MODELING CO$_2$ SPREAD IN INDOOR ENVIRONMENTS

Marina Murillo[1] and Gabriel Wainer[1]

[1]Dept. of Systems and Computer Eng., Carleton University, Ottawa, ON, CANADA

ABSTRACT

We introduce a hybrid modeling approach for simulating carbon dioxide (CO2) dispersion in indoor environments by integrating Cellular Automata (CA), Discrete Event System Specification (DEVS), and agent-based modeling. The proposed framework enhances traditional models by incorporating dynamic CO2 generators, random-walk algorithms, and CO2 sinks. We show how the method can be used to examine the effects that room layouts, occupant movement, ventilation settings, and CO2 sinks and sources placement have on indoor concentration patterns. The approach presented here enables the exploration of various configuration parameters and provides a flexible and scalable tool for understanding CO2 diffusion.

1 INTRODUCTION

Recently, maintaining healthy indoor environments while ensuring energy efficiency has become a central concern. Carbon dioxide (CO$_2$) concentration is widely recognized as a key indicator of indoor air quality, with significant implications for occupant comfort, cognitive performance, and the implementation of demand-driven ventilation and occupancy detection systems (Mendell et al. 2024). Elevated CO$_2$ levels have been associated with reduced academic performance, productivity, respiratory health issues (Sadrizadeh et al. 2022; Deng et al. 2024). However, conducting physical experiments to assess CO$_2$ dispersion across diverse indoor scenarios is often impractical due to cost, time, and logistical constraints.

To address these challenges, Modeling and Simulation (M&S) techniques have emerged as powerful alternatives, enabling researchers to evaluate air quality under controlled virtual conditions. Existing approaches include statistical-based for occupancy estimation, as well as physics-based methods such as Computational Fluid Dynamics (CFD) for simulating airflow and CO$_2$ distribution. While effective, these techniques are often limited by high computational complexity or detailed input data and calibration (Labeodan et al. 2015; Mou et al. 2022). Other methods, in particular, the hybrid combination of Cellular Automata (CA) and Discrete Event System Specification (DEVS), known as Cell-DEVS formalism, has been successfully applied to incorporate spatial and temporal dynamics, including room geometry, HVAC systems, window and door locations, and occupant presence (Khalil et al. 2020; Khalil and Wainer 2024). These models are valuable for applications such as sensor placement optimization, air quality monitoring, and infection risk assessment. Despite their versatility, most current models are limited to two-dimensional representations, restricting their applicability in scenarios where the three-dimensional airflow interactions are significant.

To overcome these limitations, this study proposes a hybrid agent-based modeling approach that combines CA and DEVS to simulate CO$_2$ dispersion in three-dimensional indoor environments. The goal of this work is to enhance spatial realism and better capture the dynamic behaviors of both occupants and airflow patterns. Our approach incorporates dynamic CO$_2$ generators, random-walk algorithms for dispersion modeling, and the inclusion of CO$_2$ sinks to assess their impact on CO$_2$ reduction. The main contributions of this work are: i) the development of a 3D model to simulate CO$_2$ spread under dynamic occupancy behavior; ii) the evaluation of different CO$_2$ sinks to assess their impact on air quality; and iii) a set of simulation experiments that validate the proposed approach and highlight its practical implications for building design and environmental control strategies.

2   RELATED WORK

Modeling CO_2 dispersion in enclosed spaces is critical for addressing global warming, enhancing indoor air quality, optimizing sensor placement, and reducing energy use. Various computational methods have emerged, such as Labeodan et al. (2015), who noted challenges in obtaining reliable data for occupancy detection due to environmental variability and human behavior. Calì et al. (2015) developed an algorithm based on CO_2 levels to estimate occupancy in various settings, although it depends on knowledge of ventilation and outdoor CO_2. Ryu and Moon (2016) leveraged a statistical decision tree and Hidden Markov Model, finding that CO_2 measurements, particularly the indoor-to-outdoor concentration ratio, are key indicators of occupancy. Batog and Badura (2013) created a simplified bedroom model, focusing on key surfaces like the bed and wardrobe, and simulated CO_2 from a sleeping occupant over eight hours in two scenarios: one without fresh air inlets and another with airflow gaps. Their findings highlights the significance of proper CO_2 sensor placement for accurate readings. Similarly, Pantazaras et al. (2016) developed a method for predictive modeling of CO_2 levels in specific indoor spaces by integrating room characteristics, ventilation rates, and occupancy patterns, emphasizing the need for careful sensor placement to enhance model accuracy.

Recent studies have investigated Computational Fluid Dynamics (CFD) modeling for indoor environments. For instance, Mou et al. (2022) used CFD to simulate airflow and CO_2 distribution in a seminar room to optimize sensor placement, revealing that airflow patterns and occupant-generated CO_2 significantly affect concentration distributions. They stressed the importance of considering room layout, ventilation design, and occupant activity in simulations. However, while CFD offers high accuracy, it can be computationally demanding for large or complex environments and requires detailed boundary conditions and expert calibration (Zhang et al. 2024). Consequently, researchers are seeking alternative modeling techniques that provide a better balance of accuracy, efficiency, and scalability.

Cellular Automata (CA) models offer another promising approach by discretizing space into a grid of cells, where global system behavior emerges from simple local interactions and transition rules (Khalil and Wainer 2024). CA-based methods provide advantages in terms of computational efficiency, scalability, and the ability to model spatially heterogeneous environments, making them highly suitable for indoor CO_2 spread simulation. In this research we aim to model the spread of CO_2 using a hybrid approach that combines Discrete-Event models specified using the DEVS formalism (Zeigler et al. 2000) and CA (Wolfram 1984; Wolfram 1983). We propose an agent-based modeling framework in a spatial environment, incorporating a stochastic algorithm to simulate the dispersion of CO_2 particles as well as random walk.

