CEUR-WS.org/Vol-3432/paper21.pdf

Challenge Problems in Developing a Neuro-Symbolic OODA Loop

Alberto Speranzon[1], Christian H. Debrunner[2], David Rosenbluth[3], Mauricio Castillo-Effen[1], Anthony R. Nowicki[2], Kevin Alcedo[3] and Andrzej Banaszuk[1]

[1] *Lockheed Martin, Advanced Technology Labs, USA*
[2] *Lockheed Martin, Missiles and Fire Control, USA*
[3] *Lockheed Martin Artificial Intelligence Center, USA*

Abstract

In this paper, we analyze the role of Neuro-Symbolic (NS) methods within the standard Observe, Orient, Decide and Act (OODA) framework and discuss their associated opportunities and challenges. To ground the discussion, we consider the OODA loop applied to a wildland firefighting use case where mixed teams of autonomous agents and humans need to collaboratively work to contain and ultimately extinguish large-scale fires. NS methods are appealing as they enable the integration of symbolic knowledge coming, for example, from time-tested firefighting tactics and procedures encoded in training manuals. However, there are key challenges in capturing and integrating such information within NS pipelines, especially in a way that is scalable with respect to the number of procedures, dynamics of the environment, and regulations. The integration of such symbolic priors within a NS OODA loop will enable autonomous systems to generate highly expressive yet compressed representations (or abstractions) of the environment and other agents' state, actions, and intent. However, NS integration brings new challenges, such as, designing abstractions that are composable and transferable across complex tasks, deciding how to ground symbols to sensory data, and verifying their validity to ensure they lead to safe decisions. Symbolic abstractions also need to deal with the fact that sensory data is uncertain—like the behavior of the fire front (boundary, rate of spread, etc.). Furthermore, prior knowledge may be incomplete and inconsistent leading to representations that need to be capable of explicitly modelling and managing ambiguity. Addressing such challenge problems will not only advance the state-of-the-art in NS AI, but also concretely demonstrate the benefits of such methods within the autonomy domain.

Keywords

Autonomous systems, Abstractions, Decision Making, Challenge Problems.

1. Introduction

In recent years, deep neural networks (DNNs) and embedded GPUs have significantly improved the capabilities of an autonomous system across the full Observe, Orient, Decide and Act (OODA) loop [1, 2]. However, we are still far from achieving the level of robustness and reliability that we expect from such systems. One of the key difficulties is that autonomous systems are generally deployed in environments that are highly dynamic and uncertain where the actual state of the environment cannot be directly observed but needs to be inferred. Neuro-symbolic techniques offer the promise to extend the virtues of DNNs and, by incorporating symbolic techniques, to increase the autonomous systems ability to deal with uncertainty and ambiguity. In this paper, we explore the opportunities offered by NSR to enhance OODA loops, but also the challenges.

Consider, for example, the wildland firefighting problem shown in Figure 1. In this type of complex mission, there are many challenges in developing an understanding of the current and future state of the fire supporting the generation of strategies and actions to effectively fight a wildland fire.

NeSy 2023, 17th International Workshop on Neural-Symbolic Learning and Reasoning, Certosa di Pontignano, Siena, Italy.
EMAIL: For all authors the email is: name.initial.lastname@lmco.com
ORCID: 0000-0002-9203-2901 (Alberto Speranzon), 0000-0001-7349-6834 (David Rosenbluth)
© 2023 Copyright for this paper by its authors.
Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).
CEUR Workshop Proceedings (CEUR-WS.org)

