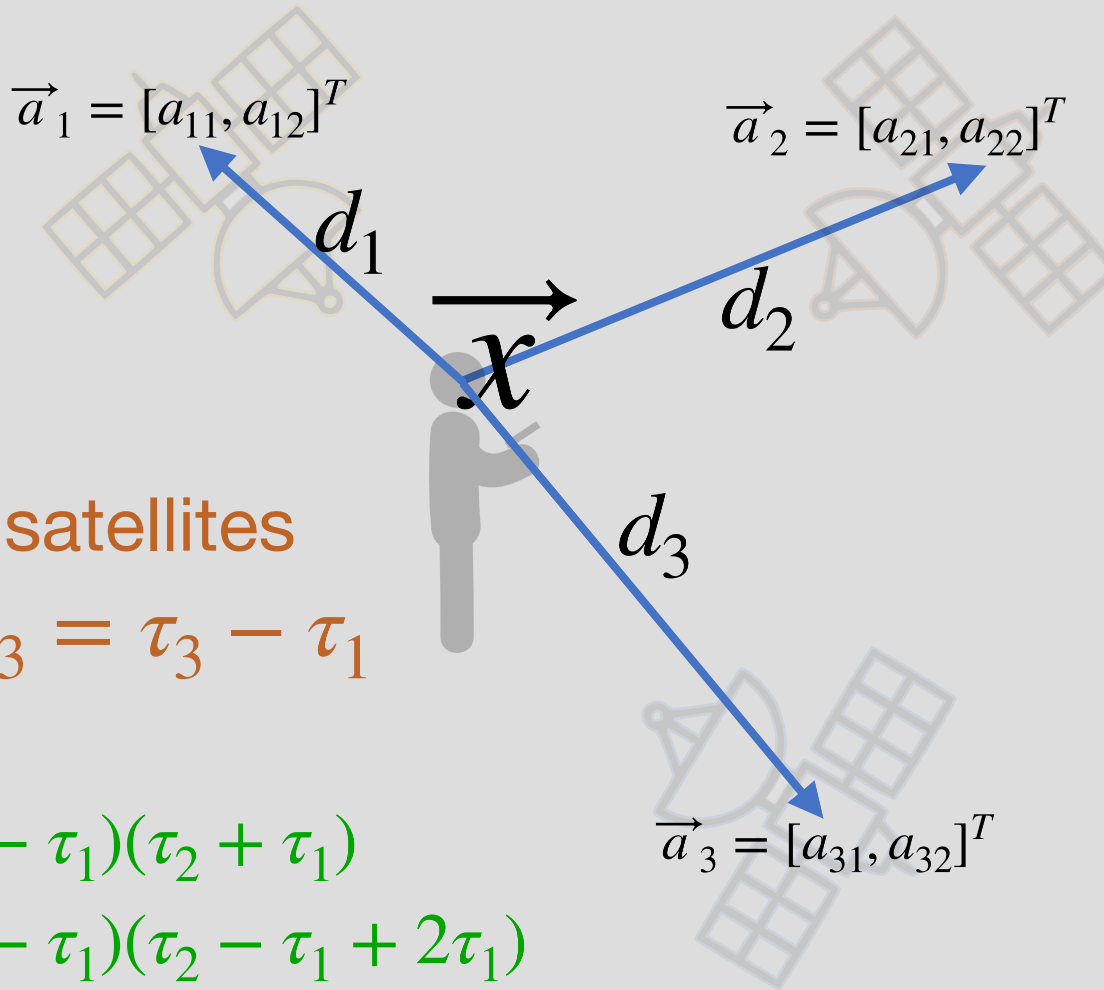
$$2(\vec{a}_1 - \vec{a}_3)^T \vec{x} = \|\vec{a}_1\|^2 - \|\vec{a}_3\|^2 + C^2(\tau_3^2 - \tau_1^2)$$

$$2\begin{bmatrix} a_{11} - a_{21} & a_{12} - a_{22} \\ a_{11} - a_{31} & a_{12} - a_{32} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \|a_1\|^2 - \|a_2\|^2 + c^2(\tau_2^2 - \tau_1^2) \\ \|a_1\|^2 - \|a_3\|^2 + c^2(\tau_3^2 - \tau_1^2) \end{bmatrix}$$

Solve via gaussian elimination!

