The path to chaotic equilibrium – A simulation of expectation inference using mixture model

Ming Cheung, Fok (Hubert)

Introduction

The study of Business cycle has been one of the key research areas in macroeconomic, models had been developed primarily to determine the path to general equilibrium. Yet, little effort had been made to examine the cause of deviating from the steady state. In view of the above, this project applies the model developed by Greenwald and Stiglitz (1993) to explore the relationship of fluctuation in asset value and aggregate pricing uncertainty, their model implies that uncertainty in the distribution of aggregate price level may lead to cyclical economic growth. Combining Veldkamp (2017) with Greenwald and Grossman model, this project has two objectives. First, show that suppliers with different information sets may engender the process of capital accumulation; second, provide evidence to the claim that agents possessing incomplete information is key to understand volatility in general price. This document will be arranged as follow, section (1) introduces the model; section (2) explains the algorithm for simulation and summarize all findings, then section (3) concludes.

The model

Greenwald and Stiglitz (1993) examine the spillover effect of uncertainty in aggregate price level on the aggregate equity accumulation. Compared to previous models, their model amplifies the effect of uncertainty on firm decision by introducing financial leverage and the cost of bankruptcy. Following illustrates the setup of the model.

The analysis was simplified where labor is the only production factor and the market can absorb all produced outputs. Hence, the production function can be written as a function of labor demand function, $L_{i,t} = \phi(q_{i,t})$, and we assume that $L_{i,t} = 0.4 * q_{i,t}$ with a constant wage($\overline{w}$). Each suppliers is endowed with an initial capital[1], $A_{i,t=0}$, all initial capital will be invested into production and all subsequent residual income will be kept as equity for future reinvestment. As profit will be realized at the end of each period, suppliers may need to borrow additional funding to pay the cost of production (wage, w_t), $B_{i,t} = P_t\overline{w}L_{i,t} - A_{i,t}$, and the borrowed capital will be repaid in next period with interest, $(1 + R_t)$. In view of the above, the ***realized*** profit of a supplier in period t depends on (i) the cost of labor, (ii) the cost of

[1] This is merely financial capital which is different from the tangible capital in conventional production function, like the Cobb-Douglas function.

borrowing $(1+R_{t-1})B_{i,t-1}$, and (iii) capital accumulated up till time $t-1$, which can be written as the following equation.

$$\Pi_{i,t} = P_t q_{i,t-1} - (1+R_{t-1})(P_{t-1}\overline{w}L_{i,t-1} - A_{i,t-1})$$
$$A_{i,t} = A_{i,t-1} + \Pi_{i,t}$$

Because of the time gap between production and profit realization, the objective of a supplier is optimizing their expected profit by forecasting future price as well as the cost and likelihood of bankruptcy. According to Markov property, their expectation of future price is bounded by their information set(I_t), which is their knowledge (or memory) about the history of aggregate price. Hence, the discrepancy between aggregate price and individual price can be represented as a ratio,($\tilde{u}_{i,t}$), where

$$\tilde{u}_{i,t} = \frac{P_{i,t}}{P_t}, and$$
$$E_t[P_{i,t+1}|I_t] = E_t[P_{t+1}|I_t]E_t[\tilde{u}_{i,t+1}],\ \ E_t[\tilde{u}_{i,t+1}] = 1$$

. Since supplier is insolvent when he/she cannot repay his/her financial obligation, which can be modified as the following inequality.

$$(1+R_{t-1})B_{i,t-1} > P_{i,t}q_{i,t-1}$$
$$(1+R_{t-1})(P_{t-1}\overline{w}L_{i,t-1} - A_{i,t-1}) > P_t\tilde{u}_{i,t}\, q_{i,t-1}$$
$$\bar{u}_{i,t} \equiv (1+R_{t-1})\frac{P_{t-1}}{P_t}\frac{(\overline{w}L_{i,t-1} - A_{i,t-1})}{q_{i,t-1}} > \tilde{u}_{i,t}$$

Therefore, the likelihood of bankruptcy depends on the likelihood of mispricing. Assuming the distribution of $\bar{\tilde{u}}_{i,t}$ is $F(.)$[2] and density $f(.)$, such that the likelihood of bankruptcy can be written as $F(\tilde{u}_{i,t}) = \int_0^{\bar{u}_{i,t}} \tilde{u}_{i,t}\, f(\tilde{u}_{i,t})\, d\tilde{u}_{i,t}$.

