Denoised MDPs: Learning World Models Better Than The World Itself

Tongzhou Wang[1] Simon S. Du[2] Antonio Torralba[1] Phillip Isola[1] Amy Zhang[3,4] Yuandong Tian[3]

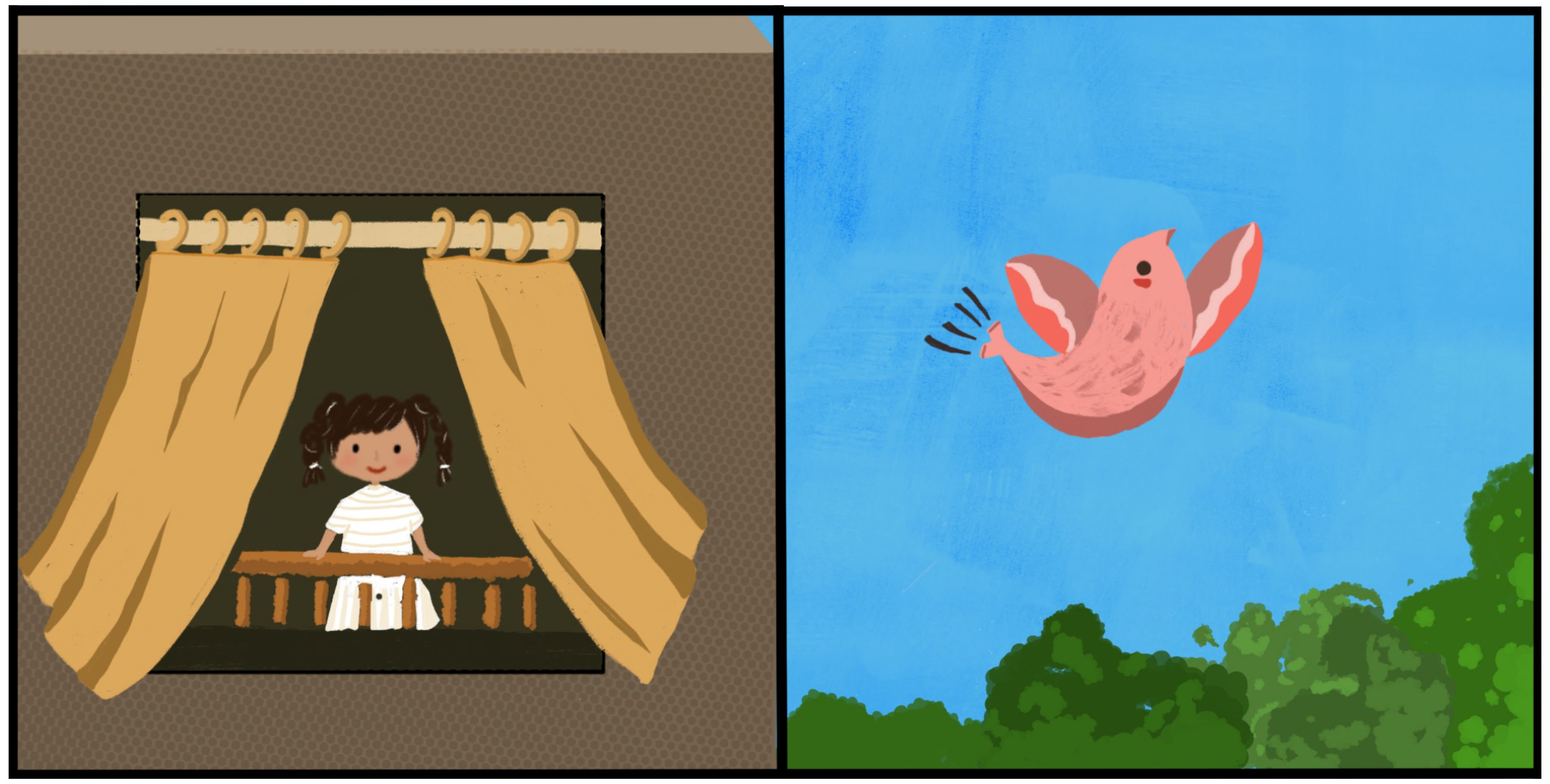
[1] MIT CSAIL [2] UNIVERSITY of WASHINGTON [3] Meta AI [4] Berkeley UNIVERSITY OF CALIFORNIA

Task solving in the noisy real world

Task solving in the noisy real world

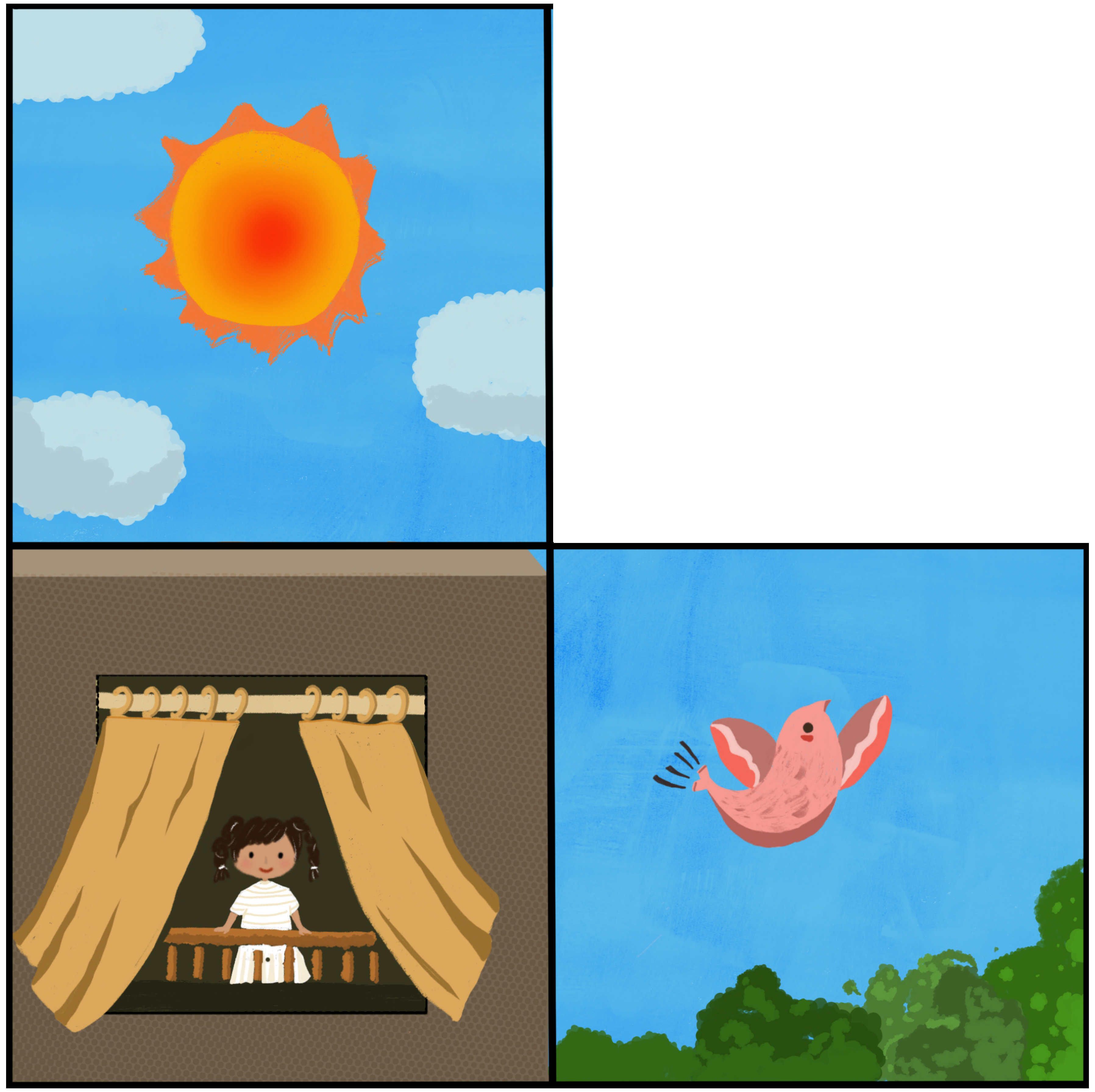
GOAL: Letting in as much sunlight as possible

Task solving in the noisy real world



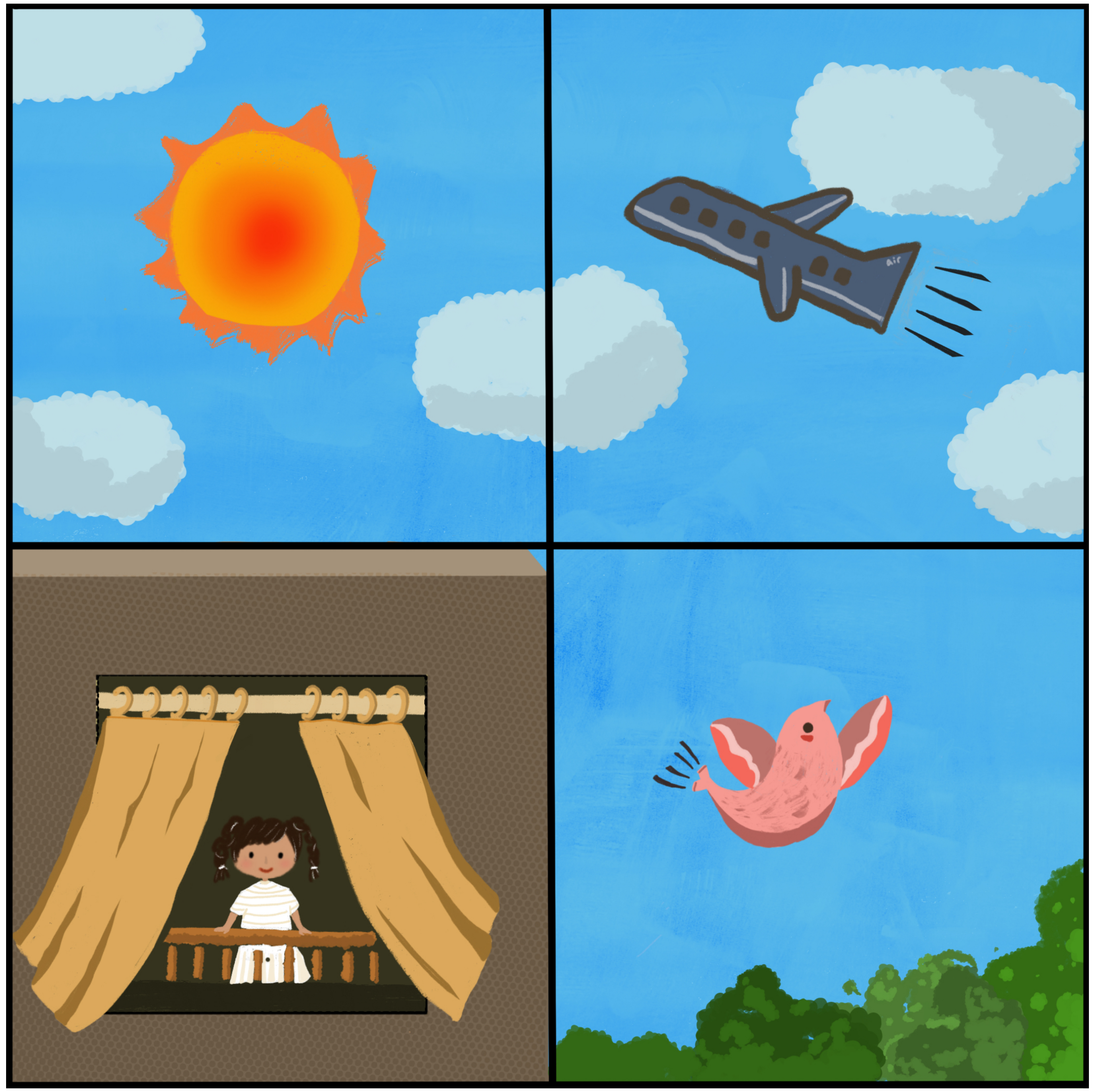
GOAL: Letting in as much sunlight as possible

