# Cell-DEVS CO2 Models With Occupants and Ducts

Mohammed Al-Saeedi
*Dept. of Systems and Computer Engineering*
*Carleton University*
Ottawa, Canada
MOHAMMEDALSAEEDI@cmail.carleton.ca

Gabriel A. Wainer
*Dept. of Systems and Computer Engineering*
*Carleton University*
Ottawa, Canada
gwainer@sce.carleton.ca

***Abstract*— Cell-DEVS is a formalism that enables the simulation of cellular models asynchronously with different timing delays. Indoor CO2 diffusion by breathing occupants has been studied taking into consideration room dimensions, ventilation, doors, windows, and occupants as a source. We present a method to define floor plans, and create 2D or 3D models quickly. We added ducts between rooms and show the indoor concentration and diffusion of CO2 with the presence of these ducts.**

***Keywords*— *Cell-DEVS, Sustainability, CO2 models.***

## I. Introduction

Measuring Carbon Dioxide (CO2) and studying its diffusion indoors has many applications, which include, but not limited to, maintaining air quality, conserving energy, and minimizing viral infections. On the one hand, many of the experiments needed to conduct studies, like the indoor diffusion of CO2, are complex to implement, models have been created to conduct such studies. On the other hand, adjusting each model manually to mimic the studied space is difficult and prone to error. We defined a CO2 diffusion model by breathing occupants using the Cell-DEVS formalism [1]. The model takes into consideration room dimensions, ventilation, doors, windows, and occupants as a source of producing CO2 [2].

Although the original tools are useful in simulating different configurations (e.g. room dimensions, window locations, and occupant's mobility), the tools have limitations as the CO2 spread model only considers 2D cellular models and the 3D interactions are not taken into consideration. Similarly, it is complex to build new experiments with different spatial configurations. To address these issues, we introduce a method to quickly build 2D or 3D scenarios for a Cell-DEVS CO2 model. We introduce different views, in particular new methods for modeling ducts between rooms and their influence in the indoor concentration and diffusion of CO2. The tool automates the simulation of models of CO2 dispersion indoors. We discuss the tool's applications, the software architecture, related tools, and a case study comparing indoor spaces with different configurations and how they affect CO2 concentration. We first explain essential background information to position the presented work. Then, we introduce the experimental setup and present various versions of the model with configurations that are replicas of real-life laboratories. Finally, we present the conclusion and propose future improvements.

## II. Related Work

There are many applications for CO2 models [3]. They can help to maintain a balance between indoor comfortable CO2 levels, while minimizing the energy consumed by ventilation systems. They can be used for detecting occupants [4], but their configuration is sensitive, thus, dispersion models help to find the best locations for the sensors and simulate their behavior based on the parameters of the indoor environments. CO2 concentration has recently been proposed as a proxy for the possibility of viral infection indoors [5], which is particularly useful in the case of the COVID-19 pandemic [6].

In [7] the authors proposed a method for tailoring models for specific spaces. The model takes into consideration the CO2 concentration, ventilation, and multiple occupants. This is used to predict CO2 concentration levels in the room. The model is only effective for short term predictions of CO2 concentration levels. The thesis dissertation presented in [8] focuses on the dispersion of hazardous gasses in closed spaces using Cellular Automata to model the influence of the spread of gas on the behavior of pedestrians. The objective is to aid designers in building public spaces that are safe during evacuations using a 2D model. Authors of [9] built a model to simulate CO2 dispersion in a bedroom with one sleeping occupant. The results show that the air quality in small rooms is unhealthy, and thus extra ventilation is required. In [10], the authors developed multi-compartment indoor air quality models based on the mass balance equation to simulate CO2 concentration. The modeling considers classroom settings for predicting CO2 levels based on the given parameters of the indoor space. Authors of [2] studied the relationship between the CO2 levels and the room configurations and how minor changes may result in significant changes in indoor CO2 levels. The model takes into consideration the configuration of ventilation, windows, doors, and occupants.

Ducts between rooms can also play a significant role in changing the indoor CO2 levels. Therefore, here we will introduce different indoor configuration of ducts between rooms and evaluates the rate of its effect on the concentration and diffusion of CO2. The research in [2] builds a cellular model to determine the best placement of CO2 sensors for occupancy detection. In [11], Khalil and Wainer advanced the previous model and validated the results using data from real-life indoor spaces. The results resembled the ground truth data of the physical system. These studies are built using Cell-DEVS and the CD++ toolkit.

