



**Gustavo Miguel
Bastos Noronha**

Título provisório



**Gustavo Miguel
Bastos Noronha**

Título provisório

Relatório de Projeto do Mestrado Integrado em Engenharia Física, realizado sob a orientação científica dos Doutores Nome completoe Nome completo, Professores Auxiliares do Departamento de Física da Universidade de Aveiro

o júri / the jury

presidente / president

Nome

Professor XXXX da Universidade da Aveiro

Arguente

Nome do Arguente

Professor XXXX da Universidade de Aveiro

Orientador

Nome completo

Professor XXXX da Universidade da Aveiro

Co-orientador

Nome completo

Professor XXXX da Universidade da Aveiro

**agradecimientos /
acknowledgements**

Obrigado

Thank you

Resumo

Bem resumido

Abstract

Asbstract

Contents

Contents	i
1 Introduction	1
1.1 One	1
1.2 Two	1
1.2.1 One	1
2 Torn to pieces. Glued together	2
2.1 Initialization	4
2.2 Cycle	5
2.3 Government	13
3 Results	14
Bibliography	16

Chapter 1

Introduction

1.1 One

1.2 Two

1.2.1 One

Chapter 2

Torn to pieces. Glued together

This work will be using terms referring to the analysed system, which will be defined next. Sets are defined by Uppercase (A), elements by lowercase (a) and operators by (\hat{A}).

Goods are used and created by Agents to perform trades.

Shares (S) are owned exclusively by Persons and each refer to a singular Factory. The sum of all shares from a single Factory must be 1

Stock ($\hat{\lambda}, \hat{\nu}$) is owned exclusively by Factories, and are of essential or luxury type. essential Stock (λ) has a maximum consumption per Person per cycle of 1, while luxury (ν) doesn't. $\hat{\mu}$ is used to denote both types

Capital (\hat{c}) is the currency, tradable with Goods. The total capital in the system is constant and defined by c_{tot}

Agents are objects capable of holding Goods.

Persons (P) is a set of person objects that can consume stock, hold shares and work on factories.

Factories (F) will hold Stock and, if it has workers ($|W(f)| > 0$), it will produce stock.

If a Factory f has leftover capital in a cycle, this capital is distributed over all Persons who hold Shares referring it.

Markets are capable of trading Goods

GoodsMarket (\hat{G}) handles Stock consumption with Capital.

WorkersMarket (\hat{W}) handles worker assignment to Factories.

SharesMarket (\hat{S}) handles Capital trade with Shares, and is divided in

Primary (\hat{S}_1) where Factories trade with Persons

Secondary (\hat{S}_2) where Persons trade with Persons

Initialization Factories and Persons are created and Shares are distributed in the desired manner, these different initial distribution methods are

Egalitarianism every Person holds the same amount of value in Shares

Ownership every factory has a single share holder

Monopoly a single share holder holds all shares

Bourgeoisie shares are distributed over a percentage of People

Their consequences will be analysed further

Cycle Interactions between all these elements occur during cycles in this order:

Production (\hat{P}) Factories utilize workers to create stock.

GoodsMarket(\hat{G})

Funding (\hat{F}) Factories decide Stock production for next cycle and, if needed, create Shares

SharesMarket(\hat{S})

WorkersMarket(\hat{W})

Operators such as \hat{c} and $\hat{\mu}$ are applied to single objects, and behave as expected, with capital and stock as their eigenvalues and p, f as their **eigenobjects**, respectively. For example, the capital of a person $p \in P$ at time step t is given by $\hat{c}_t p = c_p p = c_t(p) p$. Also, an operator $\hat{A}(a, b)$ signifies that this operator acts on objects a and b , acting as an identity operator $\mathbb{1}$ for all other objects.

Shares refer to both a person and a factory, so they are treated as a unique structure: $S(f, p) \in R^+$, $p \in P$, $f \in F$ refers to the share of person p in the factory f . Using $S(p)$ refers to a set of all non-zero shares held by person p , and $S(f)$ refers to a set of all non-zero shares held by person p :

$$S(p) = \{f \in F \mid S(f, p) \neq 0\}$$

$$S(f) = \{p \in P \mid S(f, p) \neq 0\}$$

This being said, all of the above will now be unfolded in a more detailed manner.

2.1 Initialization

The system is initialized given the following parameters: $(|P|, |F|, c_{max}, c_{min})$ and the initial condition. This initial condition contains information about how to distribute the available shares in the soon-to-be created factories.

