Research Statement

Hector Palacios*

May, 2018

My research interest is sequential decision making in meaningful *factored spaces*, which are also called *combinatorial domains*. Their states can be described by variables assigned to values in a discrete domain, such as $X \in \{a, b, c\}$ or $Position \in \{0, 1, 2, 3\}$. Their states can also be described by predicates grounded in discrete domains, such as $\text{Inside}(x, y)$ for $x \in \{a, b, c\}, y \in \{\text{truck}, \text{warehouse}\}$. My research considers the problem of *reaching a factored state that satisfies a goal*, such as $\text{Inside}(a, warehouse)$.

This research agenda should lead to effective methods for **applications in human situations**, such as business or manufacturing[1], including the transportation of goods, choosing diagnostic actions, composing web services, and coordinating actions between humans or agents.

Two fundamental areas of artificial intelligence (AI) are concerned with obtaining *controllers* for sequential decision making over atomic states [69; 57]. Model-based AI methods assume that an explicit transition function is given, thus making *Search* possible. Experience-based AI methods assume that a source of experience is available, thus making *Reinforcement Learning* (RL) possible. Fundamental algorithms for the factored case can employ any of these methods but also exploit the structure of the problem. For instance, state-of-the-art *planning* algorithms combine Search and heuristics synthetized from the given domain model [27].

My current research goal is to develop methods for **learning generalized controllers that reach a goal in meaningful factored spaces with imperfect information**. I intend to use **Reinforcement Learning and Function Approximation** for generalized planning but **without a planning model**, obtaining fixed-size-memory controllers[2]. Moreover, a single controller should operate in unseen situations of the same domain, especially larger situations.

My research focuses on a simple combinatorial domain where the factors have *loose dependency*, actions affect a few factors, and human intelligence is effective despite the size of the space. My contributions to acting under incomplete/imperfect information specialize in *islands of uncertainty with loose interactions between them*. I formalized such intuitions in my PhD thesis and other contributions, leading to both theoretical and empirical contributions[3].

Indeed, I was awarded the IJCAI-JAIR[4] Best Paper Prize 2012 for an *outstanding paper published in JAIR in the preceding five calendar years* for an article coauthored with my advisor, Hector Geffner [55]. My PhD dissertation was awarded the 2010 Best Dissertation Award by the International Conference on Automated Planning and Scheduling (ICAPS) and an Honorable Mention at the 2009 Artificial Intelligence Dissertation Award by the European Coordinating Committee for AI (ECCAI).

The methods I developed achieve great scalability while avoiding part of the exponential nature of acting in the presence of imperfect information. However, a controller produced by these techniques is an unbounded plan for a single problem, which remains in the agent memory during execution[5]. Moreover, these techniques require a detailed

*hectorpal@gmail.com. http://hectorpalacios.net

[1] We leave open the question of where the abstracted states come from. The issue is related to learning disentangled representations through acting, a problem I am also interested in. In any case, AI tools working at a given level of abstraction are very relevant for practical applications.

[2] While an unbounded search would be out of scope, fixed-size-memory controllers can benefit from a bounded lookahead, as it could mitigate dead-ends and improve the robustness of the behavior [65]. Lookahead requires a predictive model.

[3] Further details about my past contributions can be found in Appendix A.

[4] Joint award by the *International Joint Conferences on Artificial Intelligence* and the *Journal of Artificial Intelligence Research*.

[5] The plan can be exponential in the number of possible observations.

description of the actions. My current goal is to continue working on these problems but to use learning methods to simultaneously overcome these limitations.

1 Reinforcement Learning Challenges

Learning to reach goals in factored spaces can be formulated as RL, where achieving the goal leads to a high reward, as is done in RL-based methods for perfect information games [63]. However, some RL challenges become more relevant when dealing with discrete factored spaces.