In the original setup, the information set, i.e. the distribution of the historical data, was assumed to uniform among suppliers. Yet, the belief-driven model developed by Veldkamp (2017) shows that the specification on information set has deterministic effect on the path to steady state. Taking one step further, this project argues that the economy will deviate from steady state when the information set of each agent is significantly deviate from each other. Following Veldkamp model, we employ the kernel density function to construct the probability distribution for each agent, so that the probability density function of price is defined as

$$f(P_t) = \frac{1}{n_t\kappa}\sum_{s=0}^{n_t-1}\Omega\left(\frac{P_t - P_{t-s}}{\kappa}\right)$$

[2] The author did not specify the form of the distribution in the paper, but this will be specified in next section.

where κ denotes the width of the kernel; Ω denotes the standard normal density function and $F(P_t) = \int_0^\infty p_t f(p_t) dp_t$. As a conventional practice, the optimal bandwidth will be determined according to the Silverman rule, which is:

$$\kappa = \left(\frac{4\sigma_p^5}{3n}\right)^{\frac{1}{5}}$$

Apart from the risk of mispricing and bankruptcy, the output decision is also hindered by the cost of bankruptcy. To avoid excessive production, the original model suggests that the cost of bankruptcy should be positively correlated with the quantity of production. The marginal bankruptcy cost function used in the simulation is defined as:

$$c(q_{i,t}) = 0.1\log(q_{i,t}) - 0.1^2\log(q_{i,t}^2) - 0.1^2\log(q_{i,t}^3) + 0.1^{(3)\log}(q_{i,t}^4), c_{i,t} \in [0,1]$$

and the total cost of bankruptcy $\mathbb{C} = c(q_{i,t})q_{i,t}$. Furthermore, the original model assumes a risk-neutral nominal interest rate. However, the interest rate function used in the simulation is adjusted according to (i) the likelihood of mispricing and (ii) the liability-to-capital ration ($\gamma_{i,t} = (P_t\overline{w}L_{i,t} - A_{i,t})/A_{i,t}$), where

$$R_{i_t}(F(\tilde{u}_{i,t}), \gamma_{i,t}) = r_t + 0.1 * F(\tilde{u}_{i,t}) + 0.1\gamma_{i,t}$$

Given the above specification, the objective of a supplier is maximizing the **expected nominal** profit by optimizing the output quantity according to their 'beliefs' about the probability distribution of aggregate price. The objective function is written as:

$$q_{i,t}^* = \max_{q_{i,t}} E[P_{i,t+1}\, q_{i,t} + (1 + R_t)(P_t\overline{w}L_{i,t} - A_{i,t})] - c_{i,t}\, F(\overline{u}_{i,t+1})$$

or, in continuous form,

$$q_{i,t}^* = \max_{q_{i,t}} \int_0^\infty \boldsymbol{f_i}(P_{t+1})P_{t+1}q_{i,t} + (1 + R_t)(P_t\overline{w}L_{i,t} - A_{i,t})\, dP_{t+1}$$

$$- c_{i,t} \int_0^{\overline{u}_{i,t+1}} \overline{u}_{i,t+1}\, f(\overline{u}_{i,t+1}) d\overline{u}_{i,t+1}$$

At the end of each period, the aggregate price will be jointly determined by all supplier base on their expected price and production quantity, such that

$$P_{t+1} = \prod_{i=1}^{n_s} p_{i,t+1} \frac{q_{i,t+1}^*}{\sum_{i=1}^{I} q_{i,t+1}^*}$$

. And the realized profit of each supplier is:

$$\Pi_{i,t} = \frac{1}{P_{t+1}} E[P_{i,t+1}\, q_{i,t} + (1 + R_t)(P_t\overline{w}L_{i,t} - A_{i,t})]$$

, and the cycle of the economy is governed by the (aggregate) capital motion formula

$$A_{t+1} = \sum_{i=1}^{n_s} A_{i,t} + \Pi_{i,t+1}$$

The program

The flow of simulation

Table (1) depicts all the functions and figure (1) shows the flow of the simulation program. The core function is the *initiation.m*, this function initiates the simulation by specifying the following parameters: the initial capital (a_0), wage value (w_0), the number of players in the game (n_s), and the duration of the simulation (t_{end}), the history of the economy (T), the theme of the game ($sim\_opt.theme$)[3], the motion of interest rate ($sim\_opt.int$), and the memory capacity of the player ($sim\_opt.mem$).

