Forecasting the flow of urban pollution with cellular automata

Sukanya Benjavanich
Dept. of Computer Science
Heriot-Watt University
Edinburgh, UK
sb53@hw.ac.uk

Ziauddin Ursani
Heriot-Watt Unversity *and*
Route Monkey Ltd
Edinburgh, UK
ziauddin.ursani@routemonkey.com

David Corne
Dept. of Computer Science
Heriot-Watt University
Edinburgh, UK
dwcorne@gmail.com

***Abstract*—Urban pollution is a growing health hazard in many urban centres across the globe. Prominent sources of pollution include diesel and gasoline vehicles, as well as manufacturing plants, power generation processes, and other industrial activity. In order to help understand and address pollution levels, a number of cities are installing sensor arrays; these installations will in future support monitoring and tracking of pollutants, and also underpin a range of possibilities for forecasting and mitigation. In this paper we describe an approach which forecasts the future flow and intensity of pollutants around an urban area, given recent historic sensor streams. The approach employs a cellular automaton, whose parameters are learned and adapted online by an evolutionary algorithm. (*Abstract*)**

Keywords—pollution, big data, forecasting, cellular automata, evolutionary algorithm (*key words****)***

I. INTRODUCTION

Urban pollution is an increasingly serious challenge. The World Health Organization has recently estimated that, in 2012, one in nine of all deaths was due to conditions related to air pollution, and 3 million of those deaths were solely attributable to outdoor (ambient) pollution [1]. Outdoor pollution affects all regions, age groups and socioeconomic groups, In 2005, outdoor pollution was assessed to have led to around 800,000 premature deaths, and 6.4 million years of life lost, primarily in developing countries [2]. Pollution measurements and studies focus primarily on the presence in the air and effects of fine particulate matter smaller than 2.5 microns (so called: $PM_{2.5}$); high levels of $PM_{2.5}$ are implicated in mortality from cardiopulmonary disease, cancers of the trachea, bronchus, and lung, and about 1% of mortality from acute respiratory infections in children under 5 years [2].

The major contributor to $PM_{2.5}$ and other pollutants in cities is traffic, while power generation (to supply domestic and industry energy) is a secondary contributor. In the UK alone, pollution from traffic is estimated to be the cause of 5,000 premature deaths per year, more than double the toll that result from road accidents.

In recent years, an accelerating number of initiatives have begun to install air quality sensors, among other instrumentation, in urban areas. These initiatives range from small scale projects to major infrastructure procurements, and tend to focus on capturing and monitoring a broad range of data streams that characterise the urban environment. Pollution monitoring via local or regional authorities, or via universities on their behalf, has been in place for many years [4,5,6,7]. More recent developments, often developed in the framework of the 'Internet of Things' [8], aim to allow more open and real-time access to the sensor streams, and typically measure a variety of additional quantities such as sound-levels and weather. Such projects include Chicago's 'Array of Things' [9], Barcelona's Smart City Initiative [10], and many others.

Installed monitoring of air quality and associated factors is clearly fundamental for understanding how pollution impacts on the urban environment. However, little work has yet been done to exploit such data in ICT approaches that can help mitigate the effects of pollution. Accessible visualisation of the current monitored data is obviously useful, and leads to some mitigation, by virtue of alerting users to areas that should currently be avoided. However, the wealth of data available should support further possibilities.

To better exploit monitored urban air quality data, several potential use-cases can be imagined, all of which depend on being able to forecast how the air quality metrics will change in future time windows. If, at any given point in time, actors in the urban authority can forecast air quality profiles over the next few hours, this could be used in contexts such as pre-emptive (rather than reactive) traffic re-routing, road access management, pre-emptive shutdown or curtailment of operations at sites that generate pollution, and health-related interventions such as school closure or control of HVAC systems at hospitals.

However, the pollution forecasting process itself will depend strongly on environmental factors such as wind-speed, wind bearing, temperature, and humidity. It will also depend on local pollution sources, such as traffic levels and factories. In particular, pollution forecasting requires *forecasts* of all of the latter factors. Environmental agencies, such as the UK Met Office or the US NOAA, do provide forecasts of the required environmental variables, and these are typically available via a number of online services (e.g. [11]); however, these forecasts are typically at low granularity (e.g. a single temperature value per square kilometre). For appropriate fidelity in an urban environment, with complex topography and micro-climate, urban pollution flow forecasting requires higher granularity.

