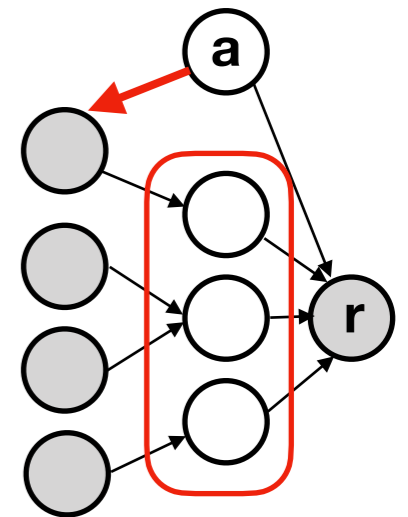
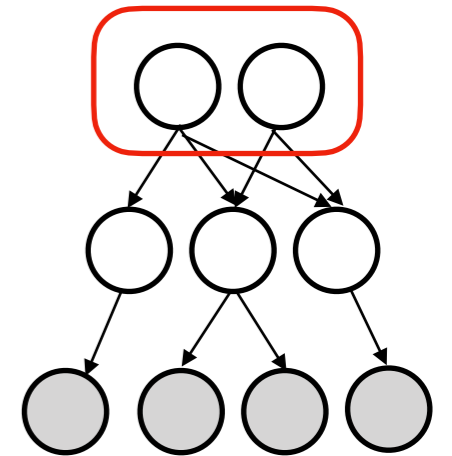


# Interplay of beliefs and representations in reinforcement learning

Mihály BÁnyai  
AGPD Lab meeting  
30.11.2022

# Task-dependent representations

- Representation: efficient compression
  - No resources to compute with all information - throw away some of it
  - May be used to solve so many different tasks that generic compression is best (c.f. Barlow)
- But tasks matter a great deal in what “efficient” compression means.
  - embed the compression process into a reinforcement learning agent
  - In RL, representations are a means to an end instead of the main goal
  - RL makes it explicit that one of the most crucial scarce resources is time
    - Efficiency of compression means sample efficiency of policy learning
    - abstractions ought to generalise well in terms of which action to take
- How we compress our sensory space is how we form beliefs about what objects and relations are there in it
  - Beliefs and representations are necessarily intertwined

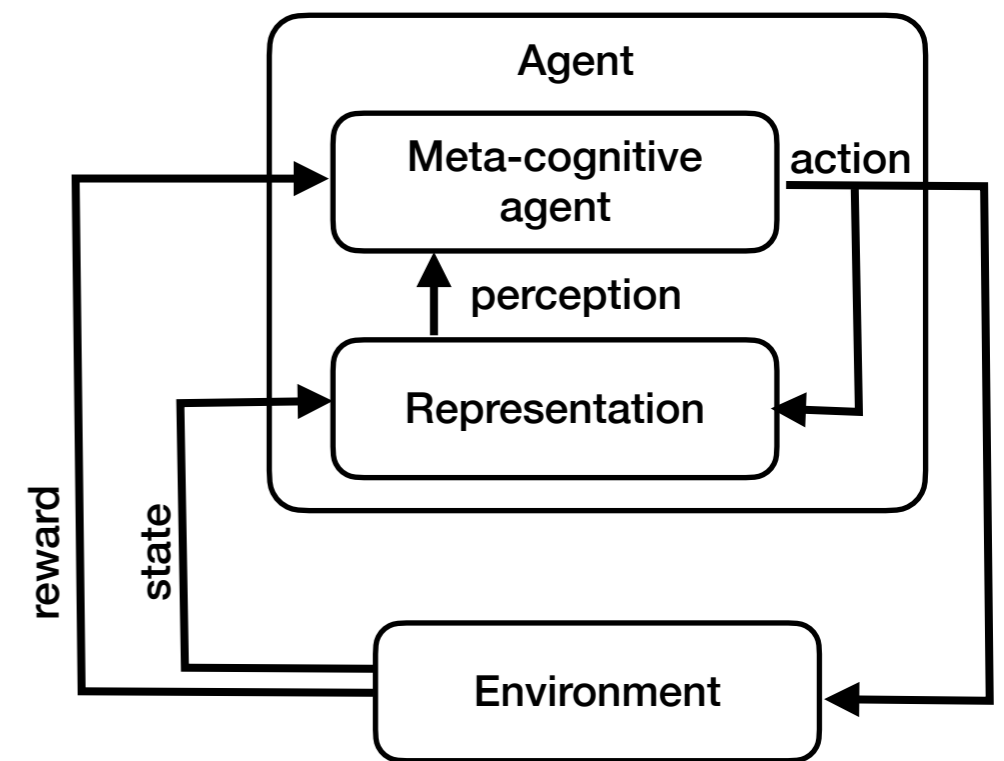


# Frameworks for parallel learning of policies and representations

- Control-as-inference
  - Calibrating approximate Bayesian inference such that it maximises an actively learned utility function
  - Assumes that the agent wants to be able to predict the statistics of observed stimuli on top of maximising reward - this is an ad-hoc modification of the RL objective
  - Resource constraints are by definition entropic - no notion of task difficulty being dependent on agent architecture
- The information bottleneck
  - Singles out reward as the distortion metric of your compression
  - More flexibility in defining the resource constraints as dynamic update rules
  - Not defined for problems where the model of the environment is unknown
- Massive issue with both: myopia
  - Instead of directed exploration in behaviour, you end up with a Boltzmann policy
  - Representational updates don't take into account what further updates you'll be able to do after them
  - Myopia drastically reduces sample efficiency

# The normative, meta-cognitive approach

- start from the basic decision making problem and look at what would be the optimal solution to it
- The agent makes 2 kinds of decisions:
  - What to do in the environment
  - How to modify its own representation
  - This ends up being an MDP
- The fully optimal solution to learning in MDPs is exploration via planning
  - In bandits, this is given by Gittins indices
  - Similar efforts in MDPs, e.g. Michael Duff's thesis (BAMDPs)
  - But it remains an intractable problem in its full generality
- Question:
  - can we give formalise the fully optimal solution,
  - and then give an approximation to it that is tractable for toy problems,
  - but remains interesting from the normative point of view?

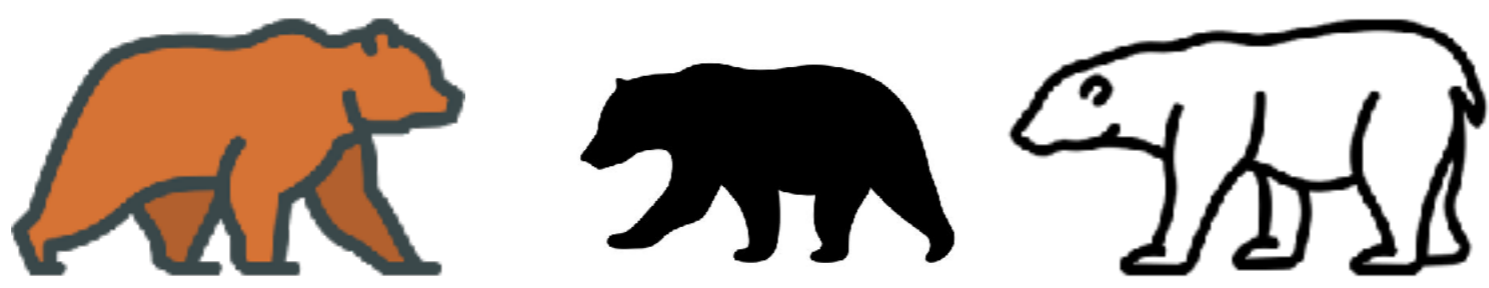


# A toy problem

Abstractions



Stimuli








Actions



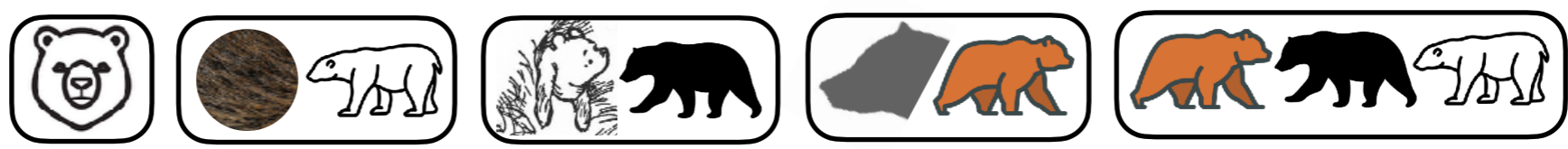
Abstractions



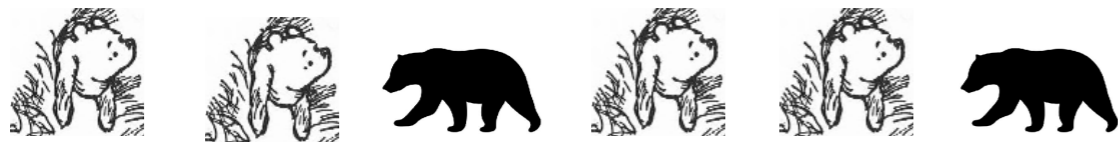
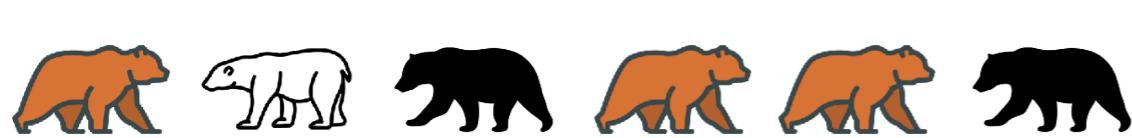
Rewards

		
	0	1
	1	0
	0	1

Possible representations

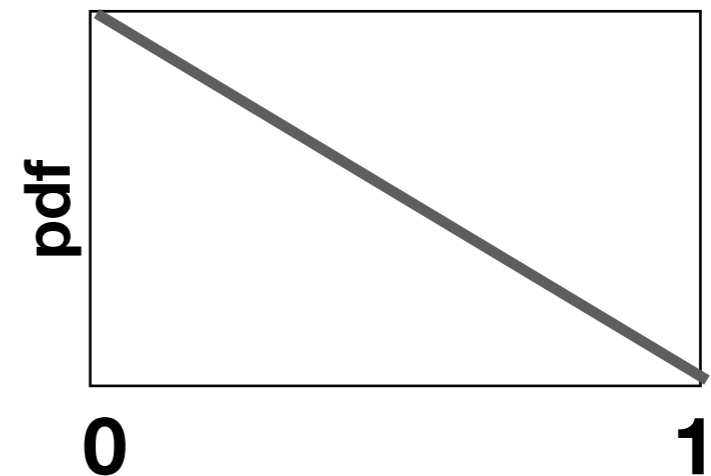


# What is a belief?

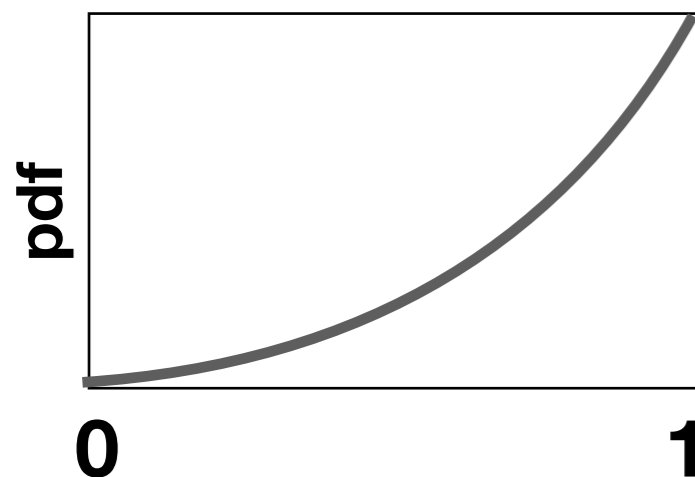


$r = 0 \quad 1 \quad 1 \quad 1 \quad 1 \quad 0$

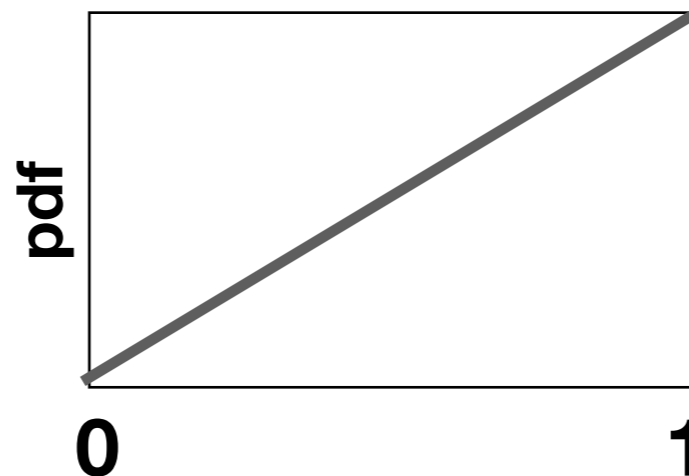
$$p(r=1 \mid \text{monkey}, \text{standing}) \\ = E[\text{Beta}(1, 2)] = 1/3$$



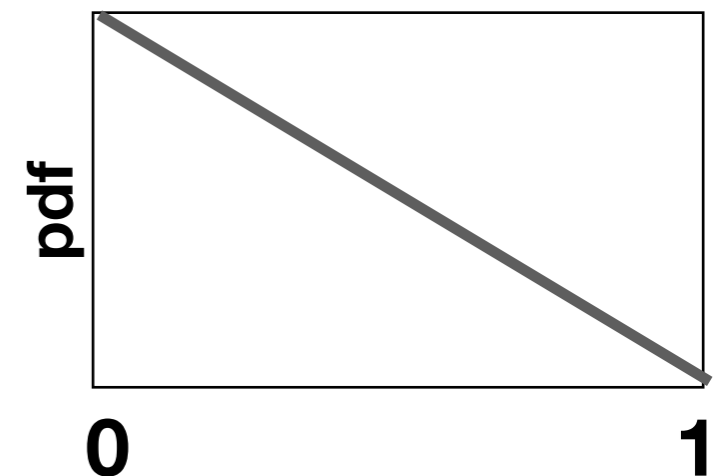
$$p(r=1 \mid \text{monkey}, \text{lying down}) \\ = E[\text{Beta}(4, 1)]$$