Name	Function
Initiation.m	This is the core function of the simulation program. Users can customize their own simulation by defining the above parameters of the game.
Simulation.m	Initiate the simulation according to the specified parameters
Player_opt .m	This function calculates the optimal production quantity and price according to the type of player and their information set.
Prod_funct.m	This calculates the expected profit according to the given probability distribution, expected price, an array of randomly generated production quantity
profit_max.m	Given the hypothetical profit generated by Prod_funct.m, this function applies the non-linear optimization method (Iscurvefit) to explore the optimal output quantity.
b_m.m	This function will be applied to generate series of historical price and interest rate.
price_update.m	Update the aggregate price by weighting the output level.
kdf.m	Generate the probability distribution
Minxk.m	Find the minimum k members of an array.

[3] Details of this parameter will be further discussed in next subsection.

Entropy.m **(Optional)**	Measuring the entropy of a dataset and the corresponding probability density value.

Table. 1 – Program list

Next, the program executes the function *b_m.m* to generate series (default length: 10000[4]) of hypothetical price data and interest rate data according to the Brownian motion formula:

$$\frac{dP}{P} = \mu dt + \sigma \epsilon_t \sqrt{dt}, \qquad \epsilon \in [0,1]$$

$$P_t = P_0 e^{\mu dt + \sigma \epsilon_t \sqrt{dt}}$$

, the result will be stored into local directory and used for future simulation. Since the theme of the game ($sim\_opt.theme$) determines the type of players involved in the game, the process of using these data to optimize the production decision varies among players.

[4] In the submission folder, the file ‘ BMPrice_long.xlsx’ contains 1000,000 data points. With longer history, the effect of inferencing point on price formation becomes significant.

