Bio-inspired computing systems: handle with care, discard if need it

Rogério de Lemos
r.delemos@kent.ac.uk
University of Kent
United Kingdom

ABSTRACT

Nature has an excellent track record in solving problems, and while biological inspired approaches draw inspiration from nature, they should not emulate it blindly. What works for nature may not work for computer systems - bio-inspired computing comes to the rescue. In this position paper, we look into the problem of bio-inspired computing from two perspectives, that of models and algorithms. In the context of self-adaptive software systems, the challenge is to come up with approaches that are able to generate specific solutions on demand and during operational-time.

CCS CONCEPTS

• **Software and its engineering → Software design tradeoffs**.

KEYWORDS

feedback control loop, models, reinforcement learning, Bio-inspired computing

ACM Reference Format:
Rogério de Lemos. 2024. Bio-inspired computing systems: handle with care, discard if need it. In *19th International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS '24), April 15–16, 2024, Lisbon, Portugal*. ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3643915.3644096

1 INTRODUCTION

In order to strive, living organisms adapt and evolve. The whole idea is quite appealing if the foresight is to perceive computer systems as living entities, hence "self-adaptation". However, context may shatter any hopes. What works for nature may not work for computer systems - bio-inspired computing comes to the rescue. Hard lessons have been learned by trying to mimic nature because it is not that simple even replicating its most simple laws. Faced with failure, creativity rebounds, and today there are several techniques that surpass nature's handling of complexity. However, this has come with a cost since it has become a challenge to explain their process for decision making. It may be acceptable for some kind of self-adaptive software systems, but not for those systems in which assurances are essential. The argument made in this position paper is that it is time to break away with the chain attaching us to bio-inspired computing just for the sake of justifying novelty. Solutions needed for supporting self-adaptation go beyond what nature is able to provide. This argument is approached from two different perspectives that of models and algorithms.

SEAMS '24, April 15–16, 2024, Lisbon, Portugal

ACM ISBN 979-8-4007-0585-4/24/04
https://doi.org/10.1145/3643915.3644096

2 DIFFERENT KIND OF MODELS

Since Dowling et al. [7] applied Collaborative Reinforcement Learning (CRL) for establishing autonomic distributed system properties, Reinforcement Learning (RL) has moved on. Today's RL policies generated either by PPO [13] or DQN [11], for example, are quite complex and far from being explainable.

In the context of cyber security, the goal is to have RL agents that aim to perform fully autonomous network defence [15], which is quite far from handcraft UML models and adaptation strategies targeted for the self-protection of systems [1]. The advantage of using RL feedback control loops, instead of feedback control loops, like MAPE-K [12], for this particular problem domain, is that the focus can go towards identifying sophisticated threat scenarios, and thus improving system protection. Again the trend is to move away from process descriptions towards data descriptions [5], as a consequence the way self-adaptive software systems are developed and how assurances are provided need to be changed.

For a start, there is a need for more data, and accurate data, however for a lot of applications there is no data available, hence data needs to be generated. The generation of synthetic datasets has become an acceptable solution since it is much cheaper to obtain, and particular scenarios can be better captured. Simulation come to rescue. A lot of deep learning models are obtained using simulated data. The validation of this data becomes challenging, but what it matters is the validation of models obtained from the data before they are deployed on real life applications. When developing future self-adaptive software systems, the focus should be data because models are very specific to the data from which they were trained. The same system in a different context may require different machine learning (ML) models because it depends on the features that capture its context. How generic models can be synthesised from specific data can also be a challenge. For example, in cyber security, the development of RL policies relies quite heavily on simulators[1] since there is a need for a lot of data. However, threats are rare events, and for the models to learn from rare events they need a lot of instances of these events. Another good example of the challenges to obtaining generic models is the fact that, for different attack patterns, there is the need for specific RL policies. Considering the landscape of threats, the protection of systems and networks requires the deployment of several RL policies, and how

[1] https://github.com/cage-challenge/cage-challenge-2 – accessed in December 2023.

the actions of these policies are coordinated is still an open research question.

There is another challenge associated with data, which is that of concept drift – the phenomenon where the statistical properties of the target variable or the input features in a machine learning problem change over time [8]. In the context of cyber security applications, attack models change with time, and the models tend to degrade over time, thus becoming less effective in detecting attacks. This is referred as adversarial drift, and it is one of the reasons that impair the deployment of ML models for detecting and mitigating attacks since the pattern of attacks may drift over time [4].

Another peculiarity regarding cyber security data, which tend to affect the quality of the models, is that most of its data is categorical, instead of numeric, which is the essence of image analysis, for example. Categorical data leads to brittle models, that is, small variations in the data can lead to miss-classifications.

All these are challenges that need to be overcame when developing data-centric models.

3 BIO-INSPIRED ALGORITHMS

Nature is inspirational, and from its different interpretations may emerge solutions that can be mapped into the artificial [3]. Whether is swarm intelligence [2] or artificial immune systems [6], these have been applied to several computer problems and with reasonable success. But these turn out to be specific solutions to specific problems.

There are two characteristics that are key when describing self-adaptive software systems: explicit feedback control loops, and models. In bio-inspired computing these may be not necessarily present. For example, fish shoals based algorithms may be based on simple rules.

The challenge here is how to incorporate quite specific solutions into more sophisticated feedback control loops that can work in collective and coordinated way.

