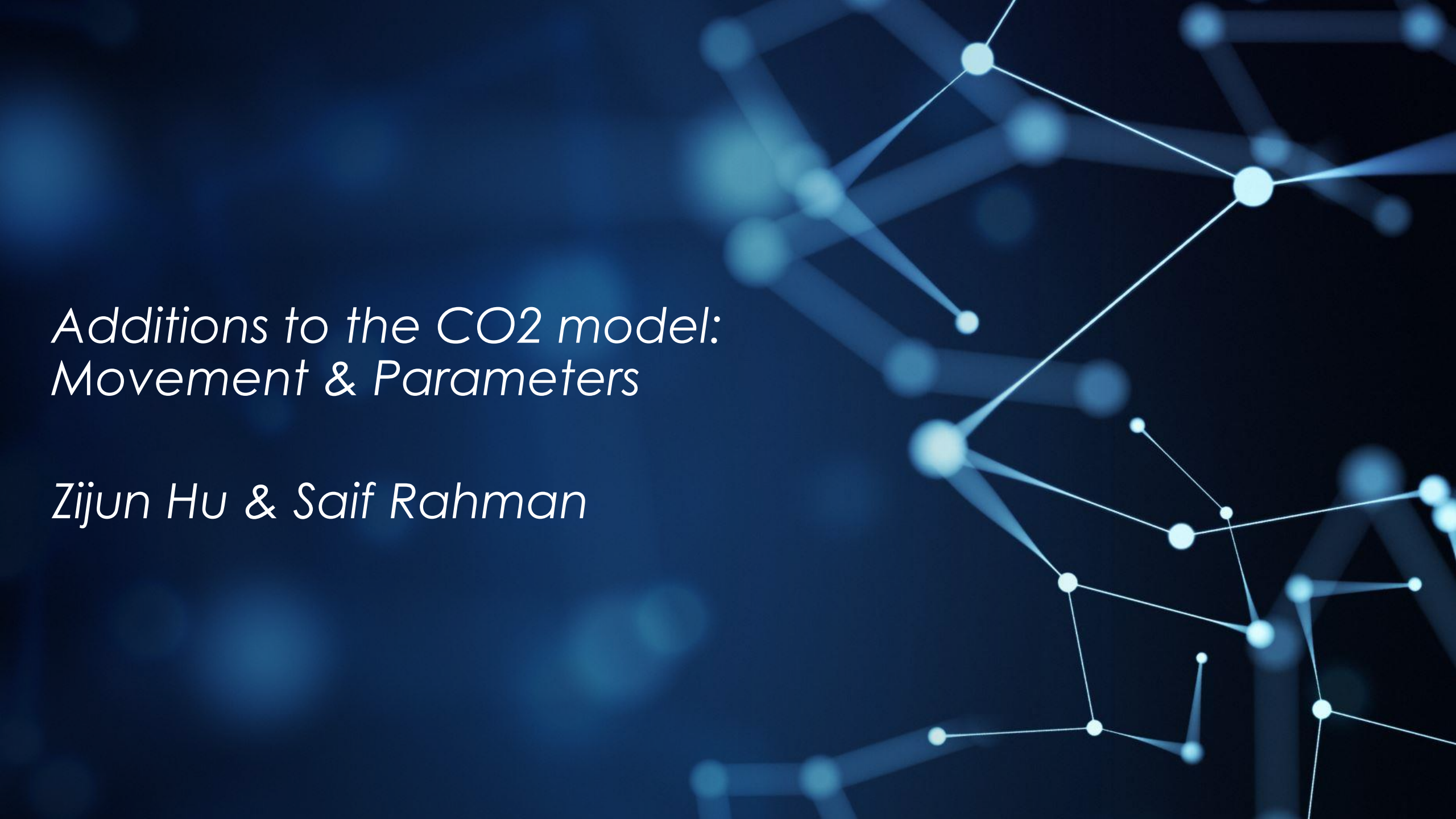
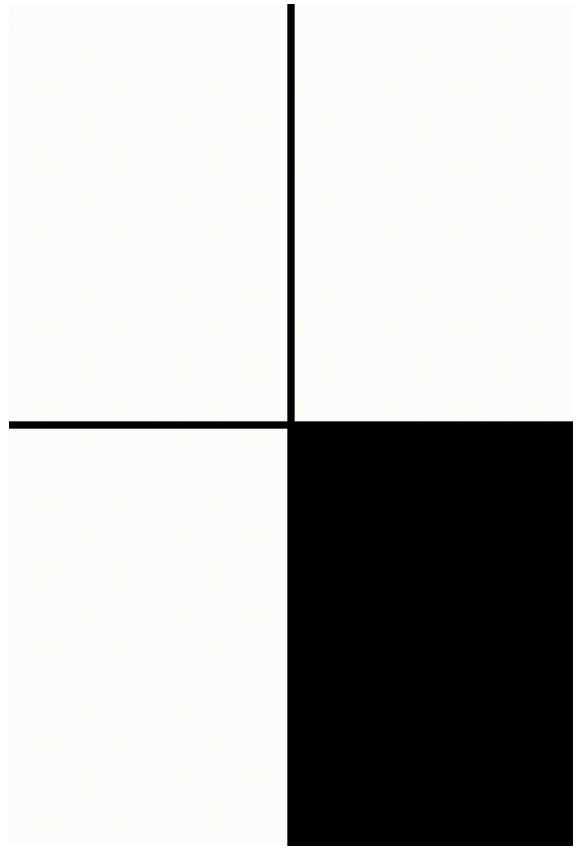