978-3-901882-99-9

To best exploit the data streams available by monitoring urban air quality, a clear challenge emerges, as follows. An ICT system needs to do all of the following: (i) forecast environmental factors relevant to pollution flow; (ii) forecast local pollution generation (e.g. from traffic and industrial sites); (iii) perform pollution flow modelling given those forecasts; (iv) achieve (i)—(iii) repeatedly as updated data arrives, and with sufficient rapidity to serve use-cases that lead to intervention and control at useful timescales.

In this paper we describe our approach to this overall task, which is centred on the use of a cellular automaton (CA) to model and simulate the flow of pollution, given 2D locational array parameters for wind-velocity and pollution source activity. The approach is under development for deployment within a commercial product development project sponsored by an environmental consultancy. For reasons of commercial sensitivity, we do not reveal the full details of the system under development, however we describe and evaluate a full version of the core research CA-based system which underpins it, and demonstrates the fundamental ideas.

The remainder is set out as follows. In section II we review relevant background. In section III we describe our approach in detail. In section IV we report on empirical exploration of our approach, evaluating how well the CA can predict the next several hours of pollution levels across an urban environment. Section V provides a concluding discussion.

II. Background

A. Pollution Modelling

The science of understanding how pollutants move in the atmosphere is commonly referred to as *atmospheric dispersion modelling* [12]. A prominent technique in this context is the 'Gaussian plume' approach, which is at the core of most practical air dispersion models [13]. The latter essentially assumes a point source of pollution at a constant rate, light winds, and incorporates many simplifying assumptions, and is driven by the following fundamental equation:

$$C(x, y, z) = f(Q, K_y, K_z)$$

indicating that, if a continuous source of pollutant operates from the origin at a rate Q (e.g. grams per second), and wind velocity is primarily in the x direction, then the concentration of pollution at any given co-ordinate is a function of Q, and the 'turbulence diffusivities', K_y and K_z. The latter capture how the 'plume' disperses in the y and z directions under light winds.

Most pollution modelling research and practice is grounded in this equation, but incorporates additional factors in accordance with the specific application in mind [14]. Complicating factors are manifold, e.g.: winds may not be light, and tend to be variable; the source may be non-continuous. Advanced models, used for (hopefully) high quality prognostics by national agencies, or to deal with special cases (such as the Fukushima accident [15]), often incorporate a variety of additional technical mechanisms, including computational fluid dynamics [16]. Evaluation, validation and comparison of models in this area, however, are all very difficult to achieve, and consequently almost non-existent [16]. Arguably, air dispersion models tend to be trusted to the extent that their outputs seem plausible, and that the science (and simplifications) behind them seem justified.

B. Cellular Automata

Cellular Automata (CA) are simple, discrete dynamic systems that are often used to model and simulate complex natural processes. They have been applied to a wide range of phenomena, including ecological competition [17], wildfires [18], freeway traffic [19], infection dynamics [20], and many more. CAs provide a convenient approach for the modelling of complex phenomena; rather than derive analytic solutions (infeasible for most systems of interest, without making gross simplifications), CAs typically require only the design of intuitive local rules to model a system's dynamics. Operation of the CA then amounts to a simulation of the system under study, and – with well-designed rules and parameterisation – the fidelity of this simulation is widely held to be effective, often displaying spatiotemporal dynamics that echo behaviour in the true system but are not apparent in the outcomes of alternative modelling approaches.

CA modelling of air pollution has been very little researched to date, however there is much active CA research in the general area of environmental applications. E.g. Nakano et al. [21] used a CA to model the dynamics of an oil slick on the Japan sea emerging from a damaged tanker; they report good agreement between the CA results and analytical solution, noting the speed advantage and modest resource requirements of the CA approach, which make it possible to use it in near real time to help develop mitigation plans when an accident occurs. Similarly, Xiao-Ping et al [22] investigated using a CA to predict point source pollution into a river via sewage influx. They achieved reportedly good results by incorporating mechanisms to account for complex river flow in the CA's rules, together with the pollutant dispersion rules. Like many authors, they emphasise the speed of the CA simulation approach in comparison to the much more resource-intensive alternatives based on partial differential equations.

