## **Agent**Lang



# Programming Language for Agent-based Modeling

Author: Tomáš Boďa | Supervisor: Mgr. Tomáš Petříček, Ph.D. | 2024

#### Introduction

AgentLang is a programming language designed for modeling agent-based simulations. It provides a straightforward structure, where the user defines an agent model and a set of properties representing the behaviour of the agent.

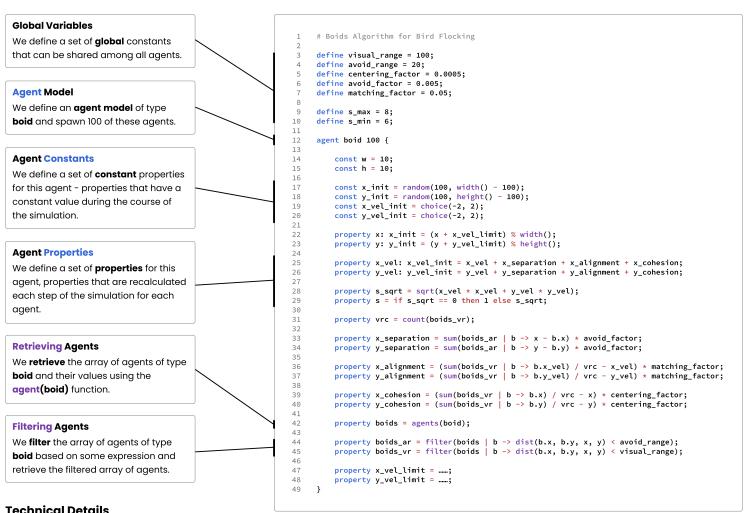
## Agent-based Modeling

Agent-based modeling is a simulation technique, where a system is modeled using a set of fundamental units of a system called agents. Each agent individually assesses the current situation and makes decisions based on a set of defined rules.

#### **Primary Goals**

Due to the increasing popularity of the agent-based modeling technique in many scientific fields, there is a need for a unified framework usable by people of all scientific fields, regardless of their technical proficiency. For these reasons, the AgentLang framework aims to provide:

- unified language with simple and straightforward syntax and structure
- only the necessary, essential core library of features
- an alternative way of modeling using a spreadsheet representation



#### **Technical Details**

AgentLang is an interpreted programming language. The source code is parsed using a pushdown automaton, which produces a semantic representation of the program in form of an abstract syntax tree (AST). The interpreter is a treewalk interpreter that traverses the AST and interprets the code node by node.

## **Interesting Concepts**

The primary structure of an AgentLang model is an agent. Its behaviour is modelled using a set of property declarations, each assigned an inline expression upon which it is recalculated. This structure is simple to understand as well as can easily be mapped to the columns of the spreadsheet interface. Moreover, interactions between agents are realized using the built-in  ${\bf set}$ comprehension structure, which takes an array of agents as the parameter, followed the name of the agent instance that is currently operated on and finally an expression based on which the set comprehension either finds, filters or sums the agent instances.

### Conclusion

The project features a fully usable programming language and a web-based interface with a code editor, a spreadsheet editor and visualisation module. The project fulfilled all of the stated goals and criteria.

#### **Acknowledgements**

I would like to thank my supervisor Mgr. Tomáš Petříček, Ph.D., whose guidance, patience and expertise have been invaluable, shaped this thesis and contributed significantly to its completion.

### **Agent**Lang

Tomáš Boďa

github.com/TomasBoda/agent-lang

