Department: Computer Simulations
Editors: Barry Schneider, bis@nist.gov and Gabriel A. Wainer, gwainer@sce.carleton.ca

Agent-based Modeling of Traffic Systems using Modelica

Victorino Sanz
Dpto. de Informática y Automática, Universidad Nacional de Educación a Distancia (UNED)

Alfonso Urquia
Dpto. de Informática y Automática, Universidad Nacional de Educación a Distancia (UNED)

***Abstract*—A micro-simulation model of traffic described using the Modelica language in combination with the Modelica libraries developed by the authors, ABMLib and CellularAutomataLib2, is presented. Modelica facilitates the description of equation-based models using the physical modeling paradigm, and its combination with the discrete-event and agent-based modeling functionality of the developed libraries constitutes a powerful and versatile tool. The functionality of ABMLib is extended to facilitate the description of individual agent behavior as a combination of equation-based and discrete-event models. In this way, vehicle dynamics and fuel consumption are modeled using equations, while driver behavior is described using an agent-based model. Vehicles move along the road, which is described as a combination of interconnected segments that constitute the environment for the agents. A cellular automaton is used to model the CO_2 emissions of vehicles to the environment. A model that represents two consecutive road segments with a semaphore between them is used to illustrate the provided functionality.**

■ **TRAFFIC** systems are almost indispensable to support our current society, from the distribution of goods to our own transportation. They constitute large and complex systems that also have a severe impact on our environment. Modeling and simulation is a suitable tool to analyze and study these systems in order to optimize their benefits and minimize their shortcomings.

Modelica is a general-purpose modeling language designed to support the physical modeling paradigm. The functionality of the language facilitates the description of equation-based models, combining continuous-time dynamics and discrete-events [1]. The authors have developed a methodology, implemented in the ABMLib library, for describing agent-based models (ABM) in Modelica, and combine them with other Modelica models [2].

Briefly, the modeling approach applied in ABMLib is to represent the behavior of agents as a set of processes arranged in a flowchart diagram. Each agent is represented by a message that flows from one process to another, performing the actions defined by these processes (e.g.,

Published by the IEEE Computer Society

move, turn, eat, etc.). The model also includes a description of the environment, where the agents interact. Parallel DEVS (PDEVS) [3] is used as the underlying formalism for the description of processes and their interconnections. Cellular automata models are also supported in Modelica by means of the CellularAutomataLib2 library, designed and developed by the authors [4].

The objective of the work presented in this manuscript is to combine equation-based, agent-based and cellular automata models to describe traffic systems using the Modelica language. An equation-based model is used to describe the dynamics (e.g., position, speed, acceleration, etc.) and fuel consumption of each individual vehicle using the modeling functionality provided by Modelica. This equation-based model is associated with an agent-based description of the driver, leading to a combined vehicle-driver model. The vehicle-driver models are also combined with a cellular automaton to observe the effect of the CO_2 emitted by each vehicle to the environment, including the diffusion and wind dispersion of the gas. The combination of multiple modeling formalisms provides a versatile approach and facilitates the description and study of traffic systems. The presented work constitutes a demonstrative example of the functionality provided by Modelica and the developed libraries for the study of traffic systems.

RELATED WORK

Three approaches are mainly used to model traffic systems: microscopic, that describes the individual behavior of each vehicle and its interactions with others; mesoscopic, that focuses on the description of groups of entities, providing a coarser description; and macroscopic, that describes the traffic as a flow using the physical laws of fluids.

The work described in this manuscript is focused on the description of vehicle and driver behavior at the microscopic scale, using a combination of equation-based and agent-based models. On one hand, equation-based models have been extensively used. A review on the evolution of traffic micro-simulation models and simulation platforms can be found in [5]. On the other hand, agent-based modeling provides a versatile and efficient approach to describe multiple types of traffic systems. Reviews of agent-based traffic simulators can be found in [6], [7].

The analysis of traffic emissions and pollution is also a relevant field of study, due to their importance for maintaining healthy environments. Approaches to estimate traffic emissions are discussed in [8] and [9] using the PARAMICS model, or to analyze the emissions in roundabout corridors in United States and Portugal [10].

TRAFFIC MODELS IN ABMLIB

The models considered in this work include the description of the vehicle dynamics, the behavior of the driver, and the conditions of the road. Additionally, a fuel consumption and CO_2 generation model has been included to illustrate the impact of the traffic on its surrounding environment. The description of these model components is detailed below, together with the extensions developed and included in ABMLib to fulfill the modeling requirements.