Task solving in the noisy real world



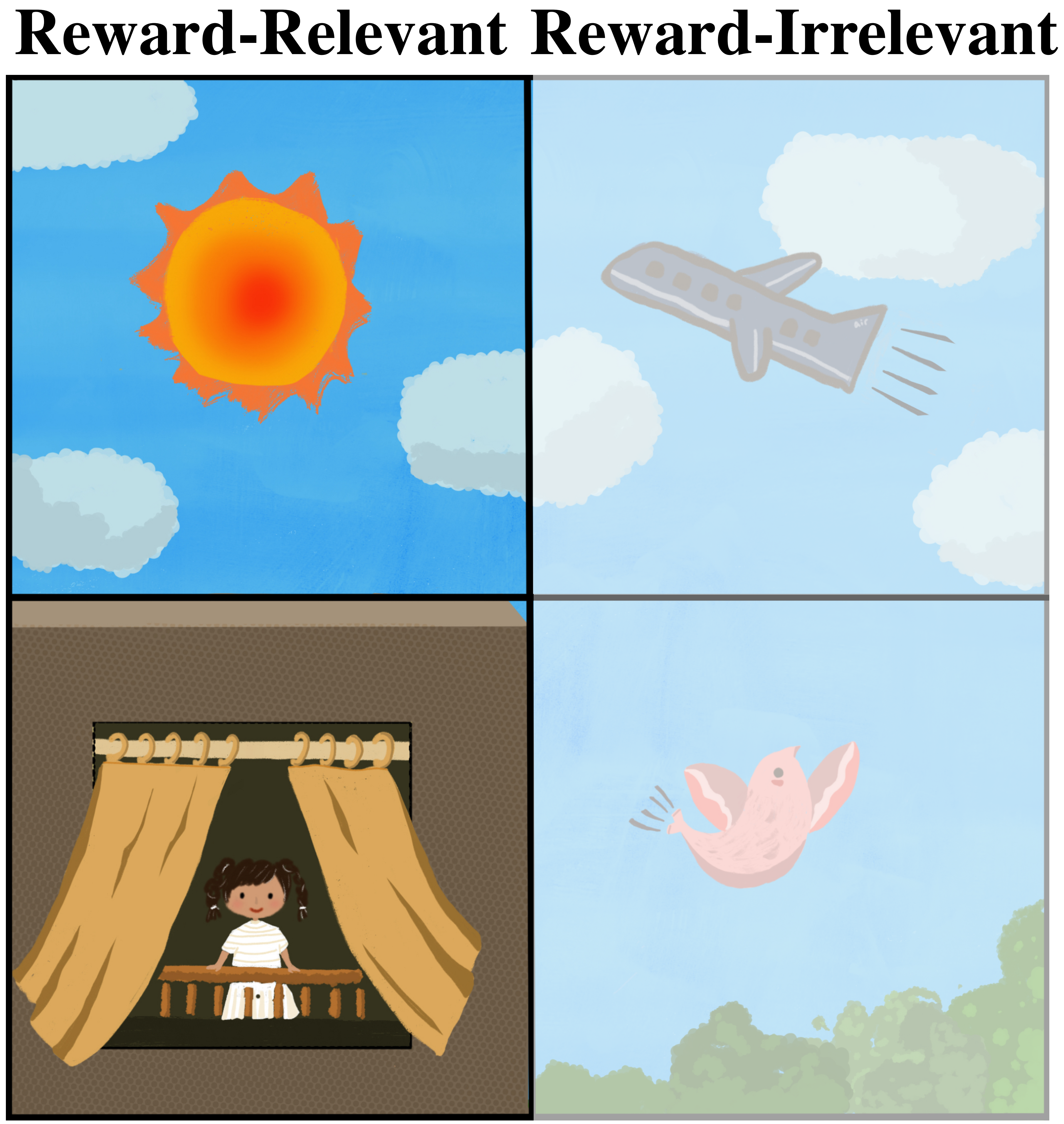
GOAL: Letting in as much sunlight as possible

Task solving in the noisy real world



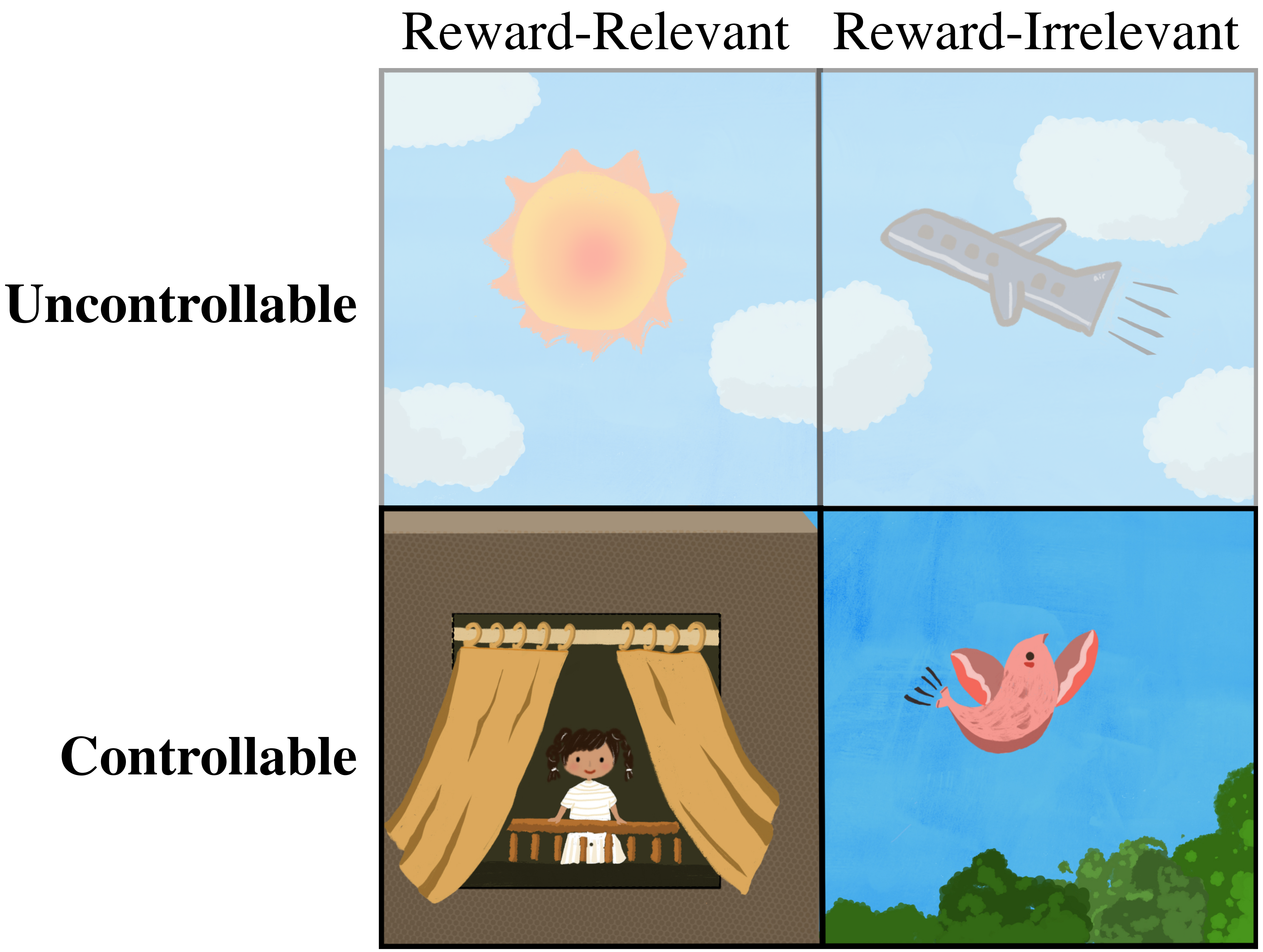
GOAL: Letting in as much sunlight as possible

Task solving in the noisy real world



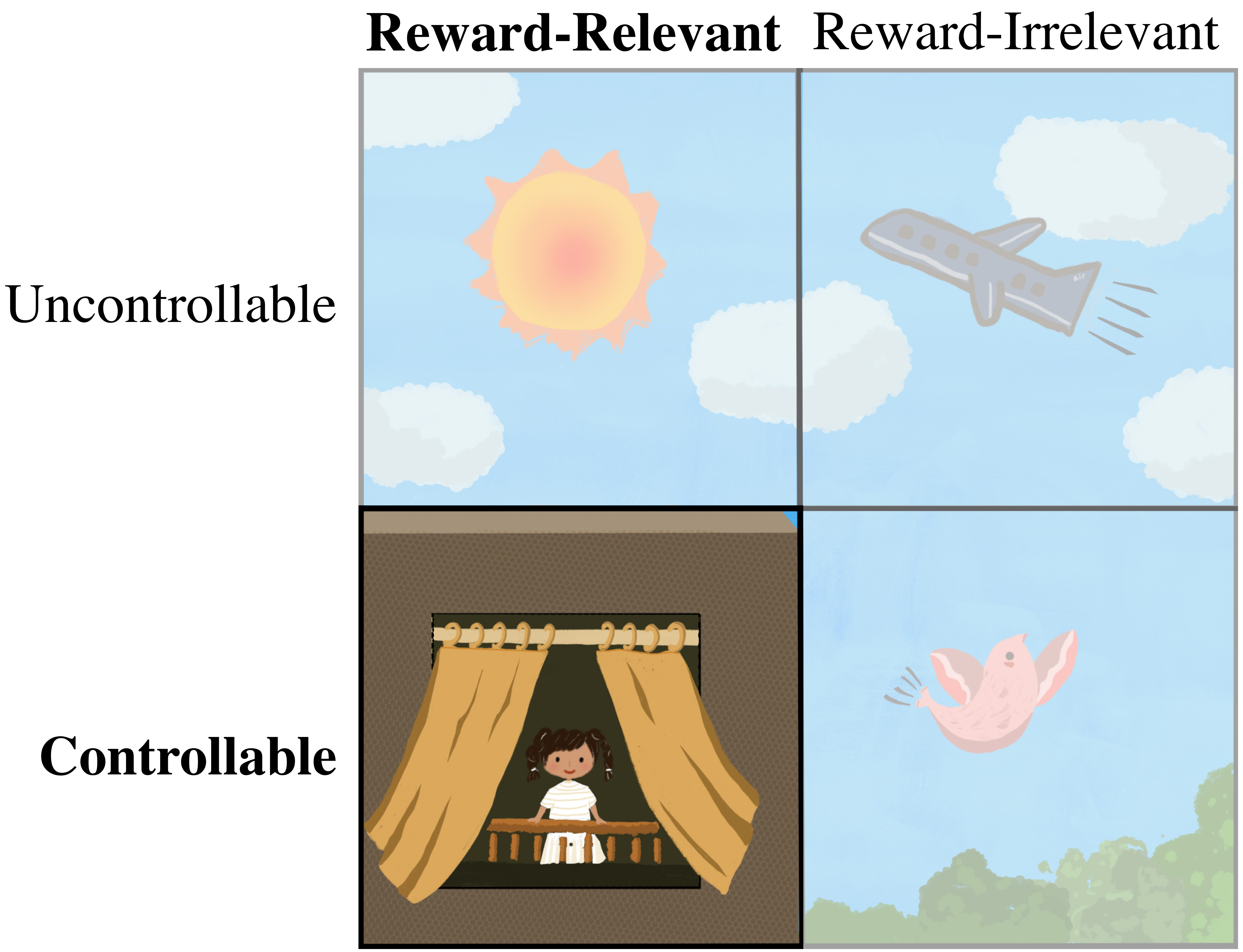
GOAL: Letting in as much sunlight as possible

