# Numerical Politics

Numerical experiments in the politics of interacting, autonomous agents

Daniel Tang

March 22, 2024

# Chapter 1

# Introduction

Numerical politics is a new approach to the study of politics in which we perform numerical experiments on simulated societies in order to gain an understanding of how organised, collective behaviour can emerge among interacting agents. This github repo (`https://github.com/danftang/NumericalPolitics`) presents a set of numerical experiments that allows us to build an understanding of this emergent behaviour while also presenting the software and numerical techniques necessary to perform these experiments. Our approach will be to start with very simple agents in simple environments and gradually introduce more complex behaviours and complex environments. This will allow us to build up our theoretical understanding and introduce new numerical techniques in a logical order.

Eventually, this repo is intended to provide the software necessary to allow anyone to set up a numerical "laboratory" and start studying simulated societies. In these pages we'll also discuss how the practice of numerical politics can contribute to our understanding of how to effectively structure and govern real-world societies.

## 1.1 The framework: thinking clearly about collective behaviour

The subject matter of numerical politics is the collective behaviour of many interacting agents. Agents interact by passing messages between eachother. Each agent has a number of "channels" for receiving messages and any agent with the channel's ID can send a message down the channel. When an agent receives a message, it can respond by updating its internal state and/or sending yet more messages. More formally, the behaviour of an agent can be defined as a probability distribution, $P(a\|m, c, \psi)$, which is the probability that an agent in state $\psi$ performs action $a$ in response to receiving message $m$ in channel $c$. The action $a$ defines the agent's new internal state and/or a set of messages passed to other agent's channels at some time after receipt of the message. At first sight this may seem a bit abstracted from real-world application, but we

choose it because it's a very flexible formalism that, on the one hand can easily be adapted to all applications, while on the other makes it easier to parallelise computations.

Our particular interest here will be to make predictive statements about individual and collective wellbeing of the agents. For this we assume there exists a function, $W(\psi)$, that is some measure of the wellbeing of an agent in state $\psi$ and define the collective wellbeing of a set of agents, $S$, as the sum of individual wellbeings $\Omega = \sum_{\psi \in S} W(\psi)$. It should be the subject of much debate exactly what the function $W(\psi)$ ought to be.

Notice that in this definition there is no mention of government. A goverment, if there is one, is encoded within the behaviours of the agents and is part of the model, as opposed to it being an exogenous actor imposing "policy interventions" on the agents. In this way, a government is best thought of as an emergent property of the agents' behaviours. We choose to make government endogenous because our interest here is *not* to simulate specific policy interventions but to understand the fundamental principles of organised, collective behaviour.

# Chapter 2

# Sugar and spice world (version 1)

We begin with an anarchic world with two commodities: sugar and spice. Suppose half the agents can farm only sugar and half can farm only spice, each at a rate of one unit per unit time. Suppose the agents randomly encouter eachother, whereupon each agent can either offer to trade or try to steal the other agent's crop. If both agents offer to trade then half the crop of each agent is swapped, however, if one agent offers to trade and the other tries to steal then the stealing agent gets half the other's crop and the other is left with only half its original crop. If both try to steal then they're both unsuccessful and no food is transferred. This is the classic prisoner's dilemma, an agent's wellbeing after an interaction is a function of the two agent's actions and is given in table 2.1.

This world is simple enough that we can see immediately that the optimal collective wellbeing occurs when all agents trade. In this case, the average wellbeing of all agents is 3. However, under what circumstances will agents reach this optimum?

