User Manual

**Wandering in the Woods Game in Python**

Introduction

This repository consists of the implementation of Wandering in the Woods Game/ Simulation in Python.

The following algorithms are currently implemented:

- [Wandering in the Woods Game in Python](#multi-agent-path-planning-in-python)

- [Introduction](#introduction)

- [Dependencies](#dependencies)

- [Centralized Solution] (#centralized-solution)

- [Conflict Based Search](#conflict-based-search)

- [Execution](#execution-1)

- [Results](#results-1)

- [Reference](#reference-1)

- [Post-Processing](#post-processing)

- [Post-processing with TPG](#post-processing-with-tpg)

## Dependencies

Install the necessary dependencies by running.

```shell

python3 -m pip install -r requirements.txt

```

## Centralized Solutions

In these methods, it is the responsibility of the central planner to provide a plan to the robots.

### Conflict Based Search

Conclict-Based Search (CBS), is a multi-agent global path planner.

#### Formatting

* Input.yaml can be edited as per configurations

agents:

* start: [0, 0] goal: [6, 6] name: agent0
* - start: [2, 7] goal: [6, 5] name: agent1
* - start: [0, 7] goal: [6, 4] name: agent2

map:

dimensions: [8, 8]

obstacles:

- !!python/tuple [0, 1]

- !!python/tuple [2, 1]

- !!python/tuple [3, 3]

- !!python/tuple [4, 3]

- !!python/tuple [5, 5]

#### Execution

Run:

```

cd ./Wandering in the Woods Game - Py

python3 Wandering in Woods-Solution Finder.py input.yaml output.yaml

```

#### Results

To visualize the generated results:

``` shell

python3 Wandering in Woods-Visualizer.py input.yaml output.yaml

```

#### Reference

- [Conflict-based search for optimal multi-agent pathfinding](<https://www.sciencedirect.com/science/article/pii/S0004370214001386>)

Results:

