ScaRLib: A Framework for Cooperative Many Agent Deep Reinforcement Learning in Scala*

First Author$^{1[0000-1111-2222-3333]}$, Second Author$^{2,3[1111-2222-3333-4444]}$, and Third Author$^{3[2222--3333-4444-5555]}$

[1] Princeton University, Princeton NJ 08544, USA
[2] Springer Heidelberg, Tiergartenstr. 17, 69121 Heidelberg, Germany
lncs@springer.com
http://www.springer.com/gp/computer-science/lncs
[3] ABC Institute, Rupert-Karls-University Heidelberg, Heidelberg, Germany
{abc,lncs}@uni-heidelberg.de

Abstract. Multi-Agent Reinforcement Learning (MARL) is an emerging field in machine learning where multiple agents learn simultaneously in a shared environment to optimize a global or local reward signal. MARL has gained significant interest in recent years due to its successful applications in various domains, such as video games, robotics, and traffic control. Collaborative Many-Agent Reinforcement Learning (CMARL) is a relevant subclass of MARL, where thousands of agents work together to achieve a common goal through coordination, making it useful in pervasive computing and swarm robotics. Currently, most existing MARL libraries are not scalable to handle the large number of agents in CMARL. To address this issue we introduce ScaRLib, a Scala framework with bindings to state-of-the-art deep learning libraries designed to support the development of CMARL systems. The framework is designed to be easily extensible, allowing to add new algorithms and new environments. The main usecases of ScaRLib are related to large scale distributed systems, for such reason it has been integrated with the aggregate computing paradigm. This paper provides a succinct description of the main features of ScaRLib and includes a basic demonstration that showcases the tool's practical uses.

Keywords: Many Agent Reinforcement Learning · Deep Learning · Aggregate Computing.

1 Introduction

1.1 A Subsection Sample

Please note that the first paragraph of a section or subsection is not indented. The first paragraph that follows a table, figure, equation etc. does not need an indent, either.

Subsequent paragraphs, however, are indented.

* Supported by Department of Computer Science and Engineering @ Alma Mater Studiorum - University of Bologna.

Sample Heading (Third Level) Only two levels of headings should be numbered. Lower level headings remain unnumbered; they are formatted as run-in headings.

Sample Heading (Fourth Level) The contribution should contain no more than four levels of headings. Table 1 gives a summary of all heading levels.

Table 1. Table captions should be placed above the tables.

Heading level	Example	Font size and style
Title (centered)	**Lecture Notes**	14 point, bold
1st-level heading	**1 Introduction**	12 point, bold
2nd-level heading	**2.1 Printing Area**	10 point, bold
3rd-level heading	**Run-in Heading in Bold.** Text follows	10 point, bold
4th-level heading	*Lowest Level Heading.* Text follows	10 point, italic

Displayed equations are centered and set on a separate line.

$$x + y = z \tag{1}$$

Please try to avoid rasterized images for line-art diagrams and schemas. Whenever possible, use vector graphics instead (see Fig. 1).

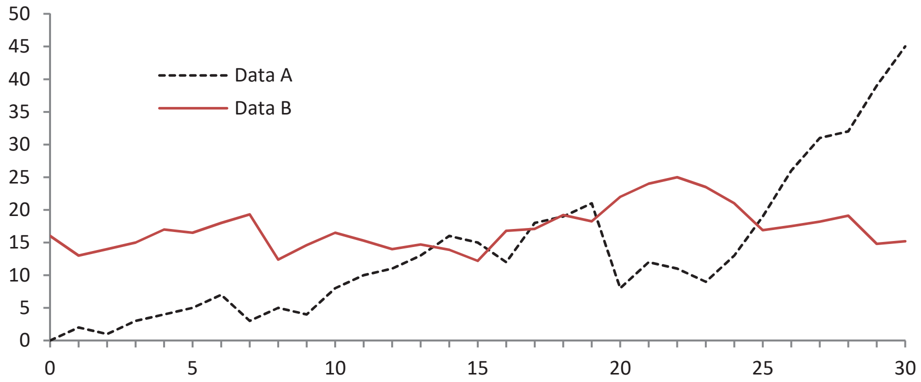

Fig. 1. A figure caption is always placed below the illustration. Please note that short captions are centered, while long ones are justified by the macro package automatically.

Program listings or code snippets in the text like **var** i:**integer**; should be set in typewriter font. The “listings” macro package can be used for reader-friendly syntax highlighting as in the following Pascal sample code:[4]

[4] For typesetting pseudocode, the “algorithmic”, “algorithm2e”, “algorithmicx”, and “program” packages are also worth considering.

```
for i:=maxint to 0 do
begin
    { do nothing }
end;
Write('end␣of␣sample␣code');
```

Theorem 1. *This is a sample theorem. The run-in heading is set in bold, while the following text appears in italics. Definitions, lemmas, propositions, and corollaries are styled the same way.*

Proof. Proofs, examples, and remarks have the initial word in italics, while the following text appears in normal font.

For citations of references, we prefer the use of square brackets and consecutive numbers. Citations using labels or the author/year convention are also acceptable. The following bibliography provides a sample reference list with entries for journal articles [1], an LNCS chapter [2], a book [3], proceedings without editors [4], and a homepage [5]. Multiple citations are grouped [1–3], [1, 3–5].

Acknowledgements Please place your acknowledgments at the end of the paper, preceded by an unnumbered run-in heading (i.e. 3rd-level heading).

References

1. Author, F.: Article title. Journal **2**(5), 99–110 (2016)
2. Author, F., Author, S.: Title of a proceedings paper. In: Editor, F., Editor, S. (eds.) CONFERENCE 2016, LNCS, vol. 9999, pp. 1–13. Springer, Heidelberg (2016). https://doi.org/10.10007/123456789_0
3. Author, F., Author, S., Author, T.: Book title. 2nd edn. Publisher, Location (1999)
4. Author, A.-B.: Contribution title. In: 9th International Proceedings on Proceedings, pp. 1–2. Publisher, Location (2010)
5. LNCS Homepage, http://www.springer.com/lncs. Last accessed 4 Oct 2017