Task solving in the noisy real world



GOAL: Letting in as much sunlight as possible

Task solving in the noisy real world



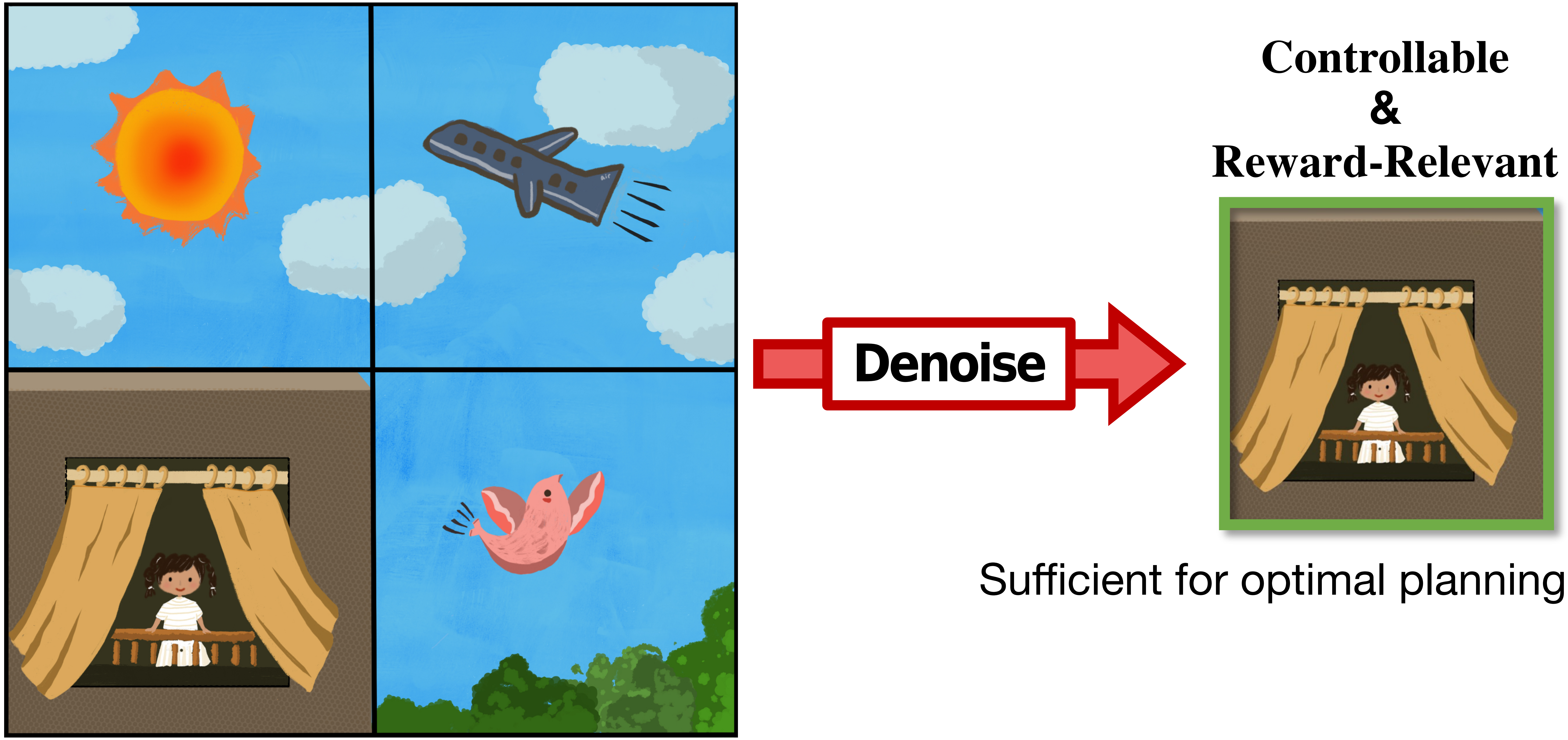
Reward-Relevant Reward-Irrelevant

Uncontrollable

Controllable

GOAL: Letting in as much sunlight as possible

Task solving in the noisy real world



Reward-Relevant Reward-Irrelevant

Uncontrollable

Controllable

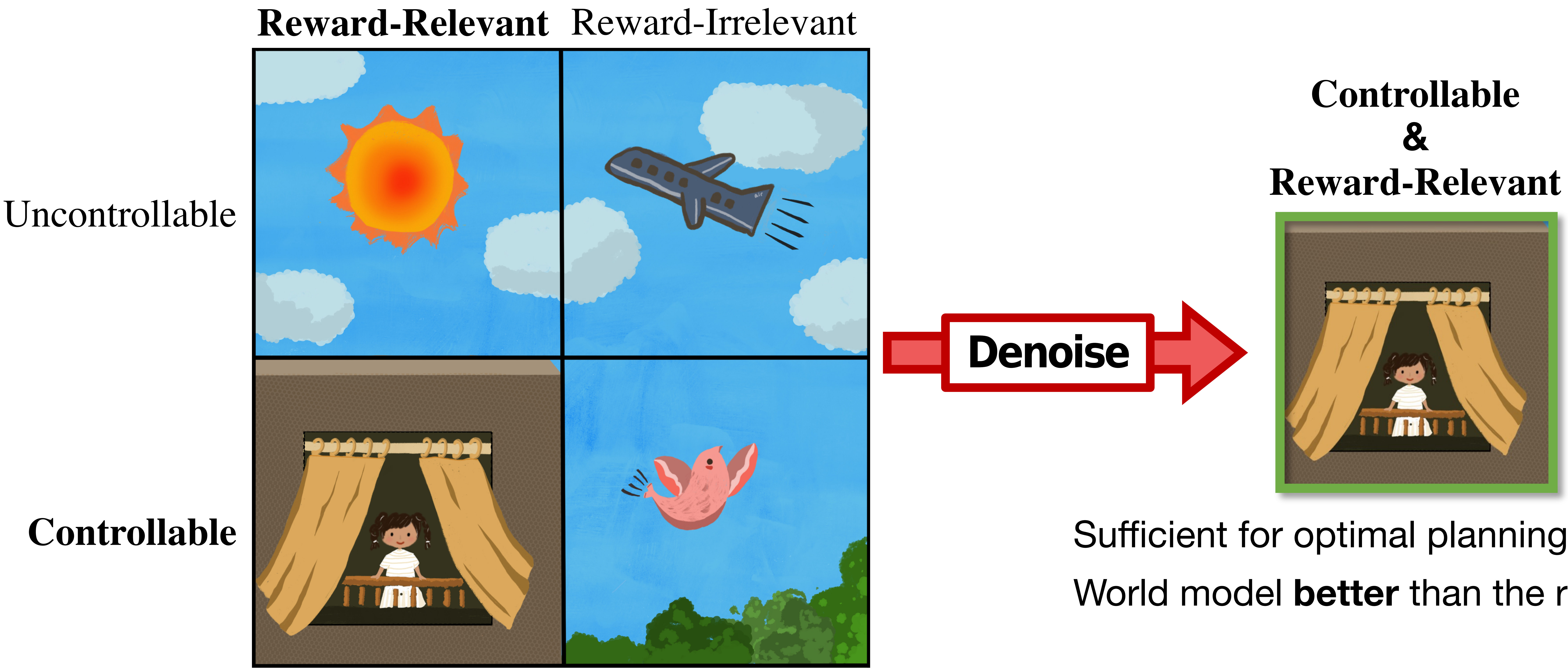
Denoise

Controllable
&
Reward-Relevant

Sufficient for optimal planning

GOAL: Letting in as much sunlight as possible

Task solving in the noisy real world



Reward-Relevant Reward-Irrelevant