The default values used are $(|P| = 100, |F| = 20, c_{max} = 1000, c_{min} = 10)$, and the default initial condition is Egalitarianism.

The first step is to create P . $p \in P$ is an object containing name, Capital, employer, luxury and essential satisfaction and share catalog. During Initialization, a name is picked for every person and capital is assigned with a value between c_{max} and c_{min} . For all Persons, employer is set to None, luxury and essential satisfaction is set to 0 and share catalog is empty.

Next, F is created, $f \in F$ are defined by an ID, Capital, workers, share holders, product is essential (True or False) and stock.

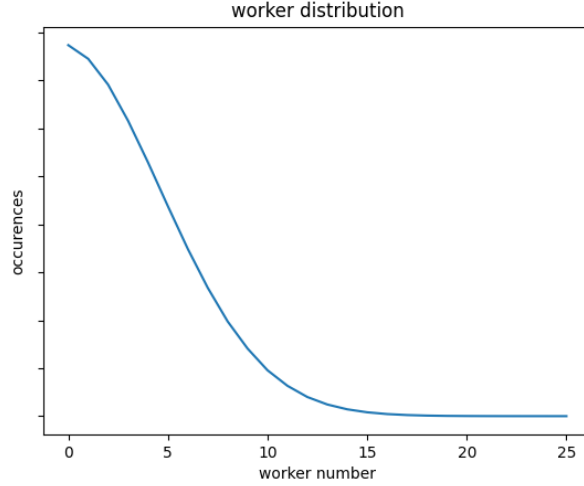


Figure 2.1: Gaussian distribution

Every factory f is assigned an ID and capital in the same way as a person. The number of workers ($|W|$) is generated by sampling a normal (Gaussian) distribution centered on one, just as depicted in figure 2.1, with a maximum of $\frac{|P|}{2}$ and that number of unemployed Persons are picked at random, and assigned as workers. This sampling operation are translated using the sample function.

$$|W(f)| = \text{sample}(\mathbb{Z}^+, \frac{|P|}{2})$$

$$W(F) = \left\{ W(f) = \bigcup_{i=1}^{|W(f)|} \{p_i \mid p_i \in_R P \cap \overline{W(F)}\} \mid f \in F \right\}$$

Where \in_R picks a single random object from a set.

The number of share holders are picked in a similar way, without the need of filtering unemployed people, and the value of each person's stock is picked by generating a uniform random number list and normalizing it.

$$\begin{aligned}
|S(f)| &= \text{sample}(\mathbb{N}, \frac{|P|}{2}) \\
S(F) &= \left\{ S(f) = \bigcup_{i=1}^{|S(f)|} \{p_i \mid p_i \in_R P \cap \overline{S(f)}\} \mid f \in F \right\} \\
S(f, p) \neq 0 &\Leftrightarrow p \in \bigcup_{i=1}^{|S(f)|} \{p_i \mid p_i \in_R P \cap \overline{S(f)}\} \wedge f \in F \\
1 &= \sum_i^{|S(f)|} S(f, p_i)
\end{aligned}$$

The first Factory's product is always essential, the second is always luxury, from there on, it's randomly picked 50% chance each. Defining F_λ and F_ν as all factories that produce essential and luxury, respectively.

$$F = \{f \mid f \in F, f_1 \in F_\lambda \wedge f_2 \in F_\nu\}$$

2.2 Cycle

This is the portion that is repeated to advance the simulation, represented by the time step operator $\hat{T}|\psi_t\rangle = |\psi_{t+1}\rangle$. This operator is simply the combination of the other cycle operators: $\hat{T} = \hat{W} \hat{S} \hat{F} \hat{G} \hat{P}$, all of these conserve total capital c_{tot} .

The capital and stock at the end of each cycle operator is written using the following short-hand notation:

$$\begin{aligned}
\hat{c}_t \hat{P} a &= \hat{c}_{\hat{P}} a = c_{\hat{P},a} a \\
\hat{c}_t \hat{G} \hat{P} a &= \hat{c}_{\hat{G}} a = c_{\hat{G},a} a \\
&\dots \\
\hat{c}_t \hat{W} \hat{S} \hat{F} \hat{G} \hat{P} a &= \hat{c}_{\hat{W}} a = c_{\hat{W},a} a = c_{t+1} a \\
\hat{\mu}_t \hat{W} \hat{S} \hat{F} \hat{G} \hat{P} f &= \hat{\mu}_{\hat{W}} f = \mu_{\hat{W},f} f = \mu_{t+1} f \\
\forall a \in P \cup F, \forall f \in F
\end{aligned}$$