The earliest CA work in air pollution seems to be [23], a feasibility study, showing how the standard advection/diffusion equations can be recast as local rules for a CA, and demonstrating, by visualization, how running the CA leads to plausible dispersion patterns. Very little work has been done so far to build on [23], until recently [24], where Laurent et al [24] have explored this idea and tested it, including its comparison with the more traditional and specialized approach, for modelling methane dispersion in air. The focus of [24] was to hybridize their CA with a neural network, trained on a database of results from the CFD simulations; this enabled the CA to rapidly estimate good parameter settings based on context. They report that their combined approach matches the accuracy of the CFD method while being c. 120 times faster.

III. CA modelling of urban pollution Dynamics

A. The Cellular Automaton Structure and Operation

The task we face is to predict the flow of pollution in an urban area over relatively short-term timescales. A representative

scenario would be: a 2—5km by 2—5km central urban area of interest; around 5—20 sensor units scattered in the area (typically installed outside buildings or on lampposts); a number of major roads criss-crossing the area, and in some cases there will be significant non-traffic sources of pollution.

We model such a scenario by encompassing the area within a 2D grid of cells. Contents of a cell, from the modelling viewpoint, are either sensor units, industry sources, or traffic sources. Cell size is chosen pragmatically. Figure 1. illustrates this, showing an 8x8 grid of CA cells covering a contrived urban area. This area contains ten sensor units (labelled 'S'). Roads are indicated by dashed lines. The arrows indicate wind and will be discussed later. Each cell of the CA therefore contains a number of sensor units (typically zero or one), and a number of sources (typically zero, one or two), which can be either industry or traffic based. Parameters for the sources (in terms of units of pollution generated per time-step) are estimated from appropriate literature. Additionally, each cell has two wind-speed parameters (North-South and East-West). Finally, each cell has a pollution level, which updates in each time-step as a function of its previous level, levels in neighbouring cells, and the action of the sources and wind.

To explain in more detail, we first introduce some notation. A cell $C(r,c)$ is identified by its row r and column c, and $P(r,c,t)$ denotes its pollution level at timestep t. We can think of this as a specific number of particles (later fitted to real-world measurements via calibration). Pollutant in a cell will *disperse* into its neighbours as a result of two processes: first, *advection*, which is the net movement caused by wind-flow; second, *diffusion*, which is the molecular process that leads to diffusion outwards from the source, even in wind-less conditions. A third factor that must obviously be taken into account is *generation* of additional pollutants within the cell, typically from traffic. There are therefore three factors which affect the pollution level at each timestep. Described next, these are the sources at the cell, the wind action at the cell (advection), and the diffusion process at the cell.

Sources: Each cell has a (possibly empty) list of sources $S(r,c) = \{s1, s2, \ldots, sn\}$; each of these sources, whether traffic or industry, is associated with a series of values, one per timestep. Thus, $G(sj,t)$ indicates the amount of pollution introduced by source sj into its cell at timestep t.

Wind: A wind-flow field exists, $WN(r,c,t)$ and $WE(r,c,t)$, respectively representing the north/south and east/west components of the (known or forecast) wind-speed at cell $C(r,c)$ at each timestep t. Correspondingly, each cell has two wind-flow parameters $N(r,c)$ and $E(r,c)$, respectively representing multipliers for the wind-flow values. These parameters enable the model to treat wind effects differently in each cell, thereby modelling local topography 'urban valley' and similar effects.

Diffusion: In each timestep, pollution at each cell reduces by a specific percentage (the same for all cells), and the levels in the 8 surrounding cells are correspondingly increased. The following therefore happens in association with each cell $C(r,c)$. First, pollution is reduced according to the diffusion parameter δ (between 0 and1):

$$P(r,c,t+1) = P(r,c,t) - \delta \cdot P(r,c,t)$$

Next, pollution levels at the eight neighbouring cells are increased, modelling the natural diffusion process. Thus, for each cell $C(p,q)$ which is a direct neighbour of $C(r,c)$:

$$P(p,q,t+1) = P(p,q,t) + 0.125 \cdot (1-\delta) \cdot P(r,c,t)$$

For edge cells, this only occurs for the existing neighbours.

