The AI economist: Improving Equality and Productivity with AI-Driven Tax Policies

Stephan Zheng, Alexander Trott, Sunil Srinivasa, Nikhil Naik, Melvin Gruesbeck, David C. Parkes, and Richard Socher, 2020, mimeo.

Presenter: Yoji Tomita

RL-GT ゼミ June 17, 2021

Table of Contents

1. Introduction

・イントロダクション

2. Economic Simulations: Learning in Gather-and-Build Games

- Economic environment について.
- まずは税の無い設定 (”free-market”) で説明する.

2.1 Notation and Preliminaries

- Partial-observable multi-agent Markov Games(MGs): $(S, A, r, \mathscr{T}, \gamma, o, \mathscr{I})$
  - S : **状態空間** (state space)
  - A : **行動空間** (action space)
  - $r_{i,t}$: **報酬関数** (reward function)
  - $\mathscr{T}$: **遷移関数** (transition function) $s_{t+1} \sim \mathscr{T}(\cdot \mid s_t, \boldsymbol{a}_t)$
  - γ : **割引因子** (discount factor)
  - $o_{i,t}$: **観測** (observation)
  - time step $t = 0, 1, \ldots, H$.

- Agents' policy : $\pi_i(\cdot \mid o_{i,t}, h_{i,t}; \theta_i)$
  - $h_{i,t}$: hidden state（自分の私的情報と, 過去の history）
  - θ_i : policy の parameter
  - エージェント i は次の最大化問題を得く policy を求める:

$$\max_{\theta_i} \mathbb{E}_{a_i \sim \pi_i, \boldsymbol{a}_{-i} \sim \boldsymbol{\pi}_{-i}, s' \sim \mathscr{T}} \left[\sum_t \gamma^t r_{i,t} \right]. \tag{1}$$

- データ効率性のため, すべてのエージェントは training の間パラメータ θ を共有する.
- 彼らの行動 $\pi_i(a_i \mid o_i, h_i; \theta)$ は, agent-specific observations o_i と hidden-state h_i によって異なる.

Gather-and-Build game

- 2 次元の grid (25×25) からなる世界が舞台.
- エージェントはフィールドを歩き回り, 資源 (石と木) を集め, それらを 1 つずつ使って家を建て, また資源を coin を介してトレードする.
- 資源は空タイルに確率的に産み出される.
- エージェントは家を建てると coin が得られるが, 得られる coin は agent の skill ごとに異なる.

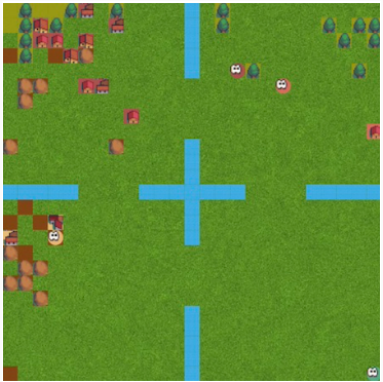
Labor and Skill.

- Agent の action(moving, gathering, trading, building) にはそれぞれ labor cost が設定されている.
- 各 time に agent がどれか 1 つ行動をとると, その labor cost がかかる.

- building skill (1 以上 3 以下) が各 agent に設定されていて, 家を建てると agent は $10\times$ skill 分の coin を得る.
- collection skill (1 以上 2 以下) もあり, 資源を拾うとこの skill 分の資源を得る (skill 1.2 の場合, 確定で 1 つ資源を得て, さらに確率 0.2 でもう 1 つ資源を得る)

Environment Scenario.

- field は水により 4 つの区域に別れている（水部分は通れない）
- 資源は空間的に集まって発生する.
- 4 agents
- buildng skills は 1.13, 1.33, 1.65, 22.2（Pareto 分布 w/ exponent $a = 4$, scale $m = 1$ の quartiles を元に設定）
- 1 episode は $H = 1000$ time steps からなる.



- Agent の utility function:

$$u_i(x_{i,t}, l_{i,t}) = \text{crra}\left(x^c_{i,t}\right) - l_{i,t}, \quad \text{where } \text{crra}(z) = \frac{z^{1-\eta} - 1}{1-\eta}, \ \eta > 0. \tag{2}$$

  - $x_{i,t} = (x^w_{i,t}, x^s_{i,t}, x^c_{i,t})$: i の保有する木・石・コイン.
  - $l_{i,t}$: 蓄積労働量.
  - η: エージェントの utility function の non-linearity をコントロールするパラメータ.
- Rational economic agent は以下の最大化を行う.

$$\forall i : \max_{\pi_i} \mathbb{E}_{a_i \sim \pi_i,\ \boldsymbol{a}_{-i} \sim \boldsymbol{\pi}_{-i}, s' \sim \mathscr{T}} \left[u_i(x_{i,0}, l_{i,0}) + \sum_{t=1}^{H} \gamma^t \underbrace{\left(u_i(x_{i,t}, l_{i,t}) - u_i(x_{i,t-1}, l_{i,t-1})\right)}_{=r_{i,t}} \right]. \tag{3}$$

Deep RL agents

- deep neural network を用いる agent policy を modelling する:

$$a_{i,t} \sim \pi(o_{i,t}^{\text{world}}, o_{i,t}^{\text{agent}}, o_{i,t}^{\text{market}}, o_{i,t}^{\text{tax}}, h_{i,t-1}; \theta)$$

  - $o_{i,t}^{\text{world}}$: 近くの状況に関する観測.
  - $o_{i,t}^{\text{agent}}$: public な agent の状況 (資源・コイン保有) と, private agent states(skill 値と labor performed)
  - $o_{i,t}^{\text{market}}$: transfer market の状況（bid, ask offer）
  - $o_{i,t}^{\text{tax}}$: tax rates