Every repetition is called a cycle, or a time step, and it acts on all agents $|\psi_t\rangle = \{P, F\}$. During **Production** \hat{P} , Factories pay salaries by evenly diving all it's capital by its

workers,

$$\begin{aligned}\hat{c}_{\hat{P}} W_t(F) &= \left\{ \left\{ \hat{c}_t p + \frac{\hat{c}_t f}{|W_t(f)|} \mid p \in W_t(f) \right\} \mid f \in F \right\} \\ \hat{c}_{\hat{P}} F &= 0\end{aligned}$$

Each factory is left with 0 capital and produces an amount of stock defined by the production function depicted in figure 2.2

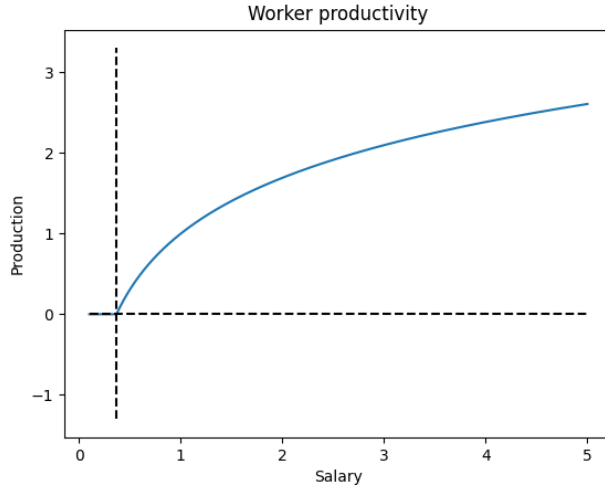


Figure 2.2: Production function

$$\begin{aligned}\hat{\mu}_{\hat{P}} f &= \mu_{\hat{P},f} f = \left(\mu_f + \log\left(\frac{\text{salary}_t(f)}{m_{\text{wage}}}\right) \right) f \\ \text{salary}_t(f) &= \frac{c_f}{|W_t(f)|}\end{aligned}$$

This function also defines the minimum wage, setting $\Delta\mu = 0 \Leftrightarrow m_{\text{wage}} = e^{-1}$ is the default value for minimum wage

Lastly, the stock price is defined:

$$\Delta\text{price}(f) = \min \left\{ 2, 0.2 + \frac{\mu_{\hat{P}}(f) - \mu(f)}{\mu(f)} \right\}$$

$$\text{price}_t(f) = \frac{1}{\mu_{\hat{P}}(f)} \left(\Delta\text{price}(f) c_f (\mu_{\hat{P}}(f) - \mu(f)) + \rho \text{price}_{t-1}(f) \mu(f) \right)$$

$\Delta\text{price}(f)$ being the return ratio per unit of stock produced this cycle, $\Delta\text{price}(f) \in [1, 2]$ and grows proportionally to factory stock, this models the law of supply and demand, the

smaller leftover stock ($\mu_{\hat{P}}(f) - \mu(f)$) a factory has, the more profit it expects, and vice versa. $price_t(f)$ is the final price per unit of stock. The price is set using a weighed median between the price of stock produced this cycle and the previous price. The price for stock produced this cycle is given by the profit margin multiplying cost of production, $\Delta price_t(f) c_f$, and for last cycle, a devaluation or appreciation effect is translated through ρ , if $\rho < 1$, stock devalues, if $\rho > 1$, stock appreciates. $\rho = \frac{2}{3}$ by default.

The **GoodsMarket** is divided into essential market \hat{G}_λ and luxury market \hat{G}_ν , $\hat{G} = \hat{G}_\nu \hat{G}_\lambda$ both markets iterate over every person the following steps: select a factory, trade with that factory, repeat. The selection step creates an ordered set of factories with non-zero stock sorted by price, $F_{G,\mu}$, and selects a factory by sampling this set using the same normal distribution depicted in figure 2.1, $\text{sample}(\mathbb{Z}^+, |F_{G,\mu}|)$. If this set is empty, that market ends, as there is no more factories to trade. Alternatively, person trades with the sampled factory: transfers capital from person to factory and destroys traded stock. In the essential market, person stops trading when it has traded one unit of essential stock or its capital reaches zero, on luxury market, person only stops trading when its luxury available capital (defined further) reaches zero. The sampling selection mechanism is mapped to

