Embodied Intelligent Agents with Cognitive Conscious and Unconscious Reasoning

Matthew Conforth and Yan Meng
Electrical and Computer Engineering Department
Stevens Institute of Technology
Hoboken, NJ 07030, USA
matthew.conforth@stevens.edu, yan.meng@stevens.edu

***Abstract*—The CIVS (Civilization-Inspired Vying Societies) system is a novel evolutionary learning multi-agent system that uses artificial life methods to produce highly-capable artificial intelligence agents proficient in one or more complex tasks as well as more general adaptability, reasoning, and survivability in dynamic, unpredictable environments. A new cognitive architecture called CHARISMA is being developed as the brain of social agents within the CIVS system. This paper mainly focuses on the development of conscious reasoning and unconscious reasoning mechanisms as well as the interactions between them for the CHARISMA cognitive architecture. Our conscious reasoning design depends on recent advances in Markov logic network algorithms, in particular the online structure learning algorithm. Our unconscious reasoning design is based on Cartesian genetic programming, which we have adapted to work with the semantic hyper network data structure (used for knowledge representation). A novelty-based evolutionary strategy is employed so that the unconscious reasoning is focused on discovering what Markov logic network based conscious reasoning may likely miss. This approach allows interactions between conscious and unconscious reasoning that would not have been possible in the past, and which we believe are vital to achieving more adaptable and creative problem-solving.**

Keywords-cognitive architecture; conscious reasoning; Markov logic networks; unconscious reasoning

I. Introduction

The interactions between conscious and unconscious reasoning are vital to human brain thought, perception, and action. For example, in order to read this sentence your brain needs to classify visual patterns into letters and words, recall the possible meanings of those words, select the particular meaning for each word that best fits the current context, and only then can the sentence itself be understood. Most likely, you are not consciously aware of this entire process. The work is mostly done unconsciously, and it is only at the level of meaning that your conscious mind gets involved. A child who is just learning to read, on the other hand, needs to consciously think through this process, and possibly add intermediate steps such as vocally sounding out the words. There is much more to unconscious reasoning then just automating routine tasks. The discovery of novel or counterintuitive solutions, and creative thinking in general, usually relies heavily on unconscious reasoning.

To sidestep the debates about machine consciousness and strong vs. weak artificial intelligence, for our purposes in the context of an artificial agent "conscious reasoning" should be taken to mean deliberative, goal-driven computation based on logic and statistics, whereas "unconscious reasoning" should be taken to mean exploratory, novelty-driven computation based on context and not constrained by rules. Leveraging the interactions between these very different kinds of reasoning will enable significant improvements in the cognitive capabilities of artificial agents, especially in the context of understanding and interacting with human users. Adding computational cognitive reasoning capabilities to robotic and software systems will facilitate more intuitive and enhanced human-system interfaces, e.g. by incorporating cognitively plausible representations and qualitative reasoning. With more commonality of reasoning, knowledge, and assumptions between intelligent systems and the humans that interact with them, these systems will be capable of more efficient operation and undertaking more challenging tasks. Human-robot interaction [1] and human-robot collaboration [2] research has demonstrated this mutually-beneficial progression. This is especially important when dealing with non-technical users, as in elderly care systems, tour guide robots, etc., or when operating in stressful conditions, such as search & rescue, disaster relief, military applications, etc.

To develop such a generally intelligent agent system where each agent in the system has cognitive reasoning capabilities and can naturally interact with humans is a challenging task. To tackle this challenge, we are developing the CIVS (Civilization-Inspired Vying Societies) system, which is a novel evolutionary learning multi-agent system loosely inspired by the history of human civilization. It uses a bottom-up artificial life approach to ultimately produce complex desired behaviors in agents and agent groups. Taking inspiration from human history, we believe that agent's intrinsic motivations and social interactions between agents are the keys to truly open-ended developmental progress. In this way, agent fitness is primarily determined by the self-improvement, the competition and cooperation among agents and agent groups. Thus, it is impossible to settle on a "good enough" configuration, since the bar for success is constantly being raised by the progress of other agents. CIVS is designed to generate artificial agents that are inherently social in how they think, learn, adapt, and operate. Through this method, the agents and agent systems have the potential to be more adaptable and generally intelligent than existing systems.