Trilateration

$$2(\vec{a}_1 - \vec{a}_2)^T\vec{x} = \|\vec{a}_1\|^2 - \|\vec{a}_2\|^2 + C^2(\tau_2^2 - \tau_1^2)$$
$$2(\vec{a}_1 - \vec{a}_3)^T\vec{x} = \|\vec{a}_1\|^2 - \|\vec{a}_3\|^2 + C^2(\tau_3^2 - \tau_1^2)$$



Problem — receiver clock is not synced to satellites

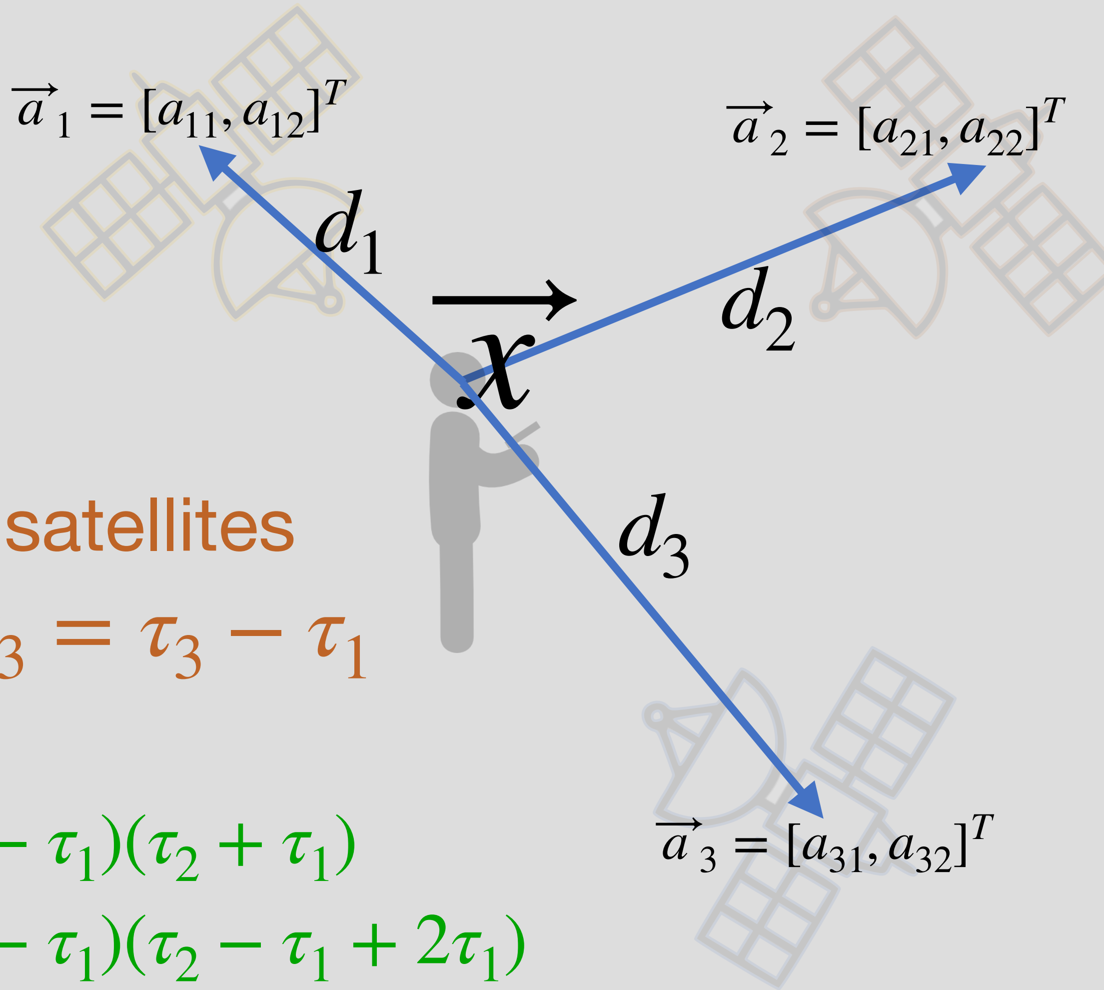
τ_1 is unknown, but $\Delta\tau_2 = \tau_2 - \tau_1$, and $\Delta\tau_3 = \tau_3 - \tau_1$ are known

