Eighth European Workshop on Reinforcement Learning

* Lille, France, June 30 — July 4, 2008
* Four solid days of 25-minute talks
* 52 accepted talks
* 140 participants
* By far the largest ever EWRL
* 3 invited talks
  * Dimitri Bertsekas, MIT
  * Jan Peters, Max-Planck Institute of Biological Cybernetics

Reinforcement Learning is multi-disciplinary

- artificial intelligence
- machine learning
- control theory
- operations research
- neuroscience
- psychology
- economics
- statistics
- ...

what is reinforcement learning?

- solving MDPs?
- learning from trial and error?
- learning-based artificial intelligence?
- understanding the mind?

- is there a key technical innovation of RL?

what is reinforcement learning?

the scientific innovation of RL is to focus on *sample-based* methods for solving *general* decision-making problems

MIND AND TIME:
A VIEW OF CONSTRUCTIVIST REINFORCEMENT LEARNING

Rich Sutton
University of Alberta

summary

- slow learning can make you learn fast
- we should focus on methods for long-term learning about structural aspects of the world
  - which features to generalize on
  - which state variables are Markov
  - which options are likely to be useful
  - which models are valid and useful
- representation-finding is the key to AI, and RL provides a good framework for attacking it

everyone wants to learn fast (in number of samples)

- we are willing to pay more for it in terms of computation and memory

- e.g., least-squares methods, batch methods, regularization, model-based methods, relational RL

- faster than regular-old, incremental, linear model-free methods like Sarsa and TD(λ)

in praise of incremental linear FA methods (1)

- incremental (easy to incorporate new data)
- intuitive and powerful
  - evidence adds and subtracts
  - good for expectations
  - good for probabilities (log-linear)
- general. linear can be as non-linear as you like by adding configural features

in praise of incremental linear FA methods (2)

- incremental linear methods can be very fast (e.g., tile coding, sparse coarse rep'ns)
- they can be made as fast as least-square methods by decorrelating the features
- approximate decorrelation can probably be done in $O(n \log n)$ time and memory
- they can learn even faster with incremental bias discovery methods such as IDBD

what is *n*?

- *n* is the number of features in the representation of the state
- a very large number, say a billion, though only a small fraction are active at any time
- state: e.g., where you are, what you are doing, what has been going on recently, what you expect...
- the state is an accumulation of input;
  n is much larger than the observation vector
- *n* (or maybe *n* log *n*) is the size of the reactive part of your mind

IDBD - stochastic meta-descent

(Sutton, Jacobs, Schraudolph)

- Incremental adaptation of per-feature step-size parameters
- can detect feature relevance; greatly accelerate learning; adapt bias
- an incremental form of hold-one-out cross validation

Incremental delta-bar-delta

$$\Delta w = \alpha * \text{error}$$

$$\Delta\alpha \propto \overline{\Delta w} * \Delta w$$

average Δw in the recent past

- IDBD is based on gradient descent at a meta level
- IDBD is a prototypical constructivist/discovery method
- e.g., make lots of features, sift them with IDBD, repeat
- IDBD is slow, but it prepares for new learning to be fast

Slow learning makes you fast

* life-long learning
* constructivism
* discovery

Slow learning makes you fast

- slowly discovering the structure of the world enables you to be fast later
- discovering feature relevance
- discovering state variables
- discovering temporal abstractions (dynamics, options)

outline

- everyone wants to learn fast
- IDBD
- model-based RL - the Dyna architecture
- constructing states and dynamics with TD nets and options
- reasoning - a unified Bellman equation for values and world models

model-based RL

- model-based or model-free?
- both!
- model = world knowledge
- constructing the model is a long-term, ongoing activity
- the system must have a life
  - planning is secondary, in the background