To achieve our goals in the CIVS system, we have proposed the CHARISMA (Context Hierarchy-based Adaptive ReasonIing Self-Motivated Agent) cognitive architecture in our previous work [3], which serves as the brain of a social agent in CIVS. The overall design of the CHARISMA cognitive architecture was inspired by Baars' Global Workspace Theory [4]. In this paper we mainly focus on the development of a new cognitive framework of conscious and unconscious reasoning mechanisms, and the interactions between conscious and unconscious reasoning. We will only provide the general architecture of this cognitive reasoning framework in this paper. More specific applications and experimental results will be provided in our future papers.

The rest of the paper is organized as follows. Section II provides relevant background information, such as the general description of the CIVS system, the CHARISMA cognitive architecture, and core reasoning in CHARISMA. The design of CHARISMA's conscious reasoning is explained in Section III. Section IV describes the CHARISMA's unconscious reasoning method. The interaction between the conscious and unconscious reasoning is discussed in Section V. Section VI gives the conclusion and describes future work.

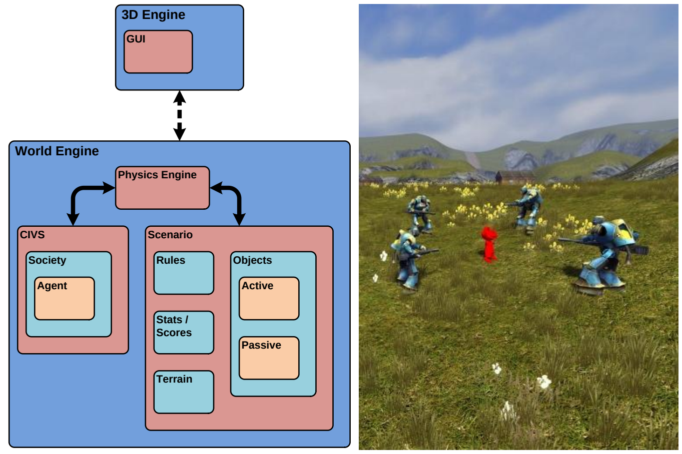
II. Background

A. Cognitive Architectures

Most cognitive architectures attempt to replicate certain behavioral and/or structural properties observed in humans, though at varying levels of abstraction. Since Newell argued for the importance of a unified theory of cognition [5], different cognitive architectures have been proposed. A recent survey of cognitive architectures is provided in [6]. The ACT-R [7] family of cognitive architectures emphasizes human psychological verisimilitude and has been used to model nuances of human memory and attention. The CLARION [8] cognitive architecture focuses on the distinction between implicit and explicit processes and the interactions between them with its dual representational hybrid structure. The ICARUS [9] cognitive architecture is primarily concerned with behavior in embodied agents; it emphasizes perception and action over abstract problem solving. The LIDA [10] cognitive architecture is based on Global Workspace Theory and focuses on neuropsychologically plausible roles of consciousness in cognition. Soar [11] began as a pure symbolic production system, and has since been enhanced and extended with specialized modules for a wide variety of different capabilities.

B. The CIVS System

The CIVS system is a multi-agent system designed to produce intelligent emergent behaviors and broad adaptability in groups of social learning agents. The general composition of the CIVS system is shown in Fig. 1(a). It consists of a World Engine which is responsible for running the simulation, and the 3D Engine which allows human observation and interaction (via the GUI) with the simulation. We have developed an embedded 3D CIVS simulator which allows human observation and interaction with the agents. Fig. 1(b) shows one snapshot of the CIVS simulator. The environment used for CIVS is a complex 3D simulated world governed by realistic physics to present the agents with a complex, challenging world. Different scenarios may add additional constraints and possibilities to the system.



(a) (b)

Figure 1. (a) The framework of the overall CIVS simulation system. (b) One screenshot of multiple agents in the CIVS simulator (four blue agents with arms cooperate to surround one red agent in the middle).