Next, two neighbouring cells are updated in accordance with the wind-flow parameters at $C(r,c)$; the basic action for neighbour $C(p,q)$ and wind-flow field value w (either $WN(r,c,t)$ or $WE(r,c,t)$) and wind-flow parameter β (either $N(r,c)$ or $E(r,c)$) is as follows:

$$P(p,q,t+1) = P(p,q,t) + w \cdot \beta \cdot P(r,c,t)$$

If $WN(r,c,t)$ is positive (negative), then the above is done for the immediately due north (south) neighbour. Similarly, If $WE(r,c,t)$ is positive (negative), then the above is done for the immediately due east (west) neighbour.

Finally, the action of each source sj at each timestep is simply to add the amount $G(sj,t)$ to the pollution level of its cell at that timestep. Slightly more formally and precisely, we can say that, for each source sj in the set of sources $S(r,c)$:

$$P(r,c,t+1) = P(r,c,t) + G(sj,t)$$

Overall CA operation is as follows. First, the CA is initialised by setting the pollution level at each cell to a starting level. Then, for a given number of timesteps, we (i) randomly permute the cells, (ii) update the cell's pollution level, and those of its neighbours, according to the above steps. The development of pollution values in each cell over time depends on environmental data (the wind-flow field, and source generation profiles), and on the wind-flow parameters for each cell, and the diffusion parameter δ. The former (wind-flow field and source generation profiles) are extrinsic to the CA, and are discussed later. Finally, if we are modelling or tracking more than one specific pollutant for which we have distinct sensor values (e.g. $PM_{2.5}$, PM_{10} and NOx), the above process would run independently in each cell for each distinct pollutant; in such cases, additional rules could be invoked to model interaction between them. Meanwhile, in the following subsection we describe how the wind-flow parameters and diffusion parameter are learned from historical data.

B. *Fitting the CA to historical data*

An Evolutionary Algorithm [25] is used to fit the CA to historical data, learning parameter values that correspond to the closest and most robust match with historical data. This process requires, ideally, the availability of historical data for each of the following, for H timesteps: (i) wind components $WN(r,c,t)$ and $WE(r,c,t)$, for each cell on the grid, for t from 1 to H. (ii) source generation values $G(sj,t)$ for all sources sj, for t from 1 to H. (iii) actual pollution-level sensor values $A(r,c,t)$, for each cell on the grid, for t from 1 to H.

Given our historical dataset, coupled with initial (perhaps arbitrary) values for the diffusion and per-cell wind-flow parameters, we can now run the CA from timesteps 1 to H. As a result of applying the rules detailed in subsection III.A, we can regard the 'output' from the CA, for each timestep from

t=2 onwards, to be the calculated pollution levels $P(r,c,t)$. It is worth remembering, however, that only a subset of the cells have sensors installed; e.g., in the scenario of Figure 1, this subset would be {B1,B4,D3,E5}. We can therefore evaluate the current values of the CA's parameters by comparing the historical values $A(r,c,t)$ with the calculated values $P(r,c,t)$, over this specific set of cells.

In more detail, we evaluate a CA parameter set by calculating the sum-squared-error (SSE), defined as:

$$\text{training\_fitness} = \sum_M \sum_{t=2}^{m} (A(r,c,t) - P(r,c,t))^2$$

where M denotes the set of cells that contain sensor units, and m, where ($1 < m < H$), is the last training-data timestep. In the algorithm described next, we use this as the fitness function, and thus aim to learn parameter values for the CA that fit well to the actual pollution values at sensor sites at times 2 to m inclusive. Meanwhile, during the learning process, we keep track of the following quantity:

$$\text{test\_fitness} = \sum_M \sum_{t=m+1}^{v} (A(r,c,t) - P(r,c,t))^2$$

which assesses the performance of the current parameter set on a *validation* set of timesteps, where $1 < m < v < H$).

To then choose the parameter values that will be used for forecasting future values, we use the parameter set that led to the best performance on test_fitness. To then deploy the CA for forecasting, we simply continue the simulation beyond timestep H, with the chosen learned values.

Finally, to capture accuracy beyond v, we define:

$$\text{performance\_fitness} = \sum_C \sum_{t=v+1}^{H} (A(r,c,t) - P(r,c,t))^2$$

which evaluates performance over *all* cells in the forecast regime at timesteps beyond v.

