Informatik Spektrum (2022) 45:295–299
https://doi.org/10.1007/s00287-022-01487-8

HAUPTBEITRAG

Reinforcement learning as a basis for cross domain fusion of heterogeneous data

Sören Christensen[1] · Sven Tomforde[2]

Accepted: 25 June 2022 / Published online: 4 October 2022

Abstract
We propose to establish a research direction based on Reinforcement Learning in the scope of Cross Domain Fusion. More precisely, we combine the algorithmic approach of evolutionary rule-based Reinforcement Learning with the efficiency and performance of Deep Reinforcement Learning, while simultaneously developing a sound mathematical foundation. A possible scenario is traffic control in urban regions.

Introduction

The basic idea of Cross Domain Fusion (CDF) is to go beyond a pure combination of different data sources, knowledge bases, models, and views towards leveraging the combination of resources to gain a more comprehensive understanding. This is assumed to result in more accurate, more stable, more interpretable, and more holistic models of conditions and processes.

CDF aims at cross-cutting integration and can thus include a wide variety of sources of knowledge and stages of the processing chain. This leads to several challenges in terms of the interaction of models and observations, the combination of heterogeneous data and the adaptive linking of steps in the processing chain. Accordingly, the main questions are what should be fused (raw data, pre-processed data, interpreted data, scientific models, derived patterns, background knowledge), how this should be done (quantitative, qualitative, supervised, unsupervised, autonomous), and where (interactive, automatic, at the sensor, in the cloud).

In this article, we propose to establish a research direction based on reinforcement learning (RL) in the scope of CDF to allow for an adaptive runtime fusion and, therefore, more appropriate modelling capabilities when interacting with processes. This means working on processed data, in combination with identified patterns and established scientific models, to perform the fusion automatically by considering rewards as feedback signals, and to do this efficiently, enabling an application directly at the sensor as well as in cloud environments.

We consider traffic control in urban regions as a possible scenario: Assume a control system combining local decisions at each intersection controller with city-based global control strategies (see [6] for a motivating example). Locally this has access to sensor information (e.g., induction loop and camera sensors), to preprocessed data from neighboured intersection controllers, and to established models (e.g., topology and simulation models reflecting the setup of the infrastructure). City-wide, this can be augmented with expected traffic patterns (i.e., known regular and seasonal demands) as well as unstructured data from the Internet (i.e., to predict unexpected events with severe impact on the observed and predicted traffic flow volumes).

In the remainder of this article, we initially summarise the background of our proposal by briefly revisiting the necessary basic foundation, develop a research agenda towards interpretable and self-explaining RL technology for CDF, and give an outlook on how this can be implemented.

✉ Sören Christensen
christensen@math.uni-kiel.de

Sven Tomforde
st@informatik.uni-kiel.de

[1] Department of Mathematics, Christian-Albrechts-Universität zu Kiel, Kiel, Germany

[2] Department of Computer Science, Christian-Albrechts-Universität zu Kiel, Kiel, Germany

Background

Reinforcement Learning (RL) models the agent's environment as a Markov Decision Process (MDP), which is given as a fivetuple <S, A, P, R, γ>. In the standard case, S is a finite set of discrete states in the environment, A contains the finite set of the agent's actions, P is the state transition probability matrix, R quantifies the reward function and $\gamma \in [0, 1]$ is a discount factor given priority to future or immediate rewards. Such an RL agent continuously interacts with the environment and learns an optimal policy based on trial and error. At each time step t, the agent observes the state $s_t \in S$ and responds with an action at. Subsequently, the environmental state is changed to s_{t+1} according to the transition probability matrix. Further, the agent receives an immediate reward rt according to the underlying (unknown) reward function R. This reward is used to update the probability to apply at in st again with the goal to maximise the cumulative discounted reward. Due to the curse of dimensionality, approximation techniques have to be used, see [1] for an overview.

Deep RL methods use deep learning to approximate any of the following components of RL: the value function, the policy, or the model (i.e., state transition function and reward function). Especially, in non-deterministic or large environments (such as real-world robotic scenarios [2] or complex video games [8]), the value of deep neural network pays off since raw sensor input can directly be forwarded to the network. Although the agents are limited to human action constraints, e.g., a maximum number of actions per minute, they reached super-human performance in the games of the Atari collection [4] or Starcraft II [7]. For the latter, so-called long short-term networks [3] play an important role in finding the perfect memory horizon.

