Bayesian Real-time Dynamic Programming

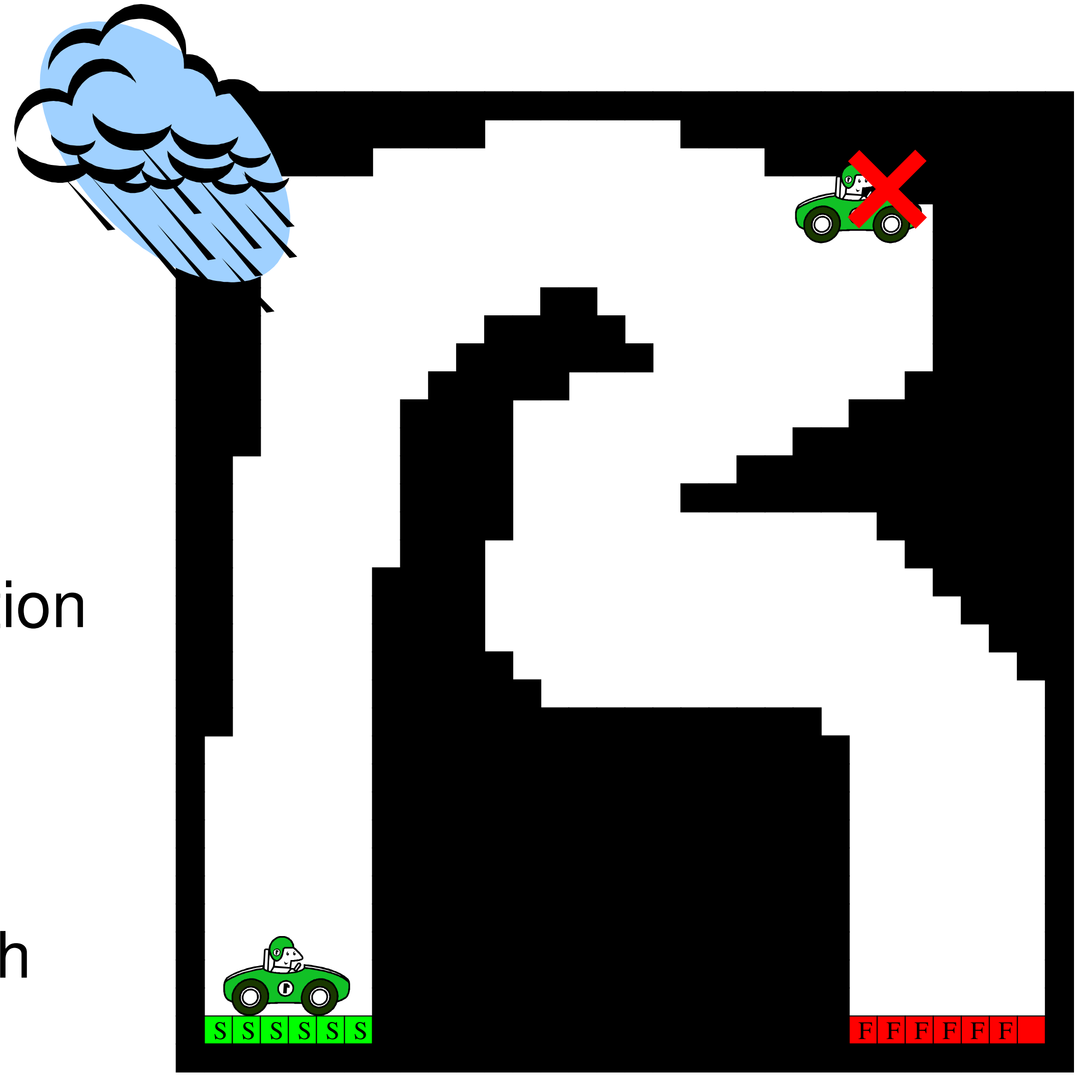
- Scott Sanner — NICTA & ANU
- Robby Goetschalckx — K.U. Leuven
- Kurt Driessens — K.U. Leuven
- Guy Shani — MSR → Ben Gurion Univ.

Outline 

- MDPs
  - Dynamic Programming (DP)
  - Real-time DP
- Caveats of RTDP and variants
  - *Value of information* to the rescue!
- Results

Running Example: Racetrack MDP

- State:
  - *(x,y)* position
  - *(x',y')* velocity
- Action:
  - *(x'',y'')* acceleration
- Objective:
  - Least-cost path from start to finish



MDP Solution

- Find a policy $\pi = \pi^*$ that maximizes:

$$V^{\pi}(s) = E_{\pi}\left[\sum_{t=0}^{\infty} \gamma^{t} \cdot r_{t} \middle| s = s_{0}\right]$$

MDP Solution

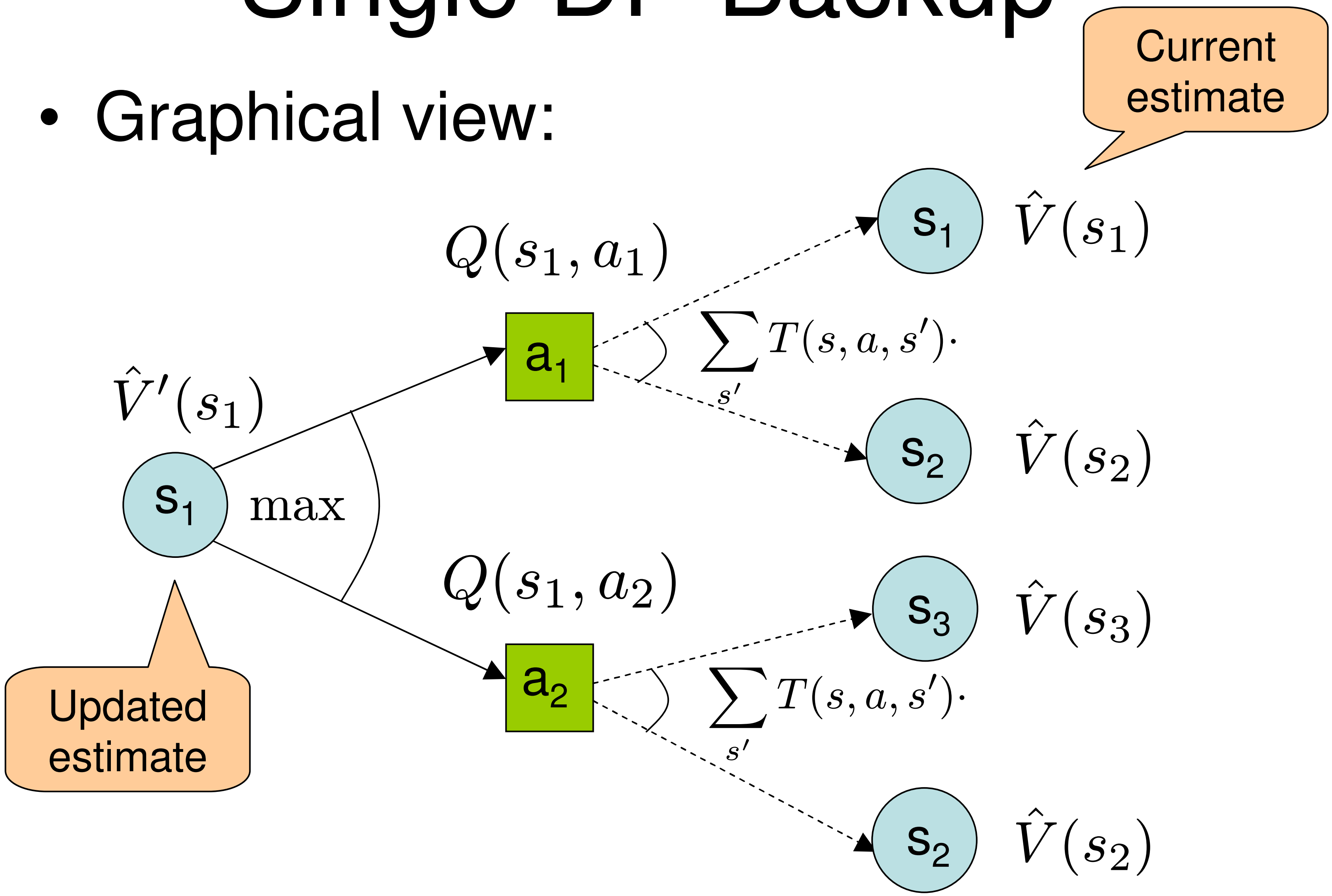
- Find a policy π=π* that maximizes:

$$V^{\pi}(s) = E_{\pi}\left[\sum_{t=0}^{\infty} \gamma^t \cdot r_t \middle| s = s_0\right]$$

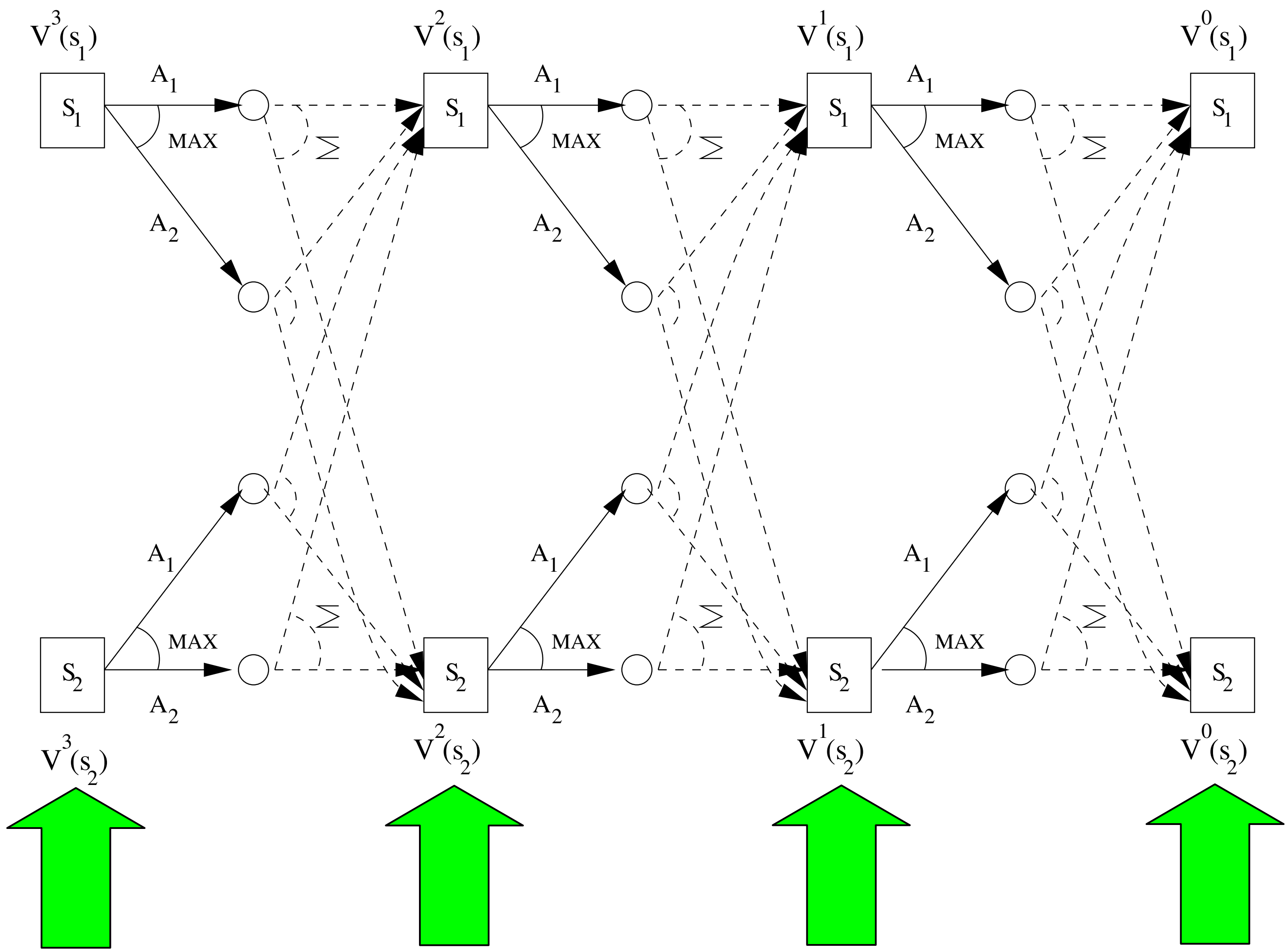
- Solve via dynamic programming (DP)

Single DP Backup

- Graphical view:

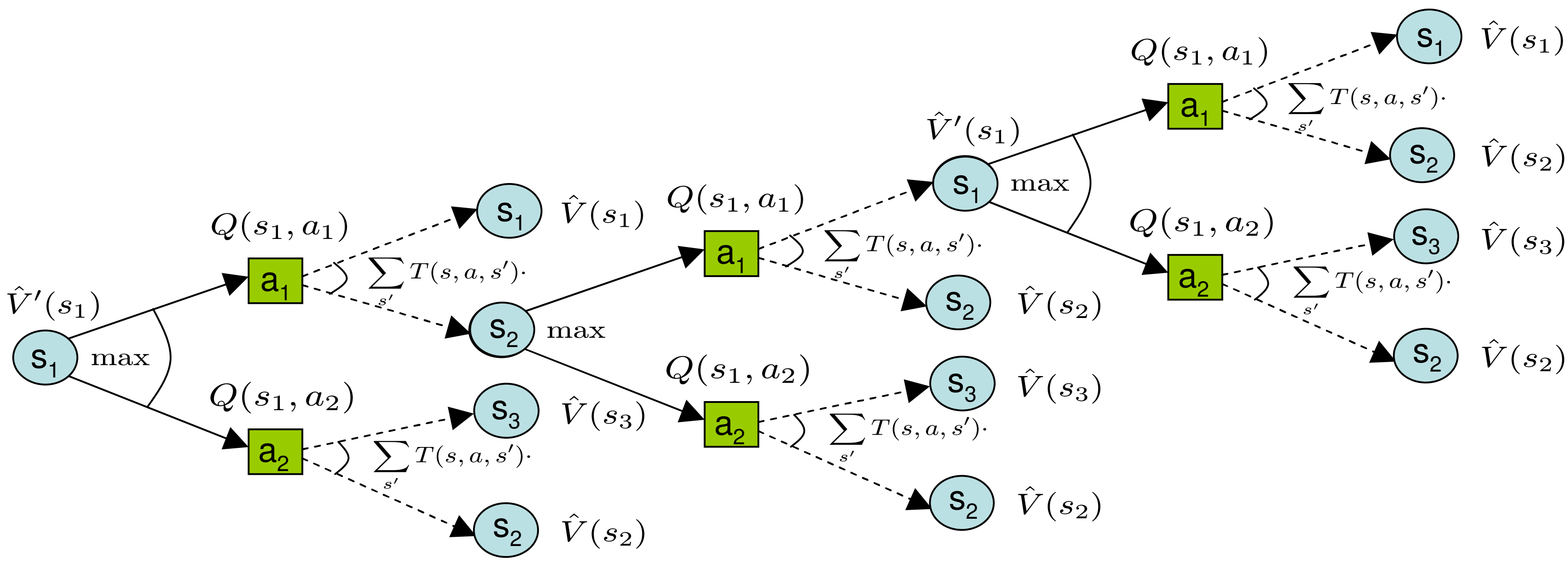


Synchronous DP Updates (VI)



Asynchronous DP Updates

- Or… can update states in any order:



- Still provably converges!

Question:
how to order updates to converge quickly?

Real-time Dynamic Programming

- ***Reachability*** and drawbacks of synch. DP (VI)

  – Better to think of ***relevance*** to optimal policy

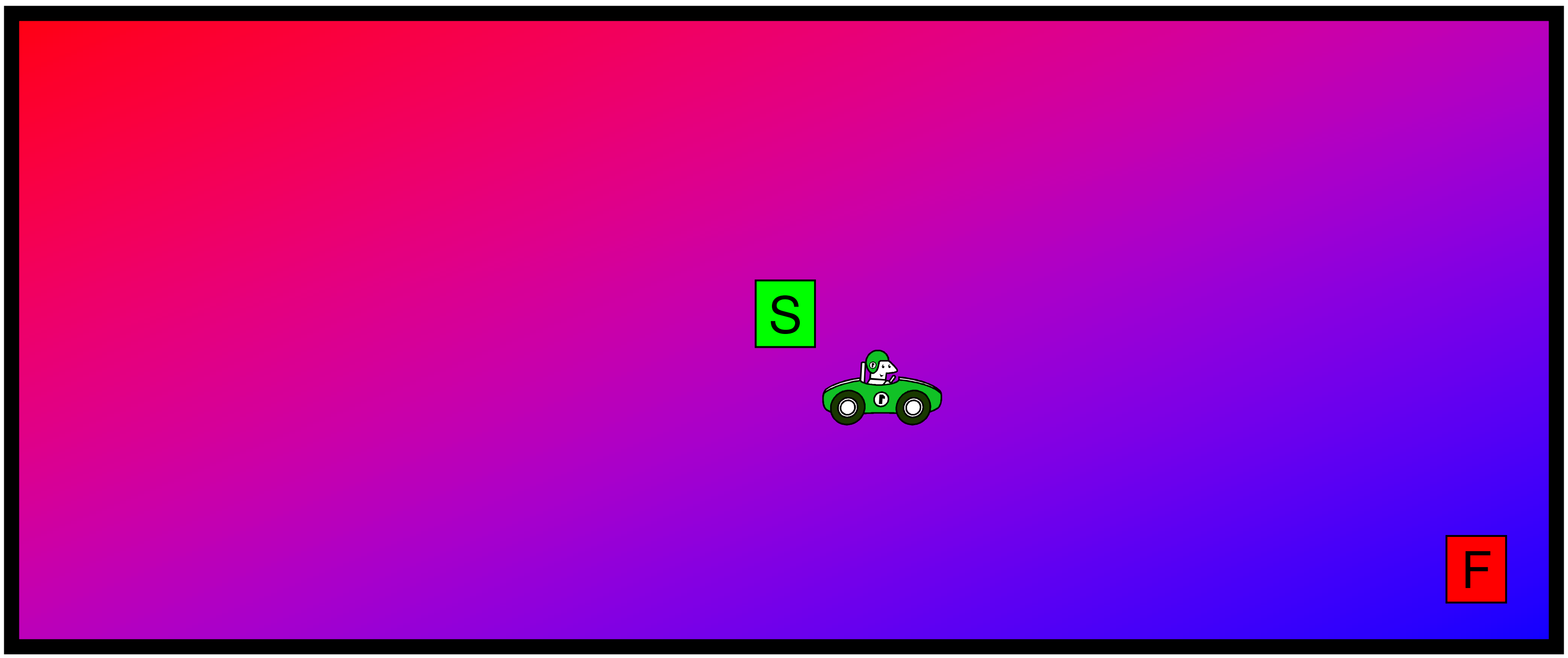
- RTDP focuses async. updates on relevant states!

Drawback of RTDP

- Focus on states with highest value uncertainty
  - i.e., highest bound gap

Unconverged

Converged



  - RTDP may search where already converged

RTDP Improvements

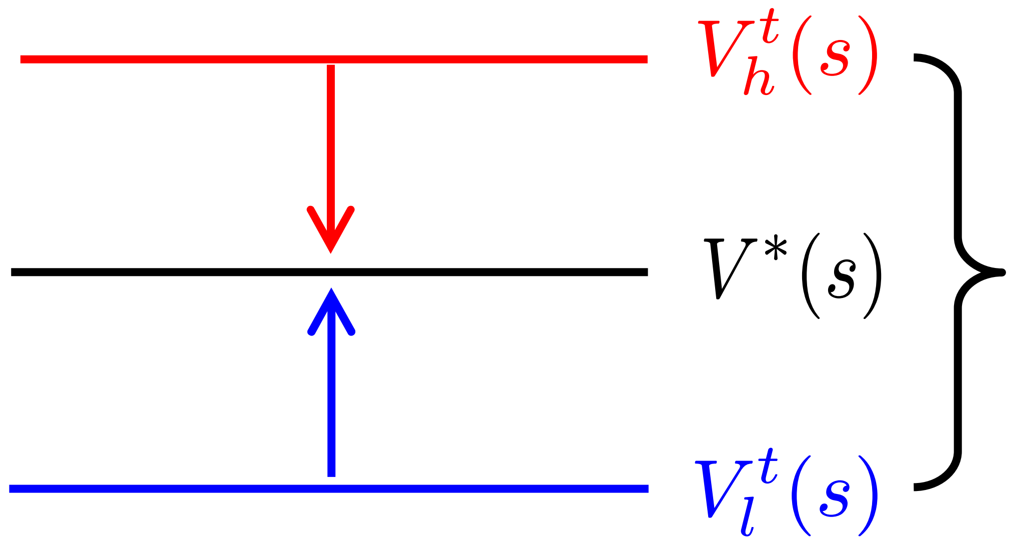
- **Labeled RTDP** (Bonet & Geffner, 03)
  - label states when convergence detected
  - don't update converged states in future!

