Topological Decompositions Enhance Efficiency of Reinforcement Learning

Michael J. Catanzaro
Geometric Data Analytics
343 W. Main St.
Durham, NC 27701
michael.catanzaro@geomdata.com

Aaron Dharna*
University of British Columbia
Geometric Data Analytics
343 W. Main St.
Durham, NC 27701
aadharna@gmail.com

Jay Hineman
Geometric Data Analytics
343 W. Main St.
Durham, NC 27701
jay.hineman@geomdata.com

James B. Polly
Geometric Data Analytics
343 W. Main St.
Durham, NC 27701
james.polly@geomdata.com

Kevin McGoff
Geometric Data Analytics
UNC Charlotte
343 W. Main St.
Durham, NC 27701
kevin.mcgoff@geomdata.com

Abraham D. Smith
Geometric Data Analytics
University of Wisconsin-Stout
343 W. Main St.
Durham, NC 27701
abraham.smith@geomdata.com

Paul Bendich
Geometric Data Analytics
Duke University
343 W. Main St.
Durham, NC 27701
paul.bendich@geomdata.com

Abstract—**Coordinating multiple sensors can be expressed as a reinforcement learning [RL] problem. Deep RL has excelled at observation processing (for example using convolution networks to process gridded data), but it suffers from sample inefficiency. To address this problem, we topologically decompose the total observation space into overlapping components, using the detection of co-incidence or spatial adjacency of the sensors to construct a stratified decomposition analogous. By allowing the RL agent to learn within the context of this decomposition and take advantage of it through action masking, we achieve positive reward and efficient gains over the learning process. We demonstrate performance and efficiency gains through several experiments using a bespoke game implementation that combines RLlib, Griddly, and Gymnasium. We draw analogies between our games and more general co-incidence in sensing space, time, or modality. We find that our decomposition can be combined with modern RL algorithms to learn high-performing sensor control policies, and our pipeline scales well as the number of sensors grows.**

Table of Contents

1. Introduction and Background

Coordinating multiple sensors can be expressed as a model-free, deep reinforcement learning [RL] problem. Deep means that deep neural networks are trained to optimize reward. Deep RL has recently been used to achieve super human level performance on games such as go [1], chess [2], Starcraft [3], and Diplomacy [4]. Team play aspects have been explored in games such as DOTA 2 [5]. Beyond examples of games, deep RL can beat expert human pilots in drone racing [6] and be used for certain control aspects of nuclear reactors [7]. In all of these cases the number of training examples used was immense. Recent theoretical works [8] have demonstrated that sample efficiency in single-agent reinforcement is low and is a core challenge.

Model-free approaches are popular in deep RL because model-based RL has difficulties learning high performing policies in practice. For example, a-priori models of agent/environment interaction are either unknown or of low fidelity. This problem is further complicated in the multi-agent setting since other agents might also be considered as part of the environment. These considerations are certainly present in the multi-agent task of joint sensing and tracking of objects.

Overall, a major defect of model-free deep RL is sample inefficiency, meaning that many samples are required for a desired level of performance. For example, [9] uses "regret" to quantify sample inefficiency. This problem is amplified in the context of tracking with a large sensor grid, because the observation spaces can explode combinatorially.

We propose and demonstrate a reinforcement learning [RL] technique that learns and exploits decompositions of state and observation spaces to overcome the sample inefficiency of existing RL methods. The proposed paradigm of topological decomposition transcends the distinction between single-agent and multi-agent learning and requires very limited sampling. This technique can be combined with common deep (model-free) RL algorithms such as proximal policy optimization [PPO], and Deep Q networks [DQN].

Our demonstration is applied to a simulation of sensor coordination, which can be interpreted as a providing single- or multi-agent states and observations. Our implementation is

* Work done during internship at GDA
979-8-3503-0462-6/24/$31.00 ©2024 IEEE

designed to track a target moving randomly through an array of sensors. The sensor array is controlled by toggling each sensor as either *on* or *off*. An on-sensor can observe a target within its field of view, whereas an off-sensor provides no readings whatsoever. We use the terms *on* and *off* as a general placeholder for activation or tasking the sensor for tracking the target. The agent receives positive reward whenever an on-sensor detects the target. There is a cost, however, to turning a sensor on given by a negative term in the reward function (Eq. (1)). To improve performance of track custody, we introduce two new methodologies for training agents, referred to as decompositions by analogy with topology. The decompositions restrict the set of actions the agent can take by enforcing that sensors far from the target remain off. We develop baselines that demonstrate the difficulty of learning a high-performing policy for this game by executing state-of-the-art deep RL algorithms—namely PPO and DQN.

Paper Outline

The rest of this paper is organized as follows. Section 2 describes the novel reinforcement learning (RL) game engine that we designed to support our experiments. The topological decomposition paradigm central to defining the new and flexible proposed RL techniques is outlined in Section 3. Experimental results are given and evaluated in Section 4, and the paper concludes in Section 5 with summary and discussion of needed future work.

2. GAME ENGINE

Our RL game engine comprises of several open source packages to provide a versatile platform for simulating grid-based games (Figure 1). We use Griddly, a flexible and extremely fast open source framework, to define the grid-world environment [10]. Griddly's games are specified by an easy-to-interpret YAML file specifying the rules, objectives, and layout of the games. This allows for simple manipulation of game mechanics, including modifications to the number and geometry of sensors in our simulated sensor array. The RL agents are trained and their models are specified using Ray's RLlib package [11]. RLlib is a state-of-the-art platform for training single and multi-agent games built on Gymnasium [12]. This combination of Griddly and RLlib allows for an efficient pipeline to design, train, and test RL agents with topological stratifications for sensor-array type simulations.

