LHMP2020

2nd Workshop on Long-term Human Motion Prediction

IEEE INTERNATIONAL CONFERENCE ON ROBOTICS AND AUTOMATION (ICRA), 2 JUNE 2020, 16:30 - 20:30 CEST ONLINE

Exploration/Exploitation in Path Planning Using Probability Propagation *

Francesco A. N. Palmieri[†] Krishna R. Pattipati[‡] Giovanni Fioretti
Francesco Verolla, Giovanni Di Gennaro[§] Amedeo Buonanno[¶]

[†]F. A. N. Palmieri with Dipartimento di Ingegneria, Università degli Studi della Campania "Luigi Vanvitelli", Aversa (CE), Italy and visiting at The University of Connecticut, Storrs, CT (USA), francesco.palmieri@unicampania.it

[‡]K. R. Pattipati is with Department of Electrical and Computer Engineering, University of Connecticut, Storrs, CT, USA, krishna.pattipati@uconn.edu

[§]G. Fioretti, F. Verolla and G. Di Gennaro are with Dipartimento di Ingegneria, Università degli Studi della Campania "Luigi Vanvitelli", Aversa (CE), Italy, giovanni.fioretti, francesco.verolla1@studenti.unicampania.it, giovanni.digennaro@unicampania.it

[¶]A. Buonanno is with ENEA, Energy Technologies Department, Portici (NA), Italy, amedeo.buonanno@enea.it

*Work partially supported by POR CAMPANIA FESR 2014/2020, ITS for Logistics, awarded to CNIT (Consorzio Nazionale Interuniversitario per le Telecomunicazioni). Research of Pattipati was supported in part by the U.S. Office of Naval Research and US Naval Research Laboratory under Grants #N00014-18-1-1238, #N00173-16-1-G905 and #HPCM034125HQU, and by a Space Technology Research Institutes grant (number 80NSSC19K1076) from NASAs Space Technology Research Grants Program.

https://www.mlsptlab-unicampania.it/

$$f(s_1,a_1) = \pi(s_1,a_1);$$

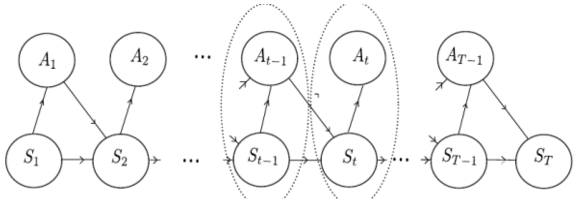
$$f(s_t,a_t) = \sum_{a_{t-1}} p(a_t|s_t,a_{t-1}) \sum_{s_{t-1}} p(s_t|s_{t-1}a_{t-1}) f(s_{t-1},a_{t-1}) \qquad t=2:T$$

$$f(s_T) = \sum_{a_{T-1}} \sum_{s_{T-1}} p(s_T|s_{T-1},a_{T-1}) f(s_{T-1},a_{T-1});$$

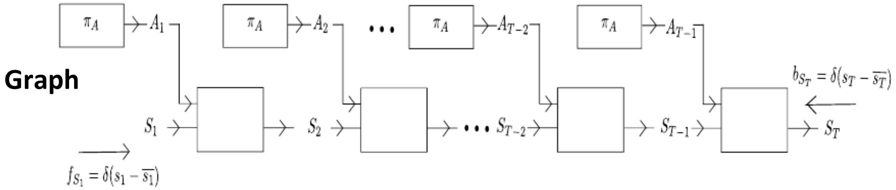
$$b(s_{T-1},a_{T-1}) \propto \sum_{s_T} p(s_T|s_{T-1}a_{T-1}) b(s_T);$$

$$b(s_{t-1},a_{t-1}) \propto \sum_{a_t} p(a_t|s_t,a_{t-1}) \sum_{s_t} p(s_t|s_{t-1}a_{t-1}) b(s_t,a_t) \qquad t=T-1:2$$

