POMDP

$$T(s,a,s') := Pr(s_t = s' \mid s_{t-1} = s, a_{t-1} = a) \ \forall t,$$

$$O(s,a,z) := Pr(z_t = z \mid s_{t-1} = s, a_{t-1} = a) \ \forall t,$$

- a set of states $S = \{s_1, s_2, \ldots s_{|S|}\}$
- a set of actions $\mathcal{A} = \{a_1, a_2, \ldots, a_{|\mathcal{A}|}\}$
- a set of observations $Z = \{z_1, z_2, \ldots, z_{|Z|}\}$
- a set of transition probabilities $T(s_i, a, s_j) = p(s_j|s_i, a)$
- a set of observation probabilities $O(z_i, a, s_j) = p(z_i|s_j, a)$
- a set of rewards $R : S \times \mathcal{A} \mapsto \mathbb{R}$
- a discount factor $\gamma \in [0, 1]$
- an initial belief $p_0(s)$

We maximize the expected cummulated reward

$$E[\sum_{t=t_0}^{T} \gamma^{t-t_0} r_t],$$

Monty Hall, Tygr

- 3 dveře, za jedněmi zlato
  - ukážete jedny
  - Monty otevře jedny ze zbývajících
  - vy volíte: stejné, jiné, nebo je to jedno?

- Tygr:
  - dvoje dveře, tygr a zlato
  - můžete poslouchat: sluch nepřesný a zlata ubývá
  - jakou zvolíte strategii?

Tygr

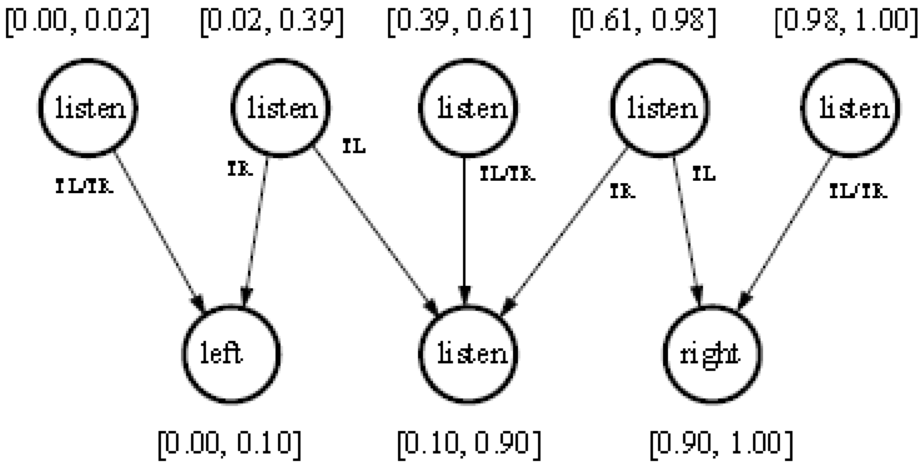
- Akce: left, right, listen
- Užitek:
  - U(l/r,zlato)=10, U(l/r,tiger)=-100, U(listen,*)=-1
- P(Listen|S): 0.85 correctly, 0.15 opposit; TR, TL
- Svět: 0.5 tiger left; po otevření náhodný reset.

Konečný horizont

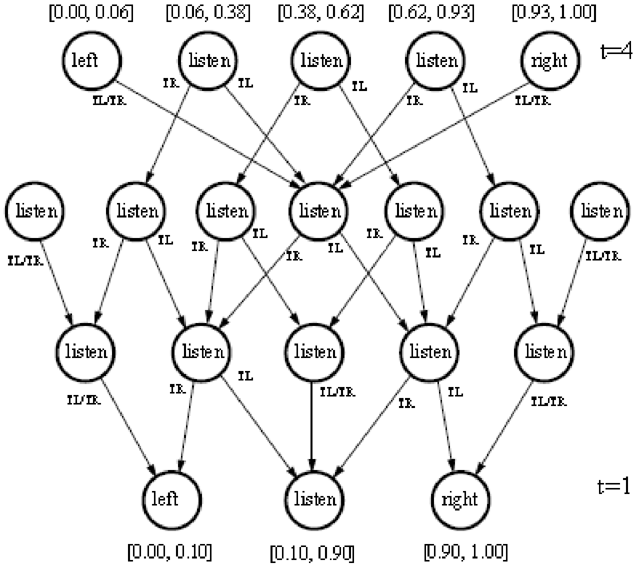
- t=1: $EU_{t=1}$(A=left/right)=(-100+10)/2=-45

