Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing

Agent-Based Simulation of Community Currencies with Basic Income

Semester Thesis

Damla Welti

dwelti@ethz.ch

Distributed Computing Group
Computer Engineering and Networks Laboratory
ETH Zürich

Supervisors:
Yann Vonlanthen, Alain Brenzikofer
Prof. Dr. Roger Wattenhofer

July 6, 2022

Acknowledgements

First and foremost, I would like to express my appreciation to both of my supervisors, Yann Vonlanthen and Alain Brenzikofer, for their continuous guidance and valuable advice at every stage of the project. My gratitude extends to Prof. Dr. Roger Wattenhofer for giving me the opportunity to conduct this thesis. Finally, a special thanks goes to the Encointer association for enabling this work in the first place.

Abstract

In economically disadvantaged countries, citizens often struggle to meet their basic needs. Encointer is a new technology that allows communities to launch their own digital currencies, so-called community currencies, to supplement national legal tender. The community currencies can not only be used for purchasing goods and services, but are also issued as a regular basic income among participants. To compensate for this rising inflation, the value of a token is lowered constantly through a mechanism known as demurrage. This semester thesis is a first attempt to simulate the interaction of these mechanisms. The aim is then to investigate whether such an environment can help citizens with little or no income to meet their basic needs and increase purchasing power.

To this end, a simulator combining the mechanisms of the dual monetary system was developed. An agent-based model was used to give individuals variations and random fluctuations. The simulator then ran through various scenarios for a fictitious community with low-income in a developing country to find answers to the questions posed.

Contents

Acknowledgements **i**

Abstract **ii**

1 Introduction **1**

1.1 Motivation 1

1.2 Assignment 1

1.3 Overview 2

2 Background **3**

2.1 Agent-Based Model 3

2.2 Complementary and Community Currencies 3

2.3 Basic Income 4

2.4 Demurrage 4

2.5 Gini Index 5

2.6 S-Curves 5

2.7 Digraphs 7

2.8 cadCAD 7

3 Method **9**

3.1 Representation of the Encointer Community 9

3.1.1 Agents Network 9

3.2 Overview of the Simulation 10

3.3 Ceremony 11

3.3.1 Attendance of Reputables 11

3.3.2 Community Growth through Newbies 13

3.3.3 Basic Income 16

3.3.4 Restriction for Bootstrappers 16

3.4 Sale and Purchase of Products 16
3.5 Demurrage 19
3.6 Product Manufacture 21
3.7 Preference Change 21
3.8 Inequality Measure 21

4 Results and Discussions 25
4.1 The OGU Community 25
4.1.1 Community without National Currency 27
4.1.2 Community with Indigent Members 33
4.1.3 Improving Welfare for Indigent Members 34
4.2 Limitations 39
4.2.1 Consumer Buying Behavior and Community Growth . . . 39
4.2.2 RAM Memory 39

5 Conclusion and Future Work 41
5.1 Conclusion 41
5.2 Future Work 41

Bibliography 43

A cadCAD Structure A-1

B Source Code B-1
B.1 GitHub Repository B-1
B.2 Simulation Parameters B-1

Chapter 1

Introduction

1.1 Motivation

More than a quarter of a billion people could slip into extreme poverty this year, living on less than $1.90 income per day.[1] A way to fundamentally change the global standard of living is to give individuals sufficient money to meet their basic needs, regardless of whether they work or can contribute in any way.[2] Most advocates of this idea aim to introduce it at the national level, as a monthly income for all legal tax residents, paid in state-issued currency. According to World Bank Data, however, over 1 billion people have no official proof of identity and thus cannot open a bank account.[3] The Encointer association seems to have the answer with a bottom-up approach.

Encointer is a digital platform that allows any community to launch their own cryptocurrency and create a micro-economy. The value of a token is determined by local supply and demand, and is not subject to changing macroeconomic factors. Besides paying for goods and services in this currency, members of the Encointer community can actively take part in regular key-signing meetups to receive a "basic income". A citizen needs no state-issued ID document to register for such a meetup. With the supplement of a currency next to the state-issued money, the purchasing power is likely to increase. Encointer currencies are moreover subject to a mechanism called "demurrage" that discourages hoarding of the income and thus, stimulates the local economy.

With the launch of the first Encointer communities, there is a desire to gain more insight into the emerging ecosystem.

1.2 Assignment

This work shall focus on developing a framework to simulate the economics of such an Encointer currency. The currency should be complementary to the national legal tender and lose its nominal value over time due to demurrage. In particular, Encointer's regular key signing meetups as a prerequisite for receiving basic income should be taken into account. The model will initiate an agent-based approach to run the same algorithm with individual agent behaviors. An initial application of the tool then aims to provide insights into community behavior and attempts to answer the following questions:

- Does the Encointer protocol help citizens who have little or no income to meet their basic needs?
- Will consumption of products increase with the introduction of an Encointer currency, and thus boost the local economy?
- Can a stable local market be created with the use of Encointer?
- Are we moving towards an equal distribution of wealth using the Encointer framework?

1.3 Overview

First, we give a theoretical background in Chapter 2. In Chapter 3, we delve into the model and implementation of our simulator. Then in Chapter 4, the simulator is applied to a fictitious community for different scenarios. Finally, in Chapter 5 we provide the conclusion and some ideas for future work.

Chapter 2

Background

2.1 Agent-Based Model

Socio-economic systems are difficult to represent in conventional equation-based models because most behaviors lack mathematical formalization. A popular approach for the analysis of complex systems is agent-based simulation, in which the interaction of agents is repeatedly simulated over time. Although there is no universal agreement on the exact definition of the term "agent", there is consensus that agents are diverse, heterogeneous, and dynamic in terms of their characteristics and rules of behavior. A behavior can be described by simple if-then rules or extend to complex behavioral models from fields such as artificial intelligence. This not only makes the modeling approach far more flexible, but also allows for individual variations and random influences.[4, 5]

2.2 Complementary and Community Currencies

The terms "community currencies" and "complementary currencies" refer to monetary systems designed to exchange goods or services. They are usually used to perform a function that cannot be adequately fulfilled by conventional money. Complementary currencies do not replace the national currency, but rather operate in parallel to it. Community currencies emphasize that the creation and use of the currency concerns small, closed organizations, in the sense that membership is required.[6, 7]

To establish a dual monetary system, a network of local businesses need to accept the alternative currency as a medium of exchange for goods and services in addition to the national legal tender. However, attention should be paid not to create more money than the local economy can absorb in terms of productive capacity, otherwise this may lead to inflation.[6]

An essential characteristic of complementary or community currencies is that they remain and circulate within their communities to a much greater extent than state sponsored currencies, which have little connection to the local community. In developing countries where local economies are still strong, complementary or community currencies not only help maintain these infrastructures, but can also protect them from global financial crises.[6]

2.3 Basic Income

The idea behind basic income is that individuals should receive a regular income that is sufficient to cover their basic needs. According to BIEN (Basic Income Earth Network)[8], there are five key attributes that define it:

- **Periodic:** Payment at regular intervals
- **Cash:** Payment in a suitable exchange medium so that recipients can decide for themselves what they want to spend it on
- **Individual:** Payment on an individual basis
- **Universal:** Payment to everyone
- **Unconditional:** Payment regardless of work or any contribution

Studies have shown that basic income has significant and long-lasting positive effects on people's income, employment, health, education, engagement, and even creativity.[6]

2.4 Demurrage

The concept of a monetary system based on a built-in depreciation was first proposed by the German monetary and social reformer Silvio Gesell (1862-1930). Meanwhile, demurrage has become a common expression in the context of complementary currencies for the built-in reduction of the nominal value of a currency over time. The depreciation is at the expense of the holder of the money throughout the entire time it is held.[9]

Typically, demurrage is a fixed amount as a percentage of the original nominal value, e.g. 1% per month. The depreciation should be a steady process on a daily basis, like interest on a savings account.[9]

The main objective of demurrage is to prevent hoarding.[9] Complementary currencies with demurrage tend to circulate much faster than government-sponsored currencies, resulting in far greater economic activity.[6]

2.5 Gini Index

A well-established statistical measure of inequality in an economy is the Gini index, also known as the Gini coefficient or Gini ratio. The Gini index is a single decimal number which ranges from 0 to 1. The higher the value is, the more pronounced is the inequality. A value of 0 indicates perfect equality, meaning all people of a population compared have exactly the same income or wealth. Conversely, a population where one member receives all the income or wealth, while everyone else having nothing, the Gini index correspond to a value of 1.[10]

The Gini index can be derived from the **Lorenz curve**, which plots the distribution of income or wealth within an economy. When depicting the income distribution, the percentiles of the population from poorest to richest is shown on the horizontal axis, while the cumulative normalized income is plotted on the vertical axis. The result is a curve which is located in the unit square of the Cartesian diagram's first quadrant. A straight diagonal line with a slope of 1 is known as the **line of equality**. The more the Lorenz curve lies below this line, the more inequality results.[11]

From a graphical perspective, the Gini index is the ratio

$$Gini\ index = \frac{A}{A+B}, \tag{2.1}$$

where A is area between the perfect equality line and the Lorenz curve representing a populations income or wealth, and $A+B$ is the total area under the perfect equality line.[11]

Since the area below the perfect equality line is 0.5 by definition, the area A corresponds to $0.5 - B$ and thus 2.1 can be expressed as

$$Gini\ index = \frac{0.5 - B}{0.5}. \tag{2.2}$$

An alternative approach to define the Gini index is using the mean absolute deviation,[13] which will not be discussed here.