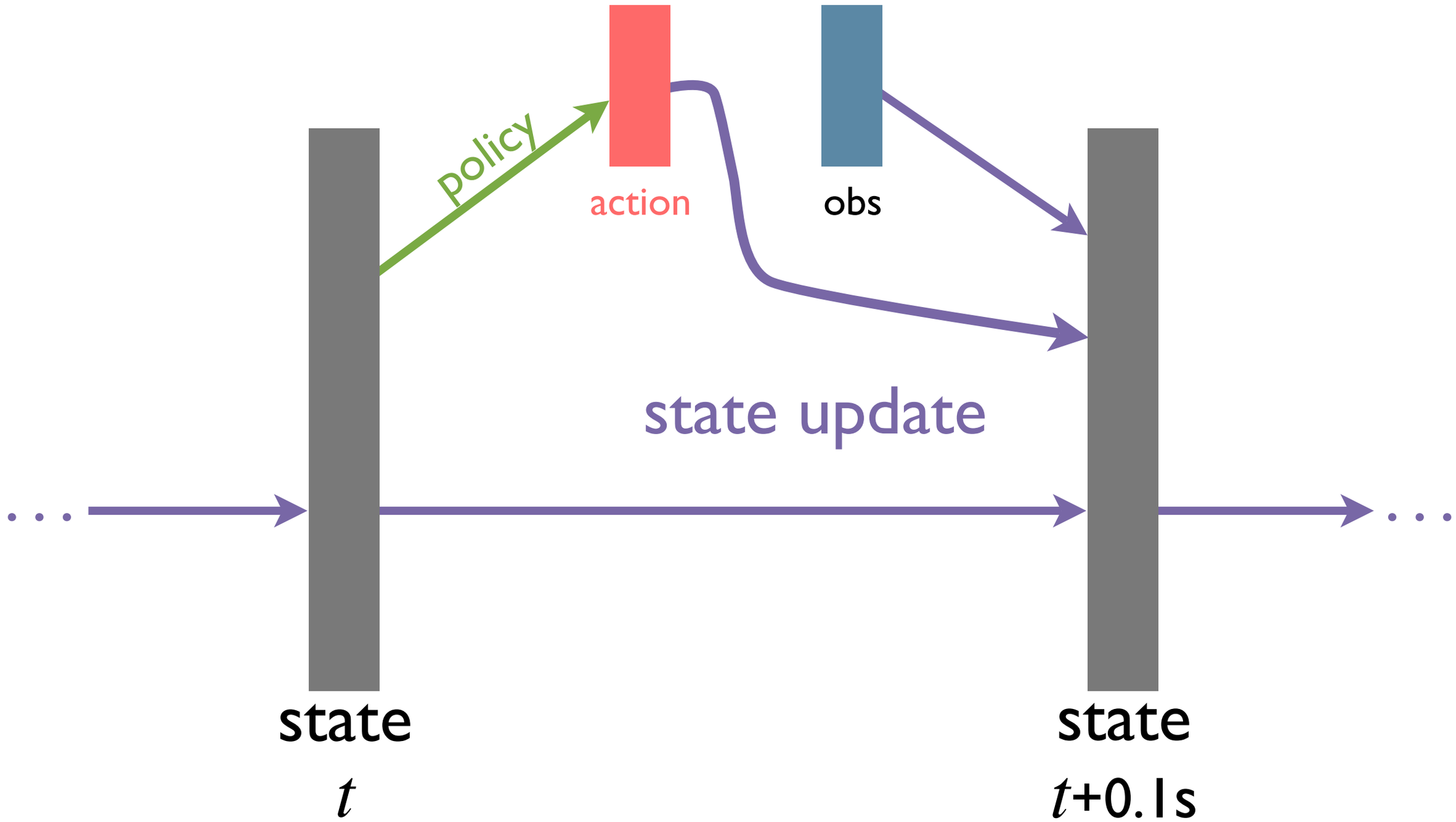
the imperative of time

- fast reaction is critical for natural minds
  - reacting quickly is the *first* requirement
- the special thing about life is that is has a "now"
  - the present is much more richly represented than it could ever be recalled

the foreground process

- the mind has certain processes that must run at top speed, say 10–100 hz
- the policy -- reaction time
- state update
- model-free learning

the reactive part of mind

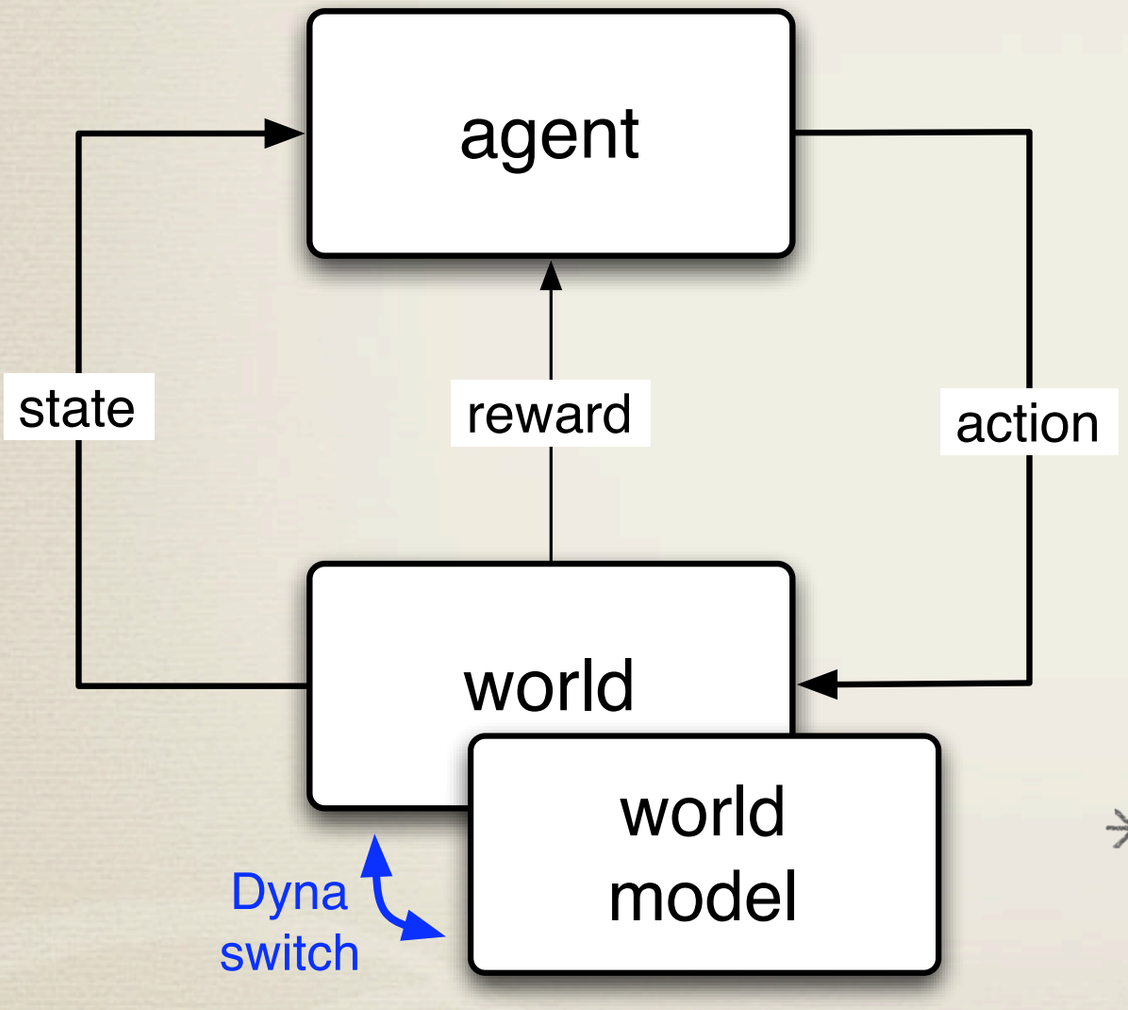


Dyna-style model-based RL

1. learn a model of the world
2. use the model to find a good policy (planning)
3. find a good policy without making or using a model (classical RL)