Uncontrollable

Controllable

Denoise

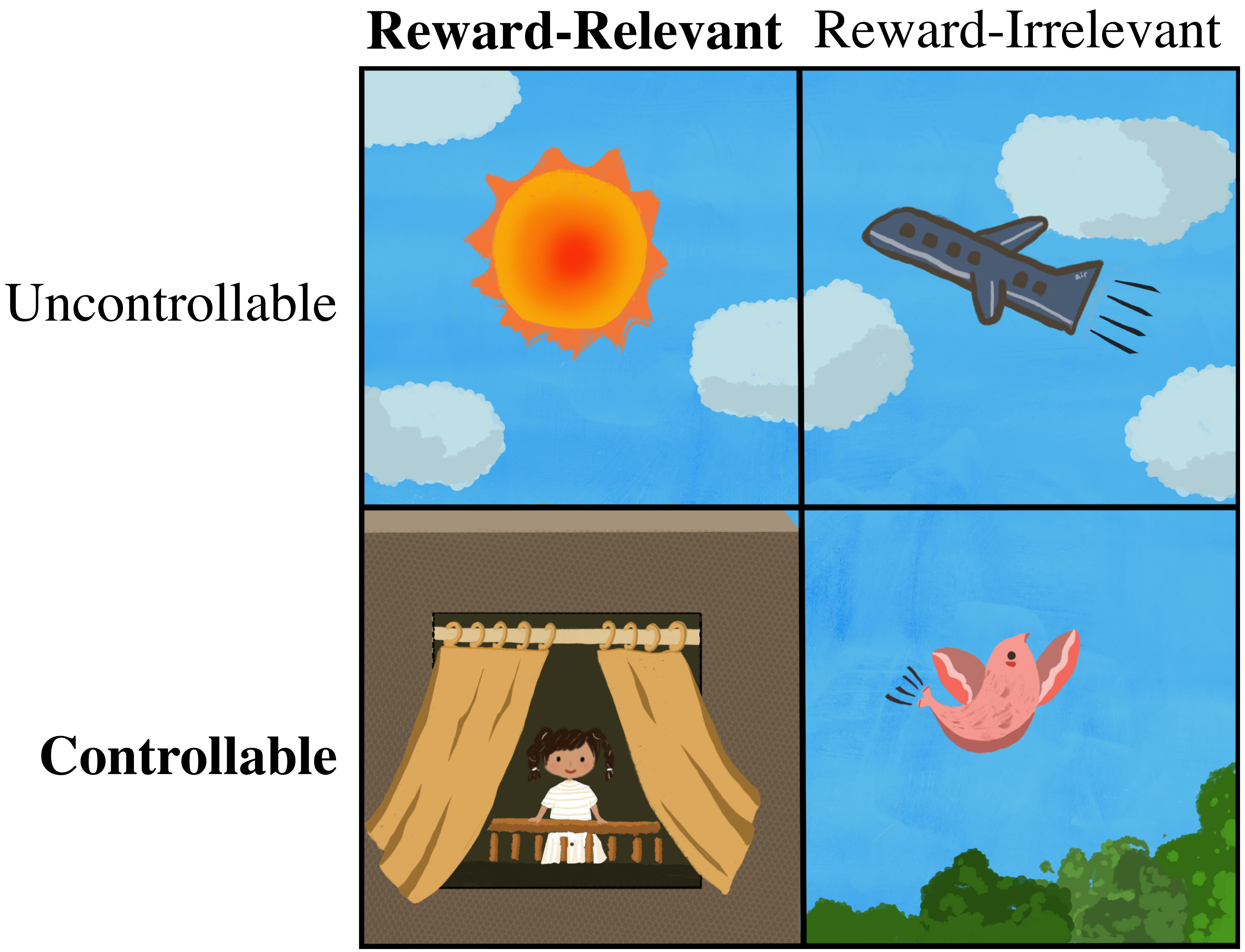
Controllable
&
Reward-Relevant

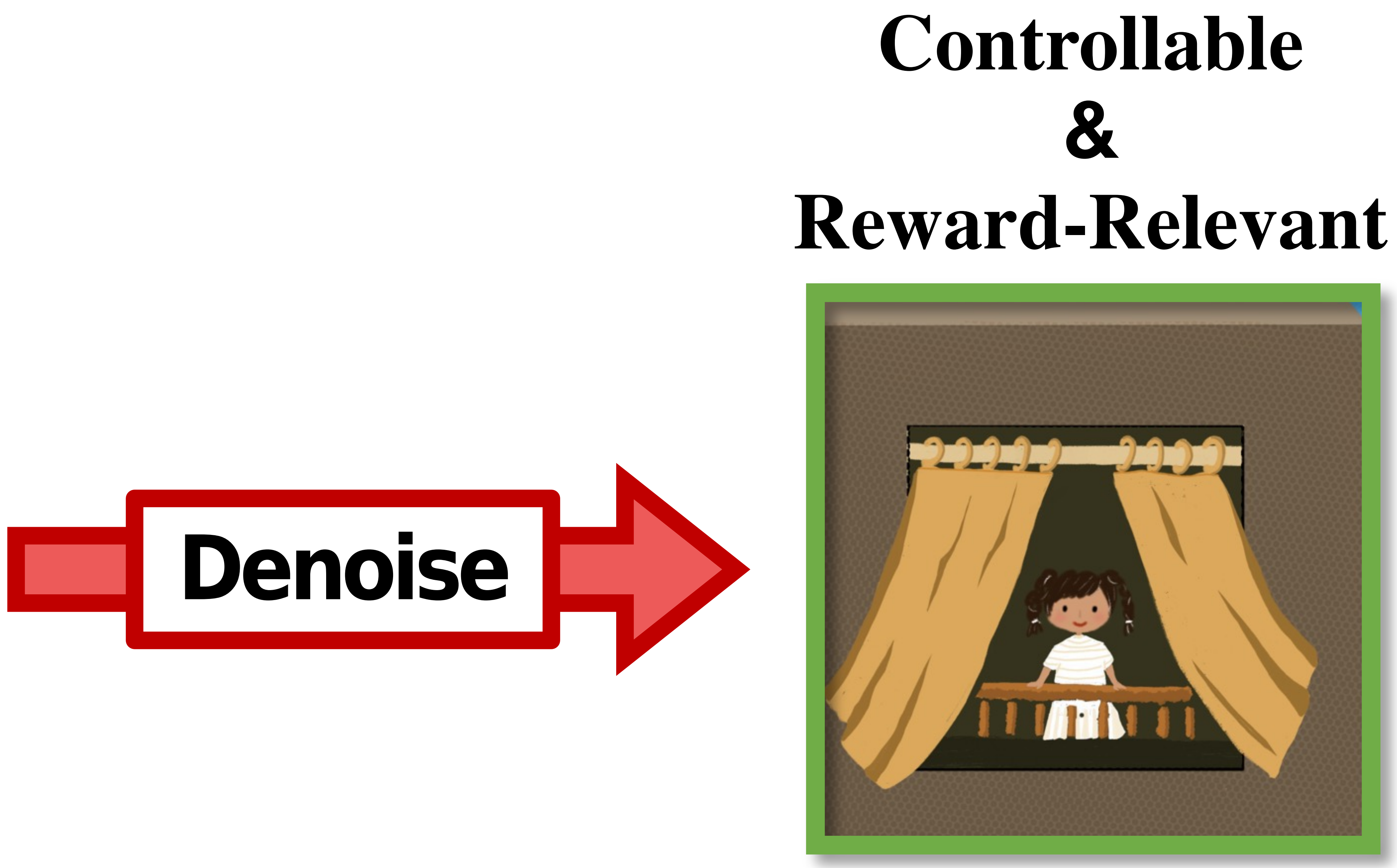
Sufficient for optimal planning

World model **better** than the real world

GOAL: Letting in as much sunlight as possible

Task solving in the noisy real world





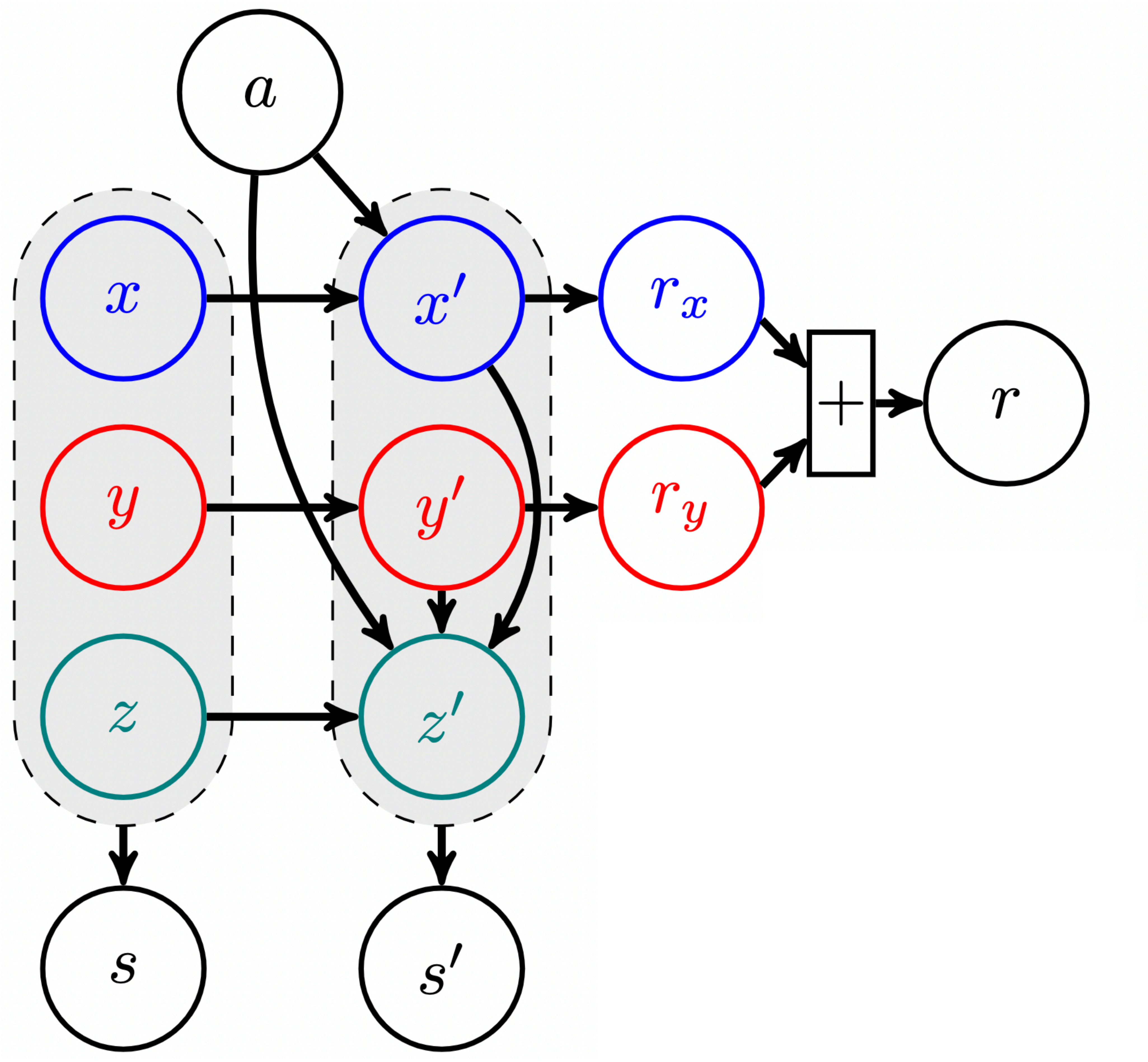
GOAL: Letting in as much sunlight as possible

Sufficient for optimal planning

World model **better** than the real world

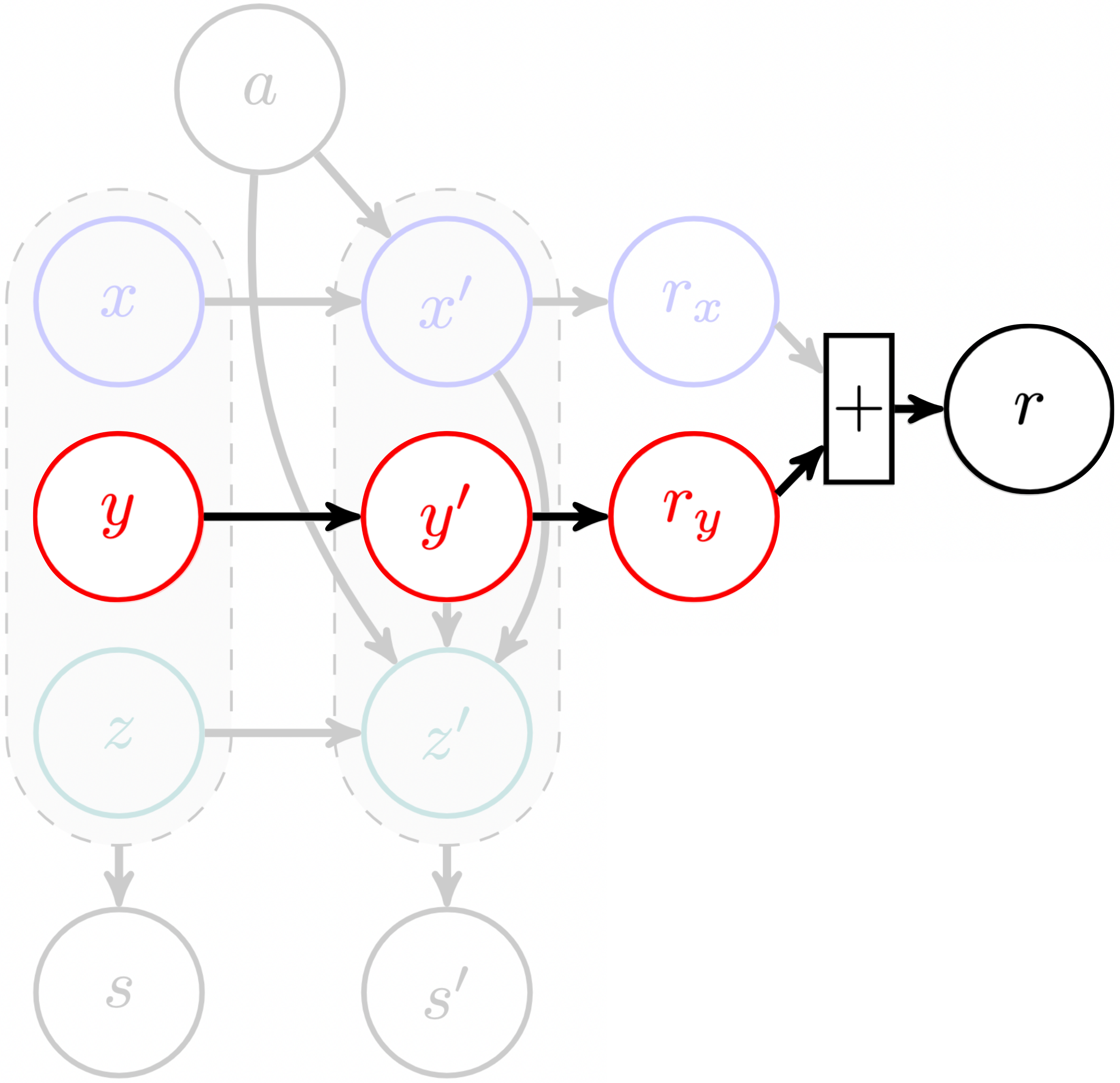
This work: fitting **factorized** transitions (with regularization) learns **denoised** models effectively