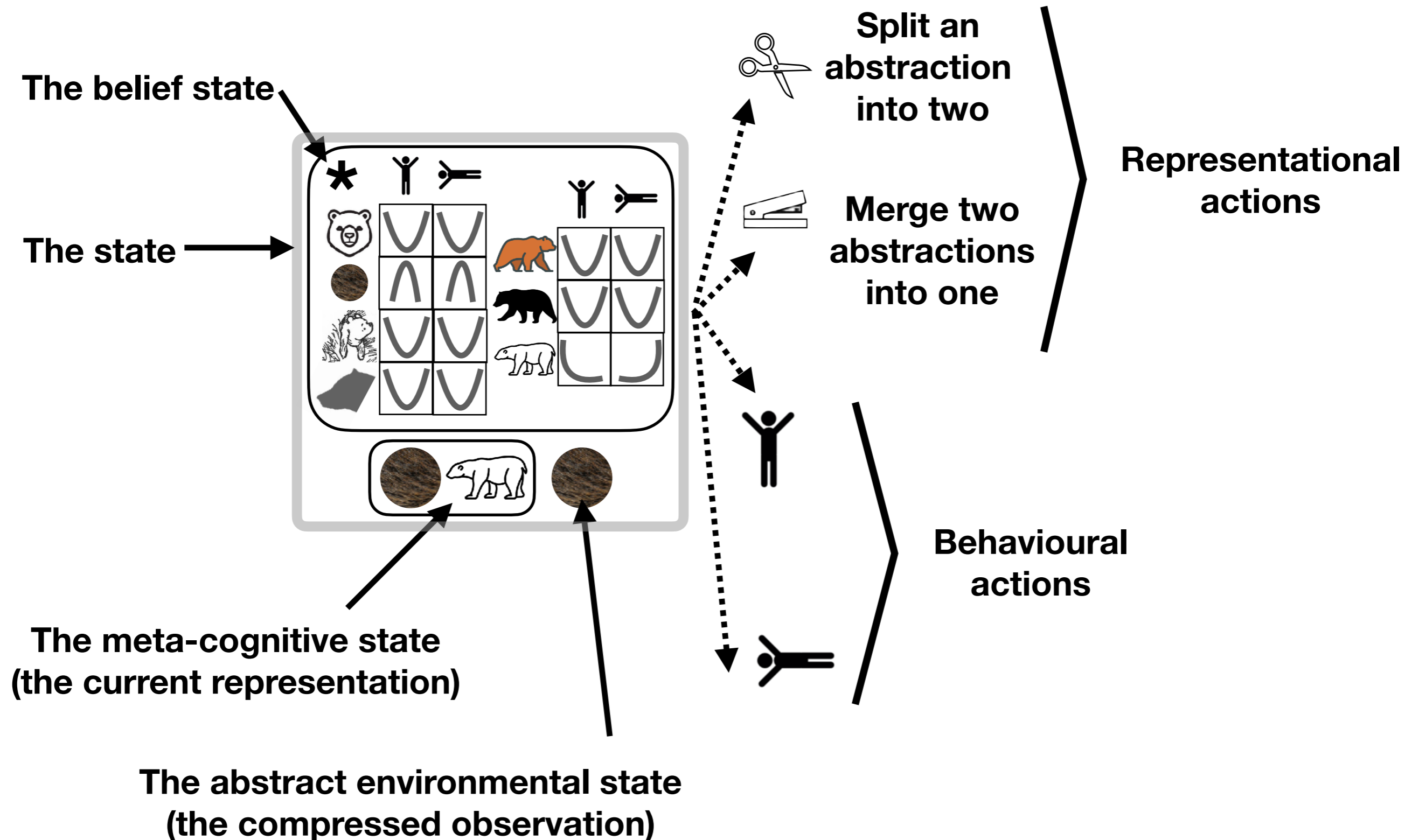
$$p(r=1 \mid \text{black bear}, \text{standing}) \\ = E[\text{Beta}(2, 1)]$$



$$p(r=1 \mid \text{black bear}, \text{lying down}) \\ = E[\text{Beta}(1, 2)]$$

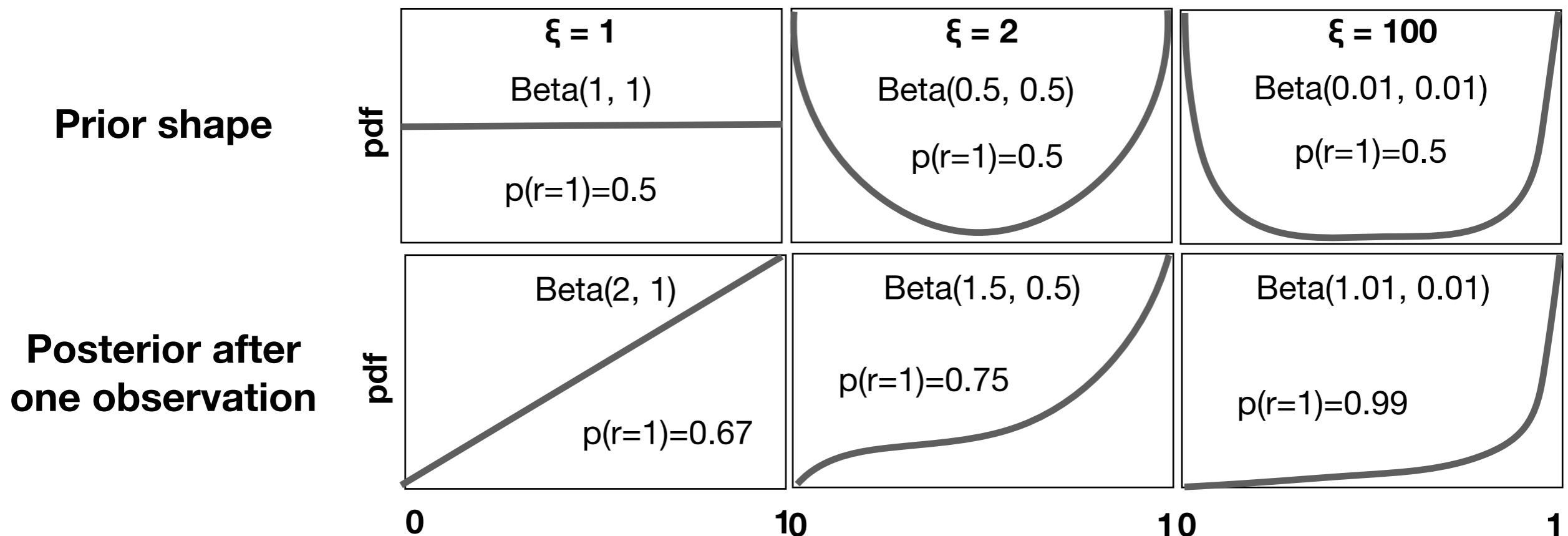


# The meta-cognitive MDP for representation learning



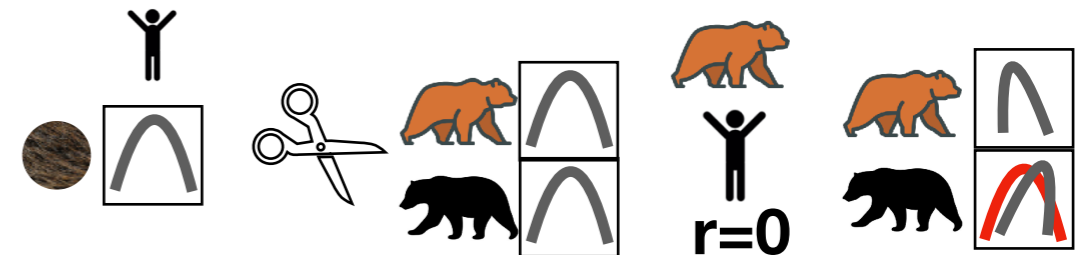
# Priors

- Assumptions about important properties of the environment
  - How independent are beliefs about different actions in the same state?
    - They might be e.g. anticorrelated or sum up to 1
  - Volatility of the environment (including how long the agent can interact with it) -  $\gamma$
  - How independent are the beliefs about different abstractions
    - Trade-off between tractability and exactness
  - Stochasticity / aleatoric uncertainty in the environment -  $\xi$



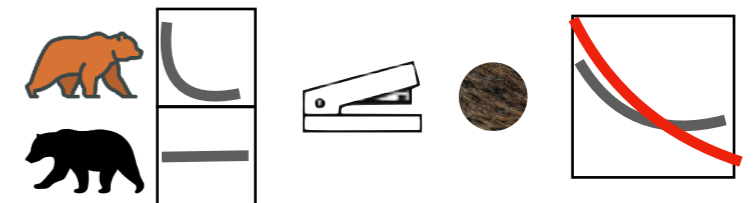
# Constrained beliefs

- Some constraint already applied: beliefs independent across abstractions, always from Beta family



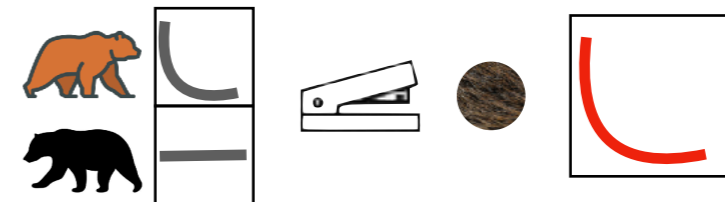
- Representational actions in general break these