$$\text{select}(A) = a \Leftrightarrow a = i\text{'th element in } A, i = \text{sample}(\mathbb{Z}^+, |A|)$$

and leads to what could be called a semi-perfect market, or weighted random market, where factories with lower prices are more likely to trade, as the probability for smaller indices of the sorted factory list is bigger, but more expensive factories that would not trade on a perfect market, may also trade, creating entropy proportional to the standard deviation of the distribution. This entropy could translate a multitude of real world parameters, such as distance, personal preference, marketing, etc. As this market repeats a given operation for every person, it can be said that

$$\hat{G}_\mu = \text{attemptTrade}_\mu(p_1) \text{attemptTrade}_\mu(p_2) \dots$$

Essential market tracks the capital available to person p by $c_\lambda(p)$, stock consumed by person p using $g_\lambda(p)$ and stock available for trade to factory f using $g_\lambda(f)$, these are initialized by

$$c_\lambda(p) = c_{\hat{P},f}, g_\lambda(p) = 0, \forall p \in P, \text{ and } g_\lambda(f) = \lambda_{\hat{P},f} \forall f \in F_\lambda.$$

$$\begin{aligned}
\text{attemptTrade}_\lambda(p) &= \begin{cases} 1 & \text{if } |F_{G,\lambda}| = 0 \vee c_\lambda(p) = 0 \\ \text{attemptTrade}_\lambda(p) \times \text{trade}_\lambda(f, \alpha, p) & \text{else} \end{cases}, \\
f &= \text{select}(F_{G,\lambda}), \\
\alpha &= \min \left\{ 1, \begin{cases} g_\lambda(f) - g_\lambda(p) & \text{if } c_\lambda(p) \geq \text{price}(f) \times g_\lambda(f) - g_\lambda(p) \\ \frac{c_\lambda(p)}{\text{price}(f)} & \text{if } g_\lambda(p) < \text{price}(f) (g_\lambda(f) - g_\lambda(p)) \end{cases} \right\}, \\
F_{G,\lambda} &= \{f \in F \text{ sorted by } \text{price}(f) \mid g_\lambda(f) > 0\}.
\end{aligned}$$

The trade_μ operator has the following behaviour:

$$\begin{aligned}
\text{trade}_\mu(f, \alpha, p) &\Rightarrow c_\mu(p) \rightarrow c_\mu(p) - \text{price}(f) \times \alpha, \\
&\Rightarrow g_\mu(p) \rightarrow g_\mu(p) + \alpha, \\
&\Rightarrow g_\mu(f) \rightarrow g_\mu(f) - \alpha
\end{aligned}$$

At the end of essential market, the leftover capital for each person, $c_\lambda(p)$, will be allocated for luxury and shares markets in a predetermined manner, $\%_\nu = 40\%$ for luxury, $\%_s = 40\%$ for shares market and saving $\%_l = 20\%$ for next cycle.

The **luxury market** tracks the capital available to person p by $c_\nu(p)$, stock consumed by person p using $g_\nu(p)$ and stock available for trade to factory f using $g_\nu(f)$, these are initialized by

$$c_\nu(p) = c_\lambda(p), g_\nu(p) = 0, \forall p \in P, \text{ and } g_\nu(f) = \nu_{\hat{P},f} \forall f \in F_\nu.$$

$$\begin{aligned}
\text{attemptTrade}_\nu(p) &= \begin{cases} 1 & \text{if } |F_{G,\nu}| = 0 \vee c_\nu(p) = 0 \\ \text{attemptTrade}_\nu(p) \times \text{trade}_\nu(f, \alpha, p) & \text{else} \end{cases}, \\
f &= \text{select}(F_{G,\nu}), \\
\alpha &= \begin{cases} g_\nu(f) - g_\nu(p) & \text{if } c_\nu(p) \geq \text{price}(f) \times g_\nu(f) - g_\nu(p) \\ \frac{c_\nu(p)}{\text{price}(f)} & \text{if } g_\nu(p) < \text{price}(f) (g_\nu(f) - g_\nu(p)) \end{cases} \\
F_{G,\nu} &= \{f \in F \text{ sorted by } \text{price}(f) \mid g_\nu(f) > 0\}.
\end{aligned}$$