Identify noises via factorized transitions



Identify noises via factorized transitions

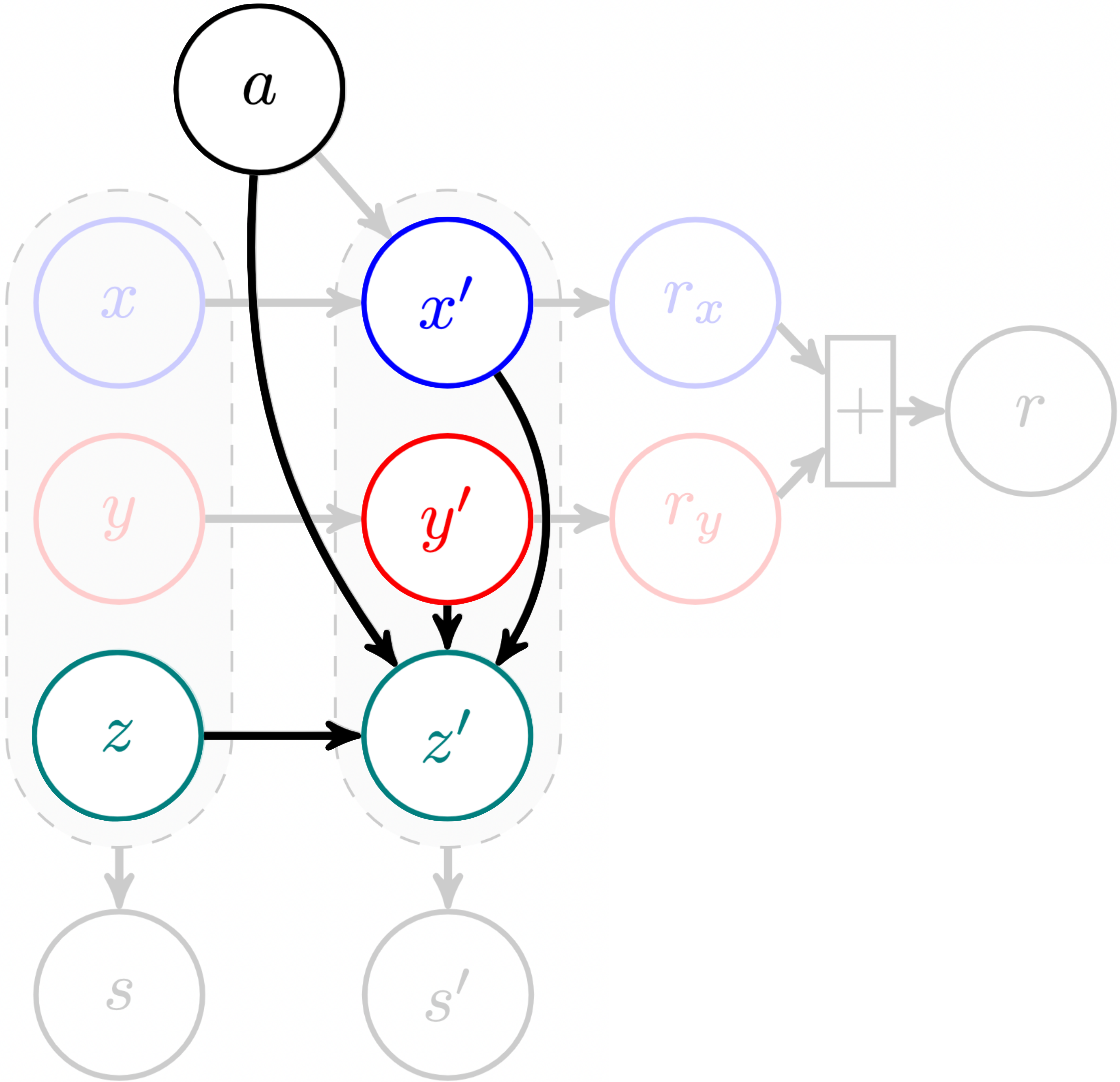
y **is uncontrollable:** not affected by actions a and only (possibly) additively affecting reward



Identify noises via factorized transitions

y **is uncontrollable:** not affected by actions a and only (possibly) additively affecting reward

z **is reward-irrelevant:** not affecting any other factor or reward

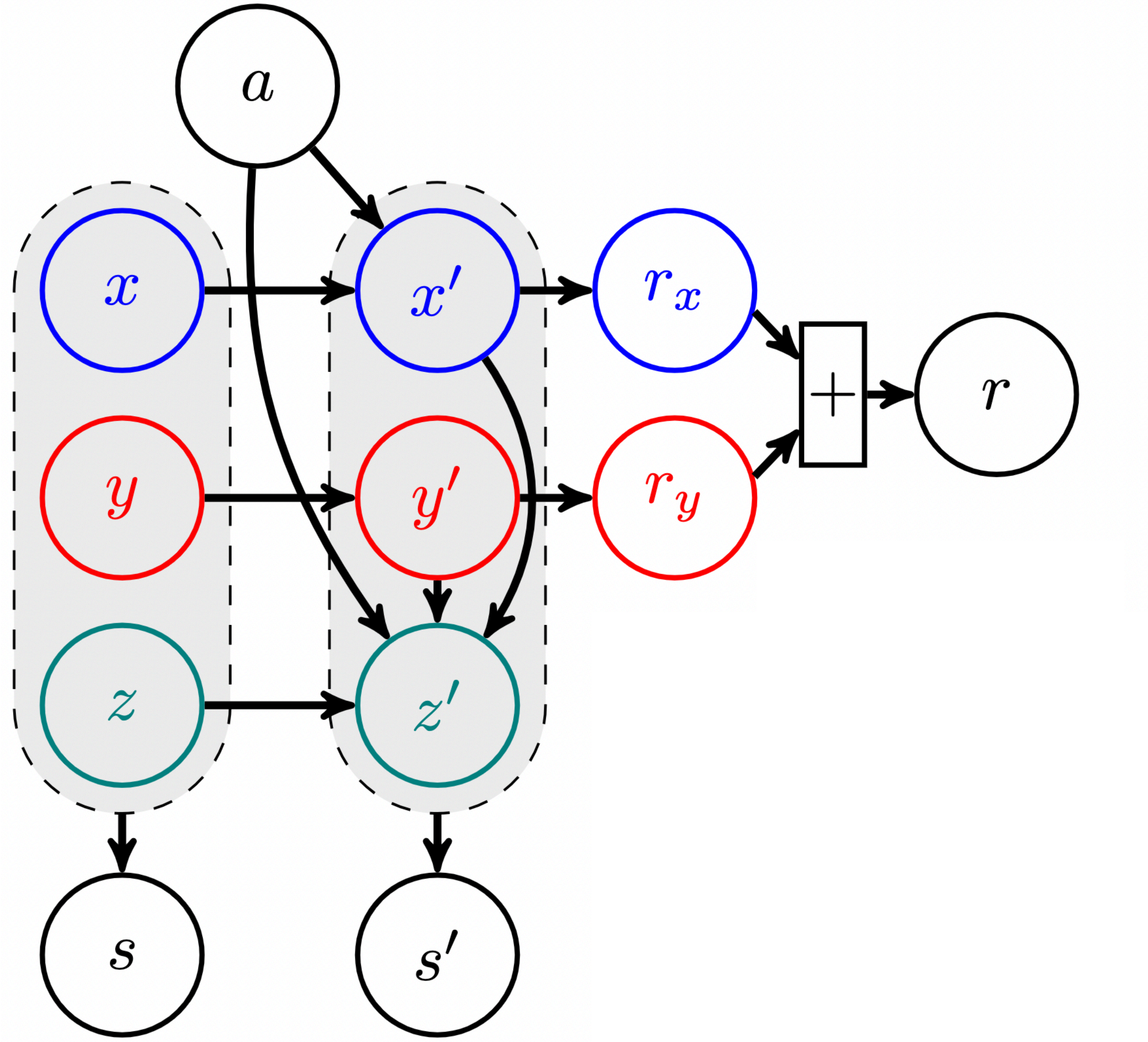


Identify noises via factorized transitions

x **contains all controllable & reward-relevant information**
x**'s dynamics are sufficient for optimal control**

y **is uncontrollable:** not affected by actions a and only (possibly) additively affecting reward

z **is reward-irrelevant:** not affecting any other factor or reward



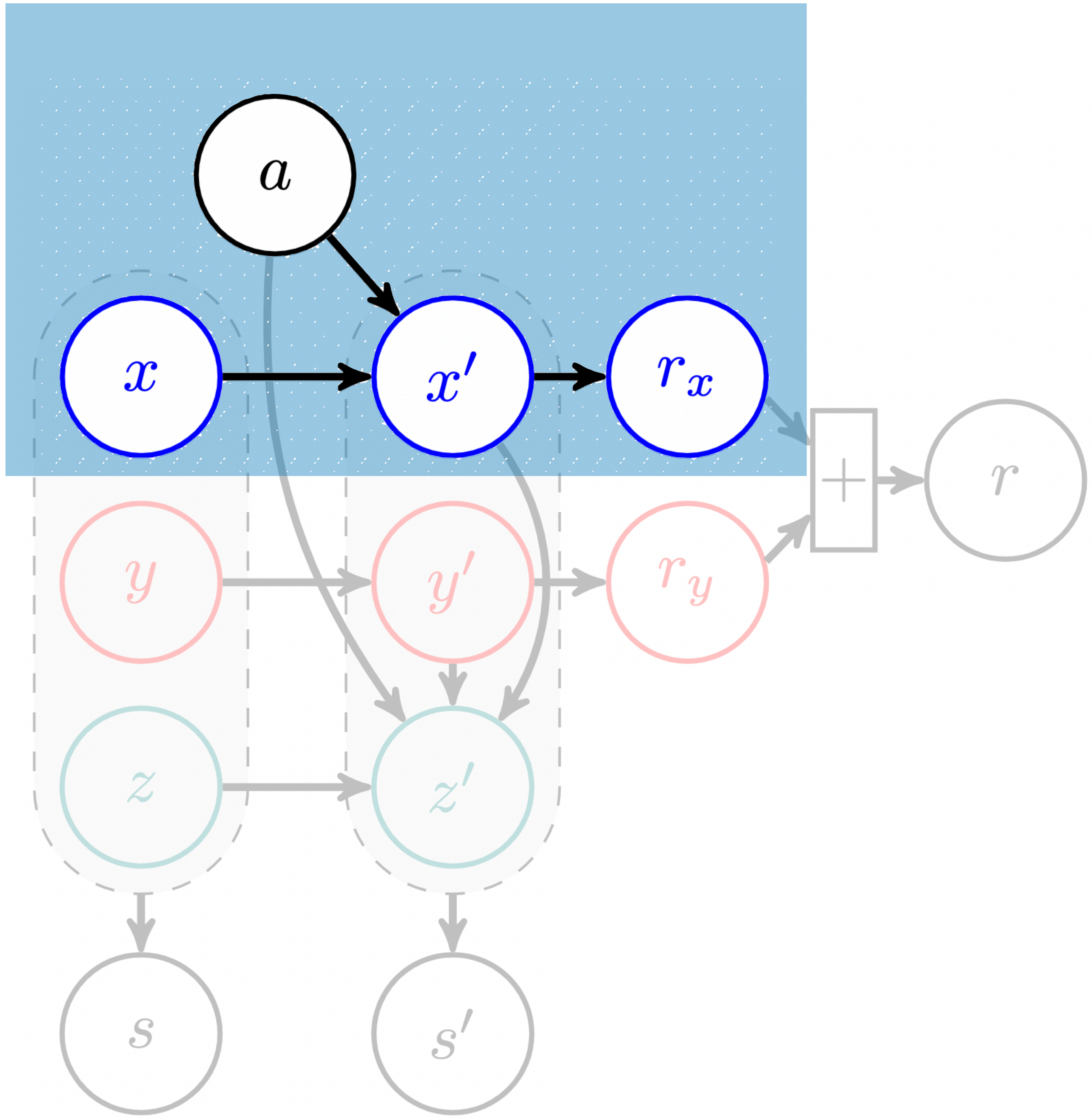
Identify noises via factorized transitions

