Collaboration in AI

Final Project

Benny Skidanov 322572926 , Matan Hazan 315198796

The natural world and the way termites construct their underground nests serve as inspiration for our project. By using Multi Agent STRIPS to mimic this type of architecture, we hoped to recreate it.

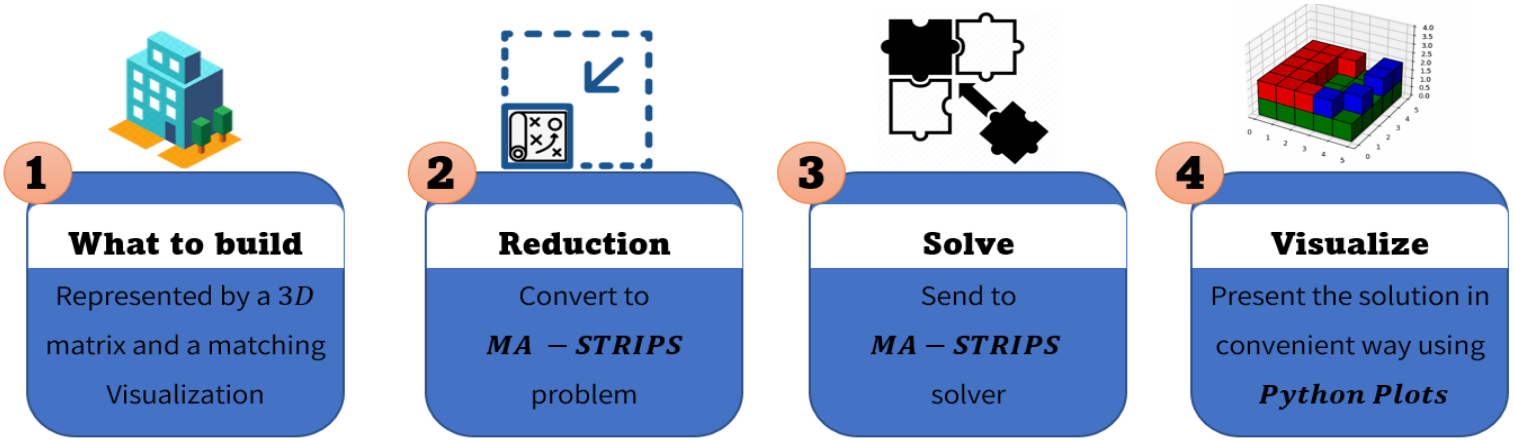
Our project's pipeline is quite straightforward.

A 3D binary matrix representing the desired construction is first produced; 1 simply denotes a block at a certain location. Additionally, we receive the quantity of agents required for the building process, and we can infer the grid size from the supplied matrix.

Then, we transform this problem into an MA-STRIPS problem.

We submit the problem to a solver, which returns a solution in the form of a plan, which we then insert into a program we made to display the plan.

Pipeline



Model

Grid : 5 X 5 X max height of building (= 4)

Agents : 2-4

Predicates

```
(:predicates
    (atagent ?a - agent ?src - location)
    (atbrick ?dst - location)
    (freecell ?src - location)
    (freeagent ?a - agent)
    (adjacent_horizontal ?src - location ?dst - location)
    (adjacent_vertical_up ?loc - location ?loc - location)
)
```