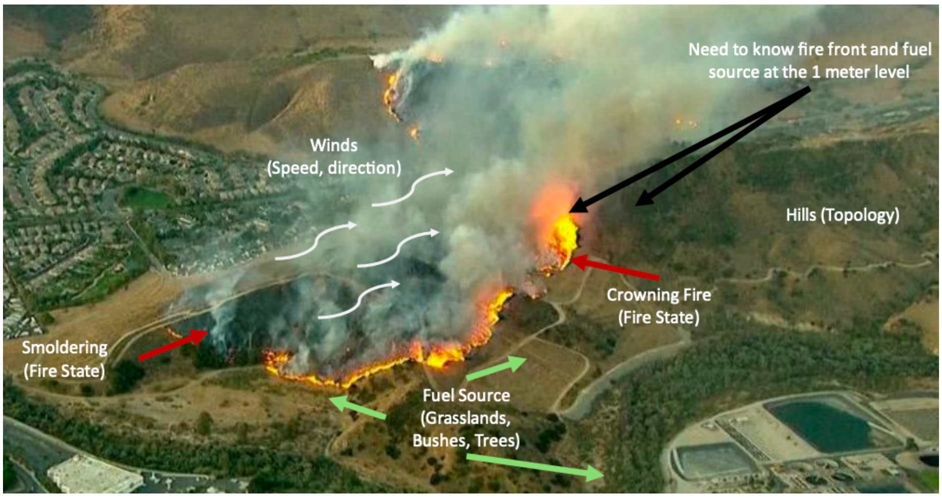
Need to know fire front and fuel source at the 1 meter level
Winds (Speed, direction)
Hills (Topology)
Crowning Fire (Fire State)
Smoldering (Fire State)
Fuel Source (Grasslands, Bushes, Trees)

Figure 1 – Example of a wildland fire. Note that fires can be in different states (e.g., smoldering, crowing fire), where wind and fuel sources affects its spreading and thick clouds of smoke reduce visibility both on the ground and in the air.

As the figure shows, fires may be in various states, burning at different rates along challenging terrains; winds and vegetation fuel the fires in a difficult to predict way and thick clouds of smoke make fire front detection and overall flight arduous. Furthermore, risks to people and infrastructure make decisions on how to tackle the fires difficult.

Purely data-driven machine learning (ML) approaches may not be sufficient to tackle the challenges of such missions. For example, we expect that deployed autonomous systems will: coordinate with humans and follow specific set of tactics and guidelines, build and reason over large-scale spatio-temporal models of the world, adapt online to new situations for which training data was not available before the mission, and manage both uncertainty and ambiguity.

Neuro-Symbolic (NS) methods provide a new framework to overcome some of the limitations of purely data-driven ML approaches [3, 4], however, missions like the wildland firefighting expose new challenges that need to be overcome to deploy safe and reliable NS OODA loops for autonomous systems. In the next section, we highlight some of these challenges to further energize the community towards the development of next generation of safe and intelligent autonomous systems.

2. Challenges in Designing NS OODA Loops

The challenges discussed in the following sections stem from the authors' recent experience in developing and integrating NS methods within autonomous systems. Not all of them are exclusive to NS-enabled autonomy.

2.1. Integration of Prior Knowledge

NS methods enable the integration of symbolic priors during both the learning and inference. In wildland firefighting, for example, symbolic priors can be a set of well-established tactics and procedures when fighting fires (e.g., never attack an aggressive fire at the head or up-slope from an active fire front) as well as priorities on how tactics should be selected (e.g., preservation of life being at the highest priority). Symbolic knowledge also imposes requirements on the "Observe" and "Orient" parts of the loop as certain types of specific features in the terrain, vegetation or land usage need to be detected and characterized.

Within the Decide portion of the loop, symbolic priors impact decisions as the autonomous systems need to conform to procedures to both coordinate effectively with humans and for ethical and legal reasons. ML methods, such as DNNs, require a lot of data to learn to recognize such complex behavioral patterns and from there select the appropriate actions. NS methods such as [5, 6, 7, 8, 9], to name a few, on the other hand provide ways to explicitly encode symbolic knowledge and constraints reducing the amount of data required for training.

However, at this stage there does not seem to exist well-established quantitative metrics that help developers understand the tradeoff between data requirements, richness of symbolic priors and performance of a NS system. In the application domain this is ultimately translating into development costs.

For example, in the wildland firefighting setting, generating (or collecting) more data, and possibly labeling the data, can be costly. Transforming firefighting procedures into ontologies and knowledge graphs by ingesting, for example, procedure manuals that encode information in highly unstructured forms, such as pictures and/or sketches can also very onerous.

