SUPPLEMENTARY MATERIAL: POPULATION SYNTHESIS AS SCENARIO GENERATION FOR SIMULATION-BASED PLANNING UNDER UNCERTAINTY

Joel Dyer
University of Oxford
joel.dyer@cs.ox.ac.uk

Arnau Quera-Bofarull
University of Oxford

Nicholas Bishop
University of Oxford

J. Doyne Farmer
University of Oxford

Anisoara Calinescu
University of Oxford

Michael Wooldridge
University of Oxford

February 7, 2024

ABSTRACT

This document contains supplementary material that accompanies Dyer et al. [2023a].

Keywords agent-based model · scenario planning · synthetic data

1 Further experimental details

1.1 Axtell's model of the emergence of firms

We provide further details on the Axtell firm model [Axtell, 1999], discussed in Section 4.1 of Dyer et al. [2023a]. As discussed there, we begin with a population of N agents, each of which is a labourer in an economy and works with some effort level $e^t_n \in [0, 1]$ at time $t \in [0, 1]$. At any given time t, agent n belongs to firm $\mathrm{f}^t_n \in \{1, \ldots, N\}$, where the model is initialised such that $\mathrm{f}^0_n = n$ (i.e., the firms are singleton sets in with each agent comprising their own firm). Over time, each agent reevaluates their position in the labour market at some agent-specific rate $\rho_n \in \mathbb{R}_{\geq 0}$, at which point the contemplative agent considers the trade-off – determined by the agent-specific parameter $\nu_n \in [0, 1]$ – between the utility they derive from (i) participating in firm f^t_n and sharing in its output, and (ii) the disutility $1 - e^t_n$ experienced from contributing to the firm's productive activities at their current effort level e^t_n.

In particular, the occurrence of these reevaluations is modelled as a separate Poisson process for each agent, where the intensity of the Poisson process corresponding to agent n is ρ_n. The model is therefore a continuous time stochastic process. Additionally, the total utility U^t_n experienced by agent n at time t is a result of the aforementioned trade-off between the agent's preference for income u^t_n (which it receives from the firm's output) and leisure (which it sacrifices in her exertion of effort directed towards the firm's productive activities), and is represented with a Cobb-Douglas function:

$$U^t_n = \left(u^t_n\right)^{\nu_n} \left(1 - e^t_n\right)^{1-\nu_n}. \tag{1}$$

The utility agent n experiences from sharing in the output of the firm's activities is modelled as

$$u^t_n = \frac{O_{\mathcal{F}^t_{\mathrm{f}^t_n}}}{\left|\mathcal{F}^t_{\mathrm{f}^t_n}\right|}, \tag{2}$$

where

$$O_{\mathcal{F}^t_j} = aE_{\mathcal{F}^t_j} + bE^{\beta}_{\mathcal{F}^t_j}, \tag{3}$$

is the total output of firm j at time t, with $\mathcal{F}_j^t = \{i : 1 \leq i \leq n, \mathrm{f}_i^t = j\}$ being the subset of the population belonging to firm j at time t,

$$E_{\mathcal{F}_j^t} = \sum_{i \in \mathcal{F}_j^t} e_i^t \tag{4}$$

being the total effort exerted by agents in firm j at time t, and where $\boldsymbol{\omega} = (a, b, \beta) \in \mathbb{R}^2_{>0} \times [1, \infty)$ are the model's structural parameters. That is, all agents in the firm share equally in the output of the firm, where the total output is a function of the total effort exerted by agents in the firm and is modelled as in Equation (3).

By considering their total utility, Equation (1), in each of the following cases, the contemplative agent decides to either: (i) remain at the same firm; (ii) join a neighbouring firm; or (iii) start a new firm. In each case, the agent is also permitted to readjust its effort levels. Defining the discrete action set

$$\mathcal{C} = \{\text{remain at current firm}, \text{move to one of its } v \text{ neighbours' firms}, \text{start a new firm}\}, \tag{5}$$

the agent's course of action – whether to update its effort level, and/or move to a neighbour's firm or begin its own firm – is the one that maximises its utility over the joint action set

$$\mathcal{S} = [0, 1] \times \mathcal{C}, \tag{6}$$

resulting in a (potentially) new effort level e_n^t and movement to a new firm (i.e., a change in value of f_n^t). The optimal effort level for each discrete choice $c \in \mathcal{C}$ is determined by Equation 5 of Axtell [1999], which takes the form

$$e_n^{t,c} = \max\left\{0, \frac{-a - 2b(E_{-n}^c - \nu_n) + \sqrt{a^2 + 4b\nu_n^2(1 + E_{-n}^c)\left(a + b(1 + E_{-n}^c)\right)}}{2b(1 + \nu_n)}\right\} \tag{7}$$

where E_{-n}^c is the sum of efforts from all other members of the firm to which n would belong under choice $c \in \mathcal{C}$.

