

Imputing Missing Events in Continuous-Time Event Streams



Hongyuan Mei¹



Guanghui Qin²



Jason Eisner¹

Event Streams

Medical

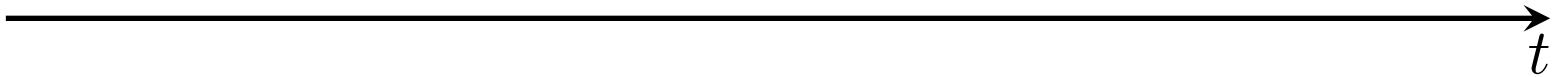


Event Streams

Shopping



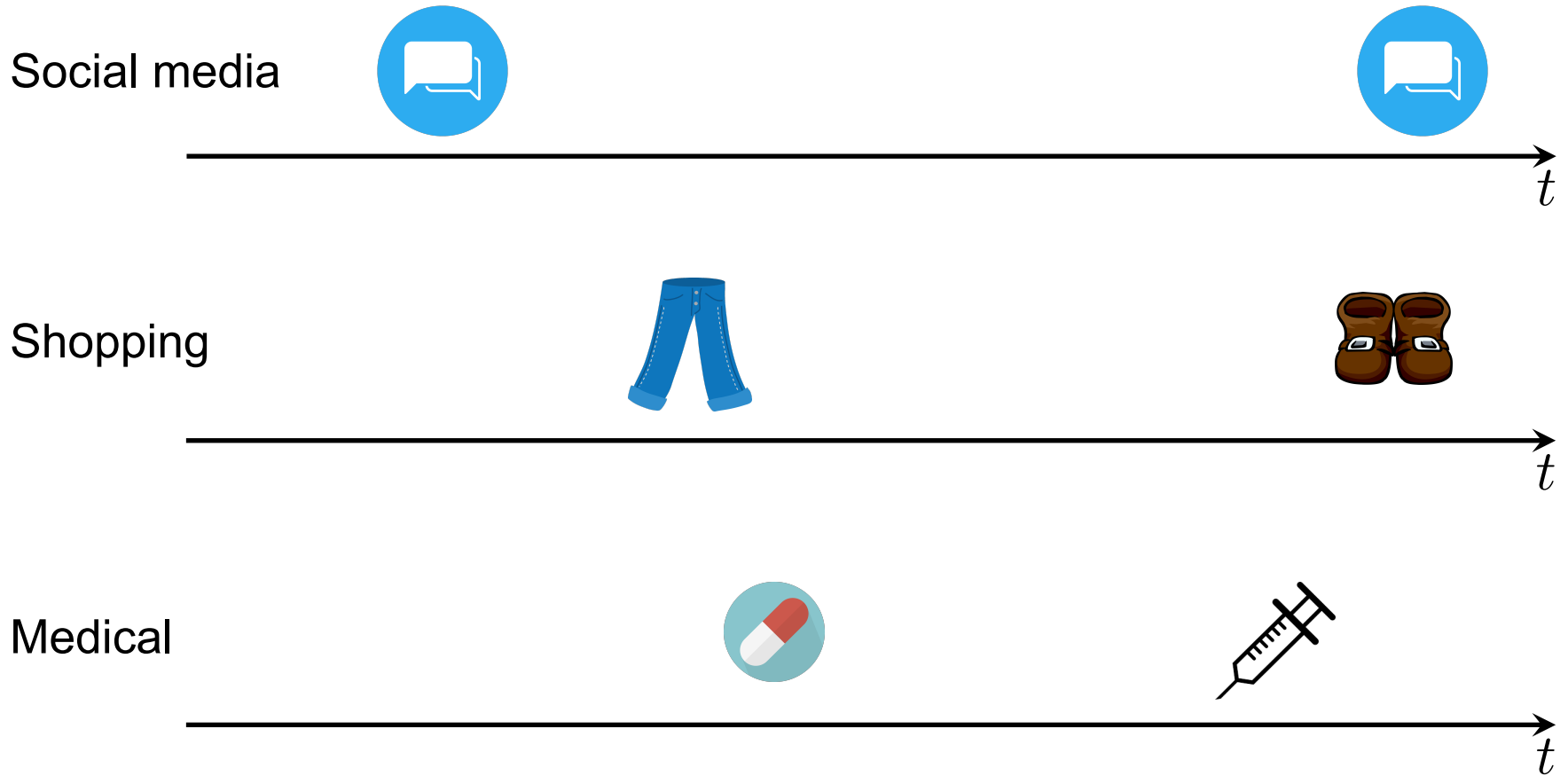
Medical



Event Streams



Event Streams



Event Streams



Let's Play a Battle Game



Let's Play a Battle Game



- Incomplete data

Let's Play a Battle Game



- Incomplete data
- Opponent tanks come

0

Let's Play a Battle Game



- Incomplete data
- Opponent tanks come



0

t_1

t_2

Let's Play a Battle Game



- Incomplete data
- Opponent tanks come
- Maybe a new factory was built to produce them?



0

t_1

t_2

Let's Play a Battle Game



- Incomplete data
- Opponent tanks come
- Maybe a new factory was built to produce them?
- When was it built?



0

t_1

t_2

Let's Play a Battle Game



- Incomplete data
- Opponent tanks come
- Maybe a new factory was built to produce them?
- When was it built?



0

?

t_1

t_2

Let's Play a Battle Game



- Incomplete data
- Opponent tanks come
- Maybe a new factory was built to produce them?
- When was it built?



0

?

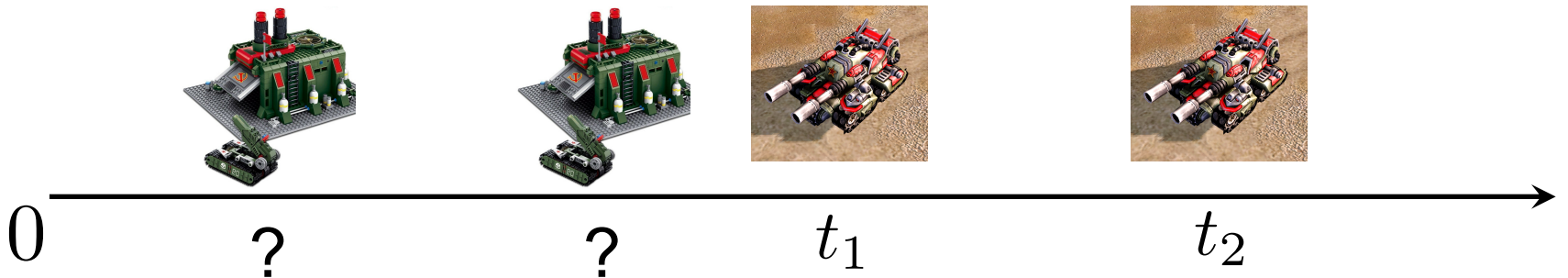
t_1

t_2

Let's Play a Battle Game



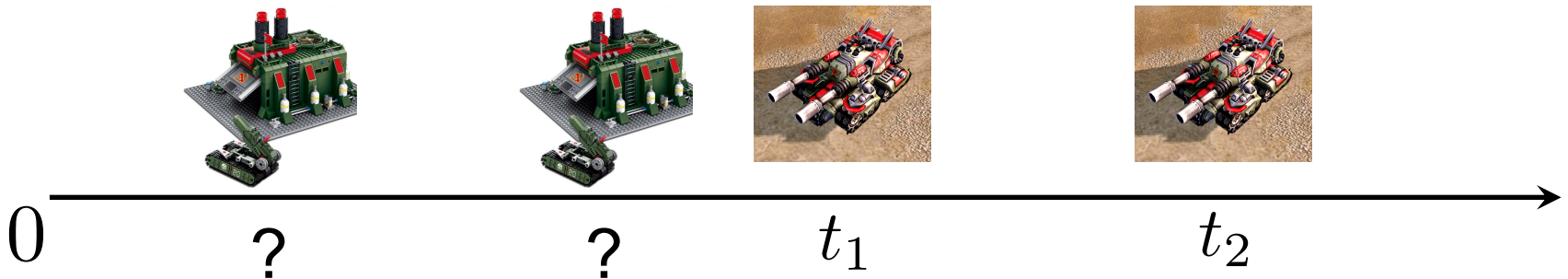
- Incomplete data
- Opponent tanks come
- Maybe a new factory was built to produce them?
- When was it built?



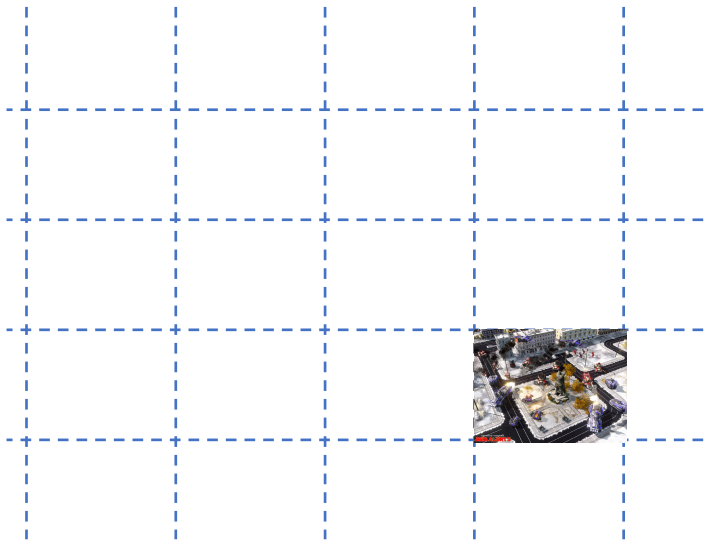
Let's Play a Battle Game



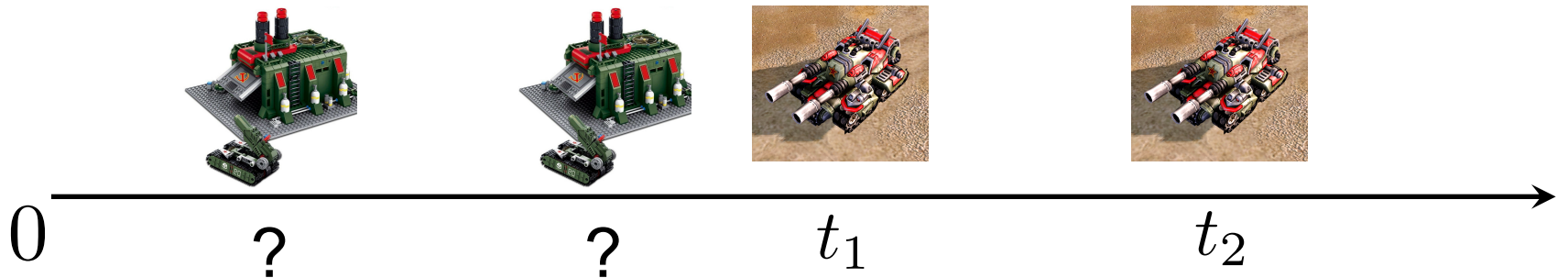
- Incomplete data
- Opponent tanks come
- Maybe a new factory was built to produce them?
- When was it built?
- Where was it built?



Let's Play a Battle Game



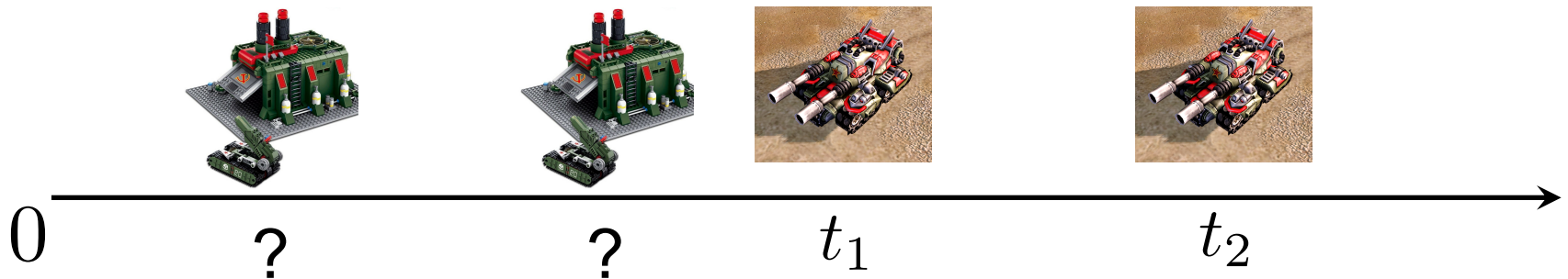
- Incomplete data
- Opponent tanks come
- Maybe a new factory was built to produce them?
- When was it built?
- Where was it built?



Let's Play a Battle Game



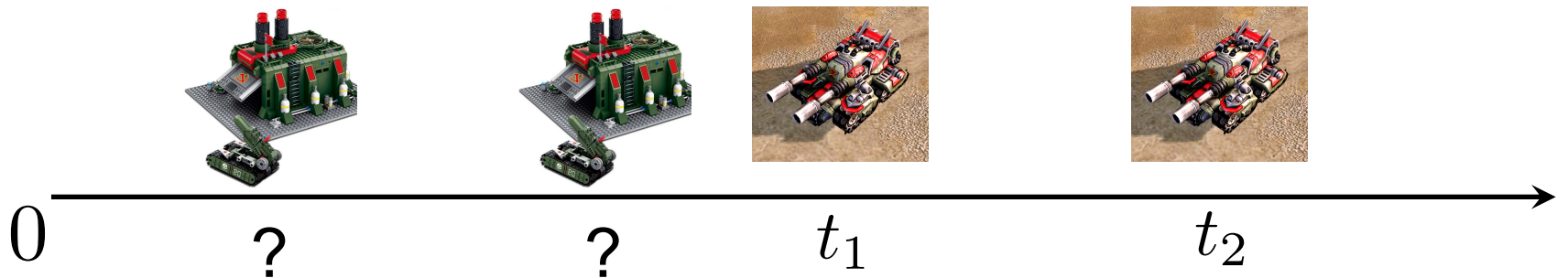
- Incomplete data
- Opponent tanks come
- Maybe a new factory was built to produce them?
- When was it built?
- Where was it built?