2.6 S-Curves

S-curves, short for sigmoid curves, are mathematical functions with an S-shaped graph. An example of this class of functions is the standard logistic function, known as *expit*, which will be used as a basis at various parts of our simulation. It represents a continuous one-dimensional probability distribution and is defined by the formula

$$expit(x) = \frac{1}{1+e^{-x}}. \tag{2.3}$$

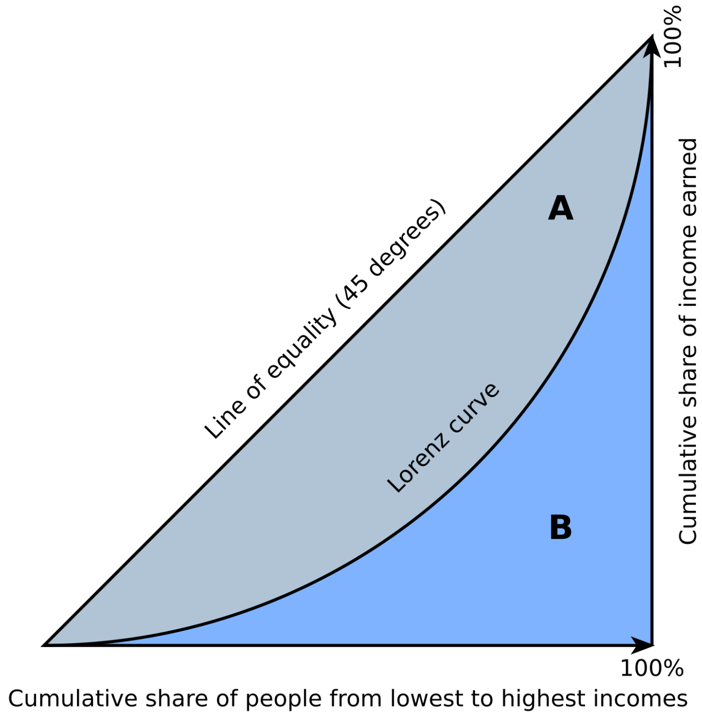

Figure 2.1: Graphical representation of the Gini index.[12]

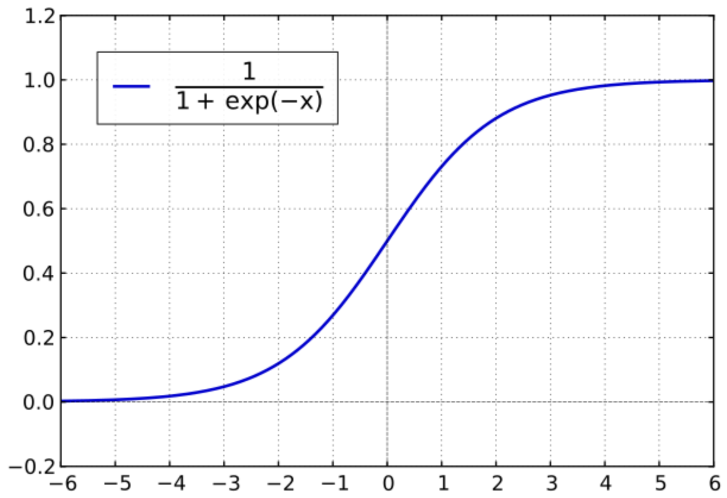

Figure 2.2: Plot of the *expit*-function which intercepts the y-axis at 0.5 and ranges between (0,1).[14]

The expit function is plotted in Figure 2.2. Its inverse is called the *logit* and is derived as

$$logit(x) = ln(\frac{x}{1-x}). \tag{2.4}$$

Note that $logit(x)$ for x=1 is undefined.

2.7 Digraphs

A digraph $G = (V, A)$, short for directed graph, consists of a set of vertices V and a set of ordered pairs of vertices A, called arcs. An arc (v_1, v_2) can be visualized as a directed line from vertex v_1 to vertex v_2. The digraph is complete when each vertex has an outgoing arc to every other vertex. Two examples are shown in Figure 2.3.

2.8 cadCAD

cadCAD is a modular *Python*-based modeling framework and powerful simulation engine. It was developed as an internal software at BlockScience and released to the public as an open source project in 2019. The acronym cadCAD stands for ***c****omplex* ***a****daptive* ***d****ynamics* ***C****omputer* ***A****ided* ***D****esign* and provides the ability to describe complex systems at any level of abstraction. cadCAD supports various system modeling approaches, including the design of agent-based models

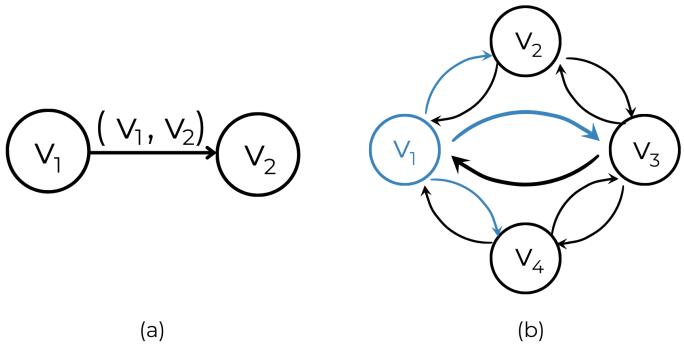

Figure 2.3: (a) Example of an arc (v_1, v_2). (b) Example of a complete digraph with vertex set $V = v_1, v_2, v_3, v_4$, where the same vertex cannot be visited twice in a row.

using the open-source *NetworkX* library.[15] The structure of a standard cadCAD notebook can be found in the Appendix A.

CHAPTER 3

Method

The following chapter describes the concept and the implementation of our simulator for an Encointer community.

The source code of our simulator is written from scratch using the programming language *Python*. We integrate the *NetworkX* library to use graph data structures and utilize the simulation engine *cadCAD* to execute our model. With the use of NetworkX, we follow an agent-based approach.

3.1 Representation of the Encointer Community

3.1.1 Agents Network

The Encointer community is modeled as a digraph, whereas a vertex, v_i is used to represent an agent i in the Encointer community. For simplicity, we assume that each agent is a consumer of products, while at the same time every agent offers exactly one type of product for sale. The product types of the agents are distinct to exclude competition. In the arc (v_1, v_2), v_1 represents an agent in the consumer role, while v_2 represents an agent in the seller role. However, in the arc (v_2, v_1), the agent v_2 is in the consumer role and v_1 switches to the seller role.

Agent Attributes

Each agent v_i has the following set of attributes:

- **Encointer Wallet, or Wallet:** Amount of the digital Encointer currency. The wallet cannot become negative.
- **National Currency Account, or Account:** Amount of money in the national currency. The account cannot become negative.
- **Wealth:** Total assets defined as *Wealth = Wallet + Account * Exchange Rate.*

- **Product Quantity:** Amount of the product in stock for sale. The amount cannot become negative.
- **Product Price:** Sales price of the product expressed in the Encointer currency. The price remains constant over time.
- **Production Speed:** Quantity of new product manufactured in one time step. The speed remains constant over time.

Attribute between two Agents

Every arc (v_i, v_j), $\forall\ i \neq j$, has an attribute called **Prefer Score** that expresses how much agent v_i wants to buy the product of agent v_j. If the prefer score is high, then v_i will buy from v_j with high probability in the next time step, whereas a prefer score of zero means that v_i will not to buy from v_j. The prefer score is non-negative.

3.2 Overview of the Simulation

Before going into the in-depth explanation of each individual simulation step, a brief overview will be given.

0. **Bootstrapping:** When launching a new Encointer currency, a number of bootstrappers perform a trustworthy setup ceremony. This is simplified by creating a set of agents at the beginning, all of whom receive the basic income.

The simulation runs in discrete time steps, where it is assumed that one time step corresponds to one day. Each time step is divided into the following substeps:

1. **Ceremony:** A check is made if a ceremony takes place that day. If this is the case, new participants can join the Encointer community and everyone who attends a meetup receives basic income.
2. **Sale and Purchase of Products:** Agents purchase products from each other and pay for these products with their Encointer wallet or national currency account.
3. **Product Manufacture:** New products are manufactured according to an agent's production speed.
4. **Preference Change:** The buying preferences are updated.
5. **Demurrage:** All account balances decrease due to demurrage.

3.3 Ceremony

Key signing ceremonies are regularly scheduled physical meetups of community members at randomly selected locations. The concept behind these gatherings is that each participant can receive a basic income. Since the number of meetups and their locations do not affect our simulation, we can model the ceremony as if it was a single gathering. If it is a ceremony day, not only agents in our economy can attend the ceremony, but also so-called "newbies" can participate and thereby become a member of the community. This is the only way in our simulation how one can become part of the Encointer community. Therefore, all agents who are part of the community, have successfully attended a meetup before and have good reputation. We call these agents "reputables". Among all the ceremony participants, a predetermined basic income will be issued to their Encointer wallets in the digital currency of the community.