The main rationale behind the design choice of our game engine is speed. Griddly provides extremely fast execution speeds of up to 70,000 game episodes per second on a single thread. Rapid game simulations allow for experimentation with a variety of sensor arrangements and topological decompositions. We can also inspect game runs visually throughout training, as shown in Figure 2.

Figure 1. Game Engine components.

3. TOPOLOGICAL DECOMPOSITIONS

Our approach to topological decomposition is inspired by the topology of stratified spaces, but it is designed to be agnostic to the underlying structure of the observation spaces. Our definitions are designed to allow sensor-control policies with excellent sample efficiency.

To formalize the approach, consider $n < \infty$ sensors. For $i = 1, \dots, n$, the ith sensor has a set O_i of possible observations, called its **observation space**. We do not assume that O_i has any particular structure, and in particular, we do not assume that all O_i are identical. For example, a given O_i could include entries such as *yes/no*, floating-point measurements *12.2 meters*, coordinates $(x, y) = (3, -7)$, labels like "cat," a waveform, a picture, and so on. Each sensor has a specific subset $N_i \subset O_i$ which represents the **null observations**. For example, N_i could be $N_i = \emptyset$ for a sensor providing labels, $N_i = \{\vec{0}\}$ for a sensor where O_i is a vector space, or $N_i = \{x : \|x\| < \varepsilon\}$ for a camera subject to thermal noise ε in an appropriate norm. A **detection** is a non-null observation $x_i \in D_i$, for $D_i = N_i^c$.

The **total observation space** is the set O of tuples $\vec{x} = (x_1, \dots, x_n)$ for $x_i \in O_i$. The null observations are $N = N_1 \times \cdots \times N_n \subset O$, and the detections are $D = N^c$. For any total observation $\vec{x}$, let $S_0(\vec{x}) = \{i : x_i \in D_i\}$, the set of sensors that have a detection (non-null observation) in $\vec{x}$. If $\vec{x} \in N$, then $S_0(\vec{x}) = \{\}$.

We assume that time is discrete, and all sensors use the same clock. A total observation at time t is written $\vec{x}(t) = (x_1(t), \dots, x_n(t))$. Through experiments or simulations or physical models, we have access to many time-sequences of observations, $\dots, \vec{x}(t-1), \vec{x}(t), \vec{x}(t+1), \dots$.

Given an observation $\vec{x}(t)$ and time-shift $\tau \in \mathbb{Z}$, let

$$S_\tau(\vec{x}(t)) = \{i : x_i(t+\tau) \in D_i\}.$$

For example, if $\tau = 0$, then $S_0(\vec{x}(t))$ consists of those sensors which have detection at time t; if $\tau = 1$, then $S_1(\vec{x}(t))$ consists of those sensors which will have detection at time $t+1$. Thus, a sensor $i \in S_\tau(\vec{x}(t))$ if and only if there is a time-sequence of observations $\vec{x}(t)$ for which the same sensor $i \in S_0(\vec{x}(t+\tau))$.

The sets $S_\tau(\vec{x})$ can be hard-coded when the target mechanics are understood, or $S_\tau(\vec{x})$ can be learned approximately in a distributional sense, and those distributions can be taken to vary over time.

At each time step t, the agent predicts a set of sensors $\hat{S}(t) \subset \{1, \dots, n\}$ that are likely to detect something and turns them "on". Given a total observation $\vec{x}(t)$, the reward function is

$$R(\vec{x}(t), t) = B \cdot \# \left(\hat{S}(t) \cap S_0(\vec{x}(t))\right) - \#\hat{S}(t), \qquad (1)$$

where B is the benefit of maintaining track custody. The reward function is designed so that the agent attempts to activate sensors to maintain custody, but is penalized for the total number of activated sensors.

When training with the reward function of Eq. (1), we use the sets S_τ to enforce policies that force the RL agent to use resources efficiently. Our **Visibility Decomposition** policy is $\hat{S}(t) \subset S_0(\vec{x}(t-1))$. Under the Visibility Decomposition, the agent is forced to turn off all sensors which cannot

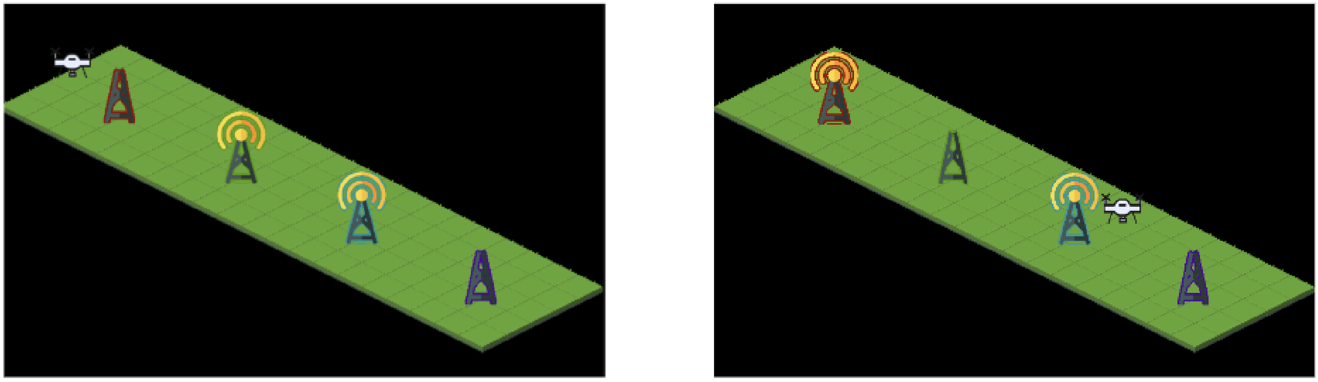


Figure 2. Game with a 4×1 sensor arrangement. Sensors are depicted as towers and the target is represented by a drone. Sensors being activated are shown by the concentric rings emanating from the tower top.