Cell-DEVS [1] allows building cell spaces in which each cell is a discrete event systems specification (DEVS) with explicit timing delays. Cell-DEVS defines a spatial model as a collection of cells arranged in a grid. Cell-DEVS has been used extensively to study many complex systems and modeling applications. This

978-1-6654-9799-2/22/$31.00 ©2022 IEEE

includes, but is not limited to, environmental, biology, pedestrian and other spatial models. The applicability and advantages of Cell-DEVS as a formalism to model CO2 diffusion in closed spaces have been presented in the literature. Among the advantages of Cell-DEVS is providing asynchronous execution, which results in a better execution time. All cells in the space include a common local computation function that uses internal state values and values received from neighboring cells to calculate a new value for the cell. Once the transition function is executed, its output is transmitted after a period of time specified by a delay function. Fig. 1 shows a schematic view of a cell and a 2-dimensional space with Von-Neumann neighborhood (N/S/E/W neighbor cells).

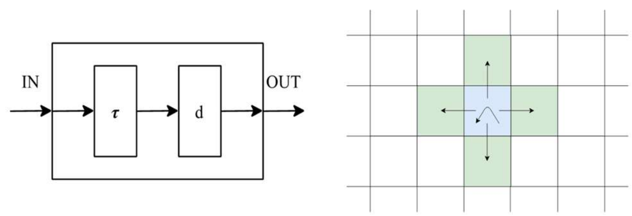


Fig. 1. Cell-DEVS model: (a) schematic of a cell; (b) 2-D Cell-DEVS.

When a cell receives an input is received, it triggers the local computation function (τ) to compute its next state. If the state of the cell changes based on the inputs, the new value is transmitted after a time specified by the delay function (d), which is a function of the state of the cell. The outputs of a cell are subsequently transmitted to the cells in the neighborhood set. Cell-DEVS uses continuous time and a discrete-event approach: if a cell becomes quiescent, it is not computed until new activity is detected. One advantage of Cell-DEVS, and its supporting tools, is the improved execution time. This is attributed to the fact that Cell-DEVS provides asynchronous execution to model the asynchronous nature of complex systems. Also, Cell-DEVS formalism offers ways to define complex timing conditions. Besides, there is an extensive set of tools available for translating the formalism into an executable model.

There are different simulators to execute Cell-DEVS models. We used the Cadmium tool, which allows users to define model inputs using JavaScript Object Notation (JSON), a data format to store and transmit large amounts of human-readable data. JSON stores data in key-value pairs allowing for the simple representation of neighborhoods, their attributes, and their relationships. Cadmium allows the user to include complex geographical inputs that load into the model at run time resulting in a flexible model that allows for efficient rapid prototyping. This section presents a new version of the Cadmium simulator that supports both classic and asymmetric Cell-DEVS models. Cadmium is a header-only library written in C++ that allows the modeling and simulation of computational models based on the DEVS formalism. In Cadmium, cells are implemented in C++ and the cell space is defined using a JSON configuration file. This approach allows us to study multiple setups by simply modifying the configuration file, thus avoiding recompilations and reducing the overall time required for exploring a scenario. Furthermore, modelers can integrate Cell-DEVS models with other DEVS models implemented with Cadmium.

First, we need to define a conceptual cellular model describing the system under study following the asymmetric Cell-DEVS formalism. The conceptual model is then translated into a computational model using the tools provided by the Cadmium library. Once we have implemented the corresponding computational model, we can run simulations over different scenarios by modifying the JSON configuration file. Then, we analyze the simulation results to gain insight into the system under study. Once we define the mathematical model of the cell space, implementing it in the Cell-DEVS tools CD++ or Cadmium is straightforward [1]. Cadmium allows defining the components in C++. First, we declare a structure used representing the state of a cell. For instance:

```
struct SIRD {
    std::vector<int> population;
    std::vector<float> susceptible;
    std::vector<float> infected;
    std::vector<float> recovered;
    std::vector<float> deceased;
}
```

The formalism allows defining connectivity and mobility factors are parameters that determine how neighboring cells affect each other. These characteristics are known as cell vicinities, and are described as:

```
struct MC {
    float connectivity;
    std::vector<float> mobility;
}
```

When a cell receives a state change message from any neighboring cell, the local computation function is triggered in the cell. This is defined in the local_computation method as follows:

```
SIRD local_computation() const override {
  SIRD res = state.current_state;

  auto new_i = new_infections(res);
  auto new_r = new_recoveries(res);
  auto new_d = new_deaths(res);

  for(int n=0; n<n_age_segments();n++) {
    res.infected[n]+=new_i[n]-new_r[n]-new_d[n];
    res.recovered[n] += new_r[n];
    res.deceased[n] += new_d[n];
    res.susceptible[i] = 1 - (res.infected[n] +
res.recovered[n] +

res.deceased[n]);
  }
  return discretize(res);
}
```

The output_delay function defines the temporal behavior of the cells. In our model, the timing behavior is simple: cells wait one day before transmitting a state change to neighboring cells:

```
T output_delay(sird const &cell_state) const
       override { return 1; }
```

Contrary to previous research that deals with the problem in a case by case manner and considers a small subset of the configuration parameters, we offer a generic model of CO2 dispersion using well-established formalism that is supported by tools. It is worth noting that the objective of this research is not to estimate the number of occupants in the room based on CO2 levels, but to provide a mechanism for studying the effect of the space settings on the measurement and the dispersion behavior of CO2. The presented solution reaches this objective while considering different configurations in the space where the CO2 sensor is to be installed; a problem that was raised by researchers in the field of occupants' detection. The model allows representing closed spaces of different sizes in 3D, a varying number of occupants, and means for CO2 to escape (an open door, a window or a ventilation port).