One of the key enablers of the success of supervised learning has been the availability of extremely large sets of labelled data, typically released as benchmark data. Many existing NS methods (for example, [28, 6]) are tested using large and well established deep learning datasets, but they are combined with small- to medium-sized, hand-build knowledge bases that do not reflect the complexity of typical applications such as the wildland fire fighting. This raises the question if the NS community should consider the development of curated symbolic knowledge to be used to evaluate and compare NS methods. This would enable researchers to, firstly, show the benefits of the methods and, secondly, provide data to develop the metrics we mentioned earlier. Although ontologies and other sources of symbolic knowledge are available, more efforts seem to be needed to “replicate” the advances in data curation and distribution that have been, in many ways, behind the success demonstrated by statistical ML.

The development of advanced Large Language Models (LLMs) [10, 11] may provide an opportunity to speed up the creation of curated symbolic knowledge for robotics applications, as shown in some recent work [12, 13]. The integration of LLMs within the NS framework is certainly a topic or future research that we believe will have strong impacts within the autonomy/robotics communities.

In the context of symbolic knowledge capture, there is a more fundamental question that we believe should be pursued more aggressively. Generally, formal knowledge is encoded via description logics, and this raises the question if this formalism is sufficient or even optimal for NS methods. Recent developments [5, 7, 8, 14], and references therein, extend or propose new logics that leverage many-valued logics [15]. Although there are extensions of ontologies to many-values/fuzzy description logics [16, 17], most of the available ontologies do not use such semantics.

2.2. Development of Abstractions

Symbols represent concepts ranging from objects and classes to tasks and plans or from beliefs to properties and strategies. To apply symbolic knowledge, a NS system must be able to ground a symbol, i.e., associate it (possibly probabilistically) with real world observations or with combinations of other related symbols. Often symbols are arranged hierarchically (e.g., in an ontology) to represent concepts at many levels of abstraction, so we refer to the association of symbols with real world observations or with combinations of other related symbols as an abstraction. Hierarchical abstractions occur naturally in the OODA loop, where “Observe” creates symbols representing objects in the environment, “Orient” groups these observed objects and relates them to each other and to higher level concepts through a rich set of relationships, “Decide” plans actions based on the hierarchical plans and goals of the agents and symbolic representations, and “Act” carries out the actions in context of the observed objects and agents and the expected plans and goals of the agents. Prior knowledge generally defines abstractions, but common sense and efficient reasoning requires an ability to fluidly relate existing abstractions to each other, to refine them based on context, and to create new abstractions for new situations. Abstractions should have, at a minimum, the following characteristics:

1) Utility: the abstraction should be compositional, modular, and reusable/transferable (ideally across applications and domains). Reusability may require adaptation of the abstraction to new domains/contexts.
2) Interpretability: the abstractions should be defined in terms of existing abstractions/symbols and observations in a way that humans can understand and such that it is easy to determine whether a given abstraction is applicable in a given context.

In the firefighting domain, abstractions within the “Orient” part of the OODA loop will be designed to support the efficient prediction of wildfire behavior from the fire’s current state, the fuel characteristics, the topography, and the weather, and to characterize the risk to structures and human life in planning the firefighting effort.

Within “Decide”, the abstractions will likely need to align with the hierarchical organization of the command structure. At the top is the Incident Commander (IC) who makes both tactical and strategic decisions about how to best fight the fire that has both a long-term and short-term effect. At the lower layers, the abstractions will capture tactics and actions other agents, both autonomous and humans, are going to take to carry out the IC decisions.

Although human experts, field manuals and fire behavior models [18] can guide the development of abstractions, as noted in Section 2.1, it can be difficult to translate these into the appropriate symbolic knowledge base that an autonomous system can both learn from and reason on.