We place each agent on a ring and form a circle graph by placing edges between each agent and its $v = 2$ neighbours. More general topologies can be used, but we consider this topology for simplicity. This topology is fixed and does not change over time.

1.1.1 The model dynamics

As discussed above, agents make decisions about whether to remain at their current firm, move to a neighbouring agent's firm, or to start a new firm, according to whichever choice maximises their utility. The agent may simultaneously update its effort level.

1.1.2 Model simulation

Algorithm 1: Outline of procedure for forward simulating the Axtell firm model.

Data: Agent-level attributes $\mathbf{z}_n \sim \iota(\cdot \mid \boldsymbol{\theta}), n = 1, \ldots, N$, from the distributions in Equation 13 of Dyer et al. [2023a]; structural parameters $\boldsymbol{\omega}$
Result: Simulated output $\mathbf{x}$
Populate model with N agents with the attributes $\mathbf{z}_n$, $n = 1, \ldots, N$;
Set $\mathrm{f}_n^0 = n$, $n = 1, \ldots, N$;
Set $t = 0$;
while $t < 1$ **do**
  Draw next event time as $t + \Delta t$, $\Delta t \sim \mathrm{Exp}(1/\sum_{n=1}^N \rho_n)$;
  if $t + \Delta t > 1$ **then**
    break
  end
  Choose corresponding agent as $n \sim \mathrm{Categorical}\left(\rho_n / \sum_{n'=1}^N \rho_{n'}\right)$;
  Evaluate Equation (7) for all discrete actions $c \in \mathcal{C}$;
  Choose action (i.e., new effort level and firm) that maximises utility given in Equation (1);
end

In summary, the model can be simulated as in Algorithm 1.

1.2 Binary opinion dynamics

Within each simulation, we generated a new random social network from the space of Barabasi-Albert random graph models in which 2 new edges added at each iteration of the graph-growing algorithm. This was not included as a structural parameter, and was held fixed during all experiments.

1.3 Training details

To target the threshold-based proposal distribution we consider with sequential Monte Carlo (TBS-SMC in the main text) we use the implementation of SMC-approximate Bayesian computation provided in the SBI software package [Tejero-Cantero et al., 2020]. To generate Figures 2 and 4 in Dyer et al. [2023a], we start with a population of the best $5 \cdot 10^3$ particles from an initial population drawn from 10^4 simulations. The total simulation budget is fixed to 10^5 and we decay the acceptance threshold ϵ with a rate of 0.8, which results into $\epsilon = 0.5$ at the end of the simulation budget.

For variational optimisation with a normalising flow (VO-NF in the main text), we consider $\mathcal{Q}$ to be the space of densities spanned by an autoregressive normalising flow with 8 transformations parameterized by 64 hidden units each. We make use of the NORMFLOWS Stimper et al. [2023] library to implement the flow and the BLACKBIRDS [Quera-Bofarull et al., 2023b] package to train the variational proposal distribution. As an optimiser, we use ADAMW Loshchilov and Hutter [2019] with a learning rate of 10^{-3}. Lastly, the expectation in Equation 10 of Dyer et al. [2023a] is evaluated with 50 Monte-Carlo samples.

1.4 Restricting normalising flow to target domain

In the simulation studies presented in Section 4 of Dyer et al. [2023a], we have considered normalising flows as a candidate family $\mathcal{Q}$. Such neural density estimators have previously been used in agent-based modelling to facilitate model calibration [see e.g., Dyer et al., 2024, 2022, Dyer, 2022, Quera-Bofarull et al., 2023a, Dyer et al., 2023b]. The output domain of a normalising flow is typically $\mathbb{R}^n$, where n is the output dimension. The parameters of probability distributions, however, often has bounded support, so the output of the flow needs to be constrained to the appropriate range. For instance, the variance of a normal distribution can only be a positive number and the probability of a Bernoulli distribution must be between 0 and 1. To remedy this, we add an additional transformation at the end of the normalising flow's chain of transformations that restricts the output domain to the desired one. The particular transformation we consider is

