Topology-Driven Solver Selection for Stochastic Shortest Path MDPs via Explainable Machine Learning

Mathieu Gravel, Jaël Champagne Gareau

Cognitive Computer Science Department
Université du Québec à Montréal

29 May 2025

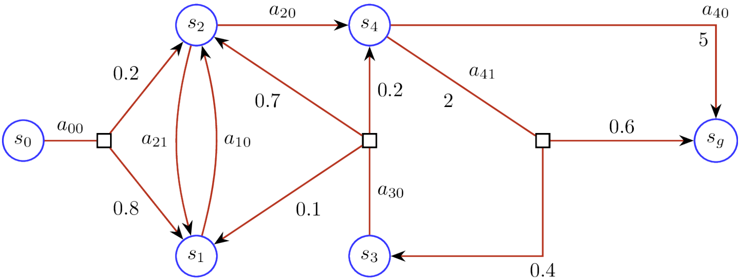
UQÀM

Canadian AI 2025

Outline

1. Markov Decision Processes - Synthetic MDP Generation
2. Explainable AI
3. Topology-Driven Solver Selection
4. Conclusion

Context : Markov Decision Processes (MDPs)



- **Markov Decision Processes** (MDPs) are a technique used to model decision-making problems under uncertain outcomes.
- The model aims to represent the states, actions and goals of a world state, in order to find the best policy for a given agent to reach a goal.

Context : MDPs Algorithms

Objective

Find a **policy** $\pi : S \rightarrow A$ that minimizes the expected total cost to reach a goal.

Classical algorithms

- Value Iteration (VI) [1]
- Policy Iteration (PI) [2]

Prioritization methods

- Generalized Prioritized Sweeping (genPS) [3]
- Partitioned, Prioritized, Parallel Value Iteration (P3VI) [4]

[1] Bellman, R. (1957). Dynamic Programming. Prentice Hall.

[2] Howard, R. A. (1960). Dynamic Programming and Markov Processes. John Wiley.

[3] Andre, D. et al. (1998). Generalized prioritized sweeping. Proceedings of the 10th International Conference on Neural Information Processing Systems (p. 1001-1007). MIT Press.

[4] Wingate, D. and Seppi, K. D. (2005). Prioritization methods for accelerating MDP solvers. Journal of Machine Learning Research, 6, 851-881.

Context : MDPs Algorithms

Objective

Find a **policy** $\pi : S \rightarrow A$ that minimizes the expected total cost to reach a goal.

Heuristic approaches

- Labeled Real-Time Dynamic Programming (LRTDP)[5]
- Improved Looped And/Or* (ILAO*)[6]

Topological approaches

- Topological Value Iteration (TVI)[7]
- Parallel-Chained Topological Value Iteration (pcTVI)[8]

[5] Bonet, B. and Geffner, H. (2003). Improving the Convergence of Real-Time Dynamic Programming. Proceedings of the 13th International Conference on Automated Planning and Scheduling (ICAPS 2003) (vol. 3, p. 12-21).

[6] Hansen, E. A. and Zilberstein, S. (2001). LAO* : A heuristic search algorithm that finds solutions with loops. Artificial Intelligence, 129(1-2), 35-62.

[7] Dai, P. et al. (2011). Topological value iteration algorithms. Journal of Artificial Intelligence Research, 42, 181-209.

[8] Champagne Gareau, J. et al. (2023). pcTVI : Parallel MDP solver using a decomposition into independent chains. Classification and data science in the digital age (p. 101-109). Springer International Publishing.

Motivation : Given a certain planning domain, which algorithm is faster ?

- Experts creating MDP domains for real-world uses cases needs to know which approach can find the optimal policy in a given time-frame.
- Which MDP solver are optimal for planning domains ?
- For certain domains, we already know the answer :
  - Dense MDPs (actions can lead to a large set of states) : **VI** and **PI** are often the best ;
  - MDPs having a large number of goal states : **heuristic approaches** are often the best.
  - MDPs having a large number of strongly connected components : **topological approaches** are often the best.

Table – Running times (ms) obtained for two different MDP solvers for two simple domains.

Name	*n*	VI	LRTDP
linearUniDir	10 000 000	2 594 650	**564**
denseProb	10 000	**196 211**	684 211

Which algorithm is faster

- What if we have a combination of the above features ?
- Is there a possible policy that could be used to select the optimal algorithm for a given MDP ?
- What are the possible distinctive features that can be extracted from an MDP ?
- To solve this issue, we can try to describe a MDP domain in classifiable variables.

Features of interest

- The **number of states** $|S|$ in the MDP, $\mathcal{O}(1)$.
- The **number of actions** $|A|$ in the MDP, $\mathcal{O}(1)$.
- The **number of goal states** $|G|$ in the MDP, $\mathcal{O}(1)$.