Algorithm 1 Ceremony

if timeStep % ceremonyInterval = 0 **then**
    $n_{reputablesCeremony} \leftarrow [], k \leftarrow 0$
    for each agent v_i in $agentsNetwork$ **do**
        $p_i \leftarrow findProbabilityCeremonyAttendance(v_i.wealth, maxWealth)$
        $X_i \sim Bin(1, p_i)$
        if $X_i = 1$ **then**
            $n_{reputablesCeremony}[k++] \leftarrow v_i$
            $v_i.wallet \mathrel{+}= basicIncome$
        end if
    end for
    $n_{newbies} \leftarrow \min(n_{exponentialGrowth}, n_{logisticGrowth})$
    for $l \in \{1, ..., n_{newbies}\}$ **do**
        generate random agents with $basicIncome$
    end for
end if

3.3.1 Attendance of Reputables

If a reputable attends a ceremony to receive basic income depends primarily on its current financial situation. Those who have less assets are more likely to go to the ceremony. The first thing to determine is whether an agent has a large balance in the Encointer wallet or national currency account combined. The difficulty is that more and more Encointer tokens are issued over time, making it impossible to set a fixed value for a large balance. For this purpose, the maximum wealth of all reputables is searched at each time step. This maximum wealth is used as a reference to determine whether or not an agent has a lot of assets.

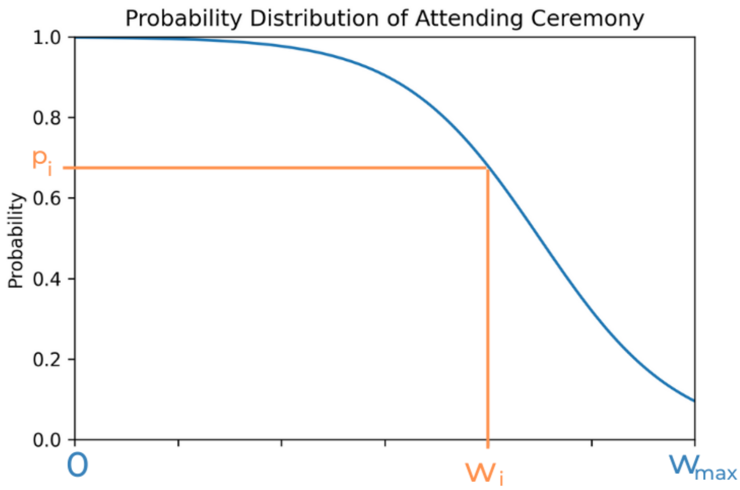

Figure 3.1: Probability distribution of an agent to attend a ceremony as a function of wealth in the community. If agent i has wealth w_i on day t, then its probability of attending the ceremony on day t is p_i.

Next, to determine the probability that a reputable participates in the ceremony, a mirrored S-curve is used. The x-axis stands for the wealth of an agent i and the y-axis represents the probability p_i that this agent attends the ceremony. An example of a probability distribution for attending a ceremony is shown in Figure 3.1.

To find the mirrored S-curve, we start by using the *expit*-function, which is shown in Figure 2.2. The *expit* ranges from 0 to 1 and can therefore be interpreted as a continuous probability distribution. We transform the *expit* by first scaling and translating, and then, since the probability must be inversely proportional to the agent's balance, we subtract the curve from the value of 1.

After finding the probability of an agent attending the ceremony, the decision whether the agent indeed goes to a meetup is implemented as a Bernoulli distribution: A agent i joins the ceremony with probability p_i and does not attend with probability q_i. In other words, $X_1, ..., X_N$ are independently distributed Bernoulli random variables, each with probability p_i taking the value 1 and with probability $q_i = 1 - p_i$ taking the value 0. This is equivalent to a Binomial distribution

$$X_i \sim Bin(1, p_i). \tag{3.1}$$

We can simply sum all Bernoulli distributed random variables $X_1, ..., X_N$

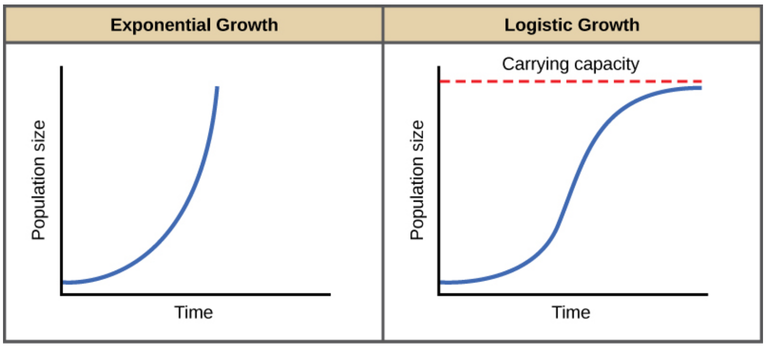

Figure 3.2: Example of an exponential and a logistic growth function.[16]

$$n_{reputablesCeremony} = \sum_{j=1}^{N} X_j, \tag{3.2}$$

to find the number of reputables that decide to participate in the ceremony. Note that $n_{reputablesCeremony} \subseteq N$.

3.3.2 Community Growth through Newbies

The Encointer protocol allows newbies to register for a ceremony and thus join the community. To ensure security, the protocol limits the maximum number of newbies joining at a ceremony to

$$n_{exponentialGrowth} = n_{reputablesCeremony} * newbieRatio. \tag{3.3}$$

If at each ceremony this number of newbies was used, the population would exhibit an exponential growth when plotted against time. Since in reality the number of people who can join the community is limited, a logistic growth is applied, which produces an S-curve. As populations become established, the logistic curve corresponds to exponential growth. At higher population densities, the growth rate slows down and reaches saturation when the size of the population approaches a threshold. An example of an exponential and logistic growth can be found in Figure 3.2.

Implementation of Community Growth

The community of Encointer should follow a logistic growth curve where the population size stabilizes at the predefined maximum population K. As basis of our logistic growth, we use again the *expit* function in Figure 2.2. We follow the steps in Figure 3.3 when going through the implementation.

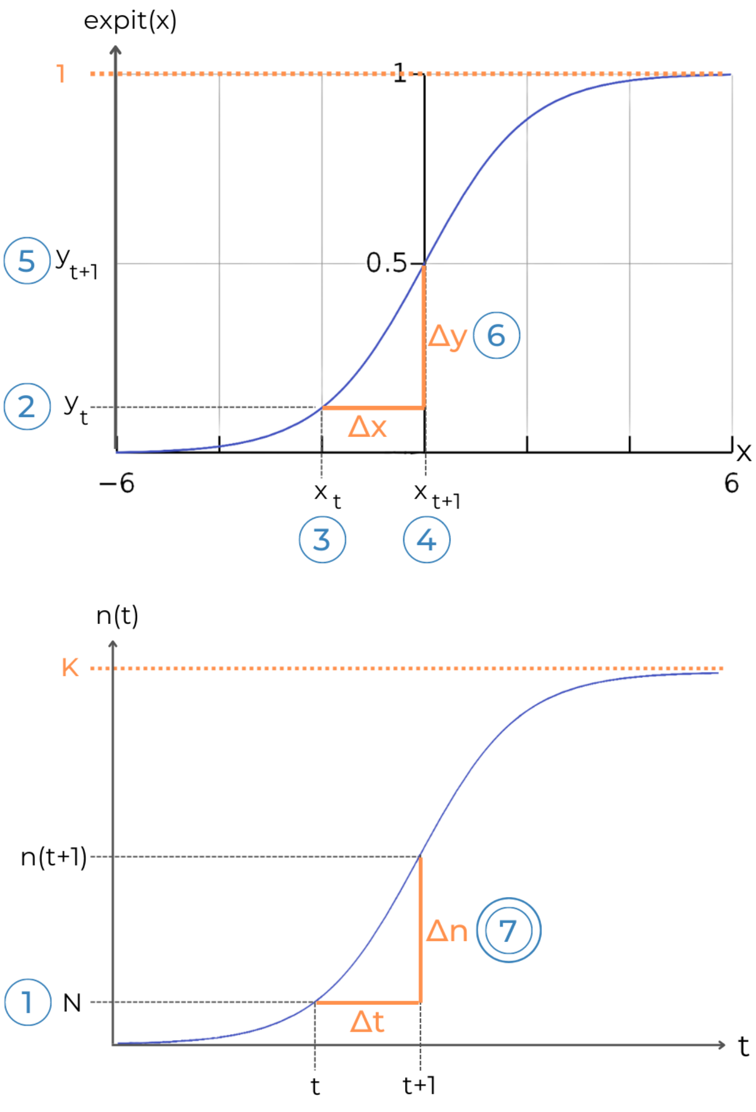

Figure 3.3: The iterative computation of the number of newbies joining the Encointer community during ceremony when following a logistic growth.