- Split: creates an anti correlation between resulting abstractions



- Merge: creates a mixture-of-Betas posterior

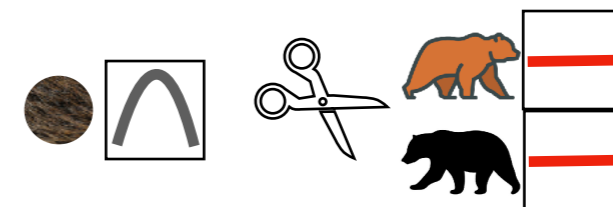
- To which the agent can match a Beta



- Even simpler updates

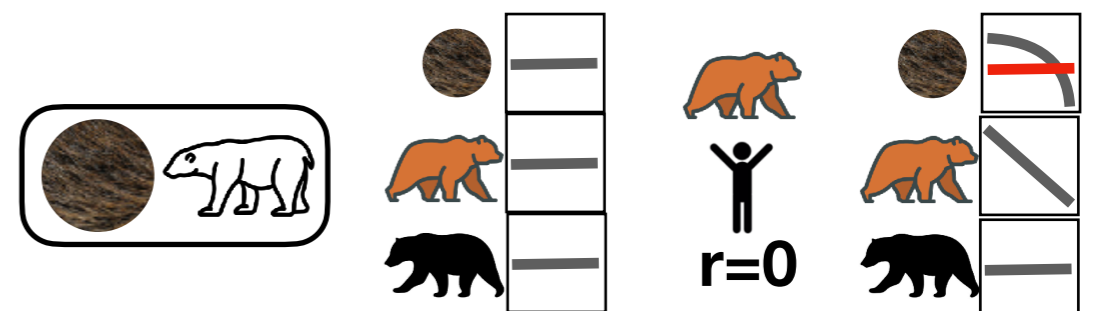
- Pooling upon merge

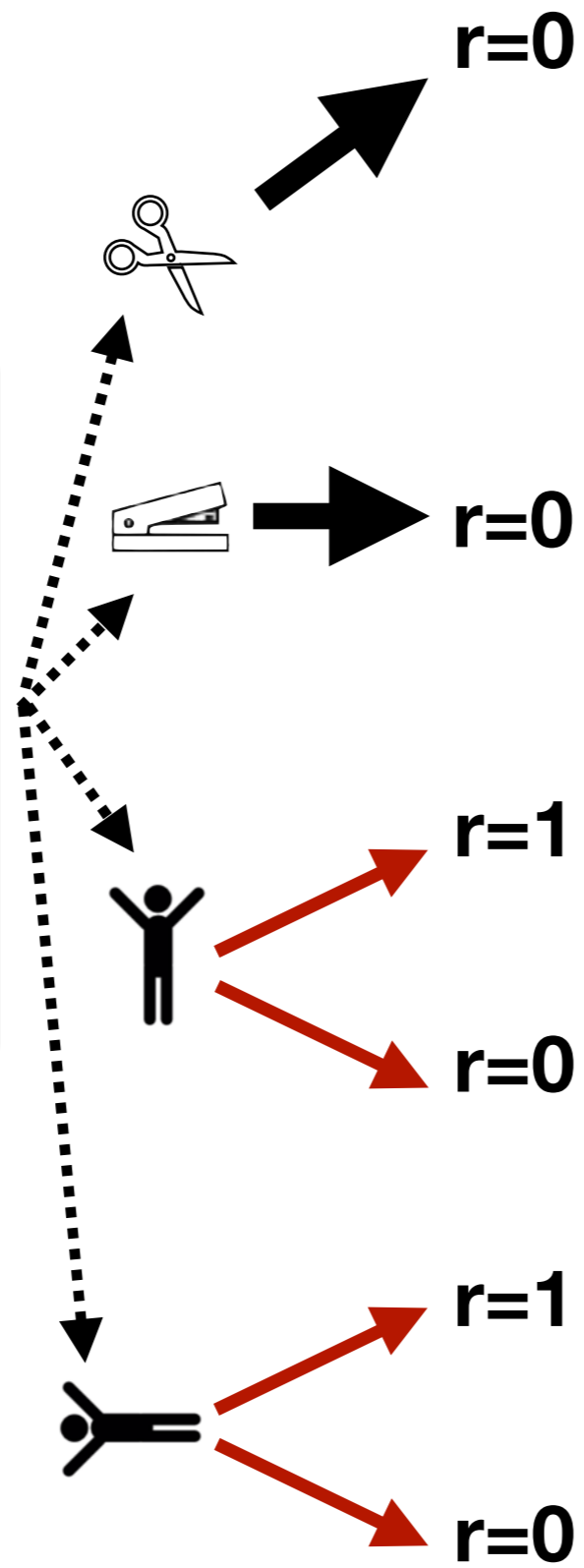
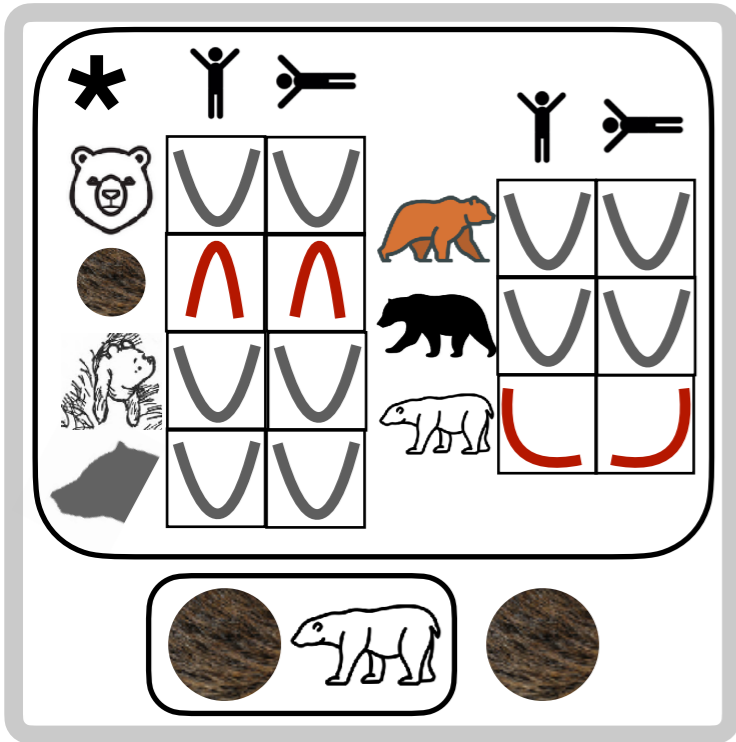
- Deleting upon split