A key challenge to take advantage of NS methods in developing next generation OODA loops is on developing theoretical and computational methods for generating and adapting abstractions across the elements of the loop as well as measuring their utility and interpretability. Each of these characteristics is critical to ensure consistency of the symbolic representations across abstractions, the management of uncertainty and ambiguities (see Section 2.3) and to support reasoning. We believe that such abstractions should be designed to ensure transferability across problem instances/types and scalable to large number of heterogenous systems.

2.3. Uncertainty and Ambiguity

Although integration of probabilistic models, to consider, e.g., sources of uncertainty, have been considered in the context of NS methods [19, 20], there are other uncertainty and ambiguity considerations that are unique to NS architectures we would like to highlight. The symbolic abstraction process transforms vector space neural representations to symbolic ones. Since this is a mapping from continuous space to a discrete space, it is necessarily lossy. It is useful to think of this process as a vector quantization which will necessarily introduce quantization error/uncertainty and quantization noise [21]. Uncertainty can be quantitatively characterized in terms of a distortion function which quantifies the cost of replacing a specific element of a set with a symbol. Vector quantization algorithms optimize a rate-distortion to maximize the coding rate/efficiently while minimizing the distortion.

Although, we believe it is appealing to consider information theoretic approaches to study quantization tradeoffs, in the context of autonomous systems using NS components, the choice of symbols is not just a “bottom-up” process, from continuous to discrete domains, but also a “top-down” process as the symbols have associated semantics that depend on the mission, external knowledge encoded in ontologies and, potentially, the need to be interpretable by humans. In the firefighting setting, if we require autonomous systems to coordinate with humans, it seems highly desirable to have symbolic representations that match humans’ representations. Although there is initial work in developing interpretable symbolic NS world models for autonomous systems [22, 23], more research seems to be necessary to better understand the tradeoffs between human interpretability, efficiency, and distortion of symbolic representations.

A different, but related, important aspect in the design of a NS OODA loop is the need to ensure that symbol groundings do not conflict across the components of an OODA loop. The use of different complex sensors (e.g., EO/IR cameras, LIDARs, etc.) within the “Observe” and the generation of multiple abstractions in “Orient” to tradeoff computation complexity and accuracy, combined with the fact that the world is only partially observable, can lead to situations where symbolic representations are inconsistent. We believe that NS OODA loops will require new approaches to represent and reason under such uncertainty and ambiguity without sacrificing reliability and assurability, see also Section 2.4.

2.4. Assurance of NS Systems

The deployment of OODA loop solutions to safety critical domains, such as wildland firefighting require demonstrably correct, reliable, and safe behavior in all foreseeable situations. In regulated industries users need explainability, transparency, and auditability of the system’s behavior [24]. We call this the assurance challenge. Assured systems aim to satisfy four Overarching Properties (OAs) listed below [25]:

- Intent: The system's *defined intended behavior* must be correct and complete with respect to the *desired behavior*.

- Correctness: The *implementation* must be correct with respect to its *defined intended behavior* under *foreseeable operating conditions*.
- Innocuity: Any part of the *implementation* that is not required by the *defined intended behavior* must not have *unacceptable impact*.
- Operation: The system must possess mechanisms for addressing *correctness* or *intent* deficiencies and for mitigating *unacceptable impacts* manifested during operation.

The verification of a system is meant to establish whether OODA components satisfy the four OAs via objective evidence, thus providing confidence that a system has been "built right". The symbolic interfaces and knowledge representations afforded by NS techniques, compared to DNN based approaches, enable that possibility. For instance, Logic Tensor Networks (LTN) [5], are suitable for implementing multiple forms of NS classification, as required for a variety of "Observe", and "Orient" functions, as well as reasoning and query-answering, required by "Decision" functions. As LTNs implement a first-order language, Real Logic, the associated semantics enable the formal specification (Intent OA) and verification (Correctness OA) of NS OODA functions expressed in the Real Logic language. Furthermore, LTN-endowed NS components could also use its symbolic knowledge representation to monitor whether the actual operating conditions differ from those assumed during design and development. This enables the autonomous system to identify possible unsafe states and trigger fail-safe behaviors (Operation OA). LTNs represent just one category of verifiable NS structures. Logical Neural Networks [8], Neural Tensor Networks [26], Tensor Product Representations [27], and Abductive Learning [29] offer similar opportunities for verification, both offline and online.

