



ELSEVIER

Contents lists available at [ScienceDirect](#)

MethodsX

journal homepage: www.elsevier.com/locate/mex



Method Article

Quantifying the health effects of exposure to non-exhaust road emissions using agent-based modelling (ABM)



Hyesop Shin

School of Geographical and Earth Sciences, University of Glasgow, G12 8QQ, UK

ABSTRACT

This paper provides an agent-based model, entitled TRAPSim, to examine the exposure to non-exhaust emissions (NEEs) and the consequent health effects of driver and pedestrians groups in Seoul. To make the model reproducible and replicable, TRAPSim uses the ODD protocol to demonstrate the details of the agents and parameters, as well as provide the codes alongside the descriptions to avoid possible ambiguity. The model's main parameters are thoroughly tested through sensitivity experiments and are calibrated with the city's air pollution monitoring networks. This paper also provides the instructions to the model, possible artefacts, and the configurations to submit the model on the HPC cluster.

- An ODD protocol is used to document the agent-based model TRAPSim.
- Sensitivity experiments and calibration are explained.
- The step-by-step codes and annotations are attached in the protocol and HPC sections.

© 2022 The Author(s). Published by Elsevier B.V.

This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

ARTICLE INFO

Keywords: Agent-based modelling, Traffic simulation, Air pollution, Exposure, NetLogo

Article history: Received 29 November 2021; Accepted 15 March 2022; Available online 25 March 2022

E-mail address: hyesop.shin@glasgow.ac.uk

<https://doi.org/10.1016/j.mex.2022.101673>

2215-0161/© 2022 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

Specifications table

Subject Area:	Environmental science
More specific subject area:	Social Simulation
Method name:	TRAPSim (Transport-related Air Pollution Simulation specialised for simulating non-exhaust emission)
Name and reference of original method:	N/A
Resource availability:	The data and the codes are stored in the Harvard Dataverse (https://doi.org/10.7910/DVN/C93XLZ) The data, codes, bug reports, and wiki pages are written on https://github.com/dataandcrowd/SeoultrafficABM/wiki NetLogo 6.0.4 was used for the model (https://ccl.northwestern.edu/netlogo/download.shtml) R 3.6.1 was used for the HPC works and post-processing analysis (https://cran.r-project.org/) Java 8 was used to run NetLogo on a headless mode (https://www.oracle.com/uk/java/technologies/javase/javase8-archive-downloads.html)

Introduction

Non-exhaust emissions (NEEs) are generated from the friction between tyres, the road surface, and pavement encrustations in the form of metallic, rubber, carbon black, and other organic substances [1,2]. Traffic-related air pollutants (TRAP) that include tyre wear, brake wear, road abrasion, and road resuspension broadly contribute to the ambient air quality [1,3,4]. Previous findings have shown the contribution of non-exhaust emissions account for from 11% [5] to 60% [6] and 73% [7] of the roadside air pollution. Although e-scooters and electric vehicles are likely to reduce exhaust emissions [8,9], individuals who get exposed to NEEs together with background air pollution can result in adverse health effects from eye irritation to lung and heart impairment [10].

To measure population exposure to NEEs at an individual level, we applied agent-based modelling (ABM). Unlike statistical approaches that only consider collective exposure levels by demographic or boundary groups at a certain time frame, ABM provides the unique characteristics (e.g. daily commute patterns, health status) of individuals over time and space [11–14]. The recreation of the real world enables users to raise awareness of more polluted areas, what it has to do with traffic, and unravel the relationship between exposure and means of transportation. Other agent-based traffic models have also simulated the vehicle emissions on a city scale, however, these models were too short to discover the harms of the different exposures based on people's and vehicle activity. In addition, since the models were focused on traffic movement, the exposure levels did not fully account for whether the individuals were staying indoors or outdoors, which, from the cases of the UK, 95% of the Londoners spend their time indoors [15].

Using the case of Seoul, we developed an agent-based model, named TRAPSim, to examine the exposure to NEEs and the consequent health effects by driver and pedestrians groups. To our knowledge, this is the first model that conjoins the mobility of vehicles and people, the generation of PM₁₀ (i.e. particles with diameters that are up to 10 μm) at every grid by vehicles and background sources, and the cumulative exposure to PM₁₀ that lead to adverse health effects.

Amongst various types of social simulation that ranges from the highly detailed version of the real world to a conceptual model, this paper positions a place as an illustrative model [16]. The illustrative model aims to communicate or make clear an idea, theory or explanation, thus is less burdened to support claims. Our expectation is to disseminate the model so that scholars in similar disciplines can use and redesign our model for their purposes.

Method details

This article provides technical documentation of TRAPSim partly based on the ODD protocol (Overview, Design Concepts and Details) [17]. The ODD protocol is a standardised method to describe simulation models, which has the advantage to be less technical and a strong focus on facilitating communication across disciplines [17]. Since the first ODD protocol was published in 2006 [18], there

were a number of updates to improve clarity and help users to replicate the model. One of the suggestions were to add codes along with the explanations to avoid the ambiguity of the explanation. Adding more information made openly available can certainly provide a better environment for the simulation community [19]. Thus, this paper adds code snippets wherever the code can support the explanation.

We outline this document to firstly introduce the model according to the structure of the ODD protocol [17], then describe the model's sensitivity experiment and calibration in the Sensitivity experiment and calibration section that was conducted for the original research ([link to preprint](#)). Things to try and notice covers the ways to use the model and reports possible artefacts and errors. Finally, Running the model on the HPC is a "how-to-use HPC" section using NetLogo as the main modelling platform, R as a compiler, and Unix codes to submit the work to the HPC.

TRAPSim was built in NetLogo 6.0.4 [20]. The data catalogue and codes are stored in the Harvard Dataverse [21], and the codes, bug reports, and wiki pages are available on [the GitHub Wiki](#).

Model purpose

The purpose of this model is to understand commuter's exposure to non-exhaust PM₁₀ emissions, and to make a preliminary estimate of their health effects.

This model illustrates the following patterns: (1) the fraction of population at risk by mode of transport and (2) the total numbers of traffic and pollution levels by road in a context that is representative of realistic conditions in the Seoul CBD.

Pattern 1: Population at risk by the mode of transport

- This pattern reflects how an individual's health might deteriorate from PM₁₀ exposure depending on the mode of transport they take, and how much time is spent under extreme PM₁₀ conditions. Health decline occurs when PM₁₀ exceeds the 100 µg/m³ level: a nominal health index is used, starting at 300, and individuals are labelled as "at risk" if the value drops below 100. The population at risk is a fraction of individuals with a health value less than 100 relative to the total population.

Pattern 2: Traffic load and pollution concentration

- This pattern emphasises the spatial variation of the pollution attempts to understand how one road is polluted relative to other roads, and how much traffic contributed to that. In other words, the commuting patterns and traffic flow generate some fraction of the emissions that impact people's health. This potentially allows a feedback between pollution and behaviour to be simulated. Although only a fraction of vehicles is represented, we can use sensitivity studies to test how important this might be to the realism of the output.

Data collection

This section describes the raw data collected for this study. The full description is presented in our website <https://ems-appendix.netlify.app/study-area-and-data-collection.html>.

Administrative boundary: The CBD area (16.7 km²) comprises two districts of Seoul, namely Jongno and Jung. Jongno has 8 sub-districts¹ and Jung has 7 sub-districts² (see Fig. 1).

Air Pollution: Hourly measured PM₁₀ was imported from two urban background stations, and two roadside stations (see Table 1). The background pollution data are used to interpolate the background areas, while the roadside stations are to calibrate the road emission levels. Assuming subway commuters travelling from distant origins, PM₁₀ was also collected from 23 background and 12 roadside stations within the city boundary.

Roads: The road layer is the most important component to simulate vehicle trips on the road network. Seoul CBD contains a mixture of two lanes, four lanes, eight lanes. The model, however, simplified the type of roads as one road (see Fig. 2B).

¹ Sajik, Cheongwoonhyoja, Samcheong, Gahoe, Jongno 1-4ga, Jongno 5-6ga, Ewha, and Hyewha.

² Sogong, Hoehyeon, Myung, Pil, Jangchoong, Gwanghee, and Euljiro.

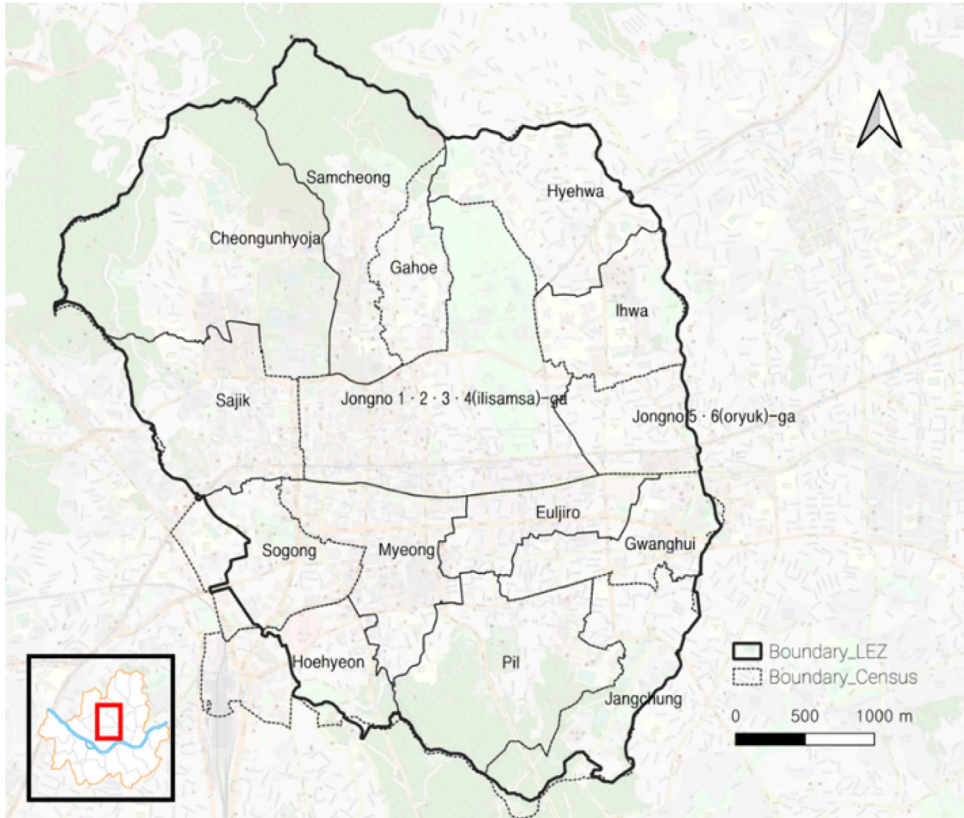


Fig. 1. Study boundary Seoul CBD.

Table 1
Monitoring Sites in Seoul CBD.

Type	Name	Location	Lat	Long
Background	Jongno	Jongno District Office	127.005005	37.5720356
Background	Jung	Seoul Arts Gallery	126.973720	37.5643110
Road	Jongno	Jongno Chapel	126.996538	37.5708913
Road	Seoul Station	Seoul Station	126.971042	37.5523812

Buildings: Buildings are used as the agent's office places which are brought in from OpenStreetMap. Once the model is loaded, all agents have a building ID allocated as their destination (see the blue polygons in Fig. 2A).

Hourly Traffic: Hourly traffic was provided from the Traffic Monitoring Department affiliated with Seoul Metropolitan Government. The data is used to feed non-resident cars into the model every minute.

Entities, state variables, and scales

- (A) **Resident vehicles:** 399 resident vehicles were sampled and imported in the model. The vehicles accounted for 1% of the total vehicles registered in each sub-district that mobilise within the district. Having tested different sample sizes ranging from 10 to 70%, the results from the sample size of 70% were not significantly different to the 10% sample size because the vehicles

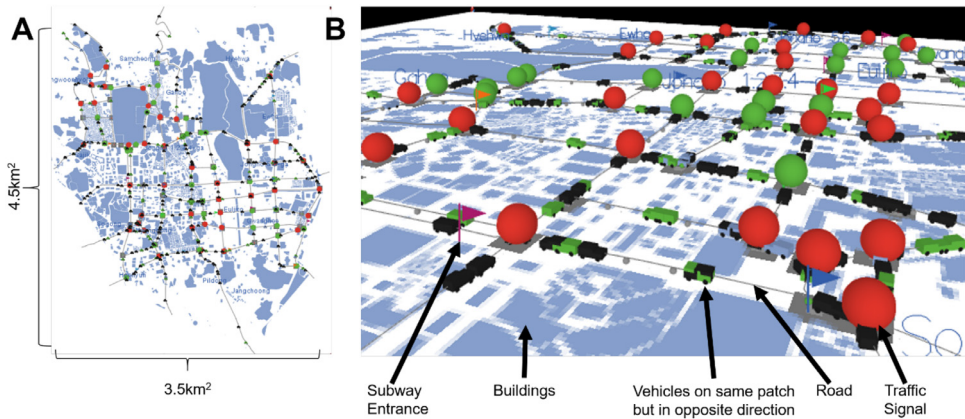


Fig. 2. Study area in 2D (A) and 3D (B).

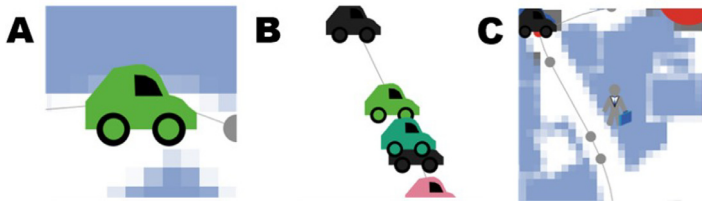


Fig. 3. Agent types used in the simulation. Type A is a resident vehicle (tied with a driver), Type B is a non-resident vehicle, and Type C is a pedestrian commuting by metro.

were queuing at the entrances ready to enter into the model environment. During each trip, the vehicles will keep some distance from other vehicles. Vehicles emit pollution to the local patches as they travel. During weekdays, trips are made along the shortest path and will not change throughout the simulation, while the weekend journeys are random. The codes are available on GitHub [to set-resident-cars](#) section.

- (B) **Non-Resident vehicles:** Unlike resident vehicles, non-resident vehicles do not have any specific navigation aims, but rather, play a role as pollution-generators inside the study domain. The vehicles will follow traffic signals and keep their distance from the vehicles in front but will be removed completely when they reach the end of the road (at the domain boundary, see Fig. 1). The randomness of travel directions is to simulate general movement during the vehicle's time in the CBD, in the absence of more detailed data. These vehicles are not present during the model settings but will appear when the model is executed. The codes are available on GitHub [to add-cars](#) section.
- (C) **Resident drivers** (tied with resident cars): The drivers are tied with cars but do not move nor appear on the interface (see Fig. 3A). This is to improve the model running speed and to prevent any computational errors between linking and unlinking cars from people. The drivers lose health when they are instantaneously exposed to the nominal PM_{10} threshold of $100 \mu\text{g}/\text{m}^3$.
- D) **Subway commuters:** To execute the model efficiently, the model populated 1932 persons (1% of the subway commuters) and gave each agent a destination point (building) within 20 patches from the subway entrance (see Fig. 3C). Once the location was assigned, the agents are asked to walk to their offices based on the local search algorithm (see details in Section). The codes are available on GitHub [to set-subway-commuters](#) section.

As mentioned in the previous section, this model consists of three types of mobile agents.

Table 2

Vehicles in the model have state variables related to their trip.

Code	Description	Example
fueltype	Type of fuel	"Gasoline"
origin	One of nodes set as origin	(node 1)
destination	Patch set as destination	(patch 40 40)
goal	The closest node from destined patch	(node 2)
path-work	List of nodes between home and work	[(node 1) (node 2)]
path-home	List of nodes between work and home	[(node 2) (node 1)]
nodes-remaining	Number of nodes from the list	24
myroad	List of roads between home and work	[(link 1 2) (link 2 3)]
current-link	Current road	(link 1 2)
district_name	Name of district	"Myungdong"
district_code	Code of district	1,102,055
link-counter	Cumulative counter to arrival	0
direction	Direction to work (1); to home(-1)	1
time-at-work	Minutes at work	524
random-car	Boolean of random / resident cars	True/false
parked	If the vehicle is parked	True/false

Table 3

People in the model have state variables related to their trip.