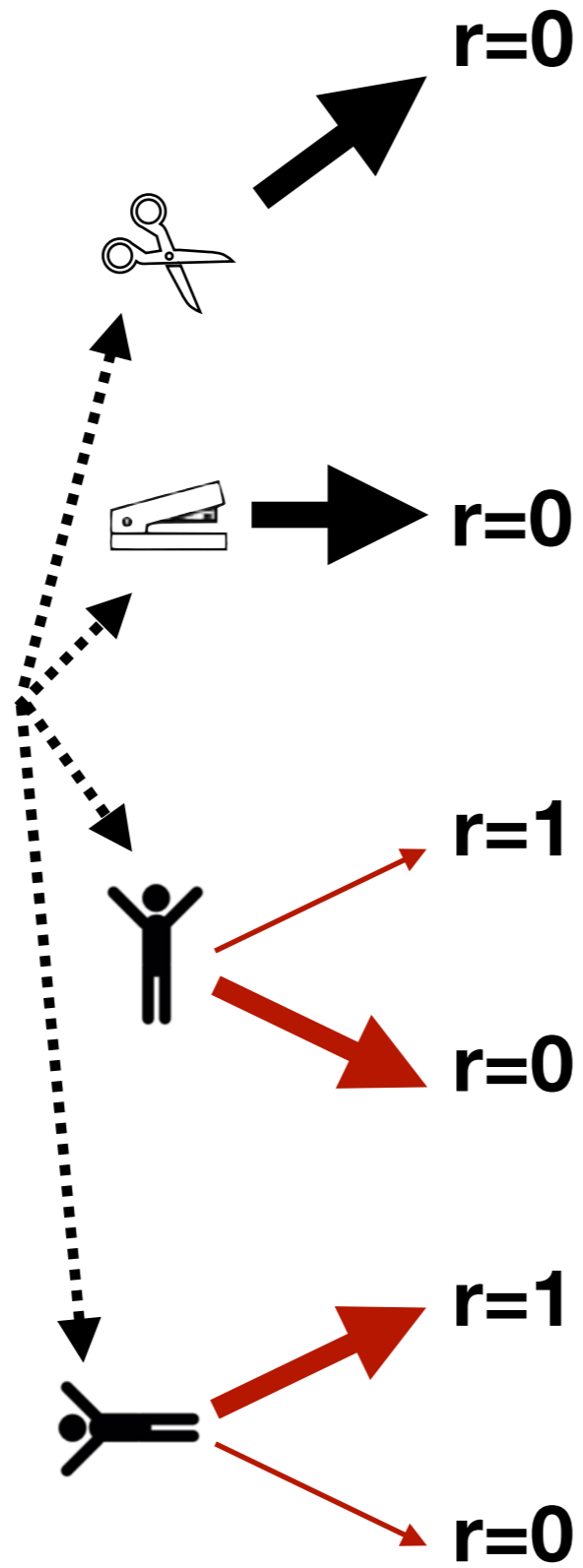
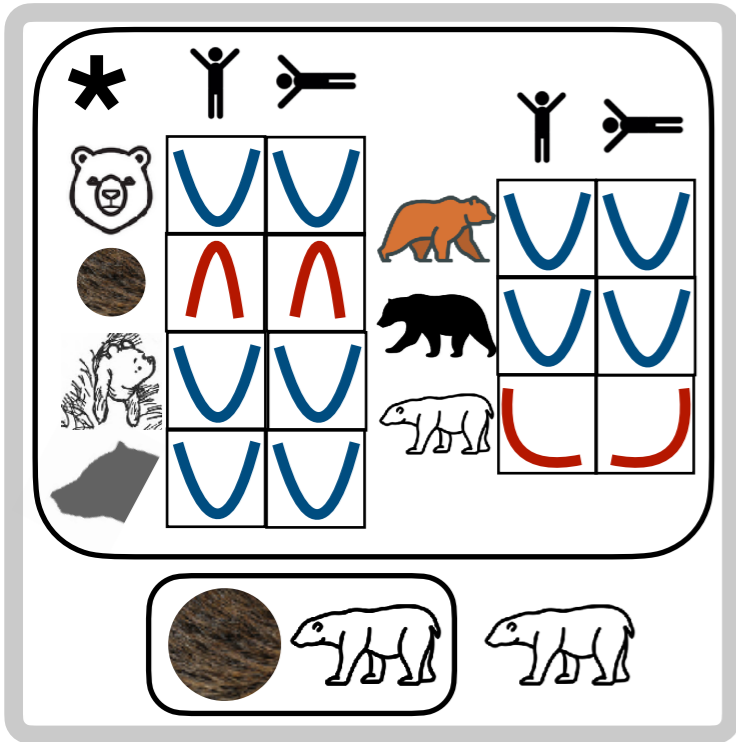


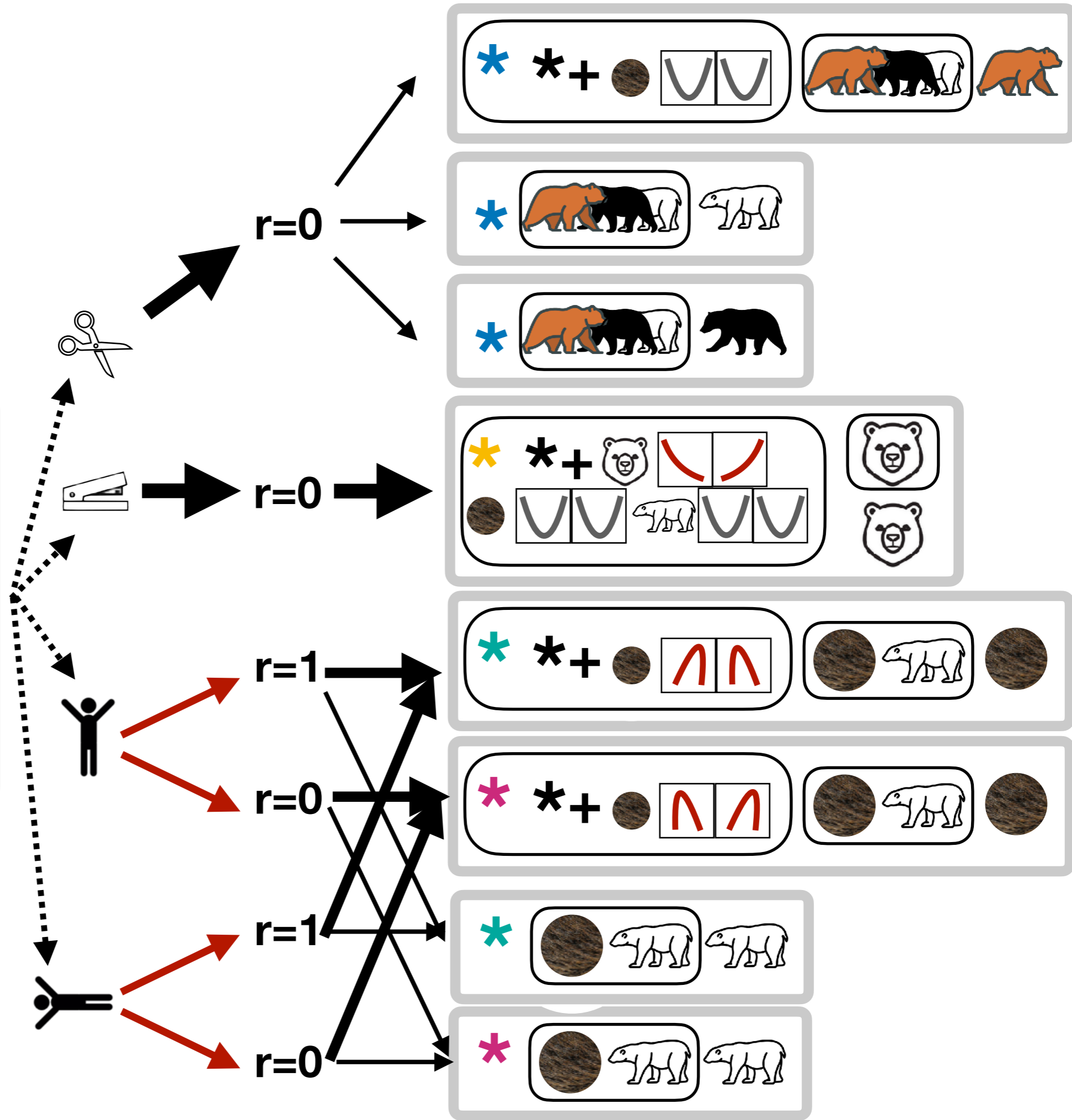
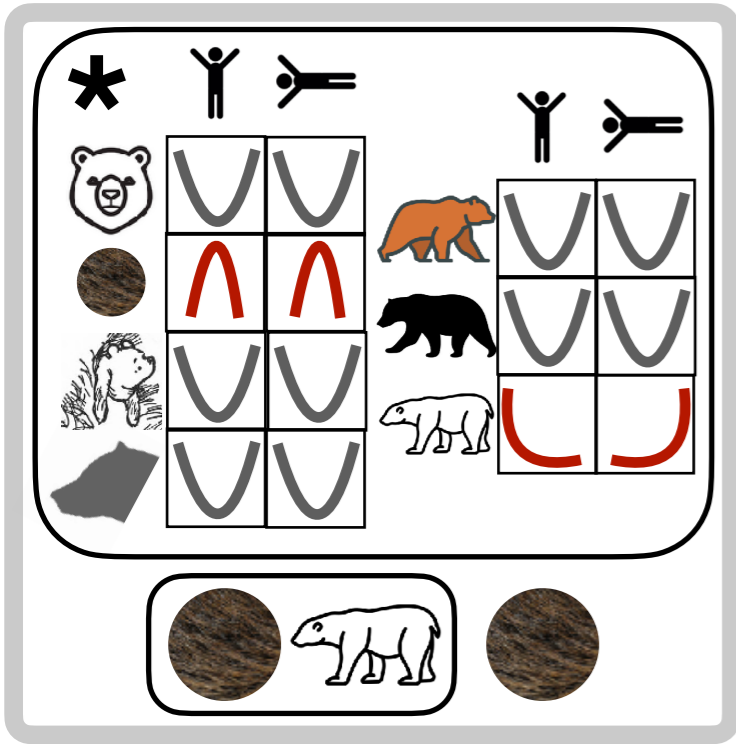
- Non-counterfactual beliefs