If we assume the agents are Q-learning then we can ask what kinds of society do the agents create for themselves using their learning. More mathematically we can look at the society as a dynamic system and ask about the distribution of wellbeing on the attractors. In the special case where the attractor is a point, we have a stable society where no amount of learning from further encounters will change any agent's policy.

| Agent 1 | Agent 2 | Agent 1 wellbeing |
|---|---|---|
| trade | trade | 3 |
| trade | steal | 0 |
| steal | trade | 4 |
| steal | steal | 1 |

Table 2.1: The wellbeing of an agent after an interaction

## 2.1 Zero memory agents

If agents have no memory of previous encounters then each encounter is a simple prisoner's dilemma situation. The state of a Q-learning agent is just the Q-values of trade, $Q_t$ and steal, $Q_s$. Equilibrium is when

$$Q_t = 3P(t) + r\max(Q_t, Q_s)$$
$$Q_s = 4P(t) + P(s) + r\max(Q_t, Q_s)$$

but

$$Q_s - Q_t = P(t) + P(s) = 1$$

so $Q_s > Q_t$ irrespective of the other agent's behaviour so a zero memory Q-learning agent will always learn to steal, leading to a society where all agents try to steal and every agent is much worse off than in a trading society, with an average wellbeing of 1.

So, memoryless Q-learning agents get stuck in an equilibrium that is far from optimal both collectively and individually.

What needs to change in order to improve these agent's lives?

## 2.2 One step memory agents

If we give the agents the ability to remember the last encounter they had with another agent (if this isn't the first encounter) then the dynamics of the society gets much more interesting.

### 2.2.1 Experiment 1: two agents

We start with the simple case of just two agents. The agents begin with a high probability of exploring the policy space (by choosing a random action with uniform probability), and this probability reduces exponentially with time.

Experiment 1 in `https://github.com/danftang/NumericalPolitics` shows that under these circumstances, the agents quickly learn to trade by both adopting the policy described in table 2.2. The first three entries in the table have analogues in human behaviour, but the last is a little unintuitive: if we both tried to steal from eachother last time, then this time I'll try to trade. This is key to the success of the policy as it means that whatever state the agents get into they quickly revert to mutual trading. As the exploration probability tends to zero, this society tends to the optimum of always trading, while remaining unexploitable (if I always try to steal from an agent with this policy, we'll flip between mutual stealing and me stealing from the agent, but my average wellbeing will be 2.5, less than if I take on the policy in table 2.2).

Note that the agents do not learn the tit-for-tat policy: if you tried to steal from me last time, I'll try to steal from you this time, but if you traded with me last time, I'll try to trade with you again. Mutual adoption of this strategy has three stable states: mutual trading, mutual stealing and alternating unilateral stealing

| My last move | Your last move | My next move | Human trait |
|---|---|---|---|
| trade | trade | trade | mutual-benefit |
| trade | steal | steal | revenge |
| steal | trade | steal | exploitation |
| steal | steal | trade | ? |

Table 2.2: The optimum behaviour of an agent with one-step memory

$TS \rightarrow ST \rightarrow ....$ However, if the agents have a non-zero probability of exploring the policy space, then mutual tit-for-tat is not stable for Q-learners because, once in a run of mutual stealing, the Q-value of stealing again (and so getting into a long period of reward 1) is half the Q-value of trading (and so getting into a long period of reward $(4+0)/2 = 2$) so it makes sense to unilaterally trade with a tit-for-tat agent after mutual stealing. Ultimately, a pair of Q-learners will eventually learn to mutually adopt the behaviour in table 2.2.

### 2.2.2 Experiment 2: many agents

Experiment 2 shows what happens as this society grows. We assume agents are able to recognize and remember the history of the first $n$ agents they meet, but after that everyone is a stranger.

# Chapter 3

# Social graph: Emergent feudalism

Agents can own land, corn and labour and are connected by directed edges representing social ties to other agents. Social ties have an associated "context/type" which is just an integer ID.

The life of an agent consists of the repetition of a meta-game which consists of deciding which game to initiate from a set of available games, based on current state. Between games, agents play games initiated by other agents. At the end of each self-initiated game an agent's body goes through a timestep (labour is reset to 1, hunger goes up) and the game repeats.

One of the games is the "eat" game. At the start of the game, corn += 100*min(land,labour) and labour is reset to 1, the agent then gets to decide how much corn to eat, and receives a reward based on amount eaten and bodily hunger.

One of the games could be the "do-nothing" game.