Tiger – horizont 2, horizont 4

-
-
-

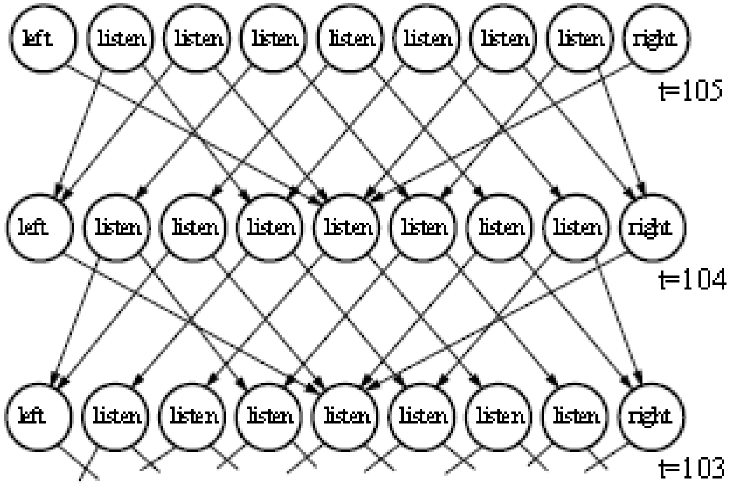


- víc „policy trees“ se stejnou akcí v kořeni



Tigr – nekonečný horizont

- γ=0.75
  - konvergence
  - policy graf obecný
  - dosažitelné z 0.5,0.5



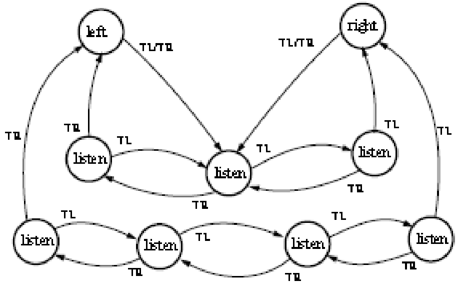


Figure 16: Policy graph for tiger example

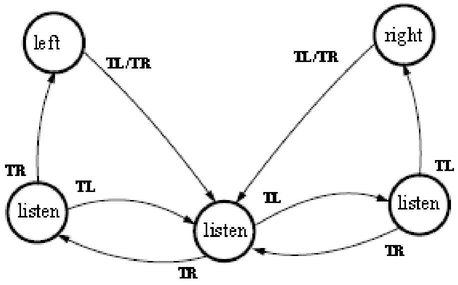
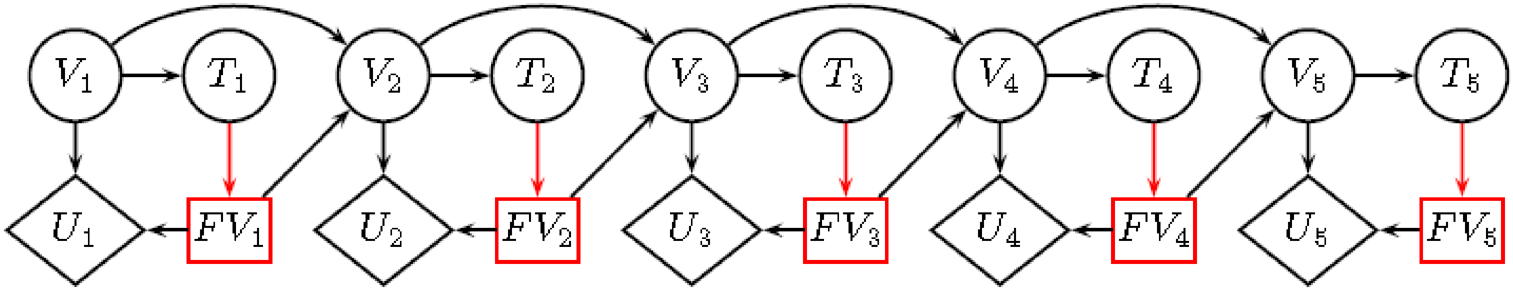


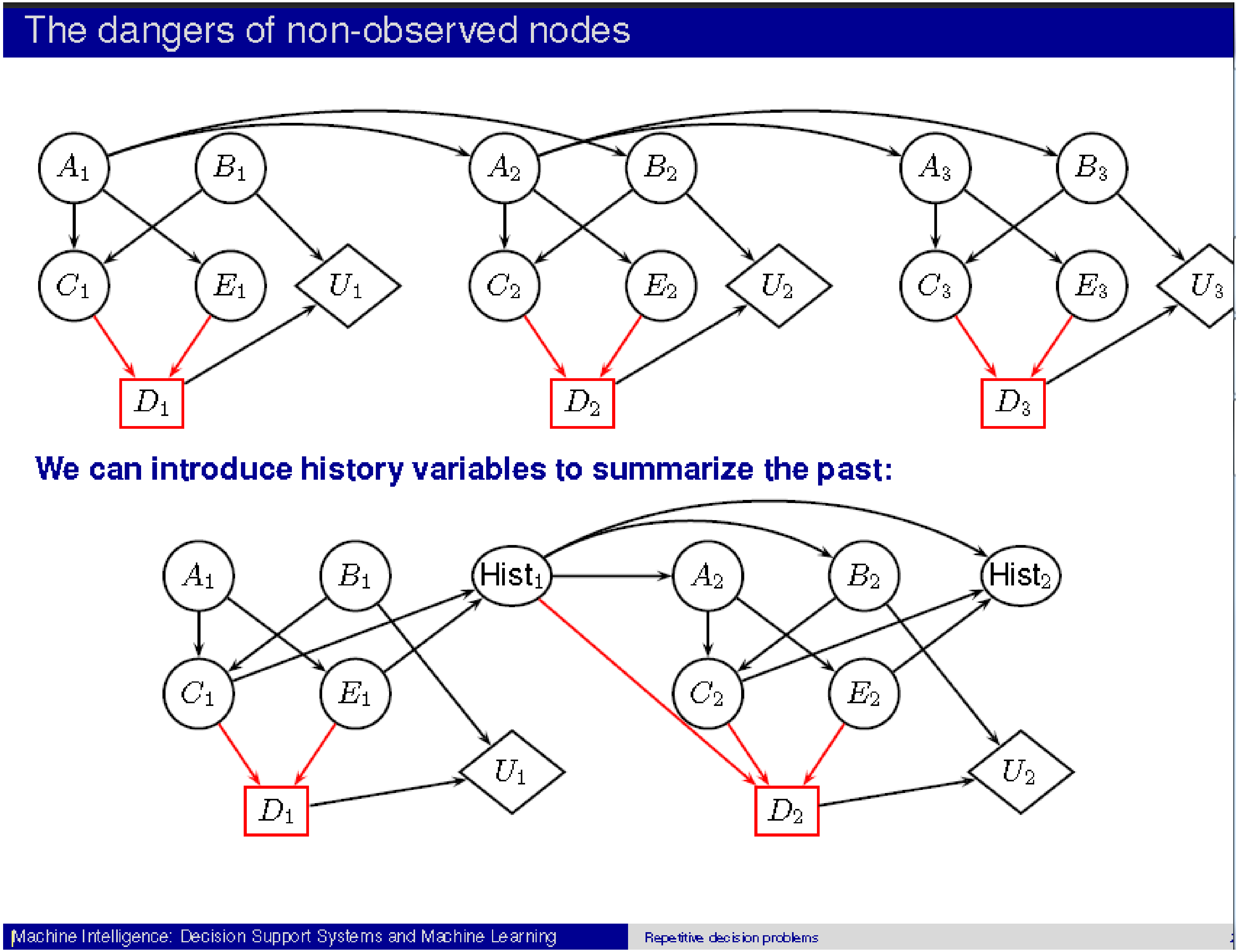
Figure 17: Trimmed policy graph for tiger example

Markovský?



- The underlying dynamics of the POMDP are still Markovian,
- we have no direct access to the current state,
- our decisions require keeping track of (possibly) the entire history of the process, making this a non-Markovian process.
  - The history at a given point in time is comprised of our knowledge about our starting situation, all actions performed and all observations seen.

Agregovat historii v jednom uzlu



The dangers of non-observed nodes

We can introduce history variables to summarize the past:

Markovský vzhledem k „Belief stavů“

- It turns out that simply maintaining a probability distribution over all of the states
  - provides us with the same information as if we maintained the complete history.
- When we perform and action and make an observation, we have to update the distribution.
  - Updating the distribution is very easy and just involves using the transition and observation probabilities.

