

CD²: An Automation Tool for Cell-DEVs CO₂ Diffusion Models

Hoda Khalil¹ and Gabriel Wainer²

¹Systems and Computer Engineering
Ottawa, Canada
hodakhalil@cmail.carleton.ca

²Systems and Computer Engineering
Ottawa, Canada
gwainer@sce.carleton.ca

ABSTRACT

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 propose an application for modeling, simulating, and visualizing CO₂. The tool automates the simulation of Cellular Discrete Event Simulation models of CO₂ dispersion indoors. We discuss the tool's applications, the software architecture, related tools, and a case study. The case study compares simulations of indoor space with different configurations. The results show how such configurations affect CO₂ concentration.

Author Keywords

Carbon Dioxide; Cell-DEVS; simulation; indoors models.

ACM Classification Keywords

• Computing methodologies → Modeling and simulation •
Computing methodologies → Modeling and simulation →
Simulation support systems → Simulation tools

1 INTRODUCTION

Designers of modern buildings strive to reduce energy consumption while maintaining a healthy and comfortable indoor environment for the occupants of the buildings. An important indicator of the quality of indoor environments is the concentration level of carbon dioxide (CO₂). Measuring CO₂ concentration is not only essential for maintaining an acceptable comfort level, but it is also useful for occupants' detection and demand control of the buildings' services. However, studying the dispersion of CO₂ indoors is not attainable by physical experiments as the gas behavior changes depending on the configuration of the space (e.g., dimensions and presence of windows). Therefore, modeling and simulation (M&S) is a viable alternative for architects.

Researchers created Cellular Discrete Event Specifications (Cell-DEVS) models to study the diffusion of CO₂ indoors. The models have been proven effective for understanding

several effects of CO₂ accumulation [7, 8]. An example of the application of such models is understanding the effect of different parameters (e.g., dimensions and locations of windows) on the air quality. Another important application that has been addressed lately, is the use of CO₂ diffusion models as a way of determining the risk of viral infections in a closed space [5]. With the need to adapt the CO₂ diffusion models to different closed spaces, we have built a tool to facilitate creating, changing, and simulating CO₂ dispersion indoors. The CD² tool (Cell-DEVs CO₂ Diffusion) offers, through a web interface, a cycle of features that starts by converting a simple floorplan into a Cell-DEVS model. Through a web interface, the user can specify the parameters of the model. The application server triggers the simulation which is performed on a remote server. Finally, the application server downloads the simulation results that the user can view. CD² neither requires technical knowledge nor M&S tools installation on the users' side. It involves simple steps to create the model, set parameters, and simulate.

Section 2 lays the foundational background. Then, section 3 summarizes the model we use. In section 4, we present the software architecture, the workflow, and the technologies used. To further illustrate the applications of CD², we present a case study in section 5 and conclude in section 6.

2 BACKGROUND

DEVS is a mathematical formalism that is used to model complex systems. Cell-DEVS on the other hand is an extension of DEVS that combines DEVS with cell spaces. The applicability and advantages of Cell-DEVS to model CO₂ diffusion indoors has been presented in the literature [7, 8]. Cell-DEVS is supported by different tools and is proven to be applicable for modeling similar complex systems [6]. In previous work, we have used Cell-DEVS and its supporting tools to create models and simulate the spread of CO₂ in closed spaces for different M&S experiments [7, 8].

There are many applications for such CO₂ models. First, it is essential to maintain a balance between indoor comfortable CO₂ levels, while minimizing the energy consumed by ventilation systems. Second, CO₂ sensors' accuracy for detecting occupants is sensitive to the indoor configuration. CO₂ dispersion models can find the best locations for the

sensors and simulate their behavior based on the parameters of the indoor environments. Finally, since occupants exhale CO₂ mixed with aerosols, the higher the CO₂ concentration, the more aerosols, including viral particles occupants may inhale. Thus, CO₂ concentration has recently been proposed as a proxy for the possibility of viral infection indoors [10].

Batog and Badura have used CO₂ dispersion M&S to simulate CO₂ dispersion in a bedroom with one sleeping occupant. The results show that the air quality in small is unhealthy, and thus extra ventilation is required. The authors have reported that CO₂ sensors' locations affect the validity of the reading [3]. Also, researchers have 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 [15].