do all of these all the time without stopping

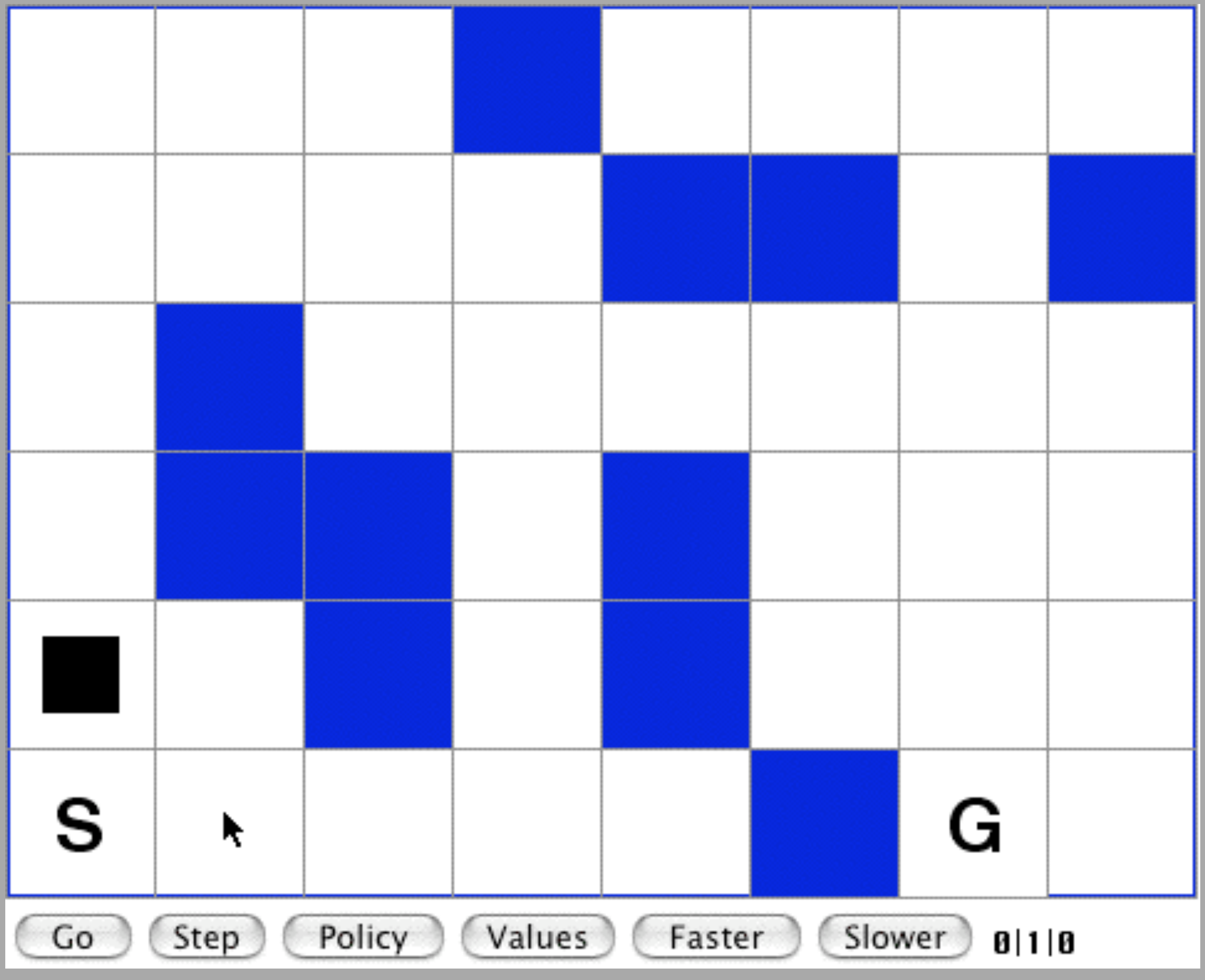
Dyna architecture



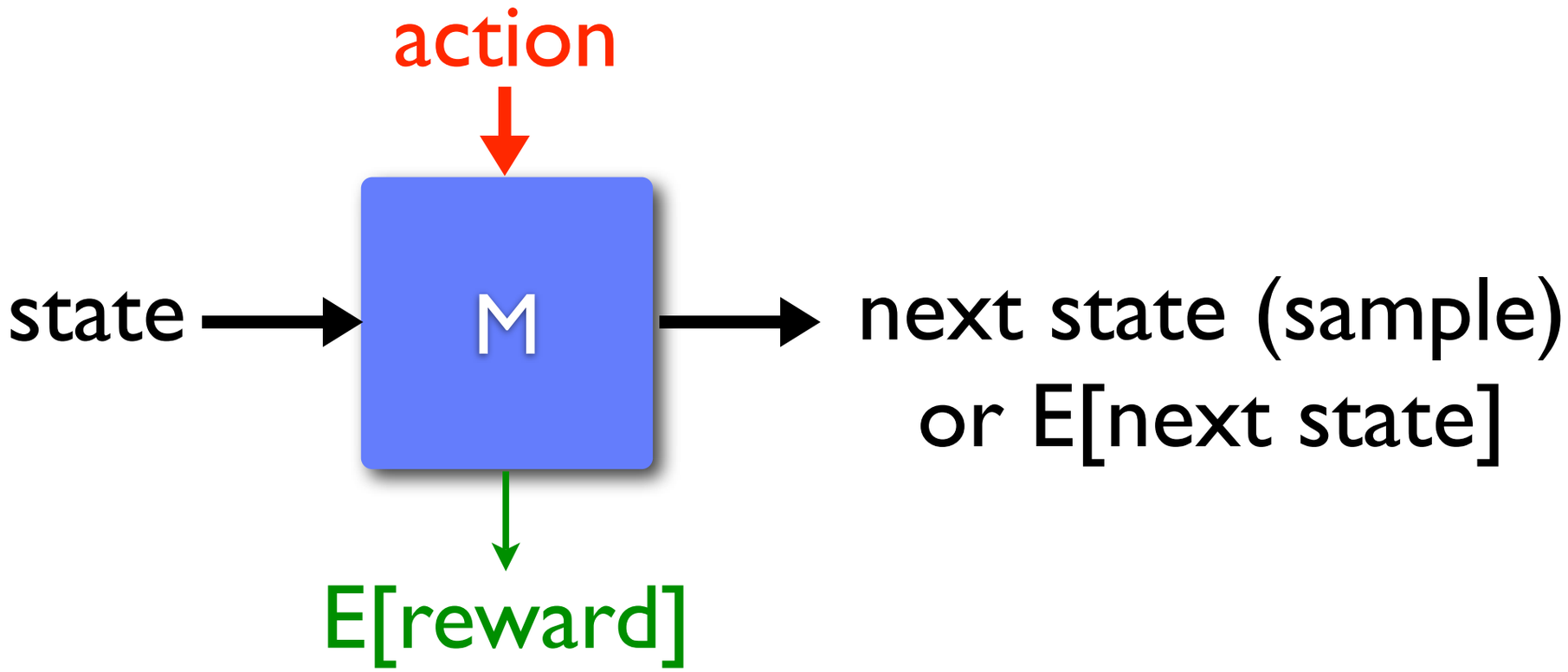
- learning and planning are achieved by the same RL algorithm (e.g., TD(0), Sarsa)
- applied to real or simulated experience
- altering the same policy and/or value function
- learning and planning are incremental, simultaneous, and asynchronous – nothing waits on anything else

Sutton 1990, Moore & Atkeson 1993, Singh 1992, Peng & Williams, 1993, Kuvayev & Sutton 1996, Degris, Sigaud & Wuillemin 2006, Paduraru 2007...