For instance, there could be abundant *dead-end* states, where the goal cannot be achieved or can only be achieved at a very high cost. These states are difficult to explore effectively [59]; thus, learning predictive models becomes a key tool beyond a pure emphasis on policy search [64; 59; 71; 70, chap 8]. In some cases, dead-ends may need to be explicitly avoided as part of the task, as in *Safe RL* [3].

Hierarchical RL (HRL) is particularly relevant in factored spaces. State-of-the-art abstractions tend to distinguish low- and high-level behavior [58; 4], although other abstractions can also rely on distinctions like a reward structure [75; 45]. New abstractions based on discovering the loose dependency structure of a domain may be crucial for better scalability.

The domains of my interest are high dimensional, but their dependency structure remains loose as the number of factors increases. Thus, the methods I develop may be suitable for dead-ends or abstractions involving a few factors. In contrast, such contributions may not apply directly to more complex settings, such as puzzles, game playing (*e.g.,* Go or Poker), or the continuous control of robot joints. However, advances on my agenda are complementary to the general advances on discovering dead-ends, achieving better exploration and discovering abstractions.

2 Evaluation, Domains and Simulators

A diversified evaluation is important, as the goal of this research is to use the same algorithm and hyperparameters to obtain controllers for different problems. A simple initial idea is to use existing *planning models* as simulators for problems ranging from full observability to partial observability and probabilistic effects [34]. In that way, the benchmarks of the *International Planning Competition* can be as beneficial as the *Arcade Learning Environment* (ALE) but for reachability in factored spaces [49; 48][6]. These benchmarks are not just toy examples, as many of them have industrial application if the controllers are able to scale sufficiently well. Moreover, their size can be gradually and meaningfully increased, making systematic evaluation easier. This diversified evaluation differs from published work that evaluates general methods on a small number of domains or reports a reduced number of experiments due to the computational cost [31].

For instances, a simple but significant result would be learning a single general controller for the *Blocks World* problem, a classical model-based planning domain [66]. This domain consists of named blocks on a table with unlimited space, where blocks can be stacked/unstacked on top of each other or over the table. A general controller receives as input the state of the world and k blocks that must be stacked on top of each other. The controller then outputs the next action. A shared challenge for both model-based and learning areas is to develop general methods that can produce a controller for the *Blocks World* [12; 73].

3 Research problems

While my goal is to obtain generalized controllers under imperfect information, it is convenient to consider simplified problems and their specific challenges.

[6] Many planning domains have generators for a family of problems, although we may need to refine them further to study the capability of the methods.

3.1 Full Observability

The case of full observability and deterministic actions is worth considering by itself [63; 65]. The controller to be learned receives as input a fixed set of variables assigned to values taken from fixed domains. By allowing a goal to be set as part of the input, the controller should generalize across that class of goals. The controller output is an action to be applied and an internal representation of the state if learning a predictive model.

State-of-the-art RL with function approximation (Deep RL) is applicable immediately, although most recent work focuses on perceptual tasks, where the inputs are pixels contiguous to each other [49]. The present problem is arguably simpler, as factors are explicit and meaningful. However, a representation still needs to be learned, as the explicit factors by themselves may not be sufficient for solving the problem [63]. Current work using auxiliary tasks, count-based exploration and intrinsic motivation are relevant to our problem [75; 45; 42; 35; 4; 67; 6; 65]. Factored spaces may need specific regularization techniques relying on loose dependency, complementary to state-of-the-art ideas based on relative entropy of policies during learning [52].

If we consider probabilistic actions, this problem is a factored version of the most commonly studied in RL.

3.2 Variable Input for Generalizing to a Family of Problems

The task in this setting is to learn a controller given a fixed set of predicates and parametric actions. The input of the controller refers to such predicates but can contain arbitrarily new categorical objects where their information is assumed to be summarized by their relationships according to the predicates[7]. The controller output is an action, referring to the objects previously perceived. The structural assumptions made about the input are also present in recent work about transfer learning [38; 80; 29; 22; 61; 62; 51; 37, sect. 15.2].