Evolutionary rule-based RL models the learning problem using a population of classifiers that are evolved online. Initially established by Holland, the field is currently characterised by the Extended Classifier System as invented by Wilson in 1995 [9]. Fig. 1 depicts an example of an XCS for continuous values as an input signal (i.e., s_t). The current knowledge is stored in the "population", where each classifier contains a *condition* part (i.e., a niche of the input space encoded as hyper-rectangle), an action (here encoded as discrete options A_1 to A_n), a predicted payoff p, a prediction error ϵ and a fitness f. Such a classifier encodes the statement "If you perform action A_i in this condition you will receive the payoff p". The error ϵ describes the reliability of this prediction, while the fitness transforms it into a strength value. Classifiers in XCS contain further attributes such as numerosity or experiences that are not relevant for the basic cycle. The standard algorithm works as follows: In each activation cycle, all classifiers of the population are checked against the input signal and the matching ones are copied to the "match set". The "prediction array" determines the fitness-weighted p-values for all contained actions in the match set. Based on a roulette-wheel scheme, one action is selected and all the supporting classifiers copied to the "action set". This action is applied to the system and the payoff is observed (at time t+1). This observed payoff is used to update the classifiers of the previous action set using the modified delta-rule (Widrow-Hoff delta rule) in combination with the moyenne adaptive modifee technique.

Another component of our approach is the theory of optimal stopping and related areas, such as optimal switching or optimal impulse control. This provides the framework to enable optimal timing for certain actions in a random environment. Mathematically, this is represented by the optimisation problem of maximising the expected payoff of the stopped process X_τ over stopping times τ. Originally, such problems were studied in sequential statistics ("When to stop collecting data?") and financial mathematics ("When to execute options?"), but the areas of application are much broader nowadays [5]. Surprisingly, however, its use as a component of an RL process has rarely been carried out to date.

One reason may be the theoretical hurdle for an adequate formulation of the problems.

Research agenda

We assume that one of the major goals of CDF is a sophisticated, automated understanding of real-world processes and behaviour. This should, for instance, serve as a basis for interaction with humans and provide means for decision support. For the scope of applied RL technology, this directly poses two strict goals:

1. The current knowledge should always be interpretable by humans in the sense that the representation allows for an intuitive understanding
2. The learning system should be able to automatically detect causalities and turn them into self-explanations comprehensible to the user

Besides these user-centred goals, the RL system should, of course, be as efficient, adaptive and appropriate as possible, fusioning information sources from various possible sources. To allow for such an approach, we propose to combine the algorithmic approach of evolutionary rule-based RL with the efficiency and performance of deep reinforcement learning (DRL), while simultaneously developing a sound mathematical foundation. This results in the formulation of the following research agenda.

Challenge A: Basic algorithmic concept Based on Wilson's XCS system for real-valued environments, the underlying