Features of interest

- The **number of states** $|S|$ in the MDP, $\mathcal{O}(1)$.
- The **number of actions** $|A|$ in the MDP, $\mathcal{O}(1)$.
- The **number of goal states** $|G|$ in the MDP, $\mathcal{O}(1)$.
- The **number of Strongly Connected Components (SCCs)** $|\mathfrak{S}|$ in the MDP, computed by Tarjan's algorithm : $\mathcal{O}(|S| + |A|)$.
- The **number of states in the largest SCC** $\max_{\mathcal{S} \in \mathfrak{S}} |\mathcal{S}|$.

Features of interest

- The **number of states** $|S|$ in the MDP, $\mathcal{O}(1)$.
- The **number of actions** $|A|$ in the MDP, $\mathcal{O}(1)$.
- The **number of goal states** $|G|$ in the MDP, $\mathcal{O}(1)$.
- The **number of Strongly Connected Components (SCCs)** $|\mathfrak{S}|$ in the MDP, computed by Tarjan's algorithm : $\mathcal{O}(|S| + |A|)$.
- The **number of states in the largest SCC** $\max_{\mathcal{S} \in \mathfrak{S}} |\mathcal{S}|$.
- The **distribution of actions**, $\mathcal{O}(|S|)$:
  $\forall k, P_k^a :=$ proportion of states which have k applicable actions.
- The **distribution of probabilistic transitions**, $\mathcal{O}(A)$:
  $\forall k, P_k^t :=$ proportion of actions which have k probabilistic transitions.

Features of interest

- The **number of states** $|S|$ in the MDP, $\mathcal{O}(1)$.
- The **number of actions** $|A|$ in the MDP, $\mathcal{O}(1)$.
- The **number of goal states** $|G|$ in the MDP, $\mathcal{O}(1)$.
- The **number of Strongly Connected Components (SCCs)** $|\mathfrak{S}|$ in the MDP, computed by Tarjan's algorithm : $\mathcal{O}(|S| + |A|)$.
- The **number of states in the largest SCC** $\max_{\mathcal{S} \in \mathfrak{S}} |\mathcal{S}|$.
- The **distribution of actions**, $\mathcal{O}(|S|)$: $\forall k, P^a_k :=$ proportion of states which have k applicable actions.
- The **distribution of probabilistic transitions**, $\mathcal{O}(A)$: $\forall k, P^t_k :=$ proportion of actions which have k probabilistic transitions.
- The **clustering coefficient** : $\mathfrak{C} := \frac{1}{|S|} \sum_{s \in S} \frac{e_s}{k_s(k_s - 1)}$, where e_s is the number of pairs of states directly reachable from s that are also directly reachable from each other, and k_s is the number of states directly reachable from s. Moreover, $\mathfrak{C}$ is set to be 0 when $k_s < 2$, $\mathcal{O}(|S|^3)$.

Features of interest

- The **number of states** $|S|$ in the MDP, $\mathcal{O}(1)$.
- The **number of actions** $|A|$ in the MDP, $\mathcal{O}(1)$.
- The **number of goal states** $|G|$ in the MDP, $\mathcal{O}(1)$.
- The **number of Strongly Connected Components (SCCs)** $|\mathfrak{S}|$ in the MDP, computed by Tarjan's algorithm : $\mathcal{O}(|S| + |A|)$.
- The **number of states in the largest SCC** $\max_{\mathcal{S} \in \mathfrak{S}} |\mathcal{S}|$.
- The **distribution of actions**, $\mathcal{O}(|S|)$:
  $\forall k, P^a_k :=$ proportion of states which have k applicable actions.
- The **distribution of probabilistic transitions**, $\mathcal{O}(A)$:
  $\forall k, P^t_k :=$ proportion of actions which have k probabilistic transitions.
- The **clustering coefficient** : $\mathfrak{C} := \frac{1}{|S|} \sum_{s \in S} \frac{e_s}{k_s(k_s - 1)}$, where e_s is the number of pairs of states directly reachable from s that are also directly reachable from each other, and k_s is the number of states directly reachable from s. Moreover, $\mathfrak{C}$ is set to be 0 when $k_s < 2$, $\mathcal{O}(|S|^3)$.
- The **goals-eccentricity** of the MDP : $\mathcal{G} := \min_{g \in G} \max_{s \in S} \bar{d}(s, g)$, where $\bar{d}(s, g)$ is the minimum number of actions (the cost of each action is not considered) that must be executed to reach g from s, $\mathcal{O}(|G|(|S| \log |S| + |A|))$.

Synthetic Graphs Generation

- To categorize the topological features descriptive of the richness, a high amount of distinct MDPs are needed.
- A small number of synthetic MDP planning domains exist that can be used, but are limited in possible edge-cases generation e.g. :
  - Layered MDPs (used to control the number of SCCs) ;
  - Chained MDPs (used to control the number of independent chains of states).
- To better represent distinct MDP cases, there are a lot more synthetic graph generation methods that can be modified to generate MDP planning domains [9].