One crucial question is what we can hope to achieve when feeding variable inputs into a fixed-size-memory controller. One can recall the notion of *indexicals*, a reference to an object but in an indirect way, such as *the block on top of the one I want to grasp* [17; 5]. Even if a situation comprises many objects, indexicals remark that effective compact representations are feasible by focusing on the task to be solved [12]. Task-oriented emerging representations are one of the main arguments in favor of deep learning and a central idea in the representations I introduced for model-based planning [8; 29; 49; 55; 1; 36; 25].

Relevant state-of-the-art methods, including *pointer networks*, memory-extended networks, and other RNN-based methods, can be used to accept and refer to variable inputs [77; 30; 68; 43]. Learning long-distance dependencies and being sensible to the input order are critical [44; 76; 78]. Graph embeddings are also relevant, as they are nongeometric and do not depend on the order or name of objects; however, such embeddings tend to rely on the local properties of the nodes in a graph [79].

3.3 Partial Observability

Let us start assuming that actions are deterministic and that the inputs are fixed-size factored observations. In this situation, the agent is forced to address its imperfect information about the environment without resorting to trial and error. While in the two previous settings, an experience is an action applied to a state, now an experience is an online execution, as each observation depends on all previous actions and an unknown initial state[8]. This setting can be formalized as a POMDP or as an MDP combined with a memory function to summarize the observation history. Our agenda requires such a function to use fixed-size memory.

My previous work in this setting relies on *islands of uncertainty with loose interactions between them*. The overall approach can be understood as mapping a POMDP into a synthesized MDP whose states consist of goal-dependent features. Effective MDPs may grow polynomially instead of exponentially with the number of variables. In addition, such MDPs can be both compact and complete, having the same solutions as the POMDP, as far as it has low *width*, a task-oriented measure of the degree of entanglement of the dependencies between latent variables [55; 1]. Such

[7] This is a consequence of the full-observability assumption.

[8] This setting can address single-player imperfect-information games, such as *Solitaire*, although that problem is out of scope [9].

features keep a compact summary of the observations, starting from the hidden and observable variables at the initial state.

Learning such a decoupled representation may sound as costly as learning a latent probabilistic distribution, but our focus should prevent us from paying such a cost. First, we assume that the latent structure has loose dependency. Second, even if the environment is very complex, achieving the goal may not need a complete model of the environment. Neither of these statements are true in situations where the imperfect information is very entangled, as in Poker.

Some ideas may come from tractable deep graphical models [41; 74]. Other relevant work avoids a single point value estimate for the complete information case using a fixed-size distributional representation [7; 19][9]. In general, this setting also calls for new forms of regularization for learning disentangled representations that are not only decomposable but have sparse dependency [72; 14].

4 Further Interests: Natural Language and Multiagent Settings

I am also interested in obtaining controllers for multiagent settings on the emergence of coordination under very low bandwidth for communication [47]. My main intuition is that most coordination relies on the simulation of the other agents and assumptions about their goals. At the same time, emerging coordination generates new questions regarding the emergence of language between bounded intelligent agents [40; 18; 50; 24; 46; 32].

I became interested in natural language during my recent experience as an industrial researcher. I am attracted to Natural Language Understanding (NLU) and goal-oriented dialogue. Although it is not usually presented in this way, I find both problems to be very close to acting under incomplete information with deterministic actions. For example, smooth recovery in front of a confusing situation is a major practical issue in dialogue. I am also interested in machine comprehension [60], especially in making a bridge between language in concrete domains and general language, a challenge I faced while working in domain-specific Question Answering with little data. This can be seen as a generalization from usual NLU task. Instead of mapping into a fixed domain-dependent interpretation, we can project the meaning of utterances into a restricted domain. This projection is a case of natural language inference and is related to embedding-based semantics [15; 23].

[9] This is related to sample-based approaches built on top of my work [16].