The final state of goods market can be seen as:

$$\begin{aligned}
\hat{c} \hat{G} \hat{P} p &= \left(c_{\hat{P},p} - (c_\lambda(p) + c_\nu(p)) \right) p \\
\hat{c} \hat{G} \hat{P} f &= \left(c_{\hat{P},p} + (g_\mu(f) \times \text{price}(f)) \right) f \\
\hat{\mu} \hat{G} \hat{P} f &= \left(\mu_{\hat{P},p} - g_\mu(f) \right) f
\end{aligned}$$

During **Funding**, factories start by projecting their attempted production, μ_{proj} , for next time step, using the estimation function depicted in figure 2.3:

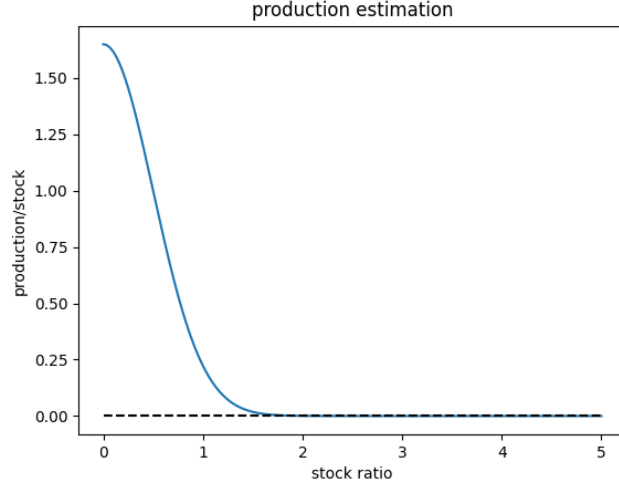


Figure 2.3: Production estimate

This function is defined by

$$\begin{aligned}\mu_{\text{proj}}(f) &= \mu_{\hat{P},f} \exp\left(-\frac{\text{ratio}(f)^2}{k} + k\right) + 1 \\ \text{ratio}(f) &= \frac{\mu(f)}{\mu_{\hat{P}}(f) - \mu(f)} \\ c_{\text{proj}}(f) &= \overline{c/\mu} \times \mu_{\text{proj}}(f) \\ \overline{c/\mu} &= \sum_{f \in F} \frac{|W(f)| \text{salary}(f)}{\mu_{\hat{P}}(f) - \mu(f)}\end{aligned}$$

Where k is a simulation constant with a default value of $k = 0.5, k \in]0, 1[$. c_{proj} the projection of the capital needed for producing this amount of stock, and $\overline{c/\mu}$ the average cost for producing a unit of stock. Then, if a factory raised enough capital during goods market in order to produce the projected stock, all remaining capital is distributed over the share holders, if it does not, then it must fund itself on the share market by selling shares.

$$\begin{aligned}c_{F,p} &= c_{\hat{G},p} + \sum_{f \in S(p)} \begin{cases} (c_{\hat{F},f} - c_{\text{proj}}) S(f,p) & \text{if } c_{\hat{F},f} > c_{\text{proj}}(f) \\ 0 & \text{if } c_{\hat{F},f} \leq c_{\text{proj}}(f) \end{cases}, \forall p \in P \\ \hat{F} f &= \begin{cases} \text{distribute}(f) f & \text{if } c_{\hat{G},f} > c_{\text{proj}}(f) \\ \text{fund}(f) f & \text{if } c_{\hat{G},f} \leq c_{\text{proj}}(f) \end{cases}, \forall f \in F\end{aligned}$$

For each factory, each of its share holders receives a percentage of its excess capital equal

to the value of their share in that factory, and nothing if no excess capital exists. In the latter case, the factory attempts to receive the remaining capital by selling its shares on the share market, if it succeeds, every share holder will now hold a smaller share of the factory, and new share holder(s) will exist. The distribute and fund functions are defined in the following manner:

$$\begin{aligned} \text{distribute}(f) &\Rightarrow \hat{c}_t \hat{F} \hat{G} \hat{P} f = c_{\text{proj}}(f) f \\ \text{fund}(f) &\Rightarrow m_1(f) = \frac{c_{\text{proj}}(f) - c_{\hat{G},f}}{c_{\hat{G},f} + \text{price}_t(f)\mu_{\hat{P},f} + \text{salary}_t(f)|W(f)|}, f \in M_1 \end{aligned}$$

$M(f)$ being the share amount (percentage of f) to be sold in the shares market, the capital needed divided by the total factory value, also referred as

$$\text{val}(f) = c_{\hat{G},f} + \text{price}_t(f)\mu_{\hat{P},f} + \text{salary}_t(f)|W(f)| \text{ capital shares}^{-1}$$

The **SharesMarket** is divided into primary and secondary, $\hat{S} = 1 + \hat{S}_1 \hat{S}_2$, during the primary shares market, factories sell shares to persons, acquiring capital, during the secondary market, persons may sell shares that they hold over factories to also acquire capital for themselves.