Code	Description	Example
origin	Subway node set as origin	(exit 20)
origin_patch	Patch of origin	(patch 20 20)
goal	Patch set as destination	(patch 40 40)
current	Current patch	(patch 30 30)
Heuristics	Distance between current and goal	0.11
arrived?	Whether they arrived to their workplace or not	T / F
time-at-work	Number of ticks spent after arrived?	480
direction	Direction to work (1), stay (0), and home (-1)	1/0/-1
arrive-tick	Ticks spent between exit and arrived?	39
Health	Nominal health level (starts from 300)	275
Hour	Hour when the agent arrives at the subway station	7
Minute	Minute when the agent arrives at the subway station	48

The state variables for the mobile agents, vehicles and people, are documented in [Tables 2](#) and [3](#). Resident vehicles have their origin and destination both in patches and nodes, where home and destination patches are considered as indoor spaces that require PM_{10} to be adjusted to the indoor level [\[22\]](#). The indoor/outdoor ratio is further explained in Section . path-work and path-home provide links between home and the destination node, and the positive direction guides vehicles to follow that of path-work, whereas the negative direction guides vehicles to follow that of path-home. Link-counter answers the question, "How many links before the vehicle stops?" Incrementing by 1, the link-counter will stop when it meets the nodes-remaining value, and then parked changes from FALSE to TRUE. After spending time-at-work for $480+\alpha$ minutes ($\alpha < 60$), the cars will start the journey back home.

For subway commuters, their tentative origin is their subway entrance. They walk to their goal patch using the shortest distance when the awareness scenarios are not activated. If the awareness scenario is activated, the individual moves to one of the three patches in the direction that has the lowest PM_{10} . Heuristics is the distance between the origin and goal, which will decrease as the individual moves towards the goal patch. If the agent reaches the goal, the arrival? the status will change from FALSE to TRUE. Note that some agents whose Heuristics is less than 1 and less than the walking speed will be stuck at that location. To avoid the error, the individuals whose Heuristics is less than 1 will automatically move to the office location and will convert their arrival? to TRUE and start working. As with resident drivers, time-at-work shows the remainder of the working time.

Table 4
Variables of a traffic signal .

Code	Description	Example
Dong_code	Code of admin (provided by Census)	1,102,055
Intersection	Boolean of road intersections	T/F
Auto?	A timer to change signals	15
Green-light?	Boolean of green lights	T/F

Table 5
Variables of a traffic signal .

Code	Description	Example
Line	Line number of Seoul Metro	1

For example, if an individual arrived at the working place indicating 500 min of `time-at-work`, the minutes will decrease every minute. As soon as the `time-at-work` indicates zero, the individual will return to its origin (home patch). For visual purposes, this model temporarily removes the workers whose `arrive-tick` is over 80 so that the display is less cluttered but makes them reappear after work (visit this [link for a short movie](#)).

Traffic signals are arbitrarily created at junctions that consist of three roads or more (see Table 4). More traffic lights are installed between road segments in the real world, but the intention here is to articulate the traffic and the resuspension of dust. The emitted PM_{10} will remain near the junctions when the vehicles are idling in front of the traffic signal. `Intersection` is a Boolean variable that determines whether they have three roads to become qualified. The duration of red and green signals are determined by a timer variable termed `Auto?`. The `Green-light?` is a Boolean variable that will allow vehicles to move when TRUE and stop when FALSE.

Subway entrances are set as an origin for subway commuters (see Table 5). There are 26 subway stations in the study area with line numbers 1, 2, 3, 4, and 5, and coded as `s_entrance`.

Regarding the **scale** of the model, TRAPSim is simulated on a two-dimensional, continuous space of the CBD of Seoul (16.7km²). The spatial extent of the real world converts to 155 horizontal and 192 vertical patches (i.e. grid-cells) in which each patch has spatial dimensions of 30 m by 30 m. The model runs at a *one-minute time step*, and variables are collectively updated until the simulation terminates. The total run time of the model is 127,740 min which is an equivalent of 3 months.

Process overview and scheduling: developing algorithms for autonomous trips

The simulation starts at 7:00am on January 1st, and ends at 23:59 on March 31st 2018 (see Fig. 4). The diagram shows the journey of vehicles and humans, and where the vehicles produce pollution (see Fig. 4 process a and Fig. 5), the agents who are exposed to over 100 μ g/m³ of ambient PM_{10} in the study area are expected to have their health decreased (see Fig. 4 process b and Fig. 6). Although the full journey to the CBD is not simulated in this study, subway commuters are assumed to be exposed to the ambient level PM_{10} between early morning and late in the evening even if they do not appear on the interface. The cumulative updates of the risk population and the PM_{10} concentration by roads are exported to a single spreadsheet at the end of the simulation.

Vehicles' routing algorithm

Vehicles are divided into two groups: (1) resident vehicles or (2) vehicles with random movement. The driver's health loss will be explained in the later section.

Vehicles in general:

- Both vehicle profiles maintain a safety distance of 1 patch (\approx 30m) between themselves and the vehicle in front. During the journey, vehicles will pollute and disperse non-exhaust PM_{10} , regardless of fuel types. Vehicles are asked to stop in front of the "Red" traffic signal. More information

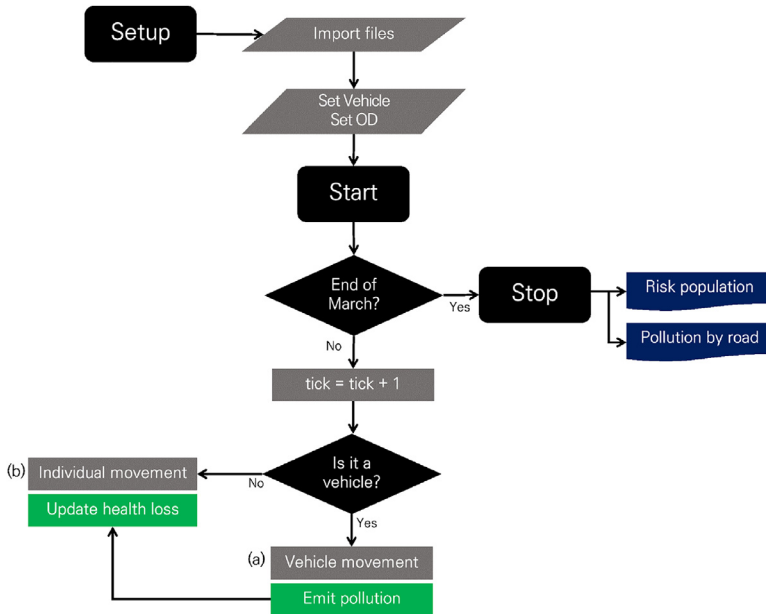


Fig. 4. A nested flow diagram describing the behaviour of agents and their landscape. During the setup period, buildings, pollution and roads are created. Registered vehicles are also created with their allocated destinations. The model starts at 6:00am on January 1st, and ends at 23:59 on March 31st 2018. Each tick will count as one minute. If the agent is a vehicle, then it follows the behaviour of a vehicle (see Fig. 5 for details); otherwise, it follows the behaviour of an individual (see Fig. 6 for details). If the simulation stops, then it will print the population at risk and pollution levels by road.

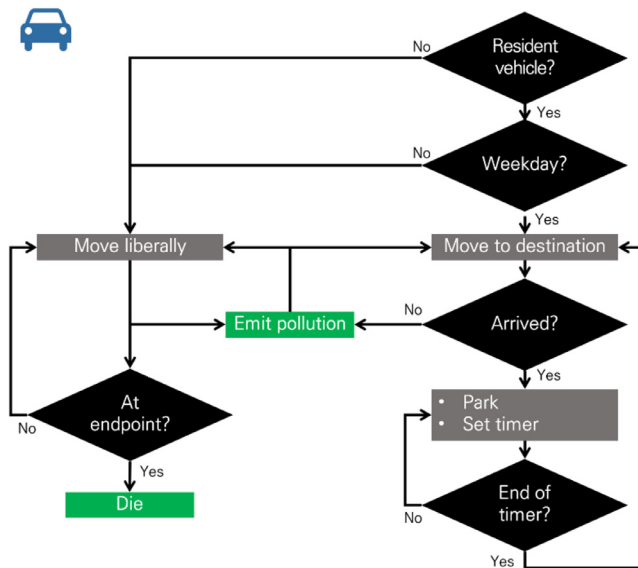


Fig. 5. Flow chart for resident and incoming vehicles. If the vehicle’s owner is a CBD resident, the vehicles will move to their assigned destination. The vehicle will emit pollution until it ends the journey. As the vehicle parks at the destination, the timer will start to countdown from $480+\alpha$ ($\alpha < 60$ min) to 0 mins and will head back home once the timer reaches zero. If the vehicle is non-resident, it will move generally and disperse non-exhaust pollution until it leaves the domain.

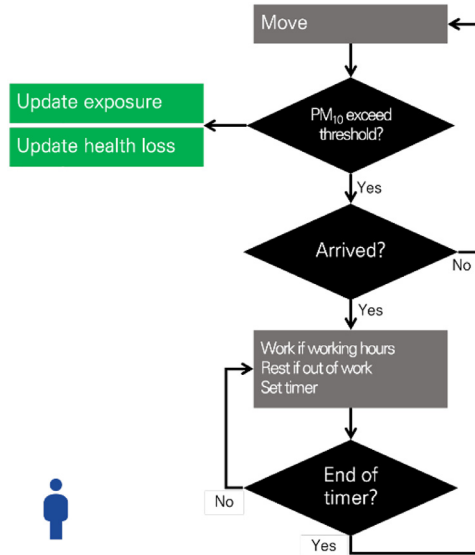


Fig. 6. Flow chart of a subway commuter's journey. While the person is walking, its health will degrade when the PM_{10} is above 100. If arrived, the person will stay until the timer ends and head back to the station.

```

to speed-up
  let max-speed 3 + random-float 2. //;Adding a range of randomness
  let min-speed .5 //;depending on the driver
  ask cars [. //; set different speed per vehicle but slow down when car is in-front
  set speed min-speed + random-float (max-speed - min-speed)
  let car-ahead one-of (cars-on patch-ahead 1) with [heading = [heading] of myself]
  ifelse car-ahead != nobody and not [parked] of car-ahead
    [ slow-down car-ahead] [set speed speed ]
end
  
```

Listing 1. NetLogo: Assigning the speed to each vehicle.

regarding the pathfinding algorithm and PM_{10} pollution will be introduced in the Sub-Model section.

- To make the vehicles move, the code asks each vehicle to accelerate up to 5 per tick, and slow down when a vehicle is heading in the same direction (see Listing 1).

Resident vehicles:

- Vehicles will move across road networks to their destination node, stop during office hours, and head back to the origin(node) again using the same route during weekdays, but will move away from the study area over the weekends for non-working activities, e.g. shopping, weekend journeys, or places to worship.
- During weekdays, each vehicle will stop the journey if the vehicle has arrived at its destination node. After its arrival, the state variable, timer, counts down from ≥ 480 min (up to 540 min). As soon as the timer reaches zero, the vehicle will head back home. Extra time from 0 to 59 min is given to all agents assuming agents walking to car parks or spending additional time to wrap up their work. Each vehicle has a driver whose health will decline if the PM_{10} inside the vehicle is over $100 \mu\text{g}/\text{m}^3$.
- See Listing 2 and Listing 3 for the NetLogo codes and annotations.

```

to travel [dist]
set current one-of nodes-here
;;; if "current" isn't assigned, then the vehicles will fly everywhere..
let dxnode distance to-node
ifelse dxnode > dist    ;;;; if the distance to the node you are heading to
                        ;;;; is greater than the node you left
[forward dist]        ;;;; go forward and reduce the 'dist'
[ move-to to-node     ;;;; move to the next node
  ifelse (direction = 1 and to-node != goal) or
         (direction = -1 and to-node != origin)
         ;;;;if the direction is 1(goal) but not heading to goal or
         ;;;;the direction is -1(home) but not heading home
  [ set link-counter link-counter + direction
    set current-link item link-counter myroad
    ;;;; add +1 to the direction on the vehicle's attribute
    ;;;; Each vehicle has a total no. of link-counter until arrival
    ifelse ([end1] of current-link = to-node)
            [set to-node [end2] of current-link]
            [set to-node [end1] of current-link]
    face to-node    ;;;; head to the next road
    travel dist - dxnode
    set-emission ]
  [ set speed 0 ;;;; if the vehicle has arrived at destination set speed 0
    park ]        ;;;; park vehicle
]
end

to park
;;; add time information
let hours item 1 table:get pm10-back (ticks + 1)
let minutes item 4 table:get pm10-back (ticks + 1)
let is-weekend? item 6 table:get pm10-back (ticks + 1)

;;; remove all the emission whilst parking
set parked true
set tyre-wear 0
set brake-wear 0
set surface-wear 0
set total-emission 0

;;; if the driver is at work (or finishes work)
ifelse to-node = goal or current = goal
[ set time-at-work time-at-work - 1
  if (time-at-work <= 0) [to-home-setup ]
]

[ if (hours = leave-home-hour and minutes = leave-home-mins) and is-weekend? = false
  [to-work-setup set time-at-work 540 + random 61] ; ; cars will go to work after 7am
]
end

```

Listing 2. NetLogo: Assigning trips for resident vehicles and park during work hours.

Vehicles with random movement:

- Vehicles are assumed to have come from the outside. These incoming vehicles make trips to any areas inside the CBD, generating vehicles from the hourly traffic data. Since the spatial extent is restricted to the CBD zone, this model made the outbound cars disappear at any endpoints of the road network. Since the model had a limited capacity of vehicles (~2500), the traffic count was further decomposed by 5% on the scenario, as well as 2.5, 10, and 20% on the sensitivity experiment. Note that if a vehicle checkpoint station had less than 1200 vehicles in an hour, then a 5% sample would not feed in any vehicles for that hour, but this was not a problem since not a large difference was seen in between the ratios - details are demonstrated in the Sensitivity section.
- See [Listing 4](#) for the NetLogo codes and annotation.

The basic code for the vehicle's movement was based on the Venice model (unpublished and eliminated, but the source code was shared until 2017 on [Professor Andrew Crooks' Webpage](#)).

```

ask cars [
  let is-weekend? item 6 table:get pm10-road (ticks + 1)
  let what-time? item 1 table:get pm10-road (ticks + 1)
  let hours item 1 table:get pm10-back (ticks + 1)
  let minutes item 4 table:get pm10-back (ticks + 1)
  let weekday? item 5 table:get pm10-back (ticks + 1)
  let travel-hours what-time? >= (8 + random 2) and what-time? < 22
  ifelse (random-car)[move speed]
  [ if is-weekend? = false and weekday? != "Mon" [travel speed]
    if is-weekend? = false and weekday? = "Mon" and hours = 6 and minutes = 59
    [to-work-setup set time-at-work 540 + random 61]
    if is-weekend? = false and weekday? = "Mon" and hours >= 7
    [travel speed]
    if is-weekend? = false and weekday? = "Mon" and hours >= 7 and minutes = 5 and parked [park]
    if (is-weekend? = true and awareness = "no" and travel-hours)
    [move speed] ;; move resident cars on weekends only when awareness off
    if (is-weekend? = true and awareness = "yes" and travel-hours and health >= 100)
    [move speed] ;; take rest when awareness on
    if (is-weekend? = true and awareness = "yes" and travel-hours and health < 100)
    [move-to origin to-work-setup] ;; take rest when awareness on
    if (is-weekend? = true and what-time? >= 23) [move-to origin to-work-setup] ]
    ;;; Flying cars may appear.
    ;;; They are just heading home without using the road links.
  ]
]

```

Listing 3. NetLogo: Asking resident cars to move randomly during the weekends. There is a technical difficulty that once the vehicle moves randomly there is no obvious way but to coerce the vehicles to move back to the origin. While the code requests the vehicles to return to the origin, the model looks as if there are flying cars.

```

to move [dist]
  set current one-of nodes-here ;;current node
  let dxnode distance to-node ;;measuring distance between current node and next node
  ifelse dxnode > dist [forward dist] [ ;;if the distance to the next node
    ;;is farther than my previous node
    let nextlinks [my-links] of to-node ;;follow the link to the next node
    ifelse count nextlinks = 1 ;;if the next node has no junctions
    [ set-next-car-link current-link to-node ] ;;then continue to move
    [ set-next-car-link one-of nextlinks with ;;if not, then take any direction
      [self != [current-link] of myself] to-node] ;;apart from the direction I came from
    move dist - dxnode ;;move function
    set-emission] ;; set emission
end

to set-next-car-link [way n] ;;the next link requires a link and a node
  set current-link way
  move-to n
  ifelse n = [end1] of way [set to-node [end2] of way] [set to-node [end1] of way]
  face to-node
end

```

Listing 4. NetLogo: Assigning random trips for non-resident vehicles.