A Model-based Planning Contributions

This is a summary of my past contributions to model-based planning. My past work can be framed under model-based AI, characterized by the formal definition of the input and output in terms of semantics [26]. Model-based AI is agnostic about the actual technique for creating solver mapping input to output. In the case of planning, a solver receives a *complete model* and returns a controller called a *plan*, whose execution achieves the goal [27].

My main contributions regard solving rich model-based planning problems by translating them into simpler forms of planning. Thus, my work has been about models and algorithms as well as relationships between them, reductions, and translations. Moreover, I have leveraged whole bodies of research and software tools by relying on other AI models.

A.1 Full Observability and Deterministic Actions: Classical Planning

The simplest model-based setting is called *classical planning*, where there is an initial fully observable state, actions are deterministic, and the goal is a partial variable assignment. Classical planning is equivalent to a factored MDP but with deterministic actions. A classical plan is a sequence of actions that maps from the initial state into a goal state. I have used combinatorial optimization to develop a classical planner [53], but I have mainly used classical planning tools for more expressive settings.

A.2 Partial Observability by Translations into Full Observability

Models for planning with partial observability represent uncertainty in one of the following two ways: a probability distribution over the possible states (*e.g.,* POMDPs) or a set of possible states (called contingent planning). The two are closely related when policies are supposed to reach a goal with certainty.

My PhD thesis focused on conformant planning, that is the problem of computing a linear sequential plan that achieves a goal with total certainty but without using observations. While MDPs represent one extreme in the spectrum of planning with partial observability, full-state observability, conformant planning represents the other extreme as observations, if any, cannot be used for bifurcating. Policies for both MDP and conformant problems are simpler to express than policies for POMDPs. Conformant planning focuses on the fundamental problem of *tracking a sufficient amount of information about the current state* to realize that the goal has been achieved. In more realistic settings where observations are allowed, robust agents still need to explore possibilities or apply actions without fully knowing the state.

The main contribution of my dissertation was translating the problem of obtaining a conformant plan into the problem of obtaining a classical plan, compiling away the uncertainty. I developed tractability conditions that avoid an exponential translation if the *conformant width* –introduced in my work– was bounded [55]. This allows me to create a planner that proved, at running time, that most benchmarks only needed translations with a quadratic increase in size. This theoretical property helped me win the conformant track of the 5th *International Planning Competition* based on a number of problems solved under time and memory restrictions [28].

For partial observability, we leveraged the conformant translations to develop algorithms for offline planning and online action selection [1]. We introduced the notion of *contingent width* for characterizing cases where polynomial-size translations were complete. The approach required two translations, one into classical planning and another one into a nonprobabilistic MDP where nondeterministic actions account for the possible observation results. Follow-up work built based on this research includes sample-based translations for contingent planning and factored online belief tracking for POMDPs [16; 10; 11].

A.3 Non-Deterministic/Probabilistic actions

We introduced two algorithms for nondeterministic actions in conformant planning [2]. One relied on replanning when a nondeterministic effect appeared, as we did for partial observability, and the other was an incomplete translation

that does not need to plan again [1].

An algorithm for probabilistic conformant planning was introduced [33], leveraging my work of conformant planning using logical circuits [56; 21]. In this case, the task is to achieve the goal with a probability over a threshold, relying on probabilistic actions. Compilation into logical circuits is an important technique inference in discrete Bayesian networks [20].

A.4 Temporal Concurrent Planning

When actions have duration, concurrency can be a problem to avoid or a way to achieve a goal. We introduced translations from temporal planning into classical planning with cost, using features to represent the concurrent aspects of the problem [36; 25]

A.5 Generalizing Finite-state Controllers

For partial observability, we developed an algorithm that creates memoryless or finite-state controllers and demonstrated its scalability to unseen instances within a family of problems [13]. The mechanism is also able to create controllers for fully observable problems, as far as they provided observation tokens for grounding the states of the controller. The algorithm was based on former translations from planning into propositional logic and SAT [39; 54].

