World Model Guided Autonomous Learning in RL

What should the agent learn next?

Aditya Raj

6th November 2026

Roadmap

1. Model based RL: learning from a model of the world
2. World models: what they are and what they give us
3. Choosing experience: curiosity and exploration
4. My research problem: what is worth learning next?
5. Takeaways and questions

What I want you to leave with

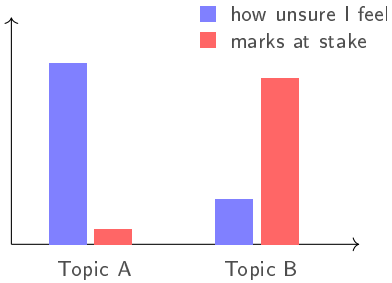
By the end, my research question should feel like the natural next step, not a sudden jump.

A warm up question

Say you have 10 hours left before an exam.

- Topic A: you feel very unsure about it, but it is worth only 2 marks
- Topic B: you feel only a little unsure, but it is worth 40 marks

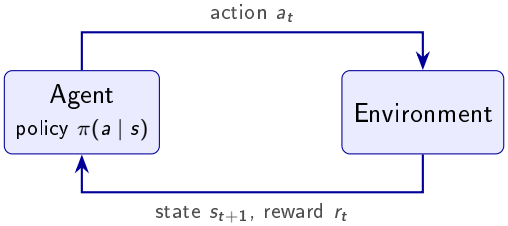
Where should the 10 hours go?



Same problem for an RL agent

Its budget is not hours, it is environment steps. Which experiences should it spend them on?

Reinforcement learning in one slide



- The agent acts, the environment responds
- It learns by trial and error to collect as much reward as possible

Example: a robot vacuum

State: where it is and what is around it
Action: move forward, turn left, turn right
Reward: plus 1 for a clean tile, a penalty for bumping a wall

The central cost

Real interactions are expensive, so sample efficiency decides what is practical.

Model free vs. model based

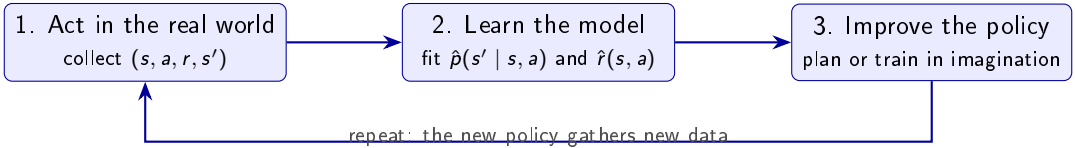
	Model free	**Model based**
Learns	a policy or value straight from experience	a model of how the world works, then a policy
Strength	simple, and performs very well in the long run	needs far fewer real interactions
Weakness	very hungry for data	a wrong model can mislead the policy

Example: learning a new city

Model free: drive around for weeks until you remember which turns work.
Model based: study the map once, then plan routes in your head before leaving home.

In model based RL the map is learned: $\hat{p}(s' \mid s, a)$ for what happens next and $\hat{r}(s, a)$ for the reward.

The model based RL loop



Example: a robot and a door

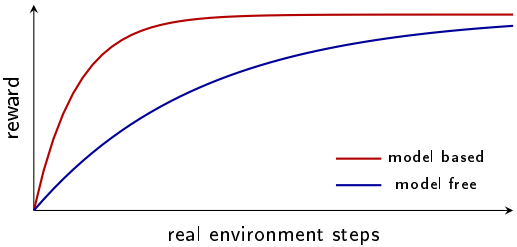
1. The robot pushes a door a few times. **2.** Its model learns that pushing opens it. **3.** It plans the whole route to the goal in its head, without walking it.

- Only step 1 touches the real world, so what step 1 collects shapes everything after it

Why it helps

- One real step can feed many practice steps inside the model
- You can plan ahead before you act
- The same model can be reused for other tasks

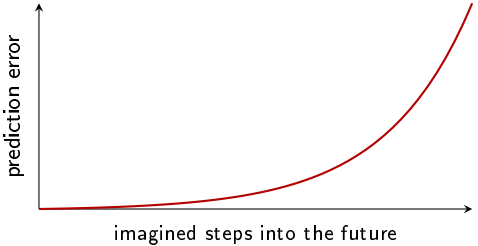
Think of a pilot in a flight simulator. Hours in the simulator save hours in a real plane.



Illustrative shape of the idea, not real data.

The catch: model errors grow

- The model is learned, so it is wrong somewhere
- In a long imagined rollout, each small mistake feeds the next prediction
- The policy may even exploit a mistake



Illustrative: errors compound with the horizon.

Two everyday pictures

Telephone game: a small slip early on turns into nonsense by the end of the line.
Wrong map: if the model thinks a wall is open, the agent plans a route straight through it.

