Collaborative Systems: Learning and Working Together

Lukas Esterle

Aarhus University and DIGIT, Denmark

Abstract
Autonomous systems interacting and collaborating need to understand their individual and collective goals as well as their individual capabilities. This keynote outlined challenges and approaches to enable systems to collaborate. Additionally we explore approaches to enable systems to learn and exploit the diverse knowledge of the individual agents in a collaborative manner.

Keywords
online multi-object k-coverage, team formation, autonomous adaptation, federated learning

The rise of computing systems is unbroken, spreading from our wrist, pockets, to our cars, houses, and even cities [1]. Individual smart devices are being deployed and distributed around our environment with their very individual goals, resources, and capabilities [2, 3]. Systems therefore collaborate only on accident or if they have explicitly been designed to do so [4]. This leads to many systems operating with limited performance as they interfere with each other in the environment. The potential of collaboration with initially unknown systems, that might be able to support each other, is completely untapped. In previous work we relied on common implementations and knowledge when utilising nature-inspired approaches. In contrast, we aim to close the gap and enable systems to acknowledge each other and build upon their individual knowledge and capabilities in more recent and ongoing work.

1. Working together

Over the past few years we explored problems requiring multiple autonomous systems to collaborate in order to maximise the performance of the collective system. The problem in question is the *online multi-object k-coverage problem* [5]. Here, moving objects can appear and disappear at random in an environment. Multiple (k) mobile observers, governed by autonomous software agents, are tasked with keeping these objects within their field of view. However, neither the location or movement patterns of the objects nor the amount or position of the other observers are known to the each system. This gives rise to a trade-off between exploration of the environment in order to find and locate new targets, and following the known objects in order to maximise their coverage.

In our initial work, we utilised direct communication in order to attract additional systems to cover specific objects. While simply broadcasting information worked well, to reduce the

lukas.esterle@ece.au.dk (L. Esterle)

communication effort required systems to have knowledge about others in their respective environments in order to target their advertisement [5]. As an alternative to direct message exchanges, [6] utilised Reynolds' flocking mechanism and adjusted the cohesion and separation parameter dependent on the number of agents in the environment. For this, entropy was used as a mechanism to attract and repel other agents.

[7] explored approaches where the agents willingness to interact is utilised to make local decisions. This willingness is influenced by the motion of the target. Specifically, the willingness of an agent to cover another object is proportional to the similarity of the direction of movement. This way we can expect the agent to cover the object for an extended period of time. [8], inspired by the feeding techniques of humpback whales and hyenas, enabled individual agents to take different tasks depending on how others are currently operating. Here, agents will deliberately observe the environment when all objects are covered sufficiently. This leads to higher detection rates and better assignment of agents to individual objects while keeping over-provisioning low. [9] investigated from where coordination control should be coming and created dedicated observer and tracker agents. Observers are tasked to roam the environment to detect new objects while tracker are dedicated to follow those object and provision them accordingly. Tracker agents are either controlled by observers or can be operate autonomously. Finally, [10] developed an near-optimised approach to solve the online multi-object k-coverage problem utilising linear programming in an aggregate computing framework.

2. Learning together

In all these cases, agents needed to know about each other. This would require developers of different agents to agree on and adhere to common interfaces and standards. In order to overcome this issue, future autonomous systems are required to have an awareness of other agents. [11, 3, 12] described different levels of such an awareness. The awareness of others can vary from simple stimulus-, time-, interaction-, to more complex goal-awareness. However, for a simple interaction, we can argue that at least time-awareness is required in order to anticipate external stimuli. With more awareness and deeper understanding of the situation (e.g. interaction- and goal-awareness), autonomous systems can operate towards collaborative behaviour. We later expanded the concept towards competence-awareness [13] to give autonomous systems the awareness of competences of themselves and others. Only with knowledge of the potential capabilities of others, we can start exploring possible collaborative actions purposefully and beyond random action exploration.

To overcome collaborative challenges, we recently started to explore potentials of collaborative multi-agent learning. Here we explore various strands of research around deep learning and aim to bring those strands together [14]. Specifically, we explore federated learning together with early exits and split computing. Federated learning are techniques where the knowledge of individual learners can be combined in a larger model [15]. By sharing the trained models only, rather than the actual training input, privacy concerns can be mitigated. This is beneficial when the individual learners are unknown or change at runtime. Usually, the tasks for which the networks are trained are known to all agents. However individual agents could utilise available data to estimate the tasks for which a network has been trained by another agent, allowing one agents to draw conclusion about the inference capabilities of another agent. Overall, trustworthiness and correctness can be verified by the individual agents upon receiving the networks by utilising previous training data before merging received networks with local networks.