$$2(\vec{a}_1 - \vec{a}_2)^T\vec{x} = \|\vec{a}_1\|^2 - \|\vec{a}_2\|^2 + C^2(\tau_2 - \tau_1)(\tau_2 + \tau_1)$$
$$2(\vec{a}_1 - \vec{a}_2)^T\vec{x} = \|\vec{a}_1\|^2 - \|\vec{a}_2\|^2 + C^2(\tau_2 - \tau_1)(\tau_2 - \tau_1 + 2\tau_1)$$
$$2(\vec{a}_1 - \vec{a}_2)^T\vec{x} = \|\vec{a}_1\|^2 - \|\vec{a}_2\|^2 + C^2(\Delta\tau_2)(\Delta\tau_2 + 2\tau_1)$$

Trilateration

$$2(\vec{a}_1 - \vec{a}_2)^T \vec{x} = \|\vec{a}_1\|^2 - \|\vec{a}_2\|^2 + C^2(\tau_2^2 - \tau_1^2)$$

$$2(\vec{a}_1 - \vec{a}_3)^T \vec{x} = \|\vec{a}_1\|^2 - \|\vec{a}_3\|^2 + C^2(\tau_3^2 - \tau_1^2)$$



Problem — receiver clock is not synced to satellites

τ_1 is unknown, but $\Delta\tau_2 = \tau_2 - \tau_1$, and $\Delta\tau_3 = \tau_3 - \tau_1$ are known

$$2(\vec{a}_1 - \vec{a}_2)^T \vec{x} = \|\vec{a}_1\|^2 - \|\vec{a}_2\|^2 + C^2(\tau_2 - \tau_1)(\tau_2 + \tau_1)$$

$$2(\vec{a}_1 - \vec{a}_2)^T \vec{x} = \|\vec{a}_1\|^2 - \|\vec{a}_2\|^2 + C^2(\tau_2 - \tau_1)(\tau_2 - \tau_1 + 2\tau_1)$$

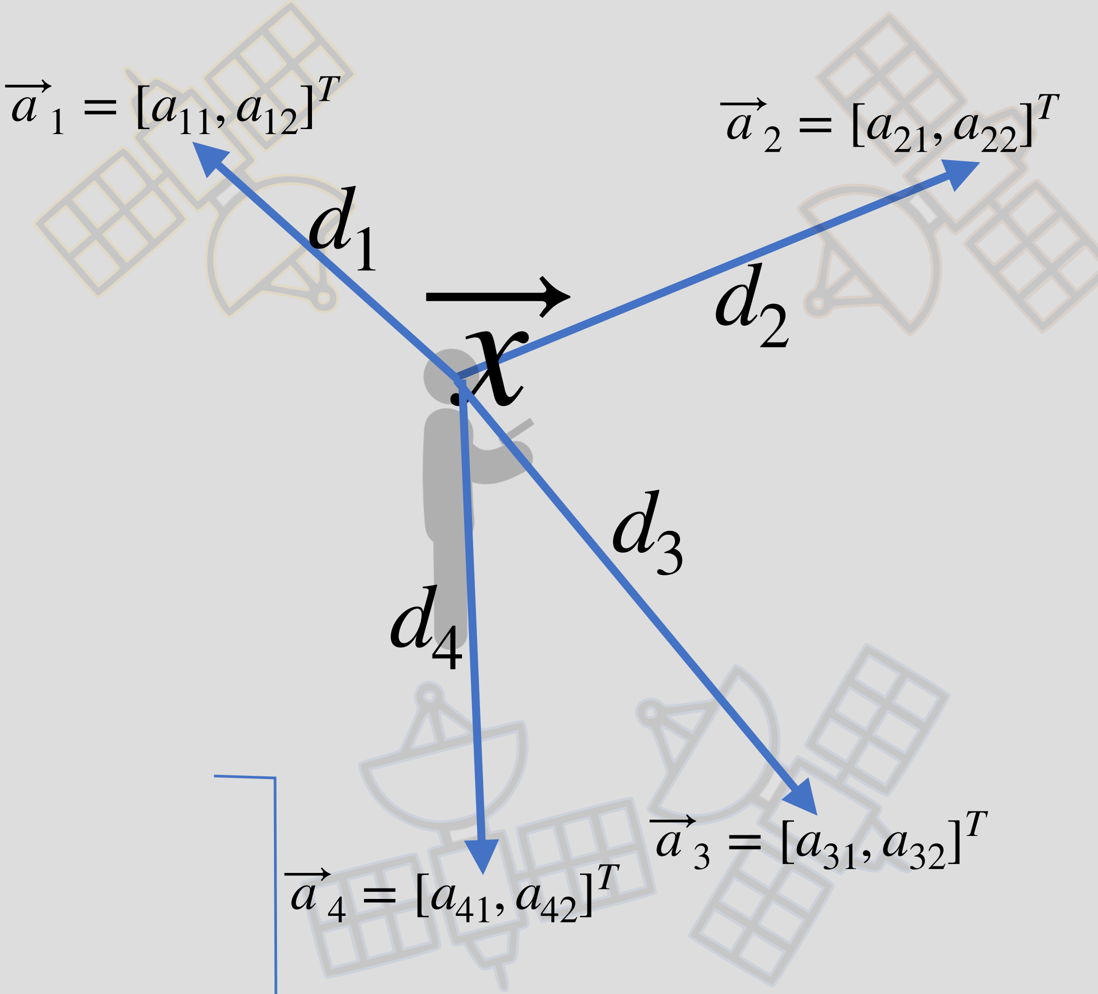
$$2(\vec{a}_1 - \vec{a}_2)^T \vec{x} = \|\vec{a}_1\|^2 - \|\vec{a}_2\|^2 + C^2(\Delta\tau_2)(\Delta\tau_2 + 2\tau_1)$$