Belief b(s) summarizes the history

- We want to optimize:
- our history is $h_t \;:=\; \{a_0, z_1, \ldots, z_{t-1}, a_{t-1}, z_t\}$
- belief: $b_t(s) \;:=\; Pr(s_t = s \,|\, z_t, a_{t-1}, z_{t-1}, \ldots, a_0, b_0).$
- initial: $b_0(s) \;:=\; Pr(s_0 = s),$
- belief update:

$$\begin{aligned} \tau(b_{t-1}, a_{t-1}, z_t) &= b_t(s') \\ &= \frac{\sum_{s'} O(s', a_{t-1}, z_t)\, T(s, a_{t-1}, s')\, b_{t-1}(s)}{Pr(z_t | b_{t-1}, a_{t-1})} \end{aligned}$$

- Markovian in b ....τ() invariant in time.

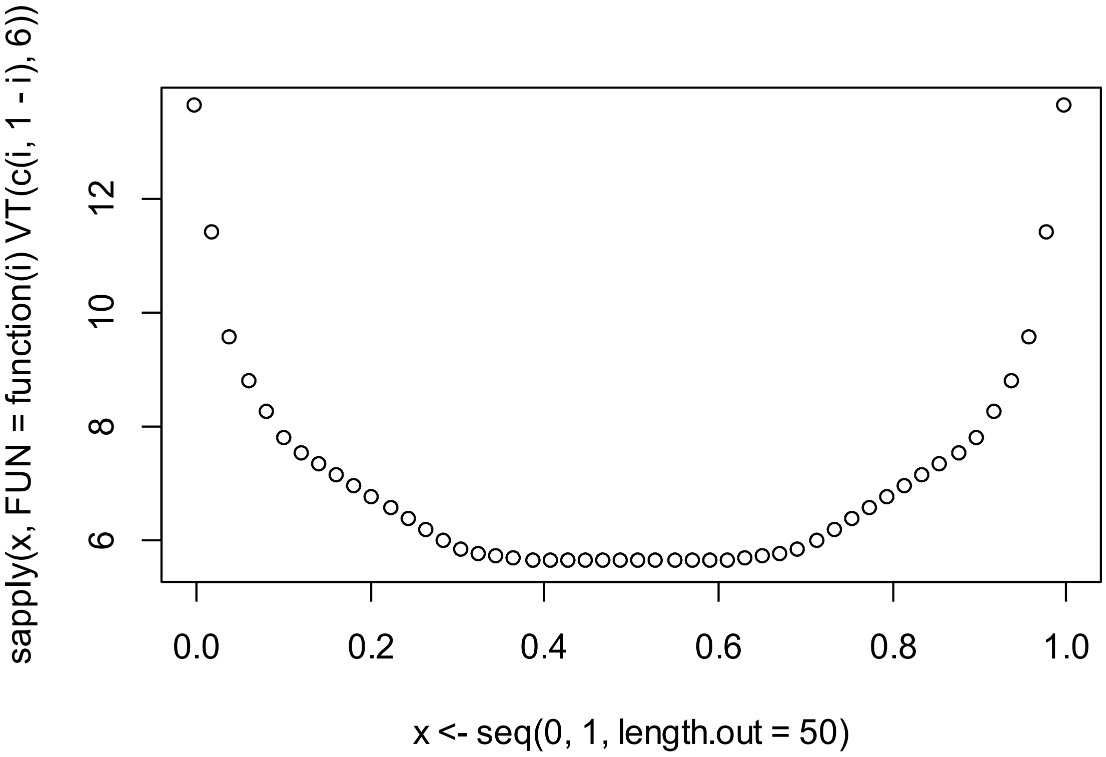
Policy, value function

- policy is a function $\pi(b) \longrightarrow a,$
- optimal policy $\pi^*(b_{t_0}) = \operatorname*{argmax}_{\pi} E_\pi \left[\sum_{t=t_0}^{T} \gamma^{t-t_0} r_t \middle| b_{t_0} \right]$
- value function:
- initial $V_0(b) = \max_a \sum_{s\in S} R(s,a)b(s),$
- recursive $V_t(b) = \max_a \left[\sum_{s\in S} R(s,a)b(s) + \gamma \sum_{z\in Z} Pr(z \mid a,b) V_{t-1}(\tau(b,a,z)) \right],$
- optimal policy for the horizon t:

$$\pi_t^*(b) = \operatorname*{argmax}_a \left[\sum_{s\in S} R(s,a)b(s) + \gamma \sum_{z\in Z} Pr(z \mid a,b) V_{t-1}(\tau(b,a,z)) \right]$$

R-code

- plot(x<-seq(0,1,length.out=50),
    - sapply(x,FUN=function(i)VT(c(i,1-i),6)))



Úkoly

- Belief [0.7,0.3] spočtěte akci a užitek (eu)
  - Totéž pro [0.001,0.999]
- Pro belief výše spočtěte nový belief, pokud jste slyšeli tygra vpravo. (querygrain)
- Určete hodnotu belief výše pro horizont 5 (VT)
  - Můžete zkusit jiný discount (gamma)
- Zobrazte (iteračně) VT pro belief z <0,1>.

α vectors and the value function V

- two states in S

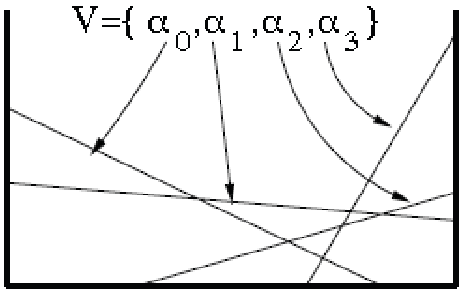
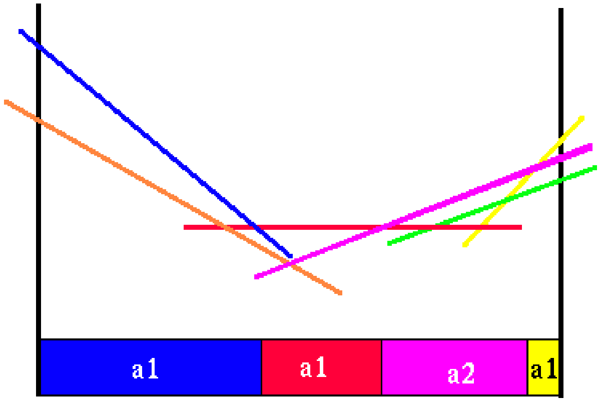
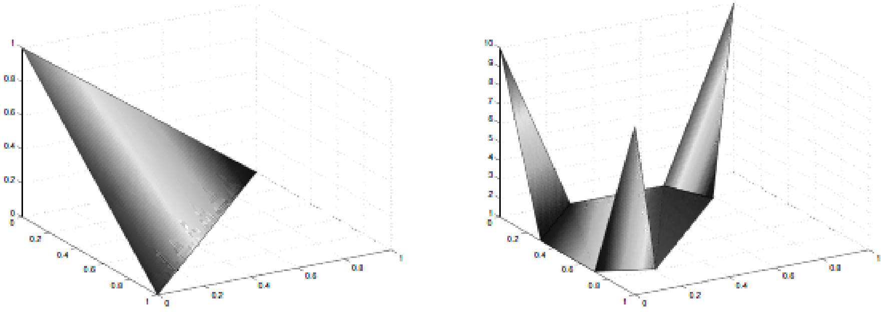


