Research Proposal

Keiran Suchak — 200888140

May 30, 2019

1 Title of Research Topic

Real-time Pedestrian Modelling: Implementing the Ensemble Kalman Filter for an Agent-Based Model

2 Research Questions and Statement

The work undertaken in this dissertation seeks to address the following research question: Can the Ensemble Kalman Filter method of data assimilation be used to improve the accuracy with which an agent-based model simulates pedestrian movement given observational data. The Ensemble Kalman Filter is a method of combining two streams of data; in our case, we aim to combine the states predicted by our pedestrian model with synthetic data as a proof of concept.

In order to answer the above question, the following objectives are set out:

1. Develop a general Ensemble Kalman Filter class in Python.
2. Apply the Ensemble Kalman Filter to our pedestrian model.
3. Compare the accuracy with which the Ensemble Kalman Filter implementation of the model simulates the pedestrian movement against the base model without data assimilation.

3 Description of Research

Knowledge of how people move around their environment can be made use of by both academics and policy-makers in the contexts of urban planning, event management and emergency response, particularly when considering urban environments. Furthermore, this may be of use to those interested in the social issues of mobility, inclusivity and accessibility of opportunities.

When considering such concepts, investigators often make use of modelling techniques. At their most fundamental, models represent our understanding of the system that we are studying — an understanding that may not be perfect (Stanislaw 1986). There exist modelling techniques for the simulation of how pedestrians move around urban spaces. However, these methods exist largely in isolation of the real-world — that is to say that whilst the simulations aim to reflect the real-world, there is no method by which we can incorporate up-to-date observations into these models to stop their divergence from reality.

Simulating pedestrian behaviour is often undertaken at a micro-scale, with such models typically aiming to model at the individual level or on a spatially fine-grained grid (Burstedde et al. 2001). One of the most prevalent simulation methods in this field is that of Agent-Based Modelling. Such methods consist of two key components: agents and environments. In an Agent-Based Model, we prescribe sets of rules by which individuals interact with each other and their local environment; as interactions take place on the micro-scale, we typically observe the emergence of structure at the macro-scale such as crowding (Batty et al. 2003) or lane formation (Liu et al. 2014). The evaluation of these rules is often not deterministic and instead introduces some element of randomness; these stochastic elements aim to emulate the variability of human behaviour. The introduction of such randomness in conjunction with an imperfect understanding of the phenomena at play, however, typically result in simulation runs diverging from the real system.

In constructing their models, agent-based modellers undertake a development process that involves model verification, validation and calibration. We can take these to mean the following:

- **Model verification**: The process of ensuring that the implementation is an accurate representation of the model (Xiang et al. 2005).
- **Model validation**: The process of ensuring that the chosen model is an accurate representation of the phenomenon that we wish to study (Crooks et al. 2008).
- **Model calibration**: The process of searching for model parameter values such that we can achieve model validation (Thiele et al. 2014).

Beyond this, modellers also make efforts to ensure that the initial model conditions are realistic by setting them based on historical data.

The practices of validation, calibration and setting initial model states based on historical data are appropriate for offline evaluations such as testing designs of new buildings or experimenting with different individual behaviours; however, when aiming to simulate events in real-time, this simple delays the inevitable divergence of the model from the real system. Furthermore, model parameters may be transient and thus require to be updated as time passes and the dynamics evolve.

Given the apparently inevitable divergence of stochastic simulations from the real system that they aim to model, one may alternatively turn to big data. Data is now being generated in higher volumes and at greater velocity than ever before (Chen et al. 2014); however there also exist issues with observational data from such systems. Whilst models typically allow us to simulate a whole system, observations are typically sparse in either time or space (or both); this is to say that we observations rarely provide complete coverage of events. We therefore seek a solution whereby we can integrate up-to-date observations into our models as the models continue to simulate the system.

One of the methods by which we can combine knowledge represented by our model with observations as they become available is through data assimilation techniques, which are most commonly used in the field of numerical weather prediction (Kalnay 2003). Such techniques are typically made up of two steps:

1. **Predict**: Run the model forward, estimating the state of the system, until new observations become available.

2. **Update**: Upon receipt of new observations, combine the model's estimate of the system state with the new data.

These steps are repeated iteratively in a cycle. It is important to note that just as there is error associated with the model, we also acknowledge that there is observational error associated with the data. The aim of incorporating the observations into the model is to improve the model accuracy with respect to the true system.