- **Bounded RTDP** (McMahon, Likhachev, Gordon, 05)
  - prioritize states with highest value uncertainty
  - "soft LRTDP"
  - "forward prioritized sweeping"

How to compute uncertainty?
~~Bellman error?~~

Value Uncertainty via Monotone Bounds

- Initialize two value functions

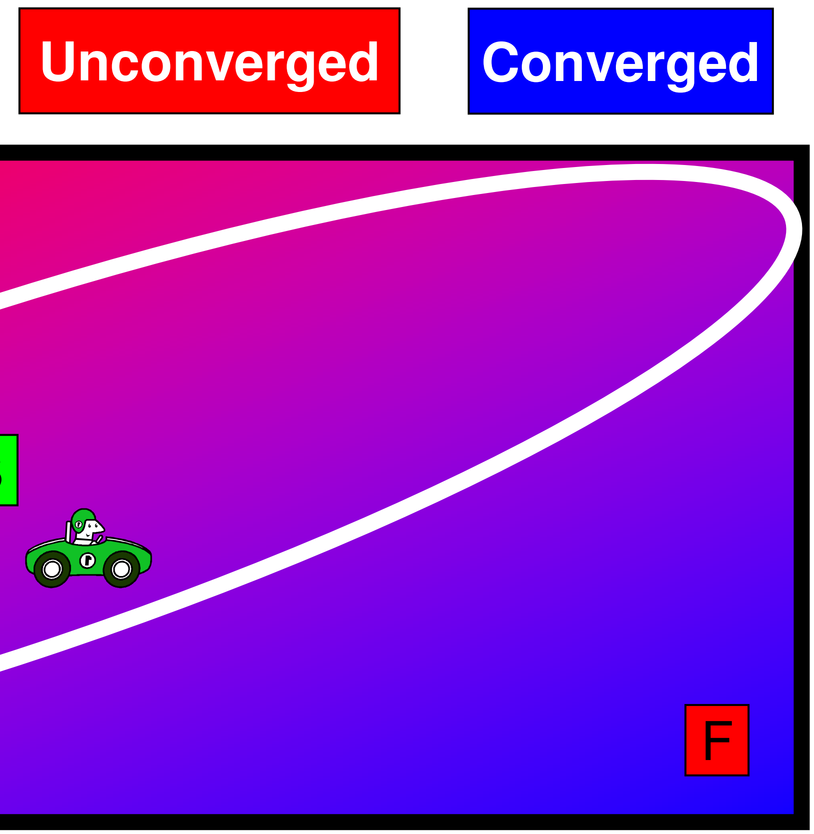


Unlike "local Bellman error", bound gap is upper bound on value error!

- Do DP updates for $V_h^t(s)$ and $V_l^t(s)$
  – Provides strict value bounds at all stages!

Bounded RTDP

- Focus DP on least converged states
  – i.e., highest bound gap



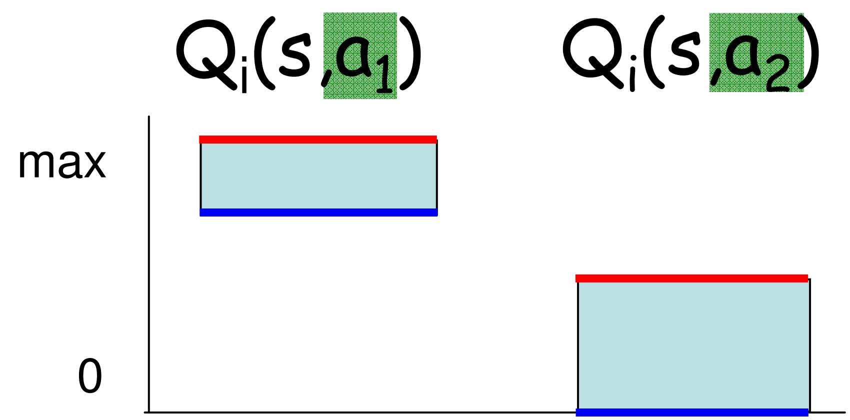
  – May search where value unlikely to change

Bayesian RTDP

Asychronous DP updates where they count!

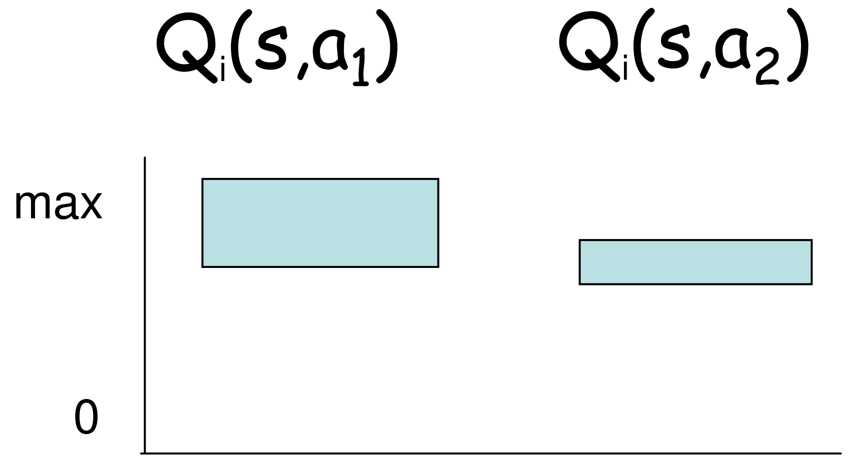
Focusing Async. DP Updates

- Examine $Q(s,a)$
  - $Q(s,a) = p_{1a}V(s_1) + \ldots + p_{ia}V(s_i) + \ldots + p_{ka}V(s_k) + R$
- Plug in $V_l(s_i)$ and $V_h(s_i)$
  - Get: $[Q_{il}(s,a), Q_{ih}(s,a)]$
- Update state s_i?
  - No. Why?

