Understanding Input Data Requirements and Quantifying Uncertainty for Successfully Modelling 'Smart' Cities *

Nick Malleson[1,3][0000−0002−6977−0615], Jonathan A. Ward[2][0000−0003−3726−9217], Alison Heppenstall[1,3][0000−0002−0663−3437], Michael Adcock[3], Daniel Tang[4], Jonathan Coello[4], and Tomas Crols[1,3][0000−0002−9379−7770]

[1] School of Geography, University of Leeds, LS2 9JT, UK
http://geog.leeds.ac.uk/
n.s.malleson@leeds.ac.uk
[2] School of Mathematics, University of Leeds, LS2 9JT, UK
http://maths.leeds.ac.uk
[3] Leeds Institute for Data Analytics (LIDA), University of Leeds, LS2 9JT, UK
http://lida.leeds.ac.uk
[4] Improbable, 30 Farringdon Road, London, EC1M 3HE, UK
http://www.improbable.io

Abstract. Agent-based modelling (ABM) is ideally suited to modelling the behaviour and evolution of social systems. However, there is inevitably a high degree of uncertainty in projections of social systems – input data are noisy and sparse, and human behaviour is itself extremely uncertain – one of the key challenges facing the discipline is the quantification of uncertainty within the outputs of agent-based models. Without an adequate understanding of model uncertainty, or a means to better constrain models to reality, simulations will naturally diverge from the target system. This limits the value of their predictions. This paper presents ongoing work towards methods that will (i) allow real-time data to be assimilated into models to reduce the uncertainty of their predictions and to (ii) quantify the *amount* of data (including overall volume as well as spatio-temporal granularity and regularity) that are required for successful assimilation (i.e. to model the system within an acceptable level of uncertainty). Specifically, this project emulates a simple system of pedestrians who all move towards an exit. The paper reports on initial experiments to constrain the range of possible model outcomes using Bayesian inference techniques as implemented in a new probabilistic programming library. Ultimately the project aims to provide valuable information about the number and type of sensors that might be required to model movements of humans around real urban systems.

* This work was supported by a European Research Council (ERC) Starting Grant [number 757455], a UK Economic and Social Research Council (ESRC) Future Research Leaders grant [number ES/L009900/1], an ESRC-Alan Turing Fellowship [ES/R007918/1] and through an internship funded by the UK Leeds Institute for Data Analytics (LIDA).

XX REWRITE ABSTRACT

Keywords: Agent-based modelling · Uncertainty · Data assimilation · Bayesian inference

1 Introduction and Background

Paper plan:

- Aims: 1. explore the amount of data required to successfully parameterise a pedestrian simulation. 2. experiment with the efficacy of Bayes nets and probabilistic programming for doing this.
- 1. Experiment with a simple model - how much data do we need to give to the Bayes Net in order to successfully find the threshold?
  1. Give it every iteration
  2. Give it 1 iteration
  3. ... how many before it does not have enough data?
- 2. Experiment with ABM.
  1. Give it every iteration
  2. Give it 1 iteration
  3. ... how many before it does not have enough data?
  4. Give it inflows and outflows
  5. Give it density of a particular square.

Individual-level modelling approaches, such as agent-based modelling (ABM), are ideally suited to modelling the behaviour and evolution of social systems. This is especially true in the context of modern 'smart' cities, where large volumes of data, supported by innovative 'big' data analytics, can be leveraged to better understand and capture the characteristics of the underlying systems. However, there is inevitably a high degree of uncertainty in projections of social systems – input data are noisy and sparse, and human behaviour is itself extremely uncertain – one of the key challenges facing the discipline is the quantification of uncertainty within the outputs of these models.

This work focusses on the simulation of *urban flows*, i.e. the movement of people around urban areas over relatively short time scales (minutes and hours). It presents ongoing work towards methods that will (i) allow real-time data to be assimilated into models to reduce the uncertainty of their predictions and (ii) to quantify the *amount* of data (including overall volume as well as spatio-temporal granularity and regularity) that are required for successful assimilation. In other words, given some human system and a simulation of that system (we study the movement of pedestrians in this case), how much data are required from the real system in order to prevent the model uncertainties from causing the simulation to rapidly diverge from reality? With too little data it will be impossible to reliably constrain the model to reality, but how much is too little? Is one well-placed footfall counter sufficient to capture the dynamics of the system, or in reality would it be necessary to track the actual movements of a large proportion of the individual people? The hypothetical model used here is a simple system of pedestrians, each of whom move from an entrance towards one of two exits. This is analogous to a train arriving at a train station and passengers moving across the concourse to leave. The model environment is illustrated in Figure 1.

