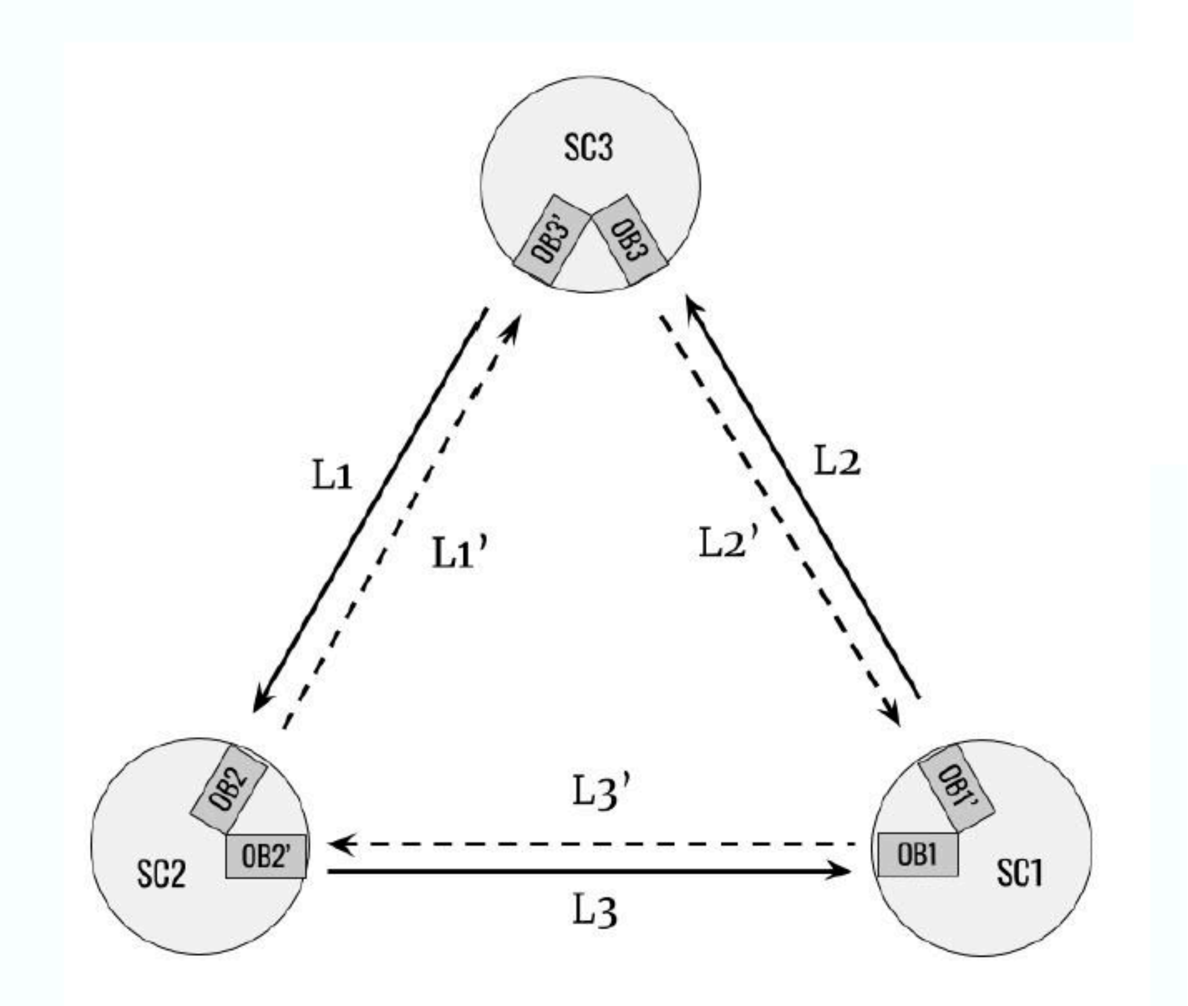
MACHINE LEARNING FOR LISA DATA PREPROCESSING