So

A good agent must know *where its model can be trusted*.

Two kinds of not knowing

	Aleatoric	**Epistemic**
Meaning	randomness in the world itself	the model simply has not learned it yet
More data?	does not help	fixes it
Example	rolling a fair die: no history tells you the next number	a door you have never tried: push it twice and you know

Why this matters for exploration

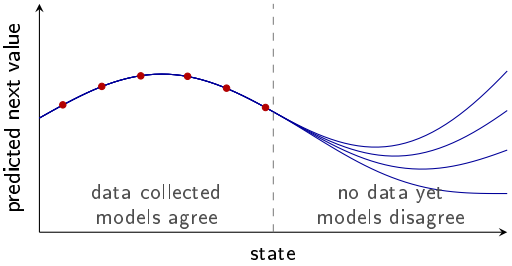
Epistemic uncertainty is the kind an agent can actually *do something about* by collecting the right experience.

How an ensemble shows what it does not know

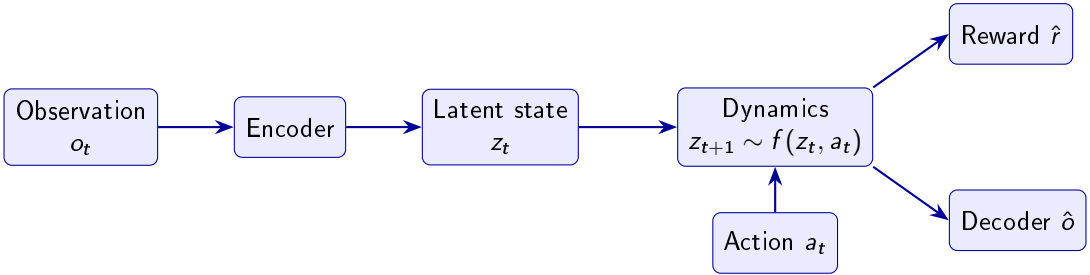
- Train K copies of the dynamics model, $f_1, \ldots, f_K$, on the same data
- Where data exists, they all agree
- Where it does not, each one guesses differently

$$\text{Disagreement}(s, a) = \text{Var}_k\left[f_k(s, a)\right]$$

Like asking five friends for directions. If they all say left, you trust it. If they all say different things, nobody really knows the road.



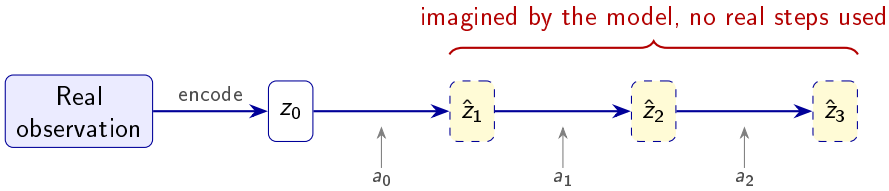
What is a world model?



Example: a glass near the edge of a table

You can picture the glass falling and breaking without ever pushing it. That picture in your head is a world model: it takes what you see and an action, and predicts what happens next.

Learning in imagination



- A world model summarises past experience into a representation that can predict imagined futures (Ha and Schmidhuber, 2018, among others)
- Latent world models predict ahead in a compact latent space, and have solved control tasks from images (Hafner et al., 2019)
- The agent can then plan and learn using only imagined predictions, as Plan2Explore does

A world model is more than a predictor

As a predictor

- "If I do a in s, what happens?"
- Supports planning and imagination

As a self check

- "Which of my predictions am I unsure about?"
- Supports choosing what to learn next

Example

A student who can predict exam questions is good. A student who also knows *which chapters they would fail* knows where to study. The second skill is what I care about.

The shift

The model does not only predict the environment, it can help decide *which experiences the agent should seek out*.

Exploration: the agent shapes its own data

- The agent only learns from experience it collects itself
- Simple tricks like epsilon greedy or random noise ignore what the agent already knows about its own gaps

Example: picking a restaurant

Random: close your eyes and point.
Curious: try the place you know least about, because that is where you learn the most.

Curiosity in RL

Reward the agent for reaching situations where its model is uncertain, or where it is about to learn something.