x **contains all controllable & reward-relevant information**

x**'s dynamics are sufficient for optimal control**

y **is uncontrollable:** not affected by actions a and only (possibly) additively affecting reward

z **is reward-irrelevant:** not affecting any other factor or reward



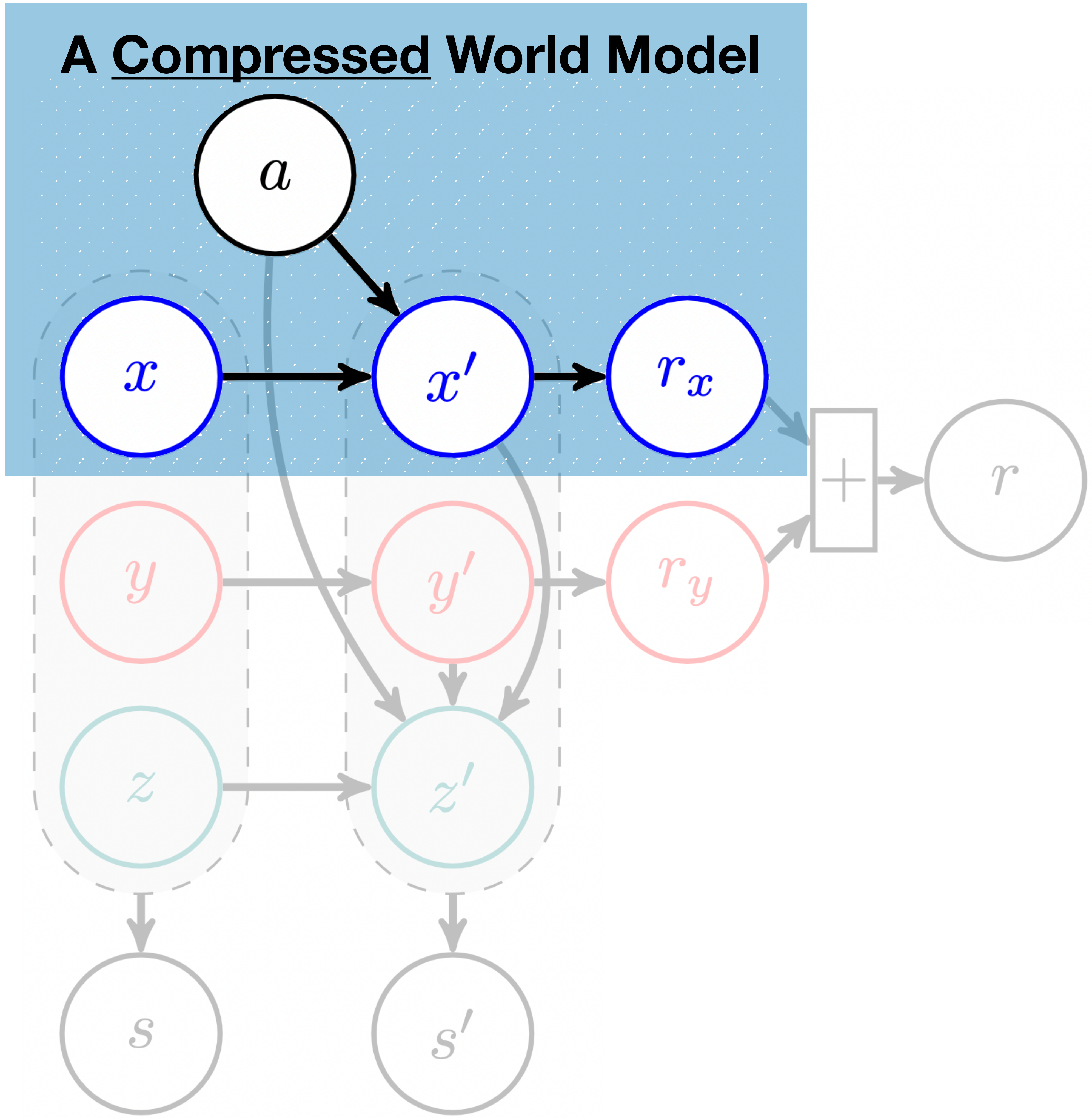
Identify noises via factorized transitions

x **contains all controllable & reward-relevant information**

x**'s dynamics are sufficient for optimal control**

y **is uncontrollable:** not affected by actions a and only (possibly) additively affecting reward

z **is reward-irrelevant:** not affecting any other factor or reward

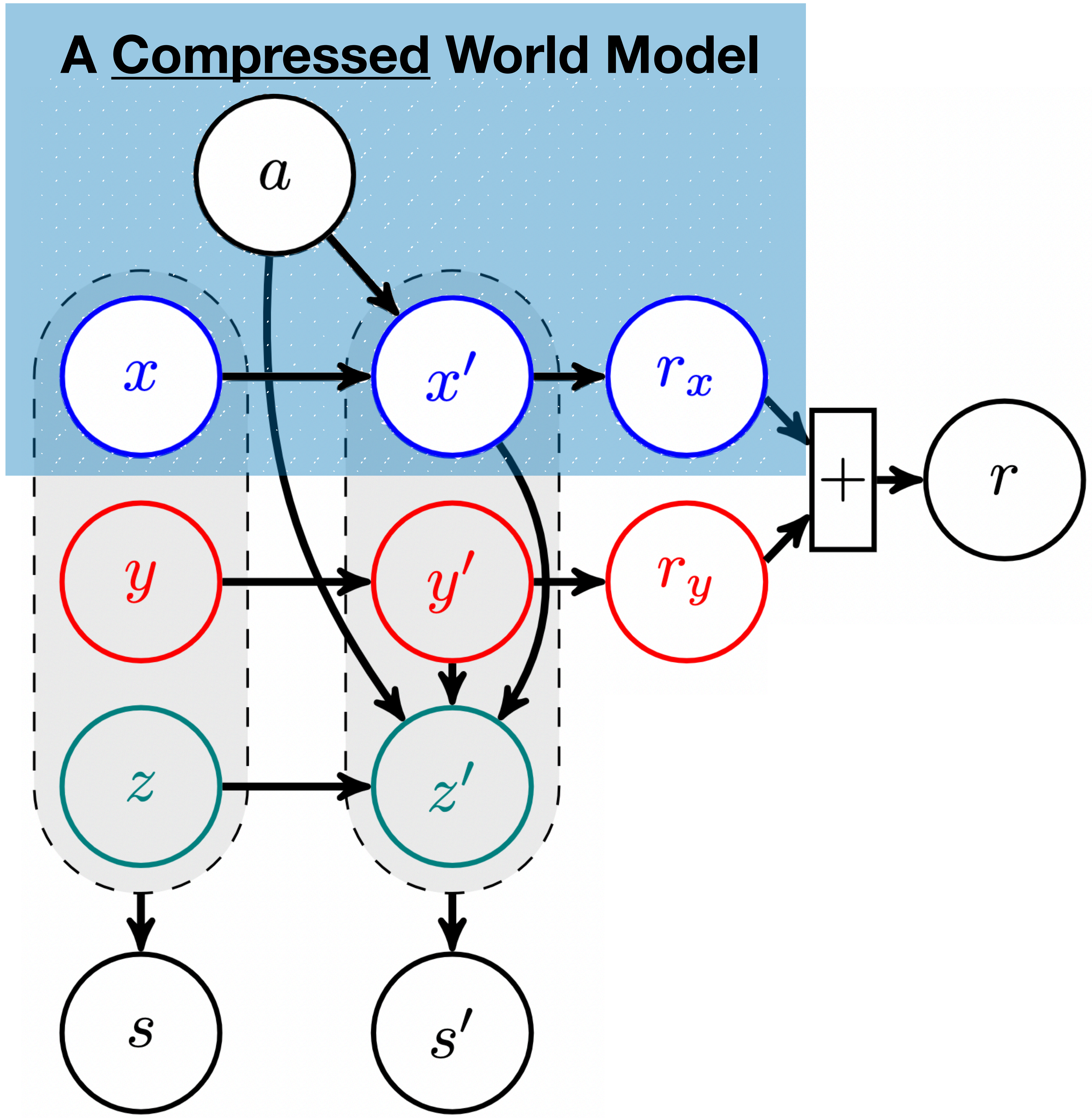


Identify noises via factorized transitions

x contains all controllable & reward-relevant information
x's dynamics are sufficient for optimal control

Denoised MDP

Factorized Model ⟶ Compressed Model


A Compressed World Model

Identify noises via factorized transitions

x contains all controllable & reward-relevant information

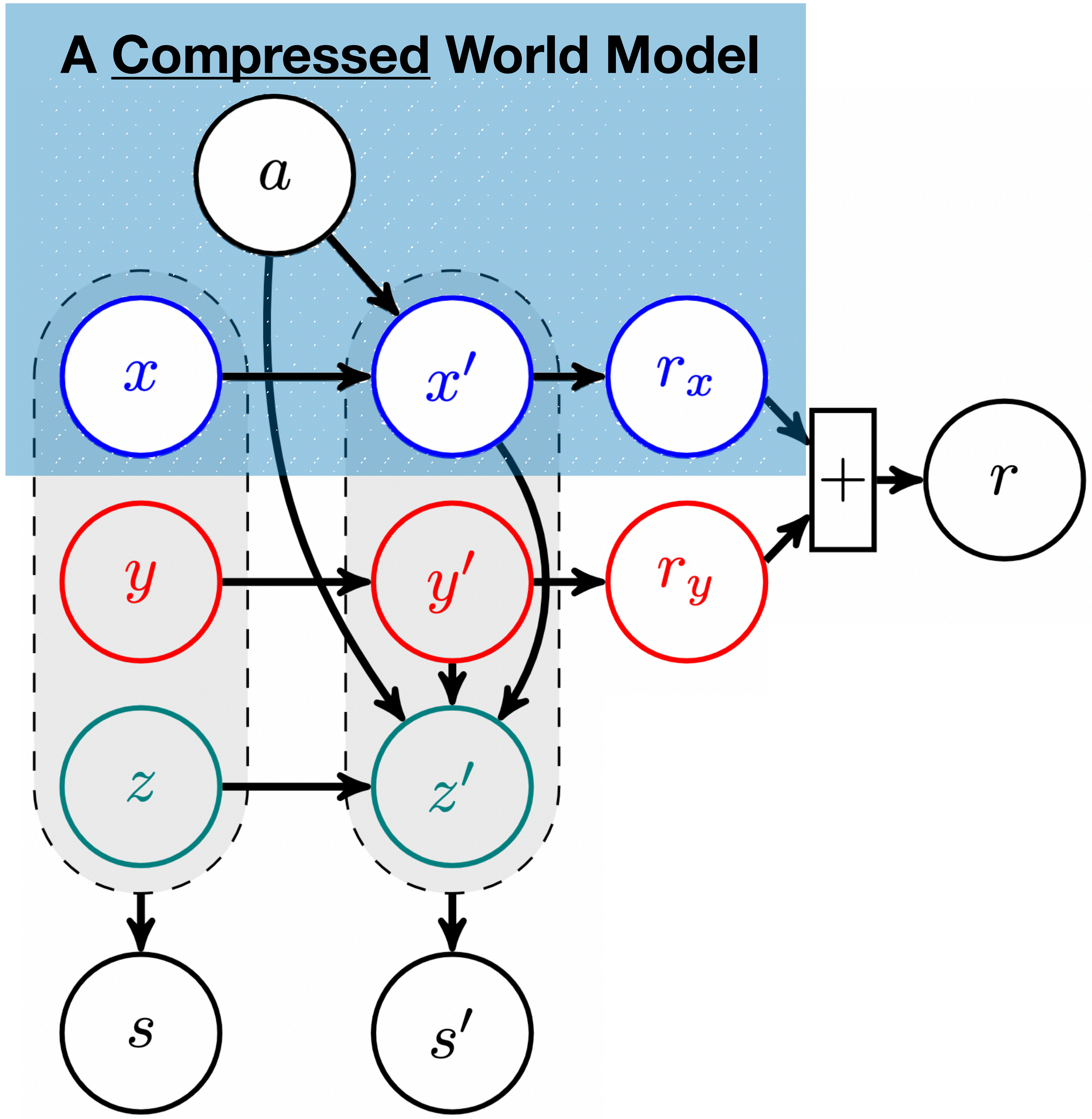
x's dynamics are sufficient for optimal control