Figure 1: POMDP value function representation



- three states in S



(a) The belief space

(b) The value function

α vector represents a hyperplane

- value function at any finite horizon t can be expressed by a set of vectors: $\Gamma_t = \{\alpha_0, \alpha_1, \dots, \alpha_m\}.$

$$V_t(b) = \max_{\alpha \in \Gamma_t} \sum_{s \in S} \alpha(s) b(s).$$

$$V_0(b) = \max_{a} \sum_{s \in S} R(s,a) b(s),$$

- from the recursive formula earlier $V_t(b) = \max_{a} \left[\sum_{s \in S} R(s,a) b(s) + \gamma \sum_{z \in Z} Pr(z \mid a, b) V_{t-1}(\tau(b,a,z)) \right],$
- we get:

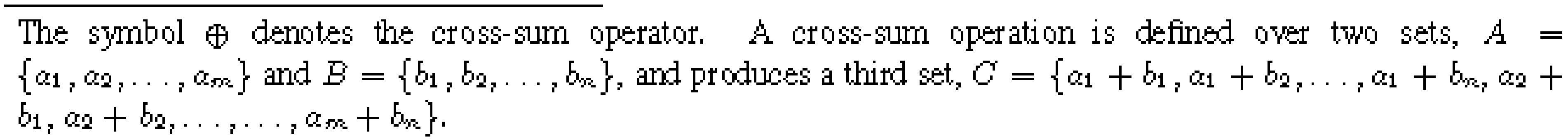
$$V_t(b) = \max_{a \in A} \left[\sum_{s \in S} R(s,a) b(s) + \gamma \sum_{z \in Z} \max_{\alpha \in \Gamma_{t-1}} \sum_{s \in S} \sum_{s' \in S} T(s,a,s') O(s',a,z) \alpha(s') b(s) \right].$$

$$|\Gamma_t| = O(|A||\Gamma_{t-1}|^{|Z|})$$

$$\begin{aligned} \tau(b_{t-1}, a_{t-1}, z_t) &= b_t(s') \\ &= \frac{\sum_{s'} O(s', a_{t-1}, z_t)\, T(s, a_{t-1}, s')\, b_{t-1}(s)}{Pr(z_t | b_{t-1}, a_{t-1})} \end{aligned}$$

One step of iteration

- intermediate sets
- $$\begin{aligned}\Gamma_t^{a,*} &\leftarrow \alpha^{a,*}(s) = R(s,a)\\ \Gamma_t^{a,z} &\leftarrow \alpha_i^{a,z}(s) = \gamma \sum_{s' \in S} T(s,a,s')O(s',a,z)\alpha_i(s'), \forall \alpha_i \in \Gamma_{t-1}\end{aligned}$$
- action reward plus cross-sum of observations
- $$\Gamma_t^a = \Gamma_t^{a,*} + \Gamma_t^{a,z_1} \oplus \Gamma_t^{a,z_2} \oplus \dots$$



The symbol $\oplus$ denotes the cross-sum operator. A cross-sum operation is defined over two sets, $A = \{\alpha_1, \alpha_2, \dots, \alpha_m\}$ and $B = \{b_1, b_2, \dots, b_n\}$, and produces a third set, $C = \{\alpha_1 + b_1, \alpha_1 + b_2, \dots, \alpha_1 + b_n, \alpha_2 + b_1, \alpha_2 + b_2, \dots, \dots, \alpha_m + b_n\}$.

-
- the new set $\Gamma_t = \cup_{a \in A} \Gamma_t^a$.
- Remove vectors that are dominated by others.

Approximation – only some b-points

- Select only a finite number of belief points
- For each, (at most) one α vector

$$\begin{aligned}
\Gamma_t^{a,*} &\leftarrow \alpha^{a,*}(s) = R(s,a) \\
\Gamma_t^{a,z} &\leftarrow \alpha_i^{a,z}(s) = \gamma \sum_{s' \in S} T(s,a,s')O(s',a,z)\alpha_i(s'), \forall \alpha_i \in \Gamma_{t-1}
\end{aligned}$$

- Max for FINITE number of b in B

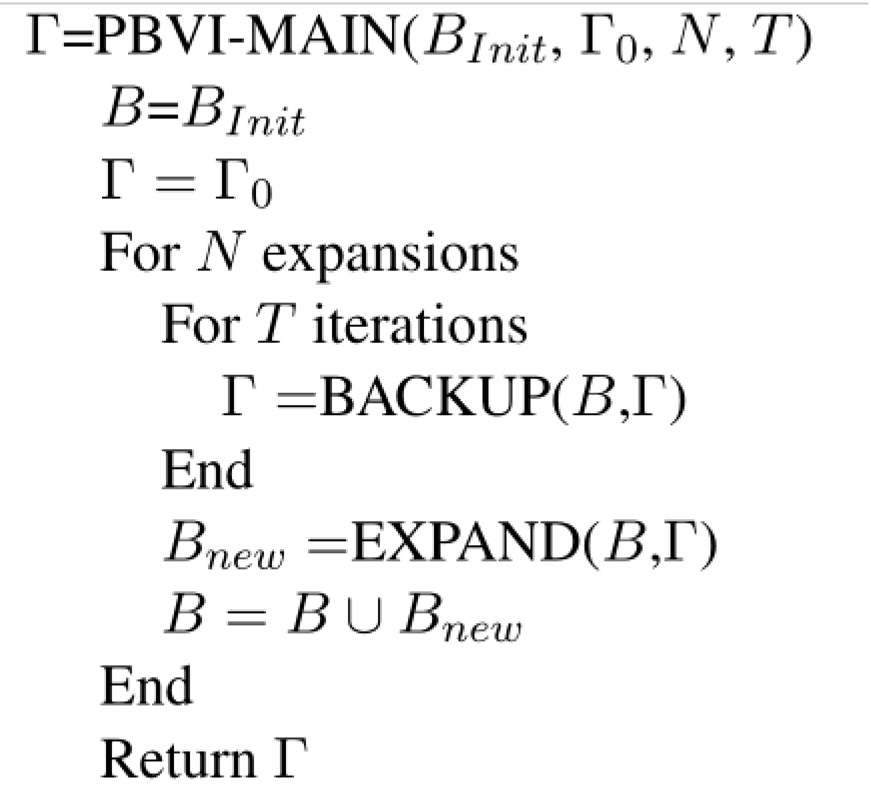
$$\begin{aligned}
\Gamma_t^a &\leftarrow \alpha_b^a = \Gamma_t^{a,*} + \sum_{z \in Z} \operatorname*{argmax}_{\alpha \in \Gamma_t^{a,z}} (\sum_{s \in S} \alpha(s)b(s)), \forall b \in B \\
\alpha_b &= \operatorname*{argmax}_{\Gamma_t^a, \forall a \in A} (\sum_{s \in S} \Gamma_t^a(s)b(s)), \quad \forall b \in B \\
\Gamma_t &= \cup_{b \in B} \; \alpha_b
\end{aligned}$$