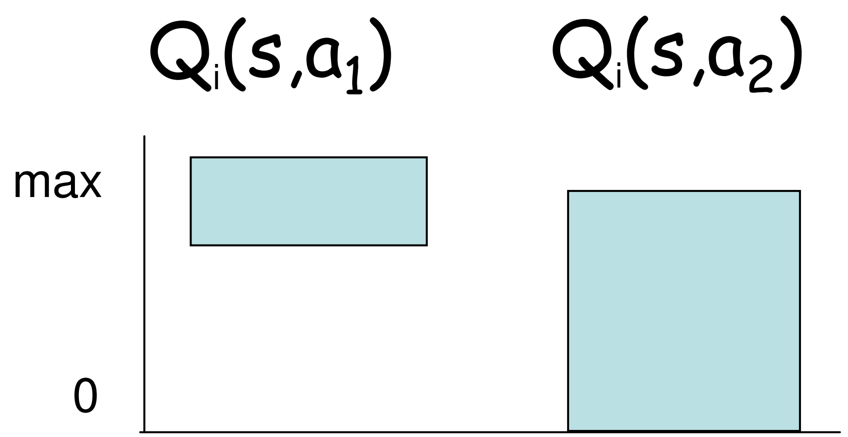


Harder Cases

- Update state s_i?
  - Maybe.
  - Why?



- Here?
  - Probably.
  - Why?



Bayesian Formalization I

- Assume uniform belief distribution over bounds
- Calculate *expected* Q-values w.r.t. beliefs

$$E[Q_{a,s}|\vec{\theta}] = R(s,a) + \int_{\vec{v}} \prod_{s'} P(v_{s'}|\vec{\theta}) \left[\vec{\Gamma}_{a,s} \cdot \vec{v}\right] d\vec{v}$$

$$= R(s,a) + \vec{\Gamma}_{a,s} \cdot \frac{\vec{V}_h + \vec{V}_l}{2}$$

$$E[Q_{a,s}|\vec{\theta}, v_t^*] = R(s,a) + \int_{\vec{v}} \delta_{v_t^*}(v_t) \prod_{s' \neq t} P(v_{s'}|\vec{\theta}) \left[\vec{\Gamma}_{a,s} \cdot \vec{v}\right] d\vec{v}$$

$$= \underbrace{E[Q_{a,s}|\vec{\theta}] - T(s,a,t)\left(\frac{V_h(t) + V_l(t)}{2}\right)}_{c_{(a,s,t,\vec{\theta})}} + \underbrace{T(s,a,t)}_{d_{(a,s,t)}} v_t^*$$

Bayesian Formalization II

- What is gain of exactly knowing v_t^*

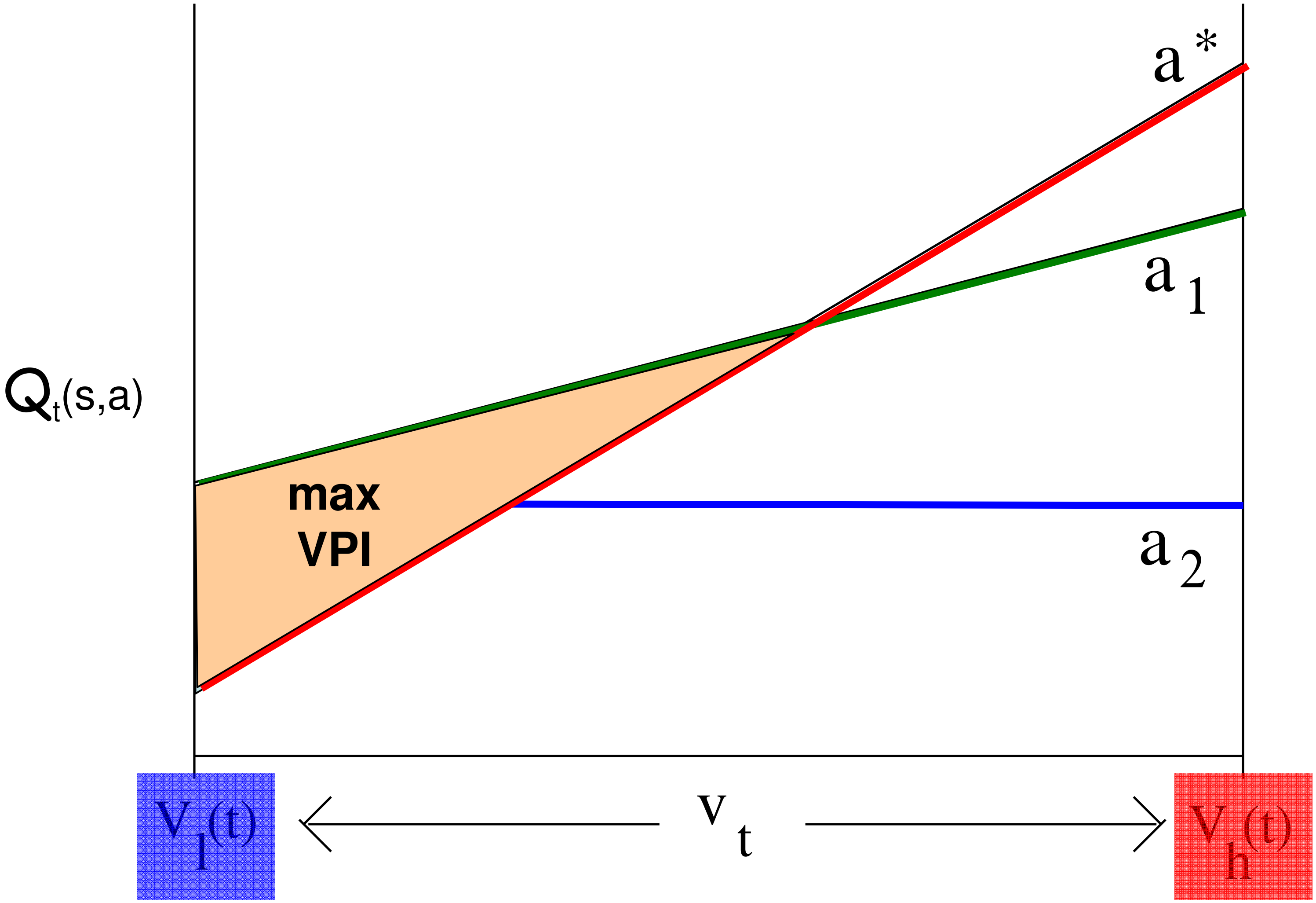
$$Gain_{s,t,a,a*}(v_t^*) =$$
$$\max\left(0, E[Q_{a,s}|\vec{\theta}, v_t^*] - E[Q_{a^*,s}|\vec{\theta}, v_t^*]\right)$$

- EVPI = expected gain of exactly knowing v_t^*

$$VPI_{s,a^*}(t) = \max_{a \neq a^*} \int_{v_t^* = -\infty}^{\infty} P(v_t^*|\vec{\theta})\, Gain_{s,t,a,a^*}(v_t^*) dv_t^*$$
$$= \frac{1}{V_h(t) - V_l(t)} \max_{a \neq a^*} \int_{v_t^* = V_l(t)}^{V_h(t)} Gain_{s,t,a,a^*}(v_t^*) dv_t^*$$

Expected VPI: Graphical View

- What is potential gain of knowing v_t better?