CA is a method for modeling cell spaces using a regular n-dimensional grid, where each cell holds one of a finite set of possible states. The system evolves in *discrete time* steps, with all cells updating their states simultaneously according to a local transition rule. This rule determines the next state of a cell based on its current value and the values of its neighboring cells (the neighborhood). Figure 1(a) shows a two-dimensional CA, where each cell either contains a value or is empty, and its next state is computed based on its current state and those of the adjacent cells (shaded area).

CA allows for detailed mathematical analysis and can effectively model complex systems through simple computations in individual cells. While CA typically uses discrete-time updates suitable for asynchronous applications like pedestrian flow and traffic, it often employs synchronous rules, complicating the representation of time-triggered activities (Wang et al. 2023; Deng et al. 2022). Recently, combining CA with other modeling techniques for simulating CO_2 dispersion has gained interest. For example, Sonnenschein et al. (2025) proposed a hybrid model integrating Land Use Regression (LUR) for baseline CO_2 concentrations with CA to assess urban traffic-related CO_2 pollution by incorporating meteorological and morphological factors. However, research applying CA for indoor CO_2 dynamics remains limited. Integrating CA with DEVS formalism and other paradigms could provide a more flexible approach for simulating CO_2 spread, enhancing temporal precision, modularity, and asynchronous modeling capabilities. Cell-DEVS is a hybrid combination of both DEVS and CA.
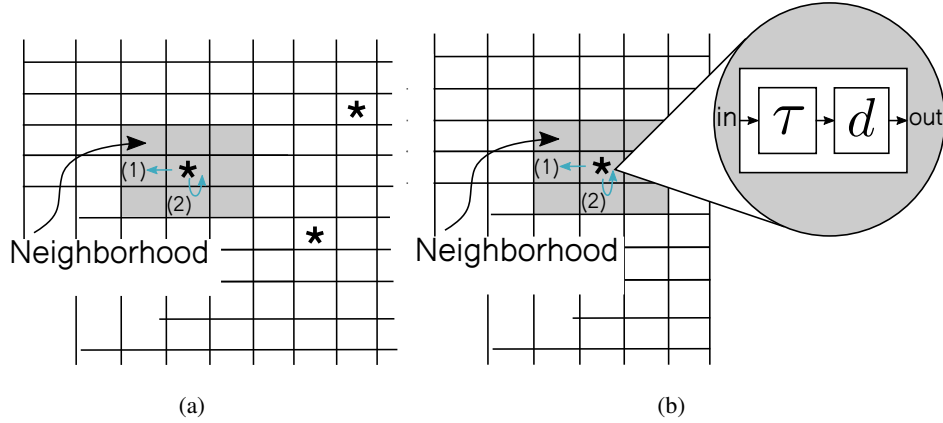
Figure 1: Two-dimensional cellular automaton with its neighborhood.

In Cell-DEVS, each agent in a cell is modeled using a hybrid combination of execution rules defined later as a DEVS atomic model. A structure coupling mechanism is used to interconnect the cells. Indeed, a Cell-DEVS model is an n-dimensional lattice where each cell is an atomic model, and the whole-cell space is a coupled DEVS model (Wainer 2009). These cells communicate with each other through input and output ports, allowing them to exchange information not only with neighboring cells but also with other models defined in different formalisms defined outside the defined cell space. Figure 1(b) provides an illustrative representation that highlights how Cell-DEVS integrates principles from both DEVS and CA.

To define a Cell-DEVS model, the first step is to define the atomic behavior of each cell a:

$$TDC = <X, Y, S, N, type, d, \delta_{int}, \delta_{ext}, \tau, \lambda, D>,$$

where X and Y are the sets of external input and output events, respectively, S is the set of possible states, N is the set of input values, $type$ specifies the type of delay (transport, inertial, or other), and d denotes the delay duration. The local computing function τ determines the cell's future state, while the output function λ generates the corresponding output. Finally, D defines how long the state is held before a transition occurs. When using a transport delay, the output is sent after the delay time d has passed. In contrast, an inertial delay is used as a preemptive mechanism; it prevents any scheduled change from taking place upon receiving an external event from a neighbor cell before the scheduled time. Once the atomic behavior of each cell is specified, the next step is to define the complete cell space.

In Cell-DEVS, this space is modeled as a coupled DEVS model, formally represented as follows:

$$GCC = <X_{list}, Y_{list}, I, X, Y, \eta, \{t_1, \cdots, t_n\}, N, C, B, Z>,$$

where X_{list} and Y_{list} denote the list of external input and output couplings, respectively; I is the set of states, X and Y are the sets of external input and output events, respectively; $\eta \in N$ defines the neighborhood size; and $\{t_1, \cdots, t_n\}$ is the number of cells in each dimension of the lattice. The cell space itself is denoted by C, with N being the neighboring cells. B represents the set of border cells, which can have different behavior than the interior cells or can be configured with wrap-around boundaries. The function Z is the translation function that maps the output port of a cell to the input port of another cell.

Random walk algorithms have shown to be useful to include stochastic behavior in the movement of particles within indoor environments. These algorithms simulate the trajectories of individual particles based on probabilistic rules, effectively capturing the effects of turbulent diffusion and airflow variability (Wang and Mu 2011; Marashian et al. 2023). In this context, a hybrid model integrating random walk algorithms with agents defined using CA and DEVS modeling and simulation could offer a more comprehensive and flexible modeling framework for simulating CO_2 dynamics in realistic indoor scenarios.

