# Bayesian Q-Learning

## Richard Dearden, Nir Friedman, Stuart Russel

TECHNISCHE UNIVERSITÄT DARMSTADT



Pictures and tables taken from the paper, if not stated differently.

# Table of Contents

# Motivation and General Idea

TECHNISCHE UNIVERSITÄT DARMSTADT

## Motivation

We want a new approach to

- balance **exploration** against **exploitation** effectively
- **dynamically adapt** to current exploration benefit
- be at least as **well-performing** as conventional approaches

## General idea

The Approach presented here

- **extends** conventional Q-Learning by maintaining probability distributions over the Q-Values in a bayesian manner
- uses its own methods to choose the **next action** to be performed
- uses its own methods to **update the policy**

# Table of Contents

# Q-Learning: Concept

TECHNISCHE UNIVERSITÄT DARMSTADT

**Basis Structure** Markov Decision Process $(S, A, p_t, p_r)$, where

- $S$ is a set of **states**
- $A$ is a set of **actions**
- $p_t(s \xrightarrow{a} t)$ is the **transition model** capturing the probability of reaching state $t$ after executing action $a$ in state $s$
- $p_r(r|s, a)$ is the **reward model** capturing the probability of getting reward $r$ when executing action $a$ in state $s$

**Learning Agent**

- learns from its experience while **acting upon** and **perceiving**s its environment
- only has its **trajectory data** $D = (s_i, a_i, r_i, s_{i+1})_{i=1,\ldots,N}$ for timesteps $i$
  - **does not know** neither $p_t$ nor $p_r$
  - **does not learn** the model $p_t$

# Q-Learning: Evaluations

TECHNISCHE UNIVERSITÄT DARMSTADT

## Objective

- agent wants to **maximize the expected discounted total reward** $E[\sum_i \gamma^i r_i]$
  - $\gamma$ is the **discount factor** trading immediate against future reward $r$

## Evaluation Functions

- $V(s)$ Value function evluating **state** $s$
- $Q(s, a)$ Value function evaluating taking **action** $a$ in **state** $s$

## Bellman Equations

- $V^*(s) = \max_a Q^*(s, a)$
  - $V^*(s)$ is the optimal expected discounted reward achievable from **state** $s$
- $Q^*(s, a) = \sum_r r \cdot p_r(r|s, a) + \gamma \cdot \sum_t p_t(s \xrightarrow{a} t) V^*(t)$
  - $Q^*(s, a)$ is the optimal expected discounted reward achievable from **state** $s$ when executing **action** $a$
  - the knowledge **to be learnt**

# Q-Learning: Algorithm

TECHNISCHE UNIVERSITÄT DARMSTADT

## The Q-Learning algorithm*

1. Let the current state be $s$.
2. **Select an action** $a$ to perform.
3. Let the reward received for performing $a$ be $r$, and the resulting state be $t$
4. **Update** $Q(s, a)$ to reflect the observation $< s, a, r, t >$ as follows:
   $Q(s, a) = (1 - \alpha)Q(s, a) + \alpha(r + \gamma \max_{a'} Q(t, a'))$
   where $\alpha$ is the current learning rate.
5. Go to step 1.

*Taken from the Paper

## Convergence Properties

$Q(s, a)$ eventually converges to $Q^*(s, a)$ for all $s$ and $a$ if

- every action is **performed infinitely often** in every state and
- $\alpha$ is **decayed appropriately**

# Q-Learning:
# Action Selection — Exploration Methods

TECHNISCHE UNIVERSITÄT DARMSTADT

## Three Methods Presented:

- Semi-Uniform random exploration
  - the **best action** is selected with probability $p$ and and a **random action** is chosen with probability $(1 - p)$
- Blotzmann exploration
  - action $a$ is chosen with probability $Pr(a) = \frac{e^{Q(s,a)/T}}{\sum_{a'} e^{Q(s,a')/T}}$
- Interval Estimation
  - action is selected based on the **expected value of the action** plus an **exploration bonus**
  - applies **statistical methods** to determine bonus