Denoised MDP

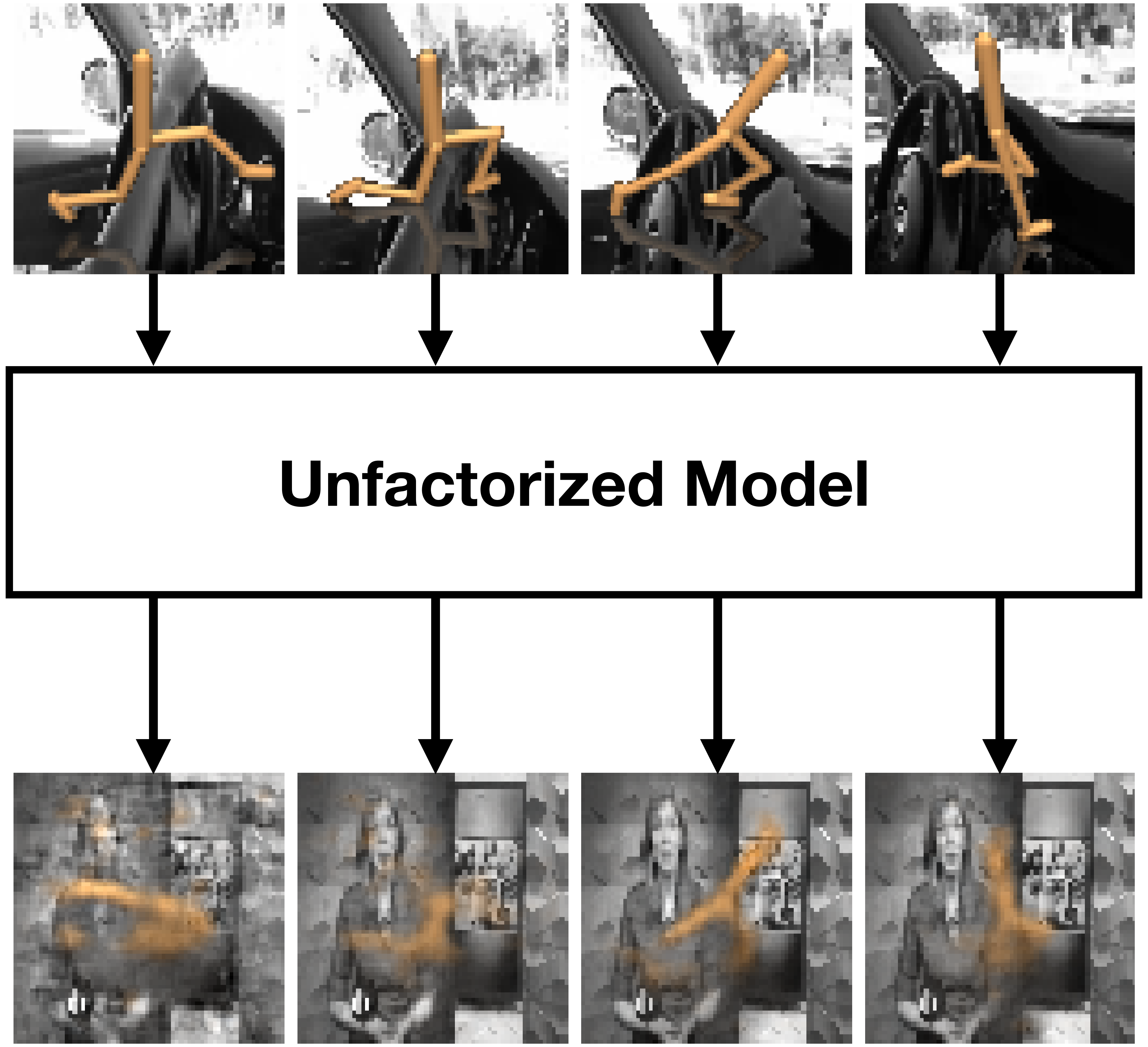
Factorized Model ⟶ Compressed Model

Algorithm

1. Fit such factorized model, regularize $I(x; \text{obs})$
2. Only use the Denoised MDP for policy training



Denoised MDP: signal-noise separation

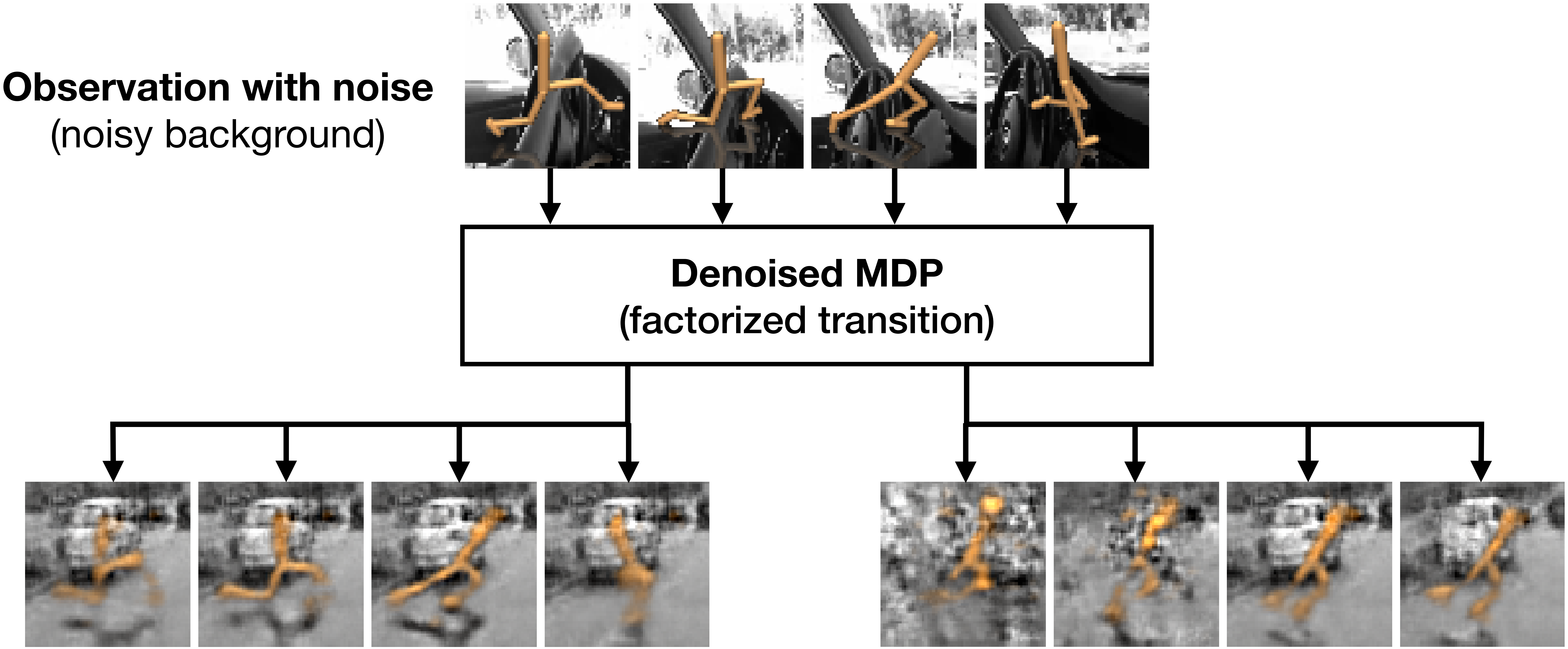


Observation with noise
(noisy background)

Unfactorized Model

Model reconstruction

Denoised MDP: signal-noise separation



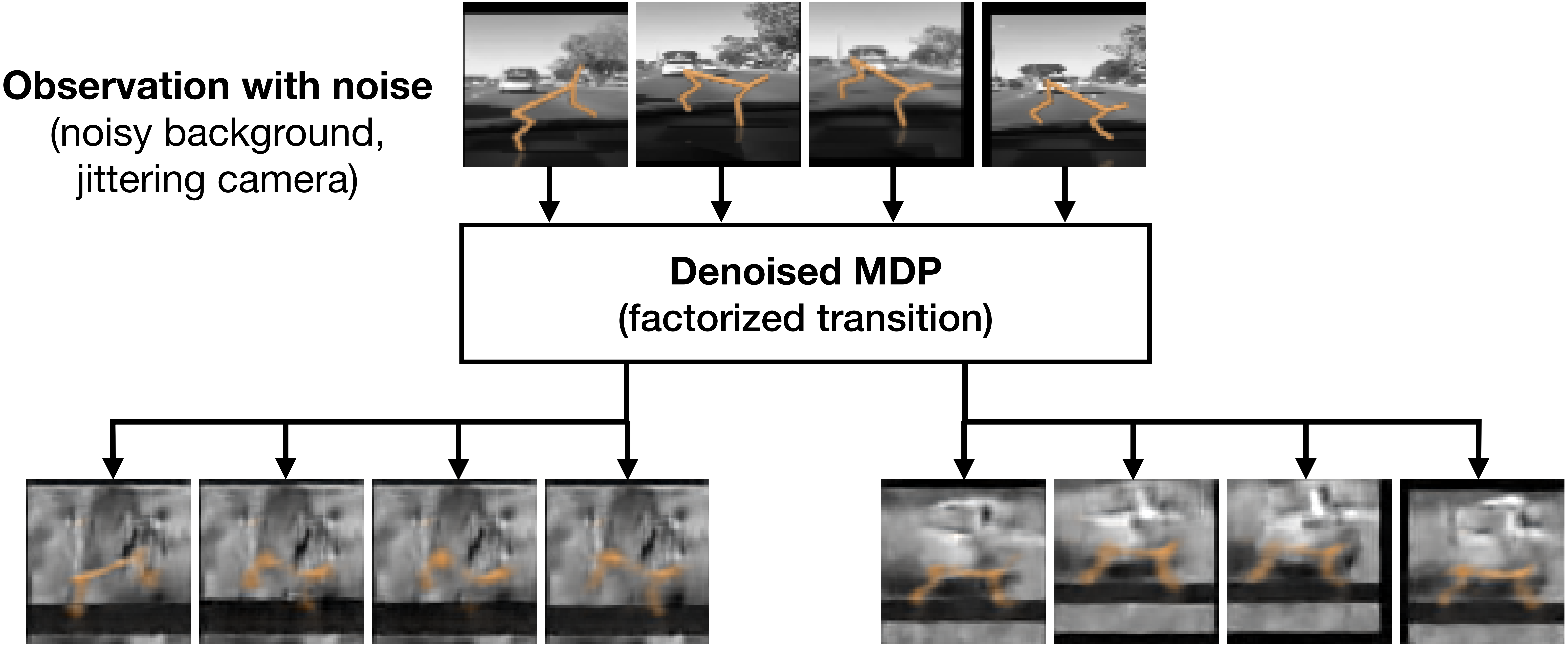
Observation with noise
(noisy background)