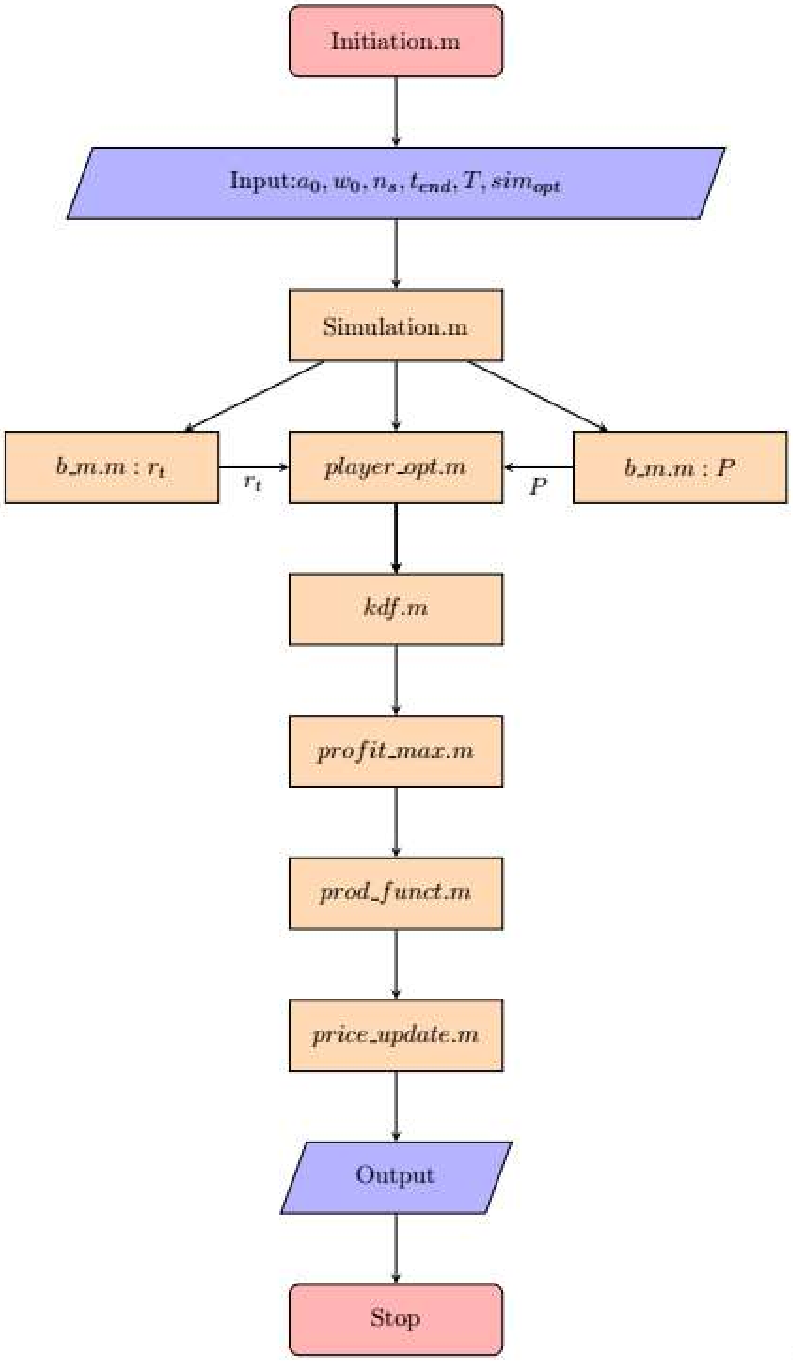

Fig. 1 Flow chart of the simulation

Before introducing further details of the simulation, following explains the algorithm of optimization, Using the historical price data, the function *player_opt.m* determines the information set according to the type of player. Because the player type defines the memory capacity and data preference, the function generates a customized array storing all 'observed' data. The function *kdf.m* then constructs the probability distribution of the data set and return series of price data point and the corresponding probability density value ($f(p)$).

The array containing price data and probability density value will then pass to the optimization function *profit_max.m*. The optimization begins with generating an array with randomly defined production quantity, and all the data will be placed into *prod_funct.m* to generate expected profit. During the process of calculation, the function expands the price and quantity array into an m-by-m matrix, e.g. $\mathbb{p}_{m,m}$and $\mathbb{Q}_{m,m}$, and compute the likelihood of mispricing ($F_{m,1}$) by calculate the cumulative sum of the probability density value$f_{m,1}$. So the expected profit function in matrix form is written as:

$$\mathbb{E}[\Pi] = f_{m,1} \cdot [\mathbb{p}_{m,m} \cdot \mathbb{Q}_{m,m} - (1 + \mathbb{r}_{m,1}) \cdot (P_t w_t L_t(\mathbb{Q}_{m,m}) - A_t)] - c(\mathbb{Q}_{m,m}) \cdot F_{m,1}$$

, the result should be an m-by-m matrix. Next, the algorithm applies a MatLab built-in optimization function – Isqcurvefit to explore optimal output quantity, the objective of the optimization function is defined as:

$$\min_q \sum (\Pi(\mathbb{p}_{m,m}, \mathbb{Q}_{m,m}, f_{m,1}) - E[\Pi])^2$$

. After that, the function *player_opt.m* returns the optimal production (q^*), the expected price, $E[P] = f_{1,m} * P_{m,1}$, and the expected profit.

When the optimization of all players had completed, the function *price_update.m* will update the aggregate price according to the overall production quantity and each individual price.

The setup of the game

Types of player

There are 5 types of players participating the game. Type (1) players act according to Veldkamp suggestion who can recall the entire history to estimate the likelihood function. Type (2) players have limited memory whose maximum memory is limited to 10% of the entire history. Type (3) and type(4) are two extreme players who will recall the higher and lowest value of the history. And type (5) players randomly sample data from the history to construct the kernel.

Environment

There are 3 variables defining the environment of the game. First, the theme of the game ($sim\_opt\,.\,theme$) ranges from: (i) 'All' which involves all types of players, (ii) 'Random' which randomly determines the distribution of the type of players, (iii) 'Econometrician' which includes only the type 1 players, (iv) '3 players' which excludes type (3) and (4) players, and (v) 'Chaos' which not only randomly initiate the distribution of player types but also randomly switch the type during simulation.