3 MODELING THE SPREAD OF CO₂ PARTICLES

In this section we present various CO_2 spread models that combine a cellular model with a discrete-event specification. Khalil et al. (2020) introduced such hybrid models to study CO_2 dispersion in a room and to identify the optimal placement and detection latency of CO_2 sensors, a complex problem in building design. A closed space was modeled as a set of neighboring cells representing the 2D layout of a room. The formal specification of the two-dimensional Cell-DEVS CO_2 model presented in (Khalil et al. 2020) is:

$$CO_2 = < X_{list}, Y_{list}, S, X, Y, \eta, N, \{t1, t2\}, C, B, Z >,$$

where $X_{list} = Ylist = \{\varnothing\}$; $S = \texttt{type} : \{0,1,2,3,4,5\}$ and $\texttt{conc} : \{\text{double}\}$; $X = Y = \varnothing$; $\eta = 5$; $N = \{(0,0),(-1,0),(0,-1),(0,1),(1,0)\}$; $t1 = 14$; $t2 = 20$; $C = \{C_{ij} | i \in [0,14] \wedge j \in [0,23]\}$; and $B = \{\varnothing\}$ (unwrapped cell space); Z is the translation function that defines the internal and external function. The local computing function τ of the atomic model of each cell and the duration function D are shown in Table 1 and Table 2, respectively. The neighborhood is a Von Neumann and only the North (N), East

<table>
<tr><td colspan="2" align="center">Table 1: Values of $\tau(N)$.</td><td colspan="2" align="center">Table 2: Values of $D(s)$.</td></tr>
<tr><td>$\tau(N)$</td><td>N</td><td>$D(S)$</td><td>S</td></tr>
<tr><td>conc = average of neighbors</td><td>type = 0</td><td>R0 += 1,000</td><td>type = 0</td></tr>
<tr><td>conc = neighbors average + 12.16 ppm</td><td>type = 1</td><td>R0 += 5,000</td><td>type = 1</td></tr>
<tr><td>conc = - 10 ppm</td><td>type = 2</td><td>R0 += 1,000</td><td>type = 2</td></tr>
<tr><td>conc = 500 ppm</td><td>type = 3</td><td>R0 += 1,000</td><td>type = 3</td></tr>
<tr><td>conc = 400 ppm</td><td>type = 4</td><td>R0 += 1,000</td><td>type = 4</td></tr>
<tr><td>conc = 300 ppm</td><td>type = 5</td><td>R0 += 1,000</td><td>type = 5</td></tr>
</table>

(E), West (W), and South (S) neighbors are considered. This model was implemented using the CD++ tool. Each area could use one of six model types, representing CO_2 levels in particles per million (ppm), based on gas diffusion rules: open-air space (constant 500 ppm); CO_2 sources (fixed emissions); walls (no diffusion); open doors, windows, and vents (diffusers with baseline CO_2 concentrations of 500, 400, and <300 ppm, respectively). Each cell in the cell space was defined by its model type. Diffuser cells were unaffected by surrounding CO_2 levels. CO_2 sources were modeled as an occupant at rest, with an average breathing interval that emitted CO_2 every 5 seconds and a diffusion interval set at 1.

We validated this model with real-world data and showed that the model effectively represented CO_2 diffusion, dissipation and latency. Based on these results, we modified the static generator (a DEVS model that represented a stationary occupant) as well as CO_2 behavior after the generator left the room. We modeled scheduled emissions and effects of movement (such as a person walking) on CO_2 distribution.

To address these limitations, we introduced a dynamic CO_2 generator to the original Cell-DEVS model. Each cell contained the `concentration`, `type`, a `counter`, and a `direction`. The rules for this model are defined as follows:

```
% rules for CO2 generator
rule : {counter += 1;} 1000 {if cell type 1 AND counter != 2 OR 5}
rule : {type = 0; add 12.16ppm of CO2; counter += 1;} 1000 {if cell type 1 AND counter = 2}
rule : {add 12.16ppm of CO2; counter = 0;} 1000 {if cell type 1 AND counter = 5}

% rules when open cells are affected by CO2 neighbors
rule: {type = CO2; counter = neighbors counter; direction = randInt(100) or neighbor
       direction; (depends if scheduled or random movement)} 1000 {if cell type 0 AND
       neighbors(counter = 2 AND type = 1 AND direction is correct)} (4 rules)

% set of rules if type 0 for CO2 dissipation
rule: {calculate average concentration of all surrounding cells} 1000 {if cell type
```

```
        open air}

% set of rules for the remaining cases
rule: {maintain preset CO2 concentration;} 1000 {if cell of other types}

rule: {counter += 1;} 1000 {any type} % default rule
```

These rules included logic to prevent generator from passing through walls, regulate CO_2 emission timing, handle transitions between cell types, and model diffusion in open-air spaces.

We tested the dynamic CO_2 generator with two types of scenarios: i) randomly, to model organic movement and distribution, and ii) a scheduled path representing a person arriving and visiting specific rooms in sequence. The environment represents a bachelor apartment layout with a door and vents in the kitchen and bathroom. To simulate active breathing, CO_2 emissions were increased every 3 seconds. The left panel in Figure 2 shows CO_2 concentration, while the right panel shows the generator's position. We can see the results from the random scenario starting at the entrance (green door). The generator remains
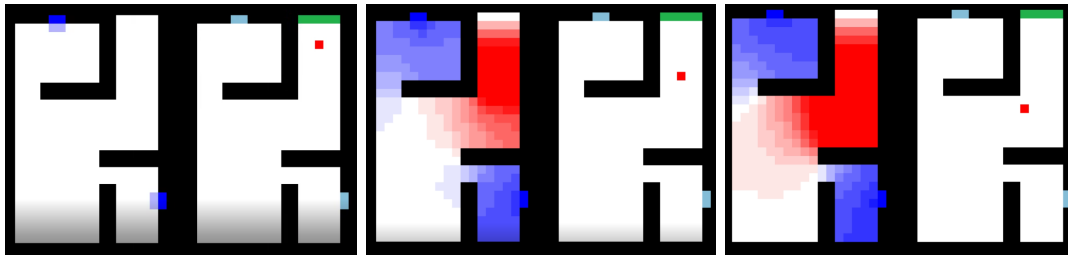