- α vector count is not increasing (wrt. size of B)!

BACKUP – eval POMDP for a fixed set B

```
$\Gamma_t$=BACKUP($B$, $\Gamma_{t-1}$)
    For each action $a \in A$
        For each observation $z \in Z$
            For each solution vector $\alpha_i \in \Gamma_{t-1}$
                $\alpha_i^{a,z}(s) = \gamma \sum_{s' \in S} T(s,a,s') O(s',a,z) \alpha_i(s'), \forall s \in S$
            End
            $\Gamma_t^{a,z} = \cup_i \; \alpha_i^{a,z}$
        End
    End
    $\Gamma_t = \emptyset$
    For each belief point $b \in B$
        $\alpha_b = \text{argmax}_{a \in A} \left[ \sum_{s \in S} R(s,a) b(s) + \sum_{z \in Z} \max_{\alpha \in \Gamma_t^{a,z}} \left[ \sum_{s \in S} \alpha(s) b(s) \right] \right]$
        If($\alpha_b \notin \Gamma_t$)
            $\Gamma_t = \Gamma_t \cup \alpha_b$
    End
    Return $\Gamma_t$
```

Iterative expansion of B



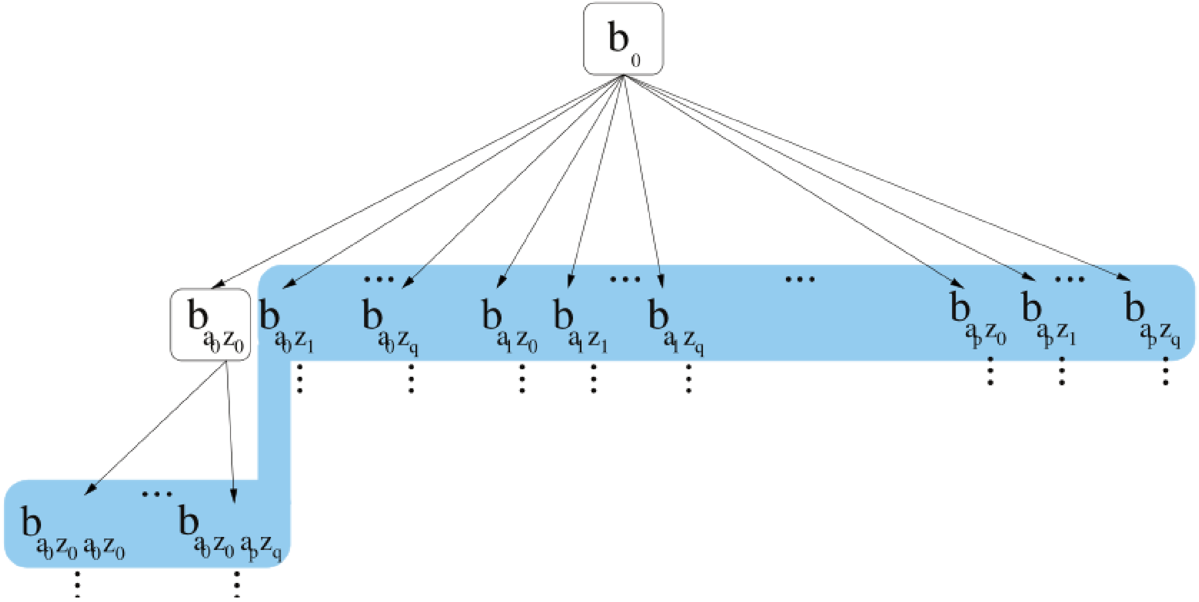
```
$\Gamma$=PBVI-MAIN($B_{Init}$, $\Gamma_0$, $N$, $T$)
    $B$=$B_{Init}$
    $\Gamma = \Gamma_0$
    For $N$ expansions
        For $T$ iterations
            $\Gamma$ =BACKUP($B$,$\Gamma$)
        End
        $B_{new}$ =EXPAND($B$,$\Gamma$)
        $B = B \cup B_{new}$
    End
    Return $\Gamma$
```

- T – either horizon or choose to bound the error

$$\gamma^t \|V_0^* - V^*\|$$

Belief set expansion

- Randomly
-
- Greedy Error Reduction



Maximal error candidate

- b' new candidate, b closest element of B
  - Error in any belief b'

$$\epsilon(b') \le \min_{b\in B} \sum_{s\in S} \begin{cases} (\frac{R_{\max}}{1-\gamma} - \alpha(s))(b'(s) - b(s)) & b'(s) \ge b(s) \\ (\frac{R_{\min}}{1-\gamma} - \alpha(s))(b'(s) - b(s)) & b'(s) < b(s) \end{cases}$$

- b on the fringe, error of tau(b,a,z) as above

$$\begin{aligned} \epsilon(b) &= \max_{a\in A} \sum_{z\in Z} O(b,a,z)\ \ \epsilon(\tau(b,a,z)) \\ &= \max_{a\in A} \sum_{z\in Z} \left(\sum_{s\in S} \sum_{s'\in S} T(s,a,s')O(s',a,z)b(s) \right) \epsilon(\tau(b,a,z)) \end{aligned}$$

Journal of Artificial Intelligence Research 27 (2006) 335-380 Submitted 05/06; published 11/06

Anytime Point-Based Approximations for Large POMDPs

Joelle Pineau JPINEAU@CS.MCGILL.CA
School of Computer Science
McGill University
Montréal QC, H3A 2A7 CANADA

Geoffrey Gordon GGORDON@CS.CMU.EDU
Machine Learning Department
Carnegie Mellon University
Pittsburgh PA, 15232 USA