Actions & Effects

```
(:action agent_move
    :parameters (?a - agent ?src - location ?belowdst - location ?dst - location)
    :precondition (and
        (atagent ?a ?src)
        (freecell ?dst)
        (adjacent_horizontal ?src ?dst)
        (adjacent_vertical_up ?belowdst ?dst)
        (atbrick ?belowdst)
    )
    :effect (and
        (atagent ?a ?dst)
        (not (atagent ?a ?src))
        (not (freecell ?dst))
        (freecell ?src)
    )
)
```

```
(:action agent_climb
    :parameters (?a - agent ?src - location ?base - location ?dst - location)
    :precondition (and
        (atagent ?a ?src)
        (freecell ?dst)
        (atbrick ?base)
        (adjacent_horizontal ?src ?base)
        (adjacent_vertical_up ?base ?dst)
    )
    :effect (and
        (atagent ?a ?dst)
        (not (atagent ?a ?src))
        (not (freecell ?dst))
        (freecell ?src)
    )
)

(:action agent_slide
    :parameters (?a - agent ?src - location ?standingbase - location ?base - location ?dst - location)
    :precondition (and
        (atagent ?a ?src)
        (freecell ?dst)
        (atbrick ?base)
        (atbrick ?standingbase)
        (adjacent_horizontal ?standingbase ?dst)
        (adjacent_vertical_up ?base ?dst)
        (adjacent_vertical_up ?standingbase ?src)
    )
    :effect (and
        (atagent ?a ?dst)
        (not (atagent ?a ?src))
        (not (freecell ?dst))
        (freecell ?src)
    )
)
```

```
(:action pickup-brick
    :parameters (?a - agent ?src - location ?above - location ?dst - location)
    :precondition (and
        (atbrick ?dst)
        (adjacent_horizontal ?src ?dst)
        (adjacent_vertical_up ?dst ?above)
        (freecell ?above)
        (atagent ?a ?src)
        (freeagent ?a)
    )
    :effect (and
        (not (freeagent ?a))
        (not (atbrick ?dst))
        (freecell ?dst)
    )
)

(:action put-brick
    :parameters (?a - agent ?src - location ?below - location ?dst - location)
    :precondition (and
        (adjacent_horizontal ?src ?dst)
        (atagent ?a ?src)
        (adjacent_vertical_up ?below ?dst)
        (atbrick ?below)
        (not (freeagent ?a))
        (freecell ?dst)
    )
    :effect (and
        (freeagent ?a)
        (atbrick ?dst)
        (not (freecell ?dst))
    )
)
```

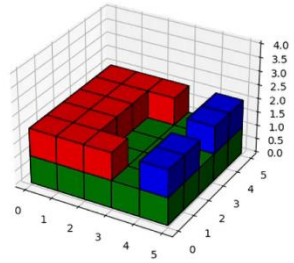
Code

Code + Reports + Presentation can be found here

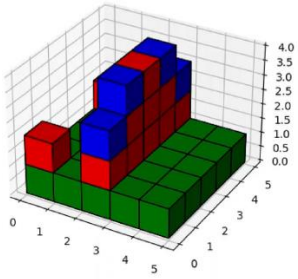
$$https://github.com/matanhaz/Collaboraion-in-AI-Project$$

Example

Start Stare



End Goal



Experiments

We conducted a number of experiments to observe how the entire pipeline outlined in the preceding section operates.

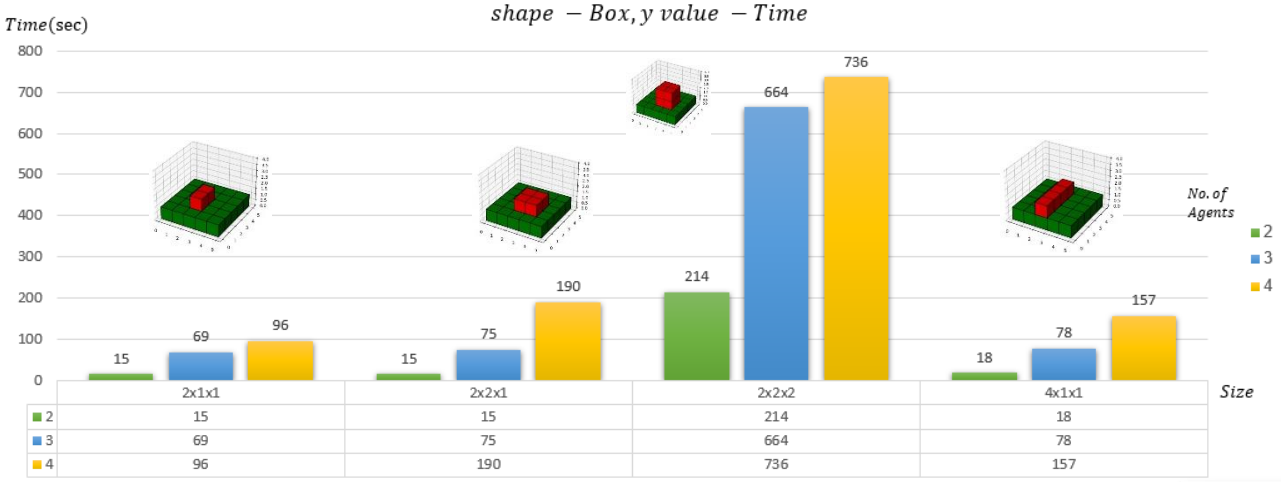
The results were recorded using the following metrics: time for the planner to output a plan, makespan of the plan, and the number of expanded states by the planner in order to find a plan.