- Necessary to motivate generalisation through merging abstractions

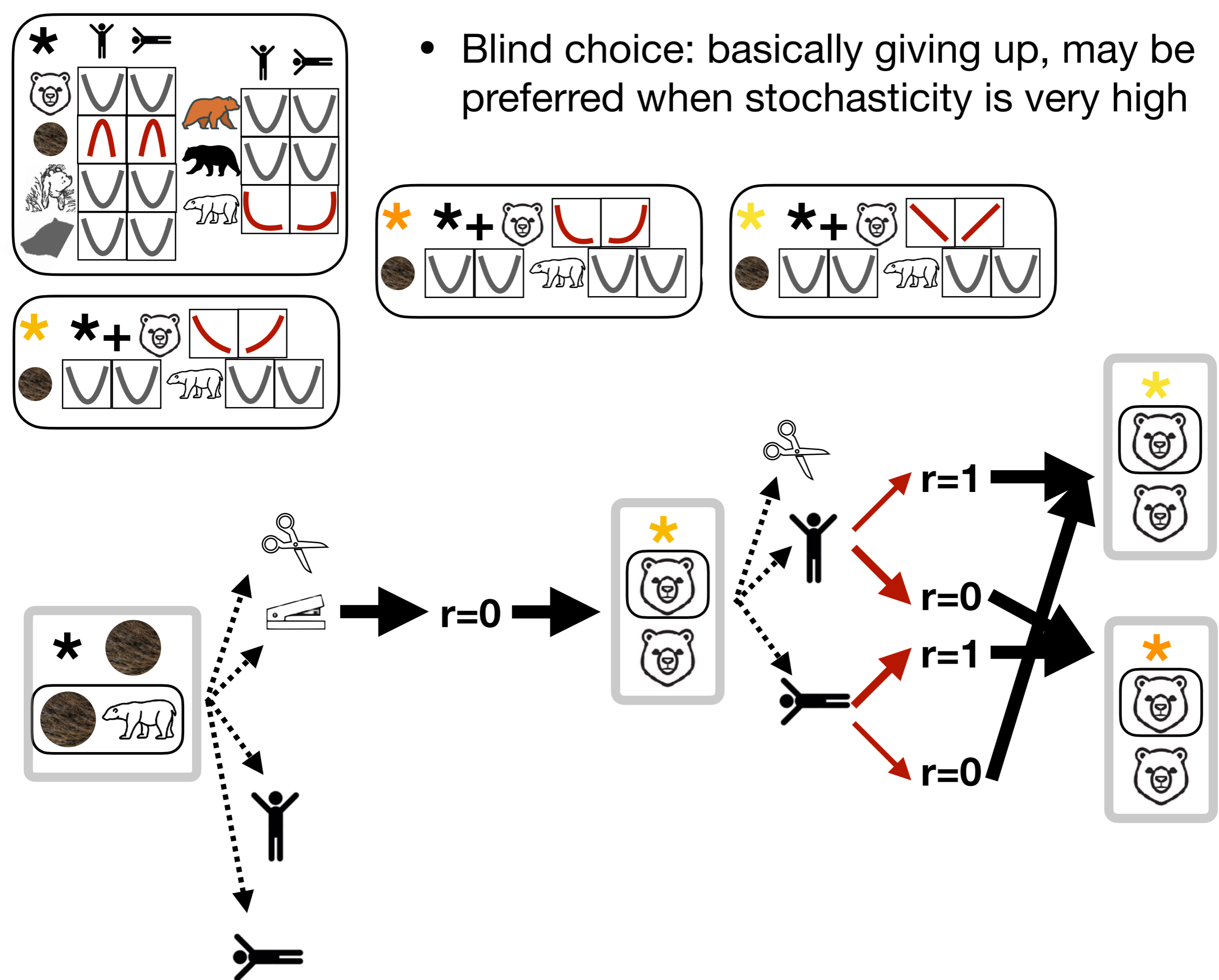




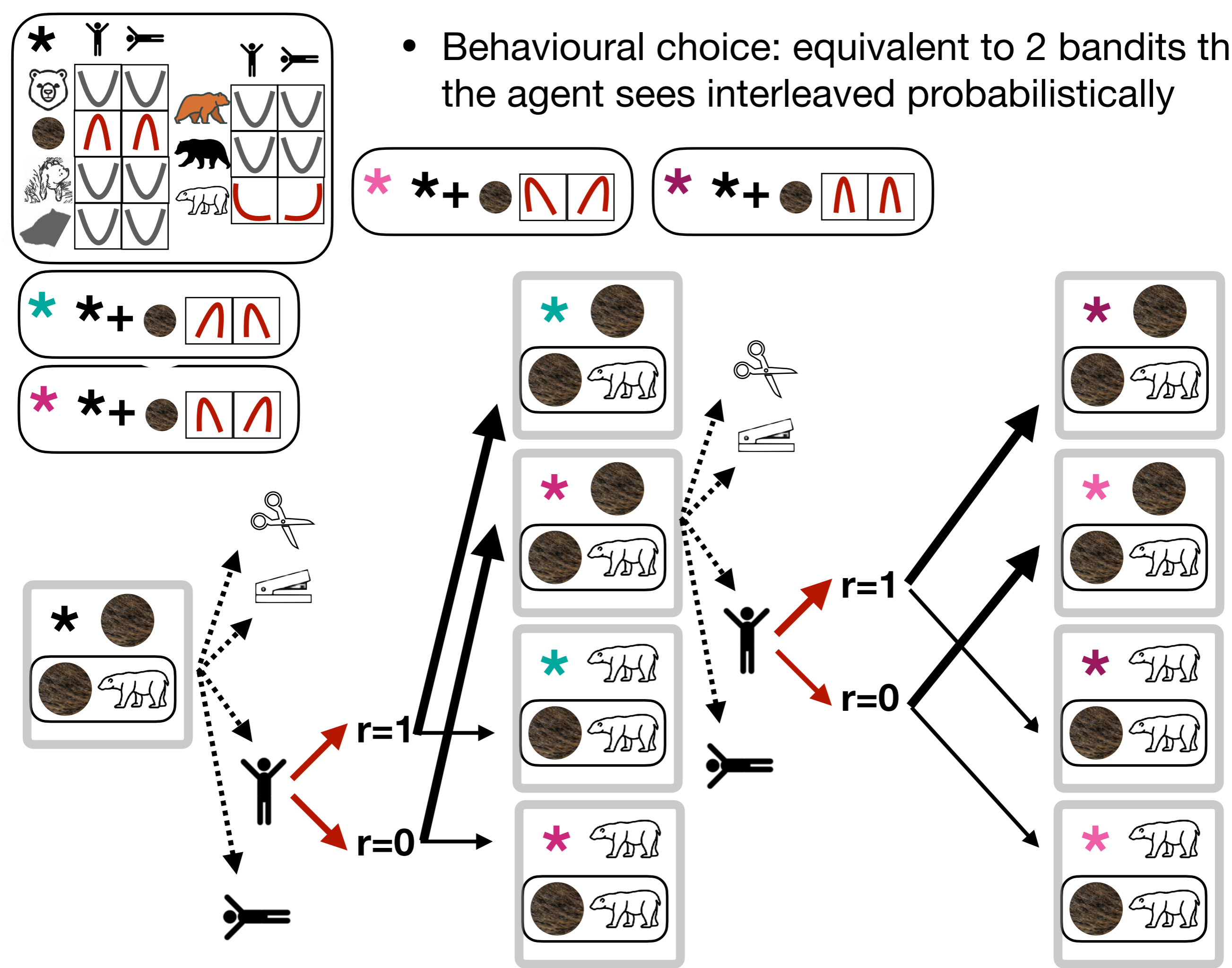


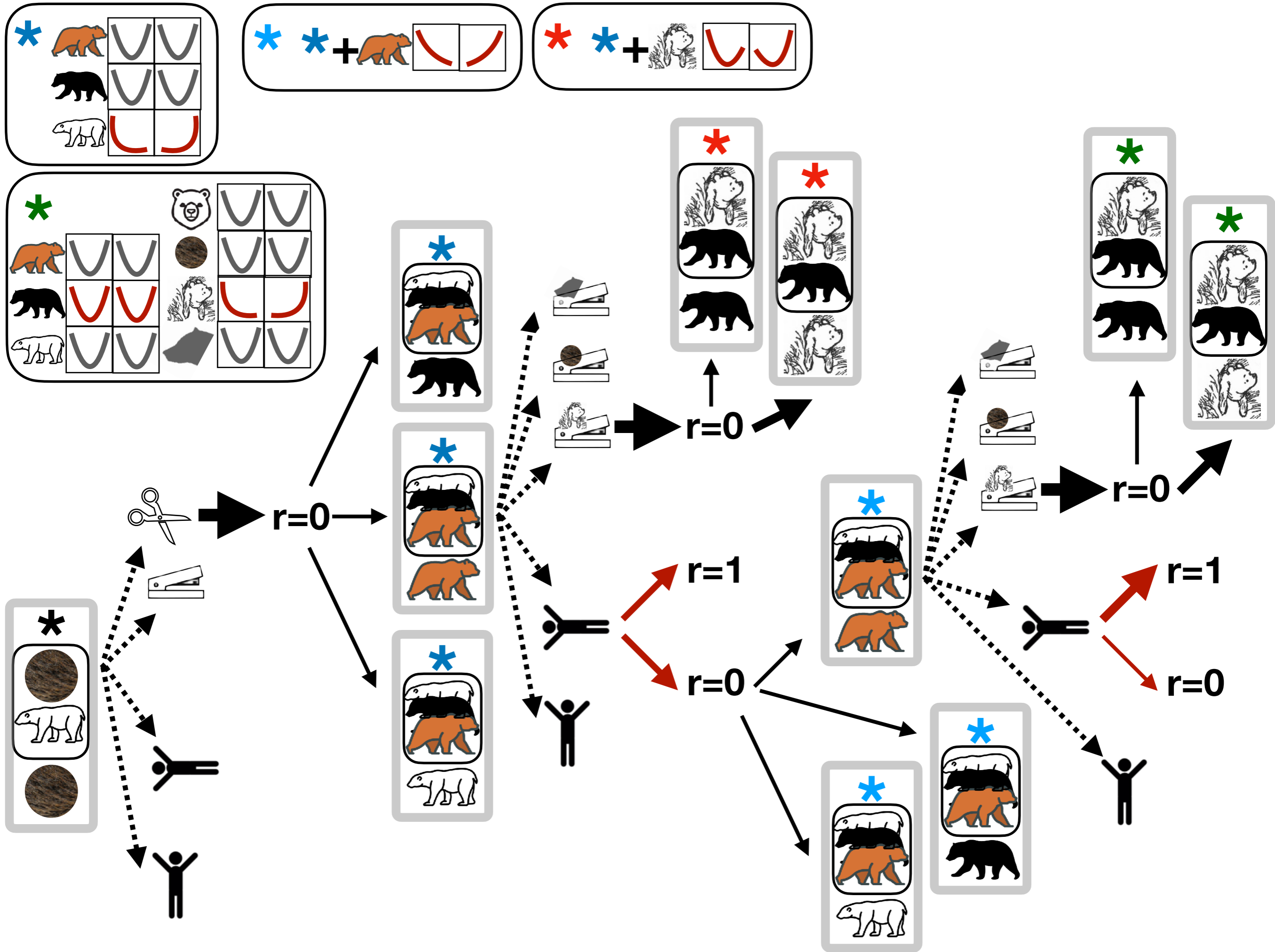


- Blind choice: basically giving up, may be preferred when stochasticity is very high



- Behavioural choice: equivalent to 2 bandits that the agent sees interleaved probabilistically