Subway commuters' routing algorithm

When the simulation commences, the subway commuters are transported to the subway entrances at the hour and minute they have on their state variables. Once the agents arrive at their subway entrances, they walk to their destination buildings using the shortest distance regardless of the pollution levels. However, if the awareness scenario is activated, they will navigate following the lowest PM_{10} to their destinations.

The codes (see Listing 5) appeared to be similar to the mechanism of the vehicles, where origin, goal, and direction leads to the mobility of the agents and the duration of the working hours (time-at-work) hold the pedestrians at their working places.

```

to move-employees
ask employees with [goal != nobody][
  set current patch-here      ///: current location
  set Heuristic distance goal ///: calculating the distance
                                ///: between current and goal patches
  ifelse (patch-here != goal and direction = 1) [
    if (awareness = "no") [walk-to-work] ///:scenario off - find the shortest path
    if (awareness = "yes") [walk-with-awareness] ///:scenario on - move to one
                                                ///:of the three patches ahead
  ]
  [ work-hard
    if (time-at-work <= 0) [ set direction -1 head-home ]]]
end

///: Here is how to walk to the office
to walk-to-work
set color 103
set arrived? false
set hidden? false
;let choices neighbors with [pavement]
let nearest min-one-of neighbors [distance [goal] of myself ]
set time-at-work 540 + random 61
face nearest
fd (0.7 + random-float .3)
end

///: Here is the scenario that detects the lowest patch in the person's direction
to walk-with-awareness
set color 103
set arrived? false
set hidden? false
set time-at-work 540 + random 61
let choice1 neighbors with [pm10 < 100]
let nearest1 min-one-of choice1 [distance [goal] of myself ]
let choice2 min-n-of 3 neighbors [pm10]
let nearest2 min-one-of choice2 [distance [goal] of myself ]
ifelse nearest1 != nobody [move-to nearest1][move-to nearest2]
face [goal] of self
fd (0.7 + random-float .3)
if [Heuristic] of self <= 1 [move-to goal]
end

///: Returning home
to head-home
set Heuristic distance origin
let hours item 1 table:get pm10-back (ticks + 1)
let nearest min-one-of neighbors [distance [origin] of myself ]
set color brown + 2
set arrived? false
set hidden? false
ifelse current = myhome
[set time-at-work 0 fd 0 set arrive-tick 0]
[ face nearest fd (.6 + random-float .4) ]
if distance origin <= 1 [ move-to myhome ]
end

///: Office hours
to work-hard
let hours item 1 table:get pm10-back (ticks + 1)
let minutes item 4 table:get pm10-back (ticks + 1)
set arrived? true
if (ticks - arrive-tick) > 80 [set hidden? true]
set direction 0
set color grey
fd 0
ifelse time-at-work < 0 [set direction -1] [set time-at-work time-at-work - 1 ]
end

```

Listing 5. NetLogo: Asking subway commuters to move to and from the office. The “awareness” is an experimental scenario that coerces pedestrians to find the lowest air pollution trajectory when walking to the destination.

```

to set-signal-colours
  let hours item 1 table:get pm10-back (ticks + 1)
  ask nodes with [intersection? = true] [
    ifelse hours >= 2 and hours < 6
      [ set auto? 10 set color green set green-light? true ] ;; set traffic light colours
      [ set auto? auto? - 1
        if auto? >= 5 [set color green set green-light? true] ;; green
        if auto? < 5 [set color red set green-light? false] ;; red
        if auto? <= 0 [set auto? 5 + random 6]]
  ]
end

```

Listing 6. NetLogo: Setting function for traffic lights and giving rules to vehicles when to stop and go when encountering the traffic signal.

Traffic signals

When the simulation starts, each signal will be given a random number between 0 and 11 and will count down to 0 (see Listing 6). Between 5–10 is the red light that allows the vehicles to pass, and 0–4 stops the vehicles. The timer will reset to a random number again once the counter reaches 0. Note between 2am and 6am the traffic lights will go green as the night traffic reduces.

Subway entrances

As the simulation commences, the model chooses 4 out of 26 random stations to create commuters. It will be a returning point for commuters to travel home.

Design concepts

Basic principles

This exposure model was developed to illustrate how the population in the CBD zone can be exposed and possibly lose health in response to non-exhaust PM₁₀ emissions. There is extensive literature on traffic-related exposure, mainly associated with NO_x emissions, or with population exposure to NO_x [23–25], but not with non-exhaust emissions of particles. With increasing awareness that non-exhaust emissions are important [1,4], this study builds a health impact assessment model based on non-exhaust PM₁₀ emissions.

The rationale is that the particles generated by non-exhaust emissions (i.e. tyre and road wear particles) have been problematic for many years [22], but despite new vehicle models that comply with the environmental regulation, the percentage of non-exhaust emissions are increasing in many countries [4,26], and population health may be under a serious threat from instantaneous pollution rise. As experts raise concerns about the potential threat that the non-exhaust particles can bring to the local atmosphere, there should be a preparation for further regulations to non-exhaust particles in the near future [4].

As a starting point, the model asked resident and non-resident vehicles to generate and disperse PM₁₀ to the local atmosphere, namely on road and nearby pavements, while subway commuters and drivers are the susceptible individuals who are exposed to PM₁₀ emissions. On the other hand, the background PM₁₀ generated the value from the urban monitoring stations within the study domain. Each agent group has different behavioural patterns, which was explained in the previous section.

Emergence

The percentage of the population at risk (i.e. those with health under 100) emerges from a balance between exposure to a PM₁₀ threshold of 100 µg/m³ and recovery. In practice, the emergence can be an acute response to PM₁₀ exposure before the natural recovery begins to take effect. The emergence pattern will differ by which means of transport the individual is commuting with. This is because subway commuters are exposed to the ambient atmosphere during their walk from subway entrances to offices, while resident drivers spend most of their time indoors or in transit but have a higher chance of inhaling polluted air from road traffic. Despite the fact that extreme particulates were even

higher than other transport modes have been investigated [27,28], this study omitted the journey of subway commuters because information of the start and end stations are not provided in the OD data, which is very crucial for microscopic modelling.

To maintain the execution speed, we requested vehicles that were exposed to over 100 of PM_{10} to reduce their health and then pass the health information to the drivers tied with the vehicle. In addition, while the vehicle is parked, we did not set further health degradation as the driver is working in a building.

Adaptation

This study has two aspects of adaptation: pathfinding and health recovery. With regard to pathfinding, the subway commuters either walk along the shortest path when the awareness scenario is deactivated or find the best way to avoid high-polluted locations of PM_{10} exceeding over $100\mu\text{g}/\text{m}^3$ when the awareness scenario is activated (see Listing 5). Note that the scenario results are not presented in this paper.

If the awareness scenario is activated but the agent struggles to find a path below $100\mu\text{g}/\text{m}^3$, the agent will then find the lowest PM_{10} of the possible routes and move to that location. Resident drivers have their health deteriorate when the patch on the road is at least $144\mu\text{g}/\text{m}^3$ because the indoor-outdoor ratio between inside-vehicle and ambient air is 0.7. If the awareness scenario is activated, the driver will take a free trip during weekends - at the beginning of Saturday or Sunday - and only when the driver's nominal health is over 100. Conversely, if the awareness scenario is not activated, the drivers will take a trip regardless of their health. Both groups have their health recovered by the same amount at a nominal value of 10 out of 300 per timestep.

Sensing

Subway commuters are exposed to the PM_{10} at which they are located. If PM_{10} is over 100, the commuters will lose health according to the health loss equation (see Section - *Health Loss*). Subway commuters also use the shortest distance to their workplace when the awareness scenario is not activating or find the lowest value of PM_{10} amongst the front three patches in the direction they are moving. Additionally, everyone has its own time of arrival at the subway station. For instance, if the hour and minute variable of AGENT X is 8 and 12, AGENT X will appear at the station at 8:12 am. Both subway commuters and drivers have fixed working hours with a few minutes of extra time (up to one hour) to finish the daily work. The extra minutes differ every day.

The vehicles can sense one radius distance between the vehicles in front and behind and the traffic signals. As with subway commuters, drivers also have their destination time to work. After departure, the vehicles travel on the shortest route to their workplace.

Interaction

Interactions occur between the PM_{10} levels and the agent's health. That is, subway commuters who are exposed to over $100\mu\text{g}/\text{m}^3$ of ambient PM_{10} on the current patch will lose health, while the drivers will lose health according to the non-exhaust emissions from vehicles. For subway commuters, the NetLogo code is shown here (see Listings 11).

Another interaction occurs between vehicles and traffic signals. The vehicles stop in front of the red lights and start when the light changes to green (see Listing 6). Listing 7 is a code chunk that reduces the speed of the vehicles to 0 when the vehicles encounter a red traffic light and requests the vehicles to accelerate when the traffic light turns green.

Stochasticity

According to Grimm [17], stochasticity can happen to individuals, environments, and the parameter. Here, we describe the stochastic process of mobile agents (vehicles, subway commuters), immobile agents (traffic signals), and a health loss parameter.

Vehicles

- Vehicles have different origin and destination locations at every setup.
- Resident vehicles park for 480 min (ticks) with a random number of extra numbers (up to 60).

```

to meet-traffic-lights
ask cars [
  if any? nodes in-radius 2 with [green-light? = false][ set speed 0 ]
  if any? other nodes in-radius 1 with [green-light? = true]
    [set speed (speed + (.5 + random-float 1)) ] ]
end

```

Listing 7. NetLogo: Asking vehicles to stop in front of the traffic lights when the green light is FALSE and accelerate when the green light is TRUE. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

- A vehicle has a minimum speed of 0.5 patches per tick and a maximum speed of 3.5. In cases of queueing, the deceleration ranges between 0–0.7 and the acceleration ranges between 0–0.5.
- Non-resident vehicles are fed into the study area according to the traffic monitoring statistics; however, the direction and time spent are random. Since the model has a limited capacity of vehicle numbers, a randomly selected 0.1% of the vehicles will disappear every minute between 10pm and 4am and 0.25% during the rest of the hours. This is to assume that the vehicles have driven out of the CBD³. For example, if there were 2000 vehicles in the study area at 10am, five vehicles⁴ will disappear, and four vehicles in the next minute.
- Resident vehicles will select a random road to travel outside of the CBD.

Subway commuters

- In the setup process, subway commuters choose a random subway station, then assign one of the buildings within 10 radii as their workplace.

Traffic signals

- Each traffic light is given a random number of 0–11 (0–4 is red and 5–10 is green). The numbers automatically count down to 0 when the simulation is activated.
- When the counter reaches 0, the signal resets to a random number between 0–11. This will give full randomness to the traffic signals in the study area.

Health loss and recovery

- When a human agent is exposed to PM₁₀ over 100 µg/m³, the health loss equation subtracts the amount of health based on the factor α , where α ranges between 0 and 0.2. The parameters are tested for sensitivity, but only one parameter is used for scenario forecasting. See Section “Health Loss and Recovery” for details (p.232).
- Infiltration ratio (indoor/outdoor ratio) varies by the microenvironment and the time spent. This study estimates the infiltration from the ambient PM₁₀ of the current patch to indoor spaces such as houses, workplaces, and transits [22]. Ratios for each microenvironment compared to the outdoors are described as follows:
 - Houses: 0.2–0.7 [22,29]. The home patch discounts the ambient PM₁₀ by an index between 0.2 and 0.7.
 - Workplaces: 0.2 [22]. The work patch discounts the ambient PM₁₀ by 0.2.
 - Vehicle: 0.7 [22]. The patch where the vehicle is stopped decreases by 0.7 of the ambient PM₁₀.
- Health recovery is stochastic at the assumption that one can recover better than another. Any agent whose health is below 100 and remains at a stable place (home/office) will recover by $10 + \epsilon$ per minute (ϵ being between 10 and 20), until its health returns to the ‘non-risk’ state.

³ Having tried multiple ways to induce the non-resident vehicles outside, the most effective method was to eliminate a random set of vehicles.

⁴ $2000 \times .0025 = 5$.

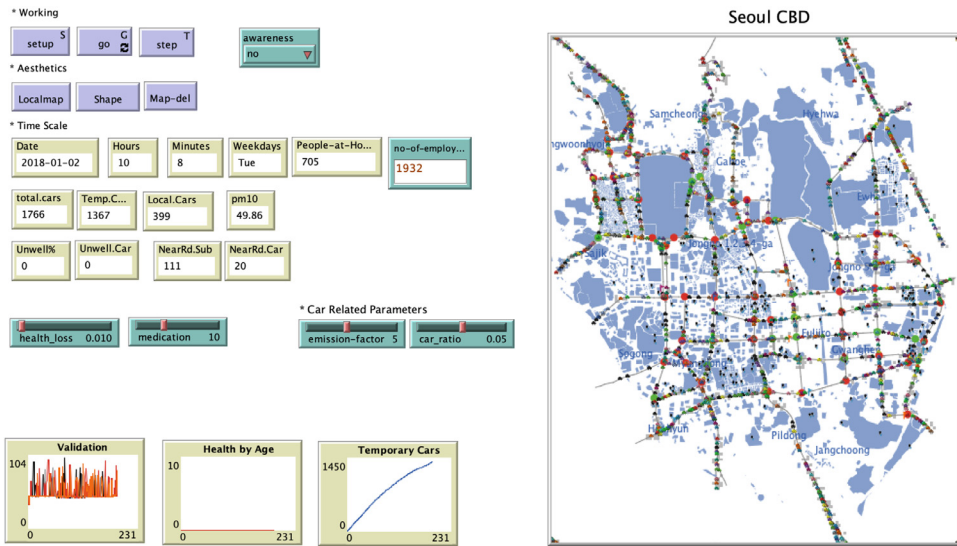


Fig. 7. NetLogo interface.

Observation

As this study examines the variability of health risk by demographic groups, the graphical output of the model shows the risk rate of subway commuters and resident drivers by time (see Fig. 7). Date, hour, and time are displayed on the interface to inform the current time. The average PM_{10} and a few road points are collectively monitored until the simulation ends. Subway commuters will not appear on weekends but will still be exposed to PM_{10} . Resident drivers, on the other hand, will travel in a random direction during weekends, but once they reach the end of the road they will stop until 10 pm and return to their origin. The returning procedure moves cars directly to their origin, which is intended to simplify the process.

Subway commuters and resident vehicles do not travel to work on weekends and national holidays. As such, the interface will look less busy on Saturdays, Sundays, Lunar New Year (February 15–17th 2018), and Independence Movement Day (March 1st 2018).

Sub-models

The pathfinding algorithm: A* and the local search algorithm

The pathfinding algorithm is a key function to assign the agent's origins and destinations. In doing so, the author initially used an Origin-Destination matrix to choose a fraction of the population from their origins and allocate it to their destinations. The fraction of the population that was allocated outside the study area was not considered for further measurement. Once the agents have their origins and destinations assigned in their attributes, the next task is to request each agent to assign the route. Amongst many methods, we used an A* algorithm for resident vehicles, and a Local Search Algorithm (LSA) for pedestrians.

For resident vehicles, the model used A* [30]. A* calculates the lowest cost distance from its origin and destination and traces the path where the cost is smaller. A* is one of the most popular pathfinding algorithms together with Dijkstra's from their vertices and segments, which in real life may represent road networks. This can be formulated as:

$$f(s) = g(s) + h(s), \quad (1)$$

where s is the state, $g(s)$ is the cost from the origin to the current s , and $h(s)$ is the heuristic estimation between the current state and destination, which adds up to the total cost at $f(s)$. In

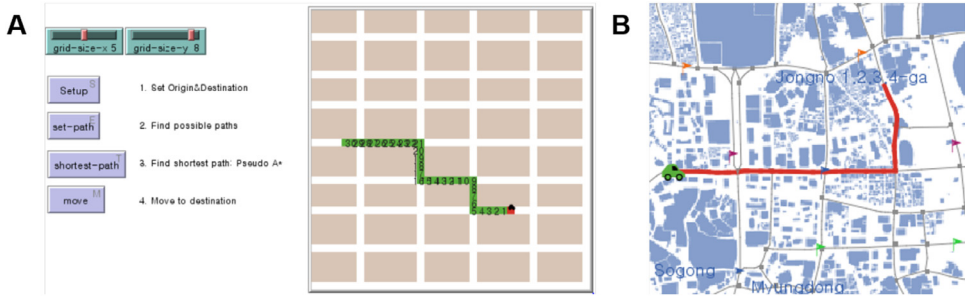


Fig. 8. (A) is a sample of an agent finding the shortest path from the origin (red patch) to its destination (light green patch), and(B) is the application of the shortest distance on link data. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

this context, an individual’s heuristic measurement is referred to as the shortest Euclidean distance to its destination. The A* algorithm is based upon Dijkstra’s algorithm but uses the heuristic framework to shorten the calculation time and optimise the shortest path.