Additions to the CO2 model: Movement & Parameters

Zijun Hu & Saif Rahman

Previous state of model

- The original model had no movement
- Many of the core settings in the model (e.g cell size, CO2 concentration etc.) were hardcoded, so any change required a recompilation and would be harder to change for an average user



The need for movement

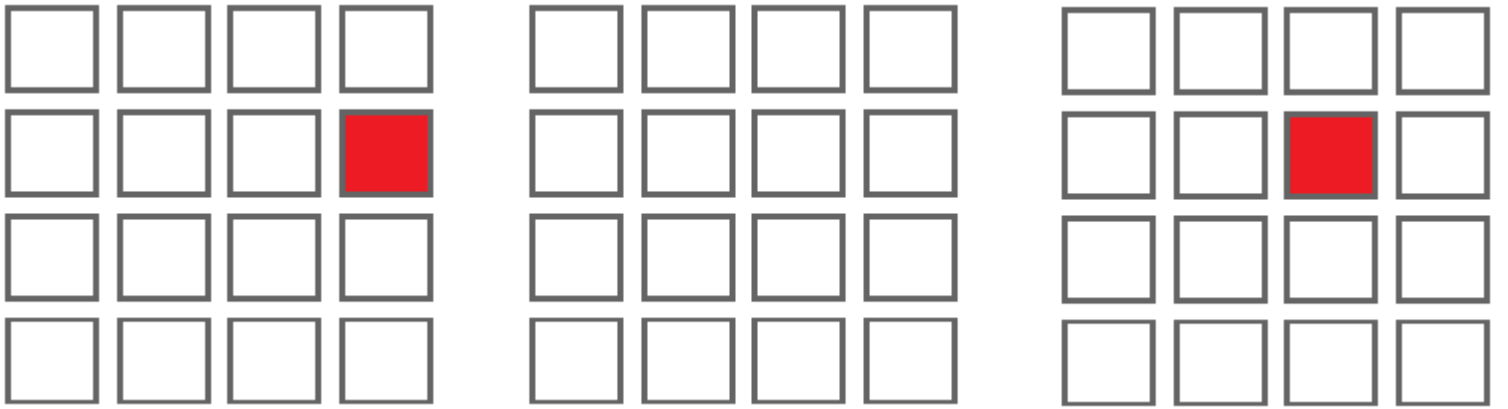
- The CO2 model assigns a type to each cell, this can be Air, CO2 Source, Impermeable Structure, Door, Window, Workstation or Vent.