C. Parameter-learning via an Evolutionary Algorithm

Given a CA setup with N cells, the task of the evolutionary algorithm (EA) is to learn good values for $2N$+1 parameters; these are the diffusion parameter, δ, and the two wind-flow parameters per cell, $N(r,c,t)$ and $E(r,c,t)$. The EA's 'chromosome' structure is therefore a straightforward real-valued vector of size $2N$+1, mapping directly onto the required parameters, each limited between 0 and 1. The EA's fitness function operates by running the CA from timesteps 1 to m with the parameter set encoded in the chromosome, and then returning the *training_fitness* as the fitness value for purposes of selection (*test_fitness* is also calculated, and recorded via a book-keeping process which remembers the parameter set that is best-so-far on *test_fitness*).

Other details of the EA are as follows: population size is 20, binary tournament selection, uniform crossover (rate 0.7), Gaussian single-gene mutation (with std 0,1), and a steady-state generation strategy. Training continued in each run until there had been no improvement for 1000 iterations.

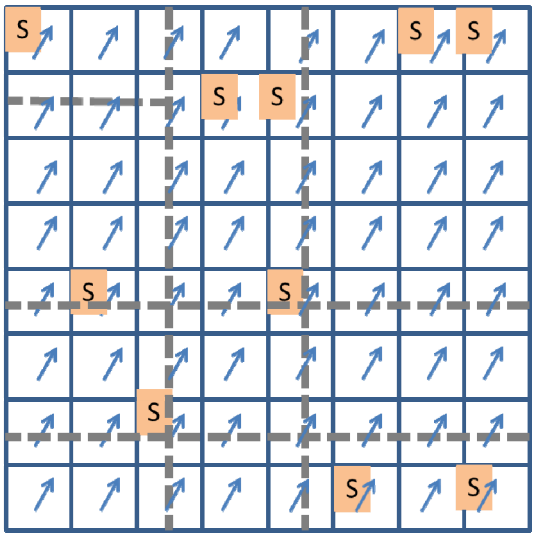

Fig. 1. Illustrating the experimental scenario; an 8x8 grid covering a 3km by 3km urban area. Significant traffic generating roadways are shown as dashed lines; wind direction is shown as a broadly northeast facing arrow in each cell; this angle (from North) varied from 30 to 80 degrees during the 48 timesteps. In most experiments reported next, ten sensor sites were used. A random configuration of 10 sensor sites is illustrated here (with 'S') – however in the experiments, the sensor positioning was randomised for each independent run.

IV. Empirical Exploration

The approach we describe is part of technology being developed in connection with an ongoing commercial project. The data-streams from that project are commercially sensitive; here we therefore describe only experiments using synthetic data. In the remainder of this section, we first describe the synthetic dataset, and then describe experiments aimed at understanding the degree to which our approach can forecast urban pollution flow in a range of scenarios.

A. Synthetic Data Generation

A partly-synthetic dataset for development and testing purposes was developed as follows, based on a putative deployment in an Asian city. An 8x8 cell CA grid was overlaid on a 3km square region. Historical wind-speed data was obtained for positions around the city corresponding to cell centres, for a period of 48 hours. Using a variety of documentary sources, and considering the road network layout in the area of interest, a realistic time-series of traffic-generated $PM_{2.5}$ levels was synthesised. Using a bespoke simulator, pollution generation and flow was then generated for the area in question for 48 timesteps. No industry sources were modelled in this case (traffic being by far the most prominent source of pollution in most city contexts). Finally, randomized parameters were generated per cell to model the way that urban topography would restrict wind-based dispersion.

In the above way, a time series of 'actual' pollution levels were synthesised at each cell. In the following experiments, we explore the ability of the CA and EA system to model and forecast in this scenario. This generally consists of training the system on a portion of the data (e.g. the first 36 hours) and inspecting its performance in matching the 'actual' values at sensor sites during the subsequent hours.

B. Pollution Flow Forecast Accuracy

We first explored performance for different numbers of sensor sites in the 64-cell scenario. Performance rapidly improved from 1 to 10 sensors, with, broadly speaking, diminishing returns beyond that point. We therefore fixed the number of sensors at 10 (as in Fig. 2) for the remaining experiments.