We tested 2 shapes (box and staircase) with various volumes and agent counts (2, 3, 4).

The results and an illustration of the desired shape created using our visualization script are shown in the following section.

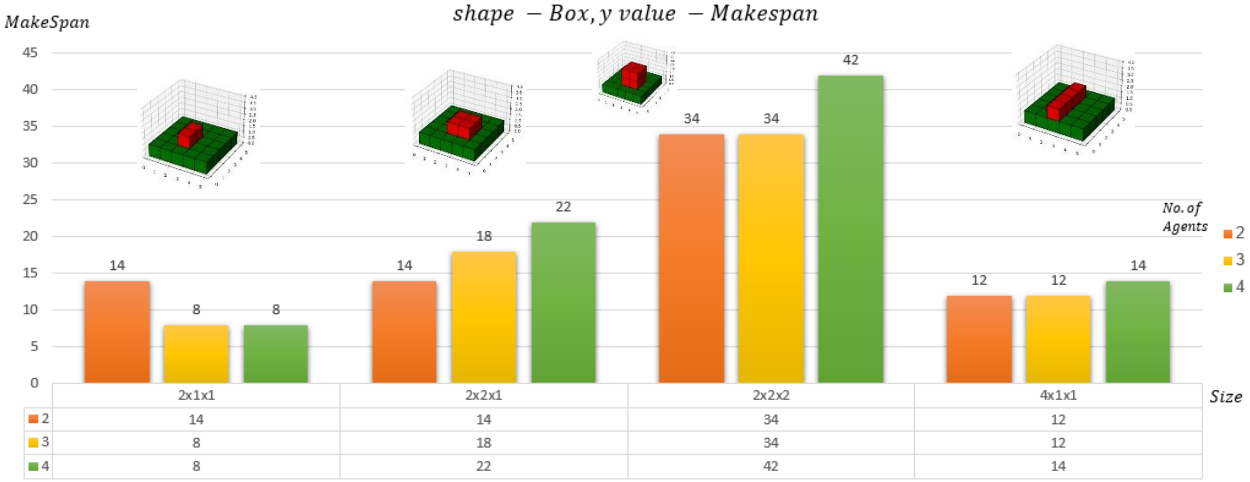