C. The CHARISMA Cognitive Architecture

The CHARISMA model is inspired by the Global Workspace Theory [4]. In this model, the role of consciousness is to broadcast the relatively small chunk of information that is currently deemed most important to a host of unconscious faculties. Coalitions of unconscious processes work together to extract information, make associations, decompose problems, etc.

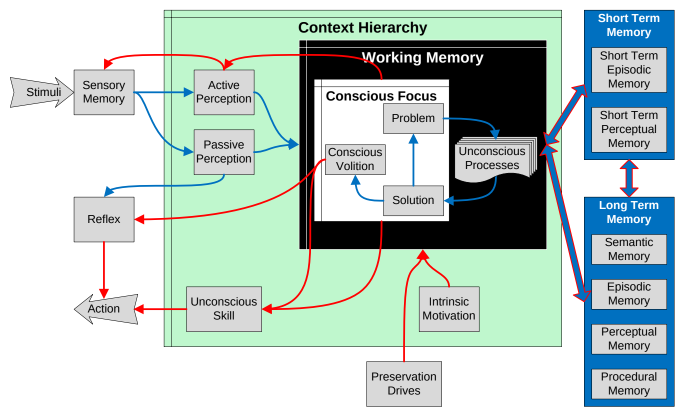


Figure 2. The framework of the CHARISMA cognitive architecture. Blue connectors represent flows that are primarily information, whereas red connectors represent flows that are primarily control signals.

As shown in Fig. 2, the general logic flow of the CHARISMA is as follows. Stimuli represent all the incoming sensory data, which will be briefly cached in sensory memory to allow active and passive perception to extract the important information and integrate it into the context hierarchy. Active perception will focus on what is most relevant to the subject currently in conscious focus, whereas passive perception will look for sudden or unexpected changes that might trigger a reflex action. Reflex will potentially short-circuit the conscious thought process, allowing very rapid response to unambiguous stimuli. The context hierarchy is the robot brain's hierarchical model of the state of the world. The contexts at the top layers are related to self, such as "self as agent" and "self as knower" which are fairly permanent. From there, the hierarchy will gradually progress from long-term to short-term contexts, with immediate situational details at the bottom. The context hierarchy will provide an organized representation of the known state of the world (including the robot's own internal state) to all the systems within the green box.

The potential knowledge/skills that CHARISMA robots need to explore are very large and high-dimensional in open-ended environments. Therefore, how to explore new knowledge/skills and what to learn are very critical questions for the robots. The robot's priorities will be determined by a 3-tier motivational system composed of preservation drives (hunger, pain, etc.), intrinsic motivation (curiosity, fun, flow, etc.), and conscious volition (deliberate plans). The intrinsic motivation subsystem will serve to prioritize learning paths based on the robot's ability to make meaningful progress along that path, which naturally leads to chains of progress. Essentially, the motivational system will motivate the robot to improve its skills and knowledge in the current context, or seek a new context if the current one has become boring (i.e., improvement is too slow). In this way, the robot will be able to make incremental progress toward complex, difficult abilities that it could never have learned directly. Meanwhile, intrinsic motivation will also make the robot's learning self-directed and fully autonomous. In addition, based on the preservation drives, the robot will create abstract needs that were found supportive of its previously established needs. Since abstract needs may be used to create new needs as well, a complex system of internal drives will emerge. Motivations resulting from these drives will compete to select intended actions and will support the robot's cognitive process of attention, planning and action monitoring. Please refer to our previous paper [3] for the details of CHARISMA architecture.

D. *Reasoning Core in CHARISMA*

The white square in Fig. 2 designates the conscious components of the CHARISMA cognitive architecture. Conscious focus governs the agent's focus of attention. In a sense, conscious focus acts as a filter, so that only the most important information from working memory enters the agent's consciousness. What is important at any given time and the agent's goals are governed by the agent's motivation systems, which we must gloss over in this paper. A separate paper will explore the agent motivation systems in detail.