# Table of Contents

# Bayesian Q-Learning: Concept

TECHNISCHE UNIVERSITÄT DARMSTADT

**What makes the approach Bayesian**

- considers **distributions over Q-Values** and updates these using bayesian methods
- uses a **random Variable** $R_{s,a}$ that denotes the *total discounted reward* received when action $a$ is executed in state $s$ and an optimal policy is followed thereafter
  - distribution of $R_{s,a}$ is to be learnt
  - distribution over its parameters is used

**Implementation** adapted Algorithm from Q-Learning

- but stores parameters for distribution of Q-Values instead of values for $Q_{s,a}$
- and **selects actions** and **updates estimates differently** in consequence

# Bayesian Q-Learning: Assumptions

TECHNISCHE UNIVERSITÄT DARMSTADT

**Assumption I**

$R_{s,a}$ has a **normal distribution** $N(\mu_{s,a}, \sigma^2_{s,a})$

- $R_{s,a}$ **decomposes** into the **discounted sum of immediate transition rewards**, each of wich is a random event
  - Thus, if $\gamma$ is close to 1, central limit theorem can be applied.
- in the following, a distribution over $\mu_{s,a}$ and $\tau_{s,a} = (\sigma^2_{s,a})^{-1}$ will be utilized
- $\mu_{s,a}$ corresponds to $Q(s, a)$

**Assumption II**

The prior distribution over $\mu_{s,a}$ and $\tau_{s,a}$ is **independent** of other prior distributions over $\mu_{s',a'}$ and $\tau_{s',a'}$ for $s \neq s'$ or $a' \neq a$.

- Implication: The prior beliefs about $R_{s,a}$ are independent of those about $R_{s',a'}$.

# Bayesian Q-Learning: Assumptions

TECHNISCHE UNIVERSITÄT DARMSTADT

**Assumption III**
The prior $p(\mu_{s,a}, \tau_{s,a})$ is a **normal-gamma** distribution.

- thus, only a tupel $< \mu_0^{s,a}, \lambda^{s,a}, \alpha^{s,a}, \beta^{s,a} >$ of **hyperparamters** needs to be maintained to represent the agent's prior over the distribution of $R_{s,a}$
- distribution can be **easily updated** by sampling

**Assumption IV**
At any stage, the agent's posterior over $\mu_{s,a}$ and $\tau_{s,a}$ is **independent** of the posterior over $\mu_{s',a'}$ and $\tau_{s',a'}$ for $s \neq s'$ or $a' \neq a$.

- This is most likely to be violated but used anyway.
  (The authors didn't state why.)

# Bayesian Q-Learning: Action Selection — Greedy Selection

TECHNISCHE UNIVERSITÄT DARMSTADT

## Action Selection

- in **each iteration** of the algorithm (before using that action to update the Q-Values)
- given a probability distribution over $Q(s, a) = \mu_{s,a}$
- **three approaches**: Greedy selection, Q-Value sampling, Myopic-VPI selection

## Greedy selection

- select action $a$ that **maximizes** $E[\mu_{s,a}]$
- generally a **bad idea**, since $E[\mu_{s,a}] = E[R_{s,a}]$
  - thus, would not attempt to perform **exploration**
  - does not take into account any **uncertainty** about the Q-Value

# Bayesian Q-Learning: Action Selection — Q-Value Sampling

TECHNISCHE UNIVERSITÄT DARMSTADT

## Q-Value Sampling

- **slect action stochastically**, based on the current subjective belief that it is optimal (similar to Boltzmann Exploration)
- Perform action $a$ **with probability of its optimality** in terms of reward

$$Pr(a = \underset{a'}{\text{argmax}}\, \mu_{s,a'}) = \int_{-\infty}^{\infty} Pr(\mu_{s,a} = q_a) \prod_{a' \neq a} Pr(\mu_{s,a'} < q_a)\, dq_a$$

- is **hard to calculate**, can in practice be avoided by sampling from $p(\mu_{s,a})$
  - Sample a value from each action using $p(\mu_{s,a})$ and select the action with the hightest sampled value
  - $p(\mu_{s,a})$ can be calculated from $p(\mu_{s,a}, \tau_{s,a})$

## Drawback

- **only considers the probability** that a is best action, and **does not consider the amount** by which choosing $a$ might improve over the current policy