Vehicle Dynamics Model

Vehicle dynamics are described using the Intelligent Driver Model (IDM) [11]:

$$\dot{v} = a\left[1 - \left(\frac{v}{v_0}\right)^{\delta} - \left(\frac{s^{\star}(v,\Delta v)}{s}\right)^{2}\right] \quad (1)$$

$$s^{\star}(v,\Delta v) = s_0 + max\left(0, vT + \frac{v\Delta v}{2\sqrt{ab}}\right) \quad (2)$$

$$s = x_l - x - L \quad (3)$$

$$\Delta v = v_l - v \quad (4)$$

where, (x, v) are the position and speed of the vehicle, and (x_l, v_l) are the position and speed of the leading vehicle. The parameters of the model are shown in Table 1.

The equations of the IDM model can be directly coded in Modelica as shown in Listing 1. An additional variable, `active`, is used to define if the vehicle is currently active in the model. An array of IDM models has to be declared, where each vehicle is identified by its index in the array, to represent the maximum number of vehicles that can be active in the simulation, since the number of variables and equations need the remain constant during Modelica simulations. During the simulation, each driver agent is linked to an IDM model of the array. The `active` variable is used as a binary flag to control the

Table 1: Parameters of the IDM and fuel consumption models.

Parameter	Unit	Value	Description
L	m	4	vehicle length
a	$\mathrm{m/s^2}$	1	comfortable accel.
b	$\mathrm{m/s^2}$	1.5	comfortable decel.
v_0	m/s	33.3	desired speed
s_0	m	2	minimum distance gap
T	s	1	desired time gap
δ	-	4	accel. control exponent
m	kg	1500	mass of the vehicle
μ	-	0.02	friction coefficient
ϕ	rad	0	uphill gradient
g	$\mathrm{m/s^2}$	9.81	gravitational accel.
ρ	$\mathrm{kg/m^3}$	1.3	air density
A	$\mathrm{m^2}$	2	cross section
c_d	-	0.3	aerodynamic drag
P_0	kW	3	idling power
γ	-	0.3	efficiency factor
w_{cal}	kW h	11	gasoline energy density

Listing 1: Modelica code of the IDM model.

```
import Modelica.Units.SI;
SI.Position x,s,sstar;
SI.Velocity v,dv;
SI.Acceleration derv;
Integer active(start = 0);

der(x)= active*max(0,v);
der(v)= active*derv;
derv = a*(1-(v/v0)^delta-(sstar/s)^2);
sstar= s0 + max(0,v*T + (v*dv)/(2*sqrt(a*b)));
s = frontx - x - L;
dv = frontv - v;
```

vehicles of the array that are active at any time during the simulation run. The value of `active` is set depending on the presence of its related agent in the environment, using the `AGpresent` function from ABMLib. Equations of inactive vehicles are not evaluated during the simulation and their variables remain constant. In this way, a continuous-time behavior can be assigned to each individual agent, and combined with its discrete-event behavior.

When a vehicle becomes active its position and speed are reinit with initial values, observed from the attributes of its related driver agent. Also, the position and speed of the leading vehicle are periodically observed from the environment. This is used to represent the concentration of the drivers on the road, and the occurrence of events in the traffic (e.g., jams, accidents, road works, etc.).

Fuel Consumption Model

The vehicle model includes a fuel consumption and CO_2 emission model [11]. This allows using these models to analyze the environmental impact of traffic systems. The fuel consumption model corresponds to Equations (5), (6), and (7), where, F is the driving resistance, P is the power demand from the motor, and $\dot{C}$ is the fuel consumption rate. Model parameters are shown in Table 1.

$$F(v,\dot{v}) = m\dot{v} + (\mu + \sin\phi)mg + \frac{1}{2}c_d\rho A v^2 \quad (5)$$

$$P(v,\dot{v}) = max[P_0 + vF(v,\dot{v}), 0] \quad (6)$$

$$\dot{C} = \frac{P}{\gamma(P,f)w_{cal}} \quad (7)$$

CO_2 emissions follow a direct relationship with fuel consumption, being 2.39 kg of CO_2 per liter of gasoline. The diffusion of CO_2 emissions from vehicles to the environment is modeled using a cellular automaton (CA). The CA represents the 2D space that surrounds the vehicles, superposed to the topology of the roads. The movement of vehicles along the roads is considered to calculate their position in the CA space. The amount of CO_2 in each cell ($P_{i,j}(t)$) is periodically updated considering the inputs from individual vehicles ($Pin_{i,j}(t)$), the diffusion of gases ($Pd_{i,j}(t)$) and the transportation due to wind ($Pw_{i,j}(t)$) [12]:

$$\begin{aligned} &P_{i,j}(t+1) = \\ &P_{i,j}(t) + Pin_{i,j}(t) + Pd_{i,j}(t) + Pw_{i,j}(t) \quad (8) \end{aligned}$$