1. At time step t, we know that $n(t) = N$ people are part of the Encointer community.

2. From this, we can calculate the ratio of the number of individuals that are part of the community at time step t:

$$y_t = \frac{n(t)}{K} = \frac{N}{K}. \tag{3.4}$$

3. To map 3.4 to the x-axis, we need to use the reverse *expit* function, the so-called *logit* function, described in 2.4:

$$x_t = logit(y_t). \tag{3.5}$$

4. Next, we have to move a step forward in time to $t+1$:

$$x_{t+1} = x_t + \Delta x. \tag{3.6}$$

   The selection of parameter Δx can be defined in the simulation.

5. The ratio of the amount of people which should be part in Encointer if a logistic curve is used can be found as

$$y_{t+1} = expit(x_{t+1}). \tag{3.7}$$

6. Now, using 3.7 and 3.4, we can derive

$$\Delta y = y_{t+1} - y_t. \tag{3.8}$$

7. Finally, we find

$$\Delta n = n(t+1) - n(t) = \Delta y * K. \tag{3.9}$$

Using the $floor$ function on 3.9 and combining 3.4 - 3.8, we find

$$n_{logisticGrowth} = \lfloor (expit(logit(\frac{N}{K}) + \Delta x) - \frac{N}{K}) * K \rfloor, \tag{3.10}$$

which represents the amount of people which should be added in this time step due to logistic growth. A $floor$ function, denoted as $\lfloor x \rfloor$, takes a real number and returns the greatest integer less than or equal to that number.

Growth during Ceremony

If we combine the restriction we have from 3.3 with the equation 3.10 it follows:

$$n_{newbies} = \min(n_{exponentialGrowth}, n_{logisticGrowth}). \tag{3.11}$$

3.3.3 Basic Income

Every agent who attends the ceremony, reputable or newbie, receives a basic income. The total number of ceremony participants is

$$n_{totalCeremony} = n_{reputablesCeremony} + n_{newbies}, \tag{3.12}$$

and the total amount of Encointer tokens minted in a ceremony corresponds

$$tokensMinted = basicIncome * n_{totalCeremony}. \tag{3.13}$$

3.3.4 Restriction for Bootstrappers

To achieve community growth, the number of bootstrappers must be limited to

$$n_{bootstrappers} \geq \frac{1}{newbieRatio}. \tag{3.14}$$

Proof. Let us assume the inequality 3.14 is wrong, in the first ceremony it would follow that $n_{reputablesCeremony} < \frac{1}{newbieRatio}$. Therefore, $n_{exponentialGrowth} = 0$. In 3.11, $n_{newbies} = 0$ will be always chosen, and thus the number of newbies joining the community would always be 0.

Thus, to make a community growth happen, 3.14 must hold. □

3.4 Sale and Purchase of Products

In the course of a day, agents purchase products to satisfy their basic needs and beyond. Each agent is defined as a consumer of products and simultaneously offers a product type for sale. An overview of how products are sold and bought during a time step is provided by the Algorithm 2. First, we find all agents that buy products during the time step. Note that an agent can buy more than one product per day. However, there is a threshold parameter that defines the maximum number of daily purchases for an agent. Next, a seller is selected for that buyer that matches the buyer's product preferences and current financial situation. If a seller is found, a monetary transaction is made between the buyer and the seller. Then, the selling agent's product quantity is reduced by one. Finally, since the buyer has just purchased the product, its desire for that product decreases and so does its preference for that product.

Algorithm 2 Sale and Purchase of Products

```
buyers ← []
buyers ← findBuyers(agentsNetwork)                          ▷ see Algorithm 3
for buyer b_i in buyers do
    select one seller s_j for b_i                           ▷ see Algorithm 4
    for (b_i, s_j) do
        if b_i.wallet ≤ s_j.productPrice then              ▷ pay with Encointer wallet
            b_i.wallet −= s_i.productPrice
            s_i.wallet += s_i.productPrice
        else                                                ▷ pay with national currency
            b_i.account −= s_i.productPrice / exchangeRate
            s_i.account += s_i.productPrice / exchangeRate
        end if
        s_j.productQuantity −= 1
        (b_i, s_j).preferScore −= preferDecrease
    end for
end for
```

Finding the Buyers

To buy a product, we first need to find the all buyers of a time step, which is shown in Algorithm 3.

The probability of being selected as a buyer depends primarily on an agent's current financial situation. Those with more wealth make more purchases. Thus, we take a similar approach to attendance of a ceremony to dynamize our probability distribution over time. We find the maximum wealth when Encointer wallet and national currency account are combined. This is used as a reference point to determine to determine the agent's financial situation. Next, a scaled and transformed *expit* function is used to determine the probability that an agent will purchase a product. Again, the x-axis represents wealth, while the y-axis represents the probability p_i of an agent being selected as a buyer in that time step. An example of the probability distribution can be found in Figure 3.4.

To determine the probability of an agent making a purchase, we use the Binomial distribution: An agent i is asked whether it wants to make a purchase. The probability of a yes-answer to the question is p_i, and the probability of answering no is $q_i = 1 - p_i$. We repeat this question n times, which is the predefined maximum number of purchases an agent can make per day. We assume that these yes-no questions are independent over the course of a day and that the probability that an agent buys a product over the course of a day does not change. This is equivalent to a Binomial distribution

$$X_i \sim Bin(maxNrOfPurchases, p_i). \tag{3.15}$$

Algorithm 3 Find Buyers

1: $buyers \leftarrow [], k \leftarrow 0$
2: **for each** agent v_i in $agentsNetwork$ **do**
3:     $p_i \leftarrow findProbabilityBuyer(v_i.wealth, maxWealth)$
4:     $X_i \sim Bin(maxNrOfPurchases, p_i)$
5:     **while** $X_i > 0$ **do**
6:         $buyers[k++] \leftarrow v_i$
7:         $X_i --$
8:     **end while**
9: **end for**
10: shuffle($buyers$)
11: **return** $buyers$

Now X_i contains the number of times agent i was selected as a buyer. In lines 5-8 of the Algorithm 3, we store the index of the buyer in a list as many times it was selected. In line 9, this list is shuffled randomly.

Selecting the Product

After the buyers are picked for this time step, it is decided for each of them which product to buy. Since we assume that agents sell distinct product types, this is equivalent to selecting a seller. The logic is shown in Algorithm 4.

Algorithm 4 Select one Seller s_j for a fixed Buyer b_i

1: $potentialSellers \leftarrow [], k \leftarrow 0$
2: **for each** agent $v_i \neq b_i$ in $agentsNetwork$ **do**
3:     **if** $(v_i.productAmounts > 0)$ **then**
4:         **if** $(v_i.productPrice \leq b_i.wallet) \vee (\frac{v_i.productPrice}{exchangeRate} \leq b_i.account)$ **then**
5:             $potentialSellers[k++] \leftarrow v_i$
6:         **end if**
7:     **end if**
8: **end for**
9: $probabilitiesSellers \leftarrow [], l \leftarrow 0$
10: **for** s_{pot} in $potentialSellers$ **do**
11:     $probabilitiesSellers[l++] \leftarrow \frac{(b_i, s_{pot}).preferScore}{totalPreferScore}$
12: **end for**
13: $s_j \leftarrow$ random choice in $potentialSellers$ with $probabilitiesSellers$
14: **return** s_j

First, we look which products are in stock for sale and can be afforded by our buyer. We store the sellers of these products in a list, which can be seen lines 2-8 of the Algorithm 4. Then, out of all these potential sellers, we want to select

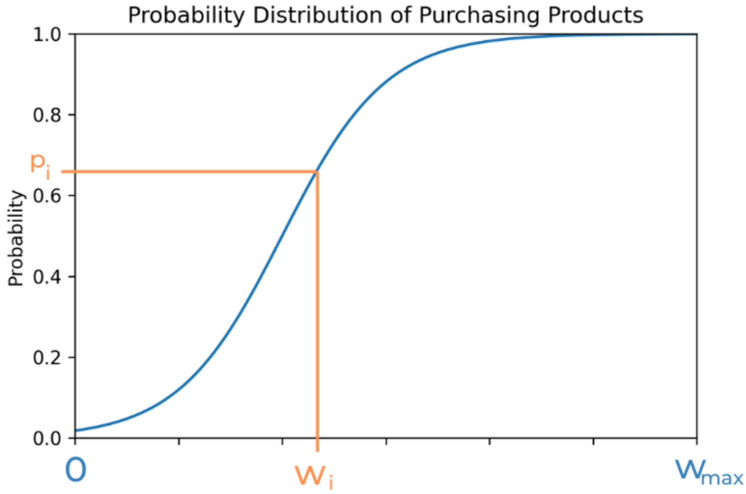

Figure 3.4: Probability distribution of an agent to buy products as a function of wealth in the community. If agent i has wealth w_i on day t, then its probability of buying products on day t is p_i.

the one for which the agent currently has the highest preference. To do this, we compute the probabilities over the potential sellers as a fraction of the prefer score (b_i, s_{pot}) and the sum of all the prefer scores. With these probabilities, we perform a random selection of which seller s_j is chosen.