Very limited prior work has been conducted in this area. Some authors have attempted to conduct data assimilation, but use agent-based models that are simple in the extreme [**?**,**?**]. Here, a more advanced model will be used in order to test how well the various methods handle complex features such as emergence and feedback. Others have developed more complex agent-based models [**?**] but those must remain mathematically coupled to an aggregate proxy model which limits the flexibility of the underlying agent-based model. The most similar work is that of [**?**], who attempt to assimilate data into a model of peoples' movement in buildings. They do this by running an ensemble of models and re-starting each one using new input conditions that are created each time new data become available. The main difference between the approach in [**?**] and this paper is that here the aim is to eventually develop methods that are able to assimilate new data that automatically *alter the state* of the simulation while it is running. This is more analogous with data assimilation techniques in other fields [**?**].

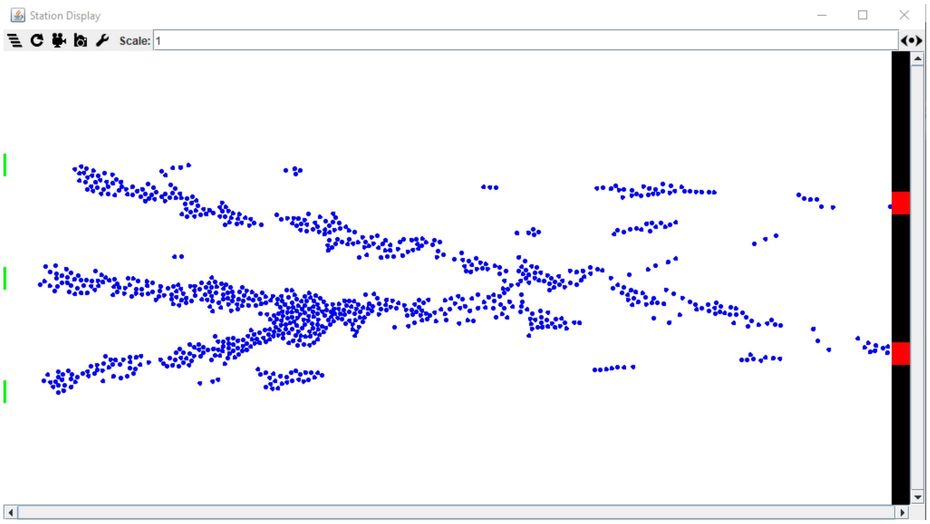


Fig. 1: A snapshot of the simple, hypothetical model that is used here. Agents arrive from the green entrances on the left and move towards the red exits on the right.

2 Methods

2.1 Overall Approach

This paper presents a work in progress, whose overall proposed approach is to:

1. Develop a agent-based simulation of pedestrian movements from an entrance to an exit using a simple social forces framework (similar to [**?**]) to control the behaviour of the agents (see Figure 1);
2. Run the simulation to generate hypothetical 'truth' data, our equivalent reality created with synthetic data. We assume this equivalent reality represents the real underlying system. We sample from this in a manner analogous to that of using sensors of peoples' movements to sample from the real world;
3. Re-run the simulation with a different random seed and use samples of varying resolution from the 'truth' data to reduce the uncertainty of the new simulation;
4. Quantify the volume (total number of samples), granularity (amount of aggregation) and regularity (the number of samples per time period) that are required to successfully model the hypothetical 'real' data.

This paper presents the preliminary results up to stage 3, with the ultimate aim of quantifying the amount of data required to successfully constrain the simulation being immediate future work.

2.2 Preliminary Results: Constraining Model Uncertainty

Our initial experiments use a prototype probabilistic programming library to perform parameter estimation using Bayesian inference (see [**?**] for a useful discussion about probabilistic programming and Bayesian inference). This work is a precursor to performing data assimilation (i.e. step 4). The probabilistic programming approach treats the pedestrian simulation as a black box, whose only *input* is a list of random numbers that are used in the simulation whenever a probabilistic decision is made and whose *output* is the number of agents in the system at each time step. All internal simulation parameters are fixed. The simulation output is deterministic given the same input, but different inputs result in different outputs, and so stochastic simulations can be performed by choosing the input at random. Figure 2 illustrates the method.

We create truth data, i.e. agent counts over time, from a single model run using a model input (a list of random decimal numbers) sampled from a Gaussian distribution. Then, using a noisy observation of the truth data, we use the probabilistic programming library to create a Bayesian network from which we can compute the posterior distribution of the input given the noisy observation. We then sample from this distribution using Metropolis-Hastings, resulting in an ensemble of model realisations in which the uncertainty of the input, and hence the output, is constrained.