# Bayesian Q-Learning: Action Selection — Myopic-VPI Selection

TECHNISCHE UNIVERSITÄT DARMSTADT

## Myopic-VPI Myopic-VPI Selection

- Method that **quantitavely considers** policy improvement through exploration
- **balances** the expected cost of doing a potentially suboptimal action against the gains from exploration

## Deduction

- There are **three Cases** thinkable when considering what can be gained by learning the true value $\mu^*_{s,a} = Q^*(s, a)$ of $\mu_{s,a} = Q(s, a)$
  1. if **knowledge does not change** the agent's policy the **rewards do not change**
  2. the new knowledge shows that an action **previously considered sub-optimal** is **revealed as the best** choice given the agent's beliefs about ohter actions
  3. the new knowledge indicates that an action that was **previously considered best** is **actually inferior** to other actions

# Bayesian Q-Learning: Action Selection — Myopic-VPI Selection

TECHNISCHE UNIVERSITÄT DARMSTADT

**Gain from learning the true value $\mu^*_{s,a}$ of $\mu_{s,a}$**

$a_1$ denotes the action currently deemed best, $a_2$ the second best respectively

$$Gain_{s,a}(\mu^*_{s,a}) := \begin{cases} E[\mu_{s,a_2}] - \mu^*_{s,a} & \text{if } a = a_1 \text{ and } \mu^*_{s,a} < E[\mu_{s,a_2}] \text{ (case 3)} \\ \mu^*_{s,a} - E[\mu_{s,a_1}] & \text{if } a \neq a_1 \text{ and } \mu^*_{s,a} > E[\mu_{s,a_2}] \text{ (case 2)} \\ 0 & \text{otherwise (case 1)} \end{cases}$$

**Expected Value of Perfect Information about $\mu_{s,a}$** (Expected Gain)

$$VPI(s,a) = \int_{-\infty}^{\infty} Gain_{s,a}(x) \cdot Pr(\mu_{s,a} = x)\, dx$$

- needed because $\mu^*_{s,a}$ is **not known in advance**
- can be reduced to a **closed-form equation** that is **efficiently computable**
- gives an **upper bound** on the myopic value of exploring action a

# Bayesian Q-Learning: Action Selection — Myopic-VPI Selection

TECHNISCHE UNIVERSITÄT DARMSTADT

## Expected VPI in closed form

$$VPI(s,a) = \begin{cases} c + (E[\mu_{s,a_2}] - E[\mu_{s,a_1}]) \cdot Pr(\mu_{s,a_1} < E[\mu_{s,a_2}]) & \text{if } a = a_1 \\ c + (E[\mu_{s,a}] - E[\mu_{s,a_1}]) \cdot Pr(\mu_{s,a} < E[\mu_{s,a_1}]) & \text{if } a \neq a_1 \end{cases}$$

$$\text{where } c = \frac{\alpha_{s,a}\Gamma(\alpha_{s,a} + \frac{1}{2})\sqrt{\beta_{s,a}}}{(\alpha_{s,a} - \frac{1}{2})\Gamma(\alpha_{s,a})\Gamma(\frac{1}{2})\alpha_{s,a}\sqrt{2\lambda_{s,a}}} \left(1 + \frac{E^2[\mu_{s,a}]}{2\alpha_{s,a}}\right)^{-\alpha_{s,a}+\frac{1}{2}}$$

## Actual Action Selection

- Take action: $\text{argmax}_a\, E[Q(s,a)] + VPI(s,a)$
- when the agent is confident of the estimated Q-Values, $VPI(s,a)$ is close to 0

# Bayesian Q-Learning: Updating Q-Values

TECHNISCHE UNIVERSITÄT DARMSTADT

**Task**

- **update the estimate** of the distribution over Q-Values $R_{s,a}$ after transitioning from a state $s$ to a state $t$ executing action $a$ and receiving reward $r$

**Problem**

- $R_{s,a}$ is distribution is over **expected total rewards**, observations are instances of **actual local rewards**
- future reward is unknown

**Solution**

- Use a **random variable** $R_t$ denoting the discounted sum of rewards from $t$ on and use it as a substitute
- $R_t$ **is distributed as** $R_{t,a_t}$ with $a_t$ being the action with the highest expected value at $t$