Dyna circa 1990

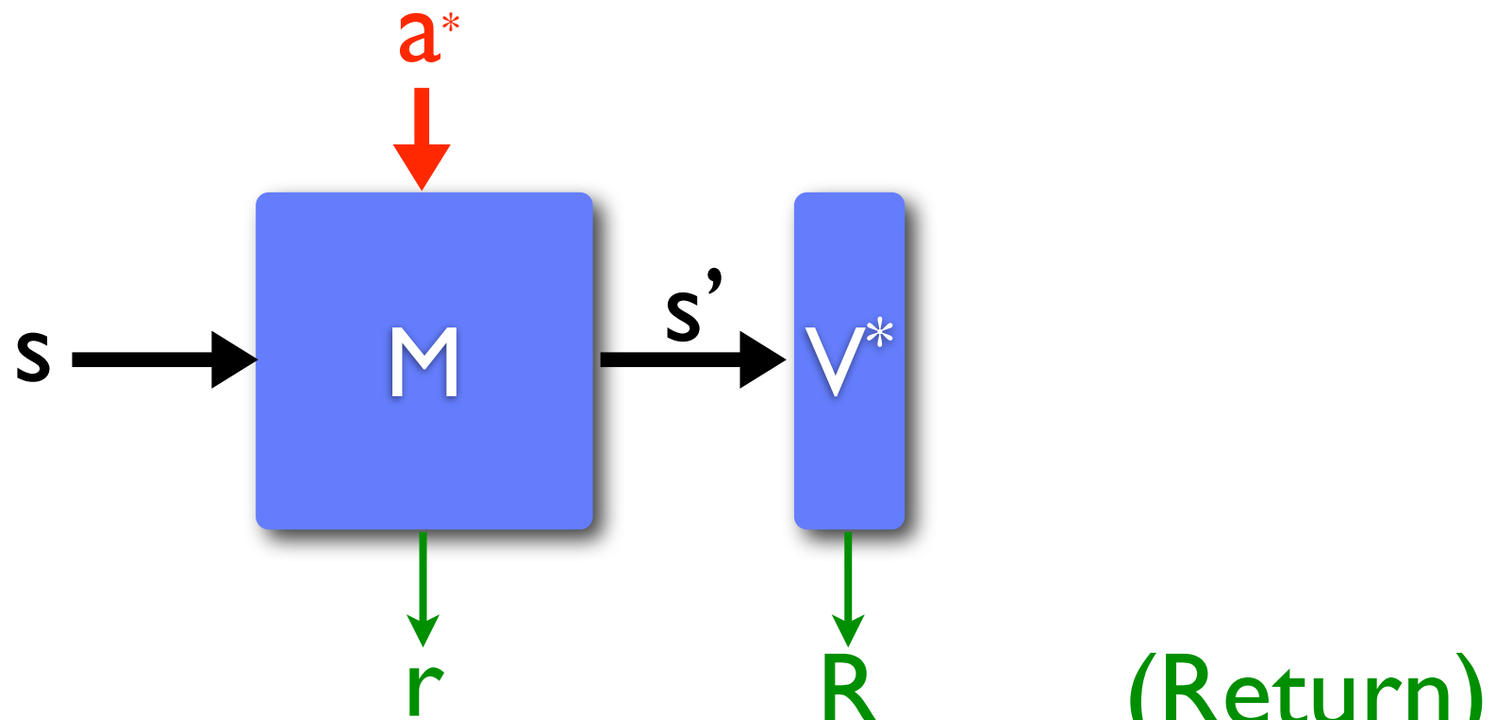


fundamental operation of a model - *projection*

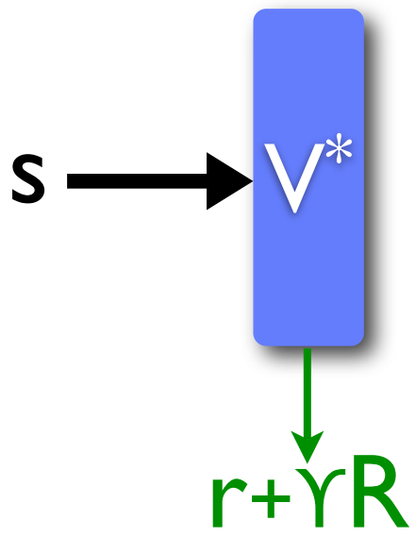


planning update, in pictures

hypothetical trajectory



provides target for



the sum of these is the target for this

where is Dyna now?

- ☑ online anytime reacting/learning/planning
- ☑ data-efficient learning control
- ☑ flexible search control (prioritized sweeping)
- ☑ consistent with flexible, powerful temporal abstraction (options)
- ☑ works with linear function approximation
- ☐ works with general FA, generative models
- ☐ works with constructivism

other methods for online learning and planning

- essentially there aren't any

  ...with any claim to generality

  ...that use general function approximation

- adaptive control methods limited to LQR systems

- LS methods are poorly suited to online use

outline

- everyone wants to learn fast
- IDBD
- model-based RL - the Dyna architecture
- constructing states and dynamics with TD nets and options
- reasoning - a unified Bellman equation for values and world models

the constructivist view of AI

- AI = making machines that can predict and control their low-level input-output stream of sensation and action

- construct a model of the world;
  find its key states and state transitions

- continually predict, control, plan

- continually reformulate representations for rapid, adaptive decision-making

constructivism

* discovery of features
* discovery of subproblems
* discovery of options
* discovery of option models
* discovery of state variables

Some current ideas for constructivism

* IDBD idea
* Simsek and Barto (systematic discovery of options)
* Makino and Takagi (discovery of TD networks)
* auxiliary problems