The interface below is an example of an A* algorithm (see Fig. 8A,B). Fig. 8A indicates a gridded guideway between the vehicle’s origin (red) and destination (green). set-path finds all the steps from the origin patch to all possible steps inside the virtual world. The code will colour the road green and add a step label on the path. During the steps, the shortest-path traces the shortest grids, then allows the vehicle along the path with the move section. Fig. 8B is an A* algorithm based on the road network, which is embedded in the NetLogo network extension, *nw* [20] (see Listing 8).

For subway commuters’ pathfinding, a Local Search Algorithm (LSA) is used (see Fig. 9. Also refer to Listing 5 for codes). LSA is an algorithm where the agent knows the goal state and the distance from the optimised path (termed error of distance) and asks the algorithm to rewrite the path to minimise further errors, which makes it memory efficient. A* was replaced with LSA because the algorithm that was asked to find the lowest pollution patch between the current step and the final goal kept changing every step, which led to repetitive recalculation on every step, slowing the execution speed.

Amongst the searching functions of LSA, this study uses a “random-walk” or “hill-climbing search”, where the agent iteratively searches the maximum value (or minimum value depending on the setting) within the boundary until it reaches the target. However, the function has a major drawback as the searching terminates either when it reaches the local maximum instead of the global maximum, or there is a huge plateau which does not have a higher surrounding value. Nevertheless, this study applied this method because the commuters in the CBD normally do not have any issues in getting lost when they are heading to work and back home. This study also created another scenario, *Awareness*, that asks agents to take an alternative route to avoid high PM₁₀. Note that pedestrians penetrating the buildings is a downside of this method, but this model remained this artefact because not much difference was discovered from the exposure outcome between straight line walking and avoiding buildings due to the temporal time step of this model.

Non-exhaust emissions and dispersion

Recent studies from the UK and Europe equally documented the main sources of non-exhaust emissions such as tyre wear, brake wear, and road surface wear [4,31]. A few papers included resuspension as a fourth contributor, but this study articulates resuspension in the dispersion section below. Fig. 10 illustrates the non-exhaust emissions, dispersion, and dilution.

According to the European Environment Agency [31], the total of non-exhaust emissions is estimated with the following equation.

$$NEE_{total} = NEE_{Tyre} + NEE_{Brake} + NEE_{Road} \tag{2}$$

- NEE_{Total} : the total non-exhaust PM emissions
- NEE_{Tyre} : PM emissions from tyre wear

```

to set-path-node
ask cars [
  ;; Randomly choose a target node to walk to
  let target [goal] of self
  if target != nobody [
    set current one-of nodes-here
    ;; Define a path variable from the current node- take all but
    ;; the first item (as first item is current node)
    let path nobody
    ask links [ set weight link-length ]
    ask current [
      set path but-first nw:turtles-on-weighted-path-to target weight];; A-star

  ;; Indicate the end node
  ask last path [
    set color [color] of self
    set b_entrance? true

    set size 0.5 ]
    ;;; assign all the nodes that leads to the destination node
    let path-work0 lput path path-work
    set path-work item 0 path-work0
    set path-work fput origin path-work
    set to-node first path
    set nodes-remaining length [path-work] of self
    set current-link link [who] of [current] of self [who] of [to-node] of self
    face to-node
    let points n-values [nodes-remaining] of self [x -> ([nodes-remaining] of self - 1) - x]
    foreach points [ x ->
      set path-home lput item x [path-work] of self path-home]
    set path-home lput origin path-home]
end

to set-path-link
ask cars [
  let vertex0 path-work ;;finding the nodes for destination
  let vertex1 remove-item 0 path-work ;;remove origin point
  let imsi0 [] ;;set temporary list
  foreach vertex0 [ x ->
    let element [who] of x
    set imsi0 lput element imsi0 ] ;;make a list for work direction
  let imsi00 bl imsi0
  let imsi1 []
  foreach vertex1 [ x ->
    let element [who] of x
    set imsi1 lput element imsi1 ] ;;make a list for home direction
  set myroad (map link imsi00 imsi1)
end

```

Listing 8. NetLogo: A* algorithm coded in NetLogo.

- NEE_{Brake} : PM emissions from brake wear
- NEE_{Road} : PM emissions due to road abrasion

Each component will be investigated in the following sections.

Tyre Wear

$$NEE_{\text{Tyre}} = \sum_{i=1}^n N_j \times M_j \times EF_{\text{Tyre},j} \times F_{s,i} \times S(V) \quad (3)$$

- NEE_{Tyre} : Total emission for the defined time and spatial boundary (g/km)
- N_j : Number of vehicles in category j within the defined spatial boundary
- M_j : Mileage (km) driven by each vehicle in category j during the defined time (**not used**)
- $EF_{\text{Tyre},j}$: TSP mass emission factor for vehicles in category j (g/km)
- $F_{s,i}$: mass fraction of Particles that can be attributed to particle size class i
- $S(V)$: Correction factor for a mean vehicle travelling a speed V

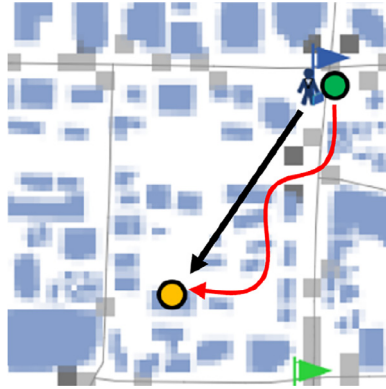


Fig. 9. The person next to the starting point (green) walks towards the goal point (yellow) following the shortest path, which is a straight line. Here, the agent decides to move closer to the goal point, but the route will be created at every step. Hardly any difference was identified between the straight line (black) and the avoid-building mode (red). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

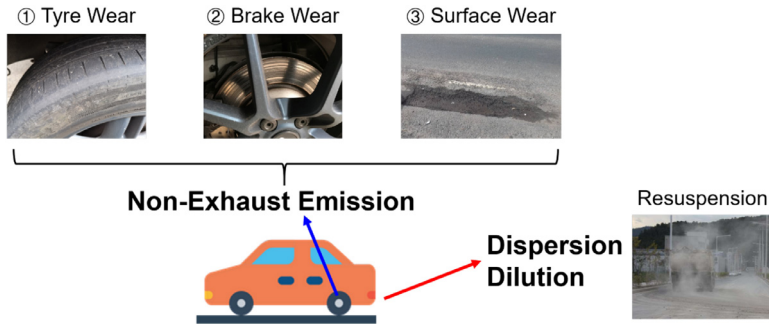


Fig. 10. Graphical explanation of non-exhaust emissions, dispersion, and dilution.

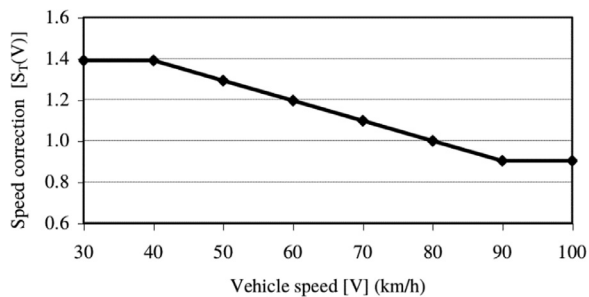


Fig. 11. Speed: tyre wear.

As this equation was designed to measure the bulk emissions from a number of vehicles (e.g. 20 g/km from 10 vehicles in a 5 km trip between 10:00 and 15:00), it is not appropriate to measure the emissions of hundreds of vehicles that have separate journeys. To find a solution, this study manipulates N_j at an appropriate number based on sensitivity analysis, converts emission levels from g/km to $\mu\text{g}/30\text{ m}$ (equal to a size of one patch in the simulation), and spatial and temporal units at 30m and on a minute by minute basis.

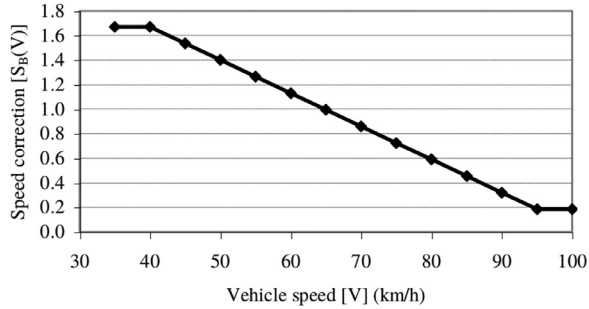


Fig. 12. Speed: Brake wear.

Table 6

TSP (Total Suspended Particles) emission factors for source category road vehicle tyre wear [31].

Vehicle class (j)	TSP emission factor (g/km)	Uncertainty range
Two-wheel vehicles	0.0046	0.0042–0.0053
Passenger cars	0.0107	0.0067–0.0162
Light-duty trucks	0.0169	0.0088–0.0217
Heavy-duty vehicles	Separate Equation	0.0227–0.0898

Table 7

Size distribution of tyre wear particles [31].

Particle size class (i)	Mass Fraction of TSP
TSP	1
PM ₁₀	0.6
PM _{2.5}	0.42
PM ₁	0.06
PM _{0.1}	0.048

Table 8

Speed Correction [31].

Velocity (km/h)	Factors (V)
V < 40	1.39
40 ≤ V ≤ 90	-0.00974 * V + 1.78
V > 90	0.902

For example, one passenger car (j) has an emission factor of 0.0107 (.0067–.0162) (g/km) (see Table 8), and to get an estimate of PM₁₀, the size distribution F_{s,i} converts the TSP estimate to PM₁₀ multiplying by a fraction of 0.6 (see Table 8). This can result in 32.1 µg/m³ per patch with an uncertainty range of 20.1–48.6.

In terms of vehicle speed, EEA sets the parameter V at 1.39 below 40 km/h, and declining effect of (-0.00974 * V + 1.78) between 40–90km/h. It assumes that frequent brakes and accelerations are expected below 40km/h but less as the vehicle speeds up.

Brake Wear

The equation for brake wear is the same as tyre wear, and has only a few differences in parameters.

$$NEE_{Brake} = \sum_{i=1}^n N_j \times M_j \times EF_{Brake,j} \times F_{s,i} \times S(V) \tag{4}$$

- NEE_{Brake}: Total emission for the defined time and spatial boundary (g/km)

Table 9

TSP (Total Suspended Particles) emission factors for source category road vehicle brake wear [31]. Here, this study only considers passenger cars.

Vehicle class (j)	TSP emission factor (g/km)	Uncertainty range
Two-wheeled vehicles	0.0037	0.0022–0.0050
Passenger cars	0.0075	0.0044–0.0010
Light-duty trucks	0.0117	0.0088–0.0145
Heavy-duty vehicles	Separate equation	0.0235–0.0420

- N_j : Number of vehicles in category j within the defined spatial boundary
- M_j : Mileage (km) driven by each vehicle in category j during the defined time (not used)
- $EF_{Br,j}$: TSP mass emission factor from road wear for vehicles in category j (g/km)
- $F_{s,i}$: mass fraction of Particles that can be attributed to particle size class i
- $S(V)$: Correction factor for a mean vehicle travelling speed V

As mentioned in the Tyre Wear section, emission factors for passenger cars must fit a unit set in the virtual environment. Thus, the $EF_{Br,j}$ value of 0.0075 (g/km) converts to 21.5 ($\mu\text{g}/\text{patch}$). The size distribution of PM_{10} is 0.98. The brake wear, particularly from the linings, are worn out quickly when the driver accelerates and decelerates frequently, and this tends to happen when the traffic volume is high.

Surface Wear (i.e. Road Abrasion)

Road surface wear is caused by the appearance of wheel marks when the vehicle passes over the road or parts of the road are destroyed by heavy vehicles. The formula is as follows.

$$NEE_{Surface} = \sum_{i=1}^n N_j \times M_j \times EF_{SW,j} \quad (5)$$

- $NEE_{Surface}$: Total emissions for the defined time and spatial boundary (g/km)
- N_j : Number of vehicles in category j within the defined spatial boundary
- M_j : Mileage (km) driven by each vehicle in category j during the defined time (**not used**)
- $EF_{SW,j}$ = TSP mass emission factor from surface wear for vehicles in category j (g/km)
- $F_{s,i}$ = Mass fraction of TSP that can be attributed to particle size class i

Dispersion and Dilution

There are many dispersion models applicable for exhaust emissions, but according to early research [23,32], many things related to non-exhaust dispersion remain unknown. The University of California, Riverside (UCR) team is conducting an on-going project to understand the severity of non-exhaust emissions at nearer roads and is currently testing non-exhaust parameters in their existing dispersion model⁵. In line with the UCR project, this study also attempts to disperse pollution with a spread function, *in-cone* in NetLogo, as a surrogate of dust resuspension.

Dilution with non-combustible dust varies by meteorological or ventilation conditions. Less road dust would be generated on rainy days due to the additional weight that is deposited by the particle substances on the ground, and during night hours when there is less traffic. Cities like Seoul have employed water spraying trucks to spray moisture on the roads on dry days, which adheres the particles on the ground as well as keeps the resuspension low as possible. Since this study does not consider humidity or rain effects, the model will use the case from Nikolova [33], where it takes 110 seconds to dilute completely. In NetLogo, this is assigned as three random ticks - ranging between 0 and 2 minutes. This study further investigates the sensitivity of road PM_{10} by controlling both dispersion ranges and the extension of dilution.

Application Inside the Simulation

It is worth mentioning that the units change inside the *in silico* environment. Since one patch is equivalent to 30 metres and one car represents 10 vehicles, a car moving from one patch to the next means 10 cars moving 30 metres. The vehicle speed inside the simulation is assigned in Table 14.

⁵ <https://ww2.arb.ca.gov/resources/documents/brake-tire-wear-emissions>

Table 10
Size distribution of brake wear particles [31].

Particle size class (i)	Mass fraction of TSP
TSP	1
PM ₁₀	0.98
PM _{2.5}	0.39
PM ₁	0.1
PM _{0.1}	0.08

Table 11
Speed Correction [31].

Velocity (km/h)	Factors (V)
V < 40	1.67
40 ≤ V ≤ 90	-0.0270 * V + 2.75
V > 90	0.185

Table 12
TSP emission factors from road surface wear [31]. Here, this study only considers passenger cars.

Vehicle class (j)	TSP emission factor (g/km)
Two-wheeled vehicles	0.006
Passenger cars	0.015
Light-duty trucks	0.015
Heavy-duty vehicles	0.076

Table 13
Size distribution of road surface wear particles [31].

Particle size class (i)	Mass fraction (FR _i) of TSP
TSP	1
PM ₁₀	0.5
PM _{2.5}	0.27

Table 14
Conversion of Vehicle Speed in NetLogo.

Original	Simulation
5km/h	0.25
10km/h	0.5
20km/h	1
40km/h	2

In previous studies, the emissions are calculated by g/km based on the total distance of which the car has travelled [34,35]. Smit [36] argued that the atmospheric pollution is combined with emissions, humidity, wind, temperature, and other uncertain factors, and therefore the calibration process is normally tested in places where there are fewer confounding variables, e.g. tunnels. Calibration with observational values can be inaccurate, but more than 15 studies have chosen this method due to restricted conditions [36].

For example, if a car travels over a patch, it releases 10µg/m³ of tyre wear, 7µg/m³ of brake wear, 10µg/m³ of surface wear, and 3µg/m³ of resuspension. It will also have a dilution at 5µg. Thus, the total PM₁₀ concentration would be the background PM₁₀ + 25µg/m³ (Tyre + Brake + Surface + Resuspension - Dilution).