Although the assurance of NS OODA loops can take advantage of explicit symbolic representations these may be still learned from combination of data and explicit knowledge.

Learned NS functions implies the presence of uncertainty and ambiguity, as mentioned before, that not only needs to be managed but also treated as an assurance gap. Seeding learning by injecting symbolic priors may accelerate learning but also introduces grounding boundary discrepancies, in other words, a discordance between real world objects and their representations. Furthermore, the validity of learned symbols and their symbolic relations are highly dependent on the quality, quantity, and relevance of the input and training data, and thus connected to the challenges in Section 2.1 and 2.2. More generally, we believe that the connections and interfaces between the symbolic and the sub-symbolic components of NS systems need to be better understood from an assurance perspective and, likely, new mitigation approaches need to be developed to provide strong safety guarantees.

3. Conclusions

NS methods can greatly improve the way we design and deploy autonomous systems in complex missions. The opportunity to incorporate prior symbolic knowledge, integrate deep learning with symbolic representations and reasoning, provide interpretability and enable formal analysis are huge advantages compared to approaches that completely rely on statistical machine learning. There are, however, challenges that we need to overcome to design next generation of NS based autonomous systems. In this paper we have listed a few challenges related to the tradeoff between data requirements and symbolic knowledge in training NS architectures, the development of composable symbolic abstraction for reasoning and the need of formalize methods for managing uncertainty and ambiguity in these novel architectures. We believe that autonomous systems/robotics can be a great application domain to drive the development of new theoretical and computational advances in NS learning and reasoning. Specifically, we have also introduced a rich problem domain, wildland firefighting, that provides the research community with a strong and concrete motivating example where we believe NS methods can make an important contribution and impact.

4. References

[1] J.R. Boyd, "Destruction and Creation", U.S. Army Command and General Staff College, 1976.

[2] J.R. Boyd, "A Discourse of Winning and Losing", Maxwell Air Force Base, Air University Press. 2018. Online: https://www.jstor.org/stable/resrep19552.

[3] M.K. Saker, L. Zhou, A. Eberhart, and P. Hitzler, "Neuro-Symbolic Artificial Intelligence." AI Communications, vol. 34, no. 3, 2021.

[4] A. A. Garcez and L.C. Lamb, "Neuro-Symbolic AI: The 3rd Wave." Artificial Intelligence Review, 2023.

[5] S. Badreddine, A.A. Garcez, L. Serafini, and M. Spranger, "Logic Tensor Networks". Artificial Intelligence, vol. 303, 2022.

[6] R. Manhaeve, S. Dumancic, A. Kimmig, T. Demeester, and L.D. Raedt, "DeepProbLog: Neural Probabilistic Logic Programming." In Advances in Neural Information Processing Systems (NeurIPS), 2018.

[7] C. Pryor, C. Dickens, E. Augustine, A. Albalak, W.Y. Wang, and L. Getoor, "NeuPSL: Neural Probabilistic Soft Logic". ArXiv, https://arxiv.org/abs/2205.14268, 2022.

[8] R. Riegel, A.Gray, F. Luus, N. Khan, N. Makondo, I.Y. Akhalwaya, H. Qian, R. Fagin, F. Barahona, U. Sharma, S. Ikbal, H. Karanam, S. Neelam, A. Likhyani, and S. Srivastava, "Logical Neural Network". ArXiv, https://arxiv.org/abs/2006.13155, 2020.

[9] J. Mao, C. Gan, P. Kohli, J. B. Tenenbaum, and J. Wu, "The Neuro-Symbolic Concept Learner: Interpreting Scenes, Words, and Sentences from Natural Supervision," ICLR, and ArXiv, https://arxiv.org/abs/1904.12584, 2019.

[10] T. Brown, B. Mann, N. Ryder, M. Subbiah, J.D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language Models Are Few-Shot Learners." In Advances in Neural Information Processing Systems (NeurIPS), 2020.