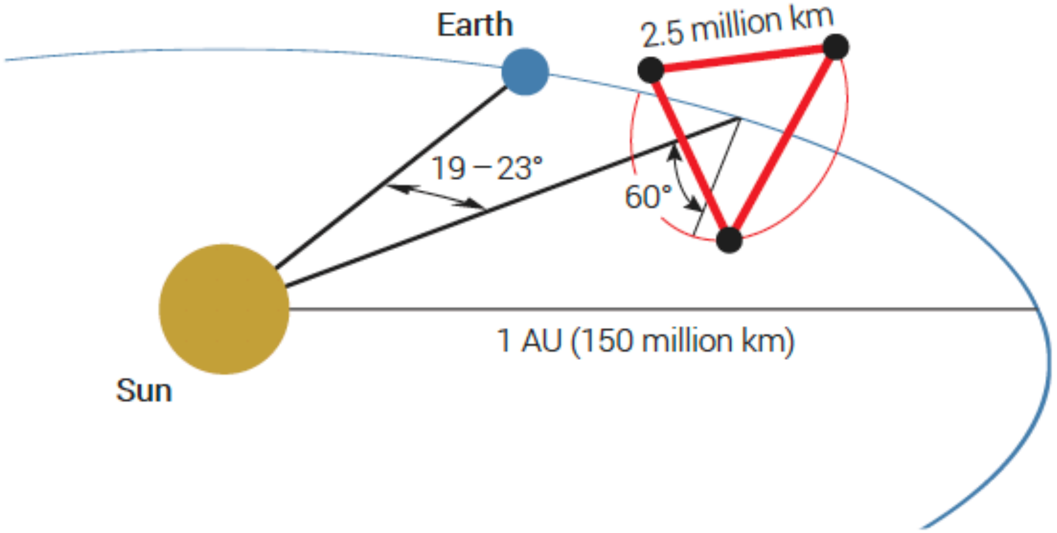
Natalia Korsakova
Robyn Munoz

Observatoire
de la Côte d'Azur

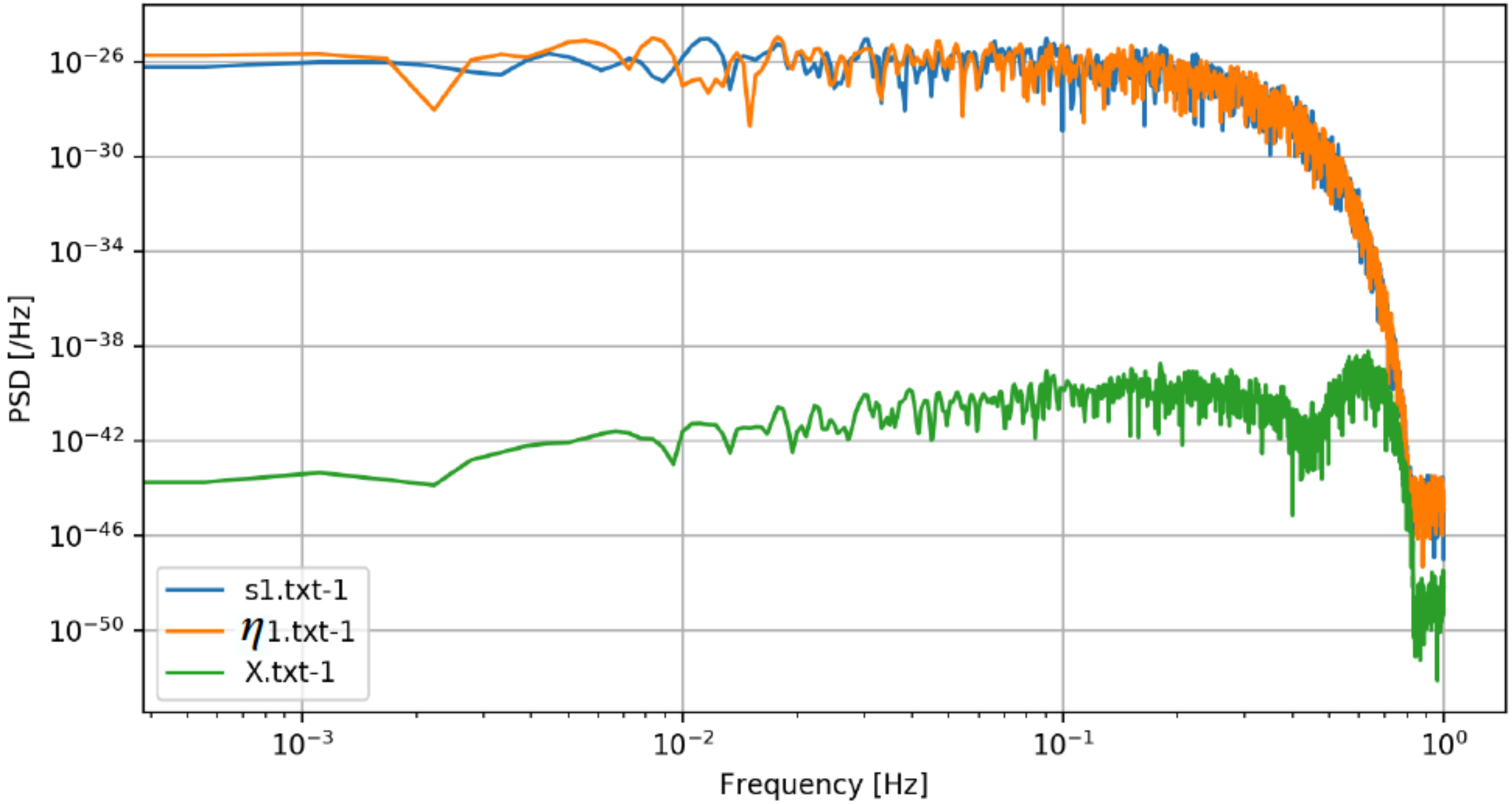
©S.Neveu

Laser Interferometer Space Antenna