Sebastian Thrun THRUN@STANFORD.EDU
Computer Science Department
Stanford University
Stanford CA, 94305 USA

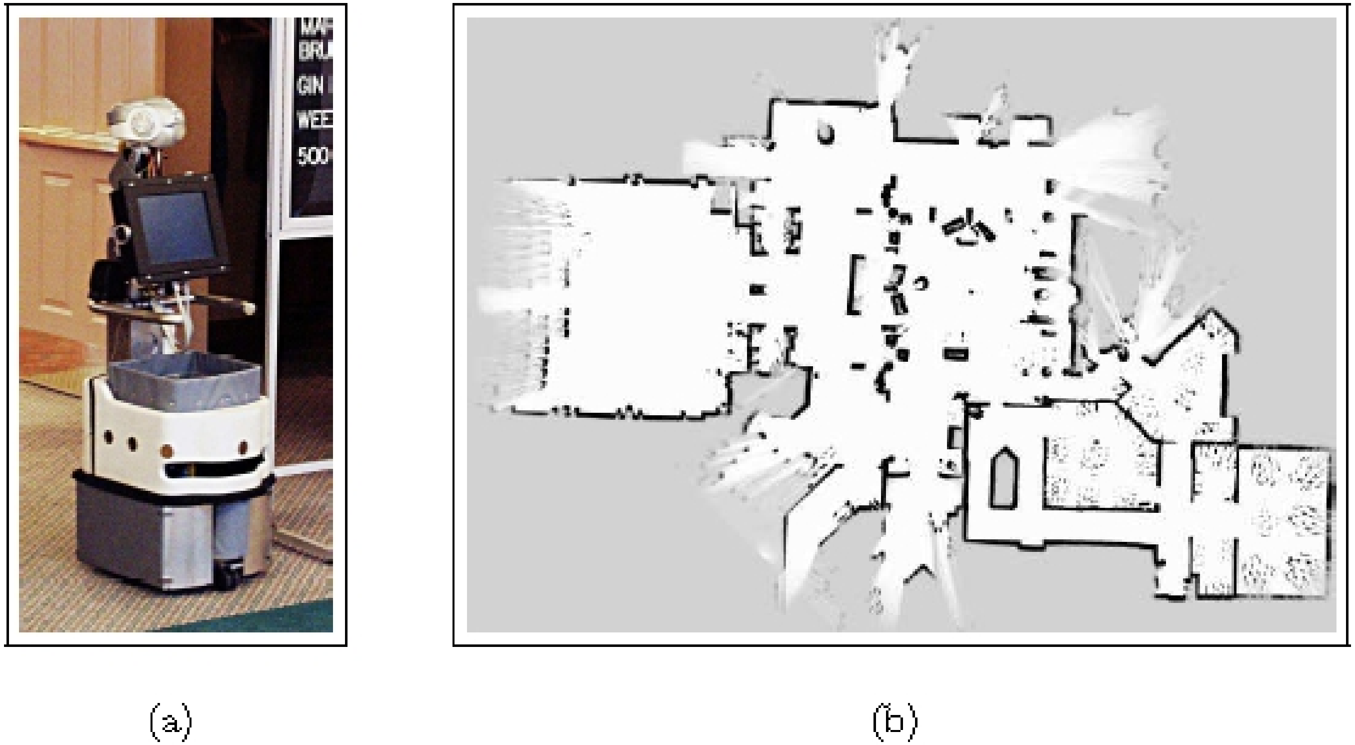
Journal of Artificial Intelligence Research 23 (2005) 1-40 Submitted 05/04; published 01/05

Finding Approximate POMDP Solutions Through Belief Compression

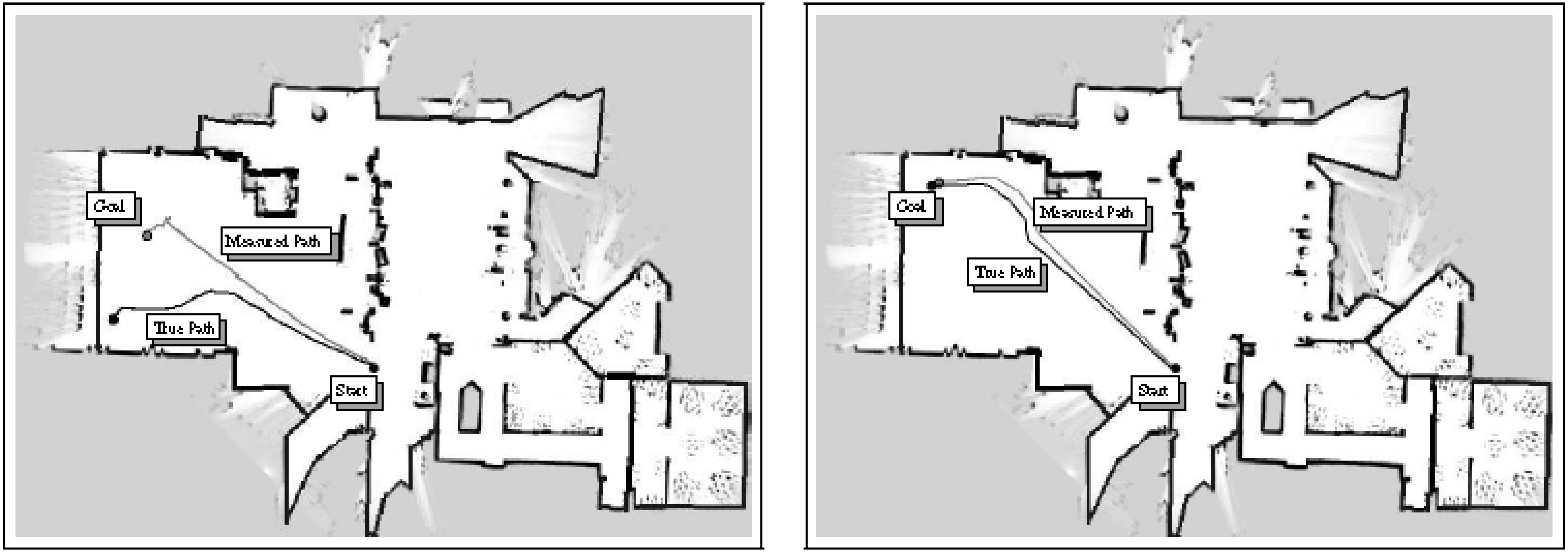
Nicholas Roy NICKROY@MIT.EDU
Massachusetts Institute of Technology,
Computer Science and Artificial Intelligence Laboratory
Cambridge, MA

Geoffrey Gordon GGORDON@CS.CMU.EDU
Carnegie Mellon University, School of Computer Science
Pittsburgh, PA

Sebastian Thrun THRUN@STANFORD.EDU
Stanford University, Computer Science Department
Stanford, CA



A planner for the mobile robot Pearl, shown in (a), must be able to navigate reliably in such real environments as the Longwood at Oakmont retirement facility, shown in (b). The white areas of the map are free space, the black pixels are obstacles, and the grey areas again are regions of map uncertainty. Notice the large open spaces, and many symmetries that can lead to ambiguity in the robot's position. The map is $53.6m \times 37.9m$, with a resolution of $0.1m \times 0.1m$ per pixel.