4 CONCLUSIONS

In summary, looking from the bio-inspired perspective, the future landscape of applying feedback control loops for supporting self-adaptation will have to change in order to incorporate more data-centric, instead of process-centric solutions [5], which may lead to specific solutions. From the perspective of bio-inspired algorithms, like in nature where different organisms express behaviours that are specific to the species, the same may happen when applying bio-inspired algorithms to self-adaptive software systems. Firstly, there is the challenge of interpreting nature laws into a computer context for which there are no direct mappings, so we may end up with very specific solutions that are problem centric. Secondly, considering the whole range of computer applications, solutions may also require to be specific for achieving optimal outcomes. From the perspective of models, instead of the generic solutions, like Rainbow that relies on architectural models [9], application specific machine learning models may take a more prominent role. This can either be as components on a wider feedback control loop, like MAPE-K, or replacing the whole feedback control loop, as it is the case for Reinforcement Learning.

The incorporation of bio-inspired algorithms as part of the self-adaptation raises several challenges. Uncertainty is one of them since data descriptions are not precise, which may lead to further uncertainty [14]. Another challenge is how to incorporate the usual cycle for synthesising machine learning models, i.e., training, testing and evaluation, into operational-time since it would not make sense to have these at development-time, for example, when considering concept drift. From the machine learning viewpoint, model synthesis during production should not be a problem, but the challenge would be how to integrate these into feedback control loop for enabling self-adaptation. For that, some inspiration from AutoML should be necessary [10], which incorporates several concepts from self-adaptation.

As a conclusion that can be drawn from this paper is that, instead of generic bio-inspired solutions that can be easily tailored to handle a wide range of problems, the challenge is to come up with approaches that are able to generate specific solutions on demand and during operational-time.

REFERENCES

[1] C. Bailey, L. Montrieux, R. de Lemos, Y. Yu, and M. Wermelinger. 2014. Run-Time Generation, Transformation, and Verification of Access Control Models for Self-Protection. In *Proceedings of the 9th International Symposium on Software Engineering for Adaptive and Self-Managing Systems* (Hyderabad, India) *(SEAMS 2014)*. Association for Computing Machinery, New York, NY, USA, 135–144.

[2] R. Ball, J. Grant, J. So, V. Spurrett, and R. de Lemos. 2007. Dependable and Secure Distributed Storage System for Ad Hoc Networks. In *Ad-Hoc, Mobile, and Wireless Networks, 6th International Conference, ADHOC-NOW 2007, Morelia, Mexico, September 24-26, 2007, Proceeedings (Lecture Notes in Computer Science, Vol. 4686)*, Evangelos Kranakis and Jaroslav Opatrny (Eds.). Springer, 142–152.

[3] P. J. Bentley. 2007. *Digital Biology: How Nature Is Transforming Our Technology and Our Lives.* Simon & Schuster.

[4] L. Cavallaro, J. Kinder, F. Pendlebury, and F. Pierazzi. 2023. Are Machine Learning Models for Malware Detection Ready for Prime Time? *IEEE Security & Privacy* 21, 2 (March 2023), 53–56.

[5] R. de Lemos. 2005. *The Conflict between Self-* Capabilities and Predictability.* Springer-Verlag, Berlin, Heidelberg, 219–228.

[6] R. de Lemos, J. Timmis, M. Ayara, and S. Forrest. 2007. Immune-Inspired Adaptable Error Detection for Automated Teller Machines. *IEEE Trans. Syst. Man Cybern. Part C* 37, 5 (2007), 873–886.

[7] J. Dowling, R. Cunningham, E. Curran, and V. Cahill. 2004. Collaborative reinforcement learning of autonomic behaviour. In *Proceedings. 15th International Workshop on Database and Expert Systems Applications, 2004.* 700–704.

[8] J. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy, and A. Bouchachia. 2014. A survey on concept drift adaptation. *Comput. Surveys* 46, 4 (March 2014), 44:1–44:37.

[9] D. Garlan, S.-W. Cheng, A.-C. Huang, B. R. Schmerl, and P. Steenkiste. 2004. Rainbow: Architecture-Based Self-Adaptation with Reusable Infrastructure. *IEEE Computer* 37, 10 (2004), 46–54.

[10] X. He, K. Zhao, and X. Chu. 2021. AutoML: A survey of the state-of-the-art. *Knowledge-Based Systems* 212 (Jan. 2021), 106622. https://doi.org/10.1016/j.knosys.2020.106622

[11] D. Horgan, J. Quan, D. Budden, G. Barth-Maron, M. Hessel, H. van Hasselt, and D. Silver. 2018. Distributed Prioritized Experience Replay. arXiv:1803.00933 [cs.LG]

[12] J. O. Kephart and D. M. Chess. 2003. The Vision of Autonomic Computing. *IEEE Computer* 36, 1 (2003), 41–50.

[13] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. 2017. Proximal Policy Optimization Algorithms. arXiv:1707.06347 [cs.LG]

[14] D. Weyns, R. Calinescu, R. Mirandola, K. Tei, M. Acosta, A. Bennaceur, N. Boltz, T. Bures, J. Camara, A. Diaconescu, G. Engels, S. Gerasimou, I. Gerostathopoulos, S. Getir Yaman, V. Grassi, S. Hahner, E. Letier, M. Litoiu, L. Marsso, A. Musil, J. Musil, G. Nunes Rodrigues, D. Perez-Palacin, F. Quin, P. Scandurra, A. Vallecillo, and A. Zisman. 2023. Towards a Research Agenda for Understanding and ManagingUncertainty in Self-Adaptive Systems. *SIGSOFT Softw. Eng. Notes* 48, 4 (oct 2023), 20–36.

[15] M. Wolk, A. Applebaum, C. Dennler, P. Dwyer, M. Moskowitz, H. Nguyen, N. Nichols, N. Park, P. Rachwalski, F. Rau, and A. Webster. 2022. Beyond CAGE: Investigating Generalization of Learned Autonomous Network Defense Policies. arXiv:2211.15557 [cs.LG]