References

[1] Alexandre Albore, Hector Palacios, and Hector Geffner. A translation-based approach to contingent planning. In *Proc. IJCAI-09*, pages 1623–1628, 2009.

[2] Alexandre Albore, Hector Palacios, and Hector Geffner. Compiling uncertainty away in non-deterministic conformant planning. In *Proc. ECAI*, pages 465–470, 2010.

[3] Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*, 2016.

[4] Pierre-Luc Bacon, Jean Harb, and Doina Precup. The option-critic architecture. In *AAAI*, 2017.

[5] Dana H Ballard, Mary M Hayhoe, Polly K Pook, and Rajesh PN Rao. Deictic codes for the embodiment of cognition. *Behavioral and Brain Sciences*, 20(4):723–742, 1997.

[6] Marc Bellemare, Sriram Srinivasan, Georg Ostrovski, Tom Schaul, David Saxton, and Remi Munos. Unifying count-based exploration and intrinsic motivation. In *Advances in Neural Information Processing Systems*, pages 1471–1479, 2016.

[7] Marc G Bellemare, Will Dabney, and Rémi Munos. A distributional perspective on reinforcement learning. In *International Conference on Machine Learning*, pages 449–458, 2017.

[8] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

[9] Ronald Bjarnason, Alan Fern, and Prasad Tadepalli. Lower bounding klondike solitaire with monte-carlo planning. In *ICAPS*, 2009.

[10] Blai Bonet and Hector Geffner. Belief tracking for planning with sensing: Width, complexity and approximations. *Journal of Artificial Intelligence Research*, 50:923–970, 2014.

[11] Blai Bonet and Hector Geffner. Factored probabilistic belief tracking. In *IJCAI*, 2016.

[12] Blai Bonet and Hector Geffner. Features, Projections, and Representation Change for Generalized Planning. *arXiv preprint arXiv:1801.10055*, 2018. URL `https://arxiv.org/abs/1801.10055`.

[13] Blai Bonet, Hector Palacios, and Hector Geffner. Automatic derivation of memoryless policies and finite-state controllers using classical planners. In *Proc. ICAPS-09*, pages 34–41, 2009.

[14] Diane Bouchacourt, Emily Denton, Tejas Kulkarni, Honglak Lee, Siddharth N, David Pfau, and Josh Tenenbaum. Learning Disentangled Representations: from Perception to Control. NIPS 2017 Workshop. `https://sites.google.com/view/disentanglenips2017`, December 2017.

[15] Samuel R Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. A large annotated corpus for learning natural language inference. *arXiv preprint arXiv:1508.05326*, 2015.

[16] Ronen I Brafman and Guy Shani. A multi-path compilation approach to contingent planning. In *AAAI*, 2012.

[17] David Chapman. Penguins can make cake. *AI magazine*, 10(4):45, 1989.

[18] Edward Choi, Angeliki Lazaridou, and Nando de Freitas. Multi-agent compositional communication learning from raw visual input. *International Conference on Learning Representations*, 2018. URL `https://openreview.net/forum?id=rknt2Be0-`.

[19] Will Dabney, Mark Rowland, Marc G Bellemare, and Rémi Munos. Distributional reinforcement learning with quantile regression. In *AAAI*, 2018.

[20] Adnan Darwiche. *Modeling and reasoning with Bayesian networks*. Cambridge University Press, 2009.

[21] Adnan Darwiche and Pierre Marquis. A knowledge compilation map. *Journal of Artificial Intelligence Research*, 17:229–264, 2002.

[22] Misha Denil, Sergio Gómez Colmenarejo, Serkan Cabi, David Saxton, and Nando de Freitas. Programmable agents. In *Advances in Neural Information Processing Systems*, 2017.