For each of an agent's social ties, there is an "encounter" game. An encounter consists of a turns-based game with another agent, with the non-initiating agent taking the first move. During an encounter an agent can: - give land - give corn - give labour - hit other with a stick - say 0 - say 1 - say goodbye

An encounter ends when both agents say goodbye, or the first-mover begins the conversation with goodbye (effectively deciding not to initiate an encounter). This ensures that a game is entered into with mutual consent, which means that both agents expect to gain from the encounter under prior assumptions and both know this and know the other knows etc.

If the expectation of choosing an encounter with a given tie falls below some threshold, the tie is removed and replaced by a new tie by uniform-randomly choosing a tie context and choosing another agent by following links uniform-randomly in the social graph and stopping with a fixed probability if an edge of that context to that agent doesn't already exist.

Agents begin with one unit of land each, and a random set of edges. [for a small

number of agents, could start with a fully connected graph and have only edge attrition]

## 3.1 Implementation

[There exists the following quality values: - expected quality of the meta-game - expected quality of a sub-game - expected quality of a type of tie each can be for a given state or integrated over a distribution of states.]

So, suppose agents have an approximating Q-function for each social tie (both inward and outward). Within an encounter, discount factor is 1.0. We assume there is no intrinsic reward for starting an encounter and no state change, so the Q-value of starting an encounter with an outward tie is just the Q-value of the current state having started the encounter times the encounter initiation discount factor.

We suppose there is a public buffer, B, of encounter behaviours randomly selected from the population, which represents third party observation of others, stories in the culture, TV etc. This provides a prior over behaviour for both parties in an encounter. If two agents willingly decide to enter into an encounter based on the prior behaiours, then it would seem that conforming to priors is in both agent's interests. [Agents could have different views of the buffer. Perhaps agents close in the social graph are more likely to see eachother's behaviour. The behaviour of direct neighbours could be identifiable. This would also allow investigation of social effects of story-telling and mass communication.] The public buffer could be used to:

1. model other in self-play
2. compare against ones own strategy (given a prior over state pairs at the start of an encounter, a policy induces a distribution over histories).
3. provide a prior over behaviour on initiation of new social ties.
4. train a separate approximator of the "empirical policy" $P(a|H)$.

When a new social tie is created, we create a new Q-function to model that tie. The initial state is generated by self-play against the empirical policy (trained on B), then subsequently learns from behaviour of the individual tie.

### 3.1.1 The imitate/innovate strategy

In this strategy we assume all agents share our policy. So, our policy plays the dual role of informing our behaviour and predicting other's. This is associated with two evidence sources: the buffer of public behaviour and the agent's private buffer of Q-learning timesteps, an agent can put different weights on the losses from these two sources, on a sliding scale between immitator and innovator (and can even adjust this over time). An agent can also use self-play against its own policy to train. In order to train against B it is necessary to know a distribution over states at the start of an encounter. This can be learned

by assuming no correlation and learning two distributions over states for any parameterised family of distributions.

Given a history, $H$,

$$P(\pi_x, \pi_y, \sigma_x, \sigma_y|H) = \frac{P(H|\pi_x, \pi_y, \sigma_x, \sigma_y)P(\pi_x)P(\pi_y)P(\sigma_x)P(\sigma_y)}{P(H)}$$

but this can be separated into players.

$$P(\pi_x, \pi_y, \sigma_x, \sigma_y|H) = P(\pi_x, \sigma_x|H)P(\pi_y, \sigma_y|H)$$

Letting $H = a_0, ...a_m$

$$P(\pi_x, \sigma_x|H) = \frac{\prod_{i=0}^{\frac{m}{2}} \pi_x(a_{2i}|\sigma(a_{<2i}, \sigma_x))P(\pi_x)P(\sigma_x)}{\int \prod_{i=0}^{\frac{m}{2}} \pi'(a_{2i}|\sigma(a_{<2i}, \sigma'_x))P(\pi'_x)P(\sigma'_x)d\pi'_x d\sigma'_x}$$

$$P(\pi_y, \sigma_y|H) = \frac{\prod_{i=0}^{\frac{m-1}{2}} \pi_x(a_{2i+1}|\sigma(a_{<2i+1}, \sigma_y))P(\pi_y)P(\sigma_y)}{\int \prod_{i=0}^{\frac{m-1}{2}} \pi'(a_{2i+1}|\sigma(a_{<2i+1}, \sigma'_y))P(\pi'_y)P(\sigma'_y)d\pi'_y d\sigma'_y}$$

where $\sigma(H, \sigma_0)$ is the state reached after starting from $\sigma_0$ and going through transitions in $H$ (where this isn't deterministic, it would have to be integrated over).

By separating over players, we can perform the marginalisation over states separately, rather than having to marginalise over all state pairs. Also if $\pi_x$ and $\pi_y$ have separate parameters then the max joint posterior is the max of the individual posteriors, so we can optimise separately.

However, better would be to make an empirical policy from B in the form of a decision tree with frequencies of behaviuor at each node. Choose a pair of start states and self-play. Repeat to generate a set of histories. Then use importance sampling to weight each member of the set against its probability in B, and calculate the sum of logs over all moves in the empirical policy.

A good social norm should make public each players internal state to the extent that it effects behaviour, which means that agents follow the empirical policy, and so the posterior should be independent of priors over state. So, we can assume uniform priors over state. Since we have no reason otherwise, we also assume uniform prior over policy space, so the posterior is proportional to the likelihood.

This has the advantage that agents can learn behaviour by observation, and we can study the effect of having different levels of innovation/copying in society. It has the disadvantage that agents can't learn to model the behaviour of deviant social ties (but could learn expected return without modelling of other through Q-learning from experience). It also has the disadvantage that deviants (i.e. agents that ) can't learn from other's behaviour or model others well [perhaps there is some truth there?].

Agents could learn about social ties over multiple encounters through simple Q-learning without modelling other at all, with the assumption of other only providing an initial state for new social-ties. Alternatively, we could maintain the assumption but do Bayesian updating on the distribution over state at the start of an encounter.

### 3.1.2 The conformist/deviant strategy

If we don't assume that others have the same policy as us, then there is a distribution over policy and state of other. After taking the expectation over policy and state of other, other's move is just an action distribution given encounter history. So we can directly model "empirical policy" in the form $P(a|H)$.

However, over multiple encounters with the same agent, we should learn the individual characteristics of the agent (which should influence our eagerness to enter into further encounters). The evidence is the public behaviour buffer and the set of encounters with a given agent.

Assume the population consists of a majority who adopt a single "social norm" policy while a minority of "deviants" choose a policy uniform randomly. If we posit a Gaussian (or exponential) prior on the proportion of deviants in society [or perhaps deviance is defined as knowingly gaining at the expense of other...but this seems to assume a more abstract norm - perhaps the fundamental norm?], then we assume the distribution of policies in the population is the MAP given the public buffer. The distribution over policies of a given acquaintance can then be updated using Bayes given the observations of his behaviour. This can then be integrated over the policy space to give a posterior action distribution given encounter history, and so an expected quality.

So, we're saying that, for each conversation type, the buffer was generated by sampling pairs of policies from a population distribution of the form

$$P_{\pi_n,\Delta}(\pi) = (1-\Delta)\delta_{\pi_n}(\pi) + \Delta P_u(\pi)$$

where $\pi_n$ is the social norm policy (encoding both sides of the dialogue), $\Delta$ is the fraction of deviants in the population and $P_u(\pi)$ is an uninformative prior from which we assume deviants are drawn. We require that the uninformative prior has the property that observing the behaviour of a sigle encounter with a policy drawn from the prior up to time t gives no information about the behaviour at time t+1, i.e. for all histories, H,

$$P_u(a_i|H) = \int \pi(a_i|H)P_u(\pi|H)\,d\pi = \int \pi(a_i|H)\frac{P(H|\pi)P_u(\pi)}{P(H)}\,d\pi = \frac{1}{|A_i|}$$

where $|A_i|$ is the number of possible actions after history $H$.

But

$$P_u(\pi|a_0...a_m) = \frac{P(H|\pi,\pi'')P_u(\pi)}{P(H)} = \frac{P(\pi)\prod_{i=0}^{\frac{m}{2}}\pi(a_{2i}|a_{<2i})}{\int P(\pi')\prod_{i=0}^{\frac{m}{2}}\pi'(a_{2i}|a_{<2i})d\pi'}$$

for the first mover. Similarly for second mover.

However, we don't know $\pi_n$ or $\Delta$, so we put an uninformative prior on $P(\pi_n)$ and, let's say, $P(\Delta) = (n+1)(n+2)\Delta(1-\Delta)^n$ for $0 \leq \Delta \leq 1$. So the posterior over $\pi_n, \Delta$ is

$$P(\pi_n, \Delta|B) = \frac{P(B|\pi_n,\Delta)P(\pi_n)P(\Delta)}{P(B)}$$

where the likelihood is given by

$$P(B|\pi_n,\Delta) = \prod_{H\in B}\int_{\pi_x}\int_{\pi_y} P_{\pi_n,\Delta}(\pi_x)P_{\pi_n,\Delta}(\pi_y)P(H|\pi_x,\pi_y)\,d\pi_x d\pi_y$$

and letting $H = a_0, ...a_n$

$$P(a_0, ...a_n|\pi_x, \pi_y) = \prod_{i=0}^{\frac{n}{2}} \pi_x(a_{2i}|a_{<2i}) \prod_{i=0}^{\frac{n-1}{2}} \pi_y(a_{2i+1}|a_{<2i+1})$$

So, we can separate players to give

$$P(B|P_{\pi_n,\Delta}) = \prod_{a_0,...a_n \in B} \int_{\pi_x} \prod_{i=0}^{\frac{n}{2}} P_{\pi_n,\Delta}(\pi_x)\pi_x(a_{2i}|a_{<2i})\, d\pi_x \int_{\pi_y} \prod_{i=0}^{\frac{n-1}{2}} P_{\pi_n,\Delta}(\pi_y)\pi_y(a_{2i+1}|a_{<2i+1})\, d\pi_y$$

but

$$\int_{\pi} P_{\pi_n,\Delta}(\pi) \prod_{i=0}^{\frac{n}{2}} \pi(a_{2i}|a_{<2i})\, d\pi = (1-\Delta) \prod_{i=0}^{\frac{n}{2}} \pi_n(a_{2i}|a_{<2i}) + \Delta \int P(\pi) \prod_{i=0}^{\frac{n}{2}} \pi(a_{2i}|a_{<2i}) d\pi$$

But

$$\int P(\pi) \prod_{i=0}^{\frac{n}{2}} \pi(a_{2i}|a_{<2i}) d\pi =$$

[need to be clearer about what we mean by uniform distribution over policies as uniformity is relative to the way we define the metric in the policy space. Even defining uniformity over a simplex requires a metric. However, a requirement of the uniform prior over a simplex is that it doesn't favour any outcome so $\int P(\pi)\pi(a_i)d\pi = 1/|A|$. We could extend this to the policy space by requiring a prior such that the history of moves gives us no information about the next move. So $\int P(\pi)\pi(a|H)d\pi = 1/|A|$ for all $H$. Many priors fulfill these requirements, but for our purposes it doesn't matter which we choose...]

but given the population, the probability of observing B is the integral over all pair assignments (for simplicity, we assume the population is large enough that we can sample with replacement) of the product of the probability of all encounters.

[Alternative: Mind reading. Assume that other has the same reward and state transition functions as self, that policy decisions only have access to current episode history, that policy maximises individual reward and priors over state at the start of encounter maximise probability of observed encounters in the buffer. After an encounter, we update the prior over start state using Bayes' rule.]

[Perhaps the whole point of roles is to allow agents to interact without having to "read minds". Once we've negotiated roles, I don't really care what your state of mind is, all I care is that you fulfil your role (or alternatively, knowing your state of mind wouldn't make me act any differently)...it simplifies theory of mind (this perhaps explains the dehumanising effect of roles)]

We can play out against an agent with posterior behaviour in order to update our own policies. [So, an abstract description of self-play would be to have a model of the world against which self can play. The model is an approxiating function that is updated with evidence from reality (and possibly from self-play). Self policy is updated by self-play with the modelled world (and possibly also directly from real world).]