detect the particle at the current time step. Our **NextStep Decomposition** policy is $\hat{S}(t) \subset S_1(\vec{x}(t-1))$. The NextStep Decomposition forces the agent to turn off all sensors which cannot detect the particle at the next time step, given its current position. The combination of this reward function and these policies helps to overcome the dimensional limitations of other RL methods that contribute to sample inefficiency.

4. Results

We compare learning outcomes achieved using off-the-shelf methods (referred to as "baseline") and the topological decompositions introduced above. The effectiveness of the Visibility and NextStep decompositions are studied with varying sensor geometry and number of sensors. The sensors are equally spaced apart and arranged in a rectangular $n \times k$ layout. If $k = 1$, we say the arrangement is 'linear' and if $k > 1$, the arrangement is a 'grid'.

Results: Baseline methods perform poorly

Our first finding is that baseline methods perform poorly as the number of sensors in the array increases. The reward curve for PPO for varying number of linearly arranged sensors is shown in Figure 5. With the exception of the 8×1 layout, none of the reward curves show evidence of learning, despite training for nearly 1,000 iterations or 2,000,000 episodes. At the beginning of training, randomized play degrades linearly as the number of sensors grows, and baseline learning methods do not overcome this degradation. The same analysis applies to baseline methods for grid arrangements, as shown in Figure 6. In all of these training runs, the agent never averages positive reward per episode.

Results: Decomposition improves reward

We next train with a varying arrangement of sensors as before, but now enforce a naive version of the Visibility Decomposition introduced above. A naive enforcement means the decomposition is applied *after* the agent has selected its action in each game step. The agent is therefore 'unaware' that its proposed action was modified. In Figure 7, we compare the reward curves for baseline versus the visibility decomposition for linear sensor arrangements. The visibility decomposition clearly outperforms the baseline methods, and maintains positive reward throughout training. The reward curves for grid layouts are shown in Figure 8 and display similar traits. Importantly, this improved performance does not degrade as the number of sensors increases.

The naive enforcement of the visibility decomposition does improve reward curves but does not lead to learning by the agent. Learning during training is indicated by increases in reward as a function of training iteration step. The lack of learning is evident in the relatively flat nature of the reward curves of Figures 7 and 8. To make this clear, the naive enforcement of the visibility decomposition reward curves have been plotted in Figures 9 and 10. In general, these curves do not exhibit any upward trend of reward and certainly do not overcome their intial reward from the first 50 training step iterations (which is essentially random play given how little the agent has been trained at this point).

Results: Decomposition through masking enables learning at scale

We next examined the effect of enforcing decompositions through action masking. Action masking is a training technique that exposes the learning agent to the rules of the decomposition *during training*, and in theory, allows the agent to learn the enforced rules on its own. The effect of masking is highlighted for linear arrangements in Figure 11. We find dramatic increases in the reward curve as a function of training iteration, displaying the characteristics of learning throughout training. The reward curve trends are also independent of sensor geometry. Thus, the visibility decomposition with masking enables the agent to learn at scale, even as the number of sensors increases. This is in stark contrast to the degradation of reward as a function of sensor quantity of Figures 5 and 6.

Figures 11 and 12 show that masked (solid lines) learning algorithms significantly improve reward over naive rule enforcement (dashed lines). Performance degrades slowly relative to problem size (i.e. number of sensors).

Results: Reward metrics and improved performance via masking

In addition to reward, we compute two other key metrics for each training iteration. The first additional metric is a *track rate*, defined as the percentage of training steps for which the particle is visible to the sensors that are on. The second is *lights on per step*, defined as the average number of sensors on per game step. These two metrics measure the trade-off between accuracy (high track rate) and efficiency (lights on per step equal to 1). In a perfect (albeit likely impossible) scenario, the track rate would be 100% and the lights on per step would be 1.

