DeepIG: Multi-Robot Information Gathering with Deep Reinforcement Learning

Alberto Viseras* and Ricardo Garcia†

I. Introduction

Information gathering (IG) with autonomous mobile robots has emerged as a prime alternative to gather information in situations that have a high risk for humans, like e.g. in search and rescue missions, and for applications in which it is desirable to reduce required time and manpower, like e.g. in environmental analysis. In such context, IG can clearly benefit from multi-robot coordination both in terms of efficiency to gather information and robustness against robotic failures.

Information gathering with multiple robots (MR-IG) has been studied for a wide range of applications such as surveillance, tracking, monitoring, to name only a few. In particular, in this paper we concentrate on MR-IG to monitor a physical process of interest like e.g. temperature, magnetic field, terrain profile, ozone concentration, etc.

State-of-the-art MR-IG algorithms belong to the family of model-based algorithms. Most widely used classes of algorithms are Gaussian processes (GPs) [1], [2], and Partially Observable Markov Decision Processes (POMDPs) [3]. Model-based algorithms assume an underlying model that describes physical properties of the process, such as spatial and temporal correlation, states' transition function, etc. Given a model, IG algorithms exploit it to derive MR coordination strategies that allow robots to gather information by optimizing some formal IG criterion.

Model-based approaches are designed to exploit properties of a particular model. This has the advantage of achieving a high performance in applications where the model accurately describes the observed process. In contrast, model-based approaches fail to gather information of processes that cannot be accurately described by existing models.

In a future robotics society, robots will need to solve new IG tasks, and, of course, many of them will not be described by existing models. This implies that we humans will have to invest our time and efforts to develop novel models and corresponding IG algorithms. Additionally, many of the new IG tasks will be too complex to be described by traditional models like aforementioned GPs or POMDPs. Nevertheless, we would like to be able to offer, rapidly and with limited effort, adequate algorithms to solve most of the MR-IG tasks.

*A. Viseras is with the Institute of Communications and Navigation of the German Aerospace Center (DLR), 82234, Oberpfaffenhofen, Germany alberto.viserasruiz@dlr.de

†R. Garcia is with the E.T.S.I. Telecomunicación at Universidad Politécnica de Madrid (UPM), 28040, Madrid, Spain ricardo.gpinel@alumnos.upm.es

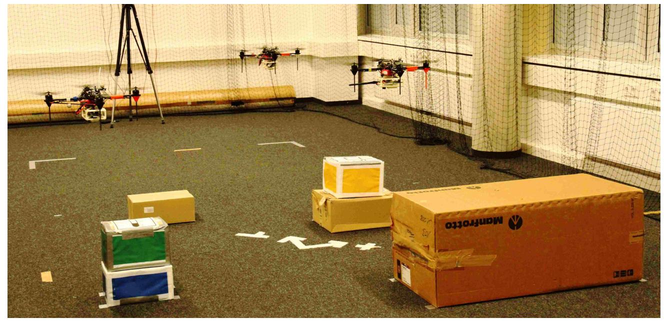


Fig. 1: Three quadcopters cooperating to map an unknown terrain profile that we built in our lab. Quadcopters run an instance of DeepIG, which uses deep reinforcement learning to teach robots how to gather information efficiently, while avoiding inter-robot collisions.

As we previously stated, model-based approaches will most likely fail to offer solutions to some of the future IG tasks. In contrast, reinforcement learning (RL) seems like a perfect fit to solve complex IG tasks. RL, in contrast to model-based approaches, does not make any assumption on the process. This has the advantage that computers can derive IG strategies, regardless of their complexity, with little human effort. Recently, RL has been exploited to solve a wide spectrum of robotic tasks, including control of a quadcopter [4], or robot navigation [5]. However, to the best of our knowledge, there is no algorithm in the literature that solves a MR-IG task using RL.

This gap in the state of the art motivates us to investigate the use of RL to monitor a physical process of interest with multiple robots. RL comprises multiple techniques to learn a mapping between robots' observations and robots' actions. In particular, here we opted for the use of Deep RL, which uses a deep neural network to implement this mapping.