Second, the motion of interest rate $(sim\_opt\ .int)$ can be chose as (i) 'Constant' or (ii) 'Stochastic' (Stoch). Finally, the memory capacity $(sim_{opt}.mem)$ can be defined as (i) 'Random' which randomly determine the size of inferred data, or (ii) 'Max' where players have the full memory of the entire history except player (2).

Analysis

Although the program provides a large degree of freedom to modify the simulation parameters, one should pay careful attention to the 'shape' of the profit function (profit_max.m). In order to find the optimal output quantity, the profit function should be a concave function of output quantity, which means the constraints on cost of bankruptcy and default related interest rate should be non-concave.

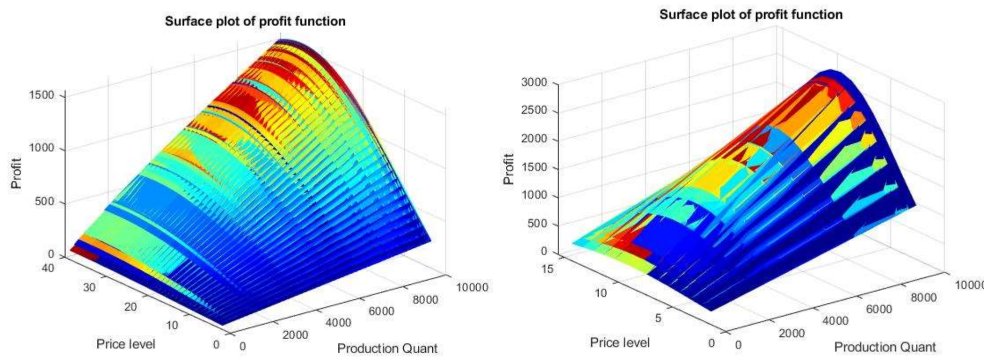

Fig. 2 Two examples of concave profit function

The simulation results are summarized according to the main objectives of the project. First, do player types matter to economic growth? Yes, the economy with only type 1,2 and 5 accumulates far more capital than the game with all type of players and Chaos game (Fig. (2))

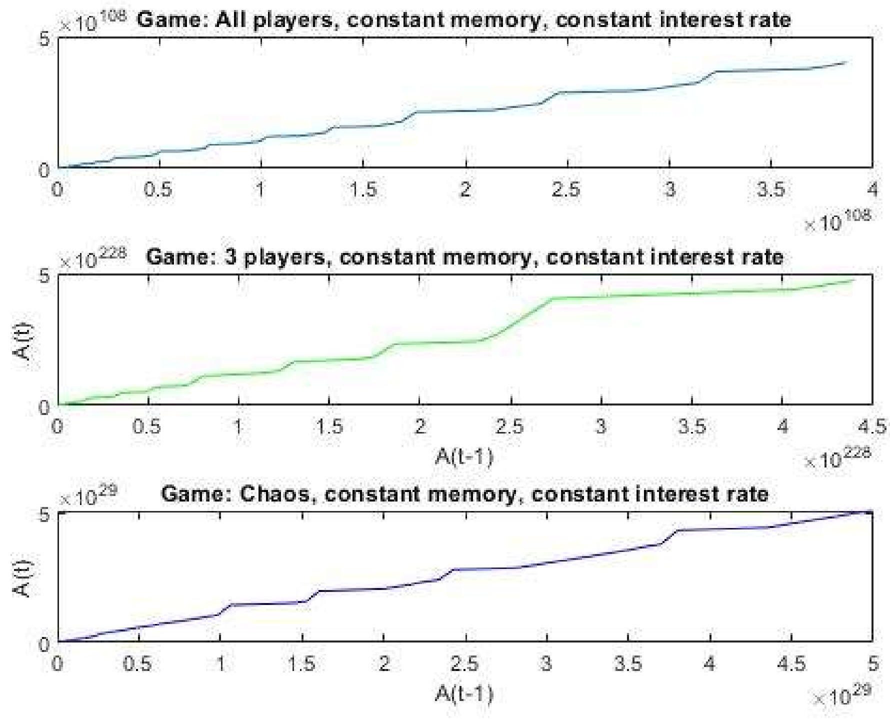

Fig. 3 Capital motion in 3 different games.