Payment of the Product

When an agent in the buyer role b_i purchases a product from an agent in the seller role s_j, a monetary transaction occurs. The buyer always prefers to pay with the Encointer wallet, since the balance decreases over time to demurrage and thus saving this currency makes no sense. If the full sales price of the product cannot be withdrawn from the Encointer wallet, the buyer pays with his local currency. Note that the product prices are in the Encointer currency, so when paying with the national currency, the exchange rate must be taken into account.

3.5 Demurrage

Demurrage causes the local currency to lose value over time. This prevents the incentive to hoard money and results in a sustainable redistribution of unused money. At each time step, the demurrage fee is deducted from the agent's account balance, burning a percentage of the agent's account balance each day.

In Encointer, the constant demurrage value is calculated from the half-life

$T_{0.5}$ of the money: This means, the money in the accounts of agents should half its value on day $T_{0.5}$. To find the demurrage, consider first an exponential decay.

Exponential Decay and Half-Life $\mathbf{T_{0.5}}$

An exponential decay occurs when a quantity decreases at a rate proportional to its current value. This process can be expressed by the equation

$$q(t) = Q_o * e^{-\lambda * t}, \tag{3.16}$$

where Q_o is the initial quantity at time $t = 0$ and λ is a positive rate, called the exponential decay constant. The time required for a quantity to fall to half its initial value Q_o is called the half-life $T_{0.5}$ and can be computed as

$$\frac{Q_o}{2} = q(t = T_{0.5}). \tag{3.17}$$

The direct relationship between the decay constant λ and the half-life $T_{0.5}$ becomes clear when the 3.16 and 3.17 are resolved into

$$\lambda = \frac{-\ln(0.5)}{T_{0.5}}. \tag{3.18}$$

Inserting λ back into 3.16 gives the exponential decay

$$q(t) = Q_o * e^{\frac{ln(0.5)}{T_{0.5}} * t}. \tag{3.19}$$

Demurrage Rate and Demurrage Fee

We want to find the demurrage rate for each time step. The exponential decay for the next time step corresponds to

$$q(t+1) = Q_o * e^{\frac{ln(0.5)}{T_{0.5}} * (t+1)}. \tag{3.20}$$

Dividing 3.20 by 3.19 gives the constant demurrage rate

$$demurrage\_rate = e^{\frac{ln(0.5)}{T_{0.5}}}, \tag{3.21}$$

by which the accounts are multiplied at each time step. This results in a decay of the accounts over time. In our simulation, we define demurrage as

$$demurrage = 1 - demurrage\_rate. \tag{3.22}$$

Then the demurrage fee for agent v_i in time step t is

$$demurrage\_fee = v_i.account * demurrage. \tag{3.23}$$

which will be subtracted from the current account balance.

3.6 Product Manufacture

The amount of products manufactured by each agent depends on its production speed. At the end of the day, the production speed is added to the product quantity of the agent. The amount of products increases linearly over time if none are sold. However, there is a maximum inventory level that cannot be exceeded.

3.7 Preference Change

An agent's preferences for all available products are individual and dynamic. In the product purchasing substep, each agent's prefer scores are reduced depending on which items were purchased during the day. This would cause the prefer scores drop to zero over time. To counteract this, the prefer scores are increased again with each time step. The idea behind this is that as time progresses, the agent will want to consume this product again. However, this preference increase must be set much lower than the decrease, especially when there are many products to consume and purchasing power is low.

3.8 Inequality Measure

In order to analyze our simulation results and display the inequality in the community, we have to measure the Gini index at each time step.

First, we have to find the Lorenz curve for wealth distribution, which was inspired from [17]. Second, the area under the curve needs to be calculated. Finally, the Gini index can be computed as the measure of inequality in our community.

Lorenz Curve of the Discrete Wealth Distribution

Every agent i in the Encointer economy has an account balance representing their wealth w_i, which we sort in ascending order:

$$0 < w_1 < w_2 < ...w_N. \tag{3.24}$$

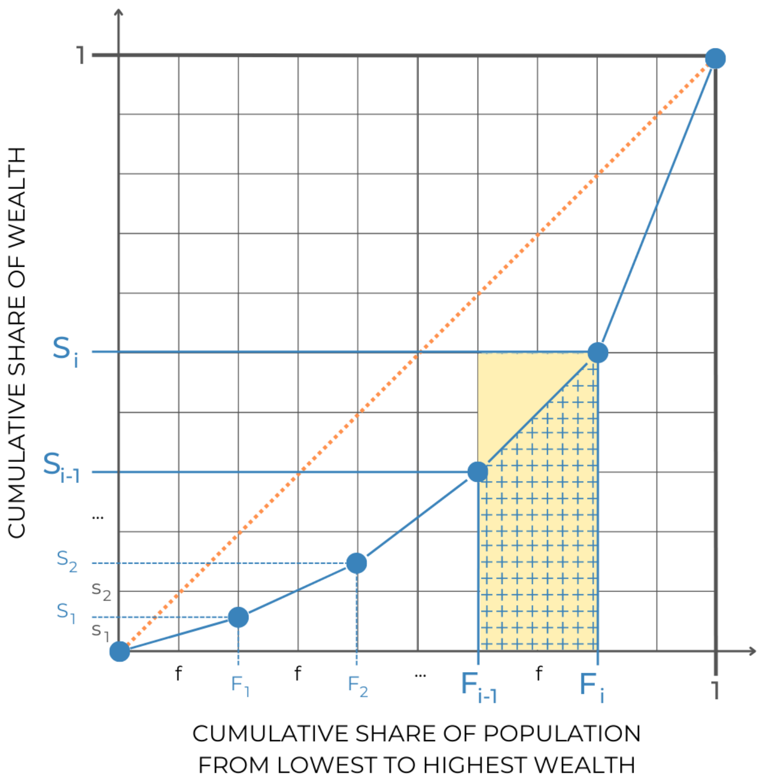

Figure 3.5: Computation of the Gini index in a recursive fashion. The sum of the trapezoids gives the area under the Lorenz curve.

The total wealth of the economy is

$$W = \sum_{i=1}^{N} w_i. \tag{3.25}$$

To compare the distribution of wealth over time, we normalize the wealth of each agent w_i by dividing it by the total wealth W, which gives us the share

$$s_i = \frac{w_i}{W}. \tag{3.26}$$

The cumulative portion of the total wealth can be found using recursion:

$$S_i = \begin{cases} S_{i-1} + s_i, & \text{if } 1 \leq i \leq N \\ 0, & \text{if } i = 0. \end{cases} \tag{3.27}$$

Thus, $S_i = s_1 + s_2 + ... + s_{i-1} + s_i$ and $S_N = \sum_{i=1}^{N} s_i = 1$.

Next, each agent represents a fraction

$$f = \frac{1}{N}, \tag{3.28}$$

of the total population N. Since $\sum_{i=1}^{N} f = 1$ holds, f is a normalized value. Similar to 3.27, the cumulative portion of the population can be described in a recursive way:

$$F_i = \begin{cases} F_{i-1} + f, & \text{if } 1 \leq i \leq N \\ 0, & \text{if } i = 0. \end{cases} \tag{3.29}$$

Again, $F_N = \sum_{i=1}^{N} f = 1$.

Then, let us represent the cumulative distribution of the population F on the horizontal axis, and the cumulative distribution of shares S on the vertical axis. The Lorenz curve is the linear interpolation of the points

$$(F_0, S_0), (F_1, S_1), (F_2, S_2), ..., (F_N, S_N). \tag{3.30}$$

Note that the curve always begin at the origin $(F_0, S_0) = (0, 0)$, ends in $(F_N, S_N) = (1, 1)$ and consists of N line segments.

Area under the Lorenz Curve

Next, we have to calculate the area under the Lorenz curve. Using a recursive approach, we can find the Lorenz area until step i with

$$A_{lorenz,i} = \begin{cases} A_{lorenz,i-1} + A_{trapeze,i} & \text{if } 1 \leq i \leq N \\ 0, & \text{if } i = 0, \end{cases} \tag{3.31}$$

whereas the trapeze area $A_{trapeze,i}$ is derived as follows:

$$A_{rectangle,i} = S_i * f, \tag{3.32}$$

$$A_{triangle_i} = \frac{(S_i - S_{i-1}) * f}{2}, \tag{3.33}$$

$$A_{trapeze,i} = A_{rectangle,i} - A_{triangle_i}. \tag{3.34}$$

Summing up all trapezes gives us the total area under the Lorenz curve

$$A_{lorenz} = \sum_{i=1}^{N} A_{trapeze,i}. \tag{3.35}$$

Gini Index

Finally, we can insert $B = A_{lorenz}$ into the Gini index described in 2.2:

$$Gini\ index = \frac{0.5 - A_{lorenz}}{0.5}. \tag{3.36}$$

Chapter 4

Results and Discussions

According to the National Commonwealth Group, the combination of basic income and complementary currency "is best applied where there is very little national currency in circulation. Thus, it currently fits developing economies."[6]

Throughout our simulations we focus on developing countries with low income, i.e. citizens with little national currency in their savings. Different scenarios were run over a five-year period using our Encointer simulator. We were inspired by the example presented by the Encointer association in their explanatory video.[18] All results and plots presented in this chapter were obtained using our simulator. The parameters used for the simulations can be found in the Appendix B.2. Our simulator also returns a large table with all computed variables for each simulation. We obtain all the exact numbers from these tables, which will not be displayed here.