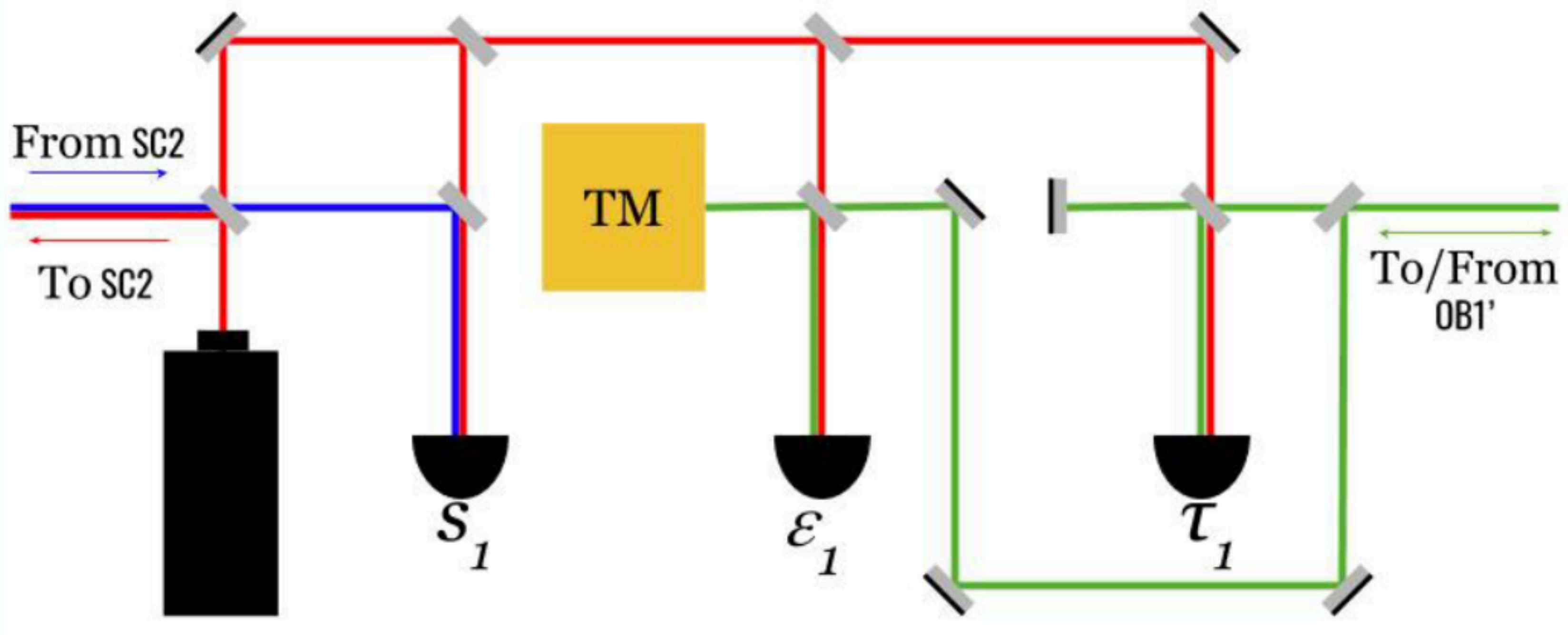


Laser Noise



PSD [/Hz]
10^{-26}
10^{-30}
10^{-34}
10^{-38}
10^{-42}
10^{-46}
10^{-50}
10^{-3}
10^{-2}
10^{-1}
10^{0}
Frequency [Hz]
s1.txt-1
η1.txt-1
X.txt-1