A large volume of work exists in which such techniques are applied to meteorological systems where the models are based on differential equations. Significantly less work exists in which data assimilation methods are applied to agent-based models (Wang et al. 2017) — in particular pedestrian models (Rai & Hu 2013, Wang & Hu 2013, 2015, Ward et al. 2016). Of the aforementioned studies, Wang & Hu (2013), Rai & Hu (2013) and Wang & Hu (2015) focus on implementing a data assimilation scheme known as the Particle Filter, whereas Ward et al. (2016) focus on implementing another scheme known as the Ensemble Kalman Filter to a very simple model; indeed, the latter piece of work admits that "the model used is model is relatively simple in comparison with typical ABMs and is not programmed in an object orientated framework". This dissertation therefore aims to expand on the pre-existing work by implementing the Ensemble Kalman Filter in conjunction with an agent-based model of pedestrians crossing a two-dimensional station from one side to the other.

4 Description of Methodology and Data

This dissertation focuses on the application of data assimilation schemes to agent-based models, in particular considering the Ensemble Kalman Filter (EnKF), which is a method derived from the Kalman Filter (KF). The aim of the KF is to combine two streams of information based on their uncertainties in order to produce an estimate for which the uncertainty is minimised (Kalman 1960). In doing so, it sequentially evolves the state and the covariance matrix (which characterises the state uncertainty). It provides an optimal solution in scenarios that fulfil the following conditions (Mandel 2009):

- The model is linear,
- The uncertainties are normally distributed.

These criteria are not always fulfilled; indeed, when considering agent-based models, we often find that the models are non-linear. Furthermore, as models grow in dimension, so does the cost of evolving the covariance matrix — this quickly becomes intractable for larger models.

In order to address some of these issues, the Ensemble Kalman Filter was developed (Evensen 2003, 2009). Critically, this approach allows us to maintain a non-linear model (Wikle & Berliner 2007). The EnKF acts as an approximation of the KF. This approximation is achieved by using an ensemble of sample state vectors to represent the state distribution. As such, the model state is represented by a matrix, $\mathbf{X}$, as follows:

$$\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_N], \tag{1}$$

where the state ensemble matrix, $\mathbf{X}$, consists of N state vectors, $\mathbf{x}_i$ ($\forall i \in (1, N)$. Similarly, the observations are represented by a matrix, $\mathbf{D}$, as follows:

$$\mathbf{D} = [\mathbf{d}_1, \dots, \mathbf{d}_N], \tag{2}$$

with each member of the data ensemble matrix, $\mathbf{D}$, being the sum of the original observation, $\mathbf{d}$, and a random vector, ε_i:

$$\mathbf{d}_i = \mathbf{d} + \varepsilon_i. \tag{3}$$

The random vector is drawn from an unbiased normal distribution:

$$\varepsilon \sim \mathcal{N}\left(0, \mathbf{R}\right), \tag{4}$$

where $\mathbf{R}$ is the data covariance matrix, representing the uncertainty in the observations.

The state of the system, $\mathbf{X}$, is iteratively forecast by the model until new observations are received; upon receipt of new data, the updated state ensemble matrix, $\hat{\mathbf{X}}$, is given by:

$$\hat{\mathbf{X}} = \mathbf{X} + \mathbf{K}\left(\mathbf{D} - \mathbf{H}\mathbf{X}\right), \tag{5}$$

where $\mathbf{K}$ is known as the Kalman gain matrix and $\mathbf{H}$ is known as the observation matrix. The observation matrix is responsible for converting the model state into observation state space. The Kalman gain matrix dictates the weighting applied to difference between the observations and the model state in updating the state, and is given by:

$$\mathbf{K} = \mathbf{Q}\mathbf{H}^T\left(\mathbf{H}\mathbf{Q}\mathbf{H}^T + \mathbf{R}\right)^{-1}, \tag{6}$$

where $\mathbf{Q}$ is the covariance matrix representing the uncertainty in the model state. The Kalman gain matrix therefore weights the update based on the proportion of the total uncertainty contributed by the model uncertainty.

This scheme will be applied to an agent-based model which simulates the motions of pedestrians moving in two dimensions across a station from one side to the other [1]. The data used for assimilation will be synthetic, and will be generated by the same model.