Several applications focus around the potential to understand, from early in the working day, how pollution will develop during the remaining working hours. Thus we focus experiments on learning up to timestep 36 (noon on day 2) and forecasting the remaining hours. Such contexts will typically require regular update taking recent data into account. This quickly becomes a concern for computational resources, especially where the CA grid is large; it is therefore of interest to examine the accuracy of forecasts when training with shorter historical windows. We therefore vary window size from 8 to 36 in steps of 4; for a window size *W,* this means that training was done only for the *W* timesteps from timestep 36 – (*W*+1) to timestep 36 inclusive. Meanwhile, while parameter-learning focusses (necessarily) on minimising error at sensor sites, in our evaluations in this section we report the root mean squared error per cell over *all* cells during the forecast timesteps.

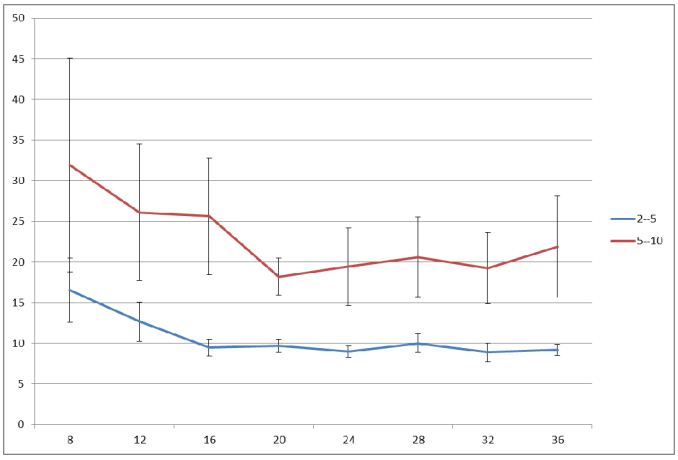

Fig. 2. Summarising performance for CA experiments with ten sensor sites and 10% input wind/traffic forecast errors, for varying training timesteps (from 8 to 36), and for two overall background wind levels, 2—5 m/s and 5—10 m/s.

Figure 2 summarises the results. The lower curve uses historical wind-speeds of 2—5 m/s. With the upper curve, we also explore artificially higher wind-speeds of 5—10 m/s. The value plotted is *performance_fitness* calculated between timesteps 37 to 48. An error of *X*% indicates that the forecast $PM_{2.5}$ level per cell per timestep was *X*% away from actual.

Considering the 2—5 m/s tests, as can be seen, the training horizon clearly has an effect on forecast accuracy when it is particularly short, and there seem to be diminishing returns above a certain level (here 20—24 hrs). However there are clearly many factors at play. In terms of computational resources, the time taken is broadly in direct proportion to the length of the training horizon, hence shorter windows are clearly preferable as long as they come with acceptable performance profile. Finally, the higher wind-speed scenario clearly compromises the ability of the CA to forecast well, however such scenarios are unusual in urban settings.

C. Sensitivity to sensor and environmental forecast error

When the CA operates beyond the horizon of its training and testing data, it depends on forecast values for both wind-speed components and industry and traffic sources. In the results shown in subsections IV.A and IV.B, the pollution source and wind-speed values used in timesteps 37 and above were the 'actuals' in the dataset. However at the time (here timestep 36) when the CA would be required to generate pollution forecasts, only wind-speed, traffic and source *forecasts* will typically be available, and these will certainly vary to some extent from the actual, and this in turn will impact on the pollution forecasts. It is therefore of interest to see how well the proposed system can cope with error in both wind-speed and source-generation forecasts.

To investigate the above, we fixed the number of sensor sites at 10, and we fixed the window size at 36, but with Gaussian error added to the wind-speed 'actuals' components, and the traffic source components, at timesteps 37—48. Figure 3 illustrates the outcomes, plotting the mean percentage errors (over all sensor sites and test time-steps) of forecast pollution levels for each of 0%, 10%, 20% and 30% error in the input forecast data. Again, each point is the mean of 20 runs.