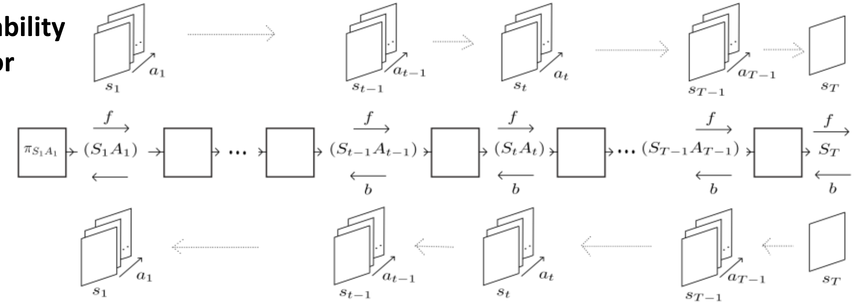
Bayesian Graph



Factor Graph



Probability Tensor Flow



Pure Diffusion

Goal

Start

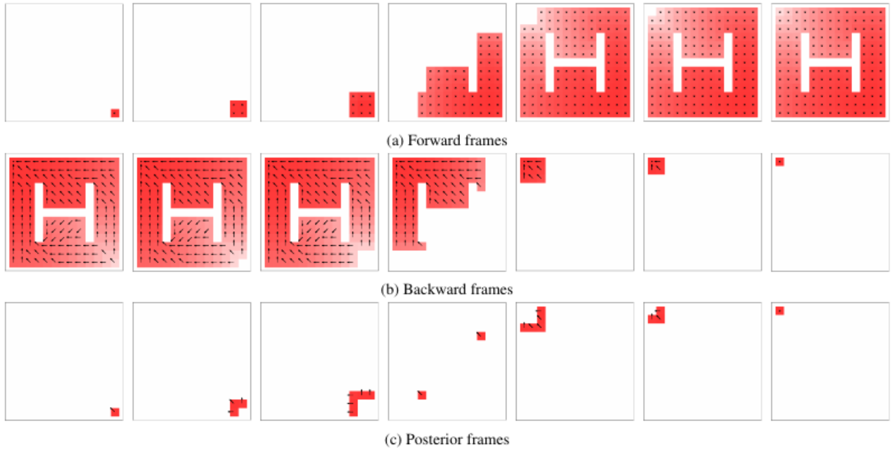
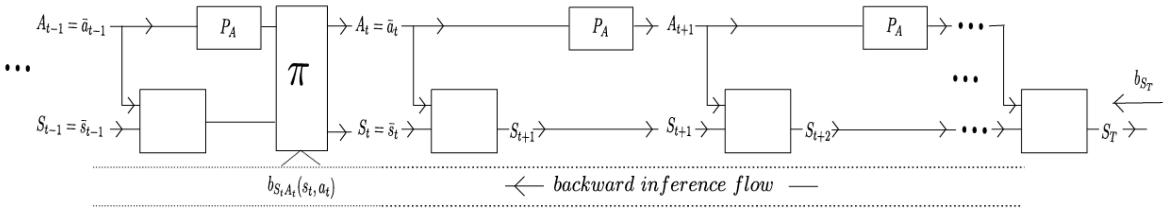


Fig. 2: Flows. From left to right, the snaps are taken at time $t=1, t=2, t=3, t=10, t=18, t=19, t=T=20$

Introducing control using the backward flow



Known MAP and Goal

G-algorithm Outline:

(a) Initialize state (goal) at time $t = T$ with $b_{S_T}(s_t) = \delta(s - \overline{g})$;

(b) Compute the complete backward flow using the last two equations in (4) from $t = T$ to $t = 2$;

(c) Initialize the state and action (start) at time $t = 1$ with $f_{S_1 A_1}(s_1, a_1) = \delta(s_1 - \overline{s}, a_1 - \bar{a})$;

(d) Set $t \leftarrow t + 1$ and compute the forward distribution $f_{S_t A_t}(s_t, a_t)$ using the second equation in (4);

(e) Compute the posterior distribution:

$p_{S_t A_t}(s_t, a_t) \propto f_{S_t A_t}(s_t, a_t) b_{S_t A_t}(s_t, a_t)$;

(f) Set $(S_t, A_t) = (\bar{s}_t, \bar{a}_t) = \text{argmax } p_{S_t A_t}(s_t, a_t)$

(g) Replace forward distribution as:

$f_{S_t A_t}(s_t, a_t) = \delta(s_t = \bar{s}_t, a_t = \bar{a}_t)$

(h) Back to (d) for the next time step.

The algorithm essentially maximizes the likelihood in a greedy fashion. The solution is reached much like in the Viterbi algorithm (poor man's Viterbi) in a large-dimensional space. Therefore, if the time horizon is set to minimum time, the solution is optimal.

Extended to multiple agents:

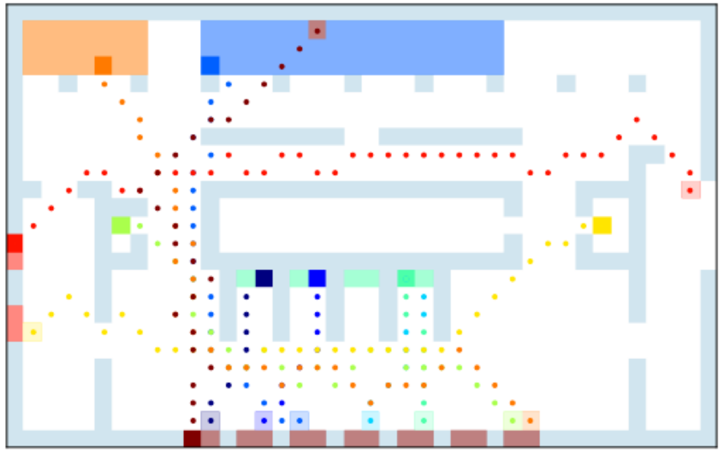


Figure 1. **Agents move in a train station-like environment from their starting positions to their goals**

[13] Francesco A.N. Palmieri, Krishna R. Pattipati, Giovanni Fioretti, Giovanni Di Gennaro, and Amedeo Buonanno, "Path Planning Using Probability Tensor Flow," *arXiv:2003.02774*, sub. for journal publication, II review, 2020.