4.1 The OGU Community

Consider a fictitious community of 50 people in Ouagadougou, Burkina Faso, where all members have very little money in their bank accounts, i.e. between 500 and 1500 CFA-Francs. To give a real indication of the value of the currency, 1.5 liters of water is worth 500 CFA-Francs while 500 grams of bread costs 676.44 CFA-Francs, according to Numbeo.[19] To keep our numbers close to reality, we set the cost of a product in our community between 500 and 1000 CFA-Francs. Further, we assume that each person sells exactly one type of product, e.g. bread, and can purchase up to five products a day.

Aminata, a papaya vendor in the community, is mostly unable to meet her basic needs because she does not have sufficient money. Like the other members, she does not have much savings in her bank account and her income depends only on how many papayas she sells per day. She notes that she is not the only one struggling. On average, a community member buys only one product per day. This can be seen Figure 4.1 generated by our simulator when only the state-issued currency is used.

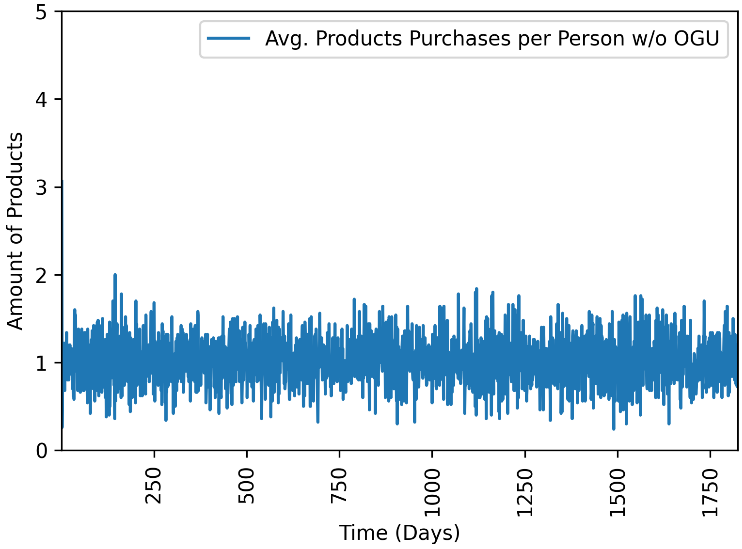

Figure 4.1: An average of one product per day is bought by a person when only state-issued currency is used.

Now Aminata has an idea how to counteract this and launches together with four of her friends the "OGU" - a new Encointer currency. They agree to set the basic income issued per person during a ceremony at 50 OGUs, while the exchange rate for an OGU token is 100 CFA-Francs. Observe that 1.5 liters of water would be priced at 5 OGUs. They decide that the nominal value of the OGU should halve within one year to prevent hoarding. After a fruitful bootstrapping session, the five community members receive their first OGUs. From then on, they decide to gather for a ceremony every ten days to collect further basic income.

Word spreads quickly, and since all the members of our fictional community live on little money, all want to collect a basic income. The only way to become part of the OGU community is to attend the upcoming ceremony. Aminata has to explain that only 1/3 of the participants in the ceremony can be accepted as new members to ensure security of the protocol. In Figure 4.2 observe that the community growth follows a logistic curve and that on day 450 the last member was able to join the community.

In an Encointer system, the size of the community goes hand in hand with the amount of money in circulation. The more people participate in the ceremony, the more OGU tokens are issued. At the same time, the supply of OGUs decreases over time due to demurrage. After the community size has stabilized, at some point the cumulative demurrage payment exceeds the total OGU tokens in circulation. Our table shows that this occurs on day 1095. The quantity of OGU tokens begins to stabilize at about 36,500 pieces, which can be seen in Figure 4.3. Further, we observe that the total amount of OGUs minted during a

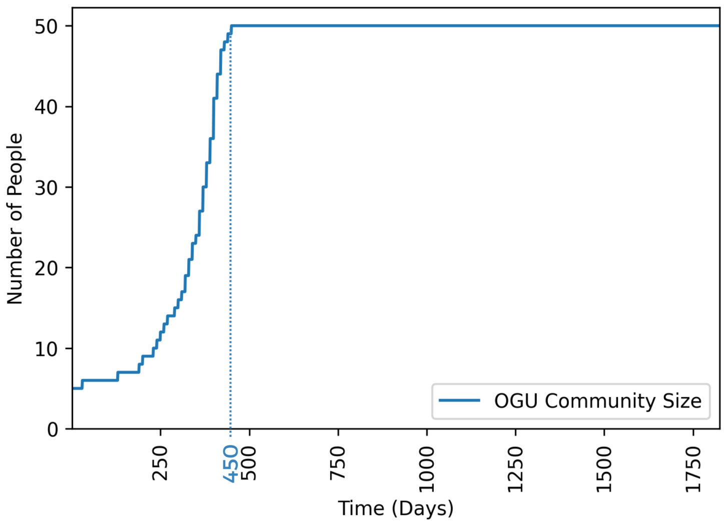

Figure 4.2: OGU community reaches the maximum population on day 450.

ceremony and the demurrage paid between two successive ceremonies level off in Figure 4.4.

Aminata and her community are happy because their new OGU currency has quadrupled their daily purchasing power, see Figure 4.5. Furthermore, we see in Figure 4.6 that all product purchases were made with the OGU currency, allowing the community to keep its national currency for savings. In this way, a stable market is created for the community in Ouagadougou.

Lastly, although the community members had only little national currency, we observe in Figure 4.7, this state-issued money was unequally distributed among individuals. With Aminata's initiative to introduce the OGU, the community is moving towards equality.

4.1.1 Community without National Currency

In our setting, every community member in Ouagadougou was handed some money in their bank account to get things rolling. However, as we mentioned in the introduction, over one billion people have no bank account due to the lack of state-issued ID. Encointer allows citizens to receive basic income without proof of identity. Therefore, our simulator supports the scenario in which citizens do not have any state-issued money.

Assume now that every member of the community has a product to sell, but is completely broke to buy anything. OGU is the only currency that comes into circulation after Aminata and her friends launch the Encointer community. The likelihood that community members will attend the ceremony regularly is higher

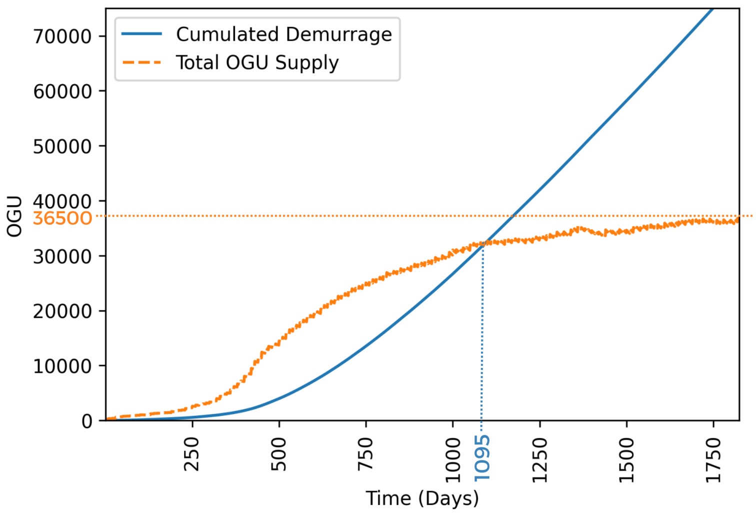

Figure 4.3: OGU supply saturates at about 36,500 tokens and the cumulated demurrage paid exceeds the OGU supply on day 1095.

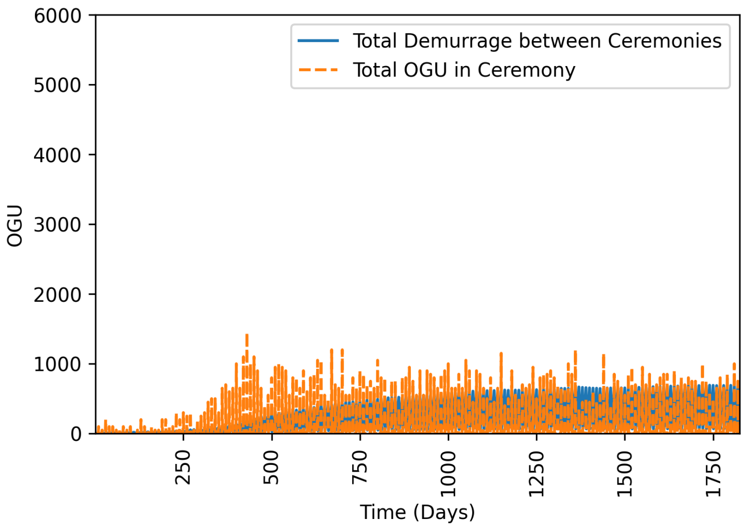

Figure 4.4: OGU tokens minted during single ceremonies and demurrage paid between two consecutive ceremonies balance each other out over time.