The core of agent reasoning is the problem → unconscious processes → solution loop. The problem subsystem is responsible for providing a coherent representation of the task, question, objective, etc. that currently has conscious focus. This has an organizational role, and also serves as a sort of preprocessing for the myriad unconscious processes. The unconscious processes are the suite of unconscious abilities, skills, intuitions, etc. that have been learned and developed by the agent. Unconscious processes are evolved using Cartesian genetic programming. Each unconscious process is fairly simple on its own, providing at most one specific service. Unlike the conscious components, the unconscious processes have direct access to working memory. The solution subsystem interprets the consensus/solution(s) constructed by the unconscious processes from the current problem. The solution subsystem is also an important component of agent long-term learning, since it monitors the expected results of the current solution, which will be needed for future comparisons with observed results.

Conscious volition can influence and even outright overrule other subsystems, such as denying a reflex action or overriding the contextual attention priorities that would otherwise determine what information enters conscious focus. Action represents any attempt the robot makes to do something tangible in the environment, such as changing its own position, manipulating an object, interacting with another robot, etc.

III. Conscious Reasoning in CHARISMA

A. *Markov Logic Networks*

Statistical relational learning [13] deals with machine learning in domains exhibiting both uncertainty and complex relational structures. Markov Logic Networks (MLNs) [14] are a relatively new approach to statistical relational learning that generalizes both full first-order logic and Markov networks. An MLN consists of a set of weighted clauses in first-order logic. This weighting softens first-order logic by making situations where some clauses are unsatisfied less likely but not impossible. An MLN functions as a template for constructing ground Markov networks. Grounding is the assignment of constants from the current context to the first-order logic variables. Different groundings (i.e. sets of constants) will produce different ground networks, potentially of widely varying size. However, all the ground networks produced by the same MLN will have certain regularities in structure and parameters, determined by the MLN.

In formal terms, given an MLN, the probability of a possible world, x, is defined as:

$$P(X=x)=\frac{1}{Z}\exp\left(\sum_{c_i\in C} w_i n_i(x)\right) \qquad (1)$$

$$Z=\sum_{x\in X}\exp\left(\sum_{c_i\in C} w_i n_i(x)\right) \qquad (2)$$

where X is the set of all ground atoms, C is the set of all clauses in the MLN, w_i is the weight of clause $c_i \in C$, and $n_i(x)$ is the number of true groundings of c_i in x. Z is just the normalization constant.

B. *Online Structure Learning*

Online Structure Learning (OSL) [15], recently proposed by Huynh and Mooney, is the first algorithm for MLNs that performs online learning of both the structure (i.e. clauses) and the parameters (i.e. weights). All earlier methods for learning the structure of an MLN [16-20] are batch algorithms, and thus are infeasible to use for a learning agent operating in a dynamic environment. OSL utilizes online max-margin structure learning with mode-guided relational path-finding and online max-margin l_1-regularized weight learning.

C. *Conscious Reasoning*

Ideally, at any given time t_i, the agent should select an action $a_j \in A$ (A being the set of all possible actions) such that progress towards achieving its current goal is maximized. In practice, this means making an educated guess about which action will be most beneficial. In other words, the agent should select the action for which the best possible world is predicted at time t_{i+1}. Making the predictions is just a matter of inferencing on the MLN. The real question is how the agent can make the best MLN based on the information it currently has. To this end, OSL is employed in our system since it would not even be possible to make best MLN prediction online without online learning of new clauses.

IV. Unconscious Reasoning in CHARISMA

A. *Cartesian Genetic Programming*

Cartesian Genetic Programming (CGP) [21-22] represents programs in the form of directed acyclic graphs. Besides the original form, many CGP variants exist, such as Modular CGP which adds automatically defined functions, Self-Modifying CGP where the phenotype varies over time, Developmental CGP which adds bio-inspired neural/cellular developmental and graph rewriting mechanisms, and Cyclic CGP which allows cycles in the graph [23]. In the original version of CGP, the genotype is a fixed-length list of integers that specifies the primitive functions to use and the connections to make, in order to construct the program graph. The function genes, i.e. the integers identifying which functions to use, have as values indices for a user-defined table of primitive functions. The connection genes specify the inputs to their respective functions. At the end of the genotype, the output genes specify what each program output is connected to.