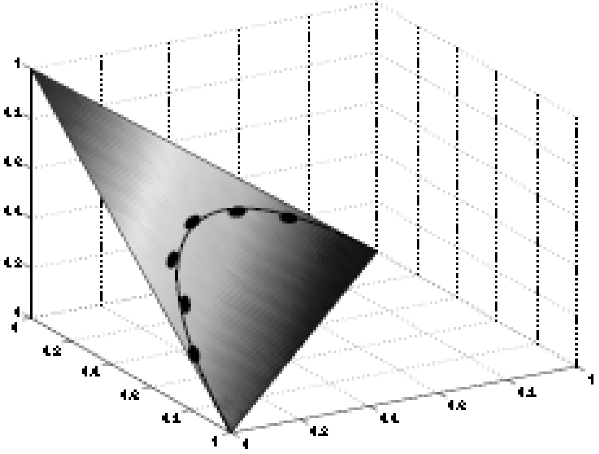


(a) Conventional controller

(b) Robust controller

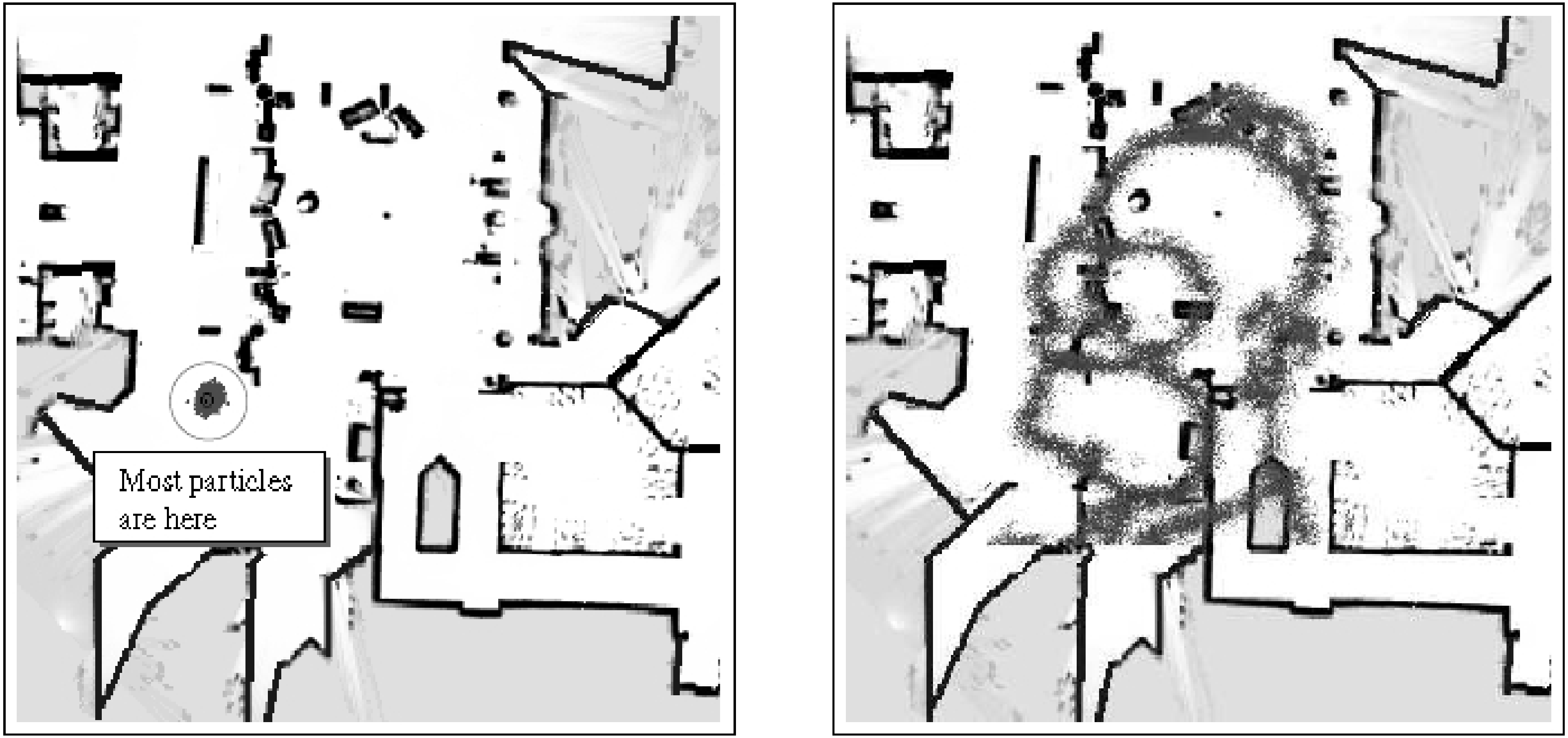
Two possible trajectories for navigation in the Longwood at Oakmont environment. The robot has limited range sensing (up to $2m$) and poor dead-reckoning from odometry. (a) The trajectory from a conventional motion planner that uses a single state estimate, and minimizes travel distance. (b) The trajectory from a more robust controller that models the state uncertainty to minimize travel distance and uncertainty.

Snažíme se o redukci dimenzionality



A one-dimensional surface (black line) embedded in a two-dimensional belief space (gray triangle). Each black dot represents a single belief probability distribution experienced by the controller. The beliefs all lie near the low-dimensional surface.