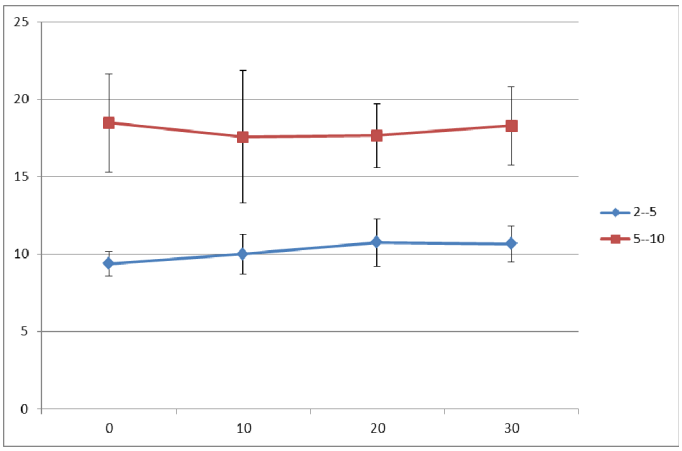

Fig. 3. Summarising forecast performance for CA experiments with ten sensor sites, but for varying amounts of noise in the wind-speed and traffic-generation forecasts for timesteps 37--48. Vertical axis is percentage error in $PM_{2.5}$ forecast per cell per timestep; horizontal axis is the level of Guassian error applied to the wind and traffic forecasts. The two lines represent 2—5m/s wind-speed and 5—10m/s wind-speed respectively.

Fig. 3 shows, as expected, that input forecast error affects pollution forecast quality. High wind-speeds clearly reduce prediction performance even if the forecasts have 0% error. However, in application scenarios, sustained high wind-speed also means reduced concern for pollution build-up, so this may have little effect on the system's ability to forecast when pollution levels become hazardous. Meanwhile, in comparison, decreasing wind and traffic forecast accuracy seems to have only a minor negative effect on pollution forecast accuracy. Experience and literature suggest that forecasts up to several hours ahead can then be expected to stay within the 10% error level. Similarly, traffic source and industry source forecasts can be expected to follow a relatively predictable daily pattern.

V. Discussion

We have presented a cellular-automaton based approach for the prediction of pollution flow in urban environments, designed for settings in which sensor stream data is available at multiple sites, collecting (minimally) pollutant and wind-speed data. The intended application scenario is one in which users regularly require a future prognosis of how the 'pollution map' will develop in the area of interest in, typically, the next few hours. Based on inspection of the resulting predictions, the users (such as local civic or health authorities) may then consider a range of mitigating actions. In this overall scenario, computation speed is an important factor, since it would typically be expected that the system could regularly absorb the latest data and refreshing its predictions. The choice of a CA approach was motivated by this need.

The general approach handles several challenges by making a variety of simplifications and workarounds. One of the main challenges is that, before pollution flow forecasts can be achieved at all, wind-speed forecasts and pollution generation forecasts (largely, this means traffic levels) also need to be forecast in advance, at multiple locations. Technologies are available to provide high-quality forecasts given the historical data streams, and at suitably rapidly timescales (we use the approach in [26]); however, in the likely event that sensor sites are relatively sparse in the area of interest, many of the required forecasts will need to be extrapolations. Meanwhile, one of the main simplifications (in the approach as described here) is to model the pollution flow only on a 2D plane, essentially ignoring atmospheric inversion and similar effects (other than the extent to which affects are captured indirectly in the learned parameters). Related simplifications are the absence of consideration of temperature, precipitation, pressure, other environmental parameters, and interaction between different pollutants.

Despite such simplifications, the 'track record' of experience with CAs suggested promise, which we find confirmed by the evaluation reported here. In the simple 'visualisation' use case, the forecast values for *all* cells would be visualised, and assumed to be of comparable accuracy to the forecasts at sensor sites. Our results suggest that – depending on reasonably accurate input forecasts – this would not present a misleading picture. Meanwhile, our approach can be extended naturally to include other environmental variables, and incorporate additional complexity, e.g. to make parameters context-dependent. Such additional complexity brings additional parameters to learn and extended learning times; on the other hand, CAs are highly amenable to parallelisation.

Acknowledgment

We are grateful to the University of the Thai Chamber of Commerce for sponsorship of Ms Benjavanich via a UTCC Scholarship, and to Innovate UK and Route Monkey Ltd for support of Ursani and Corne via KTP Partnership no. 9839.

References

[1] World Health Organization, "Ambient air pollution: a global assessment of exposure and burden of disease," WHO Publications, 2016, 121 pp. http://www.who.int/iris/handle/10665/250141

[2] Cohen, Aaron J., et al. "The global burden of disease due to outdoor air pollution." Journal of Toxicology and Environmental Health, Part A 68.13-14 (2005): 1301-1307..