Since the conception of Deep Q-Networks (DQN) algorithm [6], Deep RL has emerged as a powerful technique to handle complex sequential decision-making problems. Deep RL merges the capabilities of deep neural networks, which are able to process high-dimensional inputs and to make powerful representations, with the already successful, but limited to simpler problems, mathematical framework of RL. Deep RL has led to important breakthroughs like e.g. learning to play Atari video games [6].

Deep RL impressive breakthroughs motivate us to use it for MR-IG tasks. In fact, this corresponds to the essential contribution of our work: formulation of a MR-IG monitoring task as a Deep RL problem. This contribution comes to-

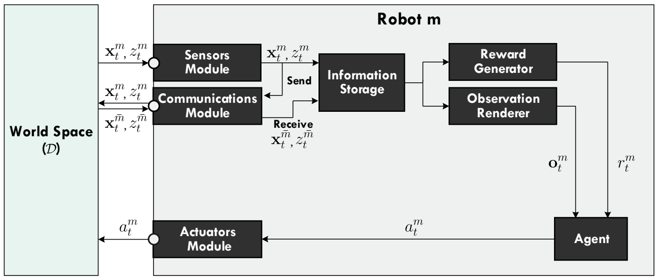


Fig. 2: Block diagram of DeepIG algorithm for an individual robot. Inter-robot cooperation is carried out through a Communications Module.

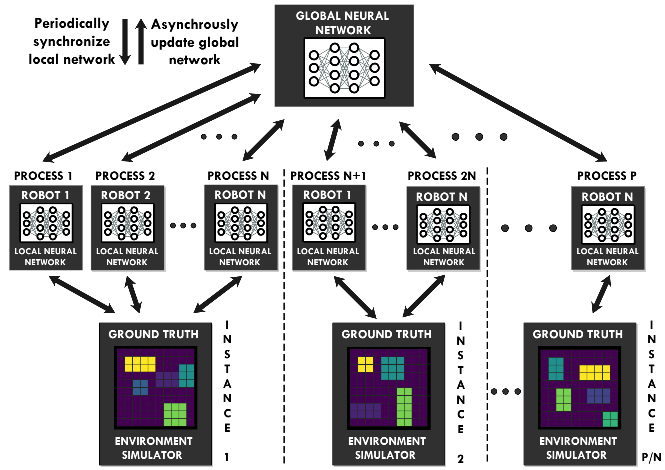


Fig. 3: MR-A3C block diagram.

gether with two additional sub-contributions that we propose in this paper. These are (i) the definition of a reward function that allows robots to gather information while avoiding inter-robot collisions, and (ii) the extension of a state-of-the-art Deep RL algorithm, Asynchronous Advantage Actor-Critic (A3C) [7], which permits learning with multiple robots.

The aforementioned contributions are the pillars of the algorithm we propose to solve MR-IG tasks. This algorithm we term it DeepIG. DeepIG is designed for tasks for which a model of the information of interest is too complicated to be derived by a human. Nevertheless, it is true that there are information distributions for which very accurate models have been derived. For example GPs have shown an outstanding performance for some IG tasks [1], [2], [8]. In order to exploit existing models, we propose in this paper an extension of DeepIG that permits incorporating existing models like e.g. GPs. This extension of DeepIG we term it model-based DeepIG (MB-DeepIG).

Let us remark that this paper is largely based on our work in [9]. Therefore, we refer the reader to [9] for additional details about DeepIG.

II. DeepIG Algorithm

DeepIG is a MR-IG algorithm. In particular, each individual robot runs in parallel an identical instance of DeepIG, and inter-robot coordination is done by means of inter-robot communication. In Fig. 2 we depict a block diagram of DeepIG. Next let us explain DeepIG in more detail.

Robots interact with the outside world through three modules: Sensors Module, Communications Module, and Actuators Module. Actuators translate a robot's planned action into a robot's movement. Sensors and communication (S&C) modules allow robots to obtain information about the world in which robots operate. In particular, at each time t, S&C modules provide each robot $m = 1, ..., N$ with process measurements z_t^m taken at positions $\mathbf{x}_t^m$, and with $\mathbf{x}_t^{\bar{m}}, z_t^{\bar{m}}$ from all $\bar{m} = 1, ..., N$ with $\bar{m} \neq m$.