Prior work: reduce uncertainty about the dynamics

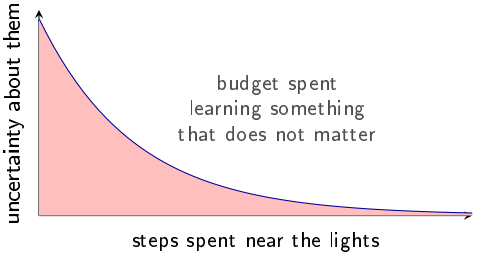
Method	What counts as worth exploring	Key idea
VIME Houthooft et al., 2016	information gained about the dynamics model	a Bayesian neural network models the dynamics, and a transition earns a bonus when it changes the model's belief a lot
Plan2Explore Sekar et al., 2020	disagreement of an ensemble of one step models	plans inside a learned world model to find places that will be novel in the future, with no task reward
OPAX Sukhija et al., 2023	epistemic uncertainty of calibrated probabilistic models	optimistic active exploration with theory behind it, and planning for new tasks afterwards

The worry that motivates my work

Example: the flickering lights

A wall of lights blinks in a fixed pattern that has nothing to do with the task. At first the agent cannot predict it, so it keeps coming back. A while later it has learned the pattern and loses interest. By then, much of the budget is gone.

The uncertainty is real and learnable. The question is whether learning it helps the agent do its job.



Illustrative.

What these methods share, and what they leave open

- Their signal is about the dynamics: information gained, ensemble disagreement, or epistemic uncertainty
- In short, they ask where is my model of the dynamics uncertain?
- None of these signals asks which information matters for the agent's future decisions

The open question

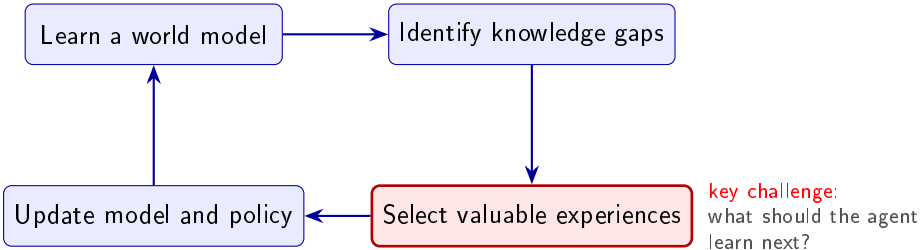
Should an agent choose experiences only when its model is uncertain, or based on what information will be useful for improving its decisions?

Research question

How can an RL agent use a learned world model to identify what it does not know, select informative experiences, and continually improve its decision making?

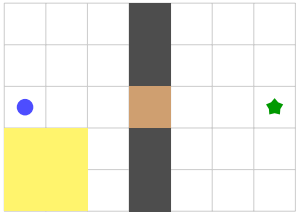
- Not just "predict the environment well"
- But "decide what to learn next, and why"

The autonomous learning loop



Every box is a known tool. The red box is the part I want to study.

A concrete example I will keep using



Blue dot: the agent
Green star: the goal, which gives the reward
Black wall with a brown door: the agent must get through the door
Yellow tiles: flicker in a learnable pattern, and have nothing to do with the goal

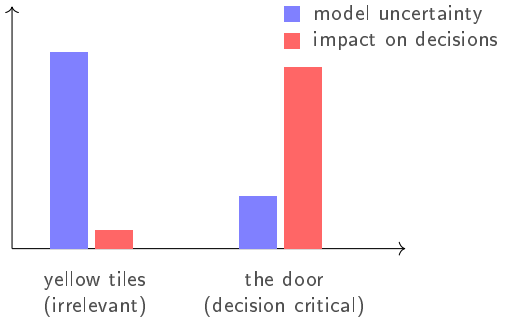
What the agent is unsure about

It is a bit unsure whether the door opens when pushed. It is very unsure what the yellow tiles will do next.

This grid is my own picture to explain the idea. It is not one of the benchmark tasks.

The problem: not all uncertainty matters equally

- Some uncertain parts are irrelevant to the task, like the yellow tiles
- Some barely uncertain parts are decision critical, like the door
- A small error at the door can change the whole behaviour



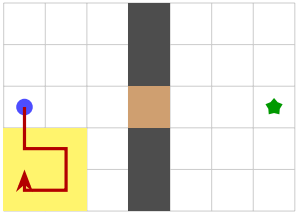
Illustrative sketch of the hypothesis, not a result.

Same as the exam

An agent that only chases uncertainty may spend a large part of its budget on the left bar.

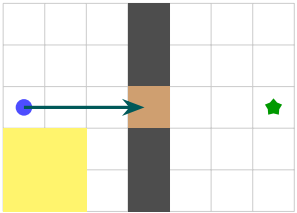
Two ways to choose experience

Dynamics directed



Heads for the yellow tiles, where uncertainty is highest.

Decision directed



Heads for the door, because that information changes what it will do.

Same interaction budget for both

Only *what gets collected* differs, so any difference in results comes from that choice.