The codes for NEE generation and dispersion are introduced in Listing 9.

```

to set-emission
  ;;; Tyre wear
  let velocity-tyre 0
  let EF1# 0
  let random_no# random 2
  ifelse speed < 2 [set velocity-tyre 1.39]
  [set velocity-tyre (-.00974 * speed) + 1.78]
  ifelse random_no# = 0 [set EF1# (0.0107 - random-float .004)]
  [set EF1# (0.0107 + random-float .006)]
  set tyre-wear (emission-factor * EF1# * .6 * velocity-tyre) * 333
  ;;; Brake wear
  let velocity-brake 0
  let EF2# 0
  let random_no1# random 2
  ifelse speed < 2 [set velocity-brake 1.67]
  [set velocity-brake (-.0270 * speed) + 2.75]
  ifelse random_no1# = 0 [set EF2# (0.0067 - random-float .004)]
  [set EF2# (0.0067 + random-float .006)]
  set brake-wear (emission-factor * EF2# * .6 * velocity-brake) * 333
  ;;; Surface wear
  set surface-wear (emission-factor * .015 * .5) * 333
  ;;; Total Emission
  let dilution# (0.5 + random-float .50) ;; wind speed, turbulence, traffic intensity
  set total-emission (tyre-wear + brake-wear + surface-wear) * dilution#
end

```

Listing 9. NetLogo: Assigning non-exhaust emission from each vehicle.

Health loss and recovery

The agent's health will decline on the assumption that it encounters over $100\mu\text{g}/\text{m}^3$ at which they are currently located. According to Shin and Bithell [37], H_{max} denotes an agent's health status at the beginning. $H(t)$ is the current health status. H_{recov} is the recovery rate where the agent's health is recovered to a certain extent when situated indoors. α is an arbitrary weight that picks up a random uniform distribution between 0 and a certain value. As the certain value affects the tipping point of the at-risk population, the parameter is further tested in the sensitivity analysis section. The codes to simulate the health loss parameter is introduced in Listing 11.

$$\text{If } PM_{10} \geq 100, \quad \frac{dH}{dT} = -\alpha(H_{\text{max}} - H(t)) + H_{\text{recov}} \quad (6)$$

While the equation above is equivalent to that of Shin [37], there are several measurements from which the application differs. First, the infiltration ratio, often termed as the I/O ratio, is used to estimate indoor exposure of individual agents. Infiltration ratio is applied to studies when only one has information about outdoor air pollution but less about indoor air pollution. Although the numbers seem quite simple, the ratio results from the consideration of the air exchange rate, windows opening, and type of housing. A few studies that used the I/O ratio also indicate that the ratio can vary by season (winter, summer) or types of microenvironments (classroom, house, office). This study chose the ratio from two studies, where Kreider et al. [22] took into account the I/O ratio from non-exhaust emissions, and Leung [29] who reviewed a wide variety of households to get a parameter (see Table 15). The outdoor PM_{10} is assigned at 1, transit at 0.7, and indoors (including house and office spaces) at 0.2–0.7.

With the equation and infiltration ratio, the health loss for both subway commuters and resident drivers is applied under the same conditions. However, the difference would be their mode of transport and behaviours during weekends.

Commuters get an equal chance of exposure to the PM_{10} threshold, but the degree of health loss will depend on how much time is spent outside when the PM_{10} is over $100\mu\text{g}/\text{m}^3$, and how long the distance is between the subway entrance and the agents office. Moreover, the agents whose office is adjacent to roads might lose more health because the pollution generated from the roadside can affect

Table 15
Indoor-outdoor ratio of ambient PM₁₀.

Type	Ratio
Outdoor	1
Transit	0.7
Indoors (house, office)	0.2-0.7

Table 16
Scenario description.

Scenario	Parameters	Description
Restrict inbound vehicles	BAU	Keep the number of inbound vehicles by minutes
	50%	Reduce inbound vehicles by 50%
	90%	Reduce inbound vehicles by 90%
Awareness	Yes	The agent finds one of the three lowest patches during the direction
	No	The agent walks on the shortest route

the indoor pollution, e.g. opening and closing windows [22][22]. Subway commuters spend their time exposed to ambient air pollution between the subway station and office. It is also considered that when the commuters travel out of the study area, they take more than an hour to arrive home⁶. Assuming the commuters stay at home between 11pm–6am, the commuter will be exposed to 0.2 times the ambient PM₁₀ of the given patch.

Resident drivers are mostly exposed to 0.7 times the ambient PM₁₀ of the patch in transits and 0.2–0.7 times of that of PM₁₀ when the vehicle is parked at the house or office. The vehicles are expected to be frequently exposed to high PM₁₀ due to the substantial load of PM₁₀ generated by road traffic.

The health loss function also applies to the pedestrians during night hours 23:00–06:00 (see Listing 10 `ask patch max-pxcor max-pycor` and `ask patches with [is-endpoint?]`). Although there are hardly any particulates assumed to float in the air during night hours, we assign this area to make sure that the I/O ratio is applied wherever an agent is regarded to stay indoors. As the codes are tied with the emission function there are no code headings (starting with `to`) or endings (ending with `end`).

Health recovery activates when the agent's health is below 100 and the agent is located at an indoor space (see Listing 11. For a subway commuter, this will be when they are at home or office, while drivers recover when the car is parked. The recovery rate is given an arbitrary number of 10 by each minute but stops working when the health of an individual goes above 100.

Scenario forecasting

This section outlines how vehicle prohibition can effectively improve air quality in Seoul CBD, as well as how people's information and awareness can prevent exposure to air pollution. The scenario was designed based on the 'Green Transport Scheme' initiated in December 2019, and thus it attempts to help measure the effectiveness of implementation that is already in place.

The Green Transport Scheme aims to improve air quality in Seoul by restricting high-emission vehicles from entering the CBD area. The municipal government restricts Grade 5 vehicles, mostly diesel cars, between 06:00 and 21:00, and violators are fined 100USD. This study looks at how the effects of non-exhaust emissions resulted from barring vehicle entry and illustrated how people's health might improve from the scenario.

In Table 16, **the first scenario is to restrict extra inbound vehicles**. It measures how PM₁₀ will improve if vehicles are restricted by 50% or 90%. **The second scenario compares the outcome of the population at risk depending on the awareness of individuals to extreme PM₁₀**. When the

⁶ Joong-ang daily article, March 7th 2019, "South Korea's office workers spend 103 min on average to get to work"


```

to pollute
ask cars with [not parked and speed > 0][
  let polluting one-of [link-neighbors] of to-node
  let mycar self
  let pm10# (item 2 table:get pm10-back (ticks + 1)) +
  random-float (item 3 table:get pm10-back (ticks + 1))

  ask patches in-cone 2.5 90 [
    set pcolor grey + 2
    set pm10 (pm10# * .73) + [total-emission] of mycar ]]
  ;; Removing 23% PM10 from vehicle contribution

ask patches with [pcolor != (grey + 2) and is-research-area? = true][
  set pm10 (item 2 table:get pm10-back (ticks + 1)) +
  random-float (item 3 table:get pm10-back (ticks + 1))]

ask cars with [owner != 0][
  ask origin [set pm10_indoor ([pm10] of patch-here) * (.2 + random-float .5)]]

;; set buildings
ask buildings [
  ask patch-here [ set pm10_indoor pm10 * (.2 + random-float .5) ]]

;; in the left bottom corner
ask patch max-pxcor max-pycor
  [ let night item 1 table:get pm10-others (ticks + 1)
    ifelse night > 23 or night < 6
    [ set pm10 ((item 2 table:get pm10-others (ticks + 1)) +
      random-float (item 3 table:get pm10-others (ticks + 1))) * .25 ]
    [ set pm10 (item 2 table:get pm10-others (ticks + 1)) +
      random-float (item 3 table:get pm10-others (ticks + 1)) * (1 - random-float .75) ]]]

;; for resident vehicles going outside
ask patches with [is-endpoint?]
  [ let night item 1 table:get pm10-others (ticks + 1)
    ifelse night > 23 or night < 6
    [ set pm10 ((item 2 table:get pm10-others (ticks + 1)) +
      random-float (item 3 table:get pm10-others (ticks + 1))) * (.25 + random-float .5) ]
    [ set pm10 (item 2 table:get pm10-others (ticks + 1)) +
      random-float (item 3 table:get pm10-others (ticks + 1)) ]]]
end

to fadeout
ask patches with [pcolor = (grey + 2)][
  ifelse countdown <= 0
  [ set pcolor white
    set countdown random 3 ]
  [ set countdown countdown - 1 ]]
end

```

Listing 10. NetLogo: Generation and dilution of PM₁₀.

awareness scenario is not activated, the subway commuters will walk on the shortest distance to their destination and the resident drivers will take free trips within or outside the CBD over the weekend regardless of their health. When the awareness scenario is activated, the subway commuters either walk on the path that does not exceed $100\mu\text{g}/\text{m}^3$ of PM₁₀ or on the lowest value of three patches in front of their path when all the surrounding patches exceed $100\mu\text{g}/\text{m}^3$. The drivers below the nominal health of 100 will not take a journey. Both scenarios are implemented in combination. The codes and the output figures are available in the GitHub repository: [Scenario.R](#).

Sensitivity experiment and calibration

This section experiments with the sensitivity of selected parameters and calibrates the modelled pollution outcomes with PM₁₀ observations. Each parameter was analysed from an average of 20 iterations that reduced possible stochastic effects. Rather than using the term ‘‘Sensitivity Analysis’’,

```

to health-loss
  ask employees [
    if ([pm10] of patch-here >= 100 and arrived? = false)
      [set health health - ((random-float health_loss) * (310 - health))]
    if ([pm10_indoor] of patch-here != nobody and [pm10_indoor] of
      patch-here >= 100 and arrived? = true)
      [set health health - ((random-float health_loss) * (310 - health))]
    if health < 100 [set unwell_history true ]
    if health < 0 [set health 0]

  ask cars with [owner != 0][
    if not parked and ([pm10] of patch-here * 0.7) >= 100
      [set health health - ((random-float health_loss) * (310 - health)) ]
    if parked and [pm10_indoor] of destination >= 100
      [set health health - ((random-float health_loss) * (310 - health)) ]
    if health < 100 [set unwell_history true ]
    if health < 0 [set health 0]

  ask drivers with [my_car != nobody][
    if [parked] of my_car = false [ set health [health] of my_car ]
  end

to health-recovery
  ask employees with [current = myhome or current = goal and health < 250]
  [ if health >= 0 and health < 100
    [set health health + (medication + random-float medication) ]]
  ask cars with [owner != 0 and parked and health >= 0 and health < 250]
  [ if health >= 0 and health < 100
    [set health health + (medication + random-float medication) ]]
  ask drivers with [my_car != nobody][ set health [health] of my_car ]
end

```

Listing 11. NetLogo: Health loss of individuals when encountering PM₁₀ of 100 µg/m³.

Table 17

Summary table of parameters used for the sensitivity experiment.

Parameter	Description	Baseline	Min	Max
Emission	Non-emission weights per vehicle	5	1	20
Dispersion	Range of emission	60	45	90
Dilution	Time until the emission dilutes	3	3	20
Car sampling	Rate of incoming cars	5%	2.5%	20%
Health loss	Parameter (α) from the health loss equation	0.1	0.03	0.2
Walking speed	Walking speed of subway commuters	0.6-1.0	0.2-0.4	1.6-1.8

we used the term “Sensitivity Experiment”. This is because each unit parameter was selected over a large range.

This study used the one-factor-at-a-time (OFAT) method to examine the sensitivity for each of the parameters. Having tested six parameters over a selected period (i.e. days when the background PM₁₀ exceeded 100µg/m³), there were no noticeable interaction effects discovered in the outcome.

This study selected six parameters (see also in Table 17):

- (non-) Emission: 1, 5 (base), 10, 20
- Dispersion: 45°, 60° (base), 90°
- Dilution: 3 ticks (base), 5–10 ticks, 10–20 ticks
- Car sampling: 2.5%, 5% (base), 10%, 20%
- Health loss: 0.03, 0.05, 0.1 (base), 0.15, 0.2
- Walking speed: 1) 0.2-0.4, 2) 0.4-0.7, 3) 0.6-1.0 (base), and 3) 1.6-1.8 patches per minute

All of the codes and the figures for the sensitivity analysis are available on our GitHub [Sensitivity subfolder](#).

Table 18

PM₁₀ concentrations in five CBD roads based on emission factors of 1, 5, 10, and 20 (Unit: $\mu\text{g}/\text{mm}^3$).

Date	Emission factor	Jongno	Sejong	Yulgok	Samil	Pirun
Overall	1	43.4	43.2	42.8	42.8	42.9
	5	60.0	60.1	62.0	61.6	61.7
	10	81.4	81.2	85.6	85.3	85.2
	20	123.3	122.6	134.1	132.7	133.4
Jan 8th	1	49.2	48.6	48.1	47.8	48.4
	5	65.5	66.7	67.8	67.9	66.9
	10	85.3	86.4	90.3	91.9	90.5
Jan 15th	20	129.6	115.3	141.1	150.1	128.7
	1	58.1	57.6	57.1	56.9	57.5
	5	75.0	75.2	77.0	76.9	75.1
Jan 22nd	10	93.8	104.3	100.8	101.6	97.1
	20	136.1	133.3	147.8	154.4	148.6
	1	37.7	37.4	37.1	37.0	37.2
Jan 22nd	5	53.9	52.5	55.1	57.3	55.4
	10	75.9	78.4	78.5	85.0	76.9
	20	114.6	117.6	127.6	127.6	124.3

PM₁₀ Levels by emission factors

This paper tested how the variability of non-emission factors affect the levels of PM₁₀ in the study area. Each line represents 5 sample points of Jongno, Sejong, Yulgok, Samil, and Pirun roads. The baseline emission factor was 5, and the alternates were 1, 10, and 20. Table 18 and Fig. 13 show how PM₁₀ results from the emission factors of vehicle agents. As a reminder, the emission parameter is the number of cars (N) represented in the equations of Section . It can be interpreted as how many cars have polluted PM₁₀ on this patch.

The levels of PM₁₀ increased linearly as the emission factors increased. Table 18 indicated that the mean PM₁₀ of Jongno was 43.4 $\mu\text{g}/\text{m}^3$, 60 $\mu\text{g}/\text{m}^3$, 81.4 $\mu\text{g}/\text{m}^3$, and 123 $\mu\text{g}/\text{m}^3$ in emission factors 1, 5, 10, and 20, respectively. The difference between each factor was 16.6 $\mu\text{g}/\text{m}^3$, 21.4 $\mu\text{g}/\text{m}^3$, and 41.6 $\mu\text{g}/\text{m}^3$, which increased proportionally as the factors increased. This linear increase was not only seen in the mean figure but also seen on any of the dates, including the peak value on January the 20th where the levels sat near 150 $\mu\text{g}/\text{m}^3$ in factor 5, but showed an increase to around 200 $\mu\text{g}/\text{m}^3$ and 250 $\mu\text{g}/\text{m}^3$ from factors 10 and 20 (see details in Fig. 13).

PM₁₀ between roads varied greatly when the emission parameter was high. Although the model did not give any direction to the vehicles nor the hierarchy of roads, PM₁₀ levels varied by 12 $\mu\text{g}/\text{mm}^3$ in Emission 20, where the lowest was 122.6 $\mu\text{g}/\text{mm}^3$ at Sejong and the highest was 134.1 $\mu\text{g}/\text{mm}^3$ at Yulgok (see Table 18). This implies that although the number of road lanes was not specified, the high parameter value can measure the variability of PM₁₀ by roads.

Dispersion and dilution

This section examines the sensitivity of dispersion and dilution parameters that affect roadside PM₁₀. The variables are conceptualised in Fig. 14. By default, each vehicle disperses non-exhaust PM₁₀ onto the neighbouring patches by an angle of 60° which dilute in 0–3 min. Having controlled the duration of dilution (< 3 ticks), the first experiment simulated the range of dispersion at 45° and 90°. Then, controlling the dispersion to 60°, the next experiment simulated the dilution process by 5+ β ticks (0< β <5) and 10+ θ ticks (0< θ <10).