Figure 2: Random CO_2 generation. Start at the entrance at 0 (left), 5 (middle), and 10 (right) minutes.

near its starting point, saturating that area with CO_2 leaving other regions mostly unaffected (similar to the static scenario) as we used a uniform distribution for movement direction, which resulted in a tendency for circular paths. Next, we run a scheduled scenario representing a person entering at home, leaving the door open, staying around 1 minute and 30 seconds in the living room, 2 minutes in the kitchen, and finishing in the bathroom. Figure 3 illustrates CO_2 distribution over the course of this planned movement. As expected,
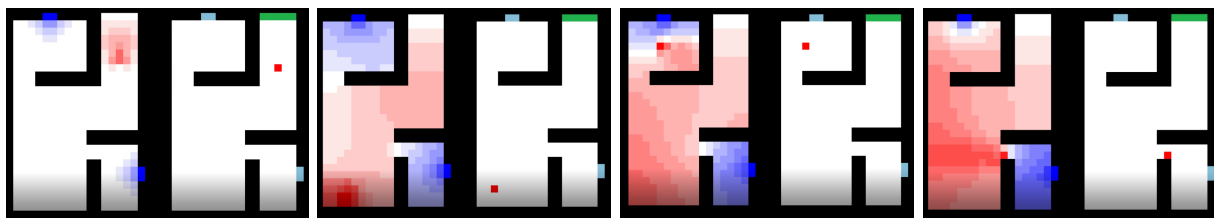

Figure 3: Scheduled CO_2 generation for times 0, 1.45, 3.45, and 4 minutes (left to right).

CO_2 spread across the apartment. Notably, a significant concentration remained in the entrance hallway, even with the door open. This highlights both the rapid spread of CO_2 and its slow dissipation, even in well-ventilated conditions.

3.1 Effect of Plants and CO_2 Sinks on Indoor CO_2 Levels

The model above was further extended to investigate CO_2 level variations in a closed indoor environment, where moving occupants act as CO_2 sources. We evaluate the effectiveness of adding plants and CO_2 sinks placed in various positions throughout the room. Each cell now considers the `type` and `concentration` of CO_2. Walls are modeled as impermeable structures that block CO_2 diffusion and maintain constant

values (barrier). Doors are initialized with a typical indoor baseline CO_2 level (500 ppm), while windows maintain lower concentration to simulate the influence of outdoor air. Ventilation cells actively reduce CO_2 concentration to simulate mechanical air extraction. Workstations behave as passive air cells, averaging the CO_2 levels of their non-wall neighbors. Simulations were carried out using the Cell-DEVS formalism and Cadmium tool.

A code snippet used to compute concentration values for plants, CO_2 sinks, and sources is shown below:

```
switch(state.current_state.type){                    new_state.concentration =
  case PLANT: { ...                                   concentration / num_neighbors - 10;
    if(neighbors.type != IMPERMEABLE){
      concentration +=                               case CO2_SOURCE: { ...
          neighbors.concentration;                     if(neighbors.type != IMPERMEABLE){
      num_neighbors++; }                                 concentration +=
    new_state.concentration =                               neighbors.concentration;
     concentration / num_neighbors -                     num_neighbors++; }
        (plant_decrease / 480); break; }             new_state.concentration =
  case SCRUBBER: { ...                                 (concentration / num_neighbors) +
    if(neighbors.type != IMPERMEABLE){                    concentration_increase;
      concentration +=                                 ...
          neighbors.concentration;
      num_neighbors++; }
```

CO_2 sources simulate human breathing by adding a fixed amount of CO_2 to their cell every ten minutes. Plants absorb a small portion of CO_2, decreasing the local concentration slightly, whereas CO_2 sinks remove a larger amount, simulating active filtration mechanisms.

Figure 4 illustrates the results for a scenario in this model. The simulation, run for 1440 minutes, examines how CO_2 levels are affected by occupants, plants, and sinks in a closed space. The initial condition, shown in Figure 4(a), features all occupants outside the room and active ventilation. Under these conditions, CO_2 remains below 500 ppm (blue cells), aided by sinks and plants absorbing CO_2. Figure 4(b)

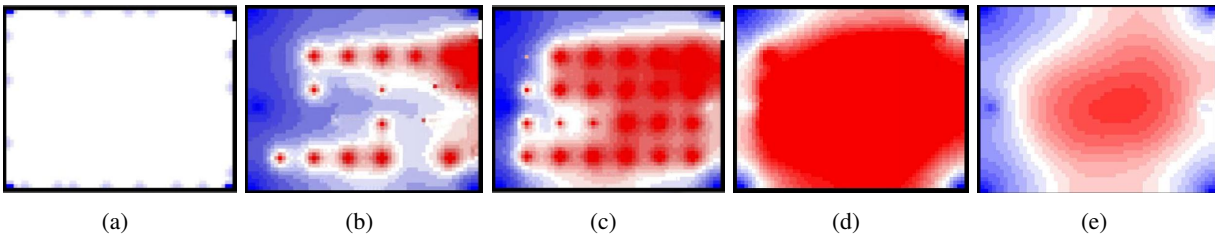| (a) | (b) | (c) | (d) | (e) |

Figure 4: Indoor CO_2 concentration with moving occupants, plants and CO_2 sinks.

depicts thirteen occupants entering the room, shown as red cells moving toward assigned workstations (gray cells). As they move, expanding red zones in the left panel indicate rising CO_2 levels, with the size of these zones varying to reflect individual breathing rates, which were randomly assigned. Over time, CO_2 concentration steadily increases due to ongoing respiration. Figure 4(c) shows all occupants at their workstations. The red regions have expanded compared to (b), indicating increased CO_2 levels, especially from occupants with higher breathing rates. As more people enter, overall CO_2 rises, but ventilation units and sinks (blue cells) help circulate air and reduce CO_2, especially near the edges where devices are present. The central area has higher CO_2 due to lack of vents, while plants along the walls have minimal impact.