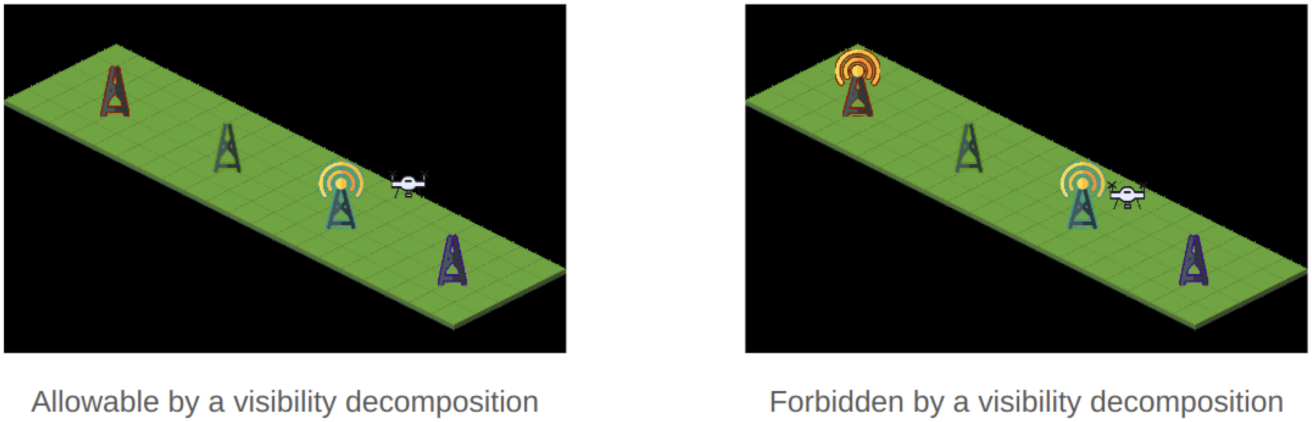


Figure 3. The visibility decomposition forces non-nearby sensors to be off, while the agent controls the nearby sensors.

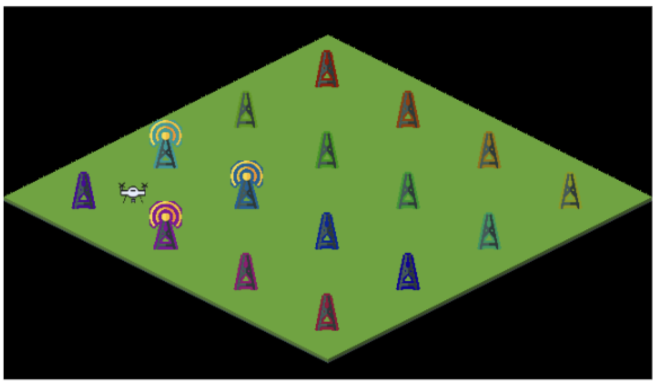


Figure 4. Allowable sensor actions using NextStep decomposition for a 4 by 4 sensor arrangement.

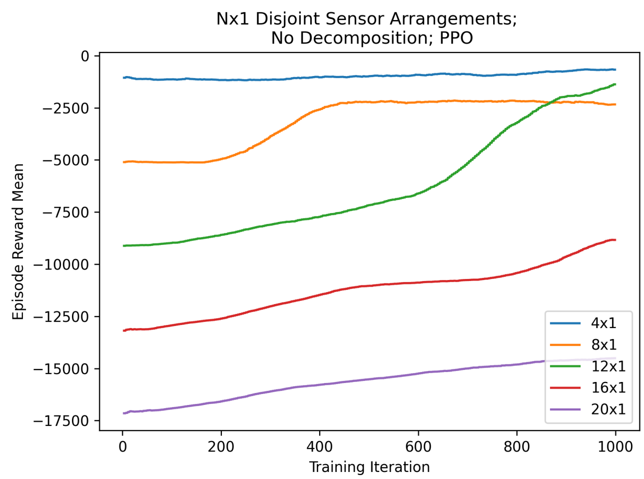


Figure 5. Baseline with increasing number of sensors, arranged in a line.

Track rate and lights on per step are annotated over the reward curves of a grid arrangement in Figure 12. We find as before that masking yields improved performance versus naive enforcement of the visibility decomposition and exhibits learning behavior. The introduced metrics also show the difference between naive and masked enforcement. As training progresses, the masked decomposition shows steady increase in track rate and lights on per step, whereas the naive decomposition shows both quantities decreasing.

Results: NextStep decomposition further improves reward for complex arrangements

Figures 13 and 14 show that the reward curves for masked enforcement of the NextStep decomposition. We find the NextStep decomposition (solid lines) outperforms the Visibility decomposition as the sensor geometry grows more complex with more sensors. The NextStep decomposition with masking allows for substantially more actions than the Visibility decomposition with masking (dashed lines). Comparing Visibility with NextStep, we find the optimal performance of the NextStep decomposition to be greater, at the cost of it being a potentially harder policy to learn and therefore slower to learn. Our empirical results indicate that NextStep outperforms Visibility, especially for sensor arrangements with more complex geometry (e.g., sensors overlapping with adjacent corners, edges, etc.).

5 x 4 game	Episode Reward Mean	Track Rate	Lights On Per Step
Baseline (B=3)	-12,500	47%	8.96
Visibility (B=3)	1,250	34%	0.4
Visibility Mask (B=10)	10,750	64%	0.75
NextStep Mask (B=10)	11,500	71%	1.11

Table 1. Statistics of our various learning paradigms for a 5×4 sensor arrangement.

Summary Statistics

Table 1 shows a summary of the different learning approaches presented for the 5×4 sensor arrangement for two values of the benefit B of Eq. (1). We see the baseline performance of PPO has extremely poor play, evidenced by the negative episode reward mean. The baseline lights on per step is also sub-optimal, given that the target can be in view of at

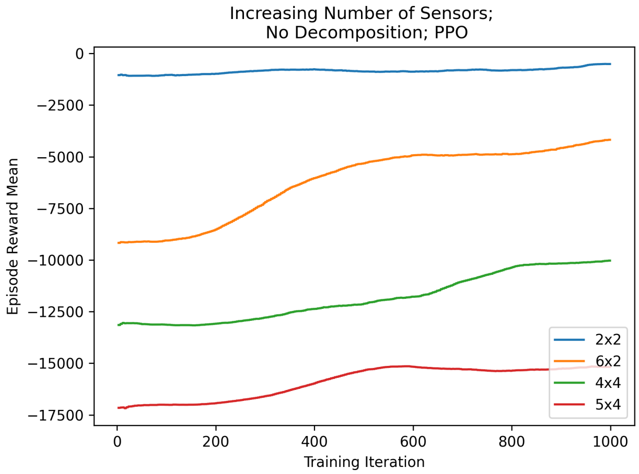


Figure 6. Baseline with increasing number of sensors, arranged in a grid.