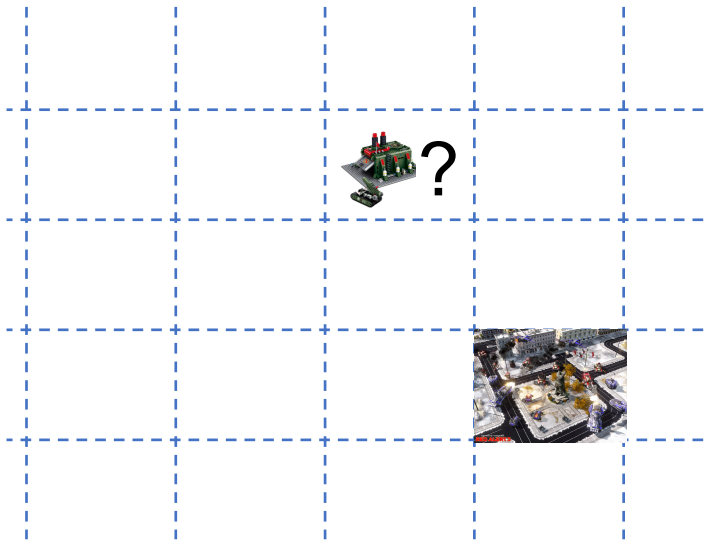
Let's Play a Battle Game



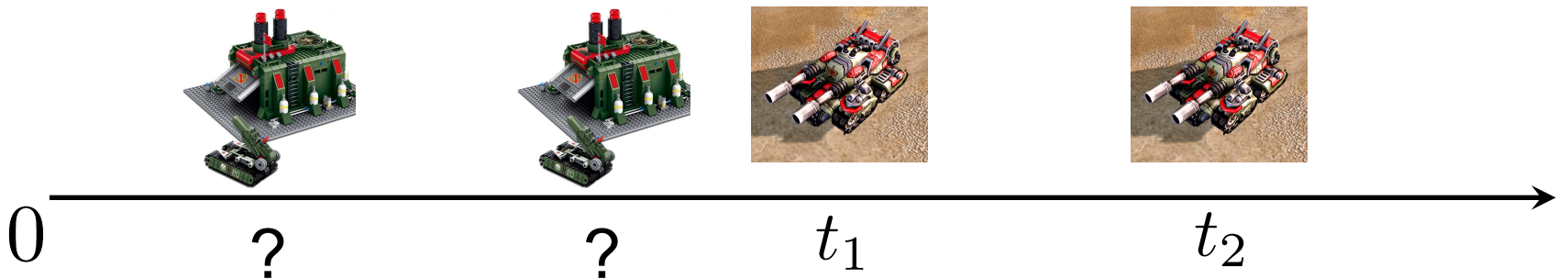
- Incomplete data
- Opponent tanks come
- Maybe a new factory was built to produce them?
- When was it built?
- Where was it built?



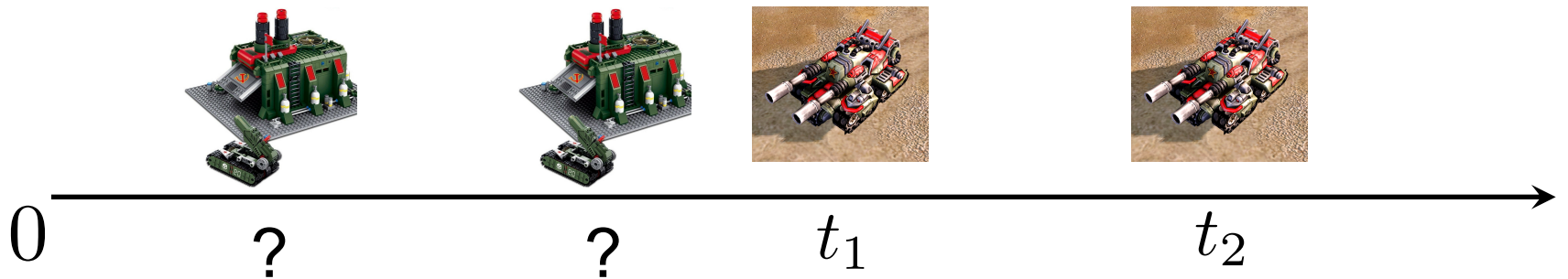
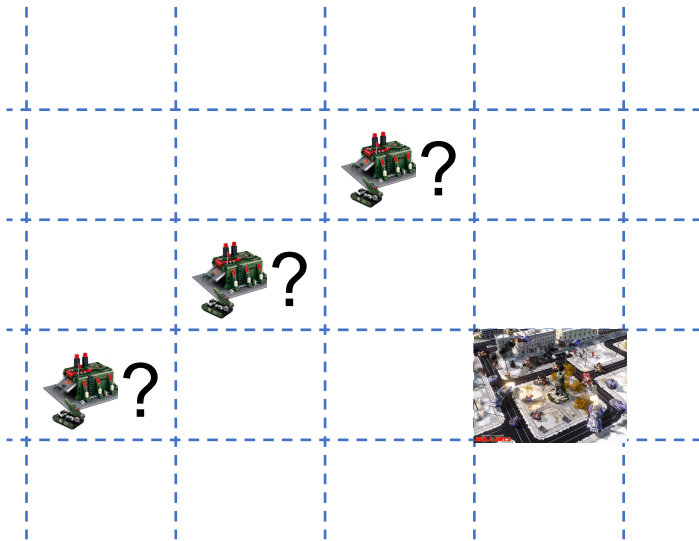
Let's Play a Battle Game



- Incomplete data
- Opponent tanks come
- Maybe a new factory was built to produce them?
- When was it built?
- Where was it built?

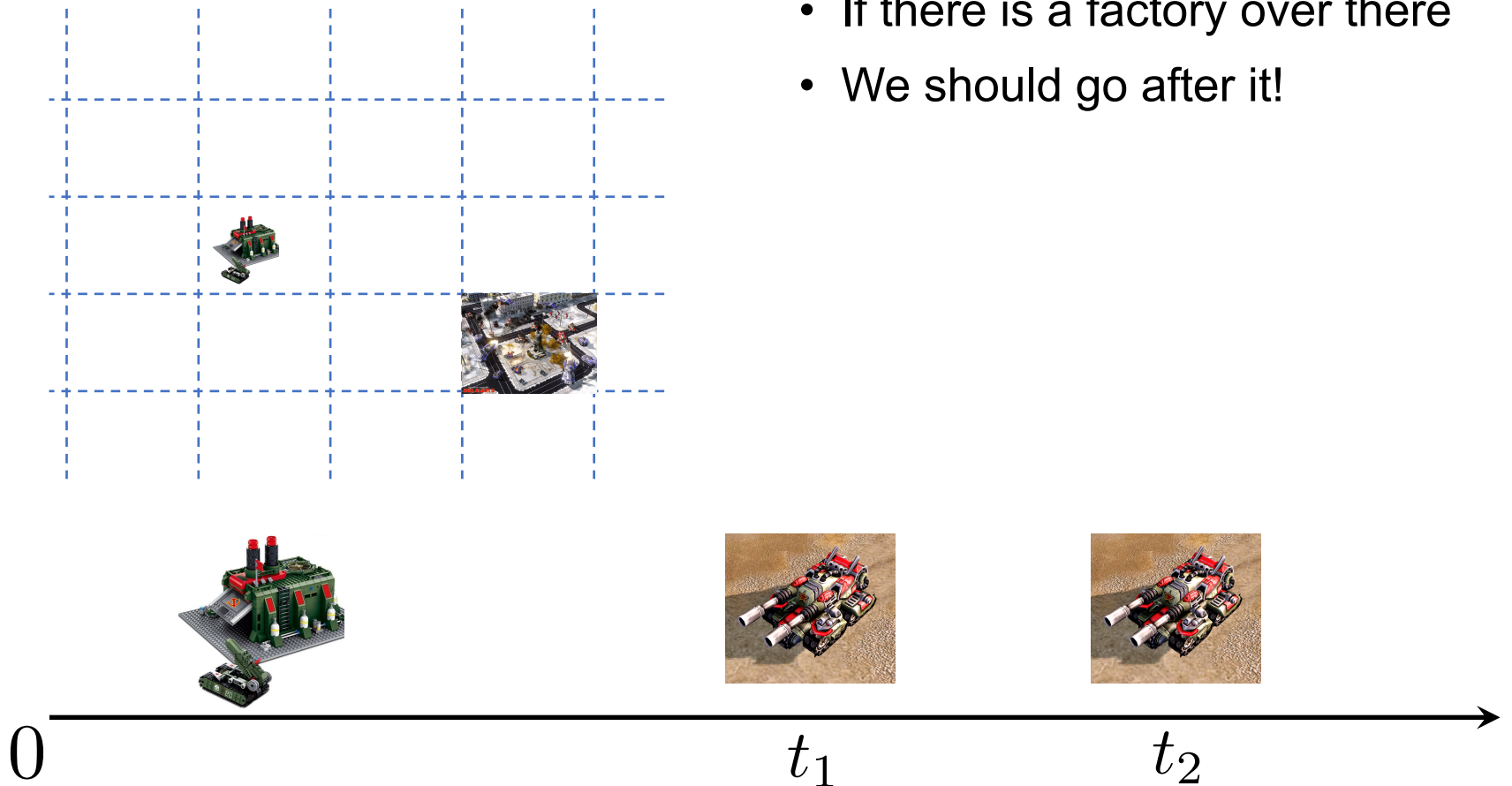


Why to Impute?

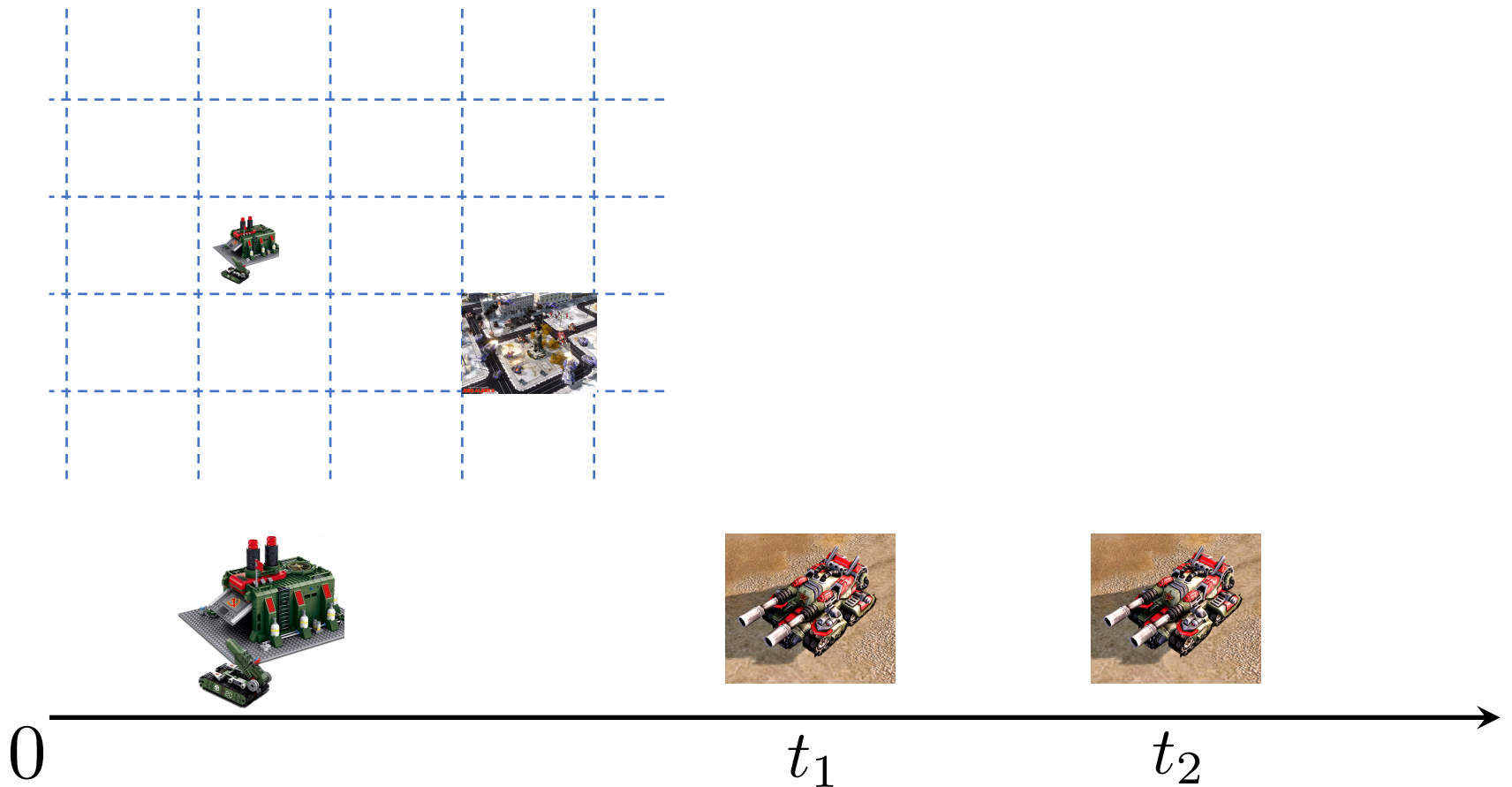


Why to Impute?

- If there is a factory over there
- We should go after it!