The amount of CO_2 transported due to diffusion is computed as [12]:

$$Pd_{i,j}(t) = \delta\left(\sum_{n=1}^{8} 0.125 P_n(t) - P_{i,j}(t)\right) \quad (9)$$

where $P_{i,j}(t)$ is the amount of gas in cell $[i,j]$ at time t, δ is the diffusion coefficient and $P_n(t)$ corresponds to the gas of neighbor n of cell $[i,j]$. Wind dispersion is computed as [12]:

$$Pw_{i,j}(t) = \beta\left(P_w(t) - P_{i,j}(t)\right) \quad (10)$$

where $P_{i,j}$ is the cell being evaluated, β is the coefficient that represents the wind force, and P_w is the source neighboring cell according to the wind direction (i.e., wind blows from P_w to $P_{i,j}$). The description of the automaton is performed using the CellularAutomataLib2 library, that combines external C code with Modelica in order to improve the performance of the simulation [4]. The simulation algorithm only evaluates the cells of the automaton that are likely to modify their state during each simulation step. Simulation steps are periodically scheduled as time events for the Modelica simulation algorithm. A graphical animation is automatically generated during the simulation run to facilitate the observation of the impact of vehicles on their environment. Similarly, additional pollution emission models, such as NO_x, can be included in the vehicle model to evaluate their environmental impact.

Driver Behavior Model

The behavior of drivers is described following the proposed ABM approach [2], and implemented using the functionality provided by ABMLib. Each agent represents a driver and is related to a vehicle model, having driver-vehicle agents, as described above. Currently, driver behavior is designed to follow a single-lane, collision-free route along the road. The flowchart diagram used to describe this behavior is shown in Figure 1a, and it is detailed below. Since this behavior corresponds to the agent-based part of the model it has to be defined together with the models used to describe the environment for the agents, as road segments. Additional driver behaviors, such as driving in multiple lanes, politeness, selecting direction at intersections, etc., could be easily included in the diagram.

Road Model

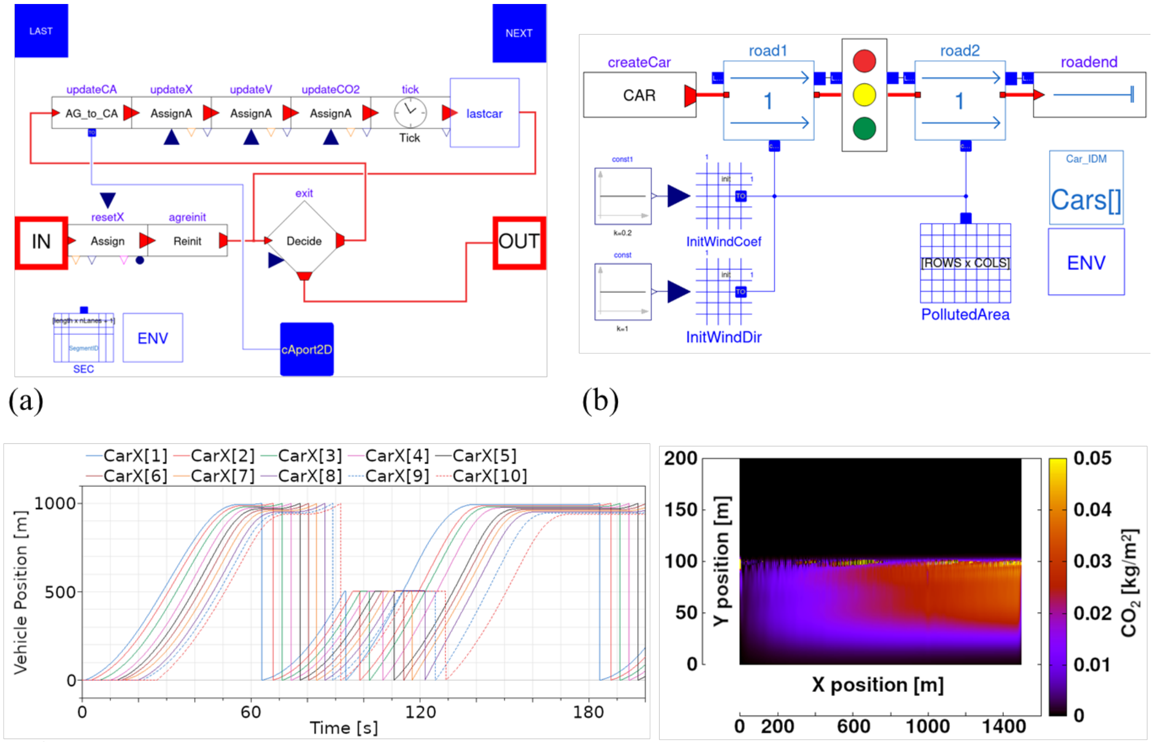
Roads constitute the environment for driver-vehicle agents, and they are described as a network of interconnected road segments. Each road segment represents a section of the road that can be connected to other sections to describe the desired route topology. The road segment model represents a single-lane road of a given length, l. ABMLib also includes components to describe sources and destinations for vehicles, and a semaphore model. Additional components can be easily modeled, including on- and off-ramps, intersections, etc.