Figure 3 compares the results of the sampling with and without the incorporation of the truth data. It is clear that when the Bayesian network makes

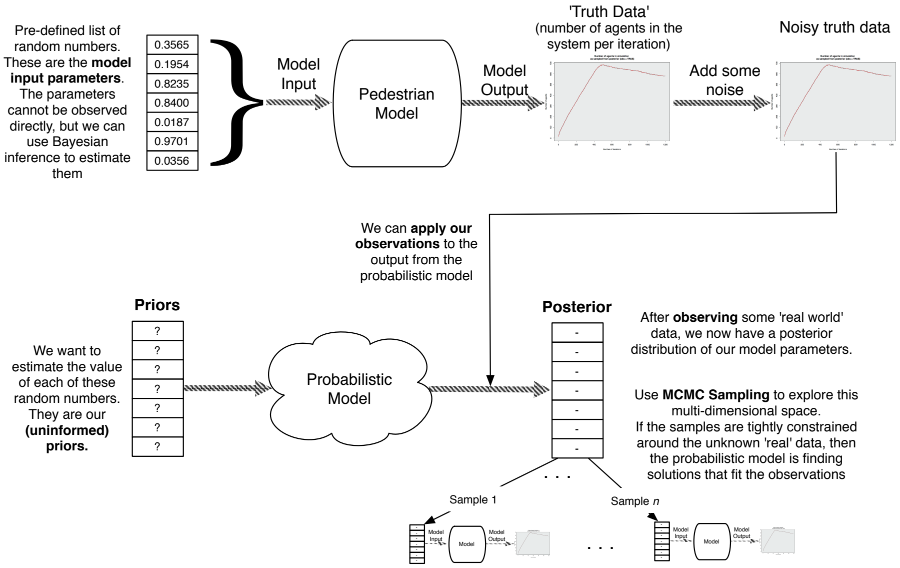
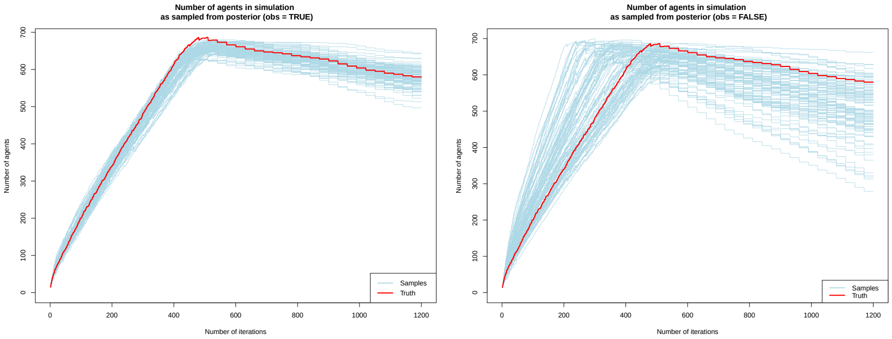

Pre-defined list of random numbers. These are the **model input parameters**. The parameters cannot be observed directly, but we can use Bayesian inference to estimate them
0.3565
0.1954
0.8235
0.8400
0.0187
0.9701
0.0356
Model Input
Pedestrian Model
Model Output
'Truth Data'
(number of agents in the system per iteration)
Add some noise
Noisy truth data
We can **apply our observations** to the output from the probabilistic model
Priors
We want to estimate the value of each of these random numbers. They are our **(uninformed) priors.**
?
?
?
?
?
?
?
Probabilistic Model
Posterior
-
-
-
-
-
-
-
After **observing** some 'real world' data, we now have a posterior distribution of our model parameters.
Use **MCMC Sampling** to explore this multi-dimensional space.
If the samples are tightly constrained around the unknown 'real' data, then the probabilistic model is finding solutions that fit the observations
. . .
Sample 1
Sample *n*
Model Input
Model
Model Output
. . .
Model Input
Model
Model Output

Fig. 2: An illustration of the procedure to perform parameter estimation.

use of the observations from the 'truth' model, the output of each model (i.e. sample of the posterior distribution) is much more similar to the truth data. In other words, the procedure is more accurately estimating the list of random numbers (the prior distribution) that were initially used as input to generate the 'truth' data. Although this result is not of value in isolation, it is very useful as a proof-of-concept. It shows that Bayesian inference on an agent-based model that has reasonably complex characteristics (namely the emergence of crowding due to agent interactions) is able to perform parameter estimation. More rigorous data assimilation is a relatively small step.

3 Link to ABMUS Workshop Themes

This paper addresses a core aspect of the ABMUS workshop theme; that of the *trust* that we can have in model outputs. The conference recognises challenges in "designing, developing and implementing **trusted** models that can be used by industry and governments to enhance decision-making". By adapting existing methods that are aimed at reducing uncertainty in models through the incorporation of up-to-date data, this work advances the methodology towards more a rigorous representation of real urban systems that will be more acceptable for policy use.



(a) constrained to observations

(b) without observations

Fig. 3: Results of sampling the posterior with observations (3a) and without (3b). When the 'truth' data are used to constrain the posterior distribution, the sampling routine is much better able to estimate the input model state, so the outcomes of the samples are much closed to the 'truth' data.