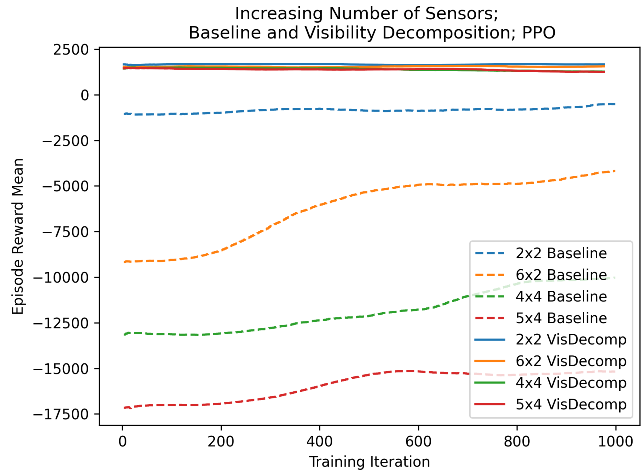


Figure 8. Visibility Decomposition with increasing number of sensors, arranged in a grid.

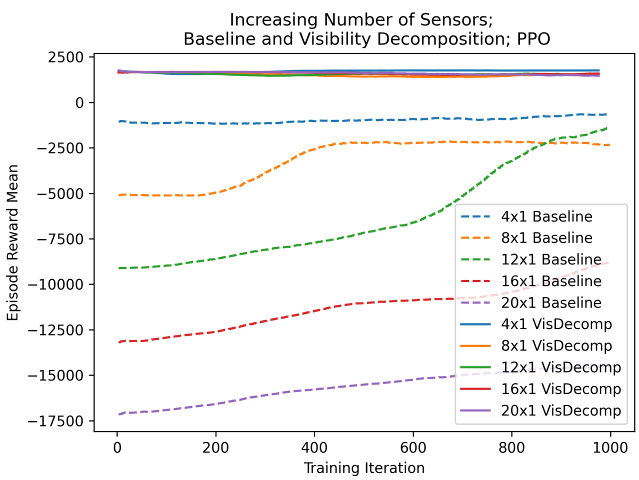


Figure 7. Visibility Decomposition with increasing number of sensors, arranged in a line.

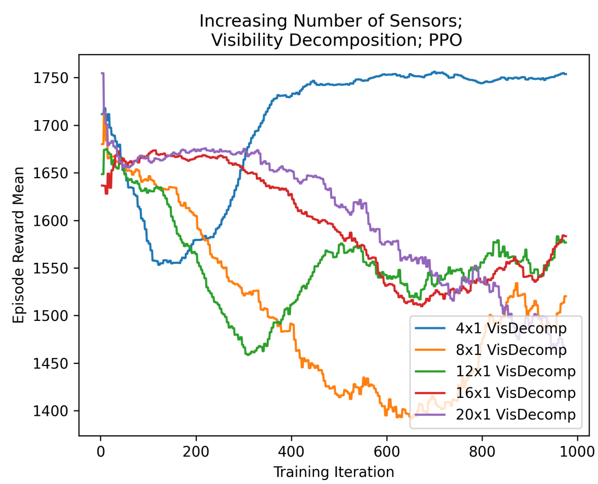


Figure 9. Visibility Decomposition with increasing number of sensors, arranged in a line.

most one sensor at any time step. Applying the Visibility decomposition (naively), we find a significant increase in the reward but at the cost of poor track custody. Instead, the Visibility decomposition with masking nearly doubled the track rate compared to its naive application, and the lights became much closer to 1. The best paradigm we studied was the NextStep decomposition, which showed the highest track rate and a lights on per step nearest to 1.

5. Discussion

Our experiments demonstrate that model-free RL methods like PPO and DQN degrade dramatically as the size, number, and arrangement of sensors increase. These scaling concerns for RL are real and significant obstacles to any attempt at autonomous multi-sensor resource management and data fusion. However, topological decompositions based on coincident detection in S_τ provides a framework for effective and efficient learning by reducing the dimension for training exploration. We found that naively applying decompositions improves baseline performance and mitigates the degradation of performance as the size of the problem grows, but it alone does not enable learning. When we enforced decompositions through action masking, we saw substantial performance gains (learning) over the baseline, achieving strong policies even as the number of sensors grows. By introducing the NextStep Decomposition, we found even more improvements in learning, especially in sensor arrays with more complicated geometry (grid compared to linear). Because of this, we believe it is possible to use the geometry of the problem to create additional decompositions which would further improve performance.

Expansion of this work should explore how these policies might perform in situations of sensor loss due to maintenance, miscalibration, interference, or adversarial attack. Moreover, the performance of this framework should be explored in multi-target tracking with heterogeneous sensor types, as well as with more dramatic geometric modifications to the sensor array.