**Means**

- two approaches presented: **Moment updating** and **Mixture updating**

# Bayesian Q-Learning: Updating Q-Values — Moment Updating

TECHNISCHE UNIVERSITÄT DARMSTADT

## Idea

- **randomly sample** $n$ values $R_t^1, ..., R_t^n$ from $R_t$
- then, **update** $P(R_{s,a})$ using $r + \gamma R_t^1 + ... + \gamma R_t^n$ as observation $R$
  - as substitute for the total discunted future reward

## Update Formula

- Calculate the **posterior** $p(\mu, \tau|R) \sim NG(\mu_0', \lambda', \alpha', \beta')$
- given the **prior** $p(\mu, \tau) \sim NG(\mu_0, \lambda, \alpha, \beta)$
- where $\mu_0' = \frac{\lambda\mu_0 + M_1}{\lambda+1}, \lambda' = \lambda + 1, \alpha' = \alpha + \frac{1}{2}, \beta' = \beta + \frac{1}{2}(M_2 - M_1^2) + \frac{\lambda(M_1 - \mu_0)^2}{2(\lambda+1)}$
- with **moments**
  $M_1 = E[r + \gamma R_t] = r + \gamma E[R_t] M_2 = E[(r + \gamma R_t)^2] = r^2 + 2\gamma r E[R_t] + \gamma^2 E[R_t^2]$

## Properties

- is **easily computable** in a simple closed-form equation
- but may experience **premature convergence**

# Bayesian Q-Learning: Updating Q-Values — Mixture Updating

TECHNISCHE UNIVERSITÄT DARMSTADT

## Method

- let $p(\mu_{s,a}, \tau_{s,a}|R)$ posterior distribution over $\mu_{s,a}, \tau_{s,a}$ after observing reward R
- the **updated distribution** over $R_{s,a}$ is $p(\mu_{s,a}, \tau_{s,a}|r + \gamma x)$, if $R_t = x$,
- since $R_t = x$ is not known in advance, the **expected future discounted reward** ("mixture posterior") is calculated

$$p_{r,t}^{mix}(\mu_{s,a}, \tau_{s,a}) = \int_{-\infty}^{\infty} p(\mu_{s,a}, \tau_{s,a}|r + \gamma x) \cdot p(R_t = x)\, dx$$

## Properties

- **does not have a closed-form** representation and is therefore **approximated**
  - using **normal-gamma distribution** after each update (Assumption III)
  - achieved **minimzing the Kullback-Leibler Divergence** from the approximation to the true distribution (non-symmetric)
  - employs **numerical methods** for optimization and integration
- **does not cause premature convergence** like Moment Updating

# Bayesian Q-Learning: Convergence

TECHNISCHE UNIVERSITÄT DARMSTADT

**Moment Updating and Q-Value sampling**

- **converges**, because
  - **mean converges** to the true Q-Value for every state-action
  - **variance converges** to 0

**Moment Updating and myopic-VPI**

- **no guarantees**, since myopic-VPI might not let all actions be executed infinitely often
- but **can be modified** to converge like with Q-Value sampling by making the action selection **noisy** (as in Boltzmann Exploration)

**Mixture updating** and any action selection

- at the time of writing the paper, **no statement** could be made
- the **authors assume** convergence at least for the mean