Technique	**Degrees Distr.**	**Clust. Coeff.**	**Diameter**
Erdös-Rényi	Binomial	small ($\bar{k}/n$)	small : $\mathcal{O}(\log(n))$
Watts-Strogatz	Almost-constant	large	small
Barabási–Albert	Scale-free ($\bar{k}^{-3}$)	large ($\bar{k}^{-1}$)	small : $\mathcal{O}(\frac{\log(n)}{\log(\log(n))})$
Kronecker	Multinomial	flexible	flexible

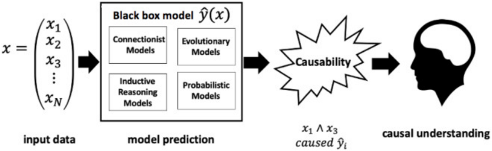
[9] Champagne Gareau, J., Beaudry, É. and Makarenkov, V. (2024). Towards topologically diverse probabilistic planning benchmarks : Synthetic domain generation for markov decision processes. In : J. Trejos, T. Chadjipadelis, A. Grané and V. Mario (dir.), Data science, classification and artificial intelligence for modeling decision making (p. 63-70). Springer International Publishing.

Solver classification and topological features as characteristics

- Classical algorithms in Artificial Intelligence can be used to represent the links between topological features and the optimal MDP resolution algorithm, such as **SVM** or **Neural-Networks**.
- The need to represent explicitly the importance of each topological features over each of the algorithms makes it a necessity to avoid black-boxes approaches.
- **Explainable AI methods** can be used to extract the topological/solver correspondances for domains experts to help them select the best methods for their use-cases.

Explainable AI approaches

- Interpretable models
  - Offers intrinsic explanations throught human-understandable structures.
  - Scale well for domains with information-rich, few-features set.
- Post-hoc methods
  - Algorithms to derive post-hoc explanations of models decisions.
  - Some of them enable “what-if” reasoning by creating approximate explanations that can be adapted and modified, such as counter-factuals explanations.



$x = (x_1, x_2, x_3, \vdots, x_N)$
Black box model $\hat{y}(x)$
Connectionist Models
Evolutionary Models
Inductive Reasoning Models
Probabilistic Models
Causability
input data
model prediction
$x_1 \wedge x_3$ caused $\hat{y}_i$
causal understanding

Figure – Source : Yu-Liang Chou, Catarina Moreira, Peter Bruza, Chun Ouyang, Joaquim Jorge, Counterfactuals and causability in explainable artificial intelligence : Theory, algorithms, and applications, Information Fusion, 2022

Goal and planning domain

- Goal : Systematically analyze the impacts of MDP topological features to describe their domain, and to select which algorithms is best adapted to offert an optimal policy.

MDP domains

- Layered domain
- WetFloor
- Single-Armed Pendulum (SAP)
- Synthetic domains
  - Erdös-Rényi
  - Barabási-Albert
  - Watts-Strogatz
  - Kronecker

Features set and Classes

MDP Solvers

- Value Iteration
- LRTDP
- ILAO*
- Topological Value Iteration

- Each MDP domain was sent as input to MDP solvers, in order to extract the **optimal solution generation time**.
- This value was used to classify which solver was categorized as the fastest.

Table – Running times (ms) obtained for each solvers on the tested domains. Fastest time on each domain is bolded.

Name	*n*	VI	LRTDP	ILAO*	TVI
linearUniDir	10 000 000	2 594 650	564	16 912 088	**497**
linearBidirDet	100 000	1 577 974	**9**	>1h	1 721 993
linearBidirProb	130	983	4 579	2 885	**982**
denseDet	10 000	1 660	**0.1**	**0.1**	1 825
denseProb	10 000	**196 211**	684 211	676 912	208 256

Training info

Classification and features analysis algorithms

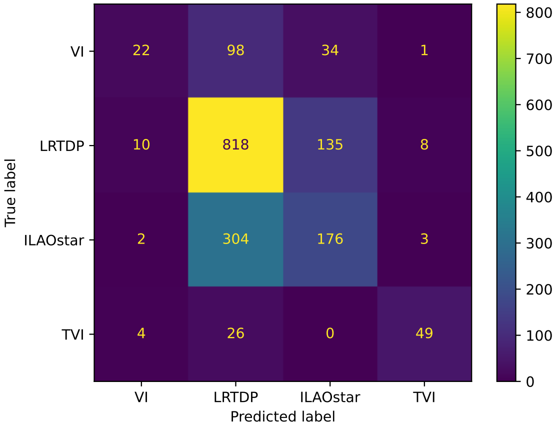
- Global classifier (LightGBM) predicts the fastest solver.
- Solver-specific classifiers (Iterative Random Forests) predict runtime distributions for individual solvers.
- Features impurities values generated for solver-specific classifiers, to extract features-specific importances for each class of MDP domains. Counter-factual explanations are generated for instances testing, to give greater assurances to the user.