Figure 4(d) shows ten occupants leaving, causing red CO_2 zones to shrink as vents, sinks, and plants absorb CO_2. The central area remains more concentrated with CO_2 than the edges, where changes are minimal, indicating that the plants had little impact on overall CO_2 levels. Figure 4(e) depicts the final scene with all occupants gone; fresh air (blue) expanded, and CO_2 (red) decreased as ventilation and sinks

removed residual CO_2. Plants, specifically Prayer plants, had minimal impact, absorbing about 215 ppm over 1440 minutes, compared to 18,158 ppm produced by a single person in the same period, showing their limited role in reducing CO_2 levels.

3.2 Effect of Distance between CO_2 Sources

This example investigates how varying the distance between CO_2 sources influences the overall indoor CO_2 concentration. Simulations were carried out using the Cell-DEVS formalism and CD++ tool. Different distances between CO_2 sources were tested to analyze how their separation affects diffusion patterns and concentration levels over time.

Each cell in the model represent an agent characterized by `type`, `counter`, and `concentration`. As before, the behavior of the model is defined by the τ function. The rules to generate cells simulating occupants are as follows:

```
{...}
% Moving person
rule : { ~c := $conc; ~ty := $type; } { $conc := ((121.6*2) + (((-1,0)~c + (0,-1)~c
        + (0,0)~c + (0,1)~c + (1,0)~c)/5)); $counter:= $counter + 1; } 5000
        { $type = -900 AND $counter < 125 }

rule : { ~c := $conc; ~ty := $type; } { $conc := (((-1,0)~c + (0,-1)~c + (0,0)~c
        + (0,1)~c + (1,0)~c)/5); $counter:= $counter + 1; $type:=-100; } 5000
        { $type = -900 AND $counter = 125 }
{...}
```

These dynamic source cells emit CO_2 for 125 update cycles to simulate active breathing, and then revert to standard air cells, simulating the departure of human activity.

Simulation results are shown in Figure 5, where magenta cells indicate the position of occupants actively emitting CO_2, red cells represent fixed CO_2 sources, white cells correspond to impermeable walls that block CO_2 diffusion, blue cells denote ventilation units maintaining ambient CO_2 levels at 300 ppm, and yellow cells depict the room's air duct system. At the beginning of the simulation, two individuals are placed
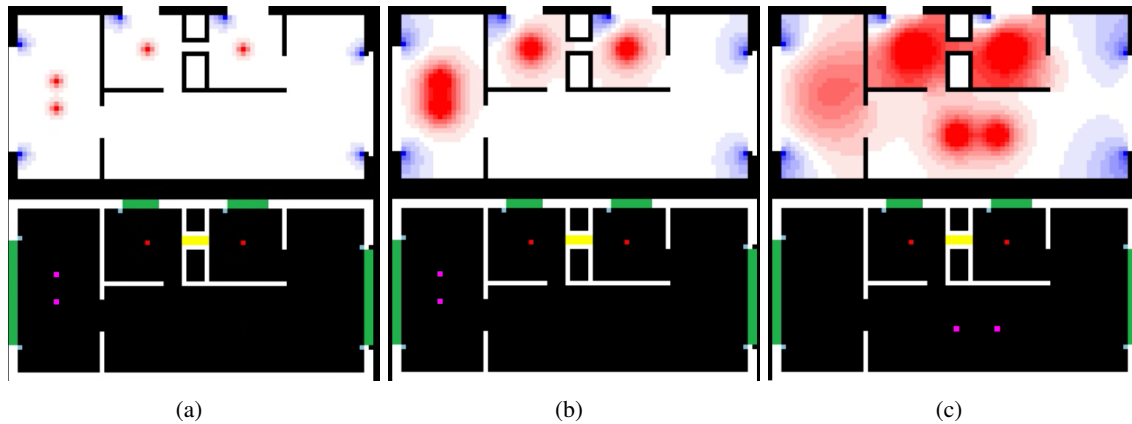

| (a) | (b) | (c) |

Figure 5: Effect of Distance on CO_2 sources.

1.25 meters apart (Figure 5(a)), and CO_2 concentration begins to rise around them (Figure 5(b)). After 125 breathing cycles, they move to the bottom room and are positioned 3.25 meters apart (Figure 5(c)). As a result, CO_2 concentration in the left room decreases due the absence of a source, while it begins to rise in the bottom room with the arrival of the new sources. The duct system enables CO_2 spread between the two upper rooms. CO_2 diffuses in all directions from the emitting cells, and the concentration

gradually increases over time. The simulation results indicate that when sources are close together, CO_2 levels rise more quickly. In contrast, when the distance between sources increases, the concentration builds up more slowly. These findings suggest that the distance between individuals influences how quickly CO_2 accumulates in indoor environments.

3.3 CO_2 Cell-DEVS Model with Random Walk Algorithm

In this section, we show how to combine cellular modeling, discrete-event simulation and random walk algorithms (in 2D and 3D environments). The idea is to model the indoor spread of CO_2 particles in a 3D environment. The cell space is used to build an agent based model in which each cell contains the behavior of the particles in a room, discretizing a continuous space using a rectangular grid. Each of the cells are updated using a continuous-time specification. In addition, a random walk algorithm is used to represent the movement of the CO_2 particles in a more realistic fashion using a stochastic model, because the randomness of the movements, makes the random walk algorithm a good approach to model the spread of air particles in 2D and 3D space. The three methods are combined and simulated for different indoor scenarios using the Cadmium tool.

