

# Sokoban With ASP-Based Hint System

Hannan Khalil, Viktor Tiger, Emil Anderö, Axel Lindh

December 5, 2025

## 1 Introduction and Motivation

Sokoban is a classical puzzle game in which a player must push boxes onto target tiles within a 2D grid world. The puzzle is well-known not only for its intuitive mechanics but also for its computational difficulty, being NP-hard. These characteristics make sokoban an excellent domain for symbolic AI techniques such as Answer Set Programming (ASP). The Motivation behind this project is to demonstrate how ASP can be integrated into an interactive system to provide reasoning capabilities such as solvability checking and intelligent hint generation. Unlike traditional sokoban implementations, our system is able to detect unsolvable states after player mistakes, compute optimal solutions, and provide real-time hints.

- *Briefly describe your chosen domain (e.g., healthcare, education, smart cities).*
- *Why is this domain interesting or useful for ontology[1]/KG/ASP modeling?*
- *Clearly define the scope of your project (what you cover, what you do not).*

Note:

- In this template, all italic text should be removed and replaced with your own text (which should not be italic); the italic text is just a placeholder letting you know what to write there.
- If you use Figures or Tables, please make sure to give each one a caption and a figure/table number and refer to them from the main text!
- References should be provided where applicable.

## 2 Methodology

Content depends on your chosen track, but structure is similar

### Ontology + KG parts (Tracks 1 + 3)

- *Describe ontology design: main classes, object/data properties, restrictions.*
- *Include a diagram or screenshot of your class hierarchy.*
- *Explain how you populated the KG (data source, manual vs. automated, number of individuals/triples).*

### ASP part (Tracks 2 + 3)

- *Describe the ASP encoding: main predicates, rules, constraints.*
- *Explain how the ASP part interacts with the ontology/KG (if applicable).*
- *Explain why ASP is suited for this reasoning.*

### Hybrid part (Track 1)

- *Include a simple architecture diagram (e.g.,  $\text{query} \rightarrow \text{KG} \rightarrow \text{fallback to LLM}$ ).*
- *Explain how the ontology/KG is used to enhance LLM responses.*

### 2.1 Evaluation

*Describe how you evaluated your system (according to the evaluation suggestions given in the project description), and present the evaluation results.*

### 2.2 Discussion and Conclusion

- *Discuss the strengths and limitations of your approach.*
- *What did you learn about combining symbolic and data-driven AI?*

## References

- [1] Guarino, Nicola, Daniel Oberle, and Steffen Staab. "What is an ontology?." *Handbook on ontologies* (2009): 1-17.