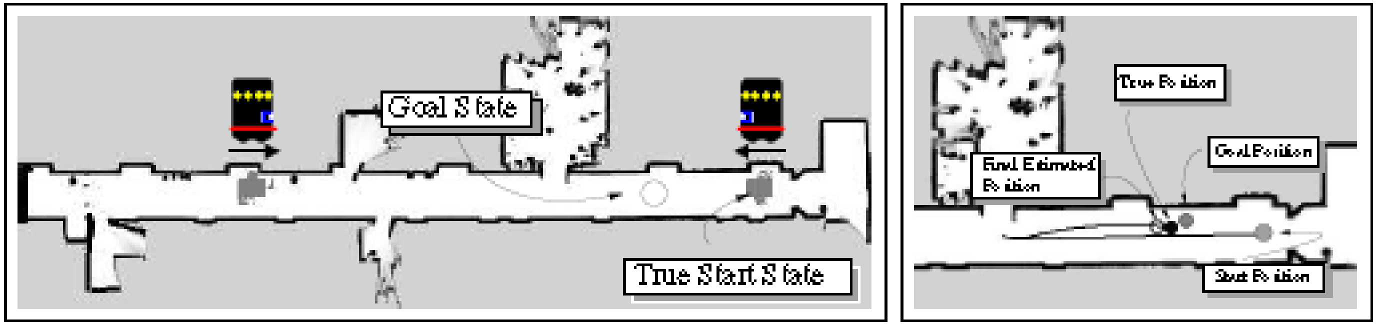
(Un)likely believes



(a) A common belief

(b) An unlikely belief

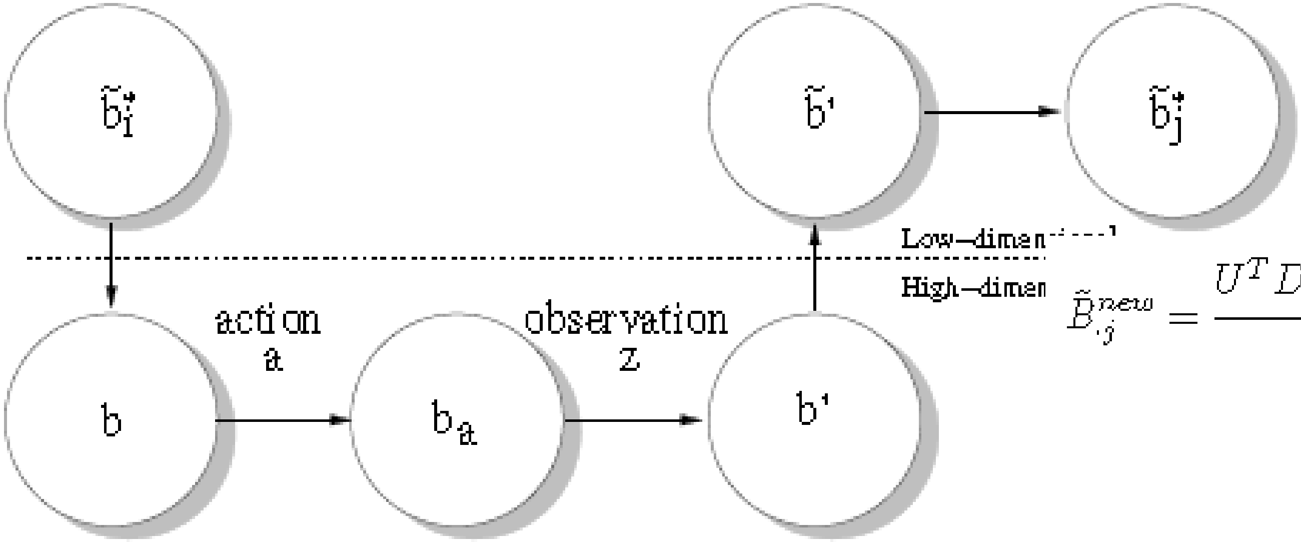
- Dvě místa vypadají stejně – častý belief, negaussovský



Computing transition function

k-nearest-neighbour, $w(\tilde{b}_j^*, \tilde{b}') = \frac{1}{k}$

$x = e^{U\bar{b}}$.



$$\tilde{B}_{.j}^{new} = \frac{U^T D_j(U\tilde{B}_{.j} + D_j^{-1}(B_{.j} - e^{U\tilde{B}_{.j}})}{(U^T D_j U + 10^{-5} I_l)}.$$

Figure 14: The process of computing a single transition probability.

For each transition $\tilde{b}_i^* \to b \to b_a \to b' \to \tilde{b}' \to \tilde{b}_j^*$ we can assign a probability

$$p(z, j|i, a) = p(z|b_a)\, w(\tilde{b}_j^*, \tilde{b}') = w(\tilde{b}_j^*, \tilde{b}') \sum_{l=1}^{|S|} p(z|s_l) b_a(s_l) \qquad (36)$$

The total transition probability $\tilde{T}^*(\tilde{b}_i^*, a, \tilde{b}_j^*)$ is the sum, over all observations z, of $p(z, j|i, a)$.

```
1. Generate the discrete low-dimensional belief space $\tilde{B}^*$ using E-PCA (cf. Table 1)
2. Compute the low-dimensional reward function $\tilde{R}^*$:
   For each $\tilde{b}^* \in \tilde{B}^*, a \in \mathcal{A}$
    (a) Recover $b$ from $\tilde{b}^*$
    (b) Compute $\tilde{R}^*(\tilde{b}, a) = \sum_{i=1}^{|S|} R(s_i, a) b(s_i)$.
3. Compute the low-dimensional transition function $\tilde{T}^*$:
   For each $\tilde{b}_i^* \in \tilde{B}^*, a \in \mathcal{A}$
    (a) For each $\tilde{b}_j^*$ : $\tilde{T}^*(\tilde{b}_i^*, a, \tilde{b}_j^*) = 0$
    (b) Recover $b_i$ from $\tilde{b}_i^*$
    (c) For each observation $z$
    (d)     Compute $b_j$ from the Bayes' filter equation (33) and $b$.
    (e)     Compute $\tilde{b}'$ from $b_j$ by iterating equation (26).
    (f)     For each $\tilde{b}_j^*$ with $w(\tilde{b}_j^*, \tilde{b}') > 0$
    (g)         Add $p(z, j|i, a)$ from equation (36) to $\tilde{T}^*(\tilde{b}_i^*, a, \tilde{b}_j^*)$
4. Compute the value function for $\tilde{B}^*$
    (a) t = 0
    (b) For each $\tilde{b}_i^* \in \tilde{B}^*$ : $V^0(\tilde{b}_i^*) = 0$
    (c) do
    (d)     change = 0
    (e)     For each $\tilde{b}_i^* \in \tilde{B}^*$:
                $V^t(\tilde{b}_i^*) = \max_a \left( \tilde{R}^*(\tilde{b}_i^*, a) + \gamma \sum_{j=1}^{|\tilde{B}^*|} \tilde{T}^*(\tilde{b}_i^*, a, \tilde{b}_j^*) \cdot V^{t-1}(\tilde{b}_j^*) \right)$
                change = change + $V^t(\tilde{b}_i^*) - V^{t-1}(\tilde{b}_i^*)$
    (f) while change > 0
```