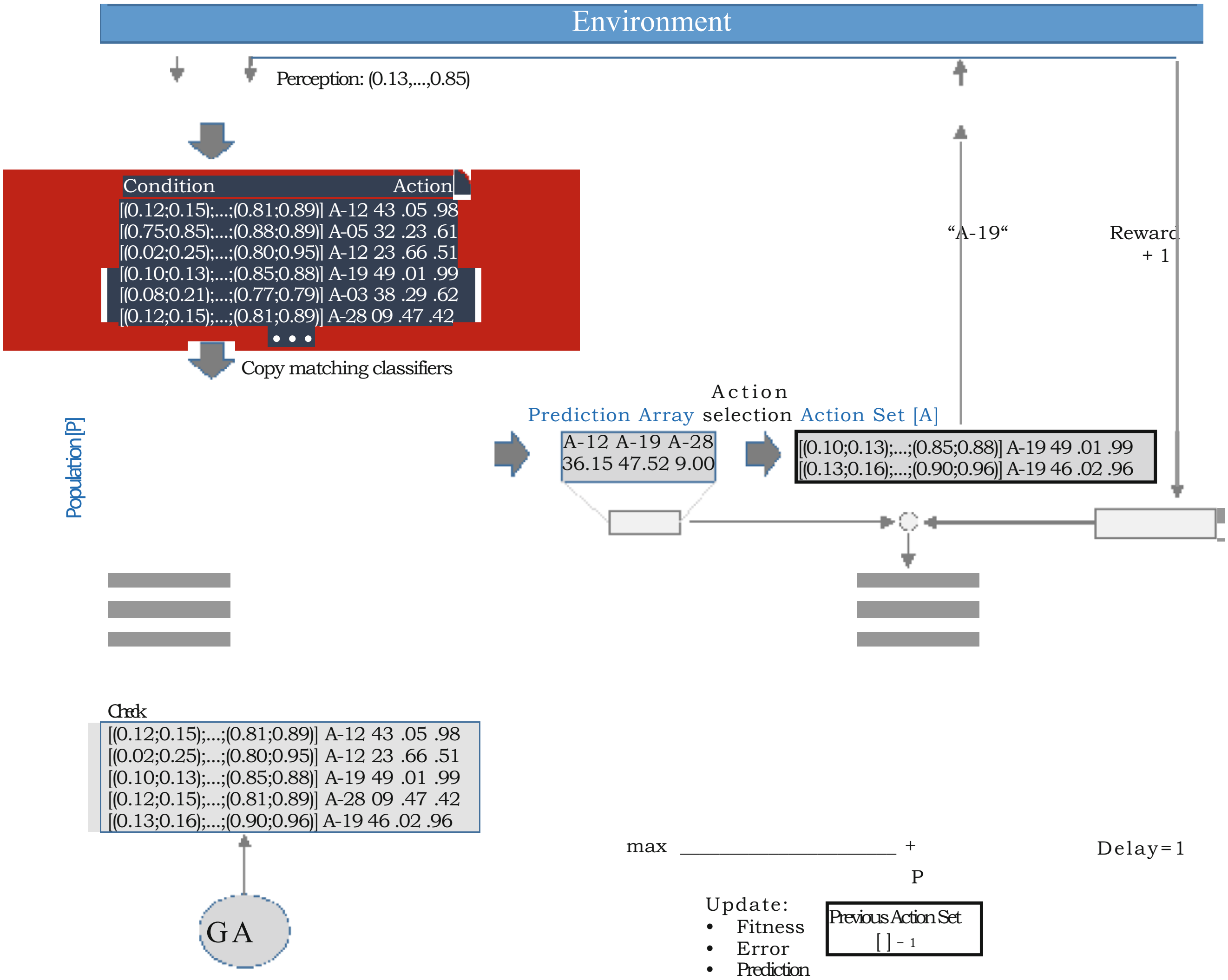
Environment
Perception: (0.13,...,0.85)
Condition Action
[(0.12;0.15);...;(0.81;0.89)] A-12 43 .05 .98
[(0.75;0.85);...;(0.88;0.89)] A-05 32 .23 .61
[(0.02;0.25);...;(0.80;0.95)] A-12 23 .66 .51
[(0.10;0.13);...;(0.85;0.88)] A-19 49 .01 .99
[(0.08;0.21);...;(0.77;0.79)] A-03 38 .29 .62
[(0.12;0.15);...;(0.81;0.89)] A-28 09 .47 .42
• • •
Copy matching classifiers
Population [P]
“A-19“
Reward
+ 1
Action
Prediction Array selection Action Set [A]
A-12 A-19 A-28
36.15 47.52 9.00
[(0.10;0.13);...;(0.85;0.88)] A-19 49 .01 .99
[(0.13;0.16);...;(0.90;0.96)] A-19 46 .02 .96
Check
[(0.12;0.15);...;(0.81;0.89)] A-12 43 .05 .98
[(0.02;0.25);...;(0.80;0.95)] A-12 23 .66 .51
[(0.10;0.13);...;(0.85;0.88)] A-19 49 .01 .99
[(0.12;0.15);...;(0.81;0.89)] A-28 09 .47 .42
[(0.13;0.16);...;(0.90;0.96)] A-19 46 .02 .96
GA
max
+
P
Delay=1
Update:
• Fitness
• Error
• Prediction
Previous Action Set
[ ] - 1

Fig. 1 Schematic illustration of the Extended Classifier System, XCS

process needs to be augmented and/or replaced by concepts using deep neural network (DNN) technology. This means keeping the algorithmic structure and initially investigating the classifier condition and matching routine, the actions, and the subsequent steps for prediction array definition and exploration.

Challenge B: Uncertainty in optimal switching problems Based on optimal stopping theory, we require a framework for an initial stochastic control decision problem that takes uncertainty into account. We need to explicitly model and analyse the incorporated uncertainty of switching decisions, which then needs to become an integral part of the algorithmic description above.

Challenge C: Runtime learning from feedback Next to the algorithmic schema of the learning technique, we have to establish the basic RL functionality. This goal can be subdivided into answering the three research questions of how the update of the classifiers of the previous cycle is done, how the covering mechanism is realised and combined with DNN-based knowledge, and how we can evolve the population over time. This further entails the aspect of how to incorporate nonstatic intervals for switching decisions and credit assignment.

Challenge D: Deep optimal switching The framework for the initial, simple switching problem is used for modelling uncertainty. This needs to be extended and revised to develop a framework for using DNNs for solving high-dimensional optimal (stopping and) switching problems with long time horizons and analysing them mathematically.