$$\boldsymbol{x}' = f(\boldsymbol{x}) = \boldsymbol{x}_{\min} + \sigma(\boldsymbol{x})(\boldsymbol{x}_{\max} - \boldsymbol{x}_{\min}), \tag{8}$$

where $\sigma(\boldsymbol{x})$ is the sigmoid function applied to each dimension such that

$$\sigma(x_i) = \frac{1}{1 + e^{-x_i}}, \tag{9}$$

and $\boldsymbol{x}_{\min}$ and $\boldsymbol{x}_{\max}$ are the minimum and maximum values the parameters $\boldsymbol{x}$ can take. Note that this technique is limited to restricting the parameter space to a hyperrectangle, which is sufficient for the cases considered in this work. It is important to also transform the probability distribution accordingly, in order to maintain the properties of normalising flows. If we simply sampled from the flow and transform the sample according to Equation 8, the regularisation term in Equation 8 of Dyer et al. [2023a] would not have the desired effect, since the flow could maximize entropy by assigning probability mass to parts of the parameter space that are mapped to very similar values after applying Equation 8. To this end, we also specify the inverse transformation, $f^{-1}(x)$, and the log determinants of the Jacobians,

$$\begin{aligned} f^{-1}(\boldsymbol{x}) &= \log\left(\frac{\boldsymbol{x} - \boldsymbol{x}_{\min}}{\boldsymbol{x}_{\max} - \boldsymbol{x}}\right) \\ \log\ |\det J_f| &= \sum_i \left(\log \sigma(x_i) + \log \sigma(-x_i) + \log(x^i_{\max} - x^i_{\min})\right) \\ \log\ \left|\det J_{f^{-1}}\right| &= -\sum_i \left(\log(x^i - x^i_{\min}) + \log(x^i_{\max} - x^i)\right), \end{aligned} \tag{10}$$

so that probability distributions can be transformed accordingly.

References

Robert Axtell. *The Emergence of Firms in a Population of Agents: Local Increasing Returns, Unstable Nash Equilibria, and Power Law Size Distributions*. 1999.

J Dyer, A Quera-Bofarull, N Bishop, JD Farmer, A Calinescu, and M Wooldridge. Population synthesis as scenario generation for simulation-based planning under uncertainty. Association for Computing Machinery, 2023a.

J Dyer, P Cannon, J Farmer, and S Schmon. Black-box bayesian inference for agent-based models. *Journal of Economic Dynamics and Control*, 2024.

Joel Dyer. *Likelihood-free Bayesian inference for dynamic, stochastic simulators in the social sciences*. PhD thesis, University of Oxford, 2022.

Joel Dyer, Patrick Cannon, J Doyne Farmer, and Sebastian M Schmon. Calibrating agent-based models to microdata with graph neural networks. In *ICML 2022 Workshop AI for Agent-Based Modelling*, 2022.

Joel Dyer, Arnau Quera-Bofarull, Ayush Chopra, J Doyne Farmer, Anisoara Calinescu, and Michael Wooldridge. Gradient-assisted calibration for financial agent-based models. In *Proceedings of the Fourth ACM International Conference on AI in Finance*, pages 288–296, 2023b.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Arnau Quera-Bofarull, Ayush Chopra, Anisoara Calinescu, Michael Wooldridge, and Joel Dyer. Bayesian calibration of differentiable agent-based models. *arXiv preprint arXiv:2305.15340*, 2023a.

Arnau Quera-Bofarull, Joel Dyer, Anisoara Calinescu, J Doyne Farmer, and Michael Wooldridge. Blackbirds: Black-box inference for differentiable simulators. *Journal of Open Source Software*, 8(89):5776, 2023b.

Vincent Stimper, David Liu, Andrew Campbell, Vincent Berenz, Lukas Ryll, Bernhard Schölkopf, and José Miguel Hernández-Lobato. Normflows: A PyTorch package for normalizing flows. *Journal of Open Source Software*, 8(86): 5361, 2023. doi:10.21105/joss.05361.

Alvaro Tejero-Cantero, Jan Boelts, Michael Deistler, Jan-Matthis Lueckmann, Conor Durkan, Pedro J. Gonçalves, David S. Greenberg, and Jakob H. Macke. Sbi: A toolkit for simulation-based inference. *Journal of Open Source Software*, 5(52):2505, August 2020. ISSN 2475-9066. doi:10.21105/joss.02505.