## III. 2D AND 3D MODELING OF CO2

In order to execute 2D or 3D scenarios for the Cell-DEVS CO2 models, we need to consider the structure, configuration, and functionalities of the tool. We use pre-defined configurations that allow to draw or import a 2D floor plan grid and convert it to a 2D or 3D Cell-DEVS CO2 model scenario as shown in Figure 2.

There are three main interacting modules. DrawGrid is used for drawing a 2D grid using the pre-defined simulation and cell configuration. It includes functionality to create a 2D and 3D Cell-DEVS CO2 models, preview the model in 3D view, and save the model for later use. The ConvertTool module creates a JSON structure of the output model while the GenerateTool module exports the output model. DrawGrid creates a model considering the dimensions of the closed space, ambient CO2 concentration, size and location of CO2 sinks (i.e. windows, doors, and ventilation ports), locations where occupants may exist, the breathing rate of occupants based on their activity level, concentration increase due to breathing occupants, and dimensions of the room. The model assumes ambient outdoor CO2 concentration of 400 particles per minute (ppm) based on the American Society of Heating, Refrigerating and Air-Conditioning Engineers standards [13]. However, this value can be adjusted as a parameter specified for each scenario, which is defined using a JSON file.

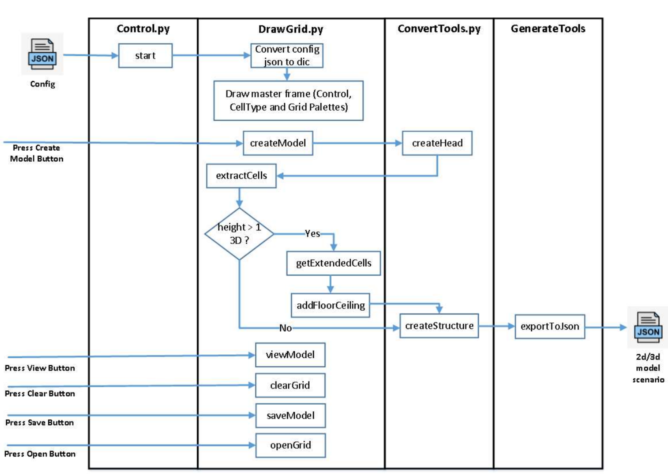


Fig. 2. Software modules and workflow

Human breathing is calculated based on the fact that humans breathe every five seconds, and the produced CO2 in every breath (exhaling and inhaling) is a parameter that depends on the activity level [14]. The general model has seven types of cells: (1) walls and obstacles that do not allow CO2 diffusion, (2) air cells whose CO2 concentration is dependent on the concentration values in their neighborhoods, (3) CO2 sources with a periodic increase in the CO2 level added at an interval to mimic breathing in addition to the CO2 diffused from the neighborhood, (4) open doors that diffuse CO2 to the rest of the building with a fixed indoor background CO2 level, (5) open windows that are also CO2 sinks with a fixed outdoor background CO2, (6) vents that diffuse gas through HVAC system with a reduced constant CO2 level, and (7) workstation cells that act as normal air cells when not occupied and as CO2 sources when occupied. The CO2 diffusion is calculated by averaging the concentration level in the Moore neighborhood of each cell. This means that to get the concentration of each cell, the concentrations in either 27 or 9 cells are averaged in the cases of 3-D and 2-D models, respectively [2].

Figure 3 illustrates the elements that could be present in the model. Parameters that we consider at this stage are the dimensions of the room, the locations of window/door and ventilation ports, the CO2 sensor placement, and the presence of occupants. We represent the closed space as a set of neighboring cells in a 2-dimensional Cell-DEVS model with different CO2 levels. We note here that we based our calculations on the facts that normal background CO2 levels measured in particle per million (ppm) range from 300 to 400, while CO2 levels in an occupied space with normal ventilation can range from 400 to 1000 ppm, and the average person exhales 0.5 L of air per breath of which 3.8% (or 38,000 ppm) is CO2. Most people breathe once every 5 seconds corresponding to an output of 228 mL of carbon dioxide every minute. Hence, we have in the model six types of spaces where the gas diffuses according to different rules: (1) open-air spaces with constant ppm CO2 level, (2) walls that are impermeable and do not allow CO2 to diffuse through them, (3) CO2 sources with a fixed level of CO2 added at an interval to mimic breathing, (4) open doors that diffuse CO2 to the rest of the building with a fixed indoor background CO2 level of 500 ppm, (5) open windows that are also CO2 sinks with a fixed outdoor background CO2 of 400 ppm, and (6) vents that diffuses gas through HVAC system with a reduced CO2 background level.