- Originally it would wait a set number of states(around 30), turn all workstations into CO2 sources and then after another number of states (around 250) it would turn them back into workstations

- This works well but does not portray movement from the door to the station and back

Idea behind the movement

By changing the current cell type and update the adjacent cell type, students can move.



The decomposition of the students' movement

Implementation

Movement route:

1. Generate CO2 Source near the entry.
2. Start moving to the workstation.
3. Stay at workstation for a while.
4. Move to the exit and leave the room.

Support parameters and new functionalities

- Support parameters:
  - studentsList: A list includes all CO2 sources that generated
  - workstationsList: A list includes all exist workstations
  - **actionList**: A list that records the future actions for CO2 sources movement

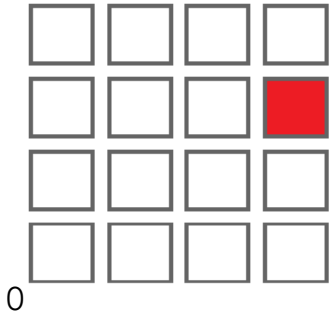
- New functionality:
  - setNextRoute(): Calculate the next location for CO2 sources
  - **navigation()**: Integrate pedestrian behaviors[1] to avoid collisions
  - moveCheck(): Check if the next location is occupied

[1] A. Al-Habashna and G. Wainer, "Modeling pedestrian behavior with Cell-DEVS: theory and applications," SIMULATION, vol. 2, no. 92, pp. 117-139, 2016.

actionList

A list that records the future actions for CO2 sources movement

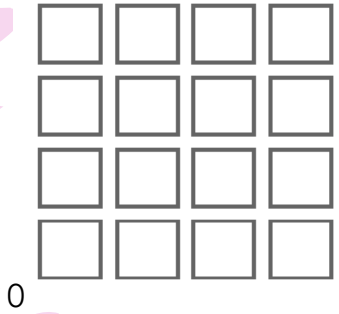
1. Cell(3,2) come in for local computation



Generate CO2 Source at cell(3,2)

Add the future action to the actionList
change cell(3,2)'s type to AIR

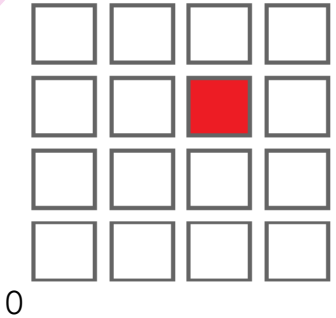
2. Cell(3,2) come in for local computation