Results showed that dispersion range displayed less sensitivity on roadside PM₁₀, except for Jongno, where the difference of cone width between 45° and 90° was around 3 $\mu\text{g}/\text{m}^3$ in Emission 5 and Emission 10, and further increased to 14 $\mu\text{g}/\text{m}^3$ in Emission 20 (see Table 19). This implies that the range of dispersion might not be sensitive to the PM₁₀ on-roads, such as Sejong and Pirun stations,

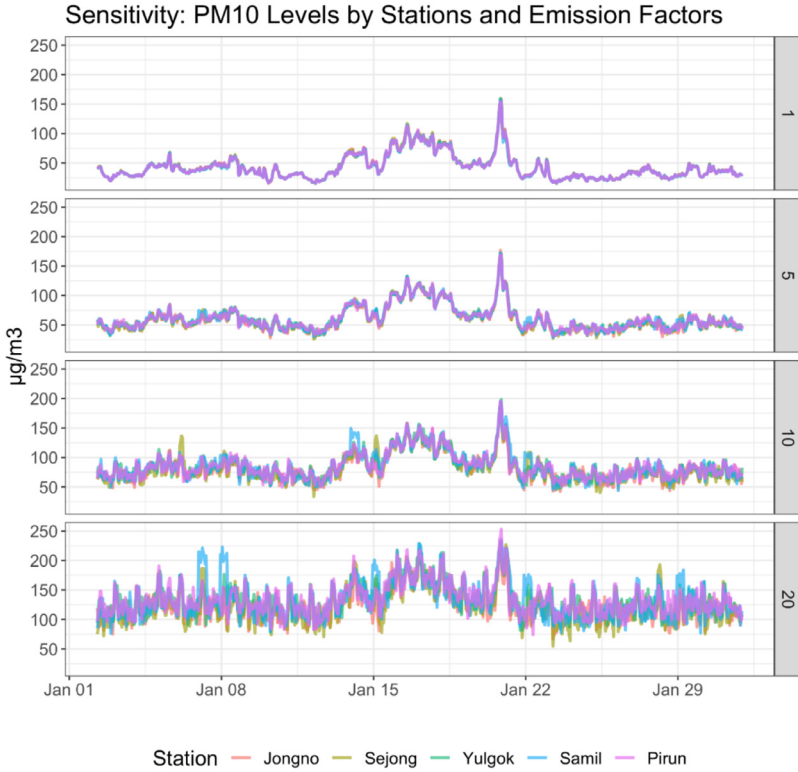
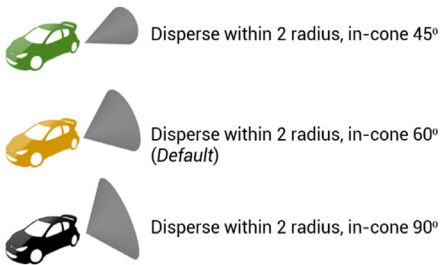


Fig. 13. PM₁₀ levels by emission factors of 1, 5, 10, and 20, each showing the N of vehicles that generate non-exhaust PM₁₀ emission. Each line represents 5 sample points of Jongno, Sejong, Yulgok, Samil, and Pirun roads. The variability at any station increases as the emission factor is increased.

Dispersion



Dilution

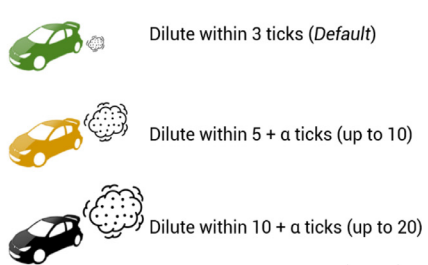


Fig. 14. Illustrations of dispersion parameters (left) and dilution parameters (right).

but from the evidence of Jongno, a distant station, it may deliver higher PM₁₀ to people walking near roads.

Unlike the dispersion results, all roads were very sensitive to the dilution period except for Emission 1 (see Table 20). In Emission 5, the default period of less than 3 min indicated an average figure of 60–62µg/m³, however, extending the period to 10 min increased PM₁₀ to 67–69µg/m³, which was 10% higher than the default.

Table 19

PM₁₀ concentrations by emission factors and dispersion range (cone width) (Unit: µg/m³).

Emission	Cone Width (°)	Jongno	Sejong	Yulgok	Samil	Pirun
1	45	50	49.6	50.5	50.3	50.9
	60	50.3	49.8	50.5	50.4	51.1
	90	50.7	49.9	50.6	50.5	51.1
5	45	58.4	55.7	58.7	58.9	60.1
	60	59.3	56.3	59	59.5	60.4
	90	60.4	56.6	59.1	59.5	60.8
10	45	73.2	71.2	77.3	77	80.5
	60	76.6	72.3	77.9	77.4	81
	90	79.6	73.1	78.5	78.1	81.8
20	45	102	100	112	113	118
	60	109	102	114	115	120
	90	116	104	115	118	120

Table 20

PM₁₀ concentrations by emission factors and (the duration until) dilution (Unit: µg/m³).

Emission	Duration	Jongno	Sejong	Yulgok	Samil	Pirun
1	3	45.5	45.8	45.8	46	46.1
	5	46.1	46	46.5	46.2	46.4
	10	46.7	46.5	46.8	46.7	47
5	3	60	60	62	62	62
	5	66	66	66	66	67
	10	67	67	68	68	69
10	3	81	81	86	85	85
	5	94	95	96	96	99
	10	99	99	100	100	102
20	3	123	123	134	133	133
	5	153	150	155	155	159
	10	164	160	165	164	167

Table 21

Car Ratio and PM₁₀ concentration (Unit: µg/m³).

Ratio	Jongno	Sejong	Yulgok	Samil	Pirun
0%	50.5	49.5	48.0	50.2	47.8
2.5%	59.5	60.0	60.3	60.9	61.4
5%	61.9	61.6	62.7	64.2	64.2
10%	62.3	61.8	63.4	64.4	64.4
20%	60.8	61.5	63.6	64.0	64.0

Table 22

Sum of standardized squared errors (SSSE) on January the 8th, 15th, and 22nd.

Date	Emission	Model	Observation	MSE
Jan 8th	1	49.2	56.4	25
	5	65.5	56.4	42
	10	85.3	56.4	419
	20	129.6	56.4	2684
Jan 15th	1	58.1	71.5	90
	5	75	71.5	6
	10	93.8	71.5	248
	20	136.1	71.5	2087
Jan 22nd	1	37.7	44.9	26
	5	53.9	44.9	41
	10	75.9	44.9	482
	20	114.6	44.9	2426

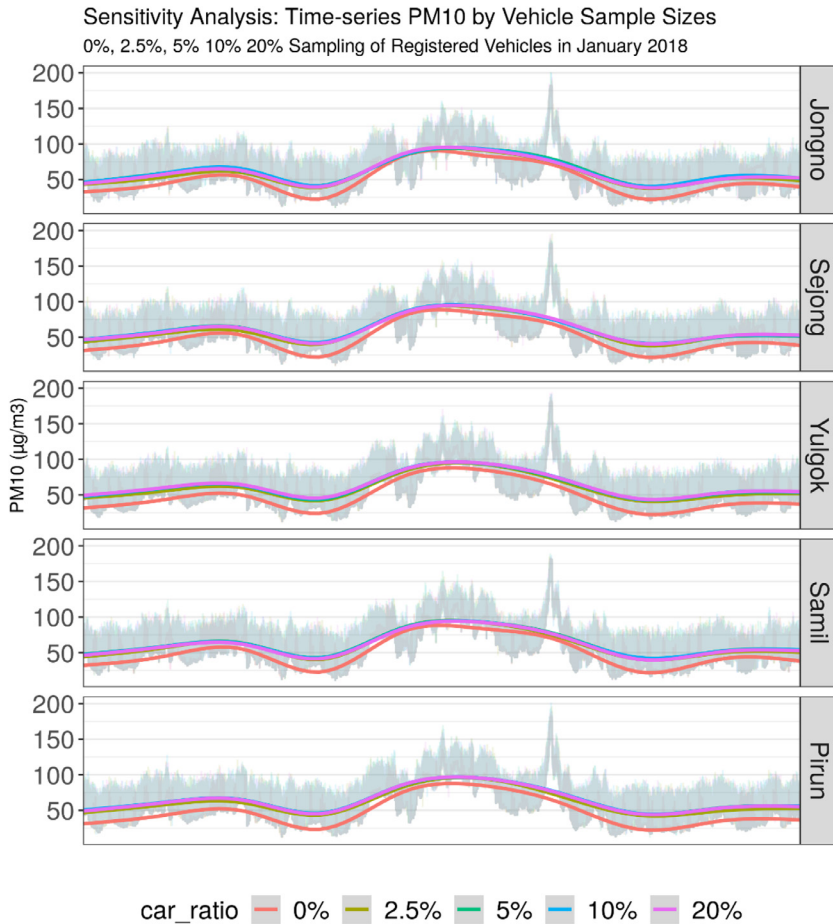


Fig. 15. PM₁₀ levels by car ratios of 0, 2.5, 5, 10, and 20%. The average values (smooth curve) of PM₁₀ is similar across roads but was different between 0% (no extra cars in the CBD) and the rest of the samples.

The difference between dilution periods increased proportionately to emission factors where the quickest (3 min) was 14–18 $\mu\text{g}/\text{m}^3$ higher than the slowest (10 min) in Emission 10 and 31–41 $\mu\text{g}/\text{m}^3$ in Emission 20. If this analysis was to represent the length of dust resuspension in the real world, say 3 min of dust floating until dilution, the deterioration of PM₁₀ can be explained by the floating particles from the vehicles that mixed well with the atmosphere. A disclaimer is that the dilution is only affected by the duration of ticks (zero wind), and no other components (e.g. wind, rain) that change dilution time.

PM₁₀ levels by car ratio

This section investigated how PM₁₀ can be sensitive to changes in car sampling (see Fig. 15). Resident vehicles were not included in this experiment as short-term journeys from the resident vehicles hardly contributed emission levels to the result (these were tested but not included in the thesis). To summarise, car ratios of 0, 2.5, 5, 10, and 20% mean sample rates of traffic counts by each minute were taken from the traffic monitoring statistics.

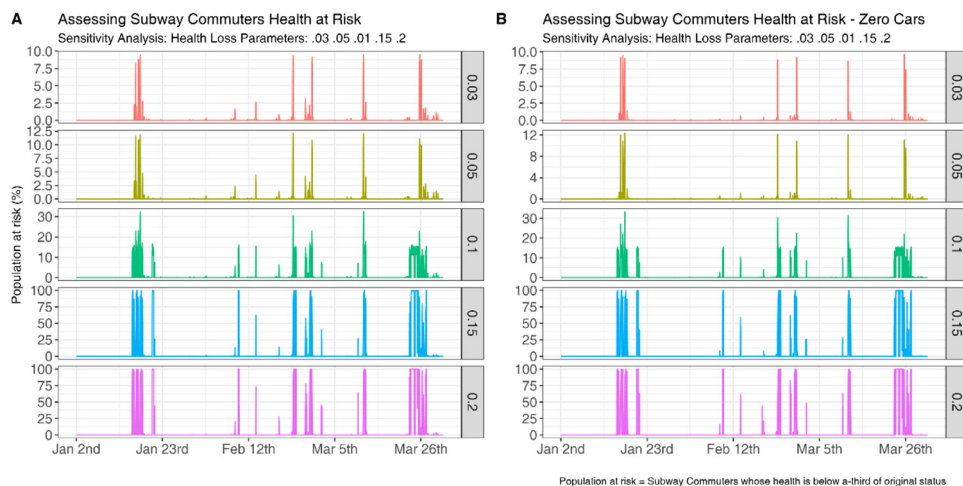


Fig. 16. Temporal change of risk rates for subway commuters (% of those with health under 100) with inbound cars (left), and without inbound cars (right).

In the 0% run, assuming no other vehicles, the roadside PM_{10} ranged between $47\text{--}50\ \mu\text{g}/\text{m}^3$ (which is equal to the background level), which was at least $10\ \mu\text{g}/\text{m}^3$ lower than the concentrations from other ratios. However, different sample sizes merely showed a small difference. For example, a 10% sample in Jongno only contributed $1.8\ \mu\text{g}/\text{m}^3$ more than that of 2.5%.

Surprisingly, all roads showed less pollution in the 20% sample because a massive number of vehicles failed to enter the study area. The queues were particularly long in Samil and Yulgok roads because Samil had fewer traffic signals at the entrance of the road which enabled vehicles to accelerate up to the core area with a few ticks but soon met several junctions, which can be depicted as a bottleneck effect; Yulgok has a roundabout that reduces the speed.

Health loss

This section investigated the health risk of subway commuters and resident drivers who are sensitive to the health loss parameters. Here, individuals only lose health when PM_{10} exceeds $100\ \mu\text{g}/\text{m}^3$, and contribute to the population at risk when one's health status falls below 100. Output A of each figure resulted from allowing extra inbound traffic in the CBD, whereas output B of each figure resulted from no other traffic than the resident vehicles.

For subway commuters, the population at risk appeared on January 20–22nd, late February, early March, and late March (see Fig. 16A). The maximum risk rate was 10% in 0.03 and proportionately increased to 30% in 0.1, but suddenly skyrocketed to 100% over 0.15. Although a lot of uncertainty from other parameters has contributed towards the outcomes, the tipping point of the health loss parameter was somewhere between 0.1 and 0.15. Several oscillations were also discovered during the extreme PM_{10} events. This was because subway commuters have different commute hours that led them to be exposed to ambient PM_{10} , and since health recovery activates when the individual arrives at home or the workplace, the risk rate oscillates frequently.

With a car-free experiment (see Fig. 16B), the results did not affect the health risks of subway commuters. This is because the trajectories of the commuters between stations and office locations were mostly distant from the road. However, the sensitivity between health loss parameters was comparable to the previous experiment: health risk proportionately rose until the parameter reached 0.1 but a sudden upsurge appeared when the parameter was over 0.15.

Compared to subway commuters, resident drivers experienced fewer occurrences of health risk, but higher surges in extreme PM_{10} episodes particularly over the parameter value of 0.15 (see Fig. 17A).

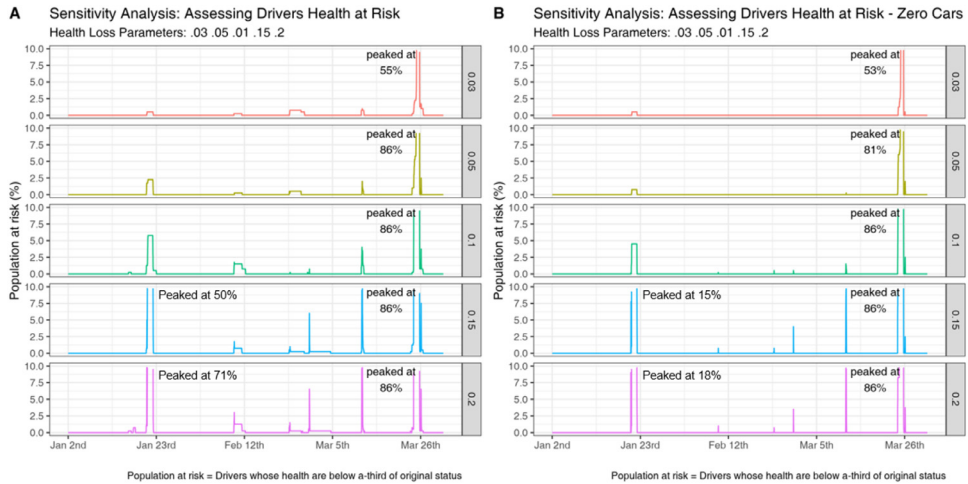


Fig. 17. Temporal change of risk rates for resident drivers (% of those with health under 100) with inbound cars (left), and without inbound cars (right).

Throughout the whole period, the health risk of resident drivers emerged during January 22nd, February 12th, March 8th, and March 24–25th, where the majority was at risk at the last peak. The prominent difference by the health-loss adjustment was very clear at the first peak where it started from less than a percent of risk at 0.03, then rose to 2.5% and 6% in 0.05 and 0.1, then surged to 50% and 71% on 0.15 and 0.2. In line with the subway commuters, a tipping point was also seen between 0.1 and 0.15.

However, the first surge that happened in 0.15 and 0.2 experiments significantly reduced to 15% and 18% in a car-free condition (see Fig. 17B). The other parameters only showed a less than 2% difference. This implies that the health risk of the drivers was not only sensitive to the health-loss parameters but also was affected by the emissions generated by non-resident traffic.