Our CO₂ models are built using CD++ [15]; a toolkit that implements DEVS and Cell-DEVS theoretical concepts. In CD++, the rules that define a Cell-DEVS model are defined using a scripting language [14]. CD² uses a CD++ engine running remotely using the RESTful Interoperability Simulation Environment (RISE) middleware [1]. CD² provides a simulation cycle for modeling a specific phenomenon: CO₂ dispersion indoors. It uses a remote server for model execution. It requires minimal knowledge of software concepts since it abstracts all M&S technicalities from the end-users. At present, there are no similar tools to serve the goals outlined earlier. However, other tools serve as containers for creating the lifecycle of cellular models, targeting experts who design and create models rather than execute them. For instance, NetLogo offers an interface, in the form of buttons and sliders, to build models from templates. However, it does not allow building hierarchical models [12, 13]. As well, the Virtual Laboratory Environment (VLE) has a graphical interface to define DEVS and Cell-DEVS models [11, 12]. The Component-Based System Modeler and Simulator (CosMos-CA) offers an interface to run the lifecycle process for DEVS and Cell-DEVS models [12]. CA Evaluation (CAEva) is designed for checking building conditions for fire accidents and is built using CD++. The tool aims to help the modelers understand the consequences of not having sufficient escape routes [4].

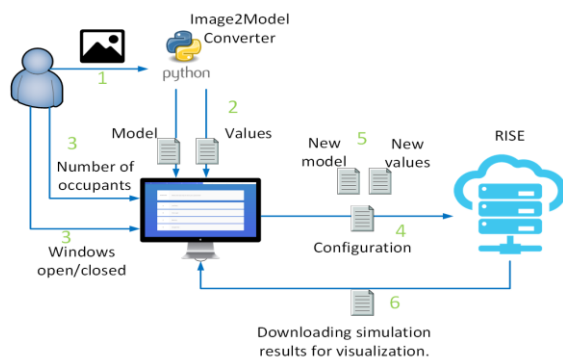


Figure 1. CD² workflow.

3 THE MODEL

The model that we focus on here represents a closed space with the following parameters: room dimensions, specification (locations and dimensions) of CO₂ sinks (windows, doors, ventilation ports), and locations of occupants. We base this model on our previously validated, against ground truth data, CO₂ models [7, 8]. The mathematical specifications of the model are tool agnostic and independent of any M&S language. However, changing the model rules to incorporate other dispersions methods is independent of the CD² code.

4 CD²

As the CO₂ diffusion models have evolved and become more useful in various applications, applying the models to scenarios of a variety of indoor spaces has gained interest. In many experiments, it is useful to vary the parameters in the model and run the simulations of the different scenarios to visualize the consequences. Such a process involves converting a floor plan to a file with initial values for the state variables in each cell in the model, setting parameters manually (e.g., number of occupants), connecting to RISE to run the simulation, downloading the simulation, and visualizing results. Handling these steps manually and connecting the output of each tool to the input of the tool used in the next stage is time-consuming, error-prone, and unintuitive. Our web application provides the user with a single interface giving them the ability to run the various steps automatically. Figure 1 explains the workflow of CD². The first step is to convert a floorplan image to a Cell-DEVS model for CD++ to understand. The usual image file formats are accepted as an input including JPEG, TIFF, and BMP. The output is a model file (.ma) and an initial values file (.val). Six types of cells (walls, air, doors, windows, diffusers ventilation ports, and grills ventilation) can be generated based on the dimensions and the color-coding of the input. The user uploads the floorplan, specify the space width in meters, and the conversion tool calculates the width, in the number of cells, of the resulting model. The length is calculated to scale based on the input. Second, the user specifies some settings; for instance, the status of the windows (closed or open). Also, the user can specify the number of occupants to be generated randomly in different empty locations as cells of type CO₂ source. The input to this step is the model created in step 1 represented by the model file and initial values file. Now, CD² establishes communication with RISE. RISE requires a specific HTTP request as a contract in the form of a header and a body to expose its services over the web. Creating this form of request is not for non-technical users that are not aware of the system's specifications. In addition, discovering errors created in such a process is time-consuming. CD² handles this detail and hides it from the users. Then, CD² connects to RISE to run the simulation. The configuration is passed to RISE as an XML file containing the model description, the simulation time, configuration parameters, and the names of the files defining the model. At this step, RISE has created a

simulation environment to run the model. Then, CD² calls the appropriate APIs to submit the model files to the RISE web service, which simulates without further user intervention. Finally, when RISE finishes simulating the model, it stores the simulation results on the RISE server. Once the simulation results are available, the user would use CD² to download the results on their workstation.

CD² is a client-server software. The client is a web interface that the users interact with. The application server is responsible for the core functionality. We have leveraged the servlet architecture and Java Server Pages (JSPs) for the webpages, while the server is coded in Java.