[11] OpenAI, "Introducing ChatGPT". Online https://openai.com/blog/chatgpt, Accessed 2023.

[12] W. Chen, S. Hu, R. Talak, and L. Carlone, "Leveraging Large Language Models for Robot 3D Scene Understanding." ArXiv, https://arxiv.org/abs/2209.05629, 2022.

[13] D. Shah, B. Osiński, and S. Levine, "LM-Nav: Robotic Navigation with Large Pre-Trained Models of Language, Vision, and Action." In Conference on Robot Learning, PMLR, 2023.

[14] R. Marinescu, H. Qian, A. Gray, D. Bhattacharjya, F. Barahona, T. Gao, R. Riegel, and P. Sahu, "Logical Credal Networks." In Advances in Neural Information Processing Systems (NeurIPS), 2022.

[15] G. Malinowski, "Many-Valued Logics". Oxford University Press, 1993.

[16] F. Bobillo, and U. Straccia, "Reasoning with the Finitely Many-Valued Łukasiewicz Fuzzy Description Logic SROIQ." Information Sciences, vol. 181, no. 4, 2011.

[17] M. Baziz, M. Boughanem, Y. Loiseau, and H. Prade, "Fuzzy Logic and Ontology-Based Information Retrieval." Fuzzy Logic: A Spectrum of Theoretical & Practical Issue, 2007.

[18] J.H. Scott, "Introduction to Fire Behavior Modeling". National Interagency Fuels, Fire, & Vegetation Technology Transfer, 2012.

[19] L. De Raedt, R. Manhaeve, S. Dumančić, T. Demeester and A. Kimming, "Neuro-Symbolic = Neural + Logical + Probabilistic", Neuro-Symbolic Artificial Intelligence: The State of the Art, P. Hitzler and M. K. Sarker, Eds. IOS Press, 2022.

[20] H. Poon, H. Wang, and H. Lang, "Combining Probabilistic Logic and Deep Learning for Self-Supervised Learning", Neuro-Symbolic Artificial Intelligence: The State of the Art, P. Hitzler and M. K. Sarker, Eds. IOS Press, 2022.

[21] R. Gray, "Vector quantization.". IEEE ASSP Magazine, vol. 1, no. 2, 1984.

[22] I. Armeni, Z.-Y. He, J. Gwak, A. R. Zamir, M. Fischer, J. Malik, and S. Savarese, "3D scene graph: A Structure for Unified Semantics, 3D Space, and Camera". International Conference on Computer Vision (ICCV), 2019.

[23] A. Rosinol, A. Gupta, M. Abate, J. Shi, and L. Carlone, “3D Dynamic Scene Graphs: Actionable Spatial Perception with Places, Objects, and Humans”. In Robotics Science and Systems, 2020.

[24] Z. Iqbal, “Assurance of Machine Learning/TinyML in Safety-Critical Domains”. IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC), 2022.

[25] C. M. Holloway, “Understanding the Overarching Properties”. National Aeronautics and Space Administration, Langley Research Center, 2019. Online: http://shemesh.larc.nasa.gov/arg/utops.pdf.

[26] R. Socher, D. Chen, C. D. Manning, and A. Ng, “Reasoning with Neural Tensor Networks for Knowledge Base Completion”. In Advances in Neural Information Processing Systems (NeurIPS), 2013.

[27] P. Smolensky, M. Lee, X. He, W.-t. Yih, J. Gao, and L. Deng, “Basic Reasoning with Tensor Product Representations”. ArXiv, https://arxiv.org/abs/1601.02745, 2016.

[28] Q. Cao, W. Wan, X. Liang, and L. Lin, “Graph Reasoning Networks and Applications,” in Neuro-Symbolic Artificial Intelligence: The State of the Art, P. Hitzler and M. K. Sarker, Eds. IOS Press, 2022.

[29] Z.-H. Zhou and Y.-X. Huang, “Abductive Learning,” in Neuro-Symbolic Artificial Intelligence: The State of the Art, P. Hitzler and M. K. Sarker, Eds. IOS Press, 2022.