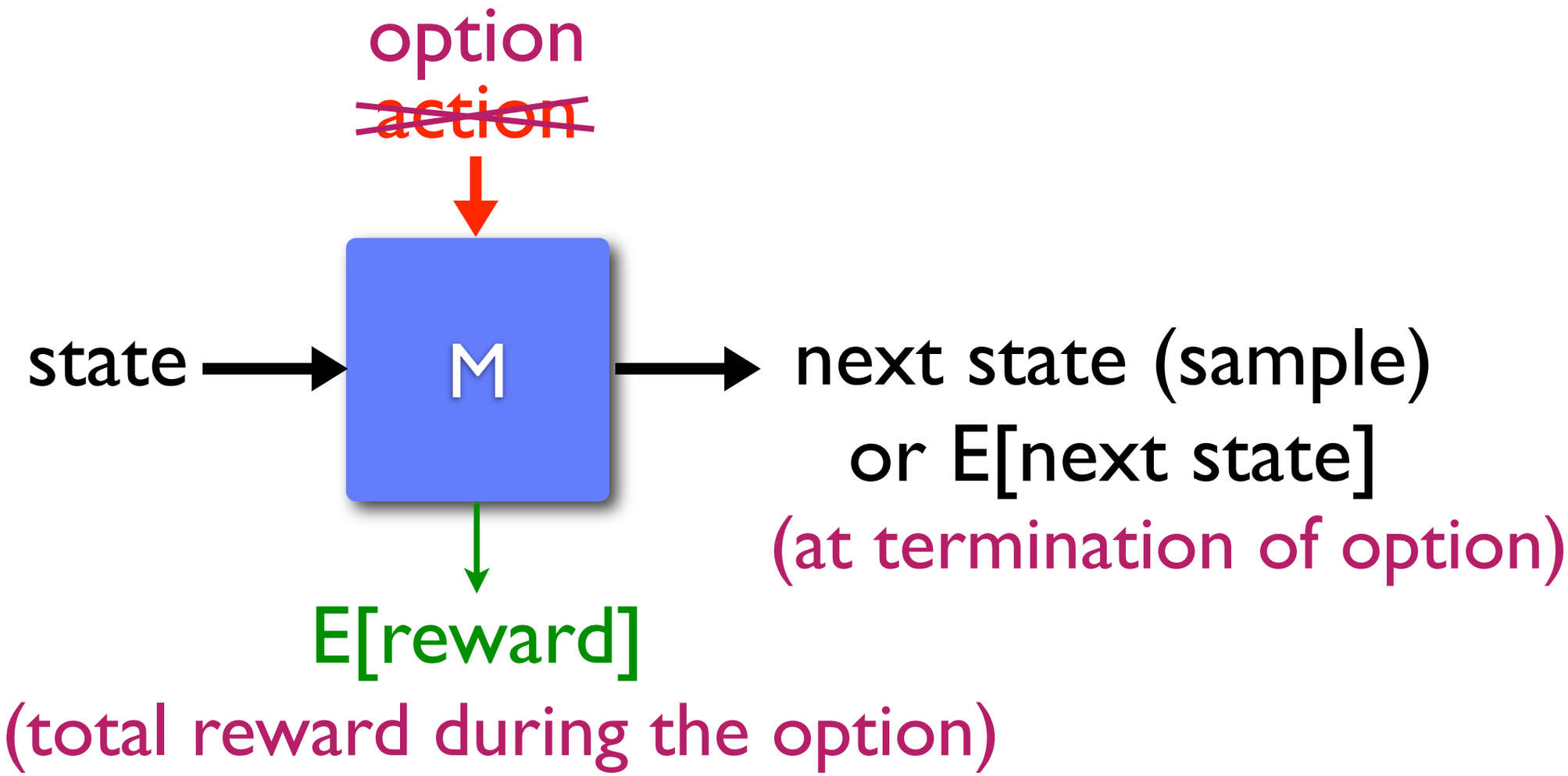
options

- temporally extended ways of behaving (policies with termination conditions)
- a generalization of primitive actions
- minimalist, lightweight
  - not necessarily to be executed
  - not necessarily hierarchical
  - not necessarily explicitly represented

option models

- predictions of of option outcomes:
  - state at termination
  - cumulative reward (or anything) along the way
- option models generalize value functions
- option models generalize one-step models of the world's dynamics – and can replace them one-for-one in familiar planning methods
- a powerful language for knowledge representation

the fundamental operation of a model - *projection* - generalizes readily to options



predictive state representations

- predictions of option outcomes (a generalization of conventional PSRs) may make good state variables
  - they can be learned independently of their use
  - they factor well; they are robust to large state spaces
  - they lead to natural abstractions
  - they work with linear function approximation

all knowledge can be expressed as option models

* options are powerful, expressive
* too much so?
  * there are so many possible options
  * and each can be the basis for state variables
* how do we pick which ones to use?

how to choose options?

* don't. not entirely.
* ultimately it has got to be the agent's job.
  our job is to set meta strategies
* options oriented around (sub)goals are a good idea
* e.g., pick a feature/signal, make it a subgoal
  * learn how to achieve it
  * learn when one can achieve it

which signals are good targets for learning about?

1. reward

2. other signals assumed a prior to be of interest, i.e., salient stimuli

3. signals found to be useful/used in predicting or controlling signals in the first two categories, or previously in this category