Time Delay Interferometry



LISA Optical Bench 1

<u>Raw Measurements</u>

$$\left\{\begin{matrix} s_1, & s_{1'}, & s_2, & s_{2'}, & s_3, & s_{3'}, \\ \varepsilon_1, & \varepsilon_{1'}, & \varepsilon_2, & \varepsilon_{2'}, & \varepsilon_3, & \varepsilon_{3'}, \\ \tau_1, & \tau_{1'}, & \tau_2, & \tau_{2'}, & \tau_3, & \tau_{3'} \end{matrix}\right\}$$

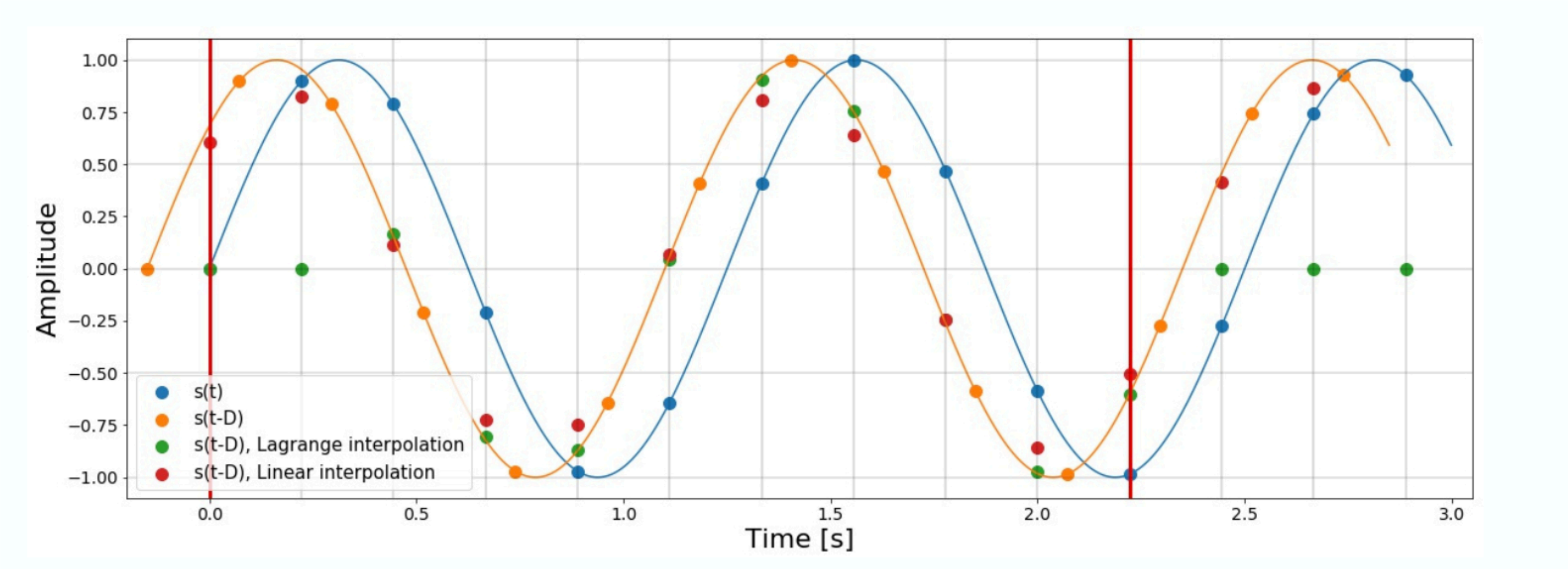
<u>Michelson Combinations</u>

$$\{X, Y, Z\}$$

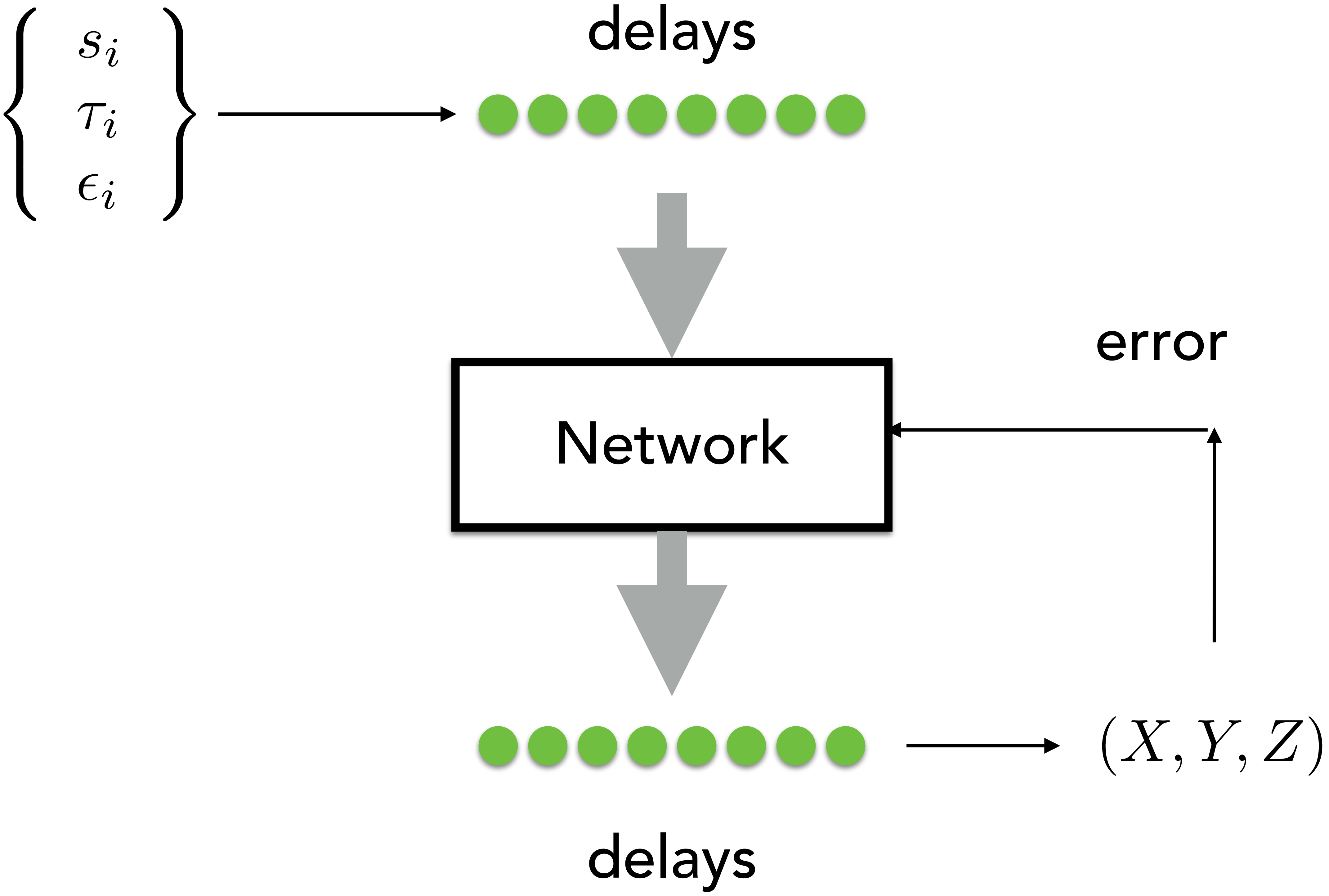
Lagrange Interpolation

Apply time delays

$$L_i(t) = L_i^0 + L_i^1 t + \ldots$$

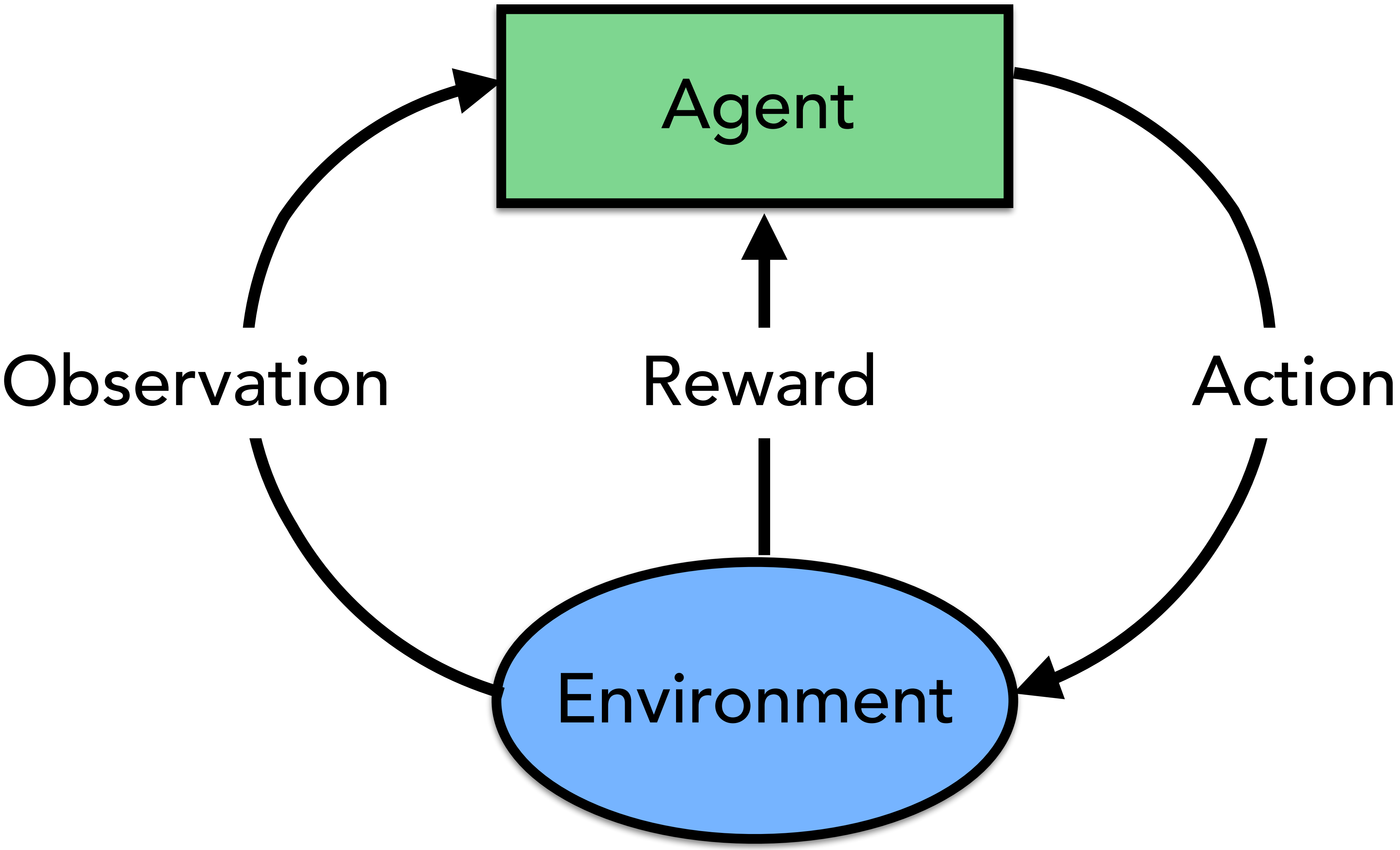


TDIR with NN



$\left\{ \begin{matrix} s_i \\ \tau_i \\ \epsilon_i \end{matrix} \right\}$
delays
Network
error
(X, Y, Z)
delays

Reinforcement learning



Policy gradient

$(a_0, r_0, a_1, r_1, \dots)$ *Action - reward pairs, trajectory*

$$R(\tau) = \sum_{t=0}^{T} \gamma^t r_t$$ *Reward*

$$J(\pi) = \mathbf{E}_{\tau \sim \pi}[R(\tau)]$$ *Performance*

$$\nabla_{\Theta} J(\pi) = \mathbf{E}[\nabla_{\Theta} \log \pi(a_t|s_t) R(\tau)]$$ *Gradient of performance can be approximated by sampling*