The difference in health risk can differ by the time the individual has spent outdoors when the ambient PM_{10} is over the threshold of $100 \mu\text{g}/\text{m}^3$, and how quickly that person recovered health. Even if 30% of subway commuters have experienced health risk, the short walking distance allowed them to recover promptly. By contrast, although drivers had fewer emergences of health risk, traffic congestion together with high background pollution had rapidly deteriorated the driver's health, especially on extremely polluted days.

This study chose one subway commuter and one driver to understand how the nominal health changed over time (see Fig. 18A). The light shaded colours shown in the background is the health status by each minute and the lines of turquoise and red are the moving averages. The health status of a subway commuter lost health earlier than the driver under the same condition. The driver might seem healthier than the pedestrian because the driver was never exposed to ambient PM_{10} which prevented multiple threats of major PM_{10} episodes. In Fig. 21B, the selected driver experienced fewer health risks in the car-free experiment, which can support the result of the population outcomes in Fig. 17B where a major fall in risk rate is for drivers in a car-free situation.

The rolling mean between the two groups converged as the parameters increased (see Fig. 18). The subway commuter's health was almost the same in different patterns, but for the drivers, the high parameter settings might have caused higher health loss even from a single pollution episode. The difference exists between the two on January 23rd because of the indoor factor of 0.7 that benefited the vehicle drivers.

In short, signs of deterioration in health appeared continuously in long-distance commuters on days when PM_{10} was on the rise, while the resident drivers had a relatively short period of commute

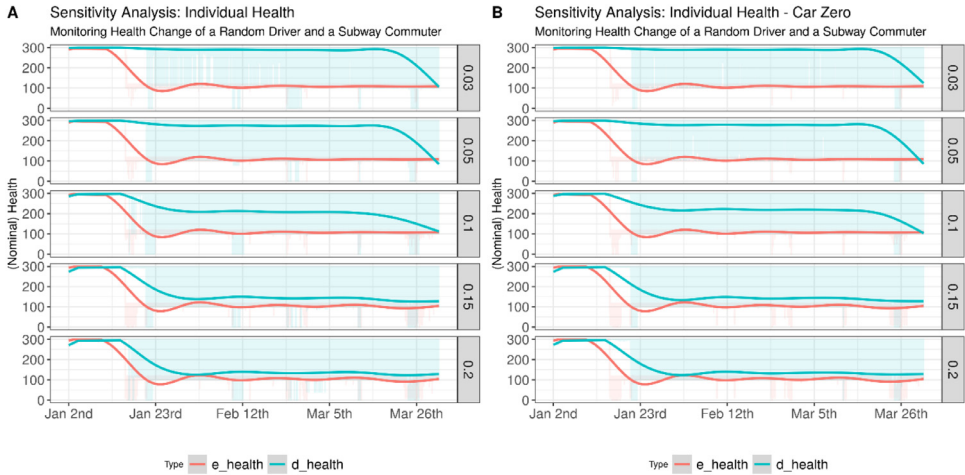


Fig. 18. Health comparison between a randomly chosen subway commuter (e_health) and a resident driver (d_health) with the case of (A) traffic and (B) traffic-free.

Sensitivity Analysis: Assessing Subway Commuters Health at Risk By Walking Speed

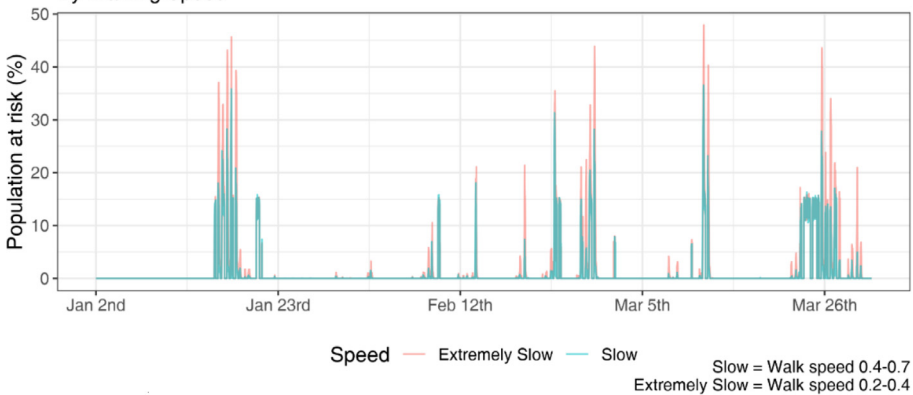


Fig. 19. Assessing subway commuters' health by different walking speed parameters.

time that prevented frequent health risk, but the extreme levels of PM_{10} led most of the drivers to an acute health crisis.

Walking speed

To test how walking speed affects the change to risk population, this section adjusted various levels of walking speed for subway commuters. Given the default speed at 0.6–1.0, the section tested (1) 0.2–0.4 patch per minute, (2) 0.4–0.7 patch per minute, and (3) 1.6–1.8 patch per minute. The range was given under the assumption that people have different walking speeds. Walking speed over 0.5 might seem rather unrealistic, but this experiment intended to illustrate how speed affects exposure levels.

The time series graph clearly showed that the onset and peak levels were very sensitive to walking speed (see Fig. 19). When the pedestrian's walking speed was "Extremely Slow" (0.2–0.4), more than 40% of the population was at risk on five different occasions with the highest peak of 47%. However, the risk rate declined by 10% when the walking speed increased to "Slow" (0.4–0.7)

and further declined by 30% when the speed increased to 1.6–1.8. This corresponds to the previous sensitivity analyses because slowing down the walking speed can mean that the person is prolonging the exposure time, which in turn causes a further health loss.

Calibration

This section calibrates emission factors with the observation values measured from Jongno roadside station. From the sensitivity analysis, it was found that the emission parameters were not only sensitive to the increase in the parameter but also to the variation between roads when the parameter was over 10. Calibration in ABM is very common as it controls the errors and the uncertainty close to the acceptable level [17].

This section did not calibrate PM_{10} across the study area because the background PM_{10} , which covers most areas, was already generated by the station data. Simulation results of Jongno were averaged from 20 iterations to avoid any noise from a particular run, then compared with the observation value. The simulation ran from the 2nd to the 31st of January 2018. This study used mean squared errors (MSE) and regression to examine the robustness of the model (see Fig. 20). MSE, as it is known, as the average squared difference between the estimated values and the actual value, can be used to compare the results in positive numbers, understanding the values closer to zero are more accurate. R^2 is useful because it is often easier to interpret since it doesn't depend on the scale of the data, and people are familiar with percentages. Note here that each method has its pros and cons and there is no ground rule in selecting a method.

On January the 8th, MSE varied largely by 25, 42, 419, and 2684 in parameters 1, 5, 10, and 20; they varied by 90, 6, 248, and 2087 on the 15th, and 26, 42, 481, and 2426 on the 22nd. Throughout the whole month, Emission 1 and Emission 5 had the lowest MSE values by 18 days and 12 days, respectively. The line graph shows that low biases for high values are observed in Emission 1, whereas high biases for low values are observed in the other parameters, but all modelled parameters could not replicate the peak values introduced from the observed values.

Regression results are similar to the MSE results, where the R^2 appeared to be highest in the lower two parameters and decreased significantly in the upper two parameters (see Fig. 20B). In line with the MSE results, the scatter plot from Emission 1 underestimated the observation values, in which most of the points were concentrated on the right side of the 1:1 line. Emission 5, on the other hand, slightly overestimated the results on the lower values but got most of the values, including the high values closer to the 1:1 line. Hence, although the overall MSE was lower and R^2 was higher in Emission 1, the author selected Emission 5 as the correct parameter. The reason being, that Emission 5 effectively expresses the extreme values on a polluted day, while at the same time predicting closer values to the truth value. Emission 1, even on a minute-by-minute basis, does not articulate the peak of particulates that have possibly dispersed into the local atmosphere.

Things to try and notice

How to use the model

Once the Netlogo interface is loaded, there are three buttons on the top row setup, go, step.

Please click on the setup to load the vehicles. The user can tick off the “view updates” tick box right next to the speed slider for quicker loading. Once the map is loaded, it is time to click go. The user can also click “step” to investigate each step. Once the simulation is running, the date, hours, and minutes change accordingly. In Fig. 7, there are also two yellow screens in the middle displayed as Unwell% and Unwell.Car, each of which accounts for the at-risk rate of pedestrians and resident vehicle drivers. This will change over time.

The health loss slide will change the level of health degrading. Since an individual's initial health begins with 300, it would be better to take the maximum of 0.2 to see a fluctuation of at-risk population throughout the simulation. However, the users have the freedom to toggle the slides. Medication is the temporary recovery level when an individual arrives indoors. This assumes that people take medicine when they feel unwell. The Emission factor is a parameter that can control the

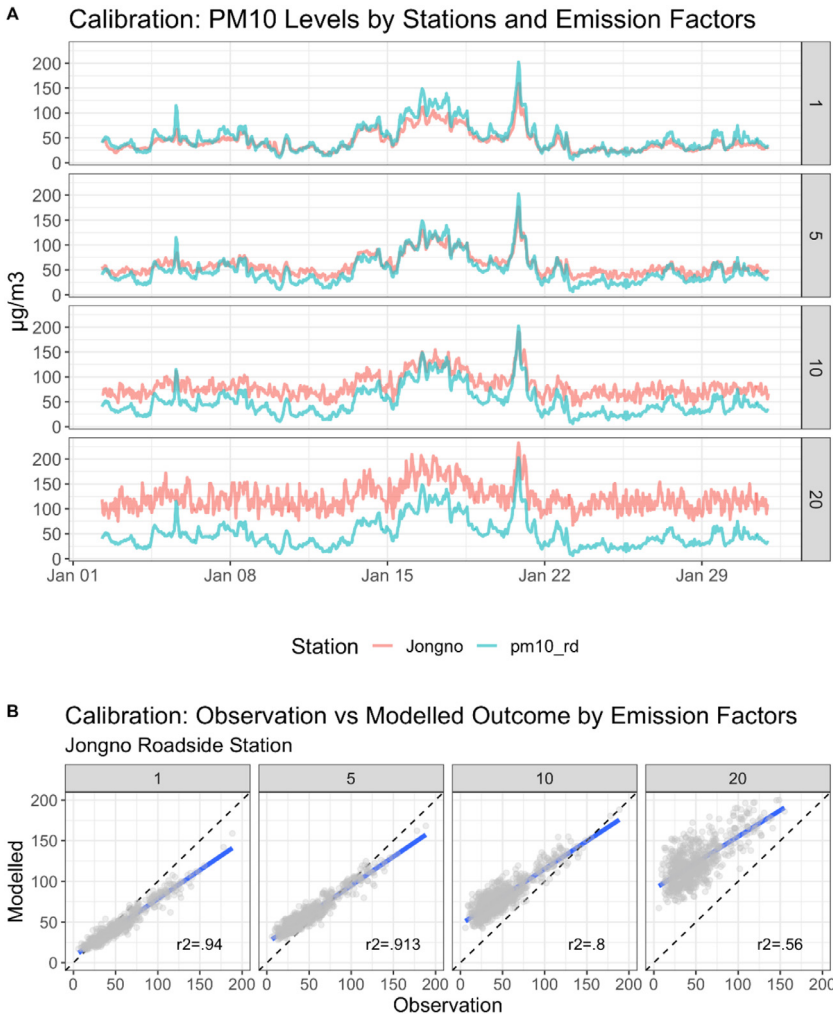


Fig. 20. Sensitivity output of adjusting the emission factor N from the equation of non-exhaust emissions (A) Jongno: Modelled, pm10_rd: Observation); (B) correlates the modelled output against the observation of Jongno roadside station. R^2 of factors 1, 5, 10, and 20 returns 0.94, 0.91, 0.8, and 0.56.

level of PM_{10} . Emission factor 1 refers to no effect while Emission factor 20 is almost 3 times higher than the ambient level. This has been fully tested through the sensitivity test. car-ratio controls the level of incoming vehicles (non-resident) to the CBD.

Possible artefacts and errors

The health loss parameter has been tested thoroughly and presented in the sensitivity section. However, if the user attempts to toggle beyond 0.2, the at-risk population will appear too early, and would never have the opportunity to recover. By contrast, if the parameter goes below 0.01, the agents will not have their health decreased irrespective of their exposure to high pollution episodes.

Likewise, the medication parameter also can cure the health status at the baseline of 10, however, if the parameter goes beyond 15, the agents who enter their offices or homes would easily recover to



Fig. 21. Visual artefacts of vehicles queuing to enter the study area.

their initial health status. This can mean that the population exposed to high pollution episodes can easily recover with some treatment. However, if the parameter is too low, say below 5, the outbreak of the at-risk population will appear much earlier. This might mean that the population is seemingly vulnerable to high pollution episodes.

This model simulated the resident vehicles to move freely during the weekend by applying the movement function of non-resident vehicles. However, there was no certain argument to spontaneously request these vehicles back to the origin point by the start of the weekday. Thus, the author set an argument to coerce the vehicles to their original locations. As a result, vehicles fly back home at midnight on Monday. The vehicles entering from the south have a long queue during the morning hours (see Fig. 21). Due to the lack of spatial extent, the queue entails an unnatural long tail that appears in the northeastern area.

Running the model on the HPC

Due to the intricacy of the rules, assumptions, and the spatial and temporal resolutions of the model, it took about 1 h and 10 min for a single run on a desktop computer.

Once the NetLogo code is completed, there are two options in which the users can choose. First is using the BehaviorSpace embedded in the NetLogo software. However, having simulated the model with BehaviorSpace, the model continued to terminate due to memory shortage. The developers from StackOverflow informed that NetLogo itself has memory limits around 1 GB, and the software is not optimised to remove the caches of the current job, which is stacked in the memory. Using multiple iterations for an intricate model will thus slow down the execution speed. The obvious reason was that the rules, assumptions, and spatial and temporal resolutions of the model was intricately designed, which took 1 h and 10 min for a single run on a desktop computer.

To speed up the modelling process, the alternative method was to implement the model on the HPC (High-Performance Computing) cluster. Here, an R package “nlrx” was used as a compiler to run

```

Sys.setenv(JAVA_HOME='/usr/local/software/spack/spack-0.11.2/opt
/spack/linux-rhel7-x86_64/gcc-5.4.0/
jdk-8u141-b15-p4aooptkqukgdix6dh5ey236kllhluvr/jre') #Java installed on HPC cluster

.libPaths("/home/hs621/renv/lib/R/library")

## Load packages
library(nlrx)
library(dplyr)
library(feather)

# Set Path
netlogopath <- file.path("/usr/local/Cluster-Apps/netlogo/6.0.4")
outpath <- file.path("/home/hs621/github/chapter5")

## Step1: Create a nl object:
nl <- nl(nlversion = "6.0.4",
        nlpath = netlogopath,
        modelpath = file.path(outpath, "CBD_Cars_July01.nlogo"),
        jvmem = 1024)

## Step2: Add Experiment
nl@experiment <- experiment(expname = "nlrx_spatial",
                           outpath = outpath,
                           repetition = 1,
                           tickmetrics = "true",
                           idsetup = "setup",
                           idgo = "go",
                           runtime = 127740,
                           evalticks=seq(1,127740, by = 1),
                           metrics = c("mean-pm10",
                                        "JongnoKerb_p",
                                        "Samil_p",
                                        "Sejong_p",
                                        "Pirum_p",
                                        "Yulgok_p",
                                        "[health] of one-of employees",
                                        "[health] of one-of cars with [not random-car]",
                                        "Drivers_p",
                                        "Walkers_p"
                                        ),
                           variables = list("car_ratio" = list(values = c(0.005,0.025, .05)),
                                             "awareness" = list(values = c("\no", "\yes"))),
                           constants = list("emission-factor" = 5,
                                             "no-of-employees" = 1932,
                                             "medication" = 10,
                                             "health_loss" = 0.1)
                           )

# Evaluate if variables and constants are valid:
eval_variables_constants(nl)

nl@simdesign <- simdesign_ff(nl = nl, nseeds = 1)

# Step4: Run simulations:
init <- Sys.time()
results <- run_nl_all(nl = nl)
Sys.time() - init
results$`random-seed`[1]
write_feather(results, paste("Scenario_", results$`random-seed`[1], ".feather", sep = ""))

```

Listing 12. R package “nlrx” used as a compiler for the HPC submission.

the model in headless mode [38]. Here, there are three separate tasks to submit the NetLogo model on the HPC as a batch process. Codes are available here: [GitHub HPC subfolder](#).