$$2(\vec{a}_1 - \vec{a}_2)^T \vec{x} - 2C^2\Delta\tau_2\tau_1 = \|\vec{a}_1\|^2 - \|\vec{a}_2\|^2 + C^2(\Delta\tau_2)^2$$

Another variable! Need 1 more equation (satellite)

Trilateration

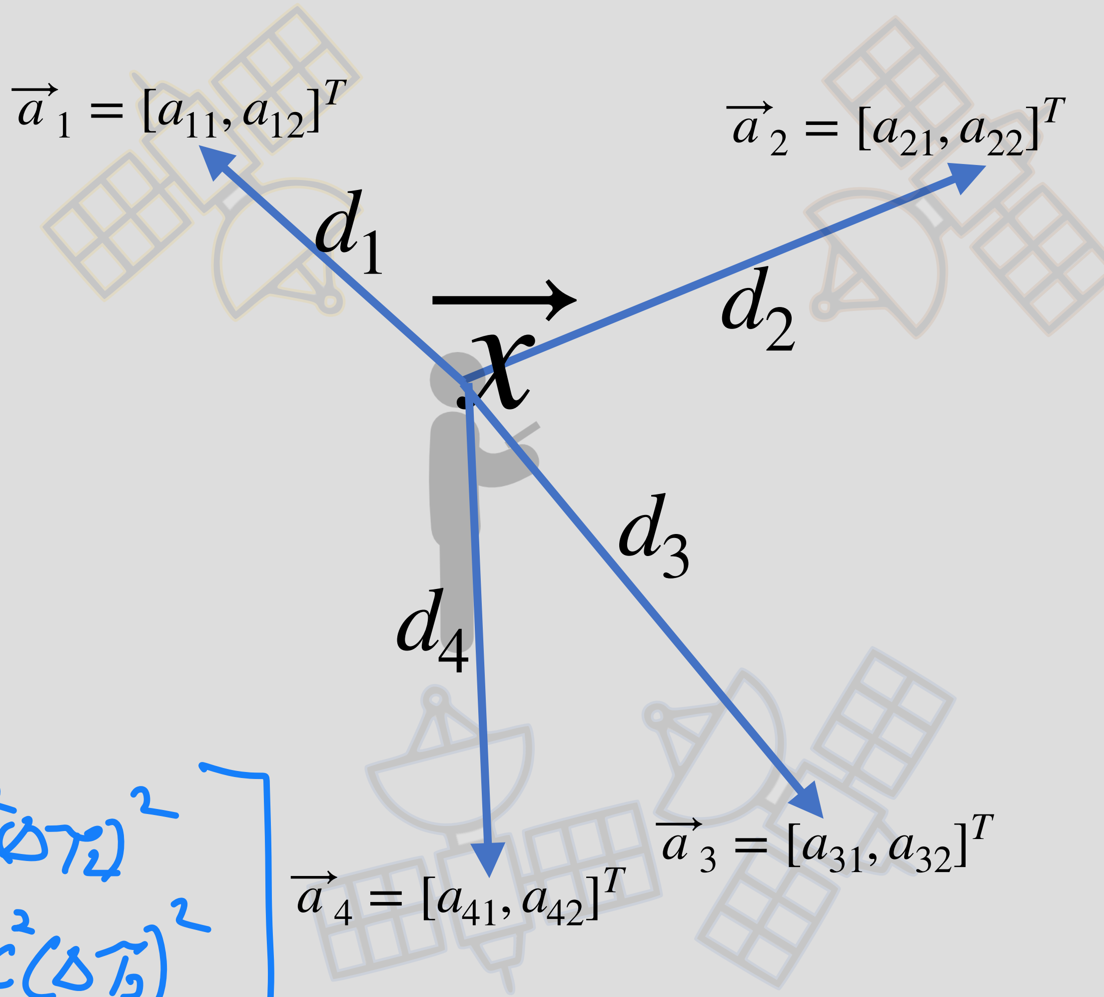
$$2(\vec{a}_1 - \vec{a}_2)^T \vec{x} - 2C^2 \Delta\tau_2 \tau_1 = \|\vec{a}_1\|^2 - \|\vec{a}_2\|^2 + C^2(\Delta\tau_2)^2$$
$$2(\vec{a}_1 - \vec{a}_3)^T \vec{x} - 2C^2 \Delta\tau_3 \tau_1 = \|\vec{a}_1\|^2 - \|\vec{a}_3\|^2 + C^2(\Delta\tau_3)^2$$
$$2(\vec{a}_1 - \vec{a}_4)^T \vec{x} - 2C^2 \Delta\tau_4 \tau_1 = \|\vec{a}_1\|^2 - \|\vec{a}_4\|^2 + C^2(\Delta\tau_4)^2$$