Denoised MDP
(factorized transition)

Learned Signal: Only agent moves

Learned Noise: Only background changes

Denoised MDP: signal-noise separation



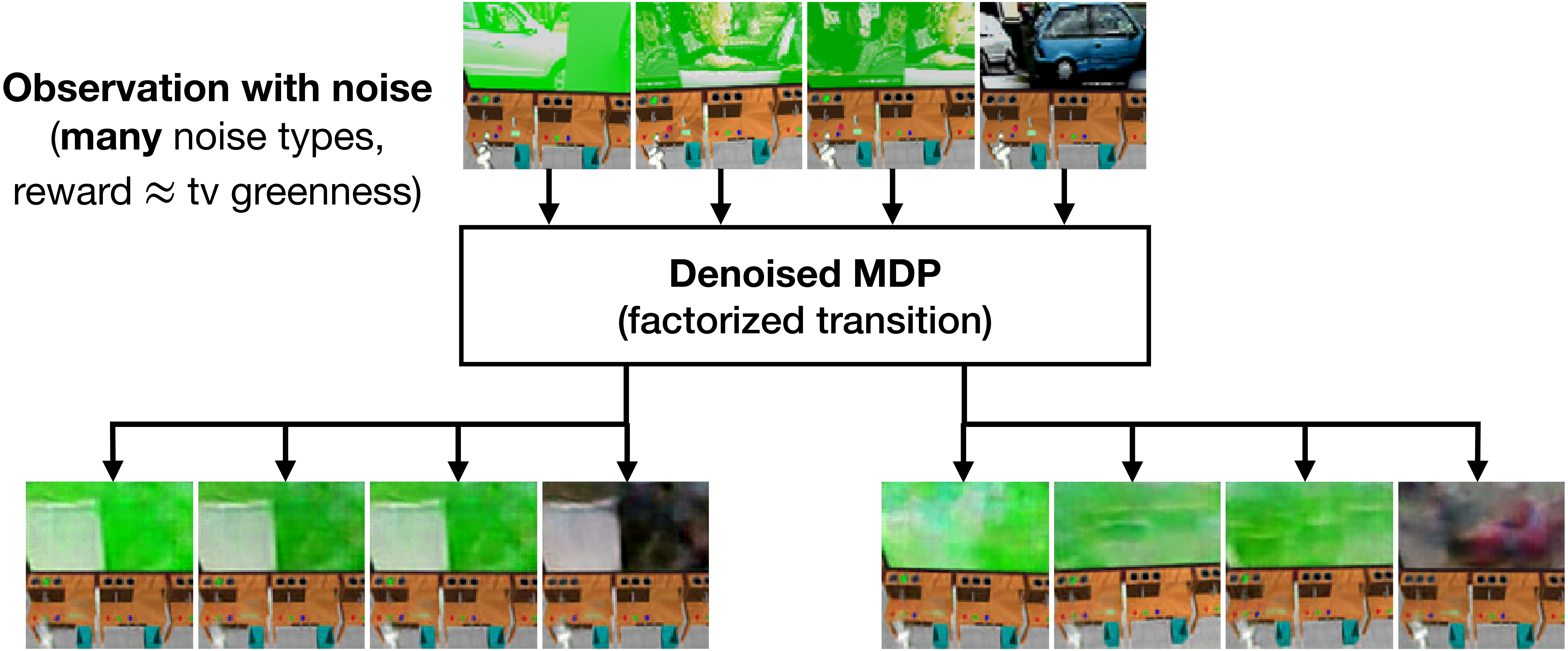
Observation with noise
(noisy background,
jittering camera)

Denoised MDP
(factorized transition)

Learned Signal: Only agent moves

Learned Noise: Camera & background move

Denoised MDP: signal-noise separation



Observation with noise
(**many** noise types,
reward ≈ tv greenness)

Denoised MDP
(factorized transition)

Learned Signal: Only agent & tv green hue

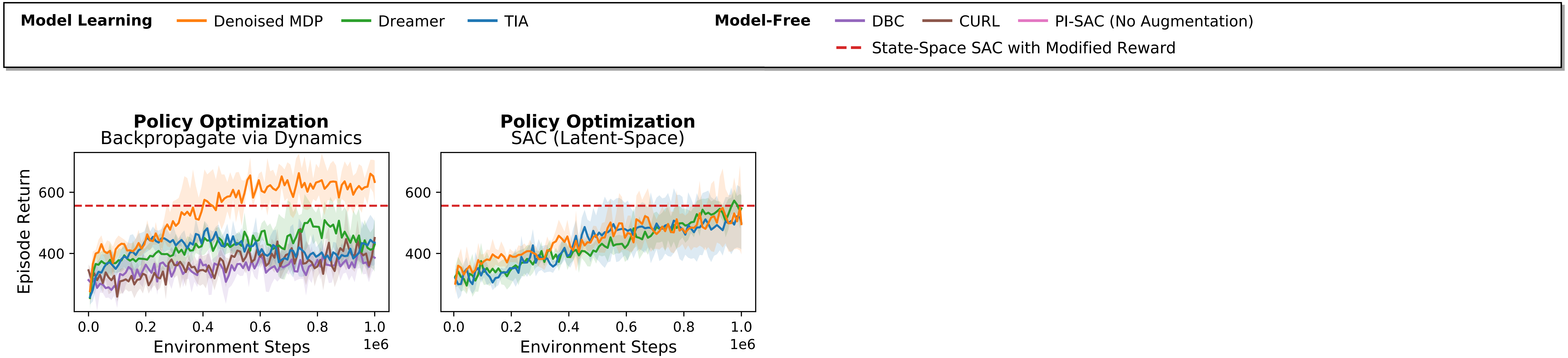
Learned Noise: All others

Better denoised model ⟹ Better policy

DeepMind Control Suite with distractors

Noise Variant	**Policy Learning:** Backprop via Dynamics			**Policy Learning:** SAC (Latent-Space)			DBC	PI-SAC (No Aug.)	CURL (Use Aug.)	State-Space SAC (Upper Bound)
	Denoised MDP	TIA	Dreamer	Denoised MDP	TIA	Dreamer				
Noiseless	801.4 ± 96.6	769.7 ± 97.1	**848.6 ± 137.1**	**587.1 ± 98.7**	480.2 ± 125.5	575.4 ± 146.2	297.4 ± 72.5	246.4 ± 56.6	417.3 ± 183.2	910.3 ± 28.2
Video Background	**597.7 ± 117.8**	407.1 ± 225.4	227.8 ± 102.7	309.8 ± 153.0	**318.1 ± 123.7**	188.7 ± 78.2	188.0 ± 67.4	131.7 ± 20.1	478.0 ± 113.5	910.3 ± 28.2
Video Background + Noisy Sensor	**563.1 ± 143.0**	261.2 ± 200.4	212.4 ± 89.7	**288.2 ± 123.4**	197.3 ± 124.2	218.2 ± 58.1	79.9 ± 36.0	152.5 ± 12.6	354.3 ± 119.9	919.8 ± 100.7
Video Background + Camera Jittering	**254.1 ± 114.2**	151.7 ± 160.5	98.6 ± 27.7	**186.8 ± 47.7**	126.5 ± 125.6	105.2 ± 33.8	68.0 ± 38.4	91.6 ± 7.6	**390.4 ± 64.9**	910.3 ± 28.2

RoboDesk with distractors

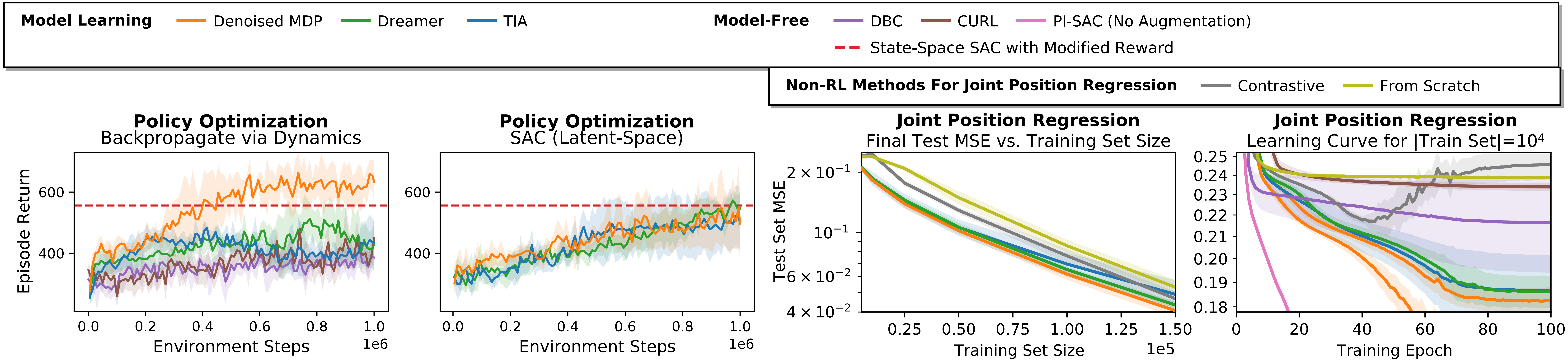


Better denoised model ⟹ Better policy & representation

DeepMind Control Suite with distractors

Noise Variant	**Policy Learning:** Backprop via Dynamics			**Policy Learning:** SAC (Latent-Space)			DBC	PI-SAC (No Aug.)	CURL (Use Aug.)	State-Space SAC (Upper Bound)
	Denoised MDP	TIA	Dreamer	Denoised MDP	TIA	Dreamer				
Noiseless	801.4 ± 96.6	769.7 ± 97.1	**848.6 ± 137.1**	**587.1 ± 98.7**	480.2 ± 125.5	575.4 ± 146.2	297.4 ± 72.5	246.4 ± 56.6	417.3 ± 183.2	910.3 ± 28.2
Video Background	**597.7 ± 117.8**	407.1 ± 225.4	227.8 ± 102.7	309.8 ± 153.0	**318.1 ± 123.7**	188.7 ± 78.2	188.0 ± 67.4	131.7 ± 20.1	478.0 ± 113.5	910.3 ± 28.2
Video Background + Noisy Sensor	**563.1 ± 143.0**	261.2 ± 200.4	212.4 ± 89.7	**288.2 ± 123.4**	197.3 ± 124.2	218.2 ± 58.1	79.9 ± 36.0	152.5 ± 12.6	354.3 ± 119.9	919.8 ± 100.7
Video Background + Camera Jittering	**254.1 ± 114.2**	151.7 ± 160.5	98.6 ± 27.7	**186.8 ± 47.7**	126.5 ± 125.6	105.2 ± 33.8	68.0 ± 38.4	91.6 ± 7.6	**390.4 ± 64.9**	910.3 ± 28.2

RoboDesk with distractors



Links & Poster

- **Poster: Today (7/20) 6:30-8:30pm. Hall E #803.**

  ✓ Clear **video visualizations**

  ✓ **Algorithm** & Information categorization details

  ✓ More **results**

- Project website: `ssnl.github.io/denoised_mdp/`

  ✓ Video visualizations

- Denoised MDP code: `github.com/facebookresearch/denoised_mdp`

  ✓ PyTorch implementation of Denoised MDP and Dreamer

- RoboDesk with Distractors code: `github.com/SsnL/robodesk`