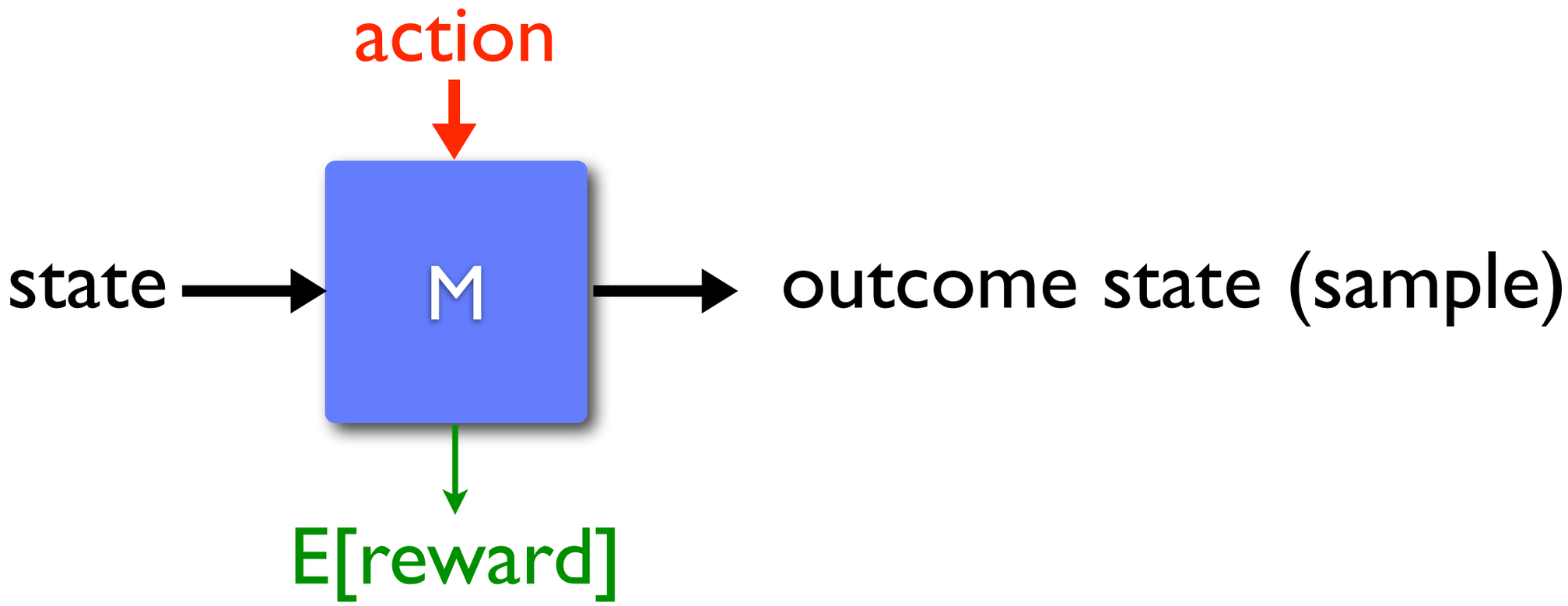
4. signals that yield to learning; that admit simple solutions; that are near deterministic...

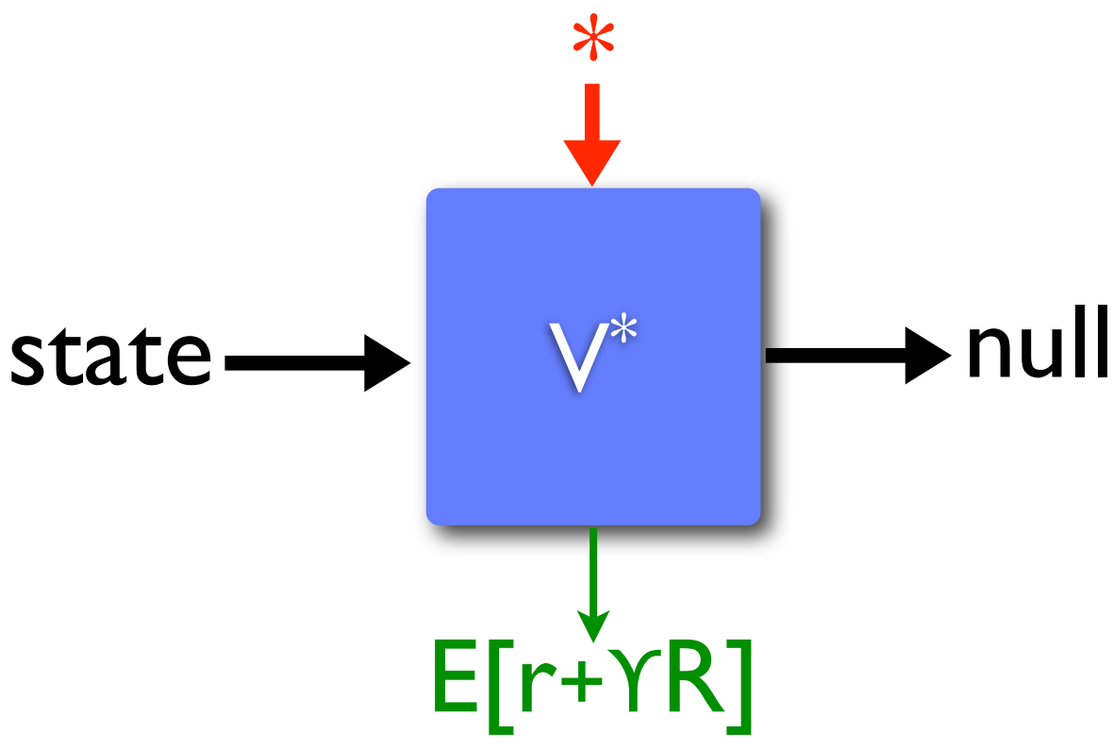
outline

- everyone wants to learn fast
- IDBD
- model-based RL - the Dyna architecture
- constructing states and dynamics with TD nets and options
- reasoning - a unified Bellman equation for values and world models

a further unity of value and model

thing being learned \ source of experience	world	world model
value function	classic RL	Dyna (planning)
world model	model learning	reason