To define the behavior of the agents on each cell, which is then reproduced in all the cells in the cell space, first we need to define a local transition function. This should take into consideration the random walk algorithm, where each object can move freely in all directions based on a stochastic rule. In Figure 6,
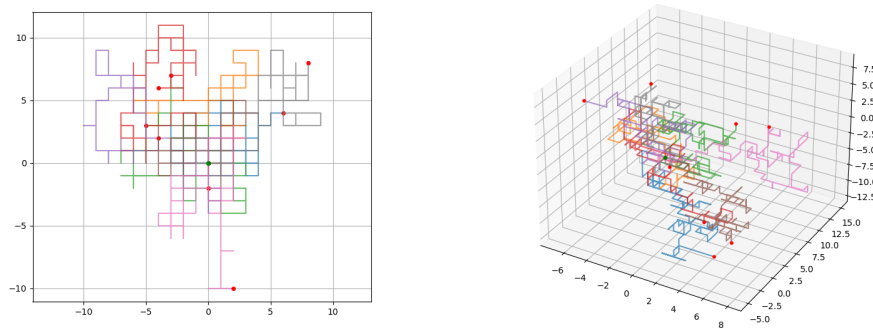

Figure 6: 2D and 3D Random Walk Models.

we show a 2D (left) and 3D (right) example of how the random walk works. In the former, the objects move up, down, left, and right at random; while in the latter the object can also move upward and downward. The probability of choosing the direction is varied and can be changed in each calculation, which is activated using a discrete-event simulation algorithm based on the DEVS formalism.

The behavior of the model is defined in τ function, which is used to define the behavior of the agent on each cell. In this case, the cells defined as sources increase the concentration of CO_2, while cells defined as vents, doors, and windows decrease it. The implementation of this behavior is as follows:

```
switch(state.current_state.type){
   case IMPERMEABLE:
     new_state.concentration = 0; break;        // Neighbor on right
   case DOOR:                                    else if((neighbors.first[0] - 1) ==
     new_state.concentration = base; break;       std::get<0>(currentLocation) &&
   case WINDOW:                                    neighbors.first[1] ==
     new_state.concentration =                        std::get<1>(currentLocation)){
         window_conc; break;                        concentration +=
   case VENTILATION:                                    neighbors.concentration +
     ...                                            floor(neighbors.concentration*
   case AIR:                                        distneg(rng)/500 );
     ...                                            num_neighbors++;
   case WORKSTATION:                                }
     ...                                        // Similar for Neighbor on bottom, left,
   case CO2_SOURCE:{ ...                        // itself and above
     if(neighbors.type != IMPERMEABLE){         else if((neighbors.first[2] - 1) ==
       // Neighbor on top                         std::get<2>(currentLocation)){
       if (neighbors.first[0] ==                   concentration +=
         std::get<0>(currentLocation) &&              neighbors.concentration +
         (neighbors.first[1] - 1) ==               floor(neighbors.concentration*
         std::get<1>(currentLocation)){           distneg(rng)/500 );
         concentration +=                         num_neighbors++;
             neighbors.concentration +          }
          floor(neighbors.concentration*      // behind
          distneg(rng)/500 );                   {...}
         num_neighbors++;
       }
     }
```

In this code, the τ function first checks the type of each cell. Cells classified as walls (IMPERMEABLE) are assigned a concentration of zero, while doors, windows, and ventilation cells are updated with predefined concentration values that represent air exchange characteristics. For CO_2 source cells, the concentration is increased by considering the contribution of neighboring cells. The CO_2 concentration of cells classified as air and workstation is updated in a similar way. To simulate the variability introduced by the random walk algorithm, a random number generator based on the Mersenne Twister algorithm was initialized using a non-deterministic seed to ensure variability across runs. A uniform integer distribution in the range [-70, 70] was used to produce evenly distributed random values within this interval.

In existing CO_2 models, particles move to the neighbor cells using a uniform method for spreading, as shown in Figure 7. The dark brown cell in the middle of the first image represents a CO_2 source that increase the concentration of the cell. In the following iterations, the CO_2 particles move to the neighbor cells and increase their CO_2 concentration.
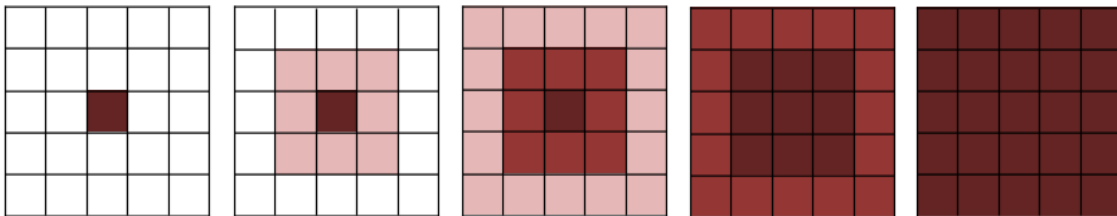

Figure 7: Deterministic spreading of CO_2 particles - Iterations #1(left) to #5 (right).

As we can see in Figure 8, when we combine the model with a random walk method, the use of a more realistic spread of CO_2 particles add randomness to the increase of CO_2 in the neighboring cells as it is done in a random way.
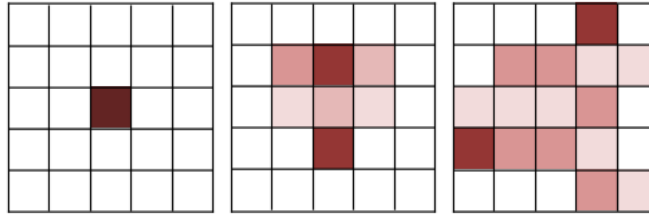
Figure 8: Random Walk spreading of CO_2 particles - Iterations #1(left) to #3 (right).