# Table of Contents

# Experiments: Setup

TECHNISCHE UNIVERSITÄT DARMSTADT

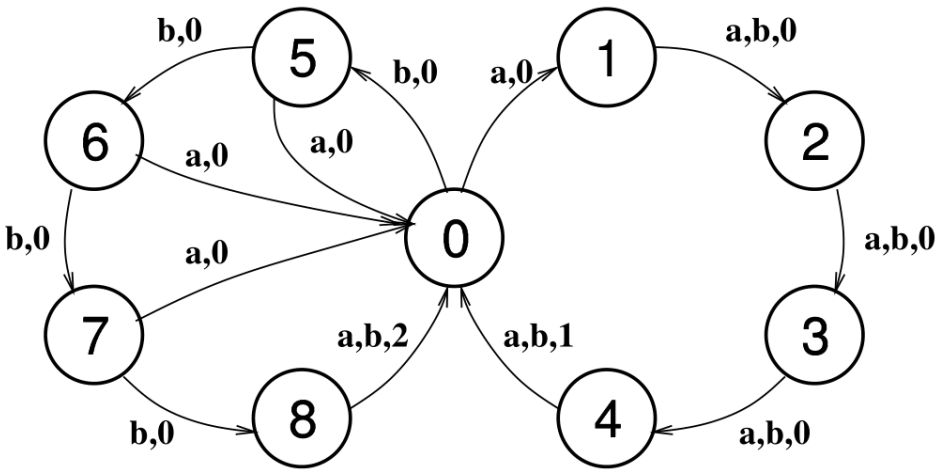
**Algortithms compared:**

- **Semi-Uniform** Q-Learning with sem-uniform random exploration
- **Boltzmann** Q-Learning with Boltzmann exploration
- **Interval** Q-Learning using the interval estimation algorithm
- **IEQL+** Meuleau's IEQL+ algorithm
- **Bayes** Byaesian Q-Learning, one for each combination out of the presented *Q-Value sampling* or *myopic-VPI* and *moment updating* or *mixture updating*

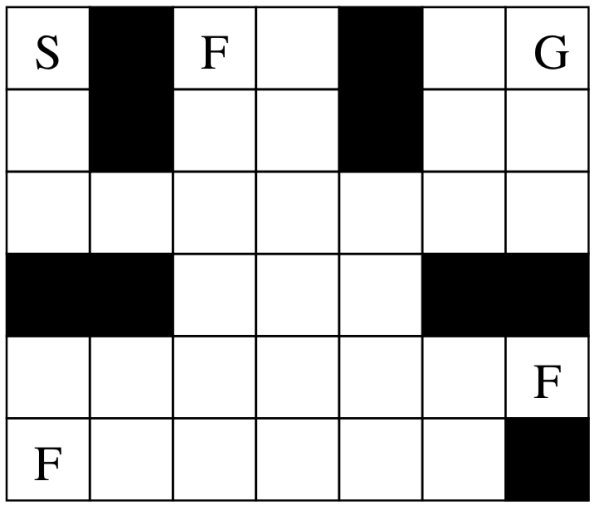
**Domains tested on**

- **Chain** Chain of six states with two actions, with probability 0.2 agent performs the other action accidentally, optimal policy is *a* everywhere
- **Loop** Two loops os states with deterministic actions and convergence trap
- **Maze** Flag collection Task in a grid world, 264 states, 0.1 probablity that the agent goes in a different direction than intended; used for scaling evaluation