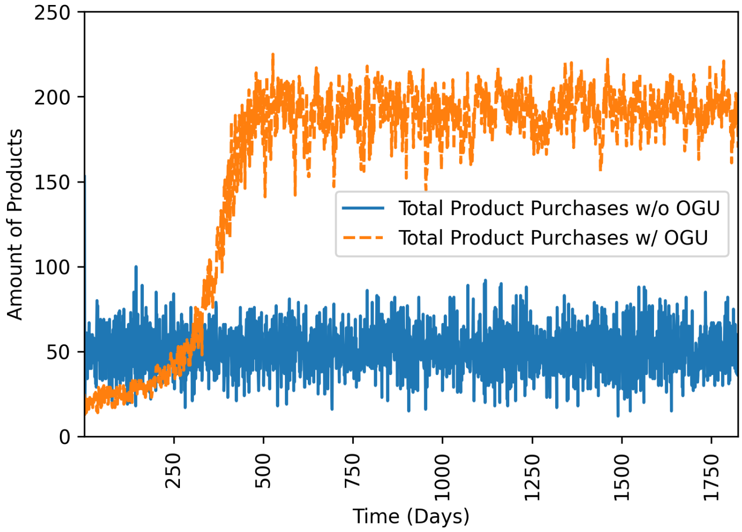

Figure 4.5: Daily purchasing power of the community is quadrupled when the OGU is launched.

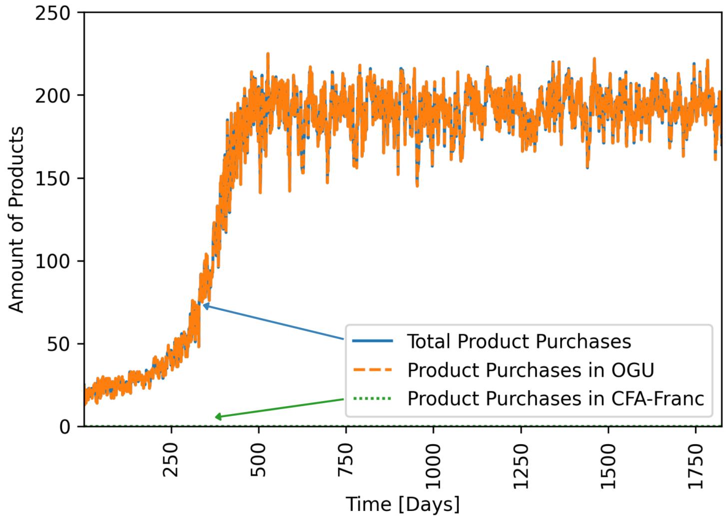

Figure 4.6: When the OGU is launched, products are only purchased with OGU, allowing to keep state-issued currency (CFA-Franc) for savings.

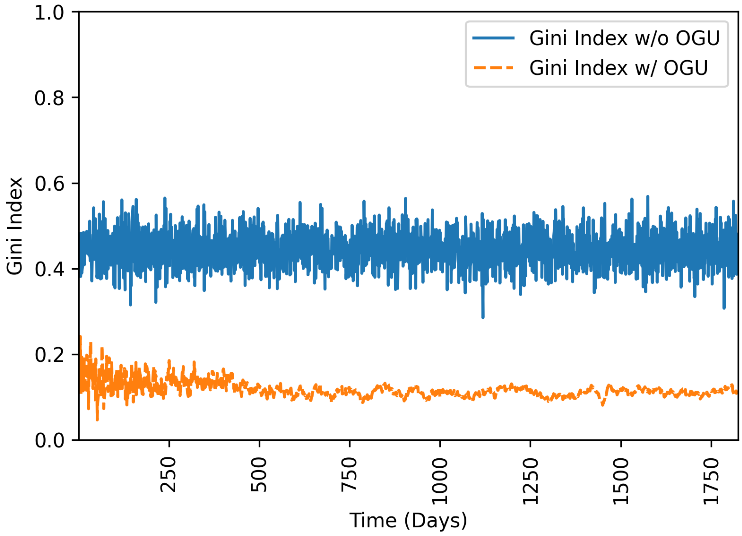

Figure 4.7: Gini index close to perfect equality for the case when OGU is introduced, while stable Gini index without the OGU currency.

than the scenario before. Thus, the citizens can join the OGU community faster. In Figure 4.8 can be observed that the OGU community reaches its maximum on day 300. With this, the cumulative demurrage fee exceeds the total OGU tokens in circulation earlier, according to our table on day 936, and the token supply saturates at around 34,000 pieces. Both can be seen in Figure 4.9.

With the basic income, Aminata can start buying products from her community members. In Figure 4.10, we see that the amount of daily products purchased reaches around 200 products per day, which means in average a person buys four products.

Notice that a population in which every citizen has zero money is, by definition, perfectly equal. With the introduction of the OGU currency, the Gini index increases at first, since initially only part of the community can participate in the ceremonies and receive their money. After everybody joined the OGU community, the Gini index decreases again until it approaches equality. This can be observed in Figure 4.11.

Our simulator shows that Encointer improves the welfare of citizens even if state-issued money is completely removed from the equation. Again, the community reaches a stable micro-economy.

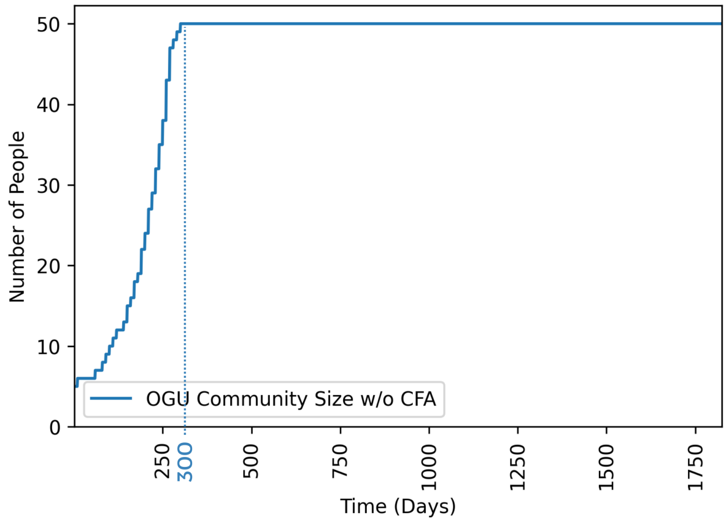

Figure 4.8: When there is no national currency (CFA) in circulation, the maximum OGU community size is reached on day 300.

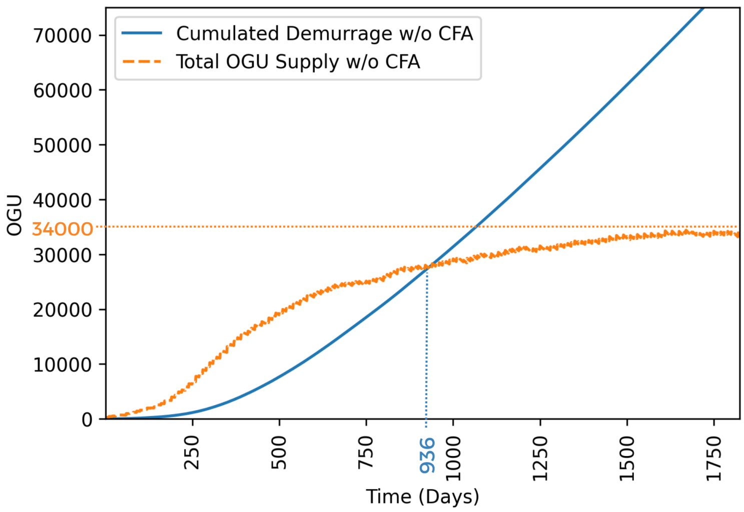

Figure 4.9: Without national currency (CFA), the OGU supply saturates at approx. 34,000 pieces and the cumulated demurrage exceeds the total OGU tokens in circulation on day 936.

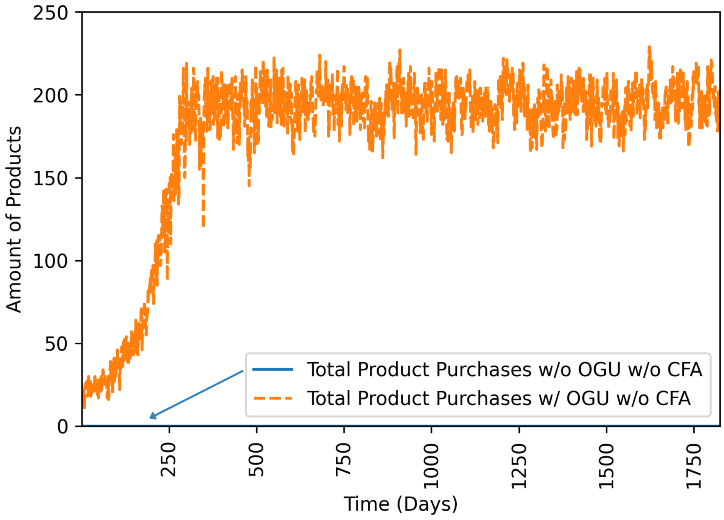

Figure 4.10: No purchasing power for a community without national currency (CFA), while the purchasing power is approx. 200 product per day when the OGU is launched.

