ABM Macro Lab: Agent-based Modelling Tools

Session 02

Gabriel Petrini

July, 2025

Outline

Technical review

Theoretical model presentation

Model implementation

Model initialization

Simulations

Technical review

Simulation pipeline

As a recap, whenever running a model on LSD, we need to:

1. Design a model (on paper);
2. Write the code implementing the equation;
3. Define the model structure and initialization;
4. Run the simulation;
5. Analyze the results.

Where are we and where are we going?
In the previous session, we implemented two different small models. Now, we will focus on a economic ABM.

Theoretical model presentation

Sequence of events

At time $t = 0$, there are NF identical incumbent firms with equal $a_{i,t=0}$ and $s_{i,0}$.
At each time step $t = 1, 2, \ldots, T$:

1. Firms learn (updates (a_i))
2. Firms computes market share (updates s_i)
3. Market concentration index (HHI) is computed
4. Firms exits the market if $s_i < s_{min}$
5. Firms enter to keep NF firms in the market
6. Incumbents variables $(a_j,\ s_j)$ are set
7. Market shares are re-scaled proportionally to ensure $\sum s_i = 1$

Model logical order (later)
The exit-entry process is time-coordinated during each simulated time step by the technical variable `exit_entry`: `HHI` > `exit_decision` > `s` rescaling

Equations

Idiosyncratic learning process:	$a_{i,t} = a_{i,t-1} \cdot (1 + \eta \cdot \theta_{i,t})$
Learning shocks	$\theta_{i,t} \sim Beta(\beta_1, \beta_2)$
Market selection	$s_{i,t} = s_{i,t-1} \cdot \left(1 + A \cdot \frac{a_{i,t} - \bar{a}_t}{\bar{a}_t}\right)$
Average productivity	$\bar{a}_t = \sum_{i=1}^{NF} s_{i,t-1} \cdot a_{i,t}$
Exit condition	$s_{i,t} < s_{min}$
Entrant productivity	$a_{j,t} = \bar{a}_t \cdot (1 + \eta \cdot \theta_{i,t})$
Entrant market-share	$s_{j,t} = 1/NF$
Market concentration index	$HHI_t = \sum_{i=1}^{NF} (s_i)^2$
Market-share adjustment	$s_i \mapsto s_i \cdot \frac{1}{\sum_{i=1}^{NF} s_i} \Rightarrow \sum_{i=1}^{NF} s_i = 1$
Fixed number of firms	$\#\{1, \ldots, n\} = NF$

Parameters and initial values

Table 1: Baseline parameters

	Desc	Value
η	Innovation opportunity support	0.3
β_1, β_2	beta distribution parameters	1.0; 5.0
A	replicator dynamics intensity	1
s_{min}	minimum market share to not exit	0.01
NF	number of firms	10
a_i	Initial Firm-level productivity	1.0
s_i	Initial Firm market-share	$1/NF$

Model structure and data organization

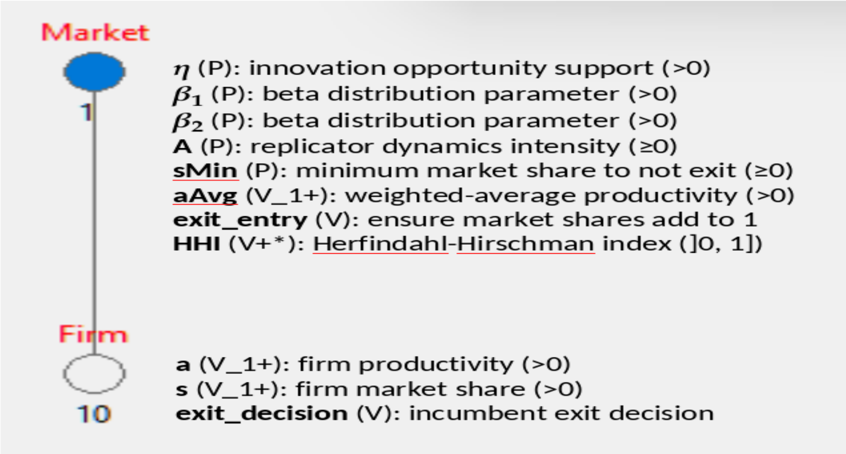


Figure 1: Structure of industry model

Model implementation

Firm-level productivity (a_i)

$$a_{i,t} = a_{i,t-1} \cdot (1 + \eta \cdot \theta_{i,t}) \tag{1}$$

```
EQUATION("a")
// Firm knowledge/productivity
v[0] = CURRENT;
v[1] = V("eta"); v[2] = V("beta1"); v[3] = V("beta2");
v[4] = beta(v[2], v[3]);
v[5] = v[0] * (1 + v[1] * v[4]);
RESULT(v[5])
```

One-liner

```
EQUATION("a")
RESULT(VL("a", 1) * (1 + V("eta") * (beta(V("beta1"), V("beta2")))))
```

Market-share (s_i)

$$s_{i,t} = s_{i,t-1} \cdot \left(1 + A \cdot \frac{a_{i,t} - \bar{a}_t}{\bar{a}_t}\right) \tag{2}$$

```
EQUATION("s")
// Firm size/market share
v[0] = CURRENT;
v[1] = V("A"); v[2] = V("a"); v[3] = V("aAvg");
v[4] = (v[2] - v[3])/v[3];
v[5] = v[0] * (1 + v[1] * v[4]);
RESULT(v[5])
```