ABMLib functionality has been extended to facilitate the description of more complex environments, and traffic systems in particular. It provides functionality to assign unique IDs to agents, independently of their initial position in the environment. The environment model is now composed of:

- Agent population, that constitutes a list of the agents present in the simulation. Thus, agent attributes can be directly observed and updated regarding the position of the agent in the environment, facilitating the interaction between the continuous-time equations of the vehicle model and the driver agents.
- Environment sections, that represent different 2D spaces where the agents can behave and interact. This facilitates the description of the road topology and the identification of leading vehicles for each agent in each section.

The road segment model includes the flowchart diagram used to represent the behavior of drivers along the road (cf. Figure 1a). Each road segment receives new agents at the `IN` port. Note the difference between ports, used as interface for message passing communication, and Modelica connectors, used for equation-based model connections. The position of each received agent is reset to 0, to represent their initial position in the local environment. The value of `x` (i.e., the continuous-time variable that represents the position of the vehicle in the IDM model) is also reset to 0. After that, agents enter a loop where their attributes are periodically updated using the values computed by the IDM model. An IDM model for each vehicle is simulated in continuous-time to compute the speed and position of the vehicle. Also, the input to the CA model for the evaluation of the CO_2 emissions is updated using the `CAport2D` connector. Vehicles exit this loop when they reach the end of the segment, l, and they are sent to the next segment using the `OUT` port.

The segment model also includes two connectors: `NEXT` and `LAST`. The `LAST` connector is used to communicate the position and speed of the last vehicle in a segment to the previous segment. The `LAST` connector needs to be connected to the `NEXT` connector of the previous segment. Thus, the `NEXT` connector of a segment is used to compute the dynamics of the IDM model for the first vehicle if the segment, since its leading vehicle is in the following segment.



(a) (b) (c) (d)

Figure 1: (a) Diagram of the road segment model. (b) Diagram of the semaphore model. (c) Positions of vehicles along the road. (d) Diffusion and dispersion of CO_2 at time=400 s.

TEST CASE

A model of two consecutive road segments with a semaphore between them is used to illustrate the provided functionality. The ABM diagram (cf. Figure 1b) is composed of `createCar`, as source of vehicles, `road1` and `road2`, as road segments, the semaphore, and `roadEnd`, as final destination for the vehicles. The semaphore model behaves like a switch between the road segments. When the light is green, the value of the LAST connector of `road2` is passed across the semaphore to the NEXT connector of `road1`. Otherwise, the variables of the NEXT connector of `road1` are set to zero, forcing the vehicles to stop at the semaphore position. The semaphore light switches every 60 s. The continuous-time model of vehicle dynamics corresponds to the array of `Car_IDM` models, named `Cars`. The cellular automaton is composed of the `PollutedArea` model, and the `initWindCoef` and `initWindDir` models that are used to configure the wind force and direction.

Vehicles arrive to the road at intervals between 1 and 5 s. Their initial speed is randomly selected between 10 and 33.3 m/s. The length of the road segments is set to 1000 m and 500 m, respectively, and the maximum number of active vehicles is set to ten. The period for the transitions of the CA is 1 s. The wind blows to the south-east direction with coefficients $\delta = 0.5$ and $\beta = 0.2$ (cf. Equations (9) and (10)).

The model has been simulated for 400 s to observe the effect of the semaphore in the emissions. The position of the vehicles during the first half of the simulation is shown in Figure 1c (only half of the simulation time span is shown to better visualize the graph). Vehicles that reach the limit of the road (i.e. 500 m of `road2`) leave the model, are set as inactive, and the values of their continuous-time variables remain constant. When a new vehicle is created, a position in the array of IDM models is assigned to the new agent and the continuous-time variables are reset to 0 and set to active again. The semaphore is red during the time intervals [0,60] and [120,180] s.

The animation of the CA model is shown in Figure 1d, where waves in the CO_2 emissions can be observed due to the effect of the vehicles that periodically stop at the semaphore.

CONCLUSION

The description of traffic systems using the Modelica language has been discussed. The presented modeling functionality can be used as a base for modeling more complex vehicle-driver behaviors (e.g., lane changing, route decision, etc.) and road topologies (e.g, including more segments or other elements). Also, these models can be applied to evaluate the performance of Low Emission Zones (LEZs), increasingly adopted across European cities.

Modelica, and the DEVSLib, ABMLib and CellularAutomataLib2 libraries constitute a versatile alternative for describing models in a multi-formalism and multi-level approach. The combination of equation-based and agent-based models using the proposed approach, at the system-level, has been previously demonstrated. Additionally, the work described in this manuscript allows to describe the individual behavior of each agent as a combination of equation-based and discrete-event dynamics, while having a variable number of agents during the simulation. The restriction of a constant number of variables and equations imposed by Modelica is surmounted by declaring a maximum amount of equation-based models, that are used on-demand becoming active when associated to a newly created agent or remaining inactive otherwise.

ACKNOWLEDGMENT

This research was supported by Vicerrectorado de Investigación, Transferencia de Conocimiento y Divulgación Científica of Universidad Nacional de Educación a Distancia (UNED), “Convocatoria Proyectos de Investigación UNED 2022” grant.

■ REFERENCES

1. P. Fritzson, *Principles of Object-Oriented Modeling and Simulation with Modelica 3.3: A Cyber-Physical Approach*. Wiley-IEEE Computer Society Pr, 2014.
2. V. Sanz and A. Urquia, “Combining PDEVS and Modelica for describing agent-based models,” *SIMULATION*, 2022.
3. B. P. Zeigler, A. Muzy, and E. Kofman, *Theory of Modeling and Simulation: Discrete Event and Iterative System Computational Foundations*, 3rd ed. New York: Academic Press, 2018.
4. V. Sanz, A. Urquia, and A. Leva, “CellularAutomataLib2: Improving the support for cellular automata modeling in Modelica,” *Mathematical and Computer Modelling of Dynamical Systems*, vol. 22, no. 3, pp. 244–264, 2016.
5. N. Raju and H. Farah, “Evolution of traffic microsimulation and its use for modeling connected and automated vehicles,” *Journal of Advanced Transportation*, vol. 2021, p. 29, 2021.
6. J. Nguyen, S. T. Powers, N. Urquhart, T. Farrenkopf, and M. Guckert, “An overview of agent-based traffic simulators,” *Transportation Research Interdisciplinary Perspectives*, vol. 12, p. 100486, 2021.
7. J. Huang, Y. Cui, L. Zhang, W. Tong, Y. Shi, and Z. Liu, “An overview of agent-based models for transport simulation and analysis,” *Journal of Advanced Transportation*, vol. 2022, p. 17, 2022.
8. A. Misra, M. J. Roorda, and H. L. MacLean, “An integrated modelling approach to estimate urban traffic emissions,” *Atmospheric Environment*, vol. 73, pp. 81–91, 2013.
9. G. Amirjamshidi, T. S. Mostafa, A. Misra, and M. J. Roorda, “Integrated model for microsimulating vehicle emissions, pollutant dispersion and population exposure,” *Transportation Research Part D: Transport and Environment*, vol. 18, pp. 16–24, 2013.
10. P. Fernandes, K. Salamati, N. Rouphail, and M. Coelho, “Identification of emission hotspots in roundabouts corridors,” *Transportation Research Part D: Transport and Environment*, vol. 37, pp. 48–64, 2015.
11. M. Treiber and A. Kesting, *Traffic Flow Dynamics - Data, Models and Simulation*. London, UK: Springer, 2013.
12. S. Benjavanich, Z. Ursani, and D. Corne, “Forecasting the flow of urban pollution with cellular automata,” in *Proceedings of the 5^{th} IFIP Conference on Sustainable Internet and ICT for Sustainability (SustainIT)*, Funchal, Portugal, 2017, pp. 1–6.

Victorino Sanz is Assistant Professor at Dpto. de Informática y Automática of UNED in Madrid, Spain. Contact him at vsanz@dia.uned.es.

Alfonso Urquia is Associate Professor at Dpto. de Informática y Automática of UNED in Madrid, Spain. Contact him at aurquia@dia.uned.es.