Much like the luxury market, shares market defines a share available capital, $c_s(p) = \%_s \times c_\lambda(p)$, factory capital $c_s(f) = c_{\hat{F},f}$ and two new share-like objects, $T_1(p)$ holds all factories that sold shares to person p , $T_1(p, f)$ holds the share amount that factory f sold to p . Now, for secondary market, T_2 must hold persons, factories and share amounts, so $T_2(p')$ holds all persons that sold shares to person p' , $T_2(p', p)$ all factory from which person p sold shares to person p' , and $T_2(p', p, f)$ the amount of shares sold from person p to p' on factory f . To exemplify, if only one share was sold from person p to p' in factory f , the share amount is $T_2(p, p', f) = T_2(p, p', T_2(p, p'))$, as $T_2(p, p')$ is a set of only one factory element.

Primary shares market attempts to trade shares for all factories that require funding by picking a random person with share available capital. The total capital spent on primary

market by person p is defined using $c_1(p) = \sum_{f' \in T_1(p)} \text{val}(f') \times T_1(f', p)$

$$\begin{aligned}\hat{S}_1 &= \bigcup_{f \in M_1} \text{attemptSell}(f, p), p \in_R P_{\hat{S}_1} \\ \text{attemptSell}(f, p) &= \left(\begin{cases} 1 & \text{if } \alpha = m_1(f) \vee |P_{\hat{S}_1}| = 0 \\ \text{attemptSell}(f, p') & \text{if } \alpha < m_1(f) \end{cases} \right) \times \text{sell}_1(f, \alpha, p), \\ \alpha &= \begin{cases} m_1(f) & \text{if } m_1(f) \times \text{val}(f) \leq c_s(p) - c_1(p) \\ \frac{c_s(p) - c_1(p)}{\text{val}(f)} & \text{if } m_1(f) \times \text{val}(f) > c_s(p) - c_1(p) \end{cases}, \\ p' &\in_R P_{\hat{S}_1}, \\ P_{\hat{S}_1} &= \left\{ p \in P \mid c_s(p) - c_1(p) > 0 \right\}\end{aligned}$$

If $|P_{\hat{S}_1}| = 0$, no more shares will be sold during primary market nor secondary market, as all share market available capital has been exhausted The sell operator will have the following behaviour:

$$\begin{aligned}\text{sell}_1(f, \alpha, p) &\Rightarrow p \in T_1(f), f \in T_1(p), T_1(f, p) \rightarrow T_1(f, p) + \alpha, \\ p &\in S(f), f \in S(p), S(f, p) \rightarrow S(f, p) + \alpha, \\ m_1(f) &\rightarrow m_1(f) - \alpha, \\ c_s(p) &\rightarrow c_s(p) - \alpha \times \text{val}(f), \\ c_s(f) &\rightarrow c_s(f) + \alpha \times \text{val}(f)\end{aligned}$$

Secondary shares market attempts to trade shares for all persons that require funding by picking a random person with share available capital.

Persons will only sell shares during secondary market if they may not be capable of fully consuming essential during the next cycle, this future capital needed projection is defined by the sum of essential capital projection and luxury capital.

$$c_{\text{ess}} = \frac{1}{|F_\lambda|} \sum_{f \in F_\lambda} \text{price}(f)$$

The total capital spent on primary and secondary market by person p is defined using $c_2(p) = c_1(p) + \sum_{p' \in T_2(p)} \left(\sum_{f \in T_2(p, p')} \text{val}(f) \times T_2(p, p', f) \right)$. For next step it is also important to

introduce the mean salary per person, $\overline{\text{salary}} = \frac{1}{|P|} \sum_{f \in F} |W(f)| \times \text{salary}(f)$