Constructing the phenotype is a recursive process starting from the outputs and connecting forward until no further connections are required. A consequence of this is that CGP genotypes often contain noncoding genes. Noncoding genes are important because they allow for silent mutations. The effect of a silent mutation is to preserve the current functionality (i.e. phenotype) while changing the set of daughter programs accessible through subsequent mutations. Silent mutations can accumulate until a later mutation reactivates the formerly noncoding genes, and thus a small mutation causes a major change in the phenotype. Analysis indicates that CGP is most efficient when a neutral search evolutionary strategy is used with large genotypes where most (e.g. 95%) of the genes are inactive [24]. In a neutral search, a parent is always replaced by its offspring if they have equal fitness.

B. *Novelty Search in Genetic Programming*

Premature convergence to local optima is a significant problem in evolutionary methods, including genetic programming [25]. It occurs when the diversity of the population dwindles before the search discovers a suitable solution, causing progress to stall. Searching based on novelty rather than an objective-based fitness metric is a radical approach to solving this problem that has yielded promising results in evolving artificial neural networks [26-27] and in genetic programming [28]. Furthermore, novelty search has been successfully combined with objective-based fitness metrics via a Pareto-based multi-objective evolutionary algorithm [29].

Almost any fitness-based evolutionary algorithm can be converted into a novelty search by simply replacing the fitness metric with a novelty metric. The intent of the novelty metric is to reward diverging from prior behaviors, thus it is basically a uniqueness score rather than a fitness score. Computationally, this can be measured using sparseness in the behavior space. Lehman and Stanley measure the sparseness ρ around individual x using the *k*-nearest neighbor algorithm:

$$\rho(x) = \frac{1}{k}\sum_{i=0}^{k} \mathrm{dist}(x, \mu_i) \tag{3}$$

where μ_i is the *i*th-nearest neighbor of x according to the behavior space distance metric, dist().

A key point is that it is novelty in the behavior space, not novelty in the genotype space, that is rewarded. Therefore dist( ) is domain-specific. The set of neighbors for comparison is composed of two distinct groups: the current population and an archive of previous novel individuals. Upon evaluation, an individual with $\rho(x) > \rho_{min}$ is automatically added to the archive. The minimum sparseness to count as "novel" ρ_{min} is a parameter that may be either fixed or dynamically adjusted over time.

C. *Adapting CGP for Semantic Hyper Networks*

A semantic hyper network (SHYNE) was proposed in [12] as a dynamic data structure for agents' knowledge representation, which is composed of nodes and links, with nodes being further divided into basic nodes and complex nodes. A basic node is similar to a node in a regular semantic network and it represents one simple or fundamental concept. A complex node is actually a sub-network composed of nodes and the links between them. Note that a complex node can act as a single node for the purposes of linking to other nodes, but its internal nodes also remain individually linkable. Another key feature of SHYNE is that both nodes and links have extensible attribute lists. Any node within the network, at least hypothetically, can be used to define a new attribute for nodes and/or links.

Each SHYNE component (i.e. node or link) has a globally unique identifier, which are the integers passed between functions in our CGP program graphs. Because the function table is domain specific, we will not provide specific function table here in this paper (specific function table will be provided in our future paper with specific applications). Instead of using modular CGP style modules, our approach is to add an unconscious process to the function table if its score exceeds a set threshold, essentially creating a new primitive function for evolving unconscious processes to potentially use.

V. Interaction between Conscious and Unconscious Reasoning

The unconscious processes also have two important roles to play. First, it improves the information available to online structure learning. Conscious focus has already done filtering based on motivational priorities. Unconscious processes can retrieve information from memory, and utilize lower-priority information from working memory to identify patterns, relationships, etc. and do other processing to provide online structure learning the best possible input based on the available information, current context, and the agent's past experience. Second, it proposes novel clauses that online structure learning might not discover on its own. Owing to their evolutionary rather than logical nature, Unconscious processes are better suited to discovering useful clauses "off the beaten path" of relational pathfinding.