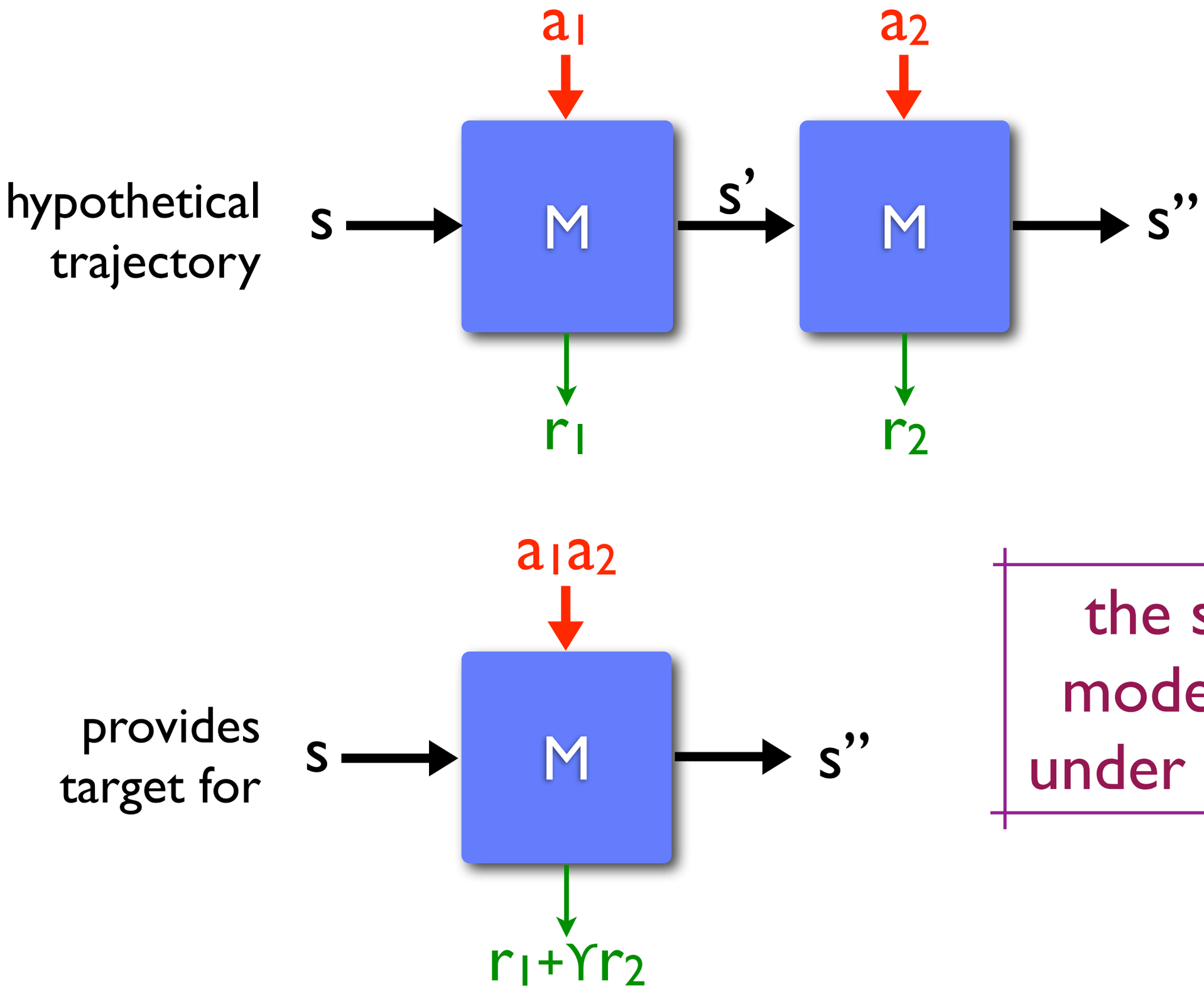




the value function
can be viewed as
an option model

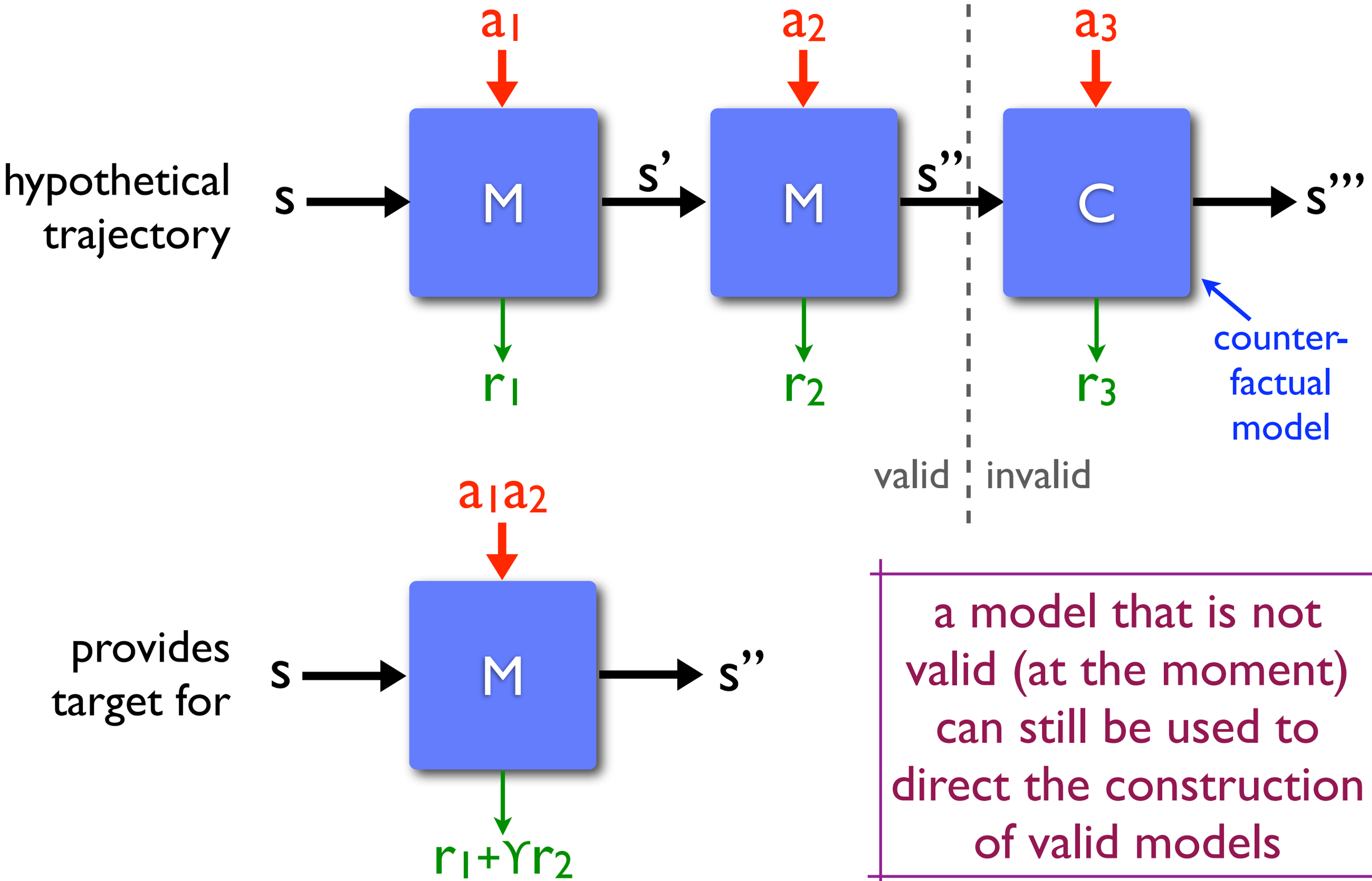
and vice versa

planning update (generalized Bellman eq)