$$\left[\quad\right]\left[\quad\right] = \left[\quad\right]$$

Trilateration

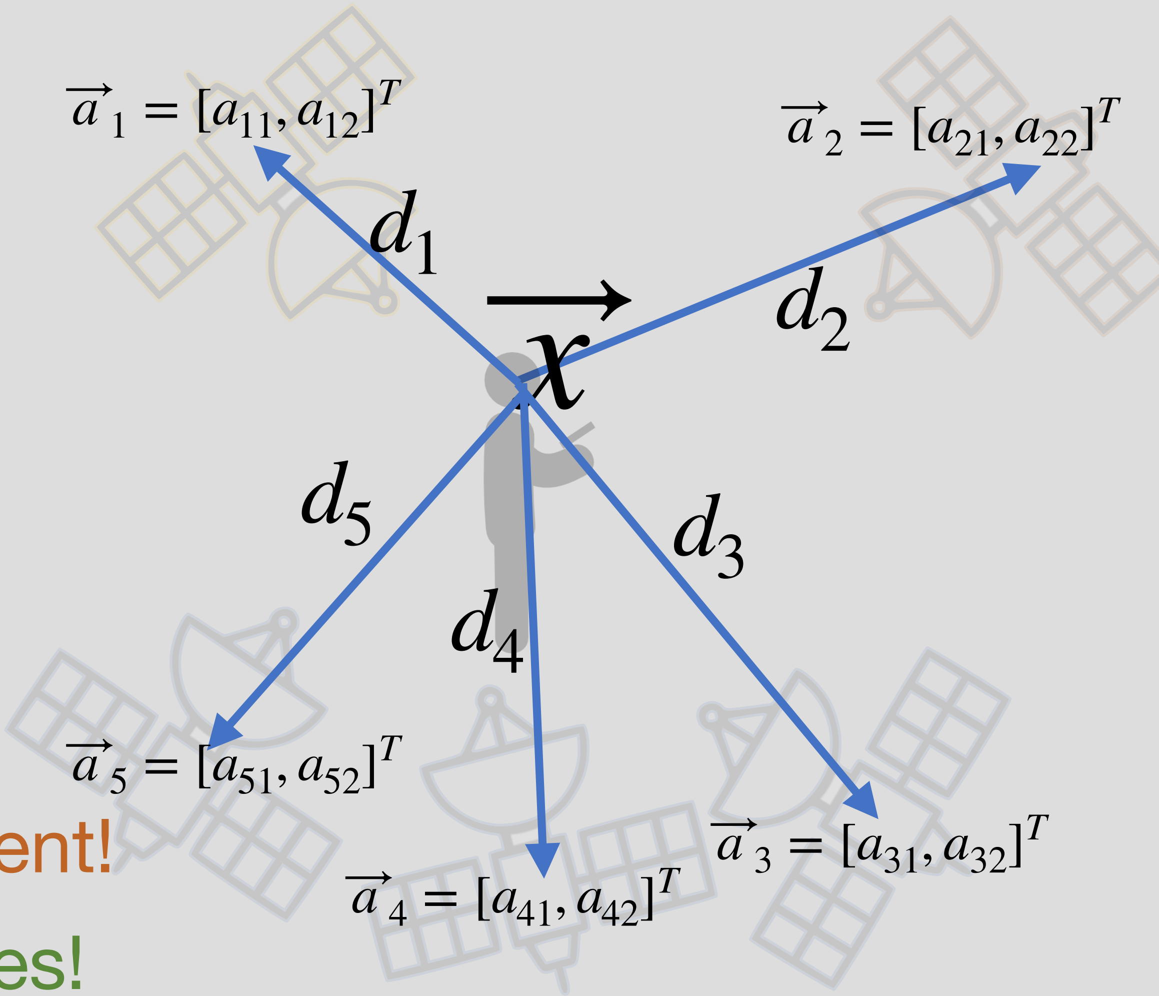
$$2(\vec{a}_1 - \vec{a}_2)^T \vec{x} - 2C^2 \Delta\tau_2 \tau_1 = \|\vec{a}_1\|^2 - \|\vec{a}_2\|^2 + C^2(\Delta\tau_2)^2$$
$$2(\vec{a}_1 - \vec{a}_3)^T \vec{x} - 2C^2 \Delta\tau_3 \tau_1 = \|\vec{a}_1\|^2 - \|\vec{a}_3\|^2 + C^2(\Delta\tau_3)^2$$
$$2(\vec{a}_1 - \vec{a}_4)^T \vec{x} - 2C^2 \Delta\tau_4 \tau_1 = \|\vec{a}_1\|^2 - \|\vec{a}_4\|^2 + C^2(\Delta\tau_4)^2$$



$$2\begin{bmatrix} a_{11}-a_{21} & a_{12}-a_{22} & -C^2\Delta\tau_2 \\ a_{11}-a_{31} & a_{12}-a_{32} & -C^2\Delta\tau_3 \\ a_{11}-a_{41} & a_{12}-a_{42} & -C^2\Delta\tau_4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \tau_1 \end{bmatrix} = \begin{bmatrix} \|a_1\|^2 - \|a_2\|^2 + C^2(\Delta\tau_2)^2 \\ \|a_1\|^2 - \|a_3\|^2 + C^2(\Delta\tau_3)^2 \\ \|a_1\|^2 - \|a_4\|^2 + C^2(\Delta\tau_4)^2 \end{bmatrix}$$

Multi-Lateration

$$2(\vec{a}_1 - \vec{a}_2)^T\vec{x} - 2C^2\Delta\tau_2\tau_1 = \|\vec{a}_1\|^2 - \|\vec{a}_2\|^2 + C^2(\Delta\tau_2)^2$$
$$2(\vec{a}_1 - \vec{a}_3)^T\vec{x} - 2C^2\Delta\tau_3\tau_1 = \|\vec{a}_1\|^2 - \|\vec{a}_3\|^2 + C^2(\Delta\tau_3)^2$$
$$2(\vec{a}_1 - \vec{a}_4)^T\vec{x} - 2C^2\Delta\tau_4\tau_1 = \|\vec{a}_1\|^2 - \|\vec{a}_4\|^2 + C^2(\Delta\tau_4)^2$$
$$2(\vec{a}_1 - \vec{a}_5)^T\vec{x} - 2C^2\Delta\tau_5\tau_1 = \|\vec{a}_1\|^2 - \|\vec{a}_5\|^2 + C^2(\Delta\tau_5)^2$$



More equations than unknowns

Q: With noise, equations will be inconsistent!

A: Find closest solution with Least-Squares!