VI. Conclusion and Future Work

Extensive interaction between conscious and unconscious reasoning mechanisms is a core aspect of the theory of mind in the intelligent CHARISMA agents. Online structure learning allows us to utilize Markov logic networks in ways which has never been possible before, because it can learn both the structure and parameters online. This makes it feasible to use Markov logic networks for conscious reasoning in a developmental learning cognitive agent. It also allows us to integrate Markov logic networks based conscious reasoning with CGP-based unconscious reasoning. We will provide some proof-of-concept demonstration of this new cognitive reasoning system in our future work based on different applications.

Since this is an ongoing project, currently we are developing the simulation to test our proposed cognitive reasoning mechanisms in a variety of physics puzzle scenarios, as shown in Fig. 3. Fig. 3 depicts an example of such a scenario. As development of the CHARISMA cognitive architecture continues, more challenging and interesting test cases will be possible. Many of the subsystems in CHARISMA are quite complex in their own right. In the future, we will present implementation and testing information about other CHARISMA subsystems and components as we gradually progress towards a fully operating implementation of our CIVS system.

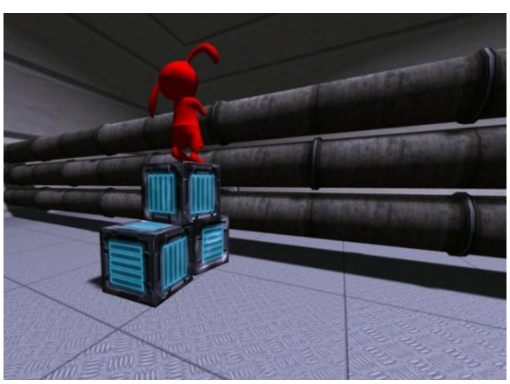


Figure 3. This physics puzzle scenario requires the agent to build a block staircase to pass the obstacle.

Acknowledgment

This project is partially supported by US DARPA. The authors would also like to thank Dr. T. N. Huynh of SRI International and Prof. R. J. Mooney of the Department of Computer Science, University of Texas at Austin for their assistance with online structure learning.

References

[1] T.W. Fong, I. Nourbakhsh, and K. Dautenhahn, "A survey of socially interactive robots: concepts, design, and applications," Tech. Rep. CMU-RI-TR-02-29, Robotics Institute, Carnegie Mellon University, 2002.

[2] A. Bauer, D. Wollherr, and M Buss, "Human-robot collaboration: a survey," Int. J. Humanoid Robotics, vol. 5, no. 1, pp. 47-66, 2008.

[3] M. Conforth and Y. Meng, "CHARISMA: a context hierarchy-based cognitive architecture for self-motivated social agents," 2011 International Joint Conference on Neural Networks (IJCNN), July 31-Aug. 5, 2011.

[4] B.J. Baars, In the Theater of Consciousness: The Workspace of the Mind, New York, NY: Oxford University Press, 1997.

[5] A. Newell, Unified Theories of Cognition, Cambridge, MA: Harvard University Press, 1990.

[6] P. Langley, J.E. Laird, and S. Rogers, "Cognitive architectures: research issues and challenges," Cognitive Systems Research, vol. 10, pp. 141-160, June 2009.

[7] J.R. Anderson, D. Bothell, M.D. Byrne, S. Douglass, C. Lebiere, and Y. Qin, "An integrated theory of the mind," Psychological Review, vol. 111, no. 4, pp. 1036–1060, 2004.

[8] R. Sun, "The CLARION cognitive architecture: extending cognitive modeling to social simulation," in Cognition and Multi-Agent Interaction, R. Sun, Ed. New York, NY: Cambridge University Press, 2006.

[9] P. Langley, "An adaptive architecture for physical agents," in Proc. 2005 IEEE/WIC/ACM Int. Conf. Intelligent Agent Technology, Compiegne, France, 2005, pp. 18-25.