After defining the cell's behavior, we define specific parameters and a configuration of the cell space as well as of each cell. An extract of a JSON configuration file, where we can include different objects like doors, walls, windows, vent, and CO_2 sources, together with their specific parameters is shown below:

```
"scenario": {
  "shape": [ 43, 40, 8],
  "wrapped": false,
  "default_delay": "transport",
  "default_cell_type": "CO2_cell",
  "default_state": {
      "counter": -1,
      "concentration": 500,
      "type": -100
  },
  "default_config": {
      "CO2_cell": {
      "conc_increase": 143.2,
      "base": 500,
      "resp_time": 1,
      "window_conc": 400,
      "vent_conc": 300}
    },
  "neighborhood": [
      {"type": "von_neumann",
       "range": 1}]
},
"cells": [{}]
```

Figures 9 and 10 present simulation results comparing CO_2 dispersion patterns using a deterministic method (left) versus the proposed random walk approach (right), for both 2D and 3D building layouts. In both scenarios, the introduction of the random walk algorithm significantly alters the CO_2 dispersion dynamics. In Figure 9, the 2D simulations highlight how the deterministic method results in smoother and
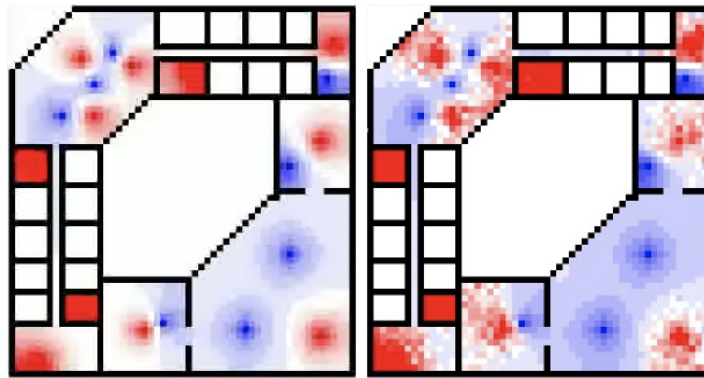

Figure 9: 2D CO_2 spread. Deterministic (left) and with Random Walk (right).

more uniform diffusion patterns centered around the CO_2 sources. In contrast, the random walk approach introduces localized variations and asymmetries, reflecting the stochastic nature of particle movement. Indeed, some regions near sources exhibit lower concentrations than surrounding cells, emphasizing the irregular dispersion introduced by random motion. Figure 10 shows similar behavior in a 3D environment. The deterministic simulation again yields a more structured and symmetric distribution, while the random
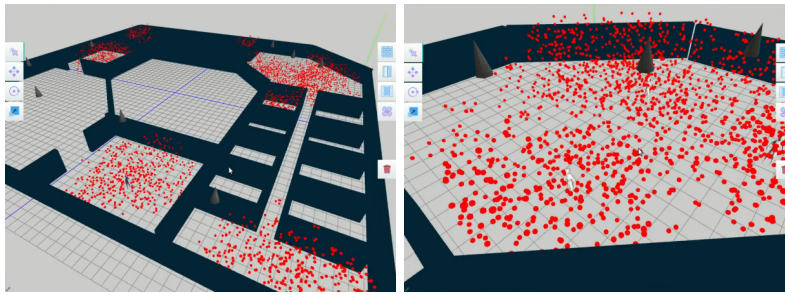
Figure 10: 3D CO_2 spread.

walk approach introduces more irregular and diffuse spread of CO_2 particles. This results in faster propagation and more realistic dispersion.

From a methodological perspective, the deterministic CO_2 model, computes the concentration of each cell as the average of the neighboring cells' concentrations, resulting in a smooth but uniform spread. Using random walk, the concentration update in each cell reflects a randomized fraction of the neighboring cells' CO_2 concentration. This modification includes local fluctuations without imposing a significant computational burden, as it avoids the need to track individual particles explicitly. In this context, the random walk approach provides a way to approximate the natural stochastic behavior of gas diffusion.

4 CONCLUSION

We presented a hybrid modeling approach for simulating CO_2 dispersion in indoor environments by integrating CA, DEVS, random walk algorithms and 2D and 3D modeling techniques. We addressed the limitations of traditional models by incorporating dynamic CO_2 generators, random-walk dispersion algorithms, and CO_2 sinks to capture more realistic indoor air behavior. Through a series of simulation experiments, we demonstrated how room layout, occupant movement, and source placement significantly influence CO_2 distribution patterns.

The proposed models are well-suited for studying CO_2 spread in enclosed spaces. They provide a flexible framework for evaluating the impact of various design and operational parameters, including sensor placement, ventilation strategies, occupancy levels, and furniture arrangements. This approach can support more informed decisions in the design and management of healthy and energy-efficient indoor environments.

ACKNOWLEDGMENTS

REFERENCES

Batog, P., and M. Badura. 2013. "Dynamic of Changes in Carbon Dioxide Concentration in Bedrooms". *Procedia Engineering* 57:175–182 https://doi.org/10.1016/j.proeng.2013.04.025.

Calì, D., P. Matthes, K. Huchtemann, R. Streblow, and D. Müller. 2015. "CO2 based Occupancy Detection Algorithm: Experimental Analysis and Validation for Office and Residential Buildings". *Building and Environment* 86:39–49 https://doi.org/10.1016/j.buildenv.2014.12.011.

Deng, X., Y. Shao, J. Song, and H. Wu. 2022. "Traffic Flow Simulation of Modified Cellular Automata Model Based on Producer-Consumer Algorithm". *PeerJ Computer Science* 8:1–16 https://doi.org/10.7717/peerj-cs.1102.

Deng, Z., B. Dong, X. Guo, and J. Zhang. 2024. "Impact of Indoor Air Quality and Multi-domain Factors on Human Productivity and Physiological Responses: A Comprehensive Review". *Indoor Air* 2024(1):1–25 https://doi.org/10.1155/2024/5584960.