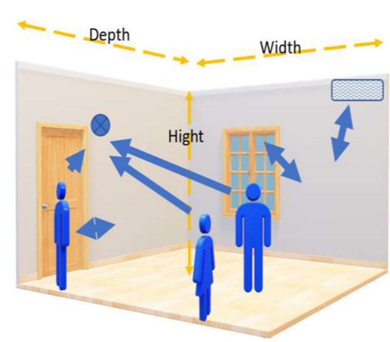


Fig. 3. Model Parameters

Each cell in the model represents 25cm × 25cm × 25cm spaces and therefore has a 15.625 L volume of air. The average person exhales 0.5 L of air with every breath at a concentration of 3.8% resulting in 19 mL of CO2 being added to the surrounding air volume with each breath. The increase in CO2 for a standard cell volume is, therefore (0.019L/15.635L)= 1.216%=12.16 ppm. Therefore, every 5 seconds, approximately 12.16 ppm of CO2 are added to the current concentration in source cells. The rate of diffusion is then controlled explicitly by the delay between each averaging event.

The Cell-DEVS space of the CO2 model is: CO2 = < Xlist, Ylist, S, X, Y, η, N, {t1, t2}, C, B, Z >, where Xlist = Ylist = {Ø}; S = type: {0, 1, 2, 3, 4, 5} and conc: {double}; X = Y = Ø; η = 5; N = {(0,0), (-1, 0), (0, -1), (0, 1), (1, 0)}; t1 = 14; t2 = 20; $C = \{C_{ij} / i \in [0,14] \wedge j \in [0,20]\}$; and B = {Ø} (unwrapped cell space). We use the Von Neumann neighborhood. The cells have 2 distinct state variables: type and concentration. Type is 0 for open-air, 1 for CO2 sources, 2 for walls/impermeable objects, 3 for open doors, 4 for open windows, and 5 for ventilation. Concentration represents the CO2 in ppm within a cell. Type 0 (open-air) and type 1 cells (CO2 sources) are the only cells that undergo concentration changes due to diffusion. Type 2 cells (walls/impermeable objects) are excluded from the averaging calculation. We assume that source cells (type 1) are always separated from walls by at least 1 cell buffer, which is a reasonable assumption since it is not common for a person to be breathing directly against a wall.

The simulation configuration defines the main Cell-DEVS CO2 model parameters such as the floor plan dimensions, the cell's neighborhood type and the different type of cells. To add new type of cell, the following parameters need to be defined under colors key value in the configuration input file:

```
"<color_name>": {
       "name" : "<cell_name>",
       "parent_cell" : "<parent_cell>",
       "alpha" : <transparency_alpha_value>,
       "bottom" : <cell_bottom>,
       "top" : <cell_top>,
       "concentration" : <CO2_concentration>,
       "type" : <cell_type>,
       "counter" : <counter_to_be_CO2_source>},
```

In addition to the current default type of cells, we defined a new cell to simulate ducts between indoor rooms. The different types of cells available in the configuration file are listed in Table 1. Any cell is part of other cells which we call a parent cell. For instance, a window cell is located within walls, so its parent cell is the wall. The parent cell is used to extend the cell's 3d structure beyond its bottom and top. The alpha value is used to define the transparency of the cell for visualization purposes. The size of a cell in the configuration corresponds to 25 $cm^3$ in the real world.

The construction of 2D or 3D Cell-DEVS CO2 model scenarios is simply defined by updating the dimensions in the configuration. A 3D model can be defined by setting the height value to greater than one.

TABLE I. TYPE OF CELLS IN CELL-DEVS CO2 MODEL

| Color | name | Parent | Alpha | Bottm | Top | C |
|---|---|---|---|---|---|---|
| Cyan | co2_source | White | 100 | 0 | 6 | 0 |
| gray | wall | Gray | 100 | 0 | 12 | 0 |
| green | door | Gray | 90 | 0 | 8 | 500 |
| yellow | window | gray | 90 | 4 | 7 | 400 |
| Blue | vent | white | 100 | 10 | 11 | 300 |
| Red | workstation | white | 100 | 0 | 4 | 500 |
| Fuchsia | duct | gray | 90 | 8 | 10 | 500 |

## IV. BUILDING 3D MODELS FOR CO2 SPREAD

In this section, we simulate and discuss the results of simulating an indoor structure with the presence of a static number of occupants but different setup scenarios of ducts between rooms.