Information collected with S&C modules is saved in an Information Storage module, and later processed by a Reward Generator and an Information Renderer. On the one hand, Reward Generator calculates a reward r_t^m, which allows the Agent to evaluate the positive/negative impact of its previous action. Our definition of reward allows us to incorporate a model of the information of interest, which is the basis of MB-DeepIG. For more details on the reward, we refer the reader to [9]. On the other hand, Information Renderer maps the output of the Information Storage into an Observation $\mathbf{o}_t^m$. In this paper, we represent $\mathbf{o}_t^m$ as an image, which is a powerful and compact container of information.

Our Agent takes as inputs r_t^m, $\mathbf{o}_t^m$. The Agent is the robot's brain, and it is the module that calculates next robot's action a_t^m. Here we consider four possible actions: $\{\uparrow, \leftarrow, \downarrow, \rightarrow\}$. Initially, we assume the Agent has no knowledge about how to cooperate with robots to gather information. Therefore we need to train the Agent. Of course, we could train robots in the real world. However, this is impractical as Deep RL algorithms typically require thousands/millions of executions to solve complex tasks like e.g. our MR-IG task. Therefore we first train the Agent in simulations with MR-A3C (see Fig. 3), and then transfer robots to the real world to solve actual MR-IG tasks.

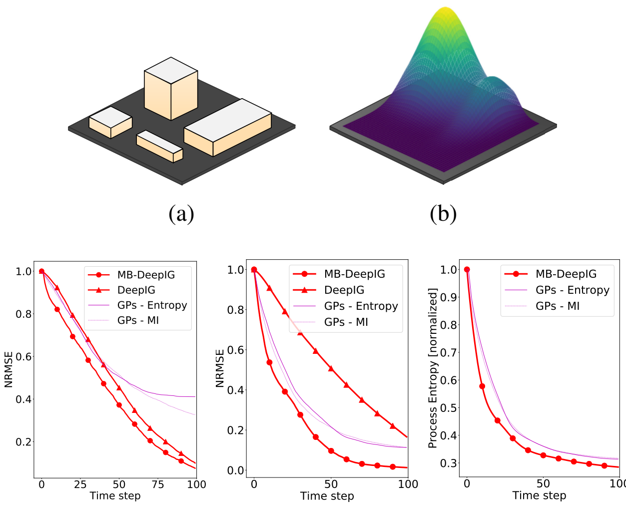
III. Simulations and Experimental Results

We evaluate DeepIG in simulations, and in an indoor experiment with three quadcopters that autonomously map an unknown terrain profile built in our lab.

A. Comparison Against Model-Based Strategies

We benchmark DeepIG and MB-DeepIG against two state-of-the-art GPs-based strategies: entropy-driven [2], and mutual-information-driven algorithms [8].

In Figs. 4c, 4d we depict the mean NRMSE between estimate and ground truth obtained for DeepIG, MB-DeepIG, entropy-driven, and MI-driven strategies. Additionally, we illustrate in Fig. 4e the posterior entropy, as calculated with the GP model, that remains about the process after the robot takes measurements. First conclusion that we can draw is that DeepIG outperforms entropy-driven, and MI-driven benchmarks for a boxes-like process (Fig. 4a), but not for the Gaussian-like one (Fig. 4b). As DeepIG does not use model information, it does not suffer from a model mismatch in the boxes-like process, and it does not benefit from a model



(a) (b)

(c) (d) (e)

Fig. 4: Evaluation for a single-robot system, and simulated IG tasks. (a, b) Example of a boxes- and a Gaussian-like process. (c) NRMSE for a boxes-like process; (d) NRMSE for a Gaussian-like process; (e) posterior entropy for a Gaussian-like process. We benchmark MB-DeepIG against DeepIG, entropy-driven [2], and MI-driven [8] strategies.

match in the Gaussian-like one. This argument is exactly the opposite for the two considered benchmarks, which explains the curves behaviour.

Next important fact that we can extract from Fig. 4 is that MB-DeepIG clearly outperforms GPs-Entropy and GPs-MI strategies in terms of NRMSE and posterior entropy. This demonstrates that MB-DeepIG was able to exploit the GPs model to learn a strategy that is more intelligent than the one used by the other benchmarks.