No knowledge of the map and limited visibility

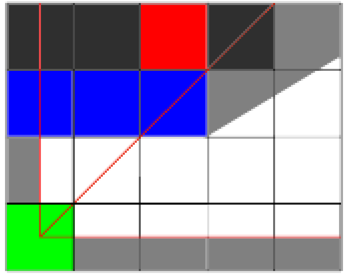


Figure 2. Agent's viewing geometry. The agent (green) looks at 315° (counterclockwise angles) with 90° viewing aperture. Because of the presence of the obstacles (blue), the goal (red) is not visible. The black areas are the invisible regions. We assume that a partially visible cell is visible (gray).

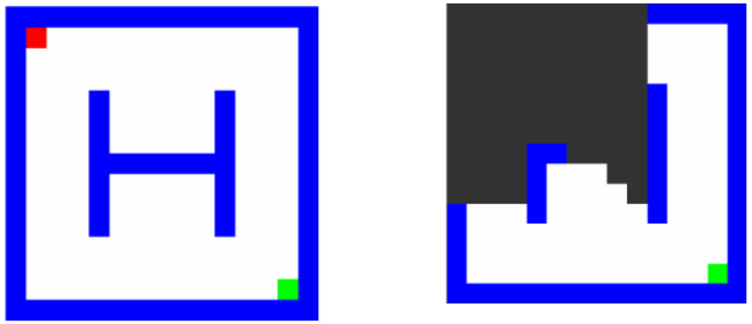
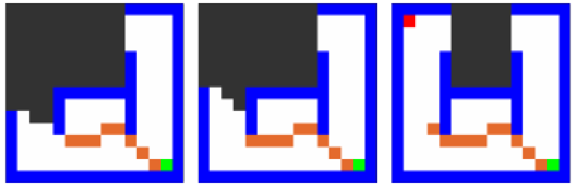
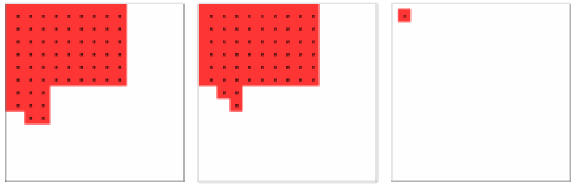


Fig. 10: Limited visibility. The black zone represents the area of the map the agent doesn't know yet.

The unknown map becomes a distributed goal



(a) Exploration frames



(b) Goals exploration frames

Fig. 11: Exploration. The frames show the position of the agent and the known area at $t = 10$, $t = 11$ and $t = 12$.

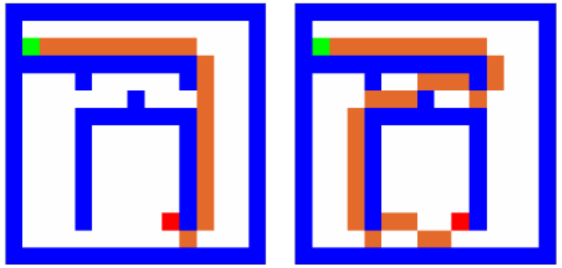
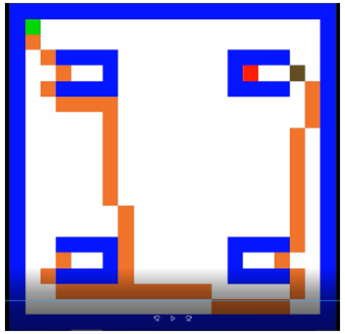
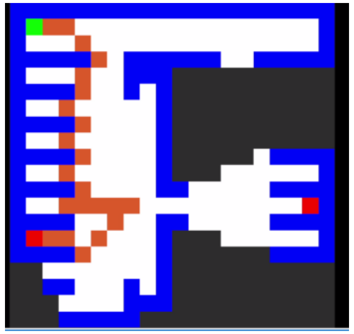


Figure 4. Paths obtained with a fully known map (left) and after exploration due to limited view (right)

Animations



Hidden target



Parking lot

Films available at the poster session, or on our website
https://www.mlsptlab-unicampania.it/research/topics/path-modeling.html