$$\begin{aligned}
M_2 &= \{p \in P \mid \%_l \times c_\lambda(p) + \overline{\text{salary}} < c_{\text{ess}} \wedge |S(p)| > 0\} \\
m_2(p) &= \min \left\{ \%_l \times c_\lambda(p) + \overline{\text{salary}} - c_{\text{ess}}, \sum_{f \in S(p)} S(p, f) \right\}, p \in M_2 \\
\hat{S}_2 &= \bigcup_{p \in M_2} \text{attemptSell}(p, p'), p' \in_R P_{\hat{S}_2} \\
\text{attemptSell}(p, p') &= \left(\begin{cases} 1 & \text{if } \alpha = m_2(p) \vee |P_{\hat{S}_2}| = 0 \\ \text{attemptSell}(p, p'_2) & \text{if } \alpha < m_2(p) \end{cases} \right) \times \text{sell}_2(p, \alpha, p') \\
\alpha &= \begin{cases} m_2(p) & \text{if } m_2(p) \leq c_s(p') - c_2(p') \\ c_s(p') - c_2(p') & \text{if } m_2(p) > c_s(p') - c_2(p') \end{cases}, \\
p'_2 &\in_R P_{\hat{S}_2} \\
P_{\hat{S}_2} &= \{p' \in P \mid c_s(p') - c_2(p') > 0\}
\end{aligned}$$

It is of note that m_1 contains shares and m_2 contains needed capital, and same applies to both α 's. The sell_2 operator will have a more complicated behaviour, as it will trade shares until α capital has been traded, so a second, simpler operator will handle trades, $\text{trade}(p, \alpha', p', f)$ trades α' shares in factory f from person p to person p' , and $\alpha' \times \text{val}(f)$ capital from person p' to person p

$$\begin{aligned}
\text{sell}_2(p, \alpha, p') &= \prod_{f \in S(p)} \begin{cases} \text{trade}(p, \alpha', p', f) & , \text{if } m_2(p) > 0 \\ & , \alpha' = \min \{S(p, f), m_2(p)/\text{val}(f)\} \\ 1 & , \text{if } m_2(p) = 0 \end{cases} \\
\text{trade}(p, \alpha', p', f) &\Rightarrow p' \in S(f), f \in S(p'), S(f, p') \rightarrow S(f, p') + \alpha', \\
&S(f, p) \rightarrow S(f, p) - \alpha', \text{if } S(f, p) = 0, p \notin S(f), f \notin S(p), \\
&m_2(p) \rightarrow m_2(p) - \alpha' \times \text{val}(f), \\
&c_s(p) \rightarrow c_s(p) - \alpha' \times \text{val}(f), \\
&c_s(p') \rightarrow c_s(p') + \alpha' \times \text{val}(f)
\end{aligned}$$

At the end of share market, because during primary market the normalization of shares in factories was violated, it is then recovered using a normalize operator norm , so that \hat{S} becomes, in fact, $\hat{S} = \text{norm} \hat{S}_1 \hat{S}_2$. This could be achieved by normalizing at every trade during primary market, reaching the same end state.

WorkersMarket handles finding new workers for every factory in the next cycle. This

is achieved by iterating over every person: finding the highest salary factory, if this salary is higher than minimum wage, person becomes worker in factory, update factory projected salary, repeat. There is no need to implement entropy in this stage because there is more persons than factories, so entropy would complicate the procedure without providing further realism.

As this market repeats a given operation for every person, $\hat{W} = \hat{\text{find}}^{|P|} = \hat{\text{find}}(p_1) \hat{\text{find}}(p_2) \dots$ it also defines a new set, H , containing every person that has been hired. This $\hat{\text{find}}$ operator acts on one person only, and behaves in the following manner:

$$\hat{\text{find}}(p) = \begin{cases} \mathbb{1} & \text{if } c_{\text{proj}}(f)/(1 + |W_{t+1}|) < m_{\text{wage}} \\ p \in W_{t+1}(f) \wedge p \in H & \text{if } c_{\text{proj}}(f)/(1 + |W_{t+1}|) \geq m_{\text{wage}} \end{cases},$$

$$p \in_R \{p \in P \mid p \notin H\},$$

$$f = \max \left\{ f \in F \text{ sorted by } c_{\text{proj}}(f)/(1 + |W_{t+1}(f)|) \right\}$$

2.3 Government

A separated organ of this system is the government. It is separated because it is not essential to the system, but it does add a layer of complexity needed to explore interesting outcomes.

The government acts at the end of the cycle, and redistributes its raised capital equally over all persons, this capital can be raised in one or both ways: **Transaction** taxing and **Wealth cap** taxing.