[23] Manaal Faruqui, Jesse Dodge, Sujay K Jauhar, Chris Dyer, Eduard Hovy, and Noah A Smith. Retrofitting word vectors to semantic lexicons. *arXiv preprint arXiv:1411.4166*, 2014.

[24] Jakob Foerster, Yannis Assael, Nando de Freitas, and Shimon Whiteson. Learning to communicate with deep multi-agent reinforcement learning. In *Advances in Neural Information Processing Systems*, pages 2137–2145, 2016.

[25] Daniel Furelos-Blanco, Anders Jonsson, Hector Palacios, and Sergio Jimenez. Forward-search temporal planning with simultaneous events. In *ICAPS Workshop on Constraint Satisfaction Techniques for Planning and Scheduling*, 2018.

[26] Hector Geffner. The model-based approach to autonomous behavior: A personal view. In *AAAI*, 2010.

[27] Hector Geffner and Blai Bonet. *A Concise Introduction to Models and Methods for Automated Planning*. Morgan & Claypool Publishers, 2013.

[28] Alfonso Gerevini, Blai Bonet, and Bob Givan. Fifth international planning competition. *IPC06 Booklet*, 2006.

[29] Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*. MIT press Cambridge, 2016.

[30] Alex Graves, Greg Wayne, and Ivo Danihelka. Neural turing machines. *CoRR*, abs/1410.5401, 2014. URL `http://arxiv.org/abs/1410.5401`.

[31] Peter Henderson, Riashat Islam, Philip Bachman, Joelle Pineau, Doina Precup, and David Meger. Deep reinforcement learning that matters. In *AAAI*, 2018.

[32] Felix Hill, Karl Moritz Hermann, Phil Blunsom, and Stephen Clark. Understanding grounded language learning agents. *arXiv preprint arXiv:1710.09867*, 2017.

[33] Jinbo Huang et al. Combining knowledge compilation and search for conformant probabilistic planning. In *ICAPS*, pages 253–262, 2006.

[34] IPC website. International Planning Competition – International Conference on Automated Planning and Scheduling. http://www.icaps-conference.org/index.php/Main/Competitions.

[35] Max Jaderberg, Volodymyr Mnih, Wojciech Marian Czarnecki, Tom Schaul, Joel Z Leibo, David Silver, and Koray Kavukcuoglu. Reinforcement learning with unsupervised auxiliary tasks. In *International Conference on Learning Representations (ICLR)*, 2017.

[36] Sergio Jiménez, Anders Jonsson, and Héctor Palacios. Temporal planning with required concurrency using classical planning. In *Proceedings of the 25th International Conference on Automated Planning and Scheduling (ICAPS)*, 2015.

[37] Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Judy Hoffman, Fei-Fei Li, C. Lawrence Zitnick, and Ross B. Girshick. Inferring and executing programs for visual reasoning. *CoRR*, abs/1705.03633, 2017. URL http://arxiv.org/abs/1705.03633.

[38] Ken Kansky, Tom Silver, David A. Mély, Mohamed Eldawy, Miguel Lázaro-Gredilla, Xinghua Lou, Nimrod Dorfman, Szymon Sidor, D. Scott Phoenix, and Dileep George. Schema networks: Zero-shot transfer with a generative causal model of intuitive physics. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, pages 1809–1818, 2017. URL http://proceedings.mlr.press/v70/kansky17a.html.

[39] Henry Kautz and Bart Selman. Pushing the envelope: Planning, propositional logic, and stochastic search. In *Proc. AAAI*, pages 1194–1201, 1996.

[40] Satwik Kottur, José M. F. Moura, Stefan Lee, and Dhruv Batra. Natural language does not emerge 'naturally' in multi-agent dialog. *CoRR*, abs/1706.08502, 2017. URL http://arxiv.org/abs/1706.08502.

[41] Rahul G Krishnan, Uri Shalit, and David Sontag. Structured inference networks for nonlinear state space models. In *AAAI*, pages 2101–2109, 2017.