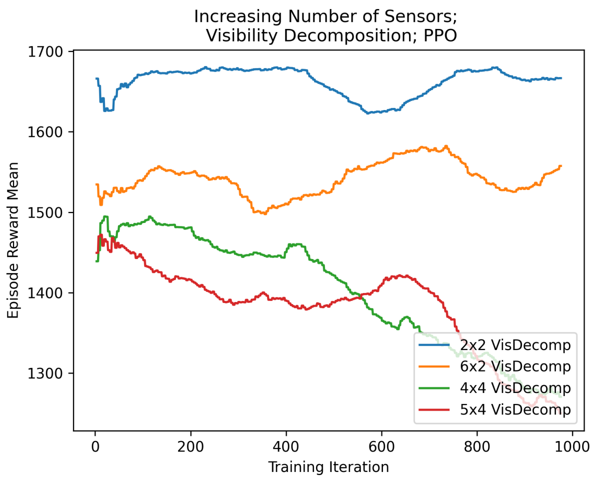


Figure 10. Visibility Decomposition with increasing number of sensors, arranged in a grid.

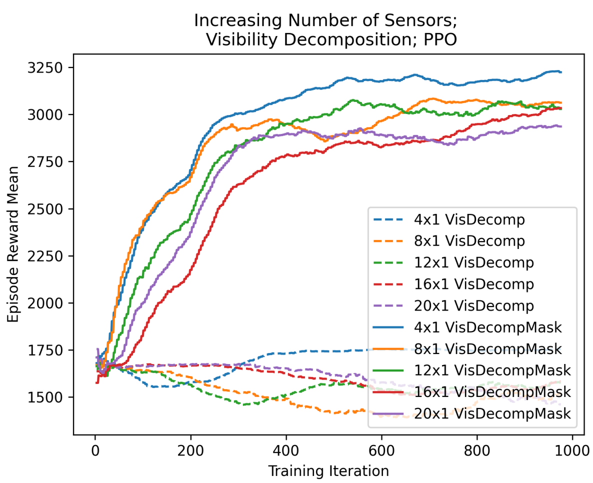


Figure 11. Visibility Decomposition with masking, for increasing number of sensors arranged in a line.

ACKNOWLEDGMENTS

All authors were supported on the "Shape Analytic and Reinforcement Learning Approaches to Closed Loop Sensing and Exploitation" project as part of the COLLABORATIVE SENSING & EXPLOITATION program funded by AFRL, under Contract # FA8750-22-C0025. We are very grateful to Dr. Peter Zulch from AFRL for technical guidance and discussions. This paper has been cleared for public release, case number AFRL-2023-4849.

REFERENCES

[1] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis, "Mastering the game of go with deep neural networks and tree search," *Nature*, vol. 529, pp. 484–503, 2016.

[2] D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel, T. Lillicrap, K. Simonyan, and D. Hassabis, "Mastering Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm," Dec. 2017, arXiv:1712.01815 [cs]. [Online]. Available: http://arxiv.org/abs/1712.01815

[3] O. Vinyals, I. Babuschkin, W. M. Czarnecki, M. Mathieu, A. Dudzik, J. Chung, D. H. Choi, R. Powell, T. Ewalds, P. Georgiev, J. Oh, D. Horgan, M. Kroiss, I. Danihelka, A. Huang, L. Sifre, T. Cai, J. P. Agapiou, M. Jaderberg, A. S. Vezhnevets, R. Leblond, T. Pohlen, V. Dalibard, D. Budden, Y. Sulsky, J. Molloy, T. L. Paine, C. Gulcehre, Z. Wang, T. Pfaff, Y. Wu, R. Ring, D. Yogatama, D. Wünsch, K. McKinney, O. Smith, T. Schaul, T. Lillicrap, K. Kavukcuoglu, D. Hassabis, C. Apps, and D. Silver, "Grandmaster level in StarCraft II using multi-agent reinforcement learning," *Nature*, vol. 575, no. 7782, pp. 350–354, Nov. 2019, number: 7782 Publisher: Nature Publishing Group. [Online]. Available: https://www.nature.com/articles/s41586-019-1724-z

[4] Meta Fundamental AI Research Diplomacy Team (FAIR)†, A. Bakhtin, N. Brown, E. Dinan, G. Farina, C. Flaherty, D. Fried, A. Goff, J. Gray, H. Hu, A. P. Jacob, M. Komeili, K. Konath, M. Kwon, A. Lerer, M. Lewis, A. H. Miller, S. Mitts, A. Renduchintala, S. Roller, D. Rowe, W. Shi, J. Spisak, A. Wei, D. Wu, H. Zhang, and M. Zijlstra, "Human-level play in the game of *Diplomacy* by combining language models with strategic reasoning," *Science*, vol. 378, no. 6624, pp. 1067–1074, Dec. 2022. [Online]. Available: https://www.science.org/doi/10.1126/science.ade9097

[5] OpenAI, C. Berner, G. Brockman, B. Chan, V. Cheung, P. Debiak, C. Dennison, D. Farhi, Q. Fischer, S. Hashme, C. Hesse, R. Józefowicz, S. Gray, C. Olsson, J. Pachocki, M. Petrov, H. P. d. O. Pinto, J. Raiman, T. Salimans, J. Schlatter, J. Schneider, S. Sidor, I. Sutskever, J. Tang, F. Wolski, and S. Zhang, "Dota 2 with Large Scale Deep Reinforcement Learning," Dec. 2019, arXiv:1912.06680 [cs, stat]. [Online]. Available: http://arxiv.org/abs/1912.06680

[6] D. Croon and G. C. H. E, "Drone-racing champions outpaced by AI," *Nature*, vol. 620, no. 7976, pp. 952–954, Aug. 2023, bandiera_abtest: a Cg_type: News And Views Number: 7976 Publisher: Nature Publishing Group Subject_term: Engineering, Machine learning. [Online]. Available: https://www.nature.com/articles/d41586-023-02506-8