[3] Yim, Steve HL, and Steven RH Barrett. "Public health impacts of combustion emissions in the United Kingdom." Environmental science & technology 46.8 (2012): 4291-4296..

[4] He, Kebin, Hong Huo, and Qiang Zhang. "Urban air pollution in China: current status, characteristics, and progress." Annual review of energy and the environment 27.1 (2002): 397-431.

[5] Vichit-Vadakan, Nuntavarn, et al. "Air pollution and respiratory symptoms: results from three panel studies in Bangkok, Thailand." Environmental Health Perspectives 109.Suppl 3 (2001): 381.

[6] Anderson, H. Ross, et al. "Air pollution and daily mortality in London: 1987-92." Bmj 312.7032 (1996): 665-669.

[7] Mage, David, et al. "Urban air pollution in megacities of the world." Atmospheric Environment 30.5 (1996): 681-686.

[8] Atzori, Luigi, Antonio Iera, and Giacomo Morabito. "The internet of things: A survey." Computer networks 54.15 (2010): 2787-2805.

[9] Moser, Whet. "What Chicago's 'Array of Things' Will Actually Do." Chicago Magazine, January 27 (2014).

[10] Bakıcı, T., E. Almirall, and J. Wareham. "A smart city initiative: the case of Barcelona." J. of the Knowledge Economy 4.2 (2013): 135-148.

[11] http://www.darksky.net

[12] Barratt, Rod. Atmospheric dispersion modelling: an introduction to practical applications. Routledge, 2013.

[13] Macdonald, Robert. "Theory and objectives of air dispersion modelling." Modelling Air Emissions for Compliance (2003): 1-27.

[14] Holmes, Nicholas S., and Lidia Morawska. "A review of dispersion modelling and its application to the dispersion of particles: an overview of different dispersion models available." Atmospheric environment 40.30 (2006): 5902-5928.

[15] Terada, Hiroaki, et al. "Atmospheric discharge and dispersion of radionuclides during the Fukushima Dai-ichi Nuclear Power Plant accident. Part II: verification of the source term and analysis of regional-scale atmospheric dispersion." Journal of Environmental Radioactivity 112 (2012): 141-154.

[16] Holmes, Nicholas S., and Lidia Morawska. "A review of dispersion modelling and its application to the dispersion of particles: an overview of different dispersion models available." Atmospheric environment 40.30 (2006): 5902-5928.

[17] Balzter, H., P.W. Braun, and W. Köhler. "Cellular automata models for vegetation dynamics." Ecological modelling 107.2 (1998): 113-125.

[18] Ghisu, Tiziano, et al. "An optimal cellular automata algorithm for simulating wildfire spread." Environmental Modelling & Software 71 (2015): 1-14.

[19] Nagel, Kai, and Michael Schreckenberg. "A cellular automaton model for freeway traffic." Journal de physique I 2.12 (1992): 2221-2229.

[20] Corne, David W., and Pierluigi Frisco. "Dynamics of HIV infection studied with cellular automata and conformon-P systems." Biosystems 91.3 (2008): 531-544.

[21] Nakano, Takaaki, J. Kasegawa, and Shin Morishita. "Coastal oil pollution prediction by a tanker using cellular automata." OCEANS'98 Conference Proceedings. Vol. 3. IEEE, 1998.

[22] Xiao-ping, Rui, et al. "Simulation of Point Source Pollution Diffusion Using a Velocity Field-cellular Automata Coupled Method." Information Technology Journal 12.20 (2013): 5424.

[23] Guariso, G. and V. Maniezzo. "Air quality simulation through cellular automata." Environmental Software 7.3 (1992): 131-141.

[24] Lauret, Pierre, et al. "Atmospheric dispersion modeling using Artificial Neural Network based cellular automata." Environmental Modelling & Software 85 (2016): 56-6

[25] Bäck, T., D.B. Fogel, and Z. Michalewicz, eds. Evolutionary computation 1: Basic algorithms & operators. Vol. 1. CRC press, 2000.

[26] Corne, David, et al. "Accurate localized short term weather prediction for renewables planning." Computational Intelligence Applications in Smart Grid (CIASG), 2014 IEEE Symposium on. IEEE, 2014.