[10] B.J. Baars and S. Franklin, "Consciousness is computational: the LIDA model of global workspace theory," Int. J. Machine Consciousness. vol. 1, no. 1, pp. 23-32, 2009.

[11] J.E. Laird, A. Newell, and P.S. Rosenbloom, "Soar: an architecture for general intelligence," Artificial Intelligence, vol. 33, pp. 1-64, 1987.

[12] M. Conforth and Y. Meng, "Self-reorganizing knowledge representation for autonomous learning in social agents," International Joint Conference on Neural Netowrks (IJCNN) 2011, July 31-Aug. 5, 2011.

[13] L. Getoor and B. Taskar (eds.), Introduction to Statistical Relational Learning, Cambridge, MA: MIT Press, 2007.

[14] M. Richardson and P. Domingos, "Markov logic networks," Machine Learning, vol. 62, pp. 107-136, 2006.

[15] T.N. Huynh and R.J. Mooney, "Online structure learning for Markov logic networks," in Proc. European Conf. on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, vol. 2, pp. 81-96, 2011.

[16] S. Kok and P. Domingos, "Learning the structure of Markov logic networks," in Proc. 22nd Int. Conf. on Machine Learning (ICML2005), 2005.

[17] L. Mihalkova and R.J. Mooney, "Bottom-up learning of Markov logic network structure," in Proc. 24th Int. Conf. on Machine Learning (ICML2007), 2007.

[18] M. Biba, S. Ferilli, and F. Esposito, "Discriminative structure learning of Markov logic networks," in Proc. 18th Int. Conf. on Inductive Logic Programming (ILP2008), pp. 59-76, 2008.

[19] S. Kok and P. Domingos, "Learning Markov logic network structure via hypergraph lifting," in Proc. 26th Int. Conf. on Machine Learning (ICML2009), pp. 505-512, 2009.

[20] S. Kok and P. Domingos, "Learning Markov logic networks using structural motifs," in Proc. 27th Int. Conf. on Machine Learning (ICML2010), pp. 551-558, 2010.

[21] J.F. Miller, "An empirical study of the efficiency of learning boolean functions using a Cartesian genetic programming approach," in Proc. Genetic and Evolutionary Computation Conf. (GECCO '99), pp. 1135–1142, Morgan Kaufmann, 1999.

[22] J.F. Miller and P. Thomson, "Cartesian genetic programming," in Proc. European Conf. on Genetic Programming, LNCS, vol. 1802, pp. 121–132, Springer, 2000.

[23] J.F. Miller and S.L. Harding, "GECCO 2010 Tutorial: Cartesian Genetic Programming" GECCO 2010, July 7–11, 2010, Portland, Oregon, USA.

[24] J.F. Miller and S.L. Smith, "Redundancy and computational efficiency in Cartesian genetic programming," IEEE Transactions on Evolutionary Computation, vol. 10, no. 2, pp. 167-174, April 2006.

[25] C. Gathercole and P. Ross, "An adverse interaction between crossover and restricted tree depth in genetic programming," in Proc. First Annual Conf. on Genetic Programming (GECCO '96), pp. 291-296, Cambridge, MA: MIT Press, 1996.

[26] J. Lehman and K.O. Stanley, "Exploiting open-endedness to solve problems through the search for novelty," in Proc. Eleventh Int. Conf. on Artificial Life (ALIFE XI), Cambridge, MA: MIT Press, 2008.

[27] S. Risi, S.D. Vanderbleek, C.E. Hughes, and K.O. Stanley, "How novelty search escapes the deceptive trap of learning to learn," in Proc. Genetic and Evolutionary Computation Conf. (GECCO 2009), New York, NY: ACM, 2009.

[28] J. Lehman and K.O. Stanley, "Efficiently evolving programs through the search for novelty," in Proc. Genetic and Evolutionary Computation Conf. (GECCO 2010), New York, NY: ACM, 2010.

[29] J.B. Mouret, "Novelty-based multiobjectivization," in Exploring New Horizons in the Evolutionary Design of Robots (IROS Workshop), 2009.