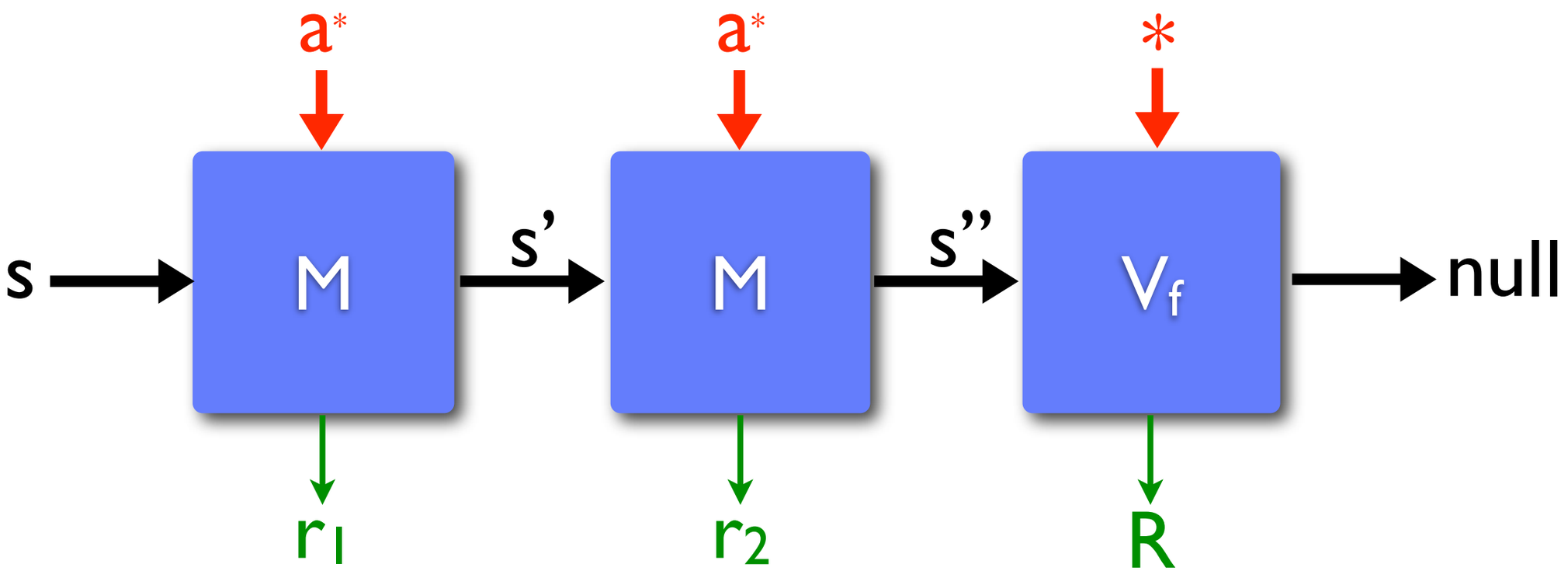
the set of valid models is closed under composition

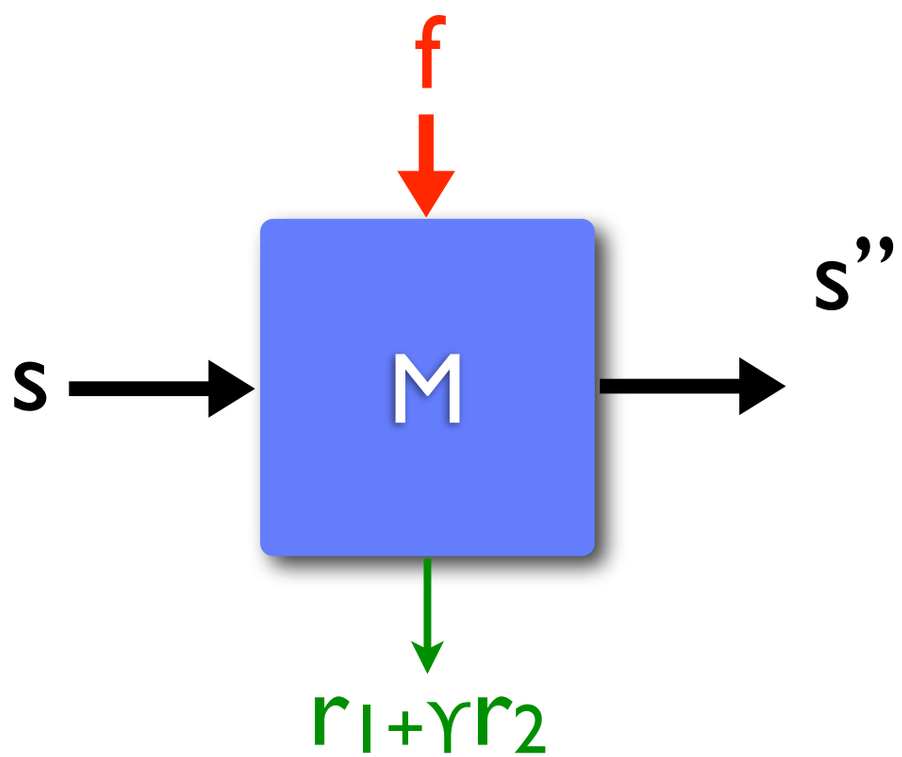
use of counter-factual models



a model that is not valid (at the moment) can still be used to direct the construction of valid models

e.g., subproblems are counter-factuals





will learn model for "f" in preparation for when the false value function V_f becomes true again

summary

- slow learning can make you learn fast
- we should focus on methods for long-term learning about structural aspects of the world
  - which features to generalize on
  - which state variables are Markov
  - which options are likely to be useful
  - which models are valid and useful
- representation-finding is the key to AI, and RL provides a good framework for attacking it

* thank you for your attention

* thanks to david silver