The two strategies side by side

	Dynamics directed	**Decision directed**
Asks	"Where is my model uncertain?"	"Which information changes my decisions?"
Collects	experience that reduces uncertainty about how the environment behaves	experience whose information matters for future decisions
Signal	uncertainty of the ensemble dynamics model	how relevant the information is to future decisions
Concern	may spend budget on dynamics that are learnable but irrelevant to reward	needs a way to judge what is relevant

What "decision directed" needs

Decision directed, in my notes

Prioritise experiences whose information is more relevant to future decisions.

- To do this, the agent needs some way to tell which information is relevant
- Doing that well is part of the project, so I treat it as something to work out and test
- Whatever the definition, I compare it against the dynamics directed strategy under the same interaction budget

In the grid example, the door is the decision relevant part and the yellow tiles are not.

Experimental setup

Agent

- Model based RL
- An ensemble based learned dynamics model
- Same interaction budget for every strategy

Environments

- Mostly existing open source benchmarks, not environments built from scratch
- First: simple **MiniGrid** tasks
- Then: selected **DM Control** tasks for continuous control

The controlled twist

- Add controlled dynamics that are learnable but irrelevant to reward, like the yellow tiles
- Vary how much of this the agent meets
- The task and the evaluation stay the same

Why this design

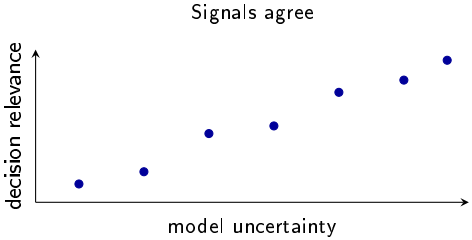
I can see exactly when uncertainty and relevance split apart, without building new environments from scratch.

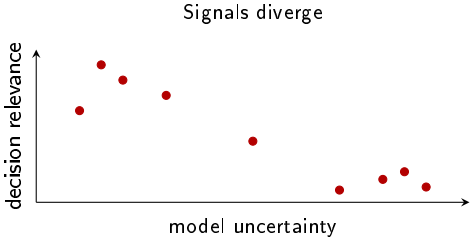
Goals

1. Measure when model uncertainty and decision relevance diverge
2. Identify what properties of an environment cause the divergence
3. Test whether decision directed experience selection improves sample efficiency
4. Understand why a strategy succeeds or fails, not only compare final rewards

What I will measure

1. **Where** each strategy collects its experience
2. **How the information signals align:** model uncertainty versus decision relevance
3. **Whether one strategy** leads to faster improvement in task performance

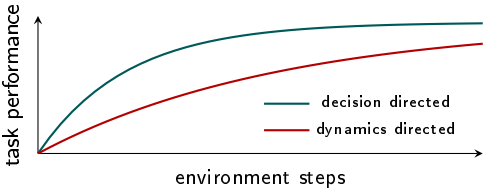




Hypothetical pictures. Each dot is one state or action. The plan is to measure how close the real data is to each.

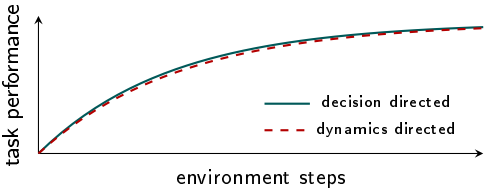
What success looks like

A positive result



Learning more about the world is not always the best use of the budget.

A negative result is useful too



General uncertainty already points to what matters. That tells us where this idea is not worth it.

Both plots are hypothetical shapes, not data.

Longer term vision

- If the first results look good, I want to move toward agents that decide for themselves what they need to learn next
- Today, agents mostly rely on whatever experience they happen to run into
- The hope is a world model that does not only predict the environment, but also points the agent to the experiences that best improve its decisions

Back to the exam

The goal is an agent that studies like a smart student: not everything, and not just what feels confusing, but what will actually move the score.

Takeaways

1. Model based RL practises inside a learned model, so real interactions go further
2. A world model can also tell the agent what it does not know
3. The exploration methods I looked at (VIME, Plan2Explore, OPAX) target uncertainty about the dynamics
4. My question: should the agent go after what improves its decisions instead?
5. Either answer is useful: a real gain, or a clear limit of the idea

Questions?

Aditya Raj

References

- Houthooft et al. **VIME: Variational Information Maximizing Exploration**. NeurIPS 2016. arXiv:1605.09674
- Sekar et al. **Planning to Explore via Self Supervised World Models**. ICML 2020. arXiv:2005.05960
- Sukhija et al. **Optimistic Active Exploration of Dynamical Systems**. NeurIPS 2023. arXiv:2306.12371
- Ha and Schmidhuber. **World Models**. 2018.
- Hafner et al. **Learning Latent Dynamics for Planning from Pixels**. 2019.
- More from my notes: arXiv:2010.14641, arXiv:2506.06725