5 CASE STUDY

In this section, we show two different scenarios where CD² is useful to predict the effects of different parameters on indoor air quality. The scenarios are a few samples of more possible scenarios that CD² can be used for. In the first scenario (Scenario I), we model a closed space with five occupants, while the windows are closed throughout the simulation. The user enters the width of the space (25 meters). The tool translates the input image into a grid of cells (109×100 cells where each cell represents a space of 25×25×25 cm³). After generating the grid (the model and the initial values files), the user specifies the windows' status and the number of occupants (to be generated in random locations). Figure 2 shows the simulation results of Scenario I where CO₂ concentration is measured in particles per million (ppm). The figure shows that the area around the occupants has a high CO₂ concentration (dark red spots) due to constant breathing even while the remaining CO₂ diffuses in the room. The vents are working constantly to offset CO₂ concentration increase as shown in the figure (blue areas).

In Scenario II, we keep the same parameters, except for the windows which we set to open. Figure 3 shows the simulation results where open windows introduce fresh air with lower CO₂ concentration. For example, the upper and lower right corners of the floorplan have lower CO₂ concentrations. From a user's point of view, simulating Scenario II requires the same process as simulating Scenario I while only unchecking the checkbox for windows status. The details of how this is done are hidden from the user. Also, applying this experiment to another floorplan requires only uploading a different image (e.g., in this example this would mean initializing more than 10000 cells). The charts in Figure 2 and Figure 3 are plots of the change in CO₂ concentration throughout the simulation when the windows are closed or open, respectively. The charted CO₂ concentration is measured in the cell with (90,90) coordinates, which is in the southeast quarter of the floorplan. The used cell represents the sensor's location. As apparent from the chart of Scenario I (see Figure 2), the CO₂ increase is offset by the existing ventilation only for a short period. Then, the CO₂ concentration keeps increasing even with the presence of the vents. On the contrary, in Scenario II the concentration keeps decreasing, below 500 ppm

providing better air quality due to the fresh air from the windows. Closer to the end of the simulation, the occupants' breath works against the effect of the fresh air. However, even then, the CO₂ concentration does not reach the same level registered with closed windows. Demos of other scenarios are available through our YouTube channel [2].

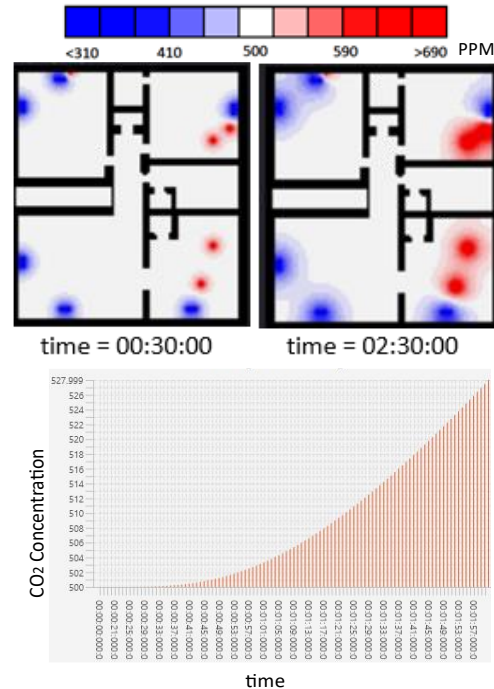


Figure 2. Simulation results of Scenario I.

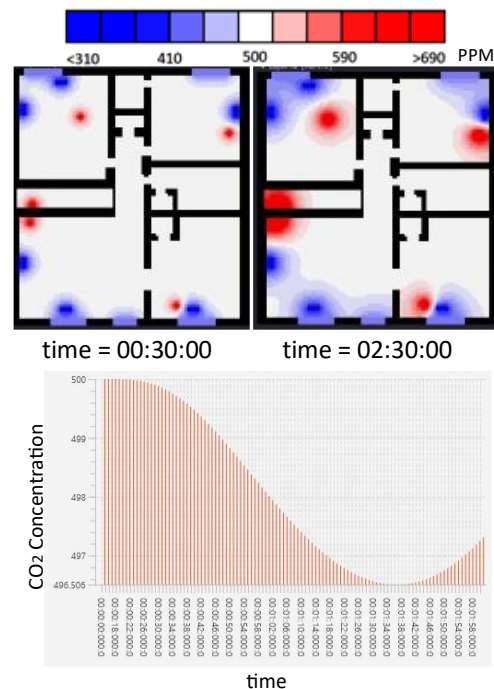


Figure 3 Simulation results of Scenario II.

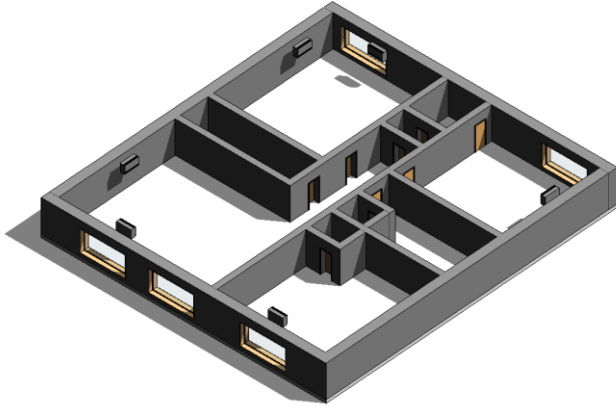


Figure 4. A BIM Model created using AutoDesk Revit© for the floorplan used in Scenario I and Scenario II.