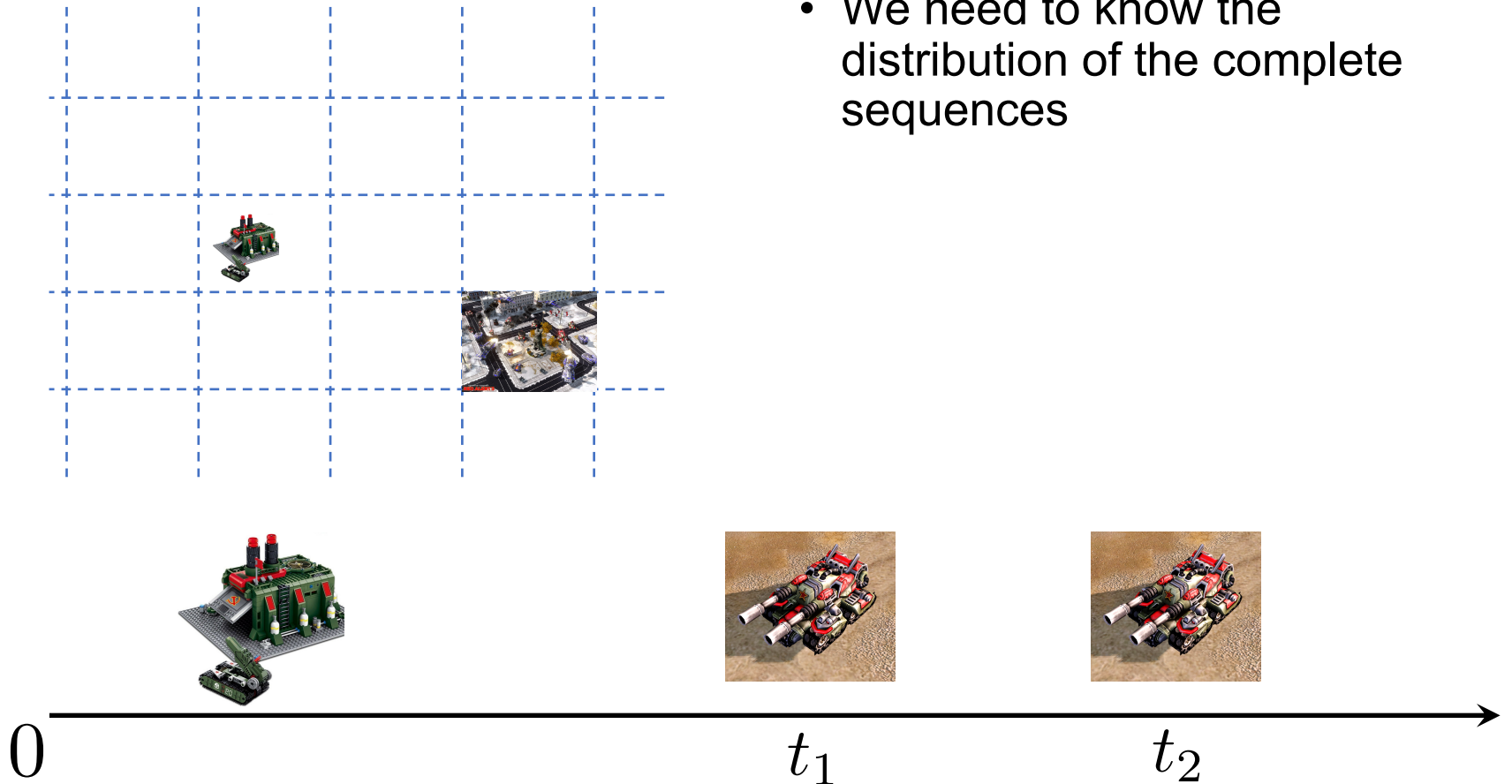


To Impute the Missing Events...



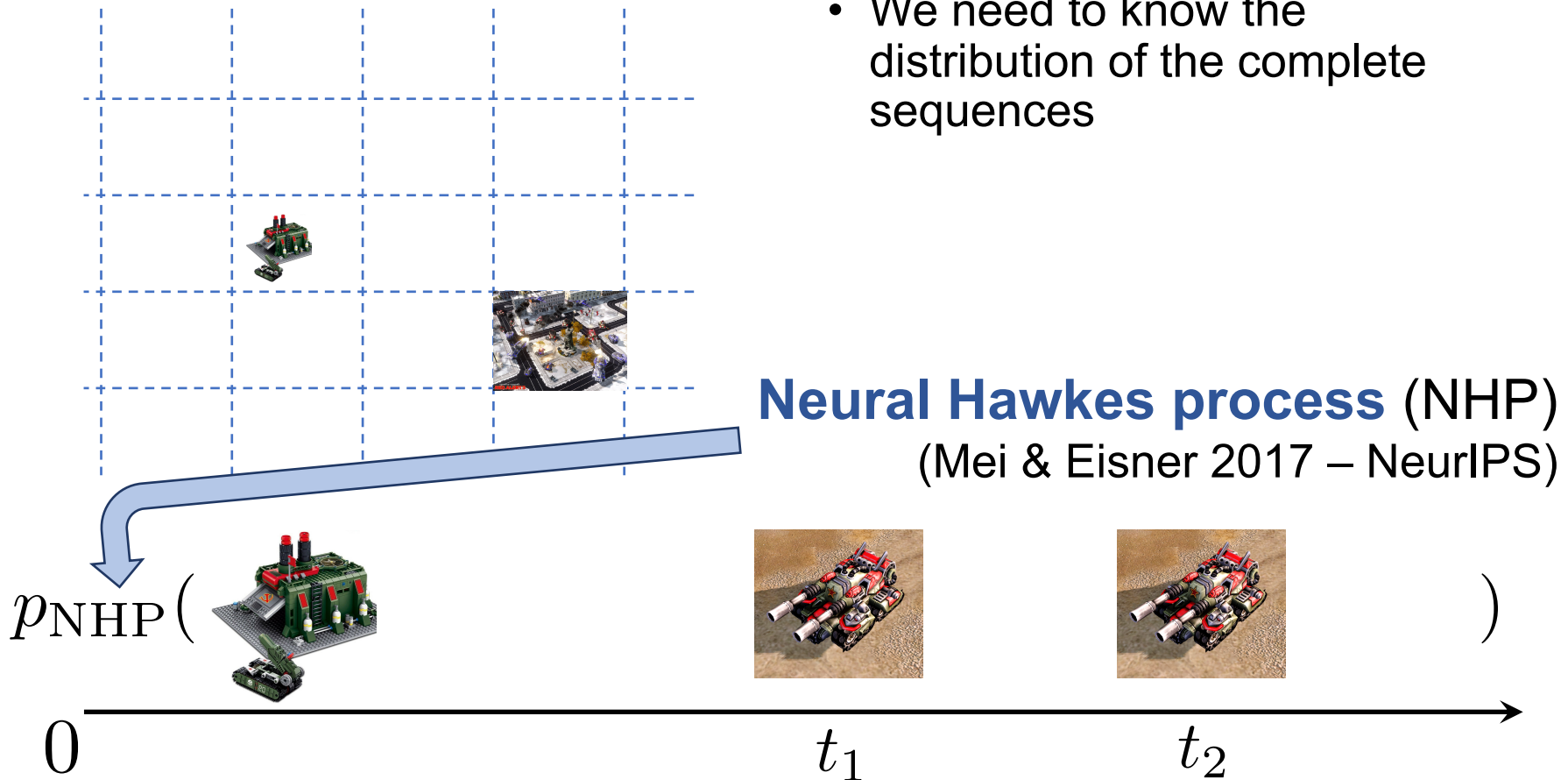
To Impute the Missing Events...

- We need to know the distribution of the complete sequences



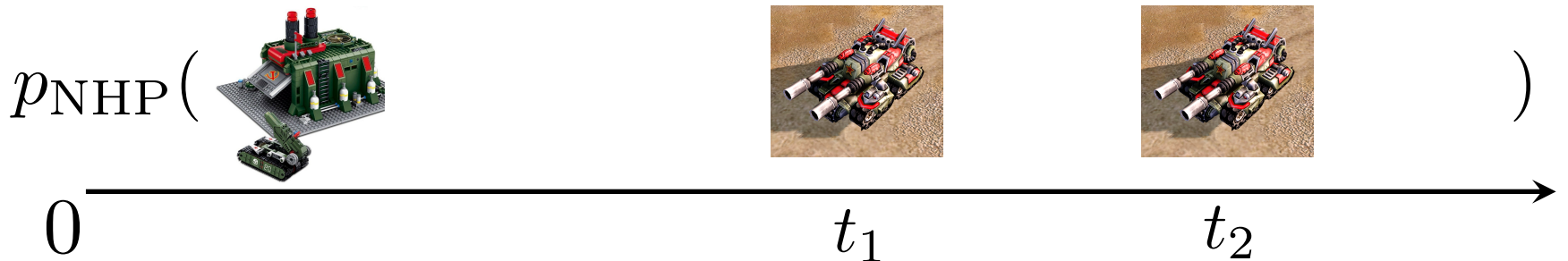
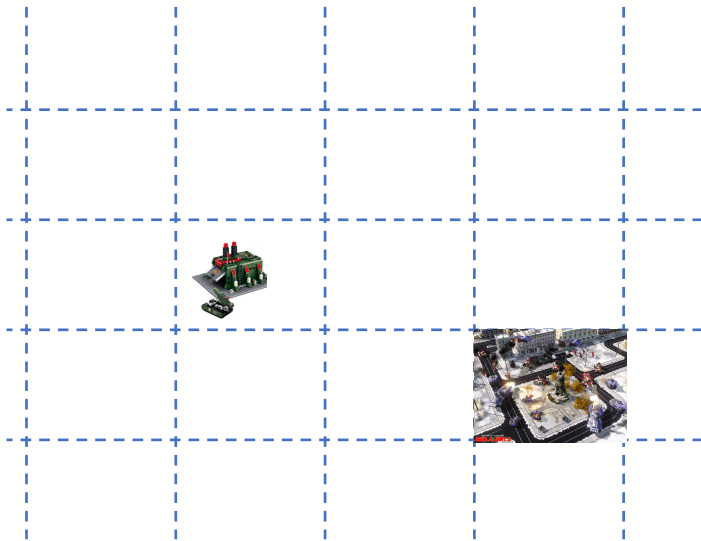
To Impute the Missing Events...

- We need to know the distribution of the complete sequences



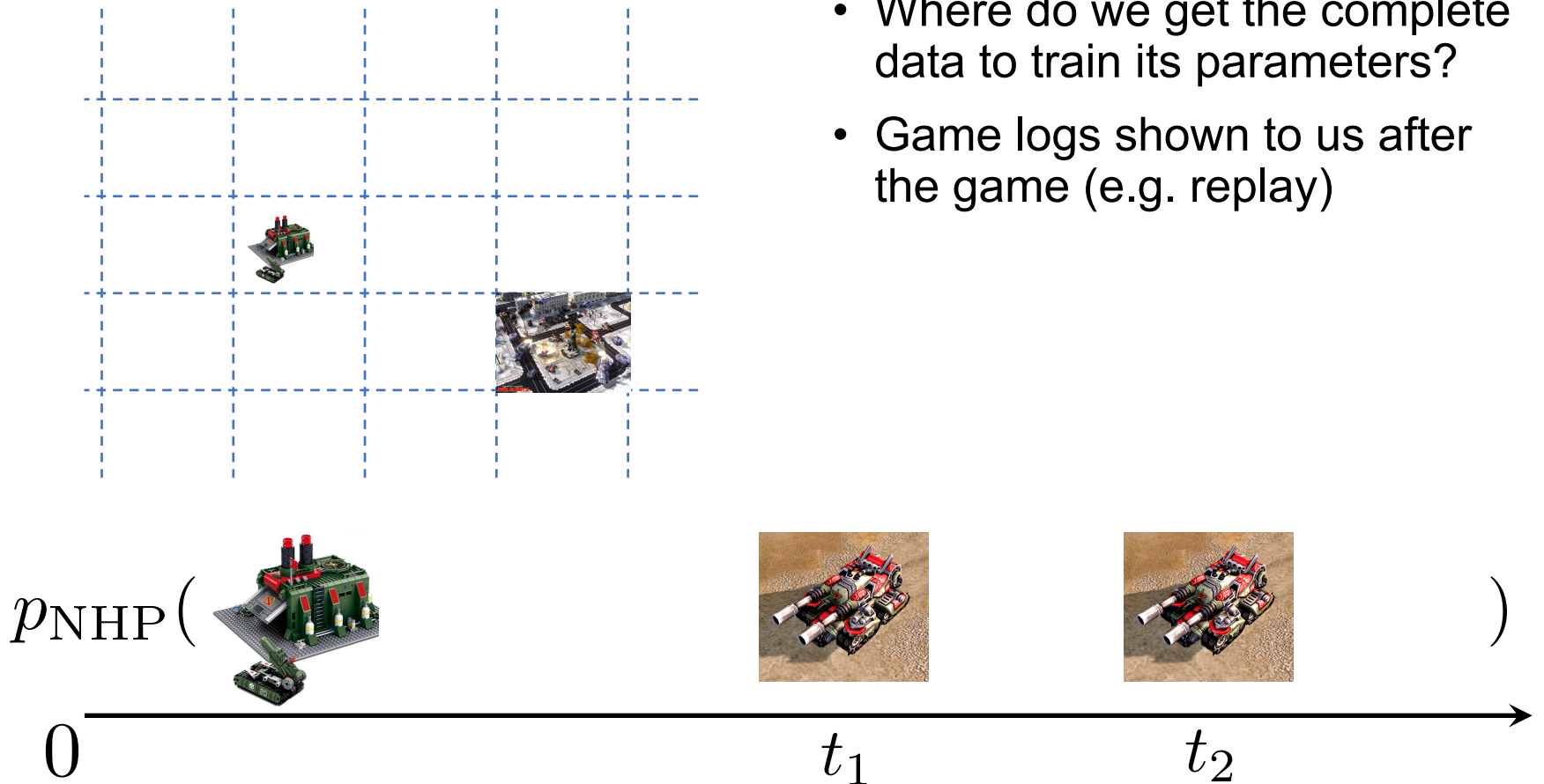
To Impute the Missing Events...

- Where do we get the complete data to train its parameters?