Change cell(3,2)'s type to AIR
Remove this command from the actionList

Add the future action to the actionList
change cell(2,2) type to CO2 Source

3. Cell(2,2) come in for local computation



Change cell(2,2)'s type to CO2 Source
Remove this command from the actionList

Add the future action to the actionList
change cell(2,2) type to AIR

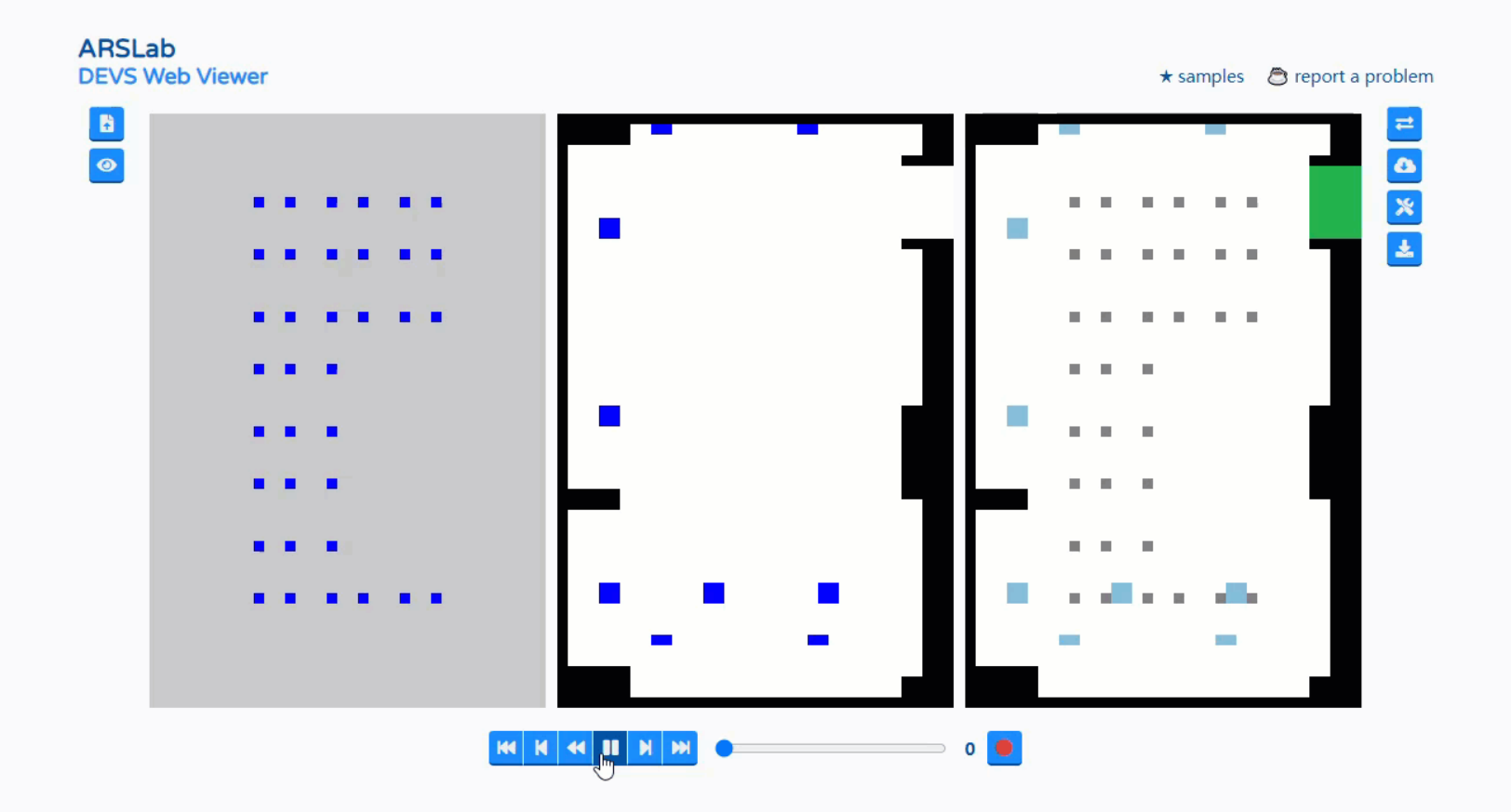
By using the actionList, each cell only manage their own type.

It's simpler and easy to manage.

Navigation

- When the JSON is first read in, we store all the locations of the workstations
- Each student generated is then assigned to a unique workstation
- To model movement in the lab scenario more accurately we have made it so that students move along the y-axis first and then the x-axis

End result



Added parameters

- All added parameters are in the json used to configure the model

- Cell size: this is used to set the length, width and height for each cell, this in turn is used to calculate the volume in the model
- CO2 concentration increase : this value can be changed to accommodate different co2 levels
- start time: the step at which the non-movement model occupies the workstations
- time active: the no. of steps the workstations remain occupied in the non-movement model

What's next?

- Further additions to parameterize the number of CO2 sources generated and the delay in between generation
- Allow users to pick the spot where sources are generated and the exit point
- Add walk through vent support in 2D model