Table 2 shows the static and variant parameters of the simulation setup. The idea is to change the number and location of ducts while keeping other parameters static and measure the diffusion and concentration of CO2 in every room. Six cells are selected to measure and compare the concentration of CO2 in every room with occupants. The idea is to show that the efficient distribution of ducts between rooms can effectively balance the level of CO2 indoor, which reduces the harmful effect on occupants resulted from exposure to high CO2 levels.

TABLE II. SIMULATION PARAMETERS

| Parameter | Value | Variant | Location Change |
|---|---|---|---|
| Dimensions | 25x30x12 | No | No |
| Rooms | 7 | No | No |
| Occupants | 6 | No | Yes |
| Occ. Movement | random | Yes | Yes |
| Ducts | 5 | Yes | Yes |
| Vents | 6 | No | No |
| Windows | 3 | No | No |

The initial simulation 2D and 3D setup are shown in Figures 4 and 5. In this scenario, three rooms have ducts, while the other three rooms have no ducts. Each room has one occupant and one vent. The central area has no occupants and has two vents.

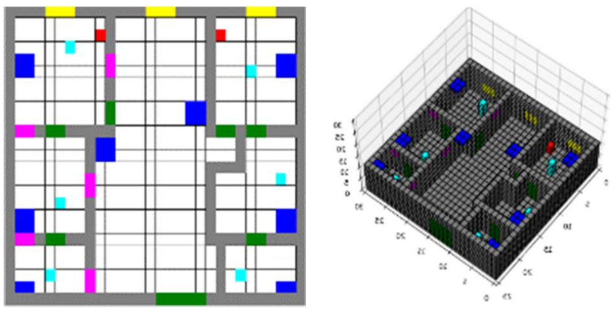

Fig. 4. Base Building Simulation Setup

The results in Figure 5a shows that the CO2 concentration in rooms with ducts is less than in those rooms without ducts. The presence of ducts between rooms assists the ventilation system in reducing the CO2 level inside these rooms. The results in Figure 5b and 5c show that cells in room 5 with ducts reaches a maximum CO2 concentration of nearly 600ppm while in room 2 without ducts reaches a maximum of nearly 1000 ppm.

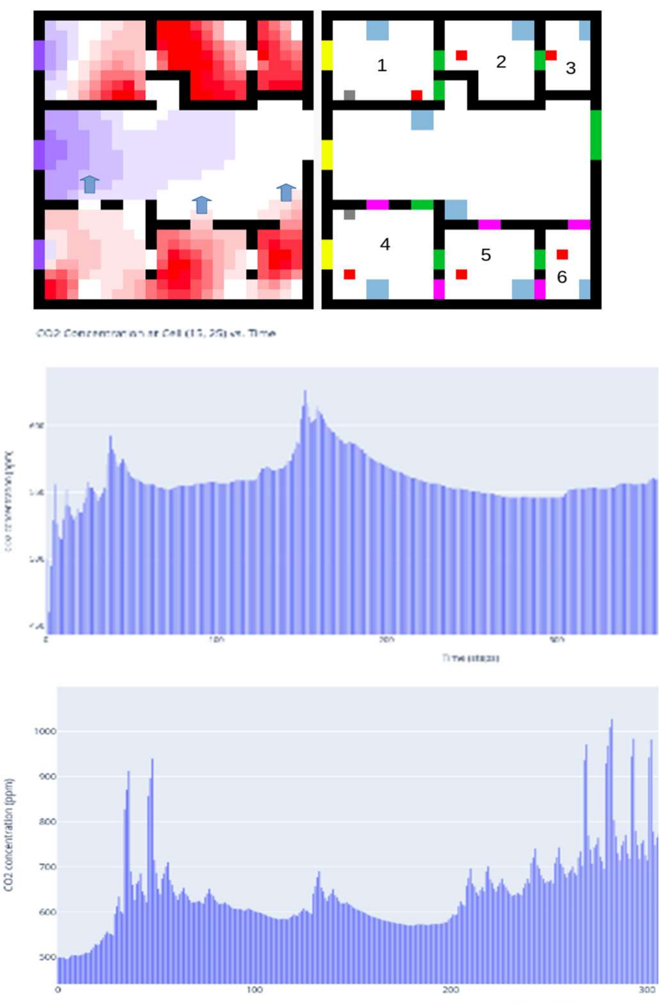


Fig. 5. Simulation results: (a) CO2 spread (red) in different rooms (numbered 1-6); (b) CO2 concentration in Room 5; (c) Room 2.