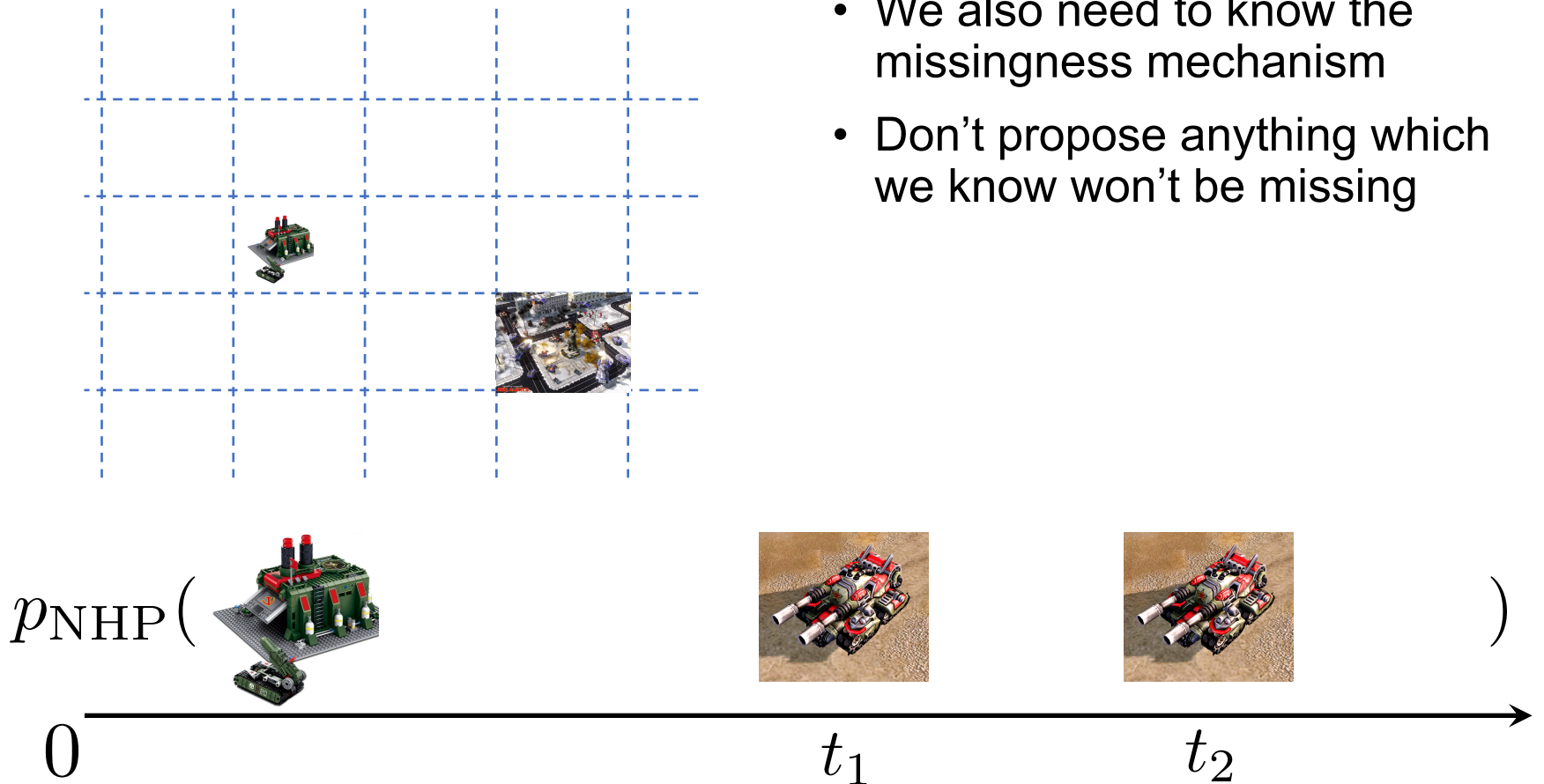
To Impute the Missing Events...

- Where do we get the complete data to train its parameters?
- Game logs shown to us after the game (e.g. replay)

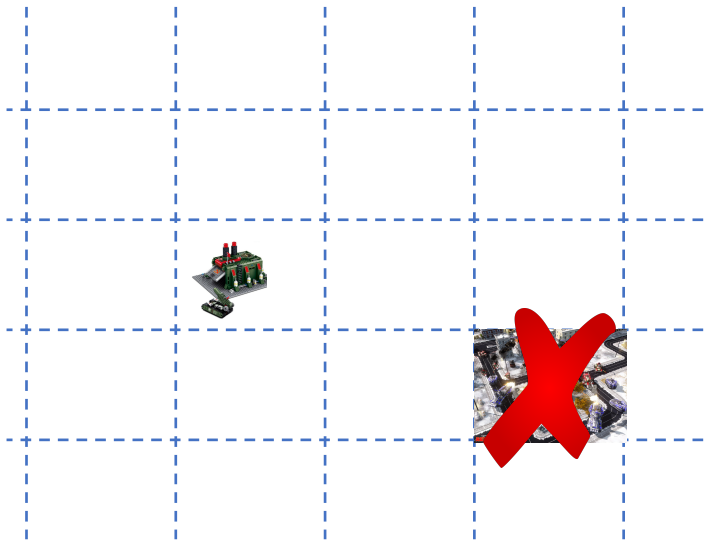


To Impute the Missing Events...

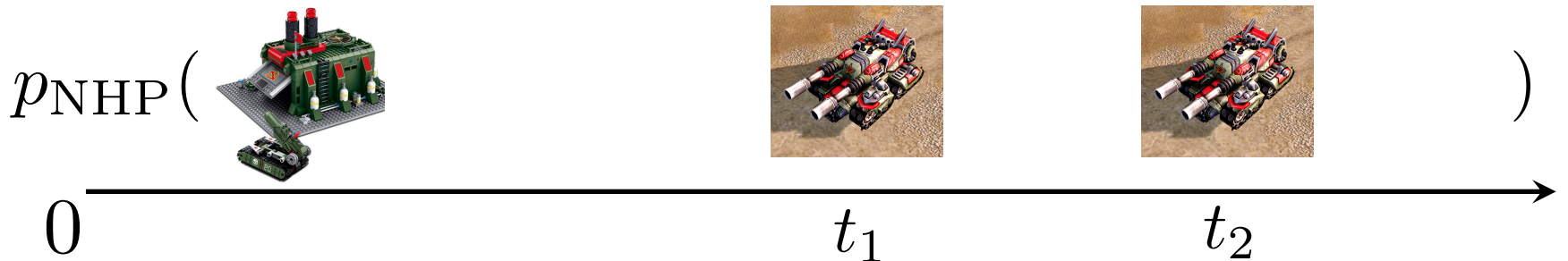
- We also need to know the missingness mechanism
- Don't propose anything which we know won't be missing



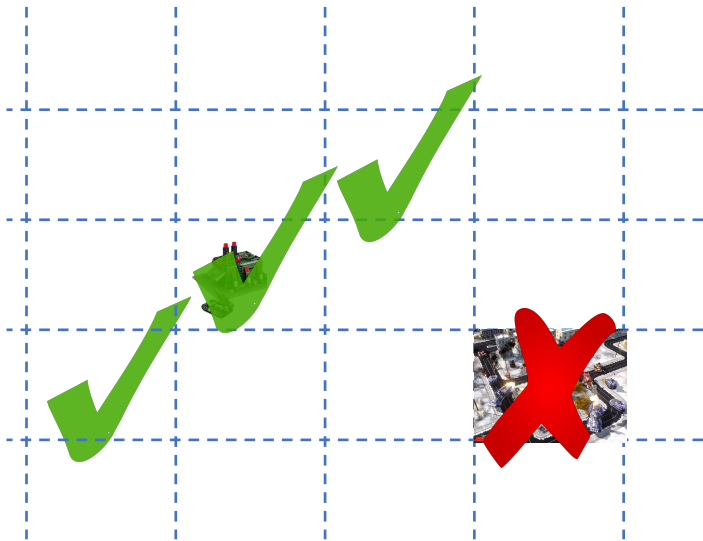
To Impute the Missing Events...



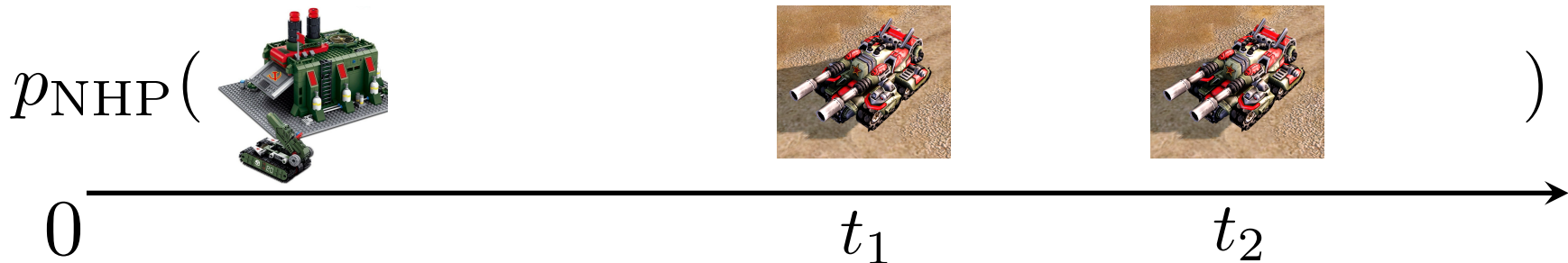
- We also need to know the missingness mechanism
- Don't propose anything which we know won't be missing
- In-view events won't be missing



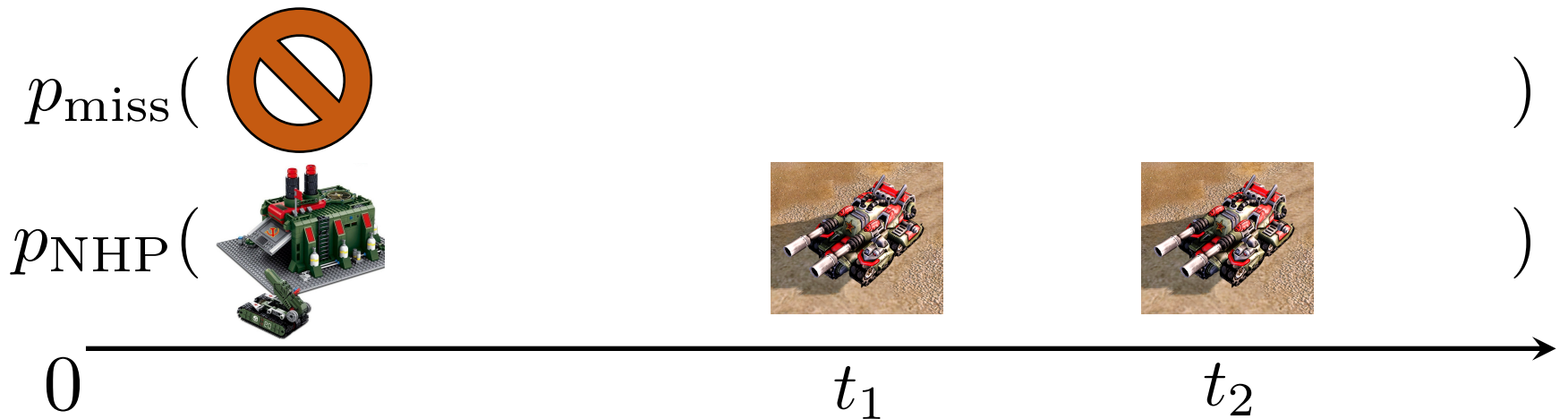
To Impute the Missing Events...



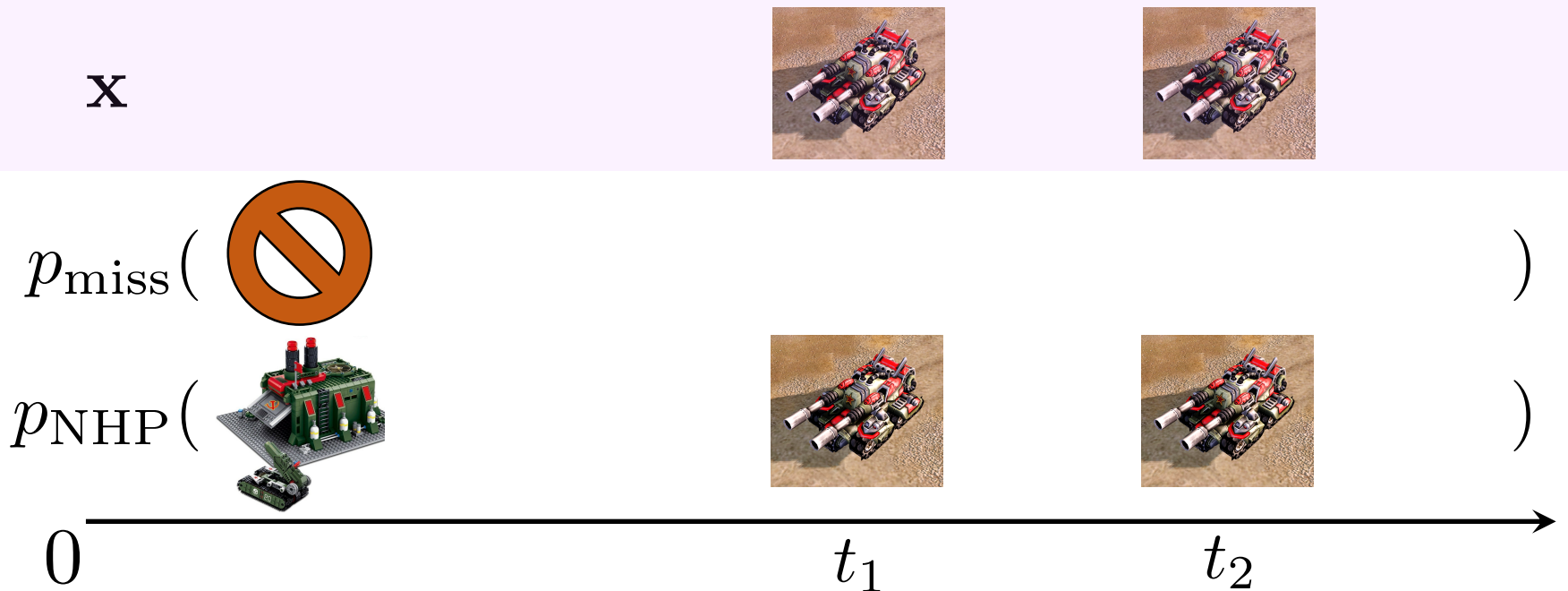
- We also need to know the missingness mechanism
- Don't propose anything which we know won't be missing
- In-view events won't be missing
- Out-of-view events must be missing



To Impute the Missing Events...



To Impute the Missing Events...



To Impute the Missing Events...

Z



X



p_{miss} (



)

p_{NHP} (



)



0

t_1

t_2

To Impute the Missing Events...

z

?

?

?

?

x



0

t_1

t_2

To Impute the Missing Events...

$$p(\mathbf{z} \mid \mathbf{x}) \propto p_{\text{NHP}}(\mathbf{x} \sqcup \mathbf{z}) p_{\text{miss}}(\mathbf{z} \mid \mathbf{x} \sqcup \mathbf{z})$$

\mathbf{z}

?

?

?

?

\mathbf{x}



0

t_1

t_2

Challenge

- Complicated p_{NHP}
- Exact inference is intractable
- How about Monte Carlo?

$$p(\mathbf{z} \mid \mathbf{x}) = ?$$

\mathbf{z}

?

?

?

?