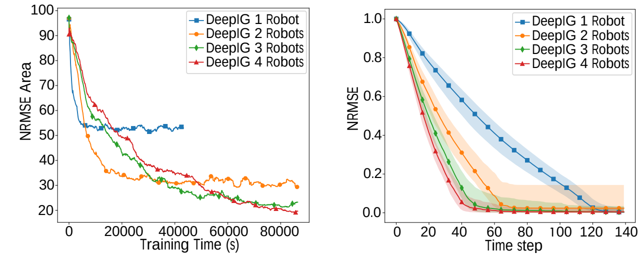
B. Scalability with Number of Robots

Next we analyze DeepIG scalability as we increase N from 1 up to 4 robots. In Fig. 5a we can observe that our learning metric, which corresponds to the mean of the NRMSE area over the training time, decreases as training time increases. This demonstrates that agents learn how to gather information. Moreover, we can also see that our learning metric reaches a lower value as N increases, which exemplifies a proper coordination between robots.

In a second step, we use the trained agents to evaluate DeepIG performance as N increases (see Fig. 5b). According to Fig. 5b we can conclude that NRMSE decreases faster as N increases. This performance gain is particularly noticeable between systems with 1 and 2 robots. We can also see that performance gain diminishes as N increases. The explanation for this behaviour is very simple: robots must avoid inter-robot collisions, which limits robots possible actions.

C. Experiments with 3 Quadcopters

We equipped three quadcopters with an ultrasound sensor facing down to map an unknown terrain profile using DeepIG (see Fig. 1). Quadcopters required 143, 70 and 59, time



(a) DeepIG learning curves. (b) Performance results.

Fig. 5: DeepIG evaluation for a multi-robot system. On the left hand side we show learning curves of DeepIG. On the right hand side we present performance results for a boxes-like process and for systems with $N = 1, 2, 3, 4$ robots.

steps with the 1, 2 and 3 robots systems, respectively. DeepIG was able to deal with measurement noise introduced by the ultrasound sensor. In fact, these results confirmed that DeepIG deep neural network could account for noisy measurements, although we did not include measurement noise during DeepIG training phase. In addition, experiments demonstrated that the policy that we learned offline in simulations can be translated to a real MR-IG task. Although we considered in this paper a toy environment and process, experiments results allow us to conclude that DeepIG was able to learn how to gather information with multiple aerial robots in a real environment.

We include in https://youtu.be/-aUUZPGiHlI a video that illustrates the DeepIG concept, the training process, and the 3 drones experiment.

References

[1] A. Singh, A. Krause, C. Guestrin, and W. J. Kaiser, "Efficient informative sensing using multiple robots," *Journal of Artificial Intelligence Research*, vol. 34, pp. 707–755, 2009.

[2] A. Viseras, T. Wiedemann, C. Manss, L. Magel, J. Mueller, D. Shutin, and L. Merino, "Decentralized multi-agent exploration with online-learning of gaussian processes," in *Robotics and Automation (ICRA), 2016 IEEE International Conference on*. IEEE, 2016, pp. 4222–4229.

[3] J. Capitan, M. T. Spaan, L. Merino, and A. Ollero, "Decentralized multi-robot cooperation with auctioned pomdps," *The International Journal of Robotics Research*, vol. 32, no. 6, pp. 650–671, 2013.

[4] J. Hwangbo, I. Sa, R. Siegwart, and M. Hutter, "Control of a quadrotor with reinforcement learning," *IEEE Robotics and Automation Letters*, vol. 2, no. 4, pp. 2096–2103, Oct 2017.

[5] J. Zhang, J. T. Springenberg, J. Boedecker, and W. Burgard, "Deep reinforcement learning with successor features for navigation across similar environments," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sept 2017, pp. 2371–2378.

[6] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. A. Riedmiller, "Playing atari with deep reinforcement learning," *CoRR*, vol. abs/1312.5602, 2013. [Online]. Available: http://arxiv.org/abs/1312.5602

[7] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," in *International conference on machine learning*, 2016, pp. 1928–1937.

[8] A. Krause, A. Singh, and C. Guestrin, "Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies," *Journal of Machine Learning Research*, vol. 9, no. Feb, pp. 235–284, 2008.

[9] A. Viseras and R. Garcia, "Deepig: Multi-robot information gathering with deep reinforcement learning," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 3059–3066, July 2019.