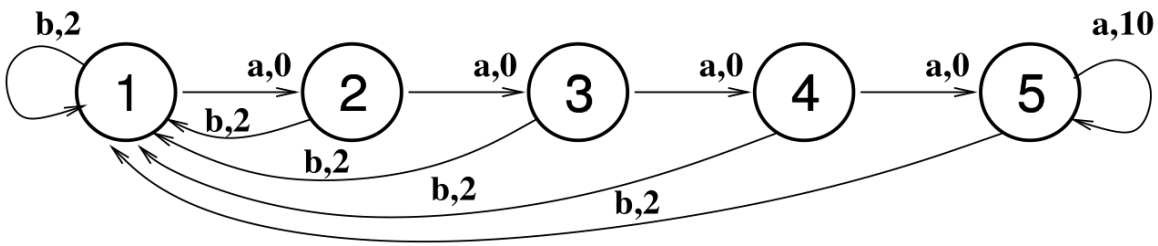
# Experiments: Setup — Domains




Loop Domain

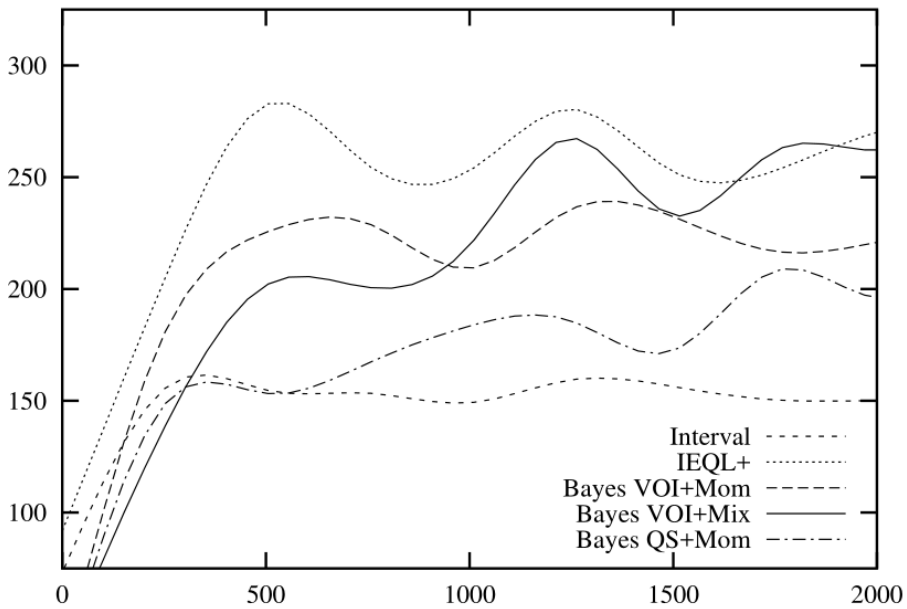



Maze Domain

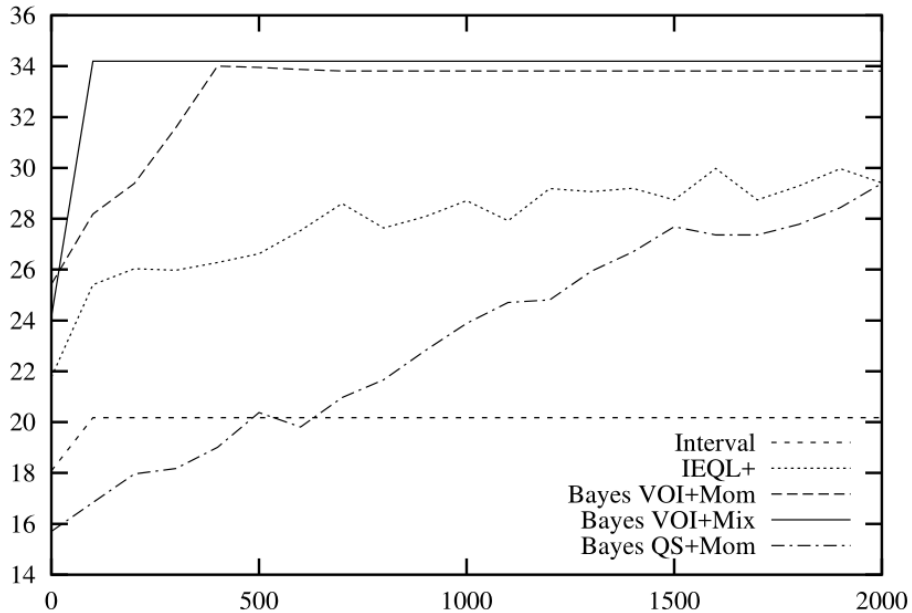



Chain Domain

# Experiments: Results



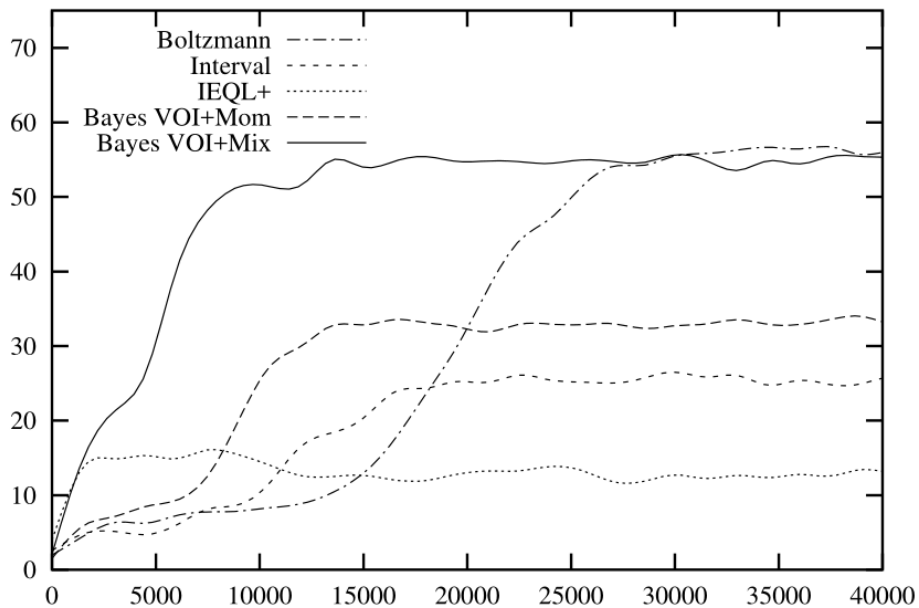

Results on chain domain




Results on loop domain

- x axis: number of steps
- y axis: actual total discounted reward
- curves averaged over 10 runs

# Experiments: Results




Results on Maze domain

- VOI+MIX consistently one of the best

| Domain | Method | 1st Phase Avg. | 1st Phase Dev. | 2nd Phase Avg. | 2nd Phase Dev. |
|---|---|---|---|---|---|
| chain | Uniform | 1519.0 | 37.2 | 1611.4 | 34.7 |
| | Boltzmann | 1605.8 | 78.1 | 1623.4 | 67.1 |
| | Interval | 1522.8 | 180.2 | 1542.6 | 197.5 |
| | IEQL+ | 2343.6 | 234.4 | 2557.4 | 271.3 |
| | Bayes QS+Mom | 1480.8 | 206.3 | 1894.2 | 364.7 |
| | Bayes QS+Mix | 1210.0 | 86.1 | 1306.6 | 102.0 |
| | Bayes VPI+Mom | 1875.4 | 478.7 | 2234.0 | 443.9 |
| | Bayes VPI+Mix | 1697.4 | 336.2 | 2417.2 | 650.1 |
| loop | Uniform | 185.6 | 3.7 | 198.3 | 1.4 |
| | Boltzmann | 186.0 | 2.8 | 200.0 | 0.0 |
| | Interval | 198.1 | 1.4 | 200.0 | 0.0 |
| | IEQL+ | 264.3 | 1.6 | 292.8 | 1.3 |
| | Bayes QS+Mom | 190.0 | 19.6 | 262.9 | 51.4 |
| | Bayes QS+Mix | 203.9 | 72.2 | 236.5 | 84.1 |
| | Bayes VPI+Mom | 316.8 | 74.2 | 340.0 | 91.7 |