\mathbf{x}



0

t_1

t_2

Sequential Monte Carlo

- Sample from $q(\mathbf{z} \mid \mathbf{x})$

\mathbf{x}
0



t_1



t_2

Sequential Monte Carlo

z_1



z_2



z_3



- Sample from $q(\mathbf{z} \mid \mathbf{x})$

\mathbf{x}



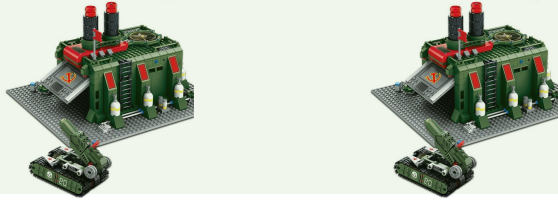
0

t_1

t_2

Sequential Monte Carlo

z_1



z_2



z_3



- Sample from $q(\mathbf{z} \mid \mathbf{x})$
- Weight them by w

$$w \propto p(\mathbf{z} \mid \mathbf{x}) / q(\mathbf{z} \mid \mathbf{x})$$

\mathbf{x}

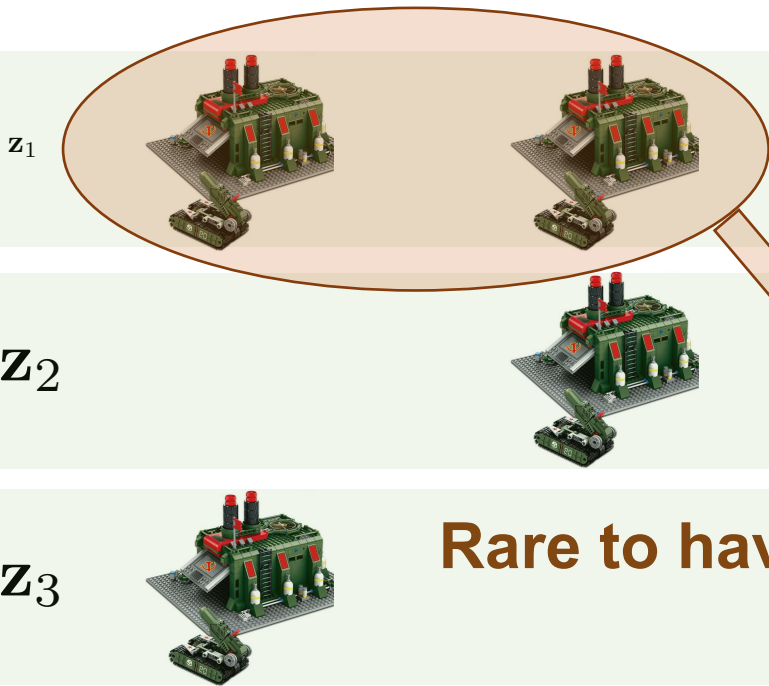


0

t_1

t_2

Sequential Monte Carlo



- Sample from $q(\mathbf{z} \mid \mathbf{x})$
- Weight them by w

$$w \propto p(\mathbf{z} \mid \mathbf{x}) / q(\mathbf{z} \mid \mathbf{x})$$

Rare to have two factories in a row!

\mathbf{x}

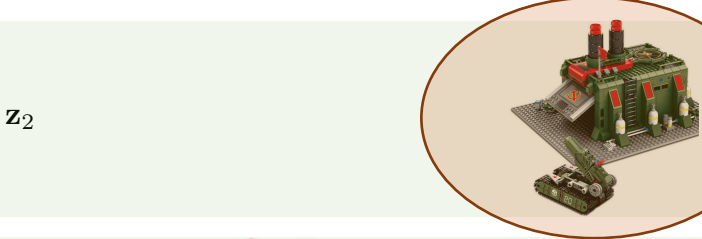
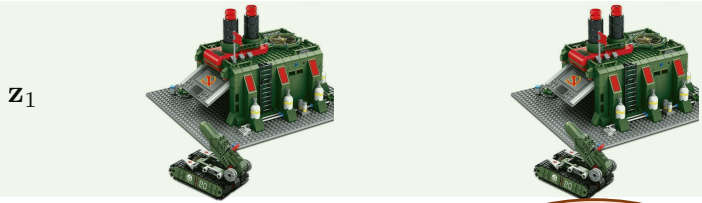


0

t_1

t_2

Sequential Monte Carlo



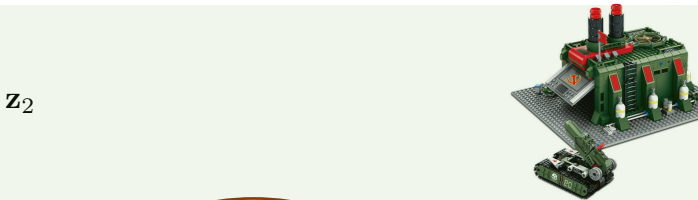
- Sample from $q(\mathbf{z} \mid \mathbf{x})$
 - Weight them by w
- $$w \propto p(\mathbf{z} \mid \mathbf{x}) / q(\mathbf{z} \mid \mathbf{x})$$