When we run the setup shown in Figure 6, we can see we modified the building configuration and all six rooms have no ducts. Each room has one occupant and one vent. The central area has no occupants and two vents.

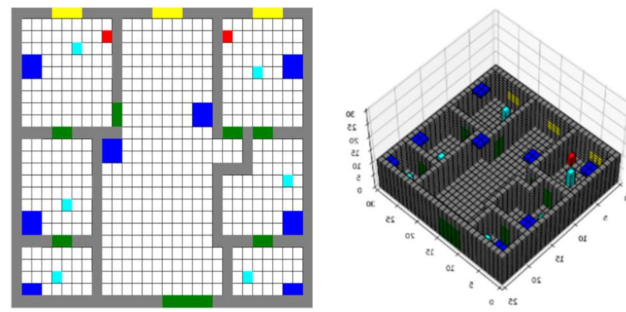

Fig. 6. Building configuration: no ducts.

The results in Figure 7 show that the CO2 level gradually increases in all rooms with occupants. Although the ventilation system is the same as in scenario one, the CO2 level in this scenario is higher. The absence of ducts between rooms increases the concentration of CO2 and reduces the efficiency of the ventilation system due to the imbalance distribution of CO2 in the apartment. The results show that cells in both rooms 5 and 2 can reach a maximum CO2 concentration of nearly 1000ppm.

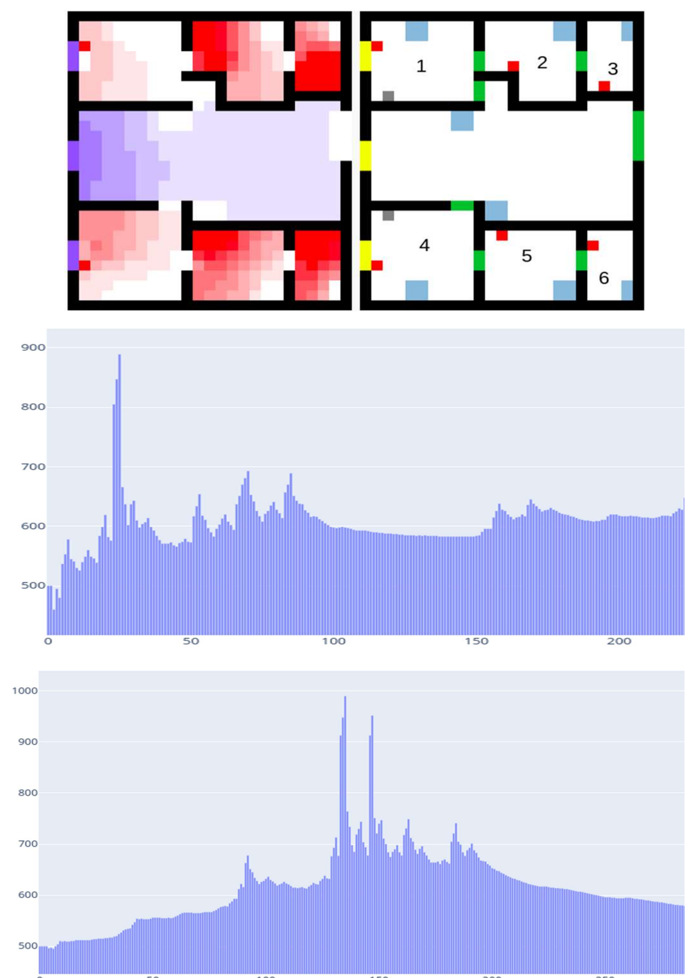


Fig. 7. Simulation results: (a) CO2 spread (red) in different rooms (numbered 1-6); (b) CO2 concentration in Room 5; (c) Room 2.

In the scenario presented in Figure 8, all six rooms have ducts. Each room has one occupant and one vent. The results in Figure 9 show that the CO2 level significantly reduced in all rooms with occupants. The presence of ducts between all rooms reduces the concentration of CO2 and assists in increasing the efficiency of the ventilation system due to the balanced distribution of CO2 inside the apartment.

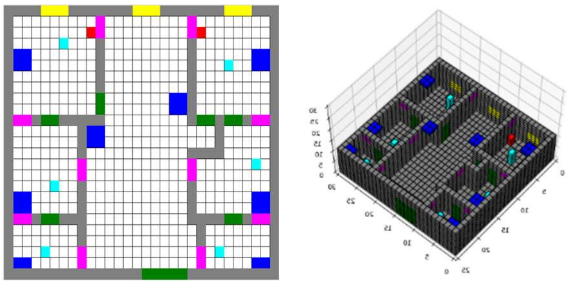

Fig. 8. Building configuration: all rooms with ducts.