Traning info - Dataset

Table – MDP Models used for the classifier training

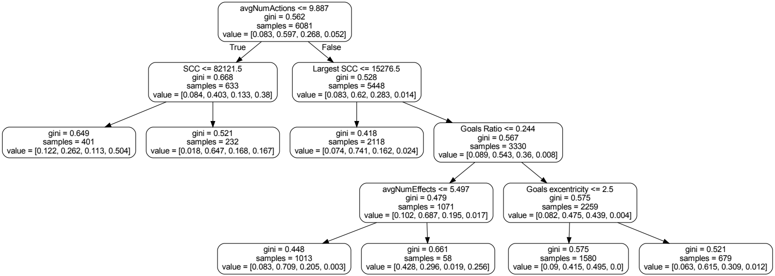
Name	Number of instances
Erdös-Rényi	1614
Barabási-Albert	1989
Watts-Strogatz (SmallWorld)	1968
Kronecker	1448
Layered	880
WetFloor	200
Single-Armed Pendulum (SAP)	91

Results – Global classifier Confusion Matrix



True label \ Predicted label	VI	LRTDP	ILAOstar	TVI
VI	22	98	34	1
LRTDP	10	818	135	8
ILAOstar	2	304	176	3
TVI	4	26	0	49

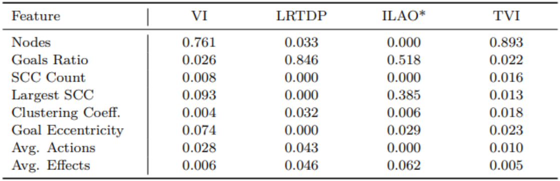
Results – Individual classifier : ILAO* explanation tree


avgNumActions <= 9.887
gini = 0.562
samples = 6081
value = [0.083, 0.597, 0.268, 0.052]
True
False
SCC <= 82121.5
gini = 0.668
samples = 633
value = [0.084, 0.403, 0.133, 0.38]
Largest SCC <= 15276.5
gini = 0.528
samples = 5448
value = [0.083, 0.62, 0.283, 0.014]
gini = 0.649
samples = 401
value = [0.122, 0.262, 0.113, 0.504]
gini = 0.521
samples = 232
value = [0.018, 0.647, 0.168, 0.167]
gini = 0.418
samples = 2118
value = [0.074, 0.741, 0.162, 0.024]
Goals Ratio <= 0.244
gini = 0.567
samples = 3330
value = [0.089, 0.543, 0.36, 0.008]
avgNumEffects <= 5.497
gini = 0.479
samples = 1071
value = [0.102, 0.687, 0.195, 0.017]
Goals excentricity <= 2.5
gini = 0.575
samples = 2259
value = [0.082, 0.475, 0.439, 0.004]
gini = 0.448
samples = 1013
value = [0.083, 0.709, 0.205, 0.003]
gini = 0.661
samples = 58
value = [0.428, 0.296, 0.019, 0.256]
gini = 0.575
samples = 1580
value = [0.09, 0.415, 0.495, 0.0]
gini = 0.521
samples = 679
value = [0.063, 0.615, 0.309, 0.012]

Results – Features analysis

Topological features per global importances :

1. State Count
2. Goals Density
3. Max SCC Size
4. Goal Eccentricity
5. Avg Stochasticity
6. Clustering Coefficient
7. Actions Density
8. SCC Count



Feature	VI	LRTDP	ILAO*	TVI
Nodes	0.761	0.033	0.000	0.893
Goals Ratio	0.026	0.846	0.518	0.022
SCC Count	0.008	0.000	0.000	0.016
Largest SCC	0.093	0.000	0.385	0.013
Clustering Coeff.	0.004	0.032	0.006	0.018
Goal Eccentricity	0.074	0.000	0.029	0.023
Avg. Actions	0.028	0.043	0.000	0.010
Avg. Effects	0.006	0.046	0.062	0.005

Figure – Features importance per individual solver

Conclusion

- We used state of the arts topological MDP features with synthetic data generation to create a training corpus for MDP domains
- We proposed a method to classify MDP algorithms per topological features, and analyzed each features importances for each family of approaches
- As future work, we plan to create a bigger corpus with more variations for MDPs to even out the MDP solver fastest instances.

Acknowledgments

LATECE

***Fonds de recherche Nature et technologies* Québec**

We acknowledge the support of the *Latece laboratory* (UQAM), Canada Research Chair in Privacy-preserving and Ethical Analysis of Big Data, and the *Fonds de recherche du Québec — Nature et technologies* (FRQNT).