**Down-weighted! Too late!
No time to produce tanks!**

\mathbf{x}

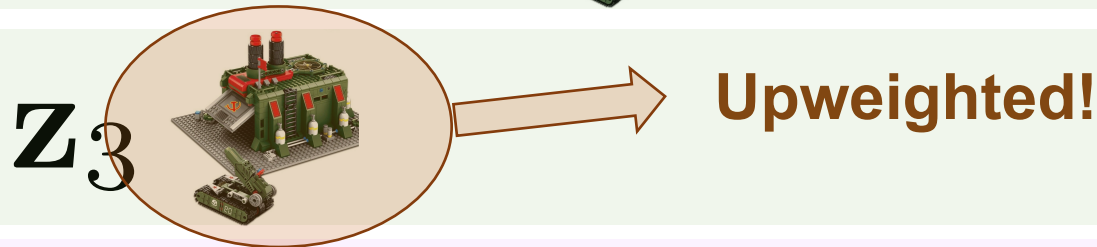


Sequential Monte Carlo



- Sample from $q(\mathbf{z} \mid \mathbf{x})$
- Weight them by w

$$w \propto p(\mathbf{z} \mid \mathbf{x}) / q(\mathbf{z} \mid \mathbf{x})$$



\mathbf{x}



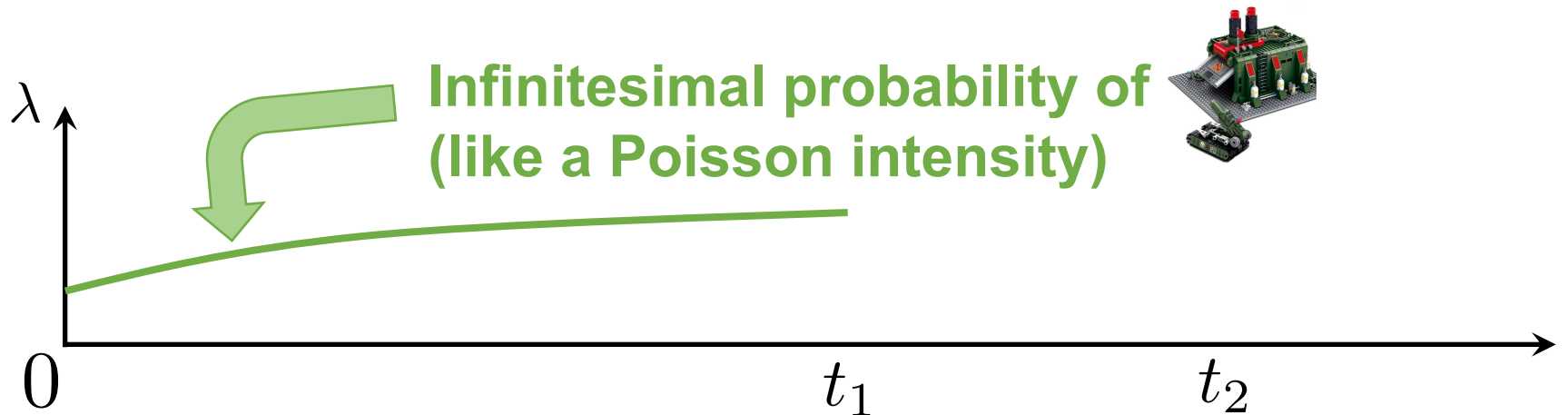
0

t_1

t_2

Sequential Monte Carlo

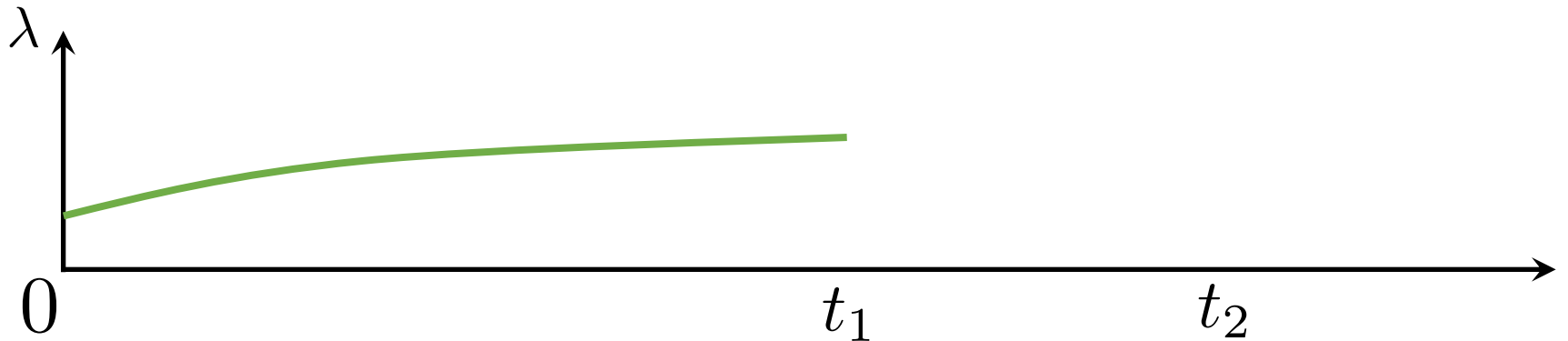
- Use the trained NHP



Sequential Monte Carlo

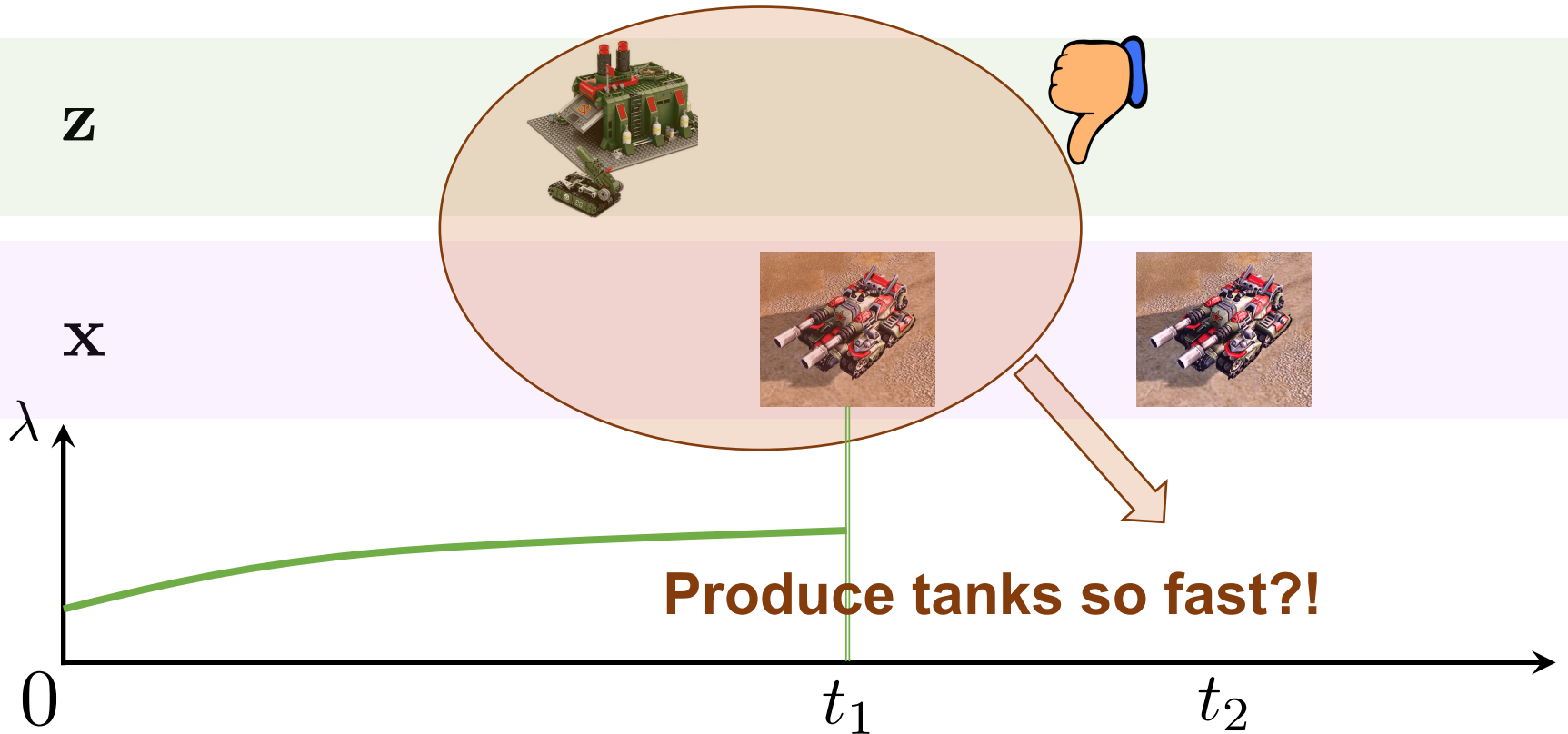
- Use the trained NHP

z



Sequential Monte Carlo

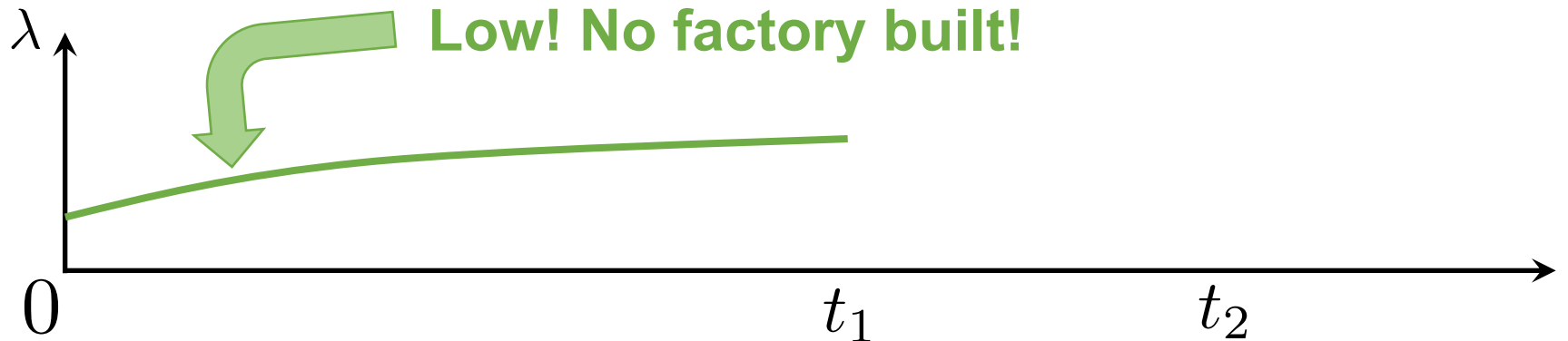
- Use the trained NHP



Sequential Monte Carlo