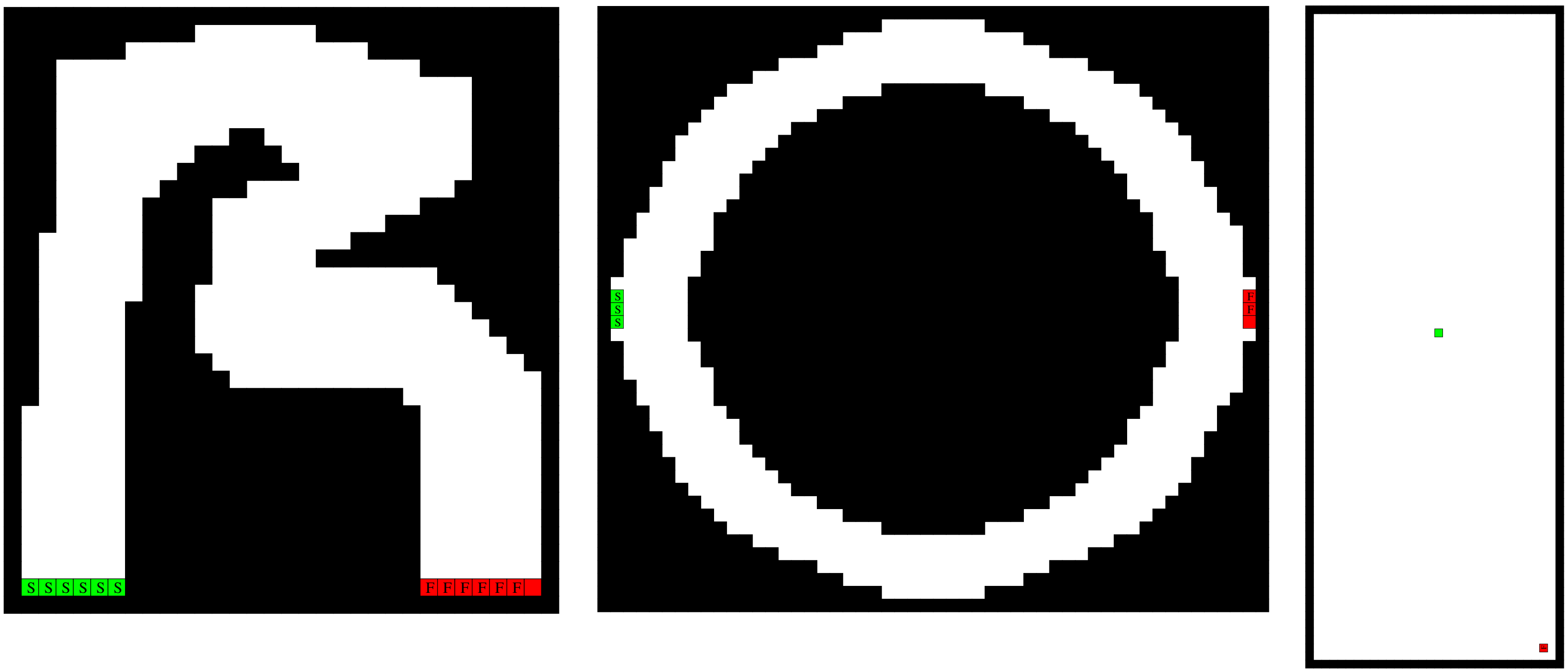


Key Observations

- VPI not only important for directing search
- Also important for **early trial termination**
  – terminate with some prob. if VPI < threshold
- And **efficient to compute**
  – Complexity of Bellman backup to compute VPI for all successor states!

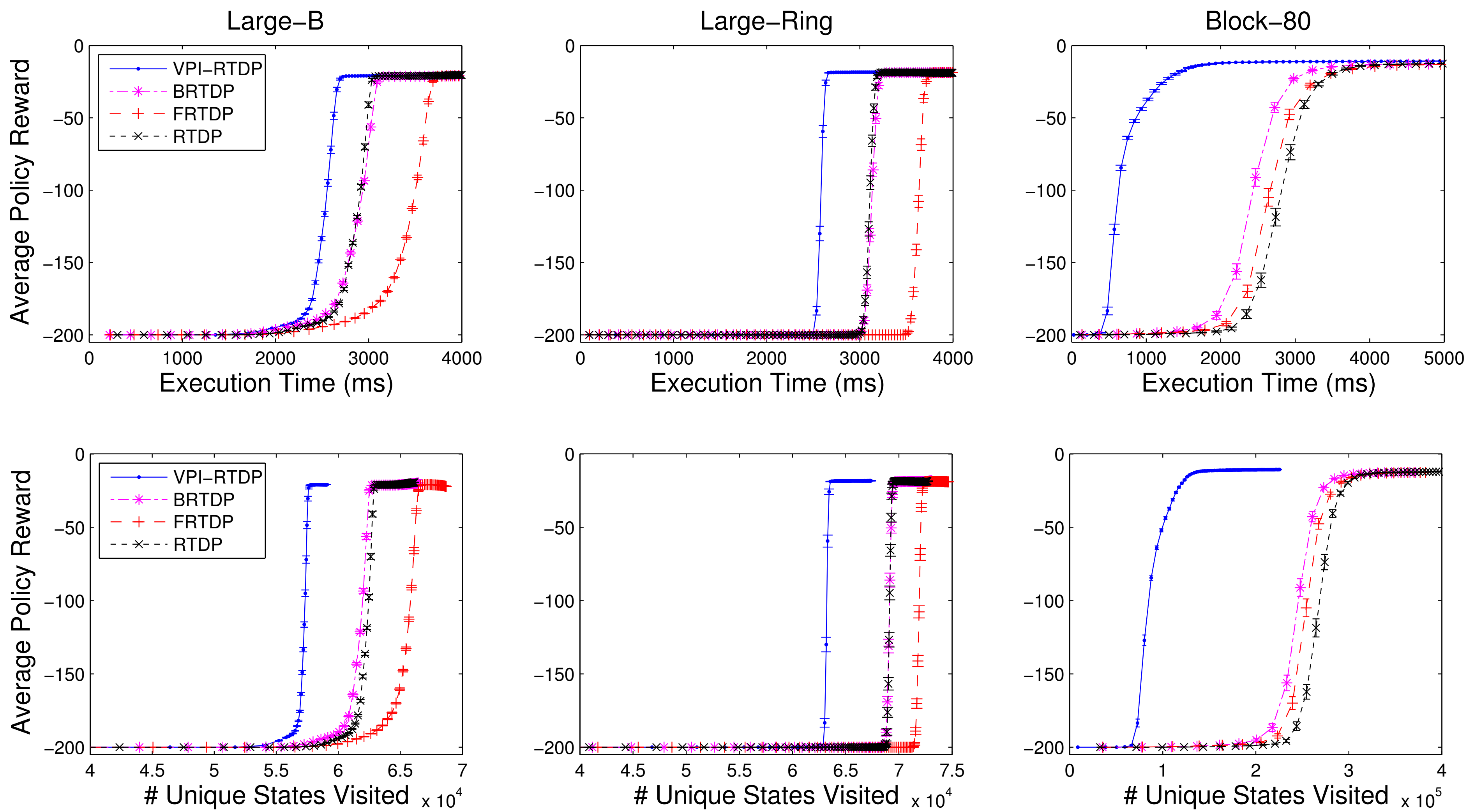
Empirical Evaluation

- Used modified Racetrack domains from (Barto, Bradtke, Singh, 1993; Smith, Simmons, 2006)