Transaction taxing occurs every time a transaction of capital occurs during a trade in the goods market, shares market and salary payment, The government taxes a given percentage of the total capital in that transaction, 10% by default, and adds it to the raised capital.

Wealth cap taxing occurs at the end of the cycle, where the government takes all capital from every person that has capital higher than a set cap, $\frac{\text{cap}}{|P|} \sum_{p \in P} c_p$. By default, cap= 4, a person can hold four times the hypothetical perfectly distributed capital.

This concludes all the possible operations carried out to create the system and its results, it is noteworthy that this model exists as a program, so all mentioned above is only a mathematical representation, the program source is available on TODO where do i put the code.

Chapter 3

Results

Every result presented next is a median of ten runs of one thousand cycles.

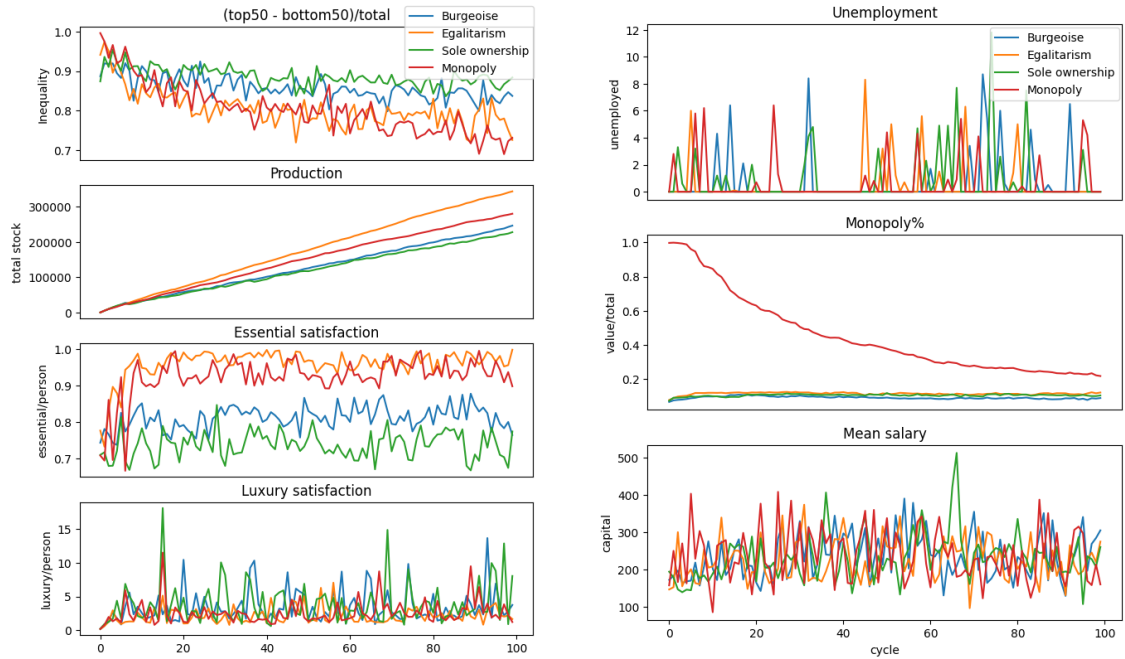


Figure 3.1: initial conditions

The first system analysis, present in figure 3.1, studies the result of all different initial conditions on a system with no government. Inequality is calculated by taking the difference of the sum of capital held by the top 50% persons and the bottom 50%, and dividing it by the total capital. if inequality = 1, the top 50% persons hold all capital, if inequality = 0.5, wealth is equally distributed. Total stock sums the stock held by all factories, essential and luxury. Essential and luxury satisfaction are the average goods consumed by each person, These results show that, for any initial condition, the system evolves in a similar way, unemployed people's salary were set to zero. Upon observation, it can be noted that initial conditions

have a small effect on these results. The most notable difference is found in the Monopoly%, calculated by dividing the maximum total share value held by persons, dividing by the sum of these. $s(p) = \sum_{f \in S(p)} S(p, f)$, $\text{Monopoly\%} = \max\{s(P)\} / \sum_{p \in P} s(p)$. The initial conditions, even if not very different for the entire system's state, plays a major role in differentiating persons, in figure TODO person capital (id on x)

It is interesting to note that, for all initial conditions, mean salary was, for most cycles, much higher than minimum salary, around 250. This value is affected by the number of people in the system and will be explored further.

Bibliography