Tents and Trees

Introduction

"Tents and Trees" is a logic puzzle that requires players to place tents in a grid next to trees while adhering to specific rules.

- Tents cannot be adjacent to each other, not even diagonally.
- Each tree must have a unique tent horizontally or vertically adjacent to it.
- The numbers along the grid's edges indicate how many tents must be placed in each row or column.

This project uses Golang for parsing, validation, and testing, and an ASP program and Clingo to solve the puzzle.

Implementation

The puzzle is presented as a two-dimensional grid of cells, where each cell can be a tree, a tent, or empty (pkg/tents/types.go). It can be serialized to and from a string and asp facts (pkg/tents/puzzle.go and pkg/tents/asp.go).

The asp solvers employed to solve the puzzle are located in pkg/asp/solution and are compiled into the binary.

This project uses clingo to solve the puzzle. The puzzle is converted into ASP facts and solved using Clingo (pkg/clingo/). The resulting solution is then parsed back into a puzzle.

Dependencies

The necessary binaries for this project are clingo for puzzle solving and golang 1.21.6 for building.

The project also requires the following dependencies:

- `github.com/alexflint/go-arg`: cli parsing
- `github.com/stretchr/testify`: test assertions
- `golang.org/x/tools`: txtarchive for testdata

These direct dependencies are included for testing and cli parsing, and are vendored with the project to ensure reproducibility.

Testing

The tests are located in pkg/tents/testdata and are written in the txtarchive[1] format. Each file must contain a puzzle section. If a json section exists, the puzzle will be serialized and checked against the json. If a solution section exists, the puzzle will be solved and checked against the solution using all three solvers. The solvers will then be compared to check if they are equivalent. If an ASP section exists, the puzzle will be converted to ASP facts and checked against the ASP section.

[1] https://pkg.go.dev/golang.org/x/tools/txtar#hdr-Txtar_format

Command line usage

The command line interface is straightforward and should be easy to understand. It can be used to convert between the puzzle and ASP formats and to solve the puzzle using different ASP solutions.

```
Usage: tents [--informat INFORMAT] [--outformat OUTFORMAT] [--solution SOLUTION] [--
nosolve] [--printspaces] [FILE]

Positional arguments:
  FILE                    stdin if not given

Options:
  --informat INFORMAT, -f INFORMAT
                          puzzle | asp [default: puzzle]
  --outformat OUTFORMAT, -o OUTFORMAT
                          puzzle | asp [default: puzzle]
  --solution SOLUTION, -s SOLUTION
                          choices | disjunction | negation [default: choices]
  --nosolve, -n           don't solve the puzzle [default: false]
  --printspaces, -p      print puzzle output with spaces between cells [default: false]
  --help, -h              display this help and exit
```

The input and output format can be specified using `-f` and `-o`. The accepted formats are the puzzle format from Moodle and the ASP facts from the ASPChef link on Moodle[2]. To convert between formats without solving the puzzle, use `-n`. For example, `./tents -o asp -n puzzlefile` will convert the puzzle to ASP facts without solving it. Use `-s` to specify the ASP program to solve the puzzle.

Ethical considerations

A potential military application for a solver that reflects the logic of the Tents and Trees puzzle could be in the strategic placement of defensive units in relation to critical assets or locations, similar to the trees in the puzzle. In this scenario, the critical assets (analogous to trees in the puzzle) could represent essential infrastructure, command centres, or other crucial resources that require protection. The defensive units, which are similar to tents, may consist of various types of forces, including ground troops, anti-aircraft systems, surveillance units, or unmanned drones, that are responsible for protecting these assets.

The primary ethical challenge pertains to the autonomous operation of systems that can use lethal force. It is important to ensure that human oversight is maintained to prevent any unintended consequences. While such systems can optimize defense postures with precision and efficiency, it raises concerns about the autonomy of decision-making processes and the need for human oversight. Any operational deployment of these systems must strictly adhere to international humanitarian laws, emphasizing the principles of distinction and proportionality.

To ensure accountability, it is essential that the deployment of autonomous defensive units adheres to ethical and legal standards and is subject to human oversight. Decisions, particularly those with life-or-death consequences, must be subject to human judgment. Questions regarding responsibility for the actions of AI systems, particularly in cases resulting in unintended civilian casualties or conflict escalation, highlight the necessity for well-defined protocols that maintain human agency in critical decision-making loops.

[2] https://asp-chef.alviano.net

The use of artificial intelligence in defence settings may trigger an arms race in military AI technologies, which could exacerbate global military tensions and destabilise already volatile regions. Urgent international cooperation is needed to establish norms and regulations that govern the development and deployment of military AI. This is to ensure that advancements in defence technologies do not compromise global peace and security. The prospect of escalating militarisation underscores the significance of this.

AI systems used in military contexts must adhere to international humanitarian law. This requires the systems to differentiate between combatants and non-combatants and evaluate the necessity and proportionality of force used in complex conflict situations. The deployment of AI in military applications must be ethical and committed to protecting civilian lives while upholding the integrity of combatant engagement. Current AI systems face a challenge in navigating the nuances of battlefield ethics.