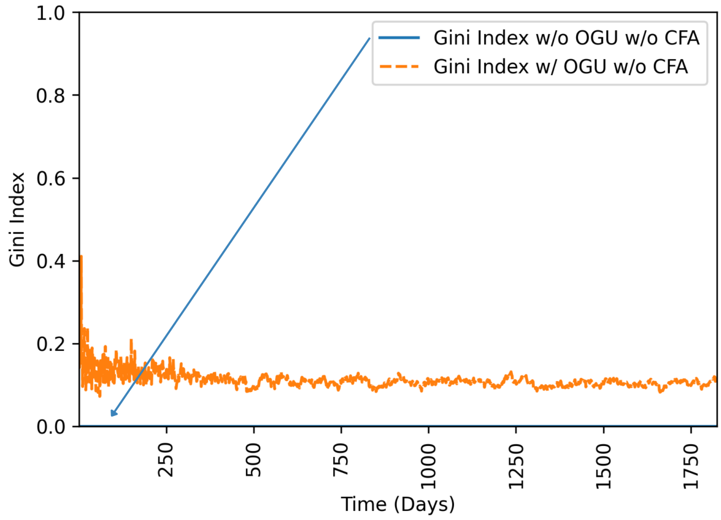

Figure 4.11: Perfect equality for a community with no national currency (CFA) and Gini index close to equality if the OGU is introduced.

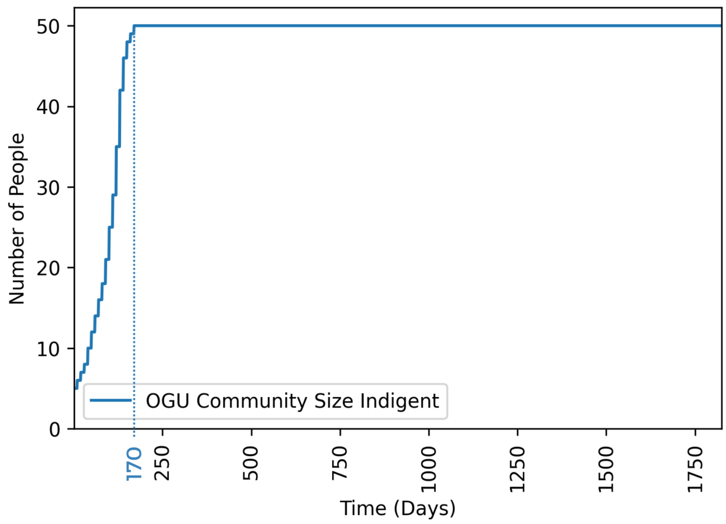

Figure 4.12: When 50% of community is indigent (no CFA and no products), the OGU community size reaches its maximum on day 170.

4.1.2 Community with Indigent Members

Next, suppose that some community members do not have any products for sale. They live only for consumption. Since in our scenario, income is proportional to products sold, these people have no visible means of income. We call them the "indigents". Without a revolution in the system, these people have no chance of meeting their basic needs.

Consider that half of our fictitious community in Ouagadougou is indigent. Again, Aminata launches the OGU community with four of her friends in the same way as before. Since indigent members do not have any national currency and cannot earn it by selling products, they are most likely to go to every ceremony. The time period for joining the OGU community has further shortened. This results in Figure 4.12 that the last person joins the OGU community already on day 170. Also, in Figure 4.13, it can be seen that the token supply is saturated at about 72,000, which is double compared to Figure 4.3 and Figure 4.9. However, on day 902 the cumulative demurrage fee exceeds the total OGU tokens in circulation which is similar to both scenarios before.

When 50% of our population do not have state-issued money or products to sell, the introduction of the OGU not only results in more products being purchased per day, but also means that those in need (the indigents) can suddenly afford products. A comparison without and with the introduction of the OGU in terms of purchasing power can be seen in Figure 4.14.

The Gini indices in this scenario, however, are quite high, since we have a

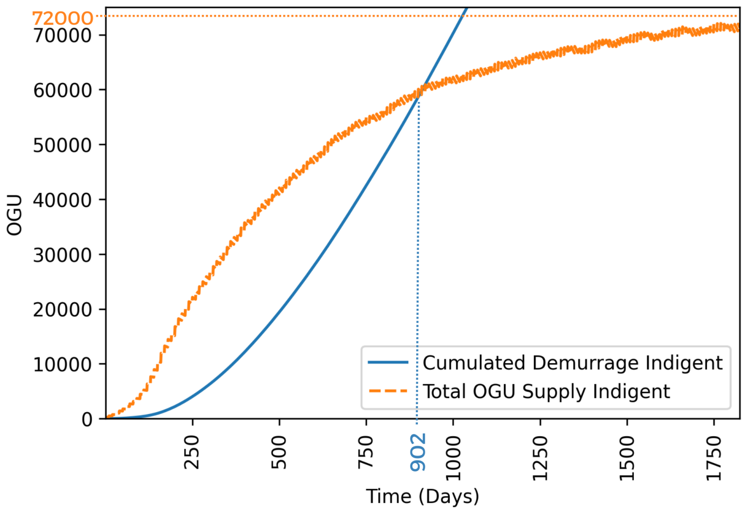

Figure 4.13: A significant increase of the token supply results, when 50% of the community are indigent (no CFA and no products).

considerable wealth difference between the indigents and the other community members. This can be seen in Figure 4.15.

4.1.3 Improving Welfare for Indigent Members

Aminata is fortunate because she has a papaya stand. But other community members who do not sell products are unhappy and are still facing starvation. The basic income of 50 OGUs is not enough for them to cover their needs. Aminata convinces her friends to increase the basic income to 250 OGUs, set the demurrage half-time to two years and shorten interval between ceremonies to five days. With this new framework, they launch the community.

The OGU community grows faster due to the shortened ceremony interval and reaches its maximum size on day 100, see Figure 4.16.

That the community members now receive in average five times more OGUs per ceremony is revealed when comparing Figure 4.17 with the Figure 4.4. The total supply of OGU tokens saturates at 750,000, which is demonstrated in Figure 4.18.

Every community member has more OGUs to purchase products. This not only results in a higher local economy, but that the 50% indigents carry also out 50% of the product purchases, which is evident in Figure 4.19.

Finally, when the OGU is introduced with the new framework proposed by Aminata, the Gini index in Figure 4.20 is close to equality.

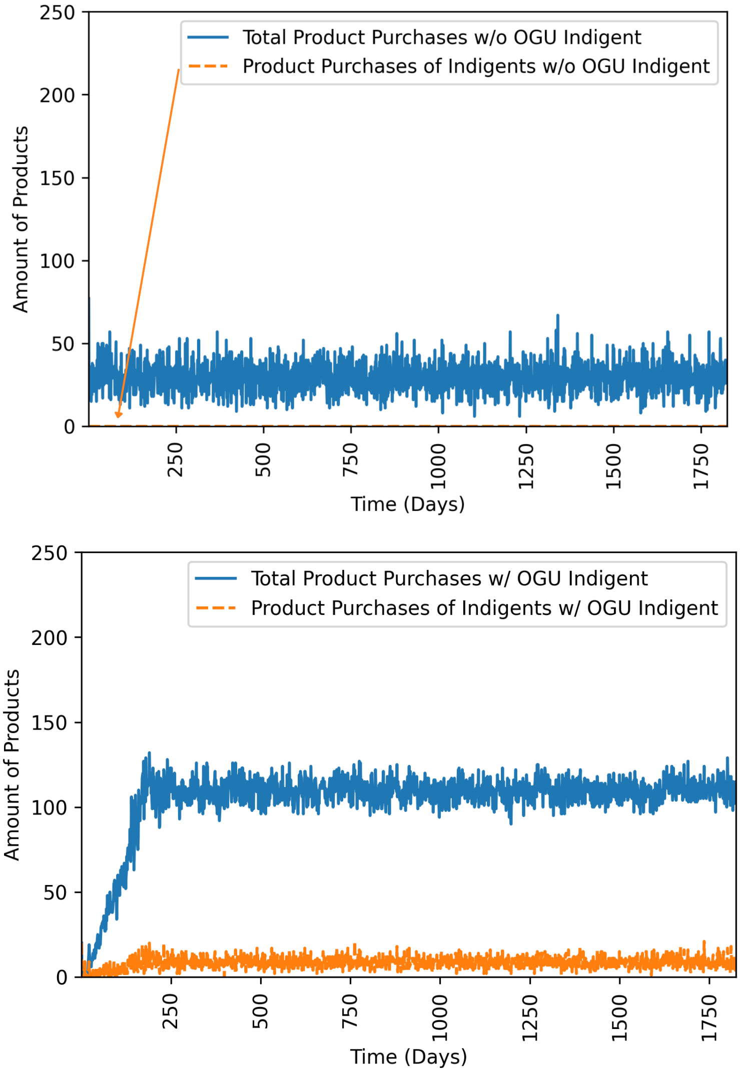

Figure 4.14: Comparison of daily product purchases when 50% of the community is indigent (no products and no savings), for the scenario with and without the introduction of the OGU.