One-liner

```
EQUATION("s") // Firm size/market share
RESULT(VL("s", 1) * (1 + V("A") * ((V("a") - V("aAvg"))/V("aAvg"))))
```

Exit condition

```
EQUATION("exit_decision")

v[0] = V("s"); v[1] = V("sMin");
// update entrant firm productivity and market share
if (v[0] < v[1]) {
  v[2] = V("eta"); v[3] = V("beta1"); v[4] = V("beta2");
  v[5] = beta(v[3], v[4]);
  v[6] = V("aAvg");
  v[7] = v[6] * (1 + v[2] * v[5]);
  WRITE( "a", v[7] );
  WRITE( "s", 1 / COUNT( "Firm" ) );
}
RESULT(0)
```

Market-level Productivity (Weighted) Average ($\bar{a}_t$)

$$\bar{a}_t = \sum_{i=1}^{NF} s_{i,t-1} \cdot a_{i,t} \tag{3}$$

```
EQUATION( "aAvg" )
// Mean knowledge/productivity
v[0] = 0;          // accumulator
CYCLE(cur, "Firm") {
  v[1] = VLS( cur, "s", 1 );
  v[2] = VS( cur, "a" );
  v[3] = v[1] * v[2];
  v[0] += v[3] ;
}
RESULT( v[0] )
```

Entry-Exit condition

```
EQUATION( "exit_entry" )
// Trigger market-wise exit-entry dynamics and re-scale shares
V( "HHI" ); // first, compute HH index before exits

// second, ensure firms have decided on exit
CYCLE(cur, "Firm") {VS( cur, "exit_decision" );}

v[0] = 1 / SUM( "s" ); // factor to scale back to sum = 1
CYCLE(cur, "Firm") { // third, rescale market shares after exits
  v[1] = VS( cur, "s" );
  v[2] = v[0] * v[1];
  WRITES( cur, "s", v[2]);
}
RESULT( SUM("s") )
```

Herfindahl-Hirschman concentration index (*HHI*)

$$HHI_t = \sum_{i=1}^{NF} (s_i)^2 \tag{4}$$

```
EQUATION( "HHI" )
// Herfindahl-Hirschman concentration index
v[0] = WHTAVE( "s", "s" );
RESULT( v[0] )
```

Note on WHTAVE(LS)

`WHTAVE` (weighed average, not used here in the strict sense) computes the sum of $s \times s$ over every firm

Model initialization

Initialization I

Firm-level initialization can be set for every NF objects on the LSD browser. The same applies for every other initial condition.

Whenever changing the number of firms NF, it is important to also a compatible initial market-share $1/NF$. This procedure should be done for every firm (instance) of the model.

In the following slide (optional), we will see an alternative way in which we control some few parameters and create $NF - 1$ copies of a example object.

```
ADDNOBJ_EX("TypeOfAgent", number, *pointer);
```

Semi-automated initialization and sensitivity analysis
By doing this, we automate the model initialization to test different model configurations (next lecture)

Initialization II

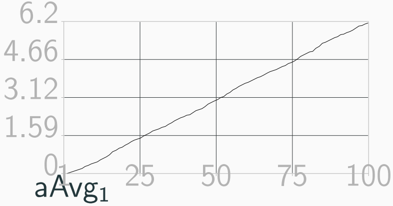
```
EQUATION( "init" )
PARAMETER;                    // turn into parameter (run once)
v[0] = V("A0"); v[1] = V("Nfirm");
v[2] = 1 / v[1]; // Fair share
CYCLE(cur, "Market"){
    cur1 = SEARCHS(cur, "Firm" );
// Overwrites the lag 1 of "a" to v[0] at time 1
    WRITELLS(cur1, "a", v[0], 1, 1);
    WRITELLS(cur1, "s", v[2], 1, 1);
// Adds N - 1 copies of cur1 agent located under cur
    ADDNOBJ_EXS(cur, "Firm", v[1] - 1, cur1);
}
RESULT( 1 )
```

Simulations

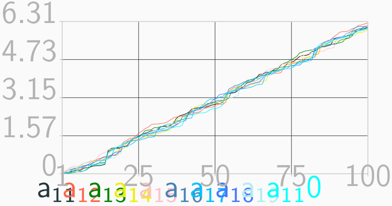
Analyzing the model results

- After a successful simulation run, in Browser click on menu Data > Analysis of Results
- To analyze the *saved* data time series, select the desired variables from the Series available list to include them in the Series selected list
- Click on Plot button to show the selected variable(s) time series plot
- Click on Statistics button to show the selected time series descriptive statistics in LSD Log
- Plots and analysis data can be *saved* pressing the Save Plot and Save Data buttons

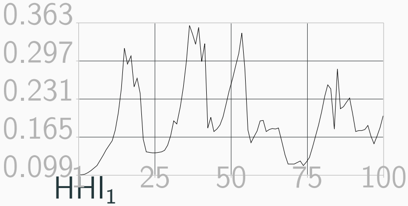
Exploring the single run results



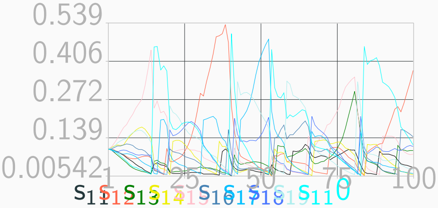
(a) Average log-productivity (a)



(b) Firm log-productivity (a)



(c) Herfindahl-Hirschman Index (HHI)



(d) Firm market share (s)