We can see that in both rooms 5 and 2 we can reach a maximum CO2 concentration of nearly 600ppm which is lower than the CO2 level earlier.

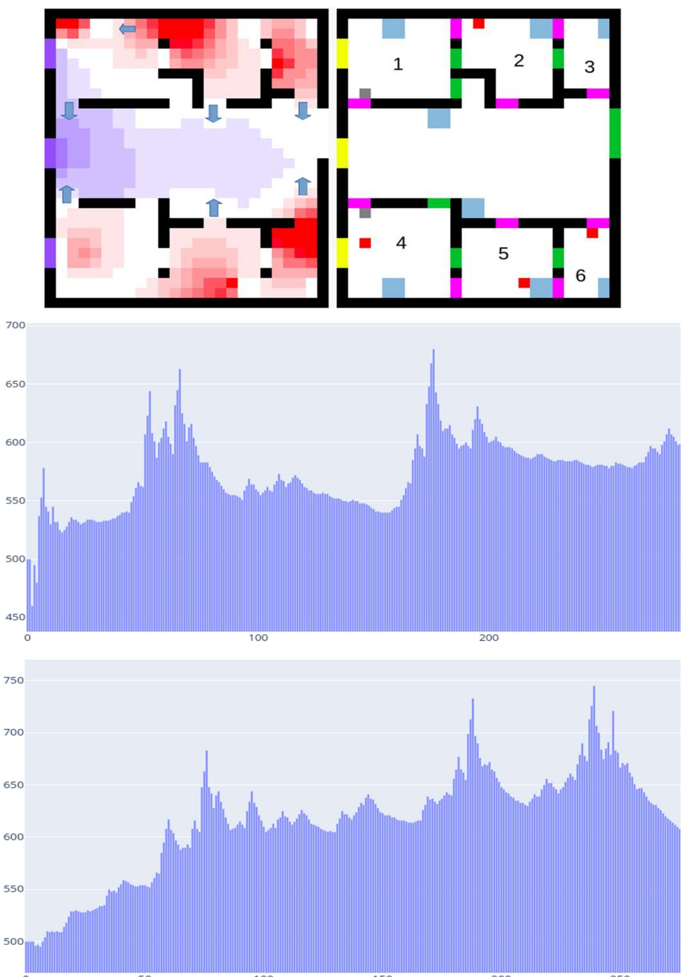


Fig. 9. Simulation results: (a) CO2 spread (red) in different rooms (numbered 1-6); (b) CO2 concentration in Room 5; (c) Room 2.

The setup shown in Figure 10 uses ducts that are also connected to the outside of the building. Each room has one occupant and one vent.

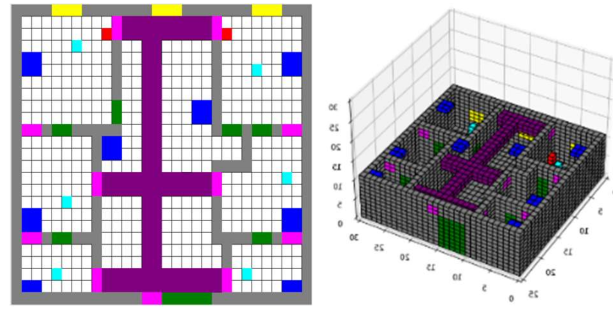

Fig. 10. Building configuration: all rooms with ducts connected outside.

As we can see in Figure 11, the CO2 level significantly reduced in all rooms with occupants. The presence of ducts between all rooms reduces the concentration of CO2 and assists in increasing the efficiency of the ventilation system due to the balanced distribution of CO2 inside the apartment.