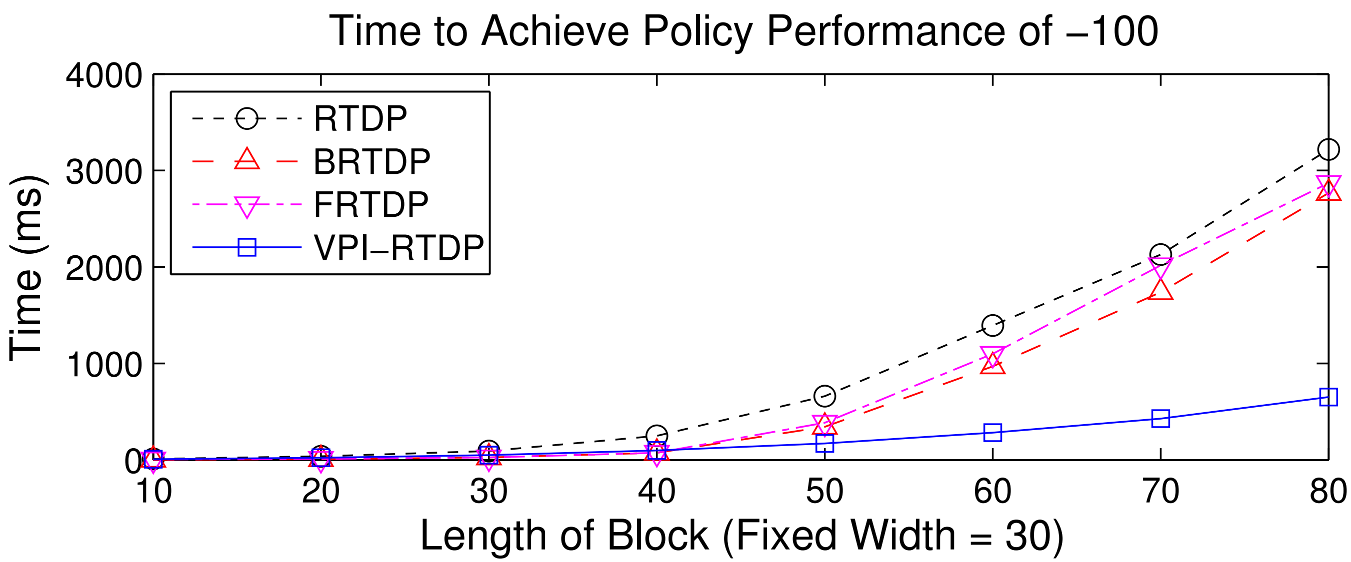
Racetrack Results

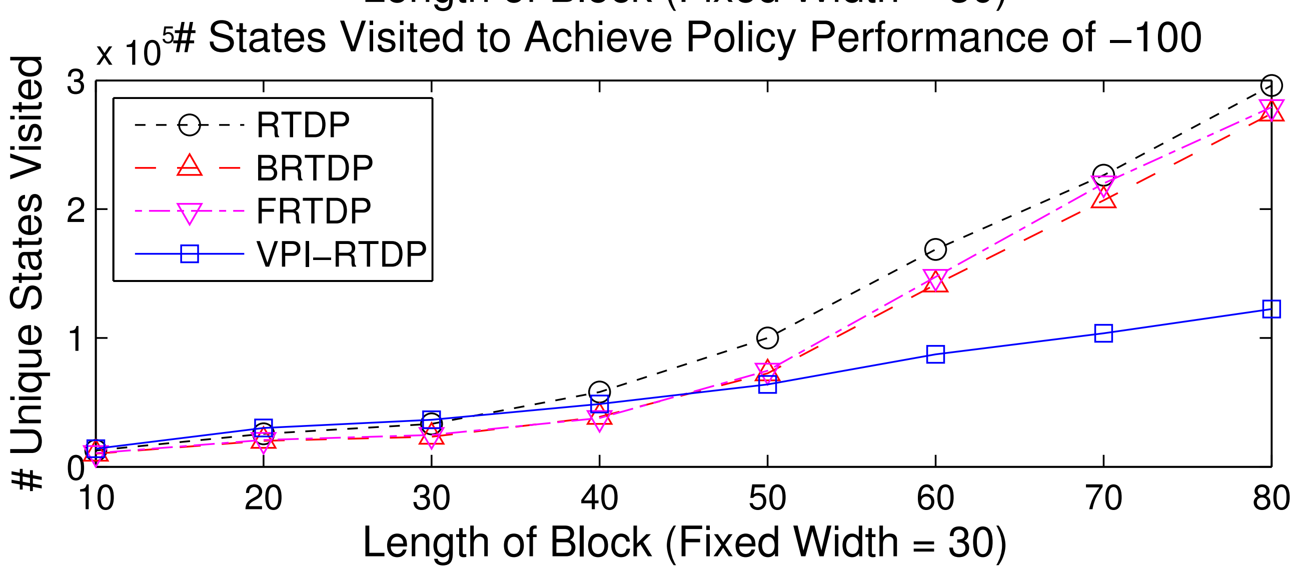
better anytime policy performance, fewer visited states