Results

Results



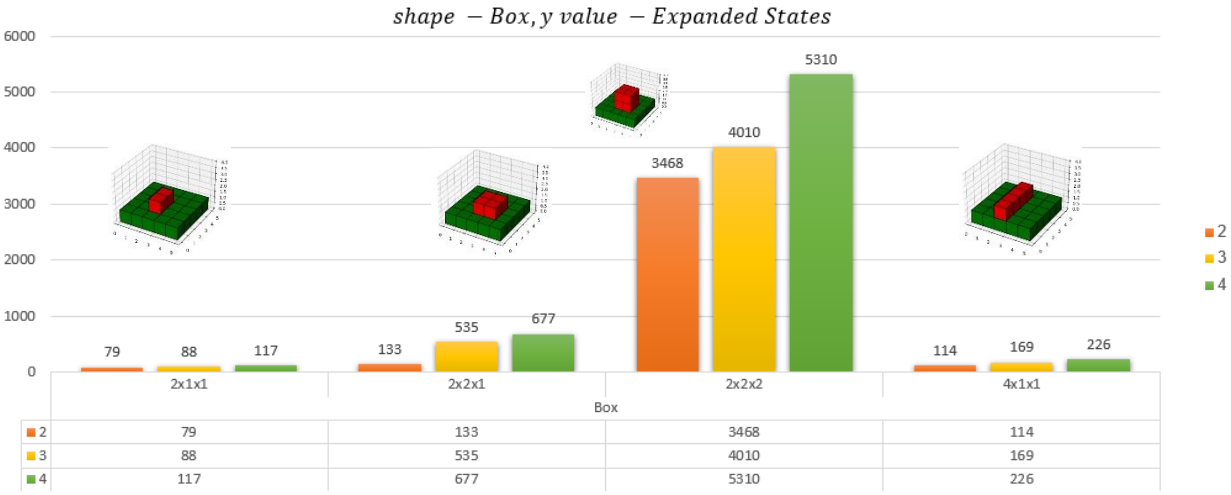
	2x1x1	2x2x1	2x2x2	4x1x1
2	15	15	214	18
3	69	75	664	78
4	96	190	736	157

Results



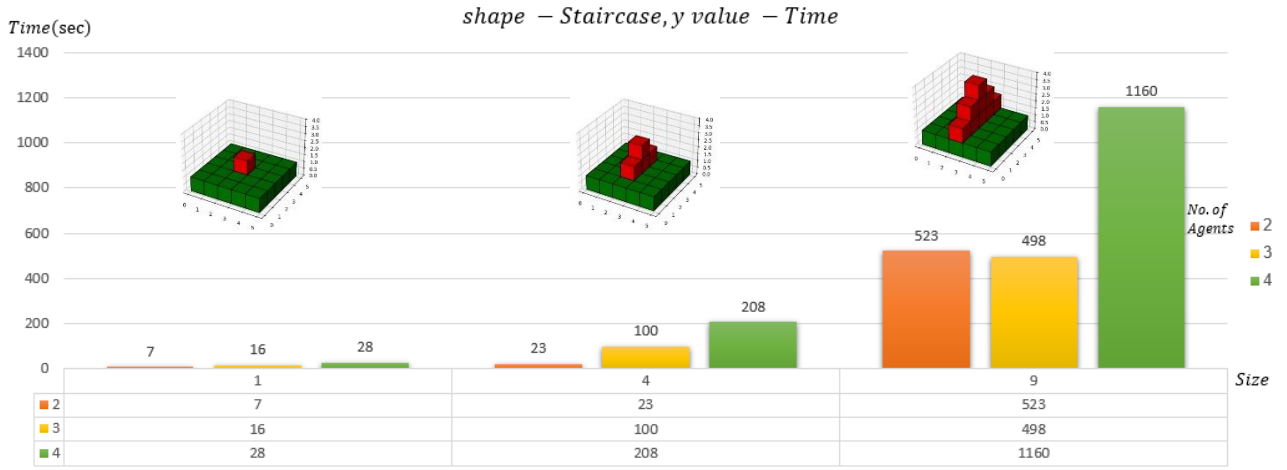
	2x1x1	2x2x1	2x2x2	4x1x1
2	14	14	34	12
3	8	18	34	12
4	8	22	42	14