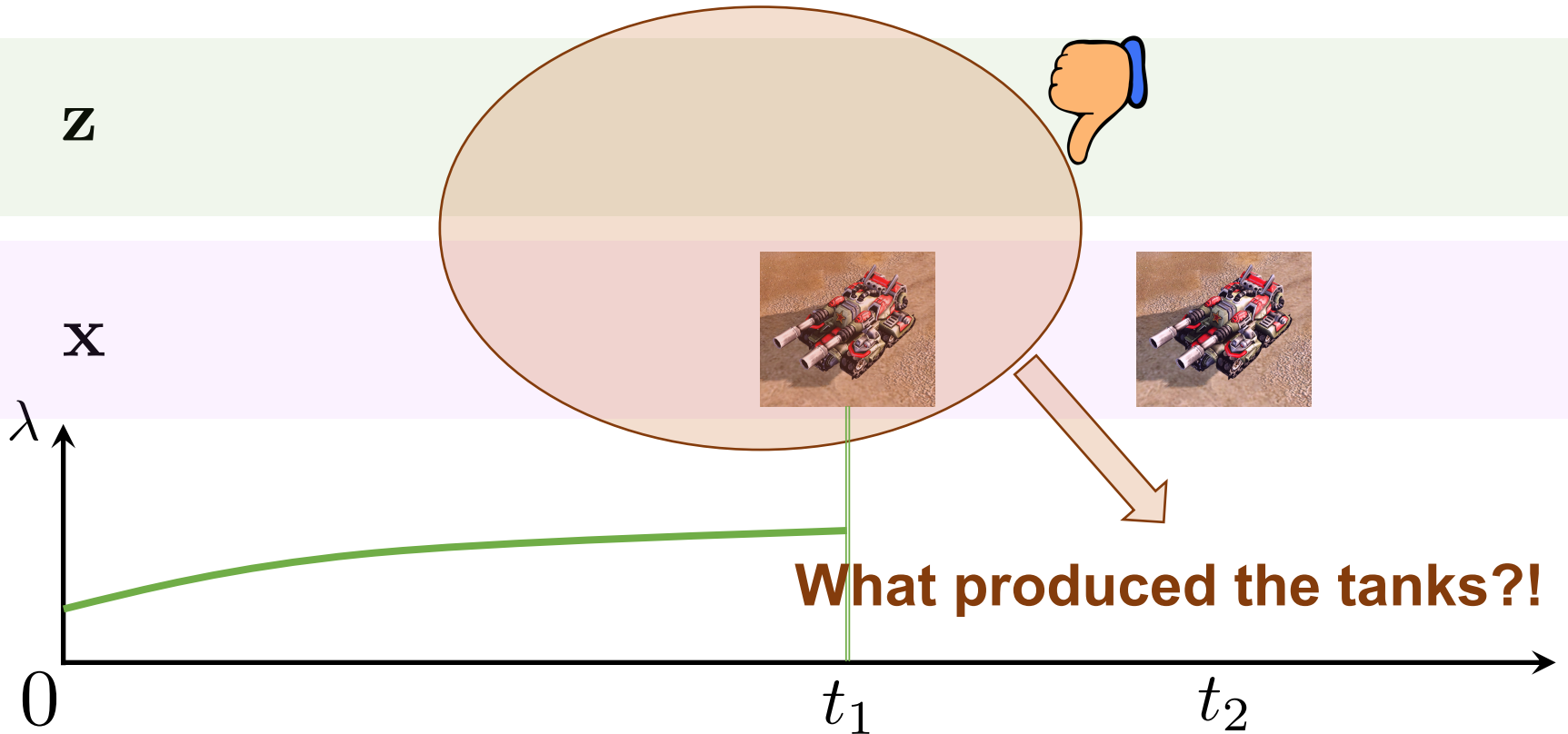
- Use the trained NHP

z



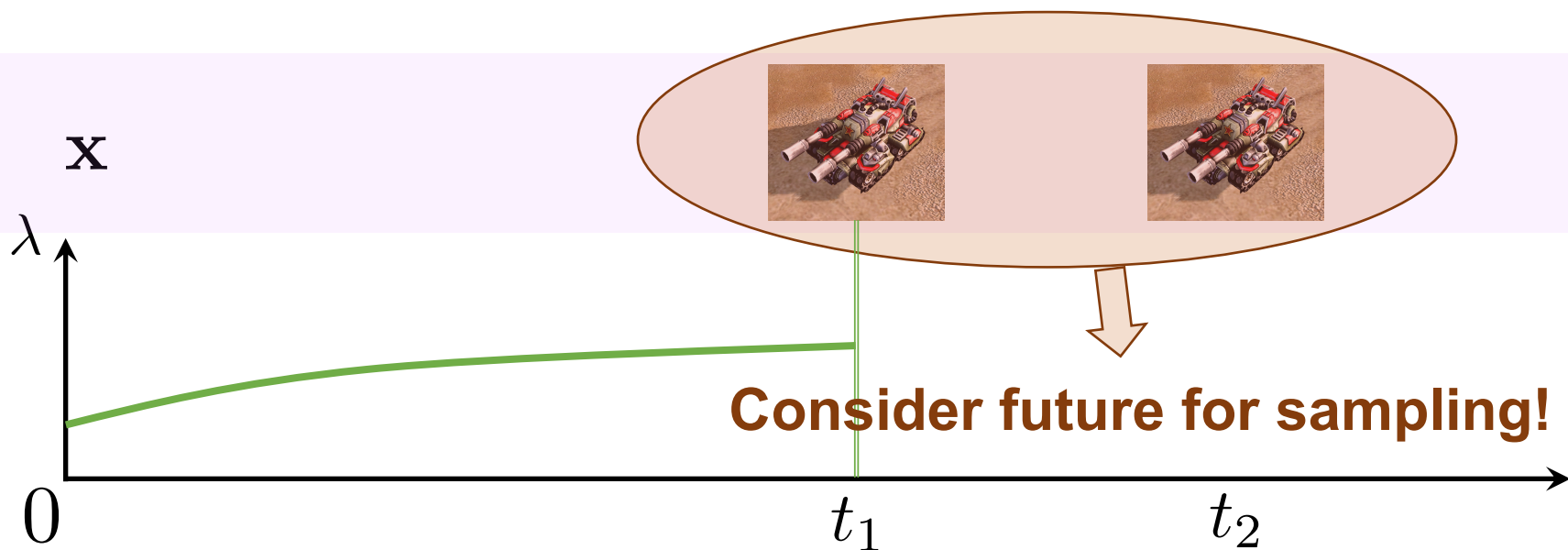
Sequential Monte Carlo

- Use the trained NHP



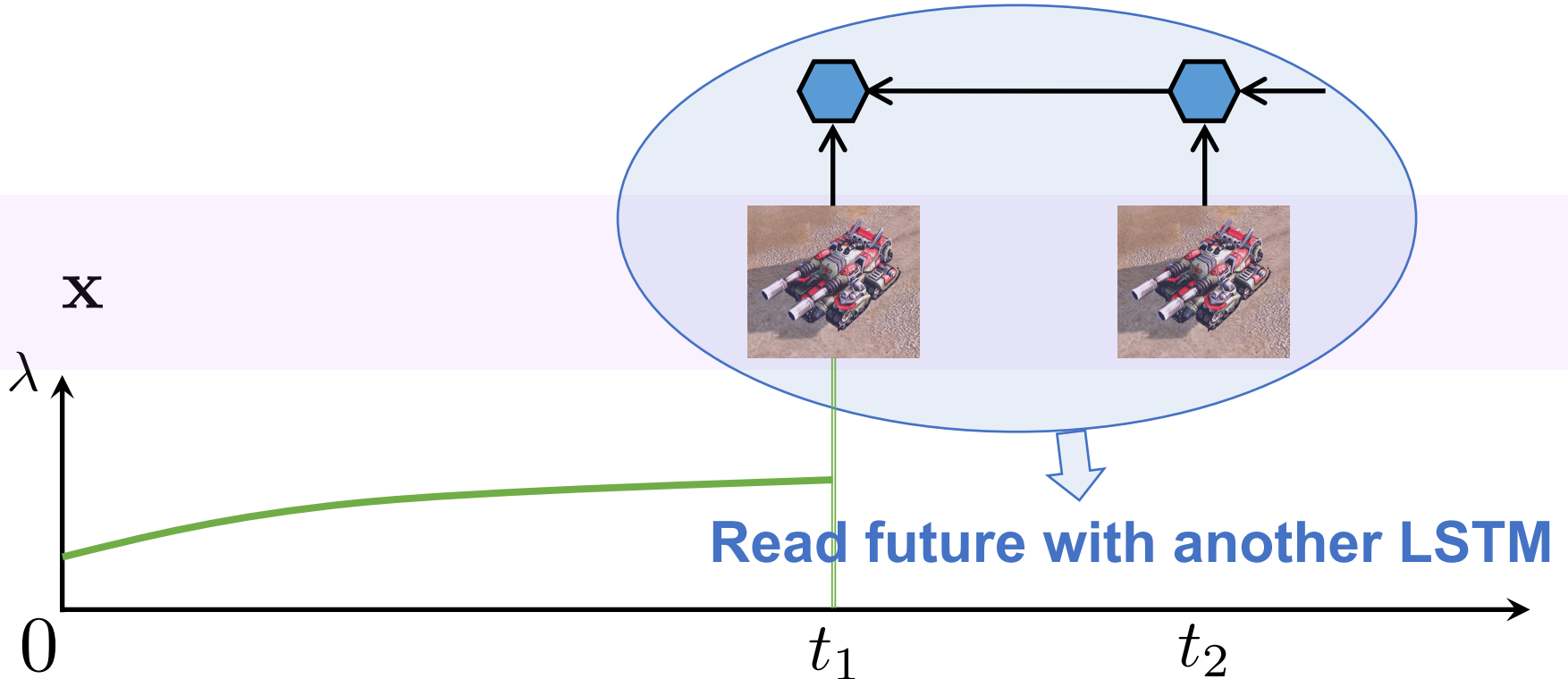
Sequential Monte Carlo

- Use the trained NHP
- Take future into account



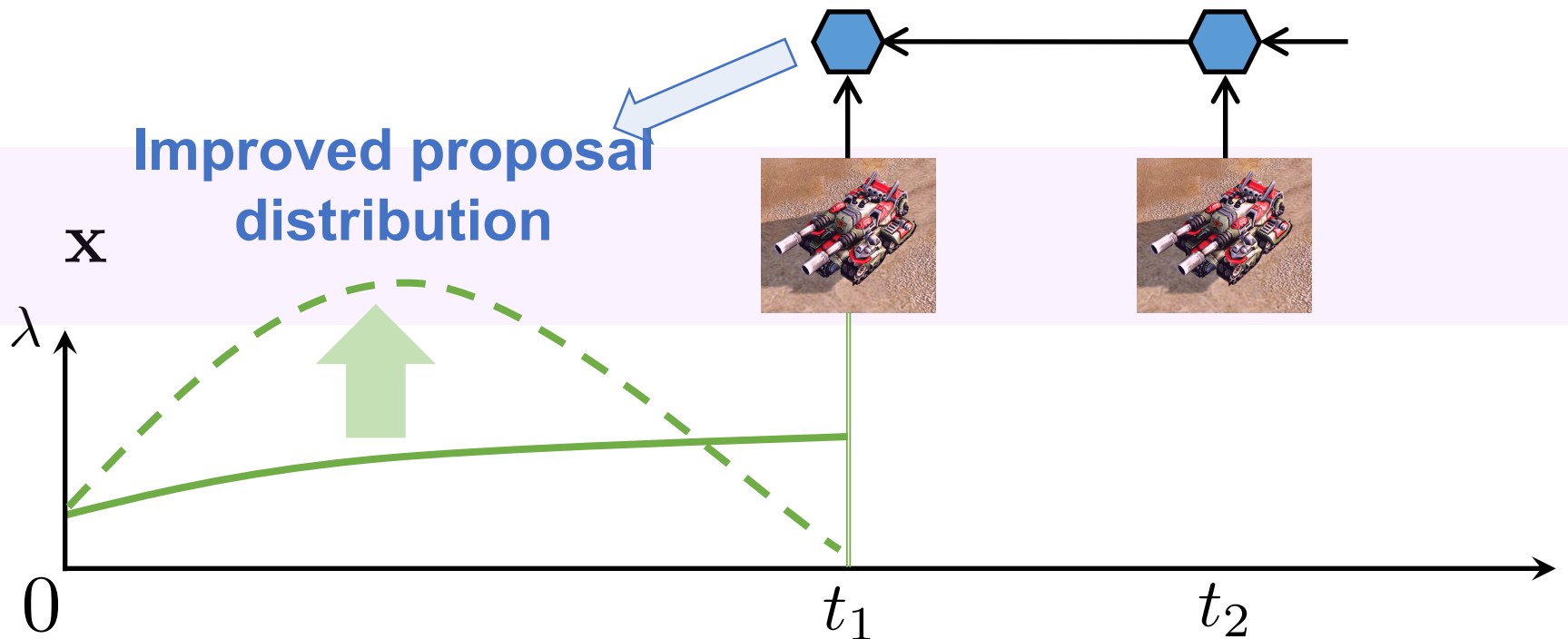
Sequential Monte Carlo

- Use the trained NHP
- Take future into account



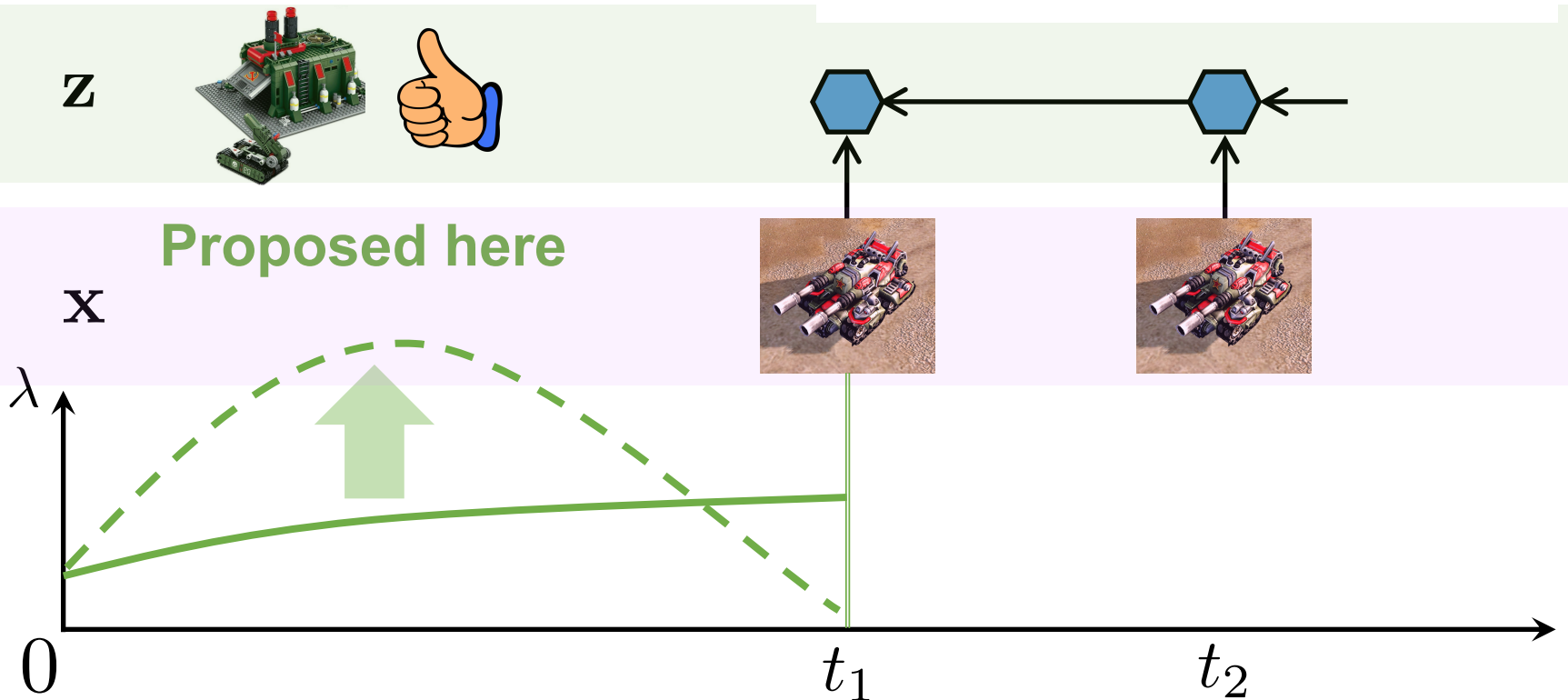
Sequential Monte Carlo

- Use the trained NHP
- Take future into account

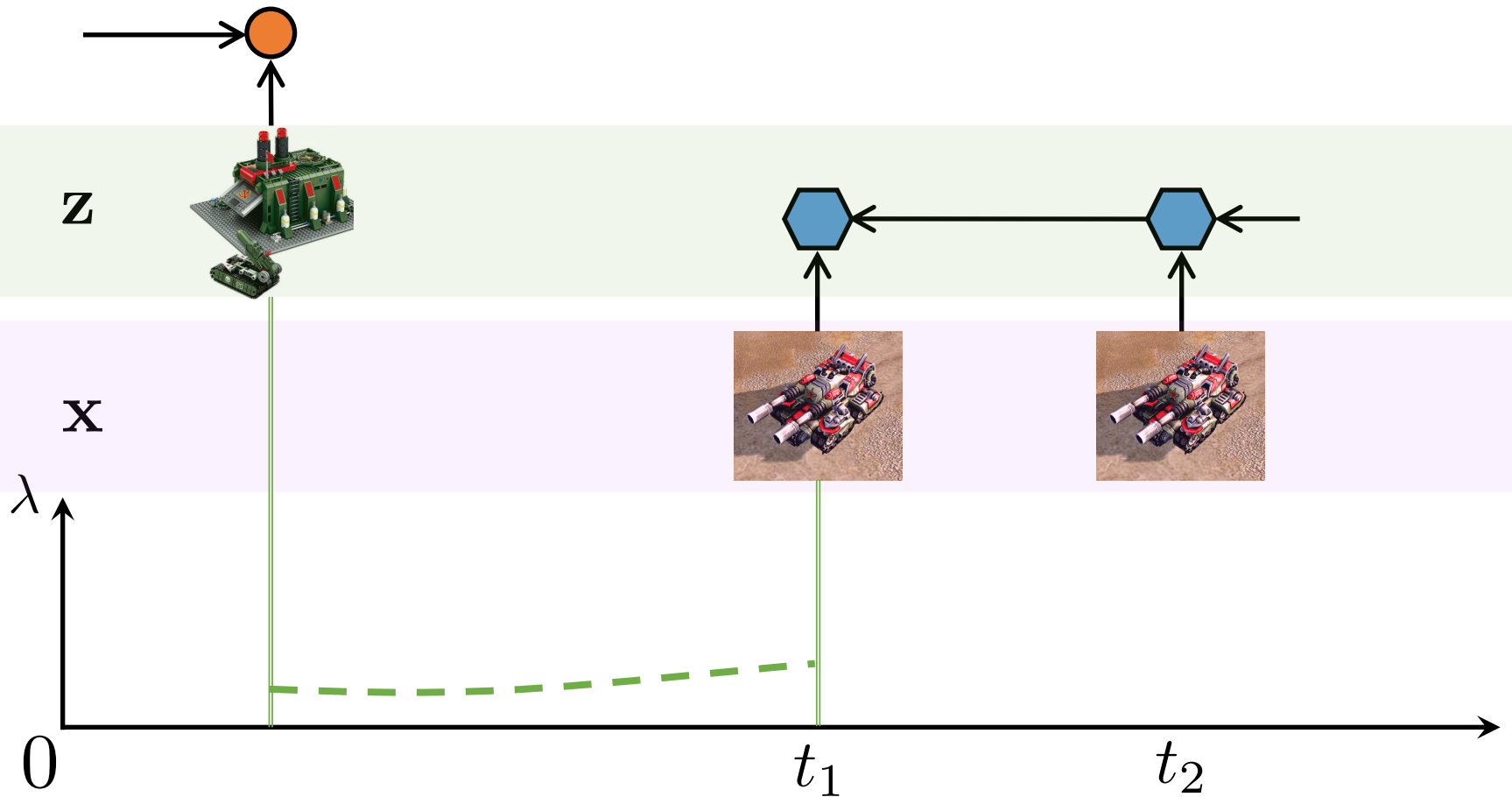


Sequential Monte Carlo

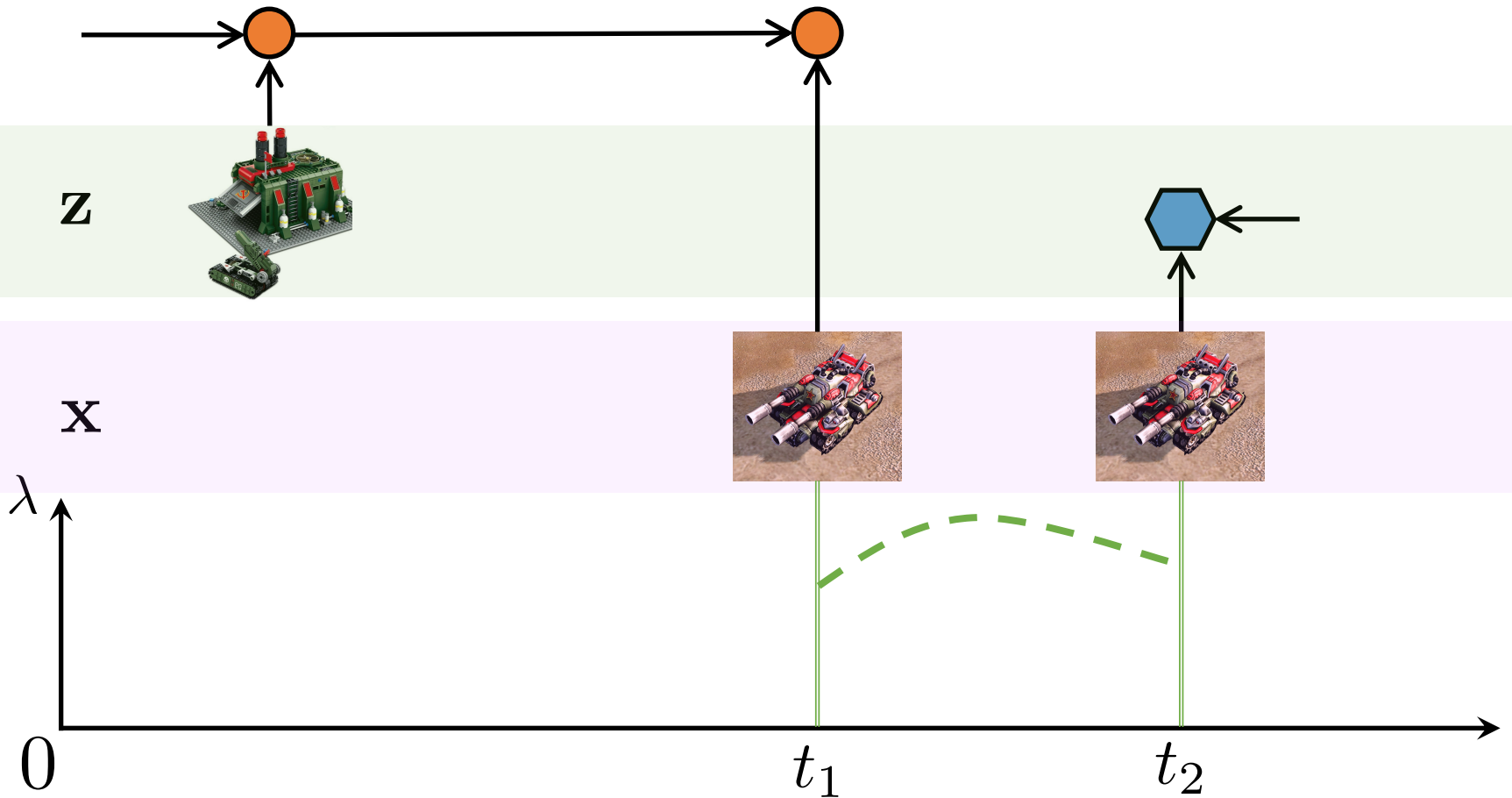
- Use the trained NHP
- Take future into account



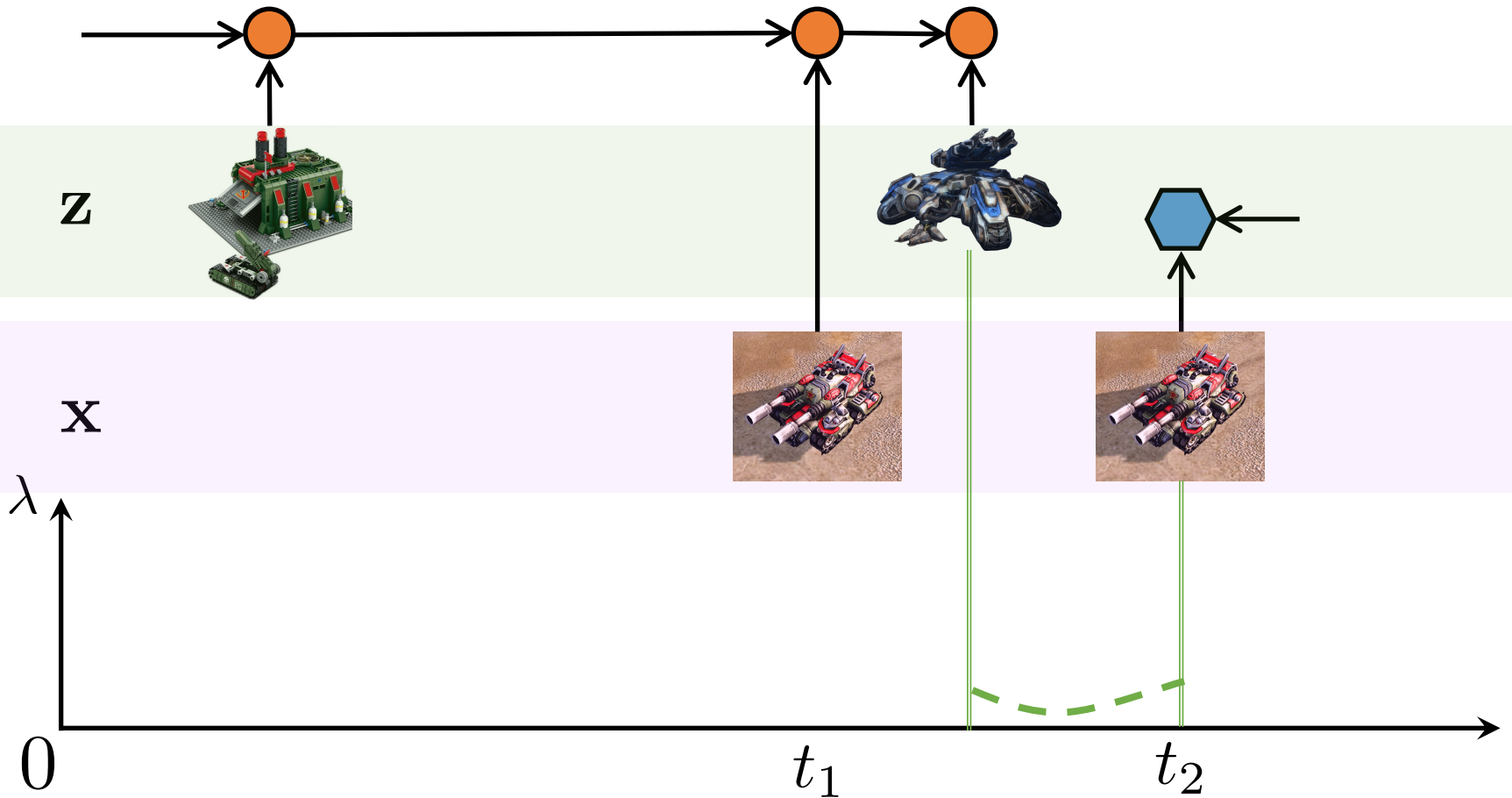
Sequential Monte Carlo



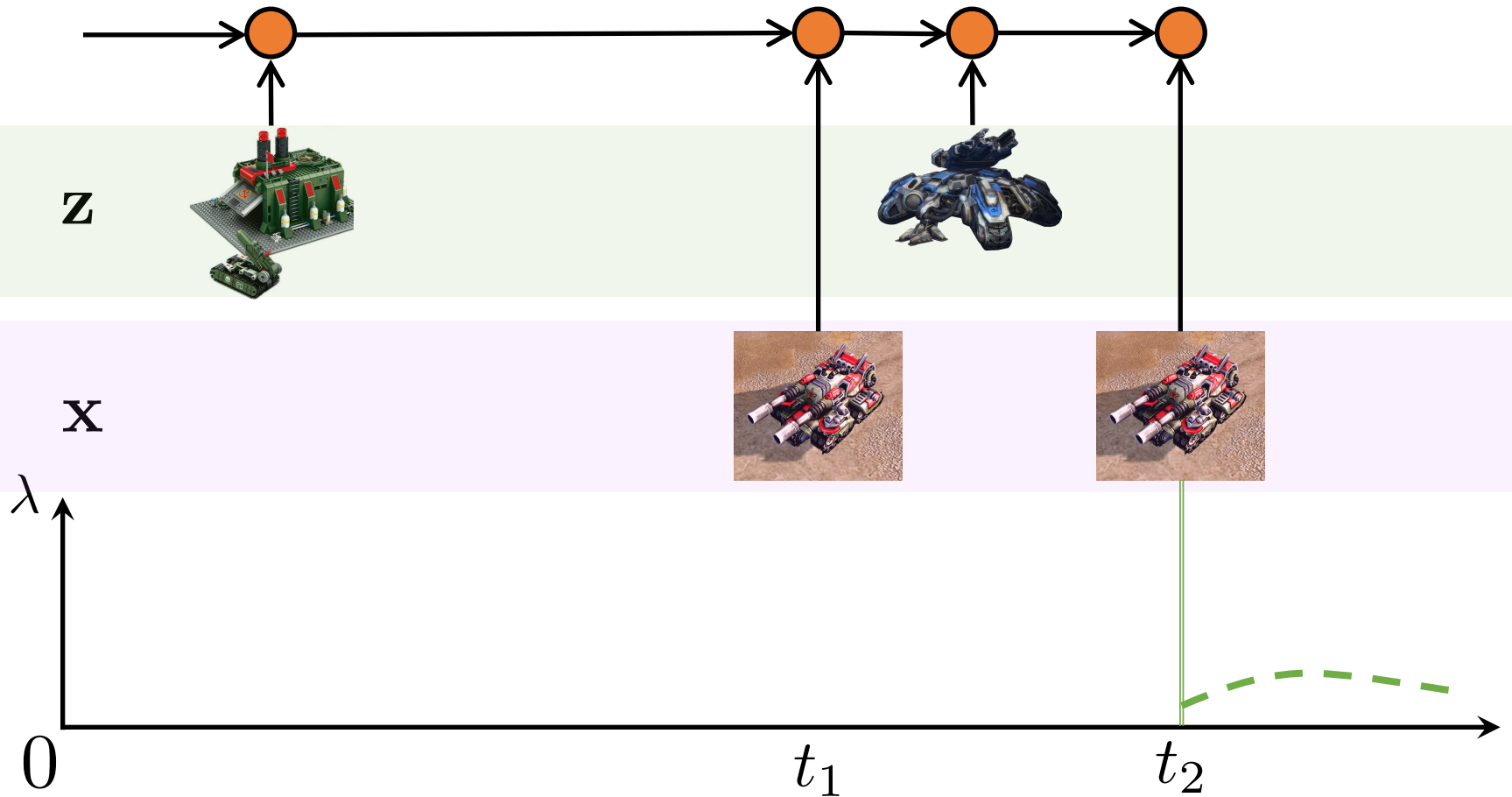
Sequential Monte Carlo



Sequential Monte Carlo



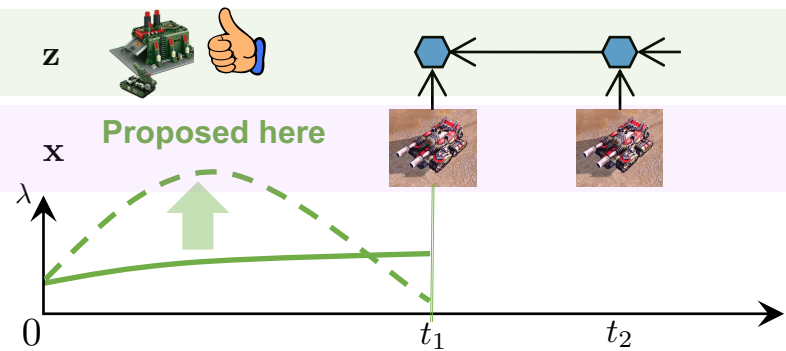
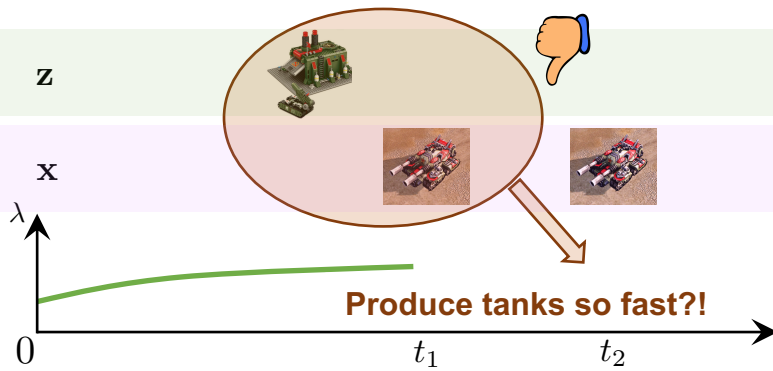
Sequential Monte Carlo



Sequential Monte Carlo

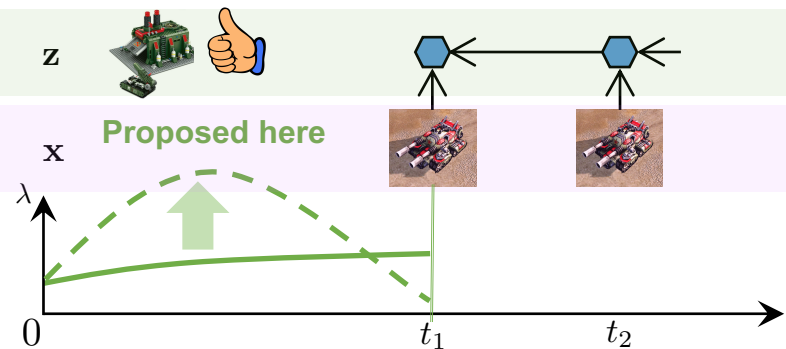
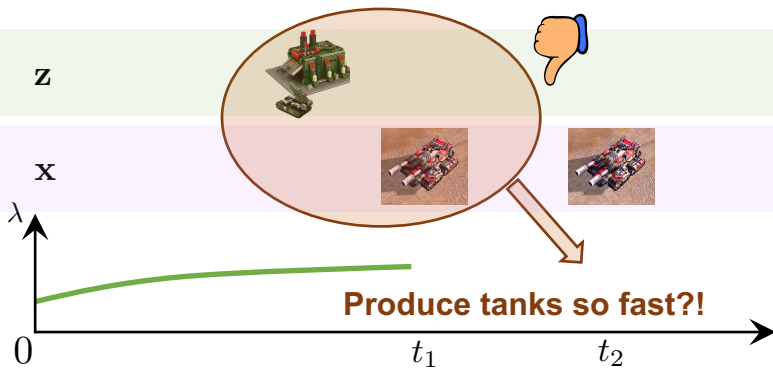
- Particle filtering

- Particle smoothing



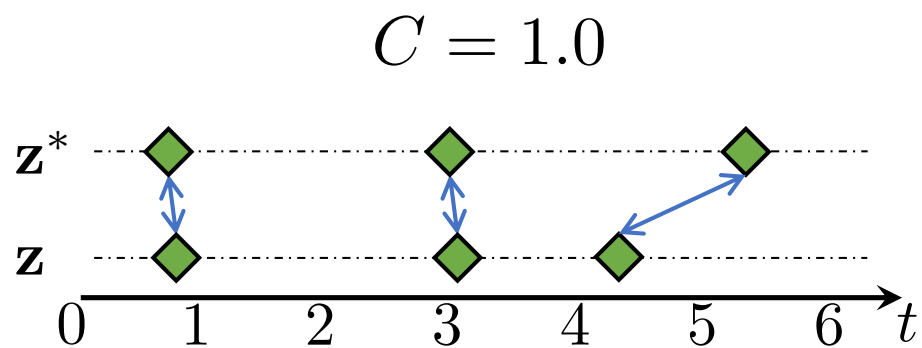
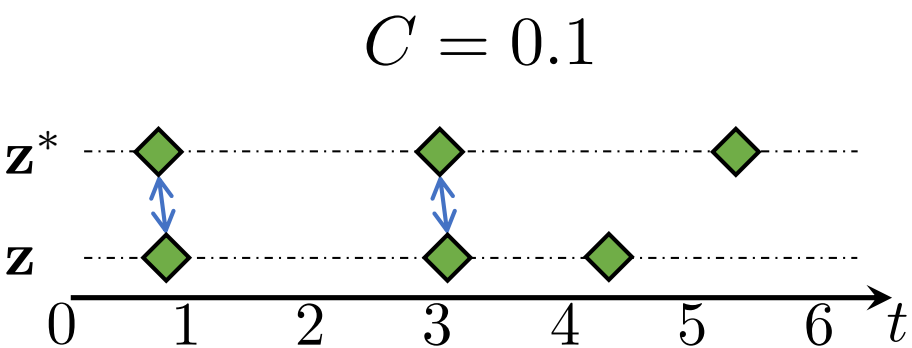
More in Our Paper

- How to train $q(\mathbf{z} \mid \mathbf{x})$ for particle smoothing



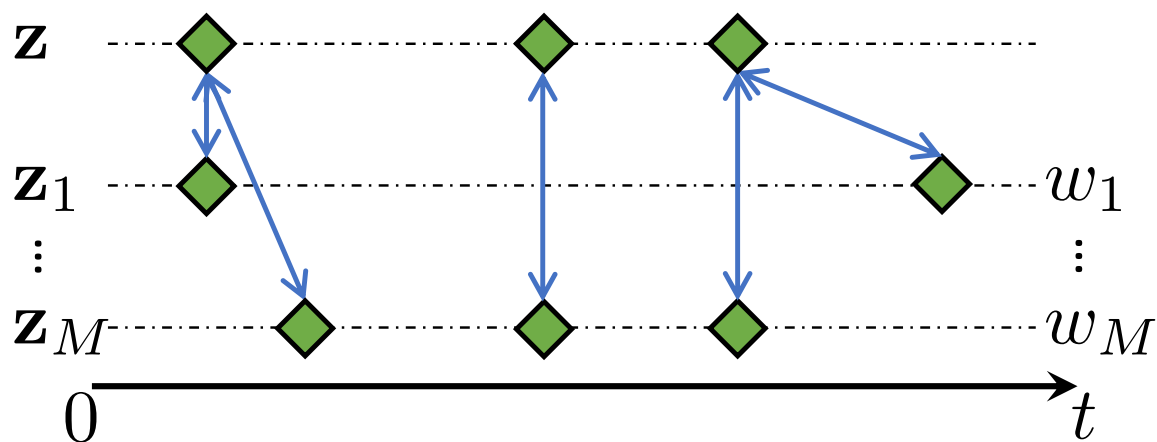
More in Our Paper

- How to train $q(\mathbf{z} \mid \mathbf{x})$ for particle smoothing
- Optimal transport distance



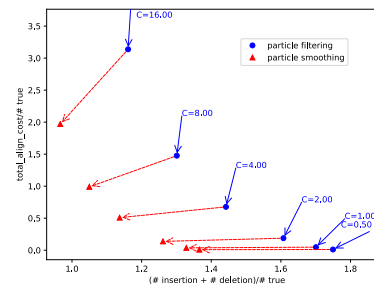
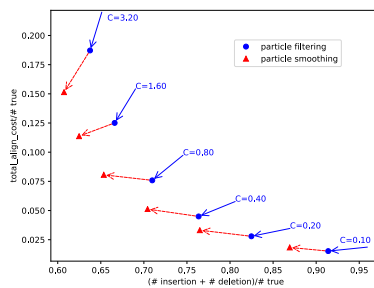
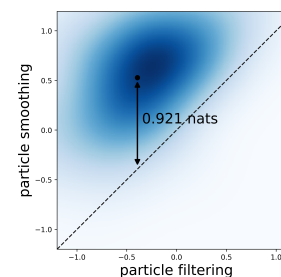
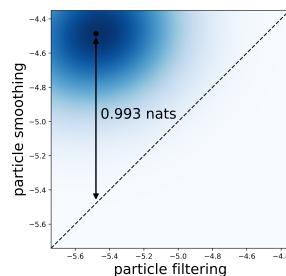
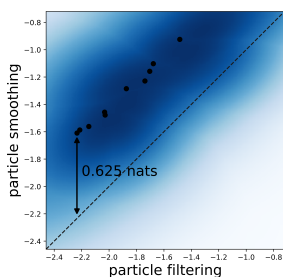
More in Our Paper

- How to train $q(\mathbf{z} \mid \mathbf{x})$ for particle smoothing
- Optimal transport distance
- Minimum Bayes risk decoding



More in Our Paper

- How to train $q(\mathbf{z} \mid \mathbf{x})$ for particle smoothing
- Optimal transport distance
- Minimum Bayes risk decoding
- Positive experimental results



Imputing Missing Events in Continuous-Time Event Streams

Tue Jun 11th 18:30 – 21:00
Pacific Ballroom #257