Scaling Performance

performance gap widens as problem size grows

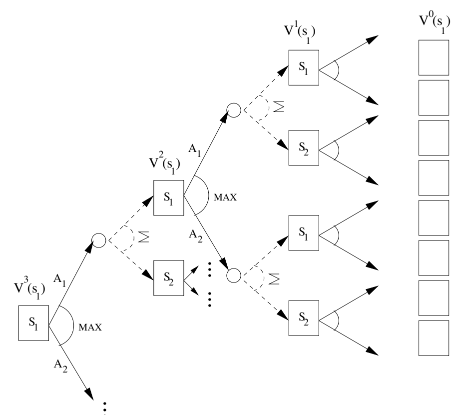




Bayesian RTDP

= more bang for your backup

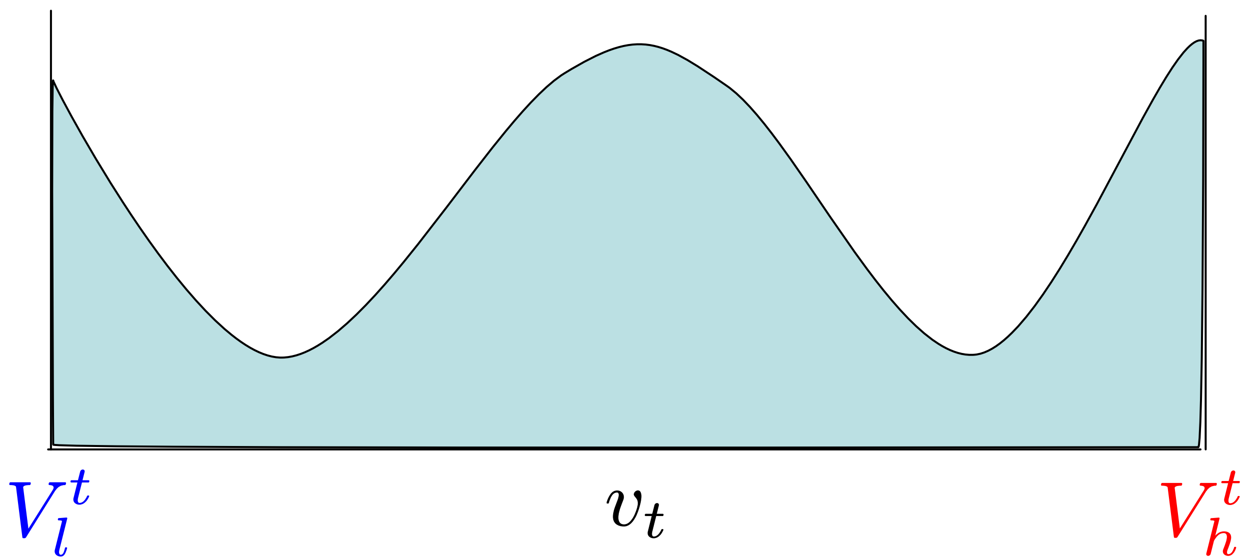
Thank you!
Any questions?



Additional Slides

Use Empirical Value Distribution?

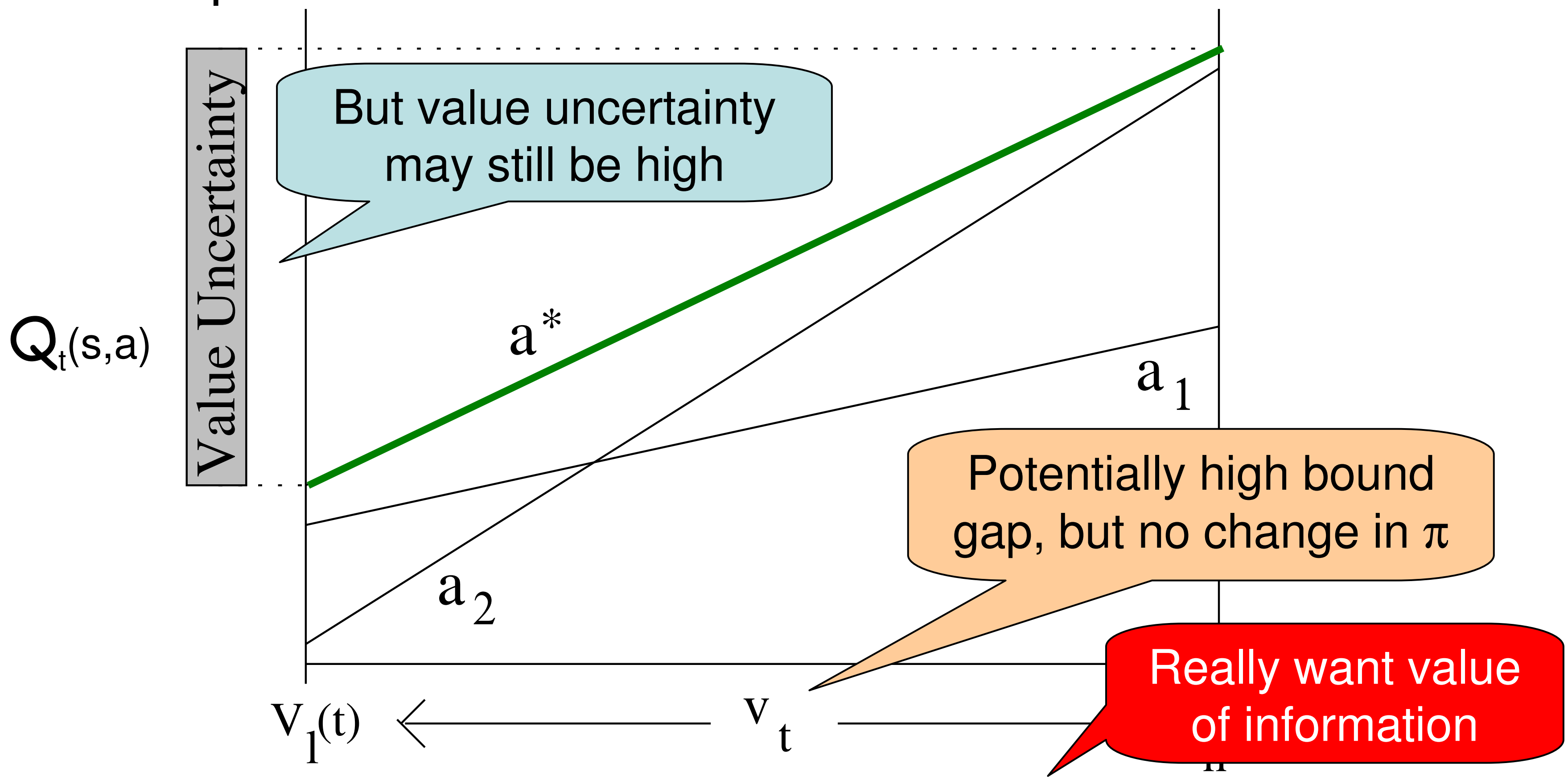
- Sketch of empirical distribution
  - Mixture of 3 normal distributions



- Distribution is changing over time
  - No single distribution seems to improve performance (I've spent a long time trying)

Drawback of *Bound Gap* Heuristic

- Bound gap: $V_h^0(s) - V_l^0(s)$ commonly used to prioritize search



- No point in reducing v_t uncertainty, from the perspective of s… it won't change the policy!

Aside

- ~~MDPs don't work~~
  - Don't confuse model with the solution!

many irrelevant states in these problems

- What researchers mean to imply is that…
  - "Heuristic search methods *often* outperform value or policy iteration in specific domains (e.g. PPDDL)"

- Async. DP offer best of both worlds (RTDP, LAO*)
  - **Convergence / optimality in limit!**
  - **Can apply search heuristics**