Results



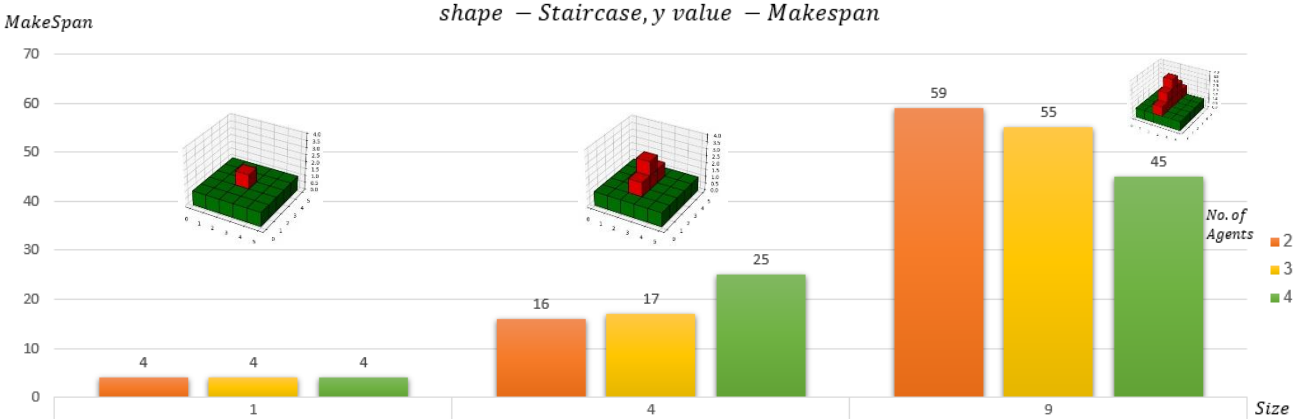
	2x1x1	2x2x1	2x2x2	4x1x1
2	79	133	3468	114
3	88	535	4010	169
4	117	677	5310	226

Results



	1	4	9
2	7	23	523
3	16	100	498
4	28	208	1160

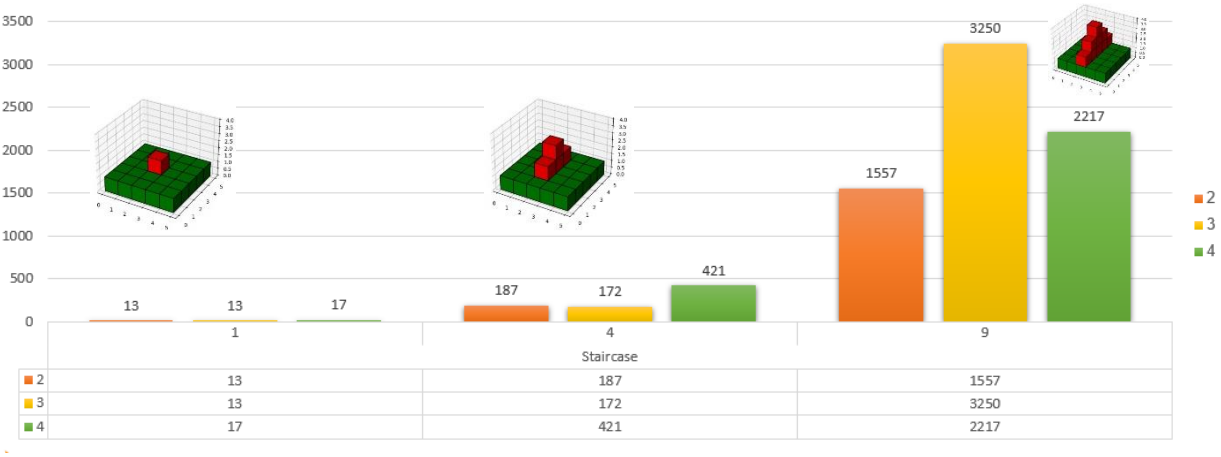
Results



	1	4	9
2	4	16	59
3	4	17	55
4	4	25	45

Results

shape — Staircase, y value — Expanded States



	Staircase		
	1	4	9
2	13	187	1557
3	13	172	3250
4	17	421	2217