[7] J. Degrave, F. Felici, J. Buchli, M. Neunert, B. Tracey, F. Carpanese, T. Ewalds, R. Hafner, A. Abdolmaleki, D. de las Casas, C. Donner, L. Fritz, C. Galperti, A. Huber, J. Keeling, M. Tsimpoukelli, J. Kay, A. Merle, J.-M. Moret, S. Noury, F. Pesamosca, D. Pfau, O. Sauter, C. Sommariva, S. Coda, B. Duval, A. Fasoli, P. Kohli, K. Kavukcuoglu, D. Hassabis, and M. Riedmiller, "Magnetic control of tokamak plasmas through deep reinforcement learning," *Nature*, vol. 602, no. 7897, pp. 414–419, Feb. 2022, number: 7897 Publisher: Nature Publishing Group. [Online]. Available: https://www.nature.com/articles/s41586-021-04301-9

[8] D. J. Foster, S. M. Kakade, J. Qian, and A. Rakhlin, "The Statistical Complexity of Interactive Decision Making," Jul. 2023, arXiv:2112.13487 [cs, math, stat]. [Online]. Available: http://arxiv.org/abs/2112.13487

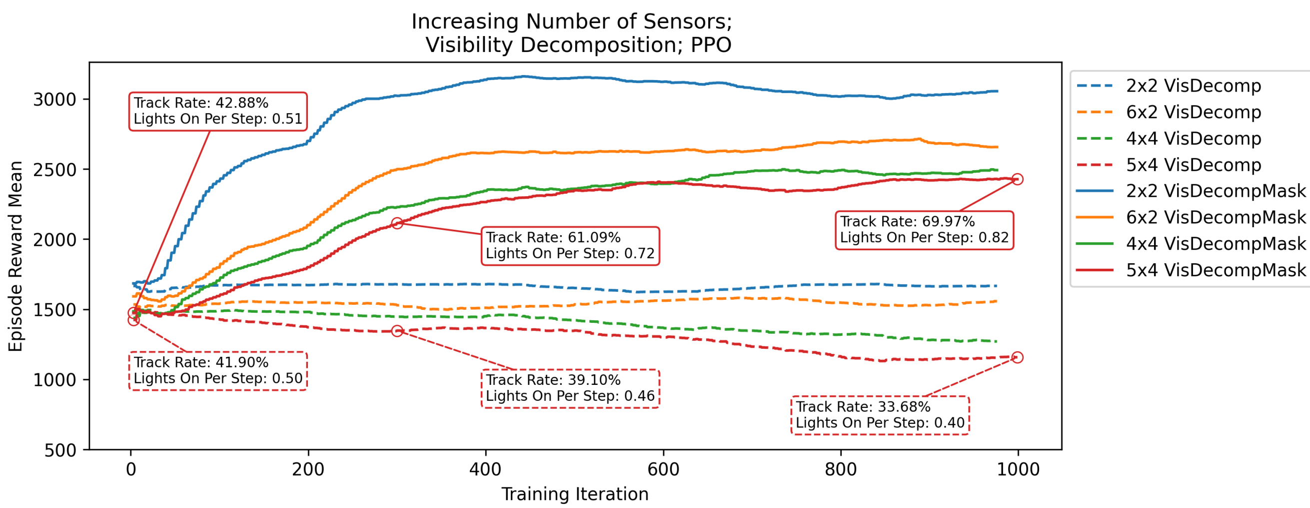


Figure 12. Visibility Decomposition with masking, for increasing number of sensors arranged in a grid.

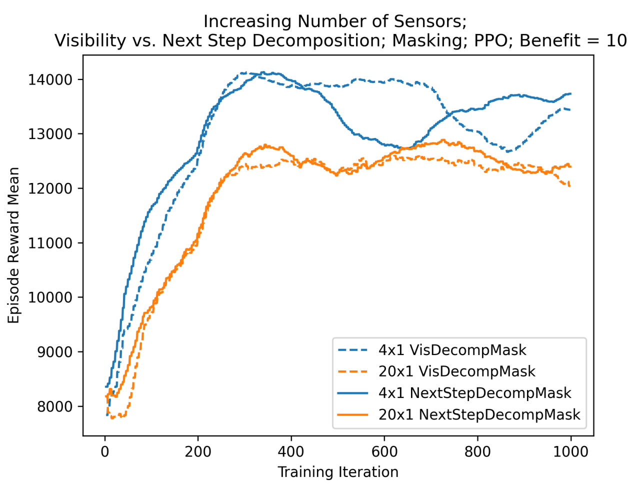


Figure 13. NextStep Decomposition with masking, for increasing number of sensors arranged in a line.

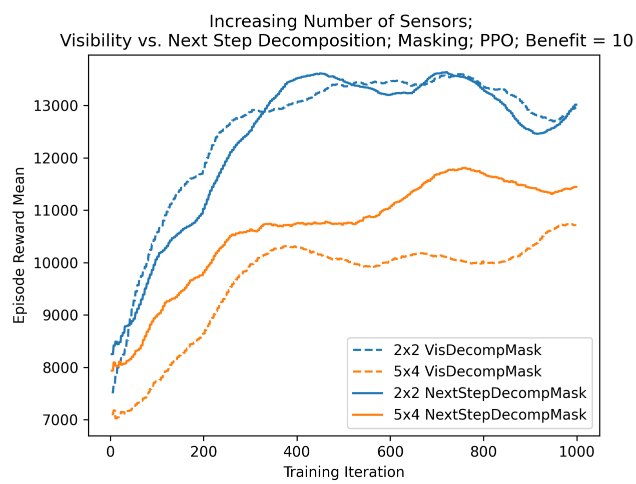


Figure 14. NextStep Decomposition with masking, for increasing number of sensors arranged in a grid.