Early exits is a concept allowing deep neural networks to not execute all neural layers but stop execution at earlier neural layers either because satisfactory results are achieved or if later layers should be executed at another computing location (e.g. from transferring computation from the edge to the cloud) [16]. We specifically work towards approaches where we utilise early exits in combination with federated learning. While general knowledge is shared with other agents, specific knowledge is retained at the individual agent. By sharing the general knowledge, inference can be improved overall. Specialising the individual agents further increases performance under the assumption of observations remaining the same (or similar) for the different agents. In future work, we are interested in collaborative agents, able to request support from others in case their own inference result is unsatisfactory. Instead of transmitting all information or even raw input data, only intermediate results from the early exits are being shared among the agents, reducing the amount of information transmitted while preserving important privacy aspects.

References

[1] M. Lombardi, F. Pascale, D. Santaniello, Internet of things: A general overview between architectures, protocols and applications, Information 12 (2021) 87.

[2] L. Esterle, R. Grosu, Cyber-physical systems: challenge of the 21st century, e & i Elektrotechnik und Informationstechnik 133 (2016) 299–303.

[3] K. Bellman, C. Landauer, N. Dutt, L. Esterle, A. Herkersdorf, A. Jantsch, N. TaheriNejad, P. R. Lewis, M. Platzner, K. Tammemäe, Self-aware cyber-physical systems, ACM Trans. on Cyber-Physical Systems 4 (2020) 1–26.

[4] K. Bellman, J. Botev, A. Diaconescu, L. Esterle, C. Gruhl, C. Landauer, P. R. Lewis, P. R. Nelson, E. Pournaras, A. Stein, et al., Self-improving system integration: Mastering continuous change, Future Generation Computer Systems 117 (2021) 29–46.

[5] L. Esterle, P. R. Lewis, Distributed autonomy and trade-offs in online multiobject k-coverage, Computational Intelligence 36 (2020) 720–742. doi:10.1111/coin.12264.

[6] D. W. King, L. Esterle, G. L. Peterson, Entropy-based team self-organization with signal suppression, in: Conf. on Artificial Life, 2019, pp. 145–152.

[7] M. Frasheri, L. Esterle, A. V. Papadopoulos, Cooperative multi-agent systems for the multi-target κ-coverage problem, in: Agents and Artificial Intelligence, 2021, pp. 106–131.

[8] L. Esterle, Goal-aware team affiliation in collectives of autonomous robots, in: Int. Conf. on Self-Adaptive and Self-Organizing Systems, 2018, pp. 90–99. doi:10.1109/SASO.2018.00020.

[9] L. Esterle, D. W. King, Loosening Control—A Hybrid Approach to Controlling Heterogeneous Swarms, ACM Trans. on Autonomous and Adaptive Systems 16 (2022) 1–26.

[10] D. Pianini, F. Pettinari, R. Casadei, L. Esterle, A Collective Adaptive Approach to Decentralised k-Coverage in Multi-Robot Systems, ACM Trans. on Autonomous and Adaptive Systems (2022) 1–39. Accepted for publication.

[11] L. Esterle, J. N. A. Brown, I think therefore you are: Models for interaction in collectives of self-aware cyber-physical systems, ACM Trans. on Cyber-Physical Systems 4 (2020).
[12] A. Diaconescu, K. L. Bellman, L. Esterle, H. Giese, S. Götz, P. Lewis, A. Zisman, Architectures for collective self-aware computing systems, in: Self-Aware Computing Systems, 2017, pp. 191–235.
[13] L. Esterle, J. N. Brown, The competence awareness window: Knowing what i can and cannot do, in: Int. Conf. on Autonomic Computing and Self-Organizing Systems, 2020, pp. 62–63.
[14] L. Esterle, Deep learning in multiagent systems, in: A. Iosifidis, A. Tefas (Eds.), Deep Learning for Robot Perception and Cognition, 2022, pp. 435–460.
[15] J. Konečnỳ, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, D. Bacon, Federated learning: Strategies for improving communication efficiency, arXiv preprint arXiv:1610.05492 (2016).
[16] S. Teerapittayanon, B. McDanel, H.-T. Kung, Branchynet: Fast inference via early exiting from deep neural networks, in: Int. Conf. on Pattern Recognition, 2016, pp. 2464–2469.