5 Timetable of Programme of Work

The work for this dissertation will be undertaken based on the following timetable:

- **6th May 2019**: Write code for general Ensemble Kalman Filter.
- **15th May 2019**: Test Ensemble Kalman Filter on pedestrian model.
- **24th May 2019**: First draft of dissertation.
- **14th June 2019**: Complete dissertation.

[1] https://github.com/Urban-Analytics/dust/tree/master/Projects/StationSim-py

6 Risk Assessment

The following factors are perceived to be risks to the success of the project:

- Time constraints (low risk, high impact): Plan to submit dissertation well in advance, allowing plenty of time for any exceptional circumstances.
- Lack of available data (low risk, high impact): Using synthetic data from the pedestrian model, which are easily generated.
- Lack of computing resources (low risk, medium impact): Initial test indicate that code can be run on laptops; previous arrangements have been made for access to the SEE server which provides a significant boost in computational performance if required.

Bibliography

Batty, M., DeSyllas, J. & Duxbury, E. (2003), 'The discrete dynamics of small-scale spatial events: agent-based models of mobility in carnivals and street parades', *International Journal of Geographical Information Science* **17**(7), 673–697.

Burstedde, C., Klauck, K., Schadschneider, A. & Zittartz, J. (2001), 'Simulation of pedestrian dynamics using a two-dimensional cellular automaton', *Physica A: Statistical Mechanics and its Applications* **295**(3-4), 507–525.

Chen, M., Mao, S. & Liu, Y. (2014), 'Big data: A survey', *Mobile networks and applications* **19**(2), 171–209.

Crooks, A., Castle, C. & Batty, M. (2008), 'Key challenges in agent-based modelling for geo-spatial simulation', *Computers, Environment and Urban Systems* **32**(6), 417–430.

Evensen, G. (2003), 'The ensemble kalman filter: Theoretical formulation and practical implementation', *Ocean dynamics* **53**(4), 343–367.

Evensen, G. (2009), 'The ensemble kalman filter for combined state and parameter estimation', *IEEE Control Systems Magazine* **29**(3), 83–104.

Kalman, R. E. (1960), 'A new approach to linear filtering and prediction problems', *Journal of basic Engineering* **82**(1), 35–45.

Kalnay, E. (2003), *Atmospheric modeling, data assimilation and predictability*, Cambridge university press.

Liu, S., Lo, S., Ma, J. & Wang, W. (2014), 'An agent-based microscopic pedestrian flow simulation model for pedestrian traffic problems', *IEEE Transactions on Intelligent Transportation Systems* **15**(3), 992–1001.

Mandel, J. (2009), 'A brief tutorial on the ensemble kalman filter', *arXiv preprint arXiv:0901.3725* .

Rai, S. & Hu, X. (2013), Behavior pattern detection for data assimilation in agent-based simulation of smart environments, *in* '2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)', Vol. 2, IEEE, pp. 171–178.

Stanislaw, H. (1986), 'Tests of computer simulation validity: what do they measure?', *Simulation & Games* **17**(2), 173–191.

Thiele, J. C., Kurth, W. & Grimm, V. (2014), 'Facilitating parameter estimation and sensitivity analysis of agent-based models: A cookbook using netlogo and r', *Journal of Artificial Societies and Social Simulation* **17**(3), 11.

Wang, M. & Hu, X. (2013), Data assimilation in agent based simulation of smart environment, *in* 'Proceedings of the 1st ACM SIGSIM Conference on Principles of Advanced Discrete Simulation', ACM, pp. 379–384.

Wang, M. & Hu, X. (2015), 'Data assimilation in agent based simulation of smart environments using particle filters', *Simulation Modelling Practice and Theory* **56**, 36–54.

Wang, P., Li, G., Ju, R. & Peng, Y. (2017), 'Random finite set based data assimilation for dynamic data driven simulation of maritime pirate activity', *Mathematical Problems in Engineering* **2017**.

Ward, J. A., Evans, A. J. & Malleson, N. S. (2016), 'Dynamic calibration of agent-based models using data assimilation', *Royal Society open science* **3**(4), 150703.

Wikle, C. K. & Berliner, L. M. (2007), 'A bayesian tutorial for data assimilation', *Physica D: Nonlinear Phenomena* **230**(1-2), 1–16.

Xiang, X., Kennedy, R., Madey, G. & Cabaniss, S. (2005), Verification and validation of agent-based scientific simulation models, *in* 'Agent-directed simulation conference', Vol. 47, p. 55.