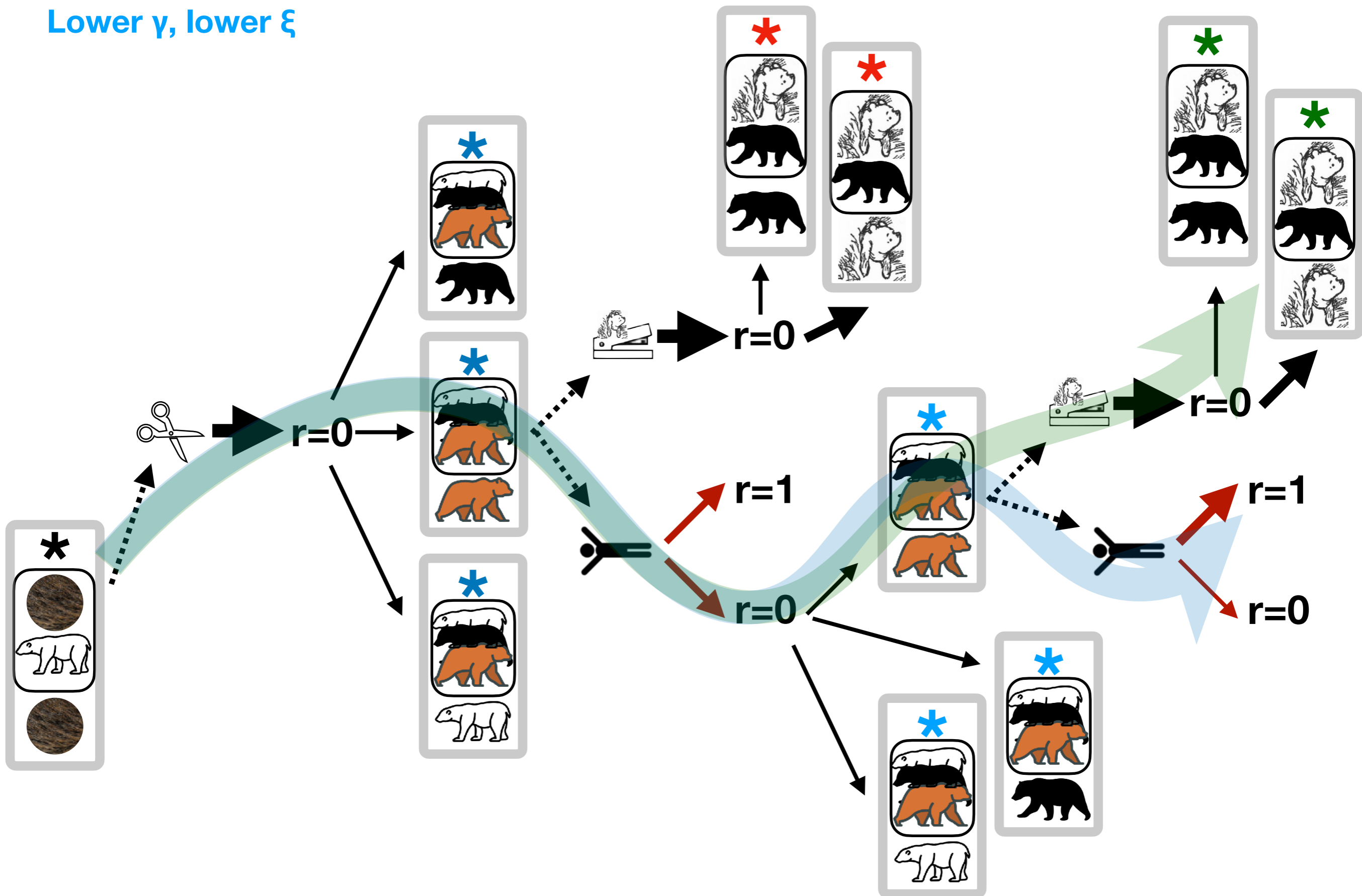


# Sources of value

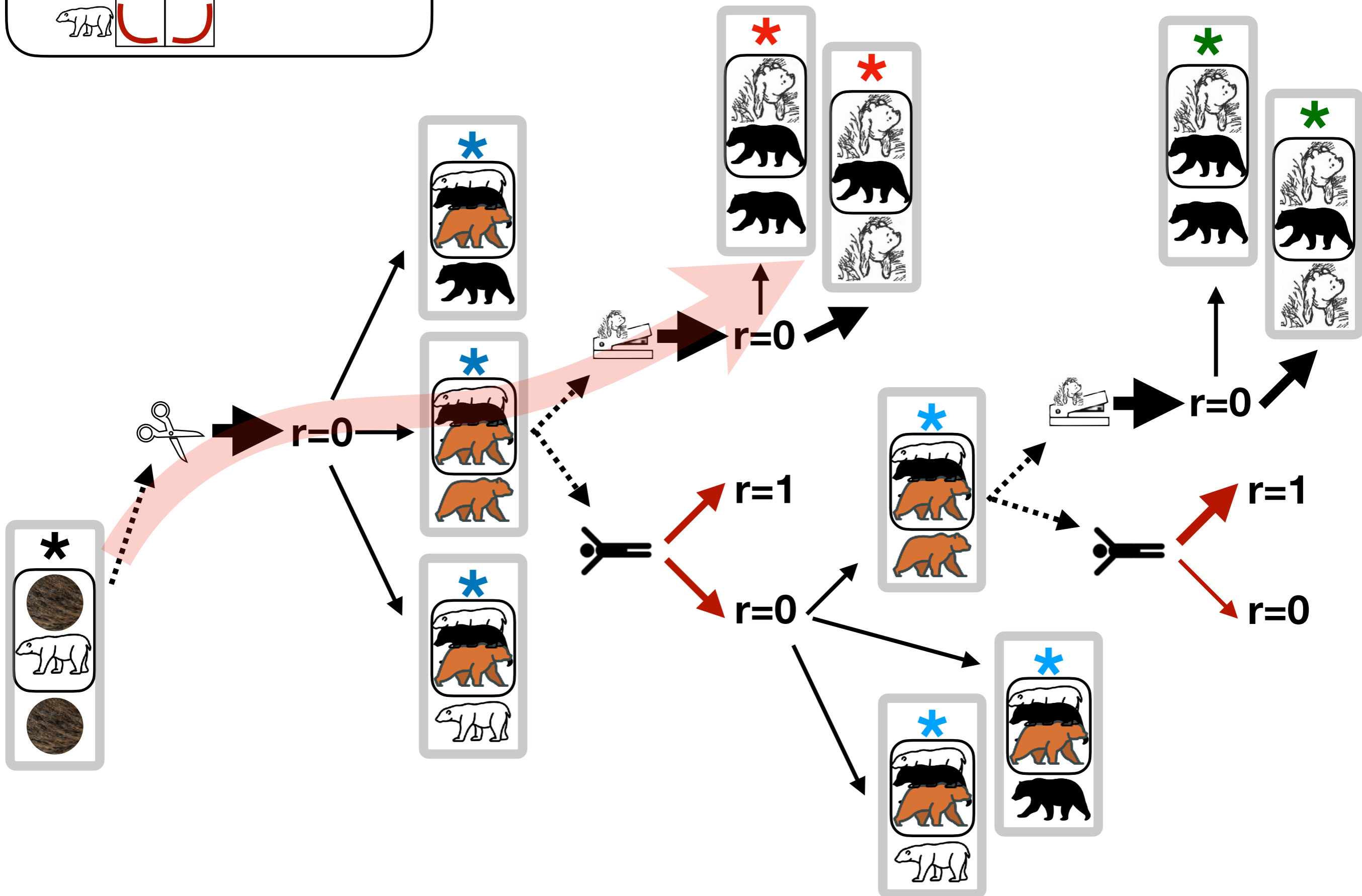
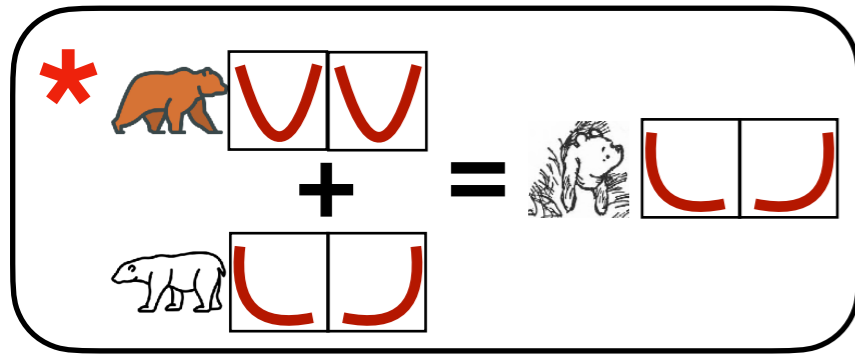
- Value of information
  - How much will 1 observation increase my future expected reward even if it confers low immediate reward?
  - Tends to go up with  $\gamma$  - the agent cares more about the far future
  - Tends to go down with  $\xi$  - more deterministic environments allow for faster learning
- Value of generalisation
  - Merging abstractions is an observation multiplier in the future
  - Tends to go up with  $\gamma$  even faster - the immediate reward for representational actions is 0
  - Tends to go up with  $\xi$  slightly - more aleatoric uncertainty means more chance for action choice being indifferent

Higher  $\gamma$ , higher  $\xi$

Lower  $\gamma$ , lower  $\xi$



An update rule that doesn't preserve uncertainty (pooling)



# How to incorporate human resource constraints?

- Resource constraints show up at various ways in the algorithm
  - In the limitations of what possible representations and representational operations are there
  - In the approximation to the Bayesian belief update
  - In the approximate tree search algorithm
- Choosing these is a non-trivial modelling step
  - Experiments may measure what computations do humans find more difficult
  - Theoretically, extending the information bottleneck formalism with architectural considerations may address the differential cost of various computations

# How to make planning useful in larger environments?

- Stochastic tree search trades off breadth for depth
- Embedding in a metric space allows for learning a structured generative model and using it for representational actions
- Amortisation
  - reusing the results of expensive computations
  - Training a neural network with the planning results as the target
- The tutorial setup
  - Do expensive planning computations in a small environment, and amortise them, so you have a fast representation learning algorithm
  - Use the fast algorithm in larger environments, so you can build up useful abstractions faster

# Conclusions

- Normative modelling runs into complexity barriers, have to keep a balance
- Representational planning is an interplay between the risk profile of the environment and the resource constraints of the agent
- The approximation the agent uses to the belief update determines what kind of abstractions it ends up with, which in turn determines what its beliefs will be
- Relating the model to human behaviour requires assumptions about the nature of approximations and the cost of computing

# Thank you



*“Sometimes you eat the bear,  
sometimes the bear eats you.”*