[42] Tejas D Kulkarni, Karthik Narasimhan, Ardavan Saeedi, and Josh Tenenbaum. Hierarchical deep reinforcement learning: Integrating temporal abstraction and intrinsic motivation. In *Advances in neural information processing systems*, pages 3675–3683, 2016.

[43] Karol Kurach, Marcin Andrychowicz, and Ilya Sutskever. Neural random access machines. *ICLR*, 2016. URL http://arxiv.org/abs/1511.06392.

[44] Brenden M Lake and Marco Baroni. Still not systematic after all these years: On the compositional skills of sequence-to-sequence recurrent networks. *arXiv preprint arXiv:1711.00350*, 2017.

[45] Romain Laroche, Mehdi Fatemi, Harm van Seijen, and Joshua Romoff. Multi-advisor reinforcement learning. 2017. URL https://www.microsoft.com/en-us/research/publication/multi-advisor-reinforcement-learning/.

[46] Angeliki Lazaridou, Alexander Peysakhovich, and Marco Baroni. Multi-agent cooperation and the emergence of (natural) language. In *International Conference on Learning Representations (ICLR)*, 2017.

[47] Adam Lerer and Alexander Peysakhovich. Maintaining cooperation in complex social dilemmas using deep reinforcement learning. *CoRR*, abs/1707.01068, 2017. URL http://arxiv.org/abs/1707.01068.

[48] Marlos C Machado, Marc G Bellemare, Erik Talvitie, Joel Veness, Matthew Hausknecht, and Michael Bowling. Revisiting the arcade learning environment: Evaluation protocols and open problems for general agents. *Journal of Artificial Intelligence Research*, 61:523–562, 2018.

[49] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529, 2015.

[50] Igor Mordatch and Pieter Abbeel. Emergence of grounded compositional language in multi-agent populations. *arXiv preprint arXiv:1703.04908*, 2017. URL http://arxiv.org/abs/1703.04908.

[51] Arvind Neelakantan, Quoc V. Le, and Ilya Sutskever. Neural programmer: Inducing latent programs with gradient descent. In *International Conference on Learning Representations (ICLR)*, 2015.

[52] Gergely Neu, Anders Jonsson, and Vicenç Gómez. A unified view of entropy-regularized markov decision processes. In *Deep Reinforcement Learning Symposium, NIPS*, 2017.

[53] Hector Palacios and Hector Geffner. Planning as branch and bound: A constraint programming implementation. In *Proc. XXVIII Conf. Latinoamericana de Informática*, pages 239–251, 2002.

[54] Hector Palacios and Hector Geffner. Mapping conformant planning into sat through compilation and projection. In *Conference of the Spanish Association for Artificial Intelligence - Current Topics in Artificial Intelligence*, pages 311–320. Springer, 2006. ISBN 978-3-540-45915-6.

[55] Hector Palacios and Hector Geffner. Compiling Uncertainty Away in Conformant Planning Problems with Bounded Width. *Journal of Artificial Intelligence Research*, 35:623–675, 2009.

[56] Hector Palacios, B. Bonet, Adnan Darwiche, and Hector Geffner. Pruning conformant plans by counting models on compiled d-DNNF representations. In *Proc. of the 15th Int. Conf. on Planning and Scheduling (ICAPS-05)*, 2005.

[57] Judea Pearl. *Heuristics*. Addison Wesley, 1983.

[58] Doina Precup. *Temporal abstraction in reinforcement learning*. University of Massachusetts Amherst, 2000.

[59] Sébastien Racanière, Théophane Weber, David Reichert, Lars Buesing, Arthur Guez, Danilo Jimenez Rezende, Adrià Puigdomènech Badia, Oriol Vinyals, Nicolas Heess, Yujia Li, et al. Imagination-augmented agents for deep reinforcement learning. In *Advances in Neural Information Processing Systems*, pages 5694–5705, 2017.

[60] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*, 2016.