| | Bayes VPI+Mix | 326.4 | 85.2 | 340.0 | 91.7 |
| maze | Uniform | 105.3 | 10.3 | 161.2 | 8.6 |
| | Boltzmann | 195.2 | 61.4 | 1024.3 | 87.9 |
| | Interval | 246.0 | 122.5 | 506.1 | 315.1 |
| | IEQL+ | 269.4 | 3.0 | 253.1 | 7.3 |
| | Bayes QS+Mom | 132.9 | 10.7 | 176.1 | 12.2 |
| | Bayes QS+Mix | 128.1 | 11.0 | 121.9 | 9.9 |
| | Bayes VPI+Mom | 403.2 | 248.9 | 660.0 | 487.5 |
| | Bayes VPI+Mix | 817.6 | 101.8 | 1099.5 | 134.9 |

Average and standard deviation of accumulated rewards over 10 runs, a phase consists of 1,000 steps in chain and loop and of 20,000 steps in maze

# Table of Contents

# Conclusion and Remarks

TECHNISCHE UNIVERSITÄT DARMSTADT

**Theirs**

according to the experiment results, for Bayesian Q-Learning it holds that

- with Q-Value Sampling and myopic-VPI the state **space is explored more effectively** than with conventional model-free Q-Learners
- **performance advantage** appears to increase as the problems become **larger**
- computational requirements are higher, though

**Mine**

Critique on the Paper

- cliffhanger with Assumption IV
- mathematical assumptions not justified sufficiently (for my abilities)
- performance tests of mixture updating done but lack of convergence might spoil generalization capabilites of the results

TECHNISCHE
UNIVERSITÄT
DARMSTADT

# Thank you for listening