CD² is still in its early phases, but at present, we are working on integrating new functionalities into CD². One upcoming functionality is giving the user the option to upload a Building Information Model (BIM) as the base model instead of the floorplan image. Figure 4 is the BIM, created using Autodesk Revit [16], of the example that we illustrated in the simulations (Figures 2 and 3). The source code of the current version of CD² and the installation manual is available via GitHub [9]. We also plan to test the tool with other existing CO₂ that includes moving occupants.

6 CONCLUSION

Motivated by the need for applying CO₂ models to a variety of indoor spaces with different configurations, we have developed an application tool (CD²) to automate the simulation cycle of CO₂ diffusion models. CD² is specifically built to facilitate the process of applying different scenarios to our CO₂ models. Hence, the interface is simple and easy to use. CD² allows its users to create a Cell-DEVS model from a simple 2-dimensional floorplan. The user can specify the settings of that space (e.g., windows status, be it closed or opened, and the number of occupants). Then, CD² connects to the RISE server to runs the simulation. Finally, the user can download the simulation results through the CD² web interface. We have illustrated through a case study how CD² facilitates experimenting with the effect of changing indoor parameters on CO₂ diffusion through simple steps.

ACKNOWLEDGMENTS

The BIM diagram was made possible through the contribution of Nicolás Arellano Risopatrón and Stephen Fai, CIMS Lab. The authors are thankful for their support.

REFERENCES

1. Al-Zoubi, K. and Wainer, G. Distributed simulation of DEVS and Cell-DEVS using the RISE middleware. *Simulation Modelling Practice and Theory* 55, (2015), 27-45.
2. ARSLab YouTube channel, <https://www.youtube.com/user/ARSLab/videos>. As of 21 November 2020.
3. Batog, P. and Badura, M. Dynamic of changes in carbon dioxide concentration in bedrooms. *Procedia Engineering* 57, (2013), 175–182.
4. Czerniak, J.M., Apiecioneck, L., Zarzycki, H., and Ewald, D. Proposed CAEva simulation method for evacuation of people from buildings on fire. *Advances in Intelligent Systems and Computing* 401, (2016), 315-326.
5. Hatcher, J. Carbon dioxide monitoring to lower the coronavirus threat. Smart Buildings Magazine. <https://smartbuildingsmagazine.com/features/carbon-dioxide-monitoring-to-lower-the-coronavirus-threat>. As of 2 November 2020.
6. Khalil, H. and Wainer, G. Cell-DEVS for social phenomena modeling. *IEEE Transaction on Computational Social Systems* 7, 3 (2020), 725–740.
7. Khalil, H., Wainer, G. and Dunnigan, Z. Cell-DEVS models for CO₂ sensors locations in closed spaces. *Proc. WSC 2020*, IEEE Press (2020).
8. Khalil, H. and Wainer, G. Modeling carbon dioxide dispersion indoors: a Cell-DEVS experiment. *LNCS. ACRI 2020*.
9. Khalil, H. and Wainer, G. CD², <https://github.com/hakhalil/SimulationChainTool>. As of 21 November 2020.
10. Peng, Z. and Jimenez, J. L. Exhaled CO₂ as COVID-19 infection risk proxy for different indoor environments and activities, medRxiv. (2020).
11. Quesnel, G., Duboz, R., and Ramat, E. The virtual laboratory environment – An operational framework for multi-modeling, simulation and analysis of complex dynamical systems. *Simulation Modelling Practice and Theory* 17, 4 (2009), 641-653.
12. Sarjoughian, H., Sarkar, S., and Mayer, G. A Novel visual CA modeling approach and its realization in CosMoS. *Proc. SpringSim*, ACM Press (2010), 1-4.
13. Tisue, S. and Wilensky, U. NetLogo: Design and implementation of a multi-agent modeling environment. *Proc. Agent 2004*, ANL (2004).
14. Wainer, G. *Discrete-event modeling and simulation: a practitioner's approach*. (1st. ed.). CRC Press, Boca Raton (2009).
15. Yalçın, N., Balta, D., and Özmen, A. A modeling and simulation study about CO₂ amount with web-based indoor air quality monitoring. *Turkish Journal of Electrical Engineering and Computer Sciences* 26, (2018), 1390-1402.
16. Zotkin, S. P., Ignatova, E. V., and Zotkina, I. A. The Organization of Autodesk Revit Software Interaction with Applications for Structural Analysis. *Proc. Eng.* 153, (2016), 915-919.