[61] Scott Reed and Nando De Freitas. Neural programmer-interpreters. In *International Conference on Learning Representations (ICLR)*, 2016.

[62] Tim Rocktäschel and Sebastian Riedel. End-to-end differentiable proving. In *Advances in Neural Information Processing Systems 30*, pages 3791–3803, 2017. URL `http://papers.nips.cc/paper/6969-end-to-end-differentiable-proving.pdf`.

[63] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484–489, 2016.

[64] David Silver, Hado Hasselt, Matteo Hessel, Tom Schaul, Arthur Guez, Tim Harley, Gabriel Dulac-Arnold, David Reichert, Neil Rabinowitz, Andre Barreto, et al. The predictron: End-to-end learning and planning. In *International Conference on Machine Learning*, pages 3191–3199, 2017.

[65] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of Go without human knowledge. *Nature*, 550(7676):354, 2017.

[66] John Slaney and Sylvie Thiébaux. Blocks world revisited. *Artificial Intelligence*, 125(1-2):119–153, 2001.

[67] Alexander L Strehl and Michael L Littman. An analysis of model-based interval estimation for markov decision processes. *Journal of Computer and System Sciences*, 74(8):1309–1331, 2008.

[68] Sainbayar Sukhbaatar, arthur szlam, Jason Weston, and Rob Fergus. End-to-end memory networks. In *Advances in Neural Information Processing Systems 28*, 2015. URL `http://papers.nips.cc/paper/5846-end-to-end-memory-networks.pdf`.

[69] Richard S. Sutton and Andrew G. Barto. *Introduction to Reinforcement Learning*. MIT Press, 1998.

[70] Richard S. Sutton and Andrew G. Barto. *Introduction to Reinforcement Learning*. MIT Press, 2018. URL `http://incompleteideas.net/book/the-book-2nd.html`. March 2018 draft of the 2nd Edition.

[71] Aviv Tamar, Yi Wu, Garrett Thomas, Sergey Levine, and Pieter Abbeel. Value iteration networks. In *Advances in Neural Information Processing Systems*, pages 2154–2162, 2016.

[72] Valentin Thomas, Emmanuel Bengio, William Fedus, Jules Pondard, Philippe Beaudoin, Hugo Larochelle, Joelle Pineau, Doina Precup, and Yoshua Bengio. Disentangling the independently controllable factors of variation by interacting with the world. In *Learning Disentangled Representations: From Perception to Control Workshop, NIPS*, 2017.

[73] Sam Toyer, Felipe Trevizan, Sylvie Thiebaux, and Lexing Xie. Action schema networks: Generalised policies with deep learning. In *ICAPS*, 2017.

[74] Raquel Urtasun. Introducing Structure in Deep Learning. `https://www.robots.ox.ac.uk/seminars/Extra/2016_09_19_RaquelUrtasun.pdf`, 09 2016.

[75] Harm van Seijen, Mehdi Fatemi, Josh Romoff, and Romain Laroche. Improving scalability of reinforcement learning by separation of concerns. 2017. URL `https://www.microsoft.com/en-us/research/publication/improving-scalability-reinforcement-learning-separation-concerns/`.

[76] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 6000–6010, 2017.

[77] Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. Pointer networks. In *Advances in Neural Information Processing Systems*, pages 2692–2700, 2015.

[78] Oriol Vinyals, Samy Bengio, and Manjunath Kudlur. Order matters: Sequence to sequence for sets. In *International Conference on Learning Representations (ICLR)*, 2016. URL `http://arxiv.org/abs/1511.06391`.

[79] Zhilin Yang, William W Cohen, and Ruslan Salakhutdinov. Revisiting semi-supervised learning with graph embeddings. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning-Volume 48*, pages 40–48. JMLR. org, 2016.

[80] Rowan Zellers and Yejin Choi. Zero-shot activity recognition with verb attribute induction. *EMNLP*, 2017.