Khalil, H., and G. Wainer. 2024. "A Framework for Modeling, Generating, Simulating, and Predicting Carbon Dioxide Dispersion Indoors using Cell-DEVS and Deep Learning". *Simulation* 100(5):421–434 https://doi.org/10.1177/00375497231212.

Khalil, H., G. Wainer, and Z. Dunnigan. 2020. "Cell-DEVS Models for CO2 Sensors Locations in Closed Spaces". In *2020 Winter Simulation Conference (WSC)*, 692–703 https://doi.org/10.1109/WSC48552.2020.9383937.

Labeodan, T., W. Zeiler, G. Boxem, and Y. Zhao. 2015. "Occupancy Measurement in Commercial Office Buildings for Demand-Driven Control Applications—A Survey and Detection System Evaluation". *Energy and Buildings* 93:303–314 https://doi.org/10.1016/j.enbuild.2015.02.028.

Marashian, S., S. Sadrizadeh, and O. Abouali. 2023. "Modeling Particle Distribution in a Ventilated Room with Modified Discrete Random Walk Methods". *International Journal of Ventilation* 22(3):289–306 https://doi.org/10.1080/14733315.2022.2143062.

Mendell, M. J., W. Chen, D. R. Ranasinghe, R. Castorina, and K. Kumagai. 2024. "Carbon Dioxide Guidelines for Indoor Air Quality: a Review". *Journal of Exposure Science & Environmental Epidemiology* 34(4):555–569 https://doi.org/10.1038/s41370-024-00694-7.

Mou, J., S. Cui, and D. W. Y. Khoo. 2022. "Computational Fluid Dynamics Modelling of Airflow and Carbon Dioxide Distribution Inside a Seminar Room for Sensor Placement". *Measurement: Sensors* 23:1–7 https://doi.org/10.1016/j.measen.2022.100402.

Pantazaras, A., S. E. Lee, M. Santamouris, and J. Yang. 2016. "Predicting the CO2 Levels in Buildings using Deterministic and Identified Models". *Energy and Buildings* 127:774–785 https://doi.org/10.1016/j.enbuild.2016.06.029.

Ryu, S. H., and H. J. Moon. 2016. "Development of an Occupancy Prediction Model using Indoor Environmental Data Based on Machine Learning Techniques". *Building and Environment* 107:1–9 https://doi.org/10.1016/j.buildenv.2016.06.039.

Sadrizadeh, S., R. Yao, F. Yuan, H. Awbi, W. Bahnfleth, Y. Bi, *et al*. 2022. "Indoor Air Quality and Health in Schools: A Critical Review for Developing the Roadmap for the Future School Environment". *Journal of Building Engineering* 57:1–23 https://doi.org/10.1016/j.jobe.2022.104908.

Sonnenschein, T. S., Z. Yuan, J. Khan, J. Kerckhoffs, R. C. Vermeulen, and S. Scheider. 2025. "Hybrid Cellular Automata-based Air Pollution Model for Traffic Scenario Microsimulations". *Environmental Modelling & Software*:1–13 https://doi.org/10.1016/j.envsoft.2025.106356.

Wainer, G. A. 2009. *Discrete-Event Modeling and Simulation: A Practitioner's Approach*. CRC Press https://doi.org/doi.org/10.1201/9781420053371.

Wang, J., W. Lv, Y. Jiang, and G. Huang. 2023. "A Cellular Automata Approach for Modelling Pedestrian-Vehicle Mixed Traffic Flow in Urban City". *Applied Mathematical Modelling* 115:1–33 https://doi.org/https://doi.org/10.1016/j.apm.2022.10.033.

Wang, P., and H. Mu. 2011. "Random-Walk Model Simulation of Air Pollutant Dispersion in Atmospheric Boundary Layer in China". *Environmental Monitoring and Assessment* 172:507–515 https://doi.org/https://doi.org/10.1007/s10661-010-1350-6.

Wolfram, S. 1983. "Statistical Mechanics of Cellular Automata". *Reviews of Modern Physics* 55(3):601–644 https://doi.org/10.1103/RevModPhys.55.601.

Wolfram, S. 1984. "Cellular Automata as Models of Complexity". *Nature* 311(5985):419–424 https://doi.org/doi.org/10.1038/311419a0.

Zeigler, B. P., H. Praehofer, and T. G. Kim. 2000. *Theory of Modeling and Simulation*. Academic Press.

Zhang, R., X. Xu, K. Liu, L. Kong, W. Wang, and T. Wortmann. 2024. "Airflow Modelling for Building Design: A Designers' Review". *Renewable and Sustainable Energy Reviews* 197:1–24 https://doi.org/10.1016/j.rser.2024.114380.

AUTHOR BIOGRAPHIES

MARINA MURILLO is a Postdoctoral Fellow at the Advanced Real-time Simulation Lab at Carleton University. Her email address is marinamurillo@cunet.carleton.ca.

GABRIEL WAINER, FSCS, received the M.Sc. (1993) at the University of Buenos Aires, Argentina, and the Ph.D. (1998, with highest honors) at UBA/Université d'Aix-Marseille III, France. In July 2000, he joined the Department of Systems and Computer Engineering at Carleton University (Ottawa, ON, Canada), where he is now a Full Professor. He is one of the founders of SIMUTools, ANNSIM, the Symposium on Theory of Modeling and Simulation, and the Symposium on Simulation in Architecture and Urban Design - SimAUD. Prof. Wainer is Editor in Chief of SIMULATION-Transactions of SCS, and a member of the Editorial Board of the Journal of Simulation (Taylor and Francis), IEEE Computing in Science and Engineering, Wireless Networks (Elsevier). He is an ACM Distinguished Speaker and a Fellow of SCS. He will be the General Chair of WinterSim 2029, in Buenos Aires, Argentina. His email address is gwainer@sce.carleton.ca.