Challenge E: Active population management To establish active management of the population to improve the efficiency of the learning behaviour, we have to investigate how the update of the classifiers of the last action sets is done. This includes implications on the covering mechanism realised and combined with DNN-based knowledge and the generation of novel classifiers.

Challenge F: Exploration vs. exploitation An integral part of RL processes is always to balance between exploration and exploitation. Especially in the optimal switching problem outlined above, there is an inherent exploration- vs.-exploitation dilemma. This demands for a mathematical foundation for applying a strategy. The same holds for the selection processes in the basic scheme where a roulette wheel approach is usually used.

Challenge G: Cross Domain Fusion @ Traffic Control From a CDF standpoint, a major challenge is an inclusion of the different types of data in all stages of the analysis. For example, in addition to the obvious quantitative data the methods must always allow for the inclusion of additional qualitative data (e.g., knowledge about major events that have not yet occurred or information on new roads). In principle, the general structure of the models used allows such inclusion. However, this must be taken into account from the beginning. Consequently, using the example of an existing traffic control system allows one to demonstrate the CDF effects performed by our novel RL approach.

Conclusion

In this article, we propose to investigate novel RL technology as a basis for CDF. This technology combines the clear algorithmic structure and inherent interpretability of the gathered knowledge of Learning classifier systems (LCS) with the performance and deep learning. With an additional, theoretical mathematical foundation from the field of optimal switching theory, this is assumed to pave the way towards more intuitive human interaction as a self-assessment of decisions, knowledge and behaviour will allow for establishing self-explanatory capabilities. Especially this intended use of the model to obtain explanations of the actual learning behaviour poses challenges on both the mathematical and the computer science side.

For the underlying stochastic control problem, a DNN-based algorithm for learning the action- and no-action regions has to be developed to obtain explainable stopping rules. As a second step, mathematical structural properties (e.g., geometric properties or the relevant underlying monotone statistics for rules) have to be studied to improve the results of the algorithms and explain their behaviour.

For the algorithmic part (i.e., the computer science side), a user-understandable behaviour needs to be established. The interpretable representation from the first phase will be augmented with identification of root causes and chains of control decisions derived from the existing population. This includes guarantees for learning and decision behaviour based on corridor representations.

There are obvious limitations of our analysis. On the mathematical side, we concentrate on the challenges coming from a probabilistic and statistical analysis and just partly go deeper into the analysis of the underlying (deterministic) optimisation issues arising in the procedures based on DNN. There are also obvious limitations of our algorithmic approach. Even though the interpretability is considered in terms of the representation of the knowledge, a sophisticated user interface with a semantic description is not part of the focus. Furthermore, a transfer of knowledge among similar systems is not directly possible in the proposed concept. Subsequent efforts will have to deal with automatic adaptations of the knowledge to new (but related) problem domains, e.g., self-adaptation behaviour of an intersection controller in the traffic control system with a similar topology model.

Funding Open Access funding enabled and organized by Projekt DEAL.

References

1. Bertsekas DP (2019) Reinforcement learning and optimal control. Athena Scientific, Belmont
2. Gu S, Holly E, Lillicrap T, Levine S (2017) Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates. In: International conference on robotics and automation. IEEE, pp 3389–3396
3. Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780
4. Mnih V, Kavukcuoglu K, Silver D, Graves A, Antonoglou I, Wierstra D, Riedmiller M (2013) Playing atari with deep reinforcement learning. arXiv preprint arXiv:1312.5602
5. Peskir G, Shiryaev A (2006) Optimal stopping and free-boundary problems. Lectures in Mathematics ETH Zürich. Birkhäuser, Basel
6. Sommer M, Tomforde S, Hähner J (2016) An organic computing approach to resilient traffic management. In: McCluskey T, Kotsialos A, Müller J, Klügl F, Rana O, Schumann R (eds) Autonomic road transport support systems. Autonomic systems. Birkhäuser, Cham, pp 113–130 https://doi.org/10.1007/978-3-319-25808-9_7
7. Vinyals O, Babuschkin I, Czarnecki WM, Mathieu M, Dudzik A, Chung J, Choi DH, Powell R, Ewalds T, Georgiev P et al (2019)

Grandmaster level in starcraft ii using multi-agent reinforcement learning. Nature 575(7782):350–354

8. Vinyals O, Ewalds T, Bartunov S, Georgiev P, Vezhnevets AS, Yeo M, Makhzani A, Küttler H, Agapiou J, Schrittwieser J et al (2017) Starcraft ii: a new challenge for reinforcement learning. arXiv preprint arXiv:1708.04782
9. Wilson SW (1995) Classifier fitness based on accuracy. Evol Comput 3(2):149–175

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.