[9] M. G. Azar, I. Osband, and R. Munos, "Minimax Regret Bounds for Reinforcement Learning," Jul. 2017, arXiv:1703.05449 [cs, stat]. [Online]. Available: http://arxiv.org/abs/1703.05449

[10] C. Bamford, S. Huang, and S. Lucas, "Griddly: A platform for AI research in games," Jul. 2022, arXiv:2011.06363 [cs]. [Online]. Available: http://arxiv.org/abs/2011.06363

[11] E. Liang, R. Liaw, R. Nishihara, P. Moritz, R. Fox, K. Goldberg, J. Gonzalez, M. Jordan, and I. Stoica, "RLlib: Abstractions for distributed reinforcement learning," in *Proceedings of the 35th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, J. Dy and A. Krause, Eds., vol. 80. PMLR, 10–15 Jul 2018, pp. 3053–3062. [Online]. Available: https://proceedings.mlr.press/v80/liang18b.html

[12] M. Towers, J. K. Terry, A. Kwiatkowski, J. U. Balis, G. d. Cola, T. Deleu, M. Goulão, A. Kallinteris, A. KG, M. Krimmel, R. Perez-Vicente, A. Pierré, S. Schulhoff, J. J. Tai, A. T. J. Shen, and O. G. Younis, "Gymnasium," Mar. 2023. [Online]. Available: https://zenodo.org/record/8127025

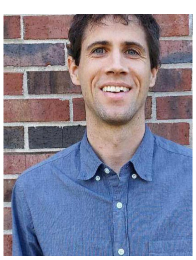
BIOGRAPHY

***Michael J. Catanzaro** is a Senior Scientist at GDA. He received his Ph.D. in Mathematics from Wayne State University in 2016, where his doctoral research involved stochastic and applied topology. He was an assistant professor at Iowa State University following a postdoctoral research position at the University of Florida. Dr. Catanzaro's recent work focuses on theoretical and applied aspects of topological data analysis (TDA). He has successfully applied TDA to the study of task modulation in fMRI data, iterated composition and fractals, and geometric aspects of multi-parameter persistent homology. He has also used topology to develop novel methods of studying exciton* scattering in physical chemistry and stochastic currents in thermodynamics.

Aaron Dharna is a doctoral student in Computer Science at the University of British Columbia. He did the research for this paper as a summer intern at GDA. He earned an M.S. in Data Science from Fordham University in 2020. Mr. Dharna's research centers around mathematical and practical aspects of meta-learning and reinforcement learning.

Jay Hineman is Chief Solutions Architect at GDA. Hereceived his Ph.D. in Mathematics from the University of Kentucky in 2012. At GDA, Dr. Hineman has applied his mathematical and computational background to integrating topological data analysis tools with machine learning techniques. He has focused on the domains of data fusion for targeting and control of system of systems for agile logistics and military medicine. He also serves as an adjunct instructor in the ECE Department at Duke University, where he leads classes about the implementation of machine learning and reinforcement learning at scale.

James B. Polly is Senior Scientist at GDA. He received his Ph.D. in 2016 from the City College of New York, where his research focused on midlatitude cyclones and the role these storms play in Earth's atmospheric energy budget. His research experience also includes numerical methods and simulation of fluids, and he has aerospace industry experience in propulsion and structural analysis. Dr. Polly is able to leverage multiple sources of remote sensing, reanalysis, and model data to inform a variety of research questions, and problems relating to the atmosphere and ocean are among his primary research interest.



Kevin McGoff is a Senior Mathematician at GDA, and is an Associate Professor of Mathematics at the University of North Carolina at Charlotte. He received his Ph.D. in Mathematics at the University of Maryland, where he specialized in dynamical systems and probability. He held a post-doctoral position at Duke University. Dr. McGoff's research interests involve understanding dynamical systems from several perspectives. From the probabilistic perspective, he analyzes the long-term behavior of systems whose rules of evolution are drawn at random. Taking a more statistical point of view, he seeks to provide rigorous performance bounds on statistical procedures for data with long-range dependence.

Abraham D. Smith is Chief Security Architect and Senior Mathematician at GDA, and is an Associate Professor in the Department of Mathematics, Statistics and Computer Science at University of Wisconsin-Stout, Wisconsin's Polytechnic University. He received his Ph.D. in Mathematics from Duke University in 2009 and held postdoctoral research positions at Fordham University and McGill University. Dr. Smith specializes in using geometric insight to reformulate open-ended data-analysis and machine-learning questions into firm mathematical theories, and then optimize those mathematical theories into concrete and efficient code. Dr. Smith is an avid scientific programmer and Linux administrator with long-time expertise across the entire software stack, and he uses these skills to design and implement parallel computing infrastructure for research code and production workflows at GDA.

Paul Bendich is Chief Scientist at GDA and is a Research Professor of Mathematics at Duke University. He received his Ph.D. in Mathematics from Duke in 2008 and held postdoctoral positions at the Institute for Science and Technology Austria and Penn State. Dr. Bendich's doctoral work laid some of the early theoretical foundations for topological data analysis (TDA). Since then, he has been at the forefront of the integration of TDA with more standard machine learning and statistical techniques. This work has found wide application in vehicle tracking, brain imaging, and image simplification, among many other areas.