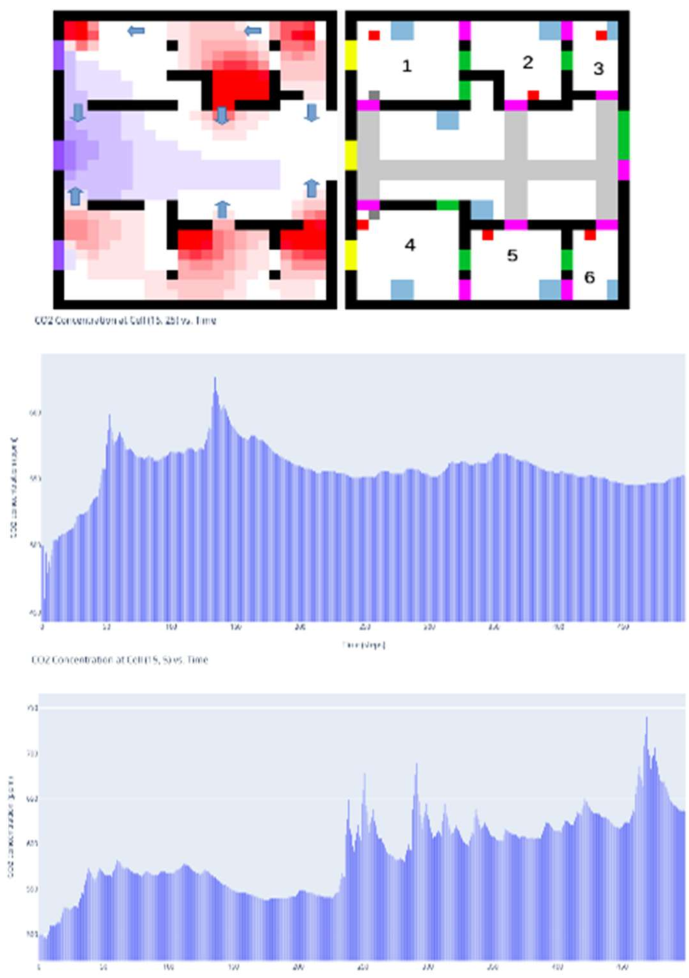


Fig. 11. Simulation results: (a) CO2 spread (red) in different rooms (numbered 1-6); (b) CO2 concentration in Room 5; (c) Room 2.

## V. CONCLUSIONS

Motivated by the need for studying the effects of room 7configuration on recorded $CO_2$ concentration [15] and the way $CO_2$ diffuses in closed areas, we developed a generic Cell-DEVS model that accepts different room configurations as input parameters. The model can help to study the effect of placing sensors in different locations, the effect of changing the ventilation, increasing the number of occupants, changing the furniture layout, and many other configuration parameters. Incorporating airflow in the room is a future feature that we are planning to add to the model. However, the model at the current state has successfully mimicked the physical system. The results show that implementing ducts between rooms and with the presence of vents can efficiently reduce the concentration of CO2 inside rooms.

## REFERENCES


[1] Wainer, G. Discrete-event modeling and simulation: a practitioner's approach. CRC Press (2009).

[2] Khalil, H., Wainer, G., Dunnigan, Z.: Cell-DEVS models for CO2 sensors locations in closed spaces. In: Bae K.-H., et al. (eds.) 2020 Winter Simulation Conference (WSC), virtual. 19.

[3] Al Horr, Y., Arif, M., Katafygiotoua, M., Mazroei, A., Kaushik, A., Elsarrag, E.: Impact of indoor environmental quality on occupant well-being and comfort: a review of the literature. International Journal of Sustainable Built Environment 5(1), 1–11 (2016).

[4] Jiang, A., Masooda, M.K., Soh, Y.C., Li, H.: Indoor occupancy estimation from carbon dioxide concentration. Energy and Buildings 131, 132-141 (2016).

[5] Rudnick, S. N. and Milton, D. K. Risk of indoor airborne infection transmission estimated from carbon dioxide concentration. Intl. J. of Indoor Environment and Health, 1, 3 (2003), 237-245.

[6] Peng, Z. and Jimenez, J. L. Exhaled CO2 as COVID-19 infection risk proxy for different indoor environments and activities, medRxiv. (2020). 15.

[7] Pantazaras, A., Lee, S.E., Santamouris, M., Yang, J.: Predicting the CO2 levels in buildings using deterministic and identified models. Energy and Buildings 127, 774–785 (2016).

[8] Makmul, J.: Microscopic and macroscopic for pedestrian crowds. Dissertation, Mannheim University (2016)

[9] Batog, P. and Badura, M. Dynamic of changes in carbon dioxide concentration in bedrooms. Procedia Engineering 57, (2013), 175–182.

[10] Yalçın, N., Balta, D., Özmen, A. A modeling and simulation study about CO2 amount with web-based indoor air quality monitoring. Turkish J. of El. Eng. and Comp. Sci. 26, (2018), 1390-1402.

[11] Khalil, H. and Wainer, G., Modeling Carbon Dioxide Dispersion Indoors: A Cell-DEVS Experiments. Proc. ACRI 2020. Lecture Notes in Computer Science, Vol 12599.

[12] Khalil, H., and Wainer, G. CD2: An Automation Tool for Cell-DEVS CO2 Diffusion Models. Proc. SimAud2021.

[13] ASHRAE Standard 62.1 Ventilation for Acceptable Indoor Air Quality, http://www.myiaire.com/product-docs/ultraDRY/ASHRAE62.1.pdf (2013).

[14] Zuraimi, M.S., Pantazaras, A., et al. Predicting occupancy counts using physical and statistical CO2-based modeling methodologies. Building and Environment 123, 517-528 (2017).

[15] Arief-Ang, I.B., Hamilton, M., Salim, F.D.: RUP: Large room utilization prediction with carbon dioxide sensor. Pervasive and Mobile Computing 46, 49–72 (2018