Using R as a compiler for high performance computing clusters

Listing 12 depicts the R code for assigning the headless simulation on the HPC using an R package nlrx [38]. Prior to the HPC code, we have fully tested the codes for the desktop machine, where the JAVA_HOME and the.libPaths are directed to their local paths. Note that the package is more friendly

```

#!/bin/bash
#!
#! Example SLURM job script for Peta4-Skylake (Skylake CPUs, OPA)
#! Last updated: Mon 13 Nov 12:25:17 GMT 2017
#!

#####
##### Modify the options in this section as appropriate #####
#####

#! sbatch directives begin here #####
#! Name of the job:
#SBATCH -J Scenario
#! Which project should be charged:
#SBATCH -A BITHELL-SL2-CPU
#! How many whole nodes should be allocated?
#SBATCH --nodes=1
#! How many (MPI) tasks will there be in total? (<= nodes*32)
#! The skylake/skylake-himem nodes have 32 CPUs (cores) each.
#SBATCH --mem=99999
#! How much wallclock time will be required?
#SBATCH --time=36:00:00
#! What types of email messages do you wish to receive?
#SBATCH --mail-type=END
#! Uncomment this to prevent the job from being queued (e.g. if
#! interrupted by node failure or system downtime):
#SBATCH --no-requeue

#! For 6GB per CPU, set "-p skylake"; for 12GB per CPU, set "-p skylake-himem":
#SBATCH -p skylake-himem

```

Fig. 22. Submitting the job to slurm (1) - the basic information.

with Java 8. The binary version of the package can be installed in Windows, Linux, and MacOSX using the R command `install.packages('nlrx')`. Once the packages are loaded, the next task (Step 1) is to assign the path for the NetLogo software and the output. Note the `nlversion` has to be identical to the version coded. NetLogo 6.0.4 was used.

The experiment section is where the configuration is happening. Since the model reached its memory ceiling after more than twice the repetition, we kept the repetition at 1 but submitted array jobs on the HPC. `metrics` is a function that exports the outputs. Our model exports 10 variables as a matrix. There are two variables `car_ratio` and `awareness` that were used for the scenario analysis. Constants are the variables that have the fixed values for the simulation.

Step 3 checks whether the variables and constants are valid for simulation. `eval_variables_constants` is used as an argument. If there is no problem with the parameters, the final job is to submit a simulation design. Since the study intends to evaluate the at-risk population at every tick, a full factorial simulation design is applied. In the package, there are other simulation models to test including Latin Hypercube Sampling (LHS), sensitivity analysis (Sobol method), and optimisation tools (genetic algorithm).

The final stage is to execute the simulation by using the `run_nl_all(nl = nl)` command. A timer command is written to check the total time spent for the model execution. If the model has finished its execution, the result is exported to a feather format. This is a form of a data frame based on Apache Arrow (<https://github.com/wesm/feather>) that allows the users to read and write data frames remarkably faster than the spreadsheet type extensions e.g. csv, xlsx. Feather is also suitable to open it from both R and Python.

Writing a slurm script

A slurm script was provided to fill in the tasks appropriate for the purpose. The file is named as "bau_slurm_submit.peta4-skylake_SeoulTraffic". The prerequisites for writing a slurm script is to ask

```

#! Optionally modify the environment seen by the application
#! (note that SLURM reproduces the environment at submission irrespective of ~/.bashrc):
. /etc/profile.d/modules.sh           # Leave this line (enables the module command)
module purge                          # Removes all modules still loaded
module load rhel7/default-peta4       # REQUIRED - loads the basic environment

module load r-3.6.0-gcc-5.4.0-bzuuksv
module load netlogo/6.0.4
module load miniconda3-4.5.4-gcc-5.4.0-hivczbz
source activate /home/hs621/renv
module load jdk-8u141-b15-gcc-5.4.0-p4aaopt

#! Insert additional module load commands after this line if needed:

#! Full path to application executable:
application="Rscript"

#! Run options for the application:
options="/home/hs621/github/chapter5/nlr_x_seoulCBD_scenario.R"

...the rest of the code is omitted due to the lack of space...

```

Fig. 23. Submitting the job to slurm (2) - detailed configuration for loading and implementing R and NetLogo.

the HPC administrator to install the appropriate versions of R, NetLogo, and Java 8. The author made a renv environment to keep all the packages and dependencies locally. Fig. 22 is the basic information when submitting the job. For example, what is the name of the project, which and how many nodes will you use (paid or free), how much wallclock time is needed, and so on. Fig. 23 indicates which software to include to run NetLogo and R. Since R works as a compiler, we use the command “Rscript” to make the compiler run on a bash script.

Submitting an array job

To consider the stochasticity of the model settings (see Section Stochasticity), 20 jobs was simultaneously submitted as a batch process. The batch process can be seen as a way of iteration. The code below can be written on the bash script.

```
sbatch --array=1-20 bau_slurm_submit.peta4-skylake_SeoulTraffic
```

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

- [1] F. Amato, Non-Exhaust Emissions: An Urban Air Quality Problem For Public Health; Impact and Mitigation Measures, Academic Press, 2018.
- [2] M. Kovoich, J.A. Parker, S.C. Oh, J.P. Lee, S. Wagner, T. Reemtsma, K.M. Unice, Characterization of individual tire and road wear particles in environmental road dust, tunnel dust, and sediment, Environ. Sci. Technol. Lett. 8 (12) (2021) 1057–1064, doi:10.1021/acs.estlett.1c00811.
- [3] B. Baensch-Baltruschat, B. Kocher, F. Stock, G. Reifferscheid, Tyre and road wear particles (trwp) - a review of generation, properties, emissions, human health risk, ecotoxicity, and fate in the environment, Sci. Total Environ. 733 (2020) 137823, doi:10.1016/j.scitotenv.2020.137823. <https://www.sciencedirect.com/science/article/pii/S0048969720313358>.
- [4] Air Quality Expert Group, Non-Exhaust Emissions from Road Traffic, Department for Environment, Food and Rural Affairs, 2019. Technical Report
- [5] R.M. Harrison, A.M. Jones, J. Gietl, J. Yin, D.C. Green, Estimation of the contributions of brake dust, tire wear, and resuspension to nonexhaust traffic particles derived from atmospheric measurements, Environ. Sci. Technol. 46 (12) (2012) 6523–6529.

- [6] N. Bukowiecki, P. Lienemann, M. Hill, M. Furger, A. Richard, F. Amato, A. Prévôt, U. Baltensperger, B. Buchmann, R. Gehrig, Pm10 emission factors for non-exhaust particles generated by road traffic in an urban street canyon and along a freeway in Switzerland, *Atmos. Environ.* 44 (19) (2010) 2330–2340.
- [7] S. Weinbruch, A. Worringer, M. Ebert, D. Scheuvs, K. Kandler, U. Pfeffer, P. Bruckmann, A quantitative estimation of the exhaust, abrasion and resuspension components of particulate traffic emissions using electron microscopy, *Atmos. Environ.* 99 (2014) 175–182, doi:10.1016/j.atmosenv.2014.09.075.
- [8] H.A. Denier van der Gon, M.E. Gerlofs-Nijland, R. Gehrig, M. Gustafsson, N. Janssen, R.M. Harrison, J. Hulskotte, C. Johansson, M. Jozwicka, M. Keuken, et al., The policy relevance of wear emissions from road transport, now and in the future—an international workshop report and consensus statement, *J. Air Waste Manag. Assoc.* 63 (2) (2013) 136–149.
- [9] V.R.J.H. Timmers, P.A.J. Achten, Non-Exhaust PM Emissions From Battery Electric Vehicles, Elsevier Inc., 2018, doi:10.1016/b978-0-12-811770-5.00012-1.
- [10] N. Khajeh-Hosseini-Dalasm, P.W. Longest, Deposition of particles in the alveolar airways: inhalation and breath-hold with pharmaceutical aerosols, *J. Aerosol. Sci.* 79 (2015) 15–30.
- [11] S. Gurram, A.L. Stuart, A.R. Pinjari, Agent-based modeling to estimate exposures to urban air pollution from transportation: exposure disparities and impacts of high-resolution data, *Comput. Environ. Urban Syst.* 75 (2019) 22–34, doi:10.1016/j.compenurbysys.2019.01.002. April 2018
- [12] U. Wilensky, W. Rand, *An Introduction to Agent-Based Modeling: Modeling Natural, Social, and Engineered Complex Systems with NetLogo*, MIT Press, 2015.
- [13] A. Crooks, N. Malleon, E. Manley, A. Heppenstall, *Agent-Based Modelling and Geographical Information Systems: A Practical Primer (Spatial Analytics and GIS)*, 1, SAGE Publications Ltd, 2019.
- [14] M. Tracy, M. Cerdá, K.M. Keyes, Agent-based modeling in public health: current applications and future directions, *Annu. Rev. Public Health* 39 (1) (2018) 77–94, doi:10.1146/annurev-publhealth-040617-014317.
- [15] J.D. Smith, C. Mitsakou, N. Kitwiroon, B.M. Barratt, H.A. Walton, J.G. Taylor, H.R. Anderson, F.J. Kelly, S.D. Beevers, London hybrid exposure model: improving human exposure estimates to no2 and pm2.5 in an urban setting, *Environ. Sci. Technol.* 50 (21) (2016) 11760–11768.
- [16] B. Edmonds, C. Le Page, M. Bithell, E. Chattoe-Brown, V. Grimm, R. Meyer, C. Montañola Sales, P. Ormerod, H. Root, F. Squazzoni, Different modelling purposes, *J. Artif. Soc. Soc. Simul.* 22 (3) (2019) 6, doi:10.18564/jasss.3993.
- [17] V. Grimm, S.F. Railsback, C.E. Vincenot, U. Berger, C. Gallagher, D.L. DeAngelis, B. Edmonds, J. Ge, J. Giske, J. Groeneveld, A.S.A. Johnston, A. Milles, J. Nabe-Nielsen, J.G. Polhill, V. Radchuk, M.-S. Rohwäder, R.A. Stillman, J.C. Thiele, D. Ayllón, The ODD protocol for describing agent-based and other simulation models: a second update to improve clarity, replication, and structural realism, *J. Artif. Soc. Soc. Simul.* 23 (2) (2020) 7, doi:10.18564/jasss.4259.
- [18] V. Grimm, U. Berger, F. Bastiansen, S. Eliassen, V. Ginot, J. Giske, J. Goss-Custard, T. Grand, S.K. Heinz, G. Huse, A standard protocol for describing individual-based and agent-based models, *Ecol. Modell.* 198 (1–2) (2006) 115–126.
- [19] D. Arribas-Bel, S. Alvanides, M. Batty, A. Crooks, L. See, L. Wolf, Urban data/code: a new EP-B section, *Environ. Plan. B Urban Anal. City Sci.* (2021), doi:10.1177/23998083211059670.
- [20] U. Wilensky, Netlogo, 1999,
- [21] H. Shin, Replication data for: exposure to traffic-related air pollution in central seoul using an agent-based framework, Harvard Dataverse, 2021, doi:10.7910/DVN/C93XLZ.
- [22] M.L. Kreider, K.M. Unice, J.M. Panko, Human health risk assessment of Tire and Road Wear Particles (TRWP) in air, *Hum. Ecol. Risk Assess.* Int. J. 26 (10) (2020) 2567–2585.
- [23] S.D. Beevers, N. Kitwiroon, M.L. Williams, F.J. Kelly, H. Ross Anderson, D.C. Carslaw, Air pollution dispersion models for human exposure predictions in London, *J. Expo. Sci. Environ. Epidemiol.* 23 (6) (2013) 647–653, doi:10.1038/jes.2013.6.
- [24] J.I. Halonen, M. Blangiardo, M.B. Toledano, D. Fecht, J. Gulliver, H.R. Anderson, S.D. Beevers, D. Dajnak, F.J. Kelly, C. Tonne, Long-term exposure to traffic pollution and hospital admissions in London, *Environ. Pollut.* 208 (2016) 48–57, doi:10.1016/j.envpol.2015.09.051.
- [25] K.D. Min, S.J. Yi, H.C. Kim, J.H. Leem, H.J. Kwon, S. Hong, K.S. Kim, S.Y. Kim, Association between exposure to traffic-related air pollution and pediatric allergic diseases based on modeled air pollution concentrations and traffic measures in seoul, korea: a comparative analysis, *Environ. Health* 19 (1) (2020) 6, doi:10.1186/s12940-020-0563-6.
- [26] H. Al-Thani, M. Koç, C. Fountoukis, R.J. Isaifan, Evaluation of particulate matter emissions from non-passenger diesel vehicles in Qatar, *J. Air Waste Manag. Assoc.* 70 (2) (2020) 228–242, doi:10.1080/10962247.2019.1704939.
- [27] P. Mao, J. Li, L. Xiong, R. Wang, X. Wang, Y. Tan, H. Li, Characterization of urban subway microenvironment exposure—A case of Nanjing in China, *Int. J. Environ. Res. Public Health* 16 (4) (2019) 625, doi:10.3390/ijerph16040625. <https://pubmed.ncbi.nlm.nih.gov/30791659> <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6406341/>.
- [28] J.D. Smith, B.M. Barratt, G.W. Fuller, F.J. Kelly, M. Loxham, E. Nicolosi, M. Priestman, A.H. Tremper, D.C. Green, PM2.5 On the London underground, *Environ. Int.* 134 (2020) 105188, doi:10.1016/j.envint.2019.105188. <http://www.sciencedirect.com/science/article/pii/S0160412019313649>.
- [29] D.Y.C. Leung, Outdoor-indoor air pollution in urban environment: challenges and opportunity, *Front. Environ. Sci.* 2 (2015) 69, doi:10.3389/fenvs.2014.00069.
- [30] W. Zeng, R.L. Church, Finding shortest paths on real road networks: the case for A*, *Inter. J. Geogr. Inf. Sci.* 23 (4) (2009) 531–543, doi:10.1080/13658810801949850.
- [31] EMEP/EEA, EMEP/EEA Air Pollutant Emission Inventory Guidebook 2019, European Environment Agency, 2019. Technical Report
- [32] J. Panko, J. Chu, M.L. Kreider, K.M. Unice, Measurement of airborne concentrations of tire and road wear particles in urban and rural areas of France, Japan, and the united states, *Atmos. Environ.* 72 (2013) 192–199, doi:10.1016/j.atmosenv.2013.01.040.
- [33] I. Nikolova, S. Janssen, P. Vos, P. Berghmans, Modelling the mixing of size resolved traffic induced and background ultrafine particles from an urban street canyon to adjacent backyards, *Aerosol. Air Qual. Res.* 14 (1) (2014) 145–155.
- [34] B. Srimuruganandam, S.M. Shiva Nagendra, Analysis and interpretation of particulate matter - PM10, PM2.5 and PM1 emissions from the heterogeneous traffic near an urban roadway, *Atmos. Pollut. Res.* 1 (3) (2010) 184–194, doi:10.5094/APR.2010.024. <http://www.sciencedirect.com/science/article/pii/S1309104215305456>.

- [35] M. Ferm, K. Sjöberg, Concentrations and emission factors for PM_{2.5} and PM₁₀ from road traffic in Sweden, *Atmos. Environ.* 119 (2015) 211–219, doi:[10.1016/j.atmosenv.2015.08.037](https://doi.org/10.1016/j.atmosenv.2015.08.037).
- [36] R. Smit, L. Ntziachristos, P. Boulter, Validation of road vehicle and traffic emission models - a review and meta-analysis, *Atmos. Environ.* 44 (25) (2010) 2943–2953, doi:[10.1016/j.atmosenv.2010.05.022](https://doi.org/10.1016/j.atmosenv.2010.05.022).
- [37] H. Shin, M. Bithell, An agent-Based assessment of health vulnerability to long-Term particulate exposure in Seoul Districts, *J. Artif. Soc. Soc. Simul.* 22 (1) (2019).
- [38] J. Salecker, M. Sciaini, K.M. Meyer, K. Wiegand, The nlrX r package: a next-generation framework for reproducible NetLogo model analyses, *Methods Ecol. Evol.* 10 (11) (2019) 1854–1863, doi:[10.1111/2041-210X.13286](https://doi.org/10.1111/2041-210X.13286).