Second, do memory capacity plays an important role? Memory plays an important role in determining the volatility of price level. Given similar theme and interest rate motion, games with limited memory capacity generates higher price volatility. (Fig. (3).)

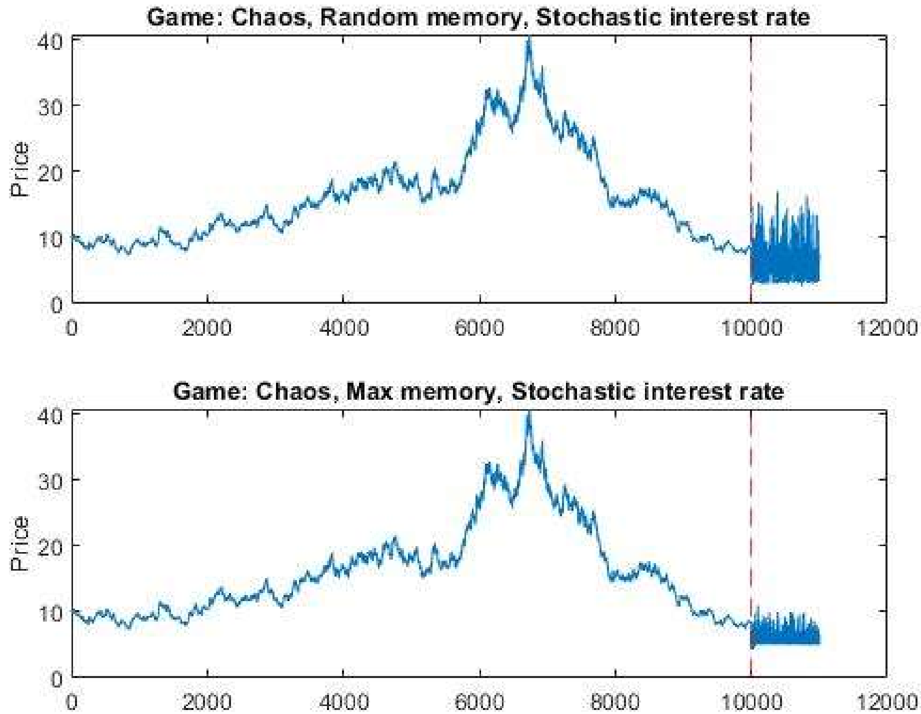

Fig.4 Price trend

This simulation shows that adding uninformed players into the game would increase the noisiness of the data. By inferring from these data would generate ambiguous signals which might hinder potential economic growth and lead to economic crisis.

Conclusion and improvement

Undoubtedly, the simulation merely indicate that imperfection information may lead to suboptimal pricing decision which are not unexpected. But the result points out an important question: what is an optimal information which can lead to consensus. The simulation fails to replicate the price trend history despite allowing homogenous (type 1) players with maximum memory, it means that information other than price history could be a potential indicator of the level change in price.

It is also worthwhile to mention the model oversimplifies the reality. First, the model does not consider the impact of demand side, for example the labor and consumption market. But one should be aware that adding extra complexity into this highly non-linear model may result in the impossibility in finding optimal solution. Second, one can refine this simulation by adding the condition that players would convert to the same type during the simulation. By doing that, it provides the possibility of converging to steady state.

Reference

Greenwald, B. C., & Stiglitz, J. E. (1993). Financial market imperfections and business cycles. *The Quarterly Journal of Economics*, *108*(1), 77-114.

Kozlowski, J., Veldkamp, L., & Venkateswaran, V. (2017). The tail that wags the economy: Belief-driven business cycles and persistent stagnation. *NYU Law and Economics Research Paper*, (15-25).

Appendix (I) Common errors during simulation

Following provides a list of commonly seen errors and corresponding solution to it.

Errors	Cause	Solution
1. Cannot find Z value for surface plot	The program automatically determines a long bandwidth while given a short dataset, then the result would be a scalar rather than array.	Re-run the program
2. Fail to find solution for IsCurvefit	User impose restrictive punishment on the cost of bankruptcy and the production function, then it is possible to have no optimal solution	Re-run the program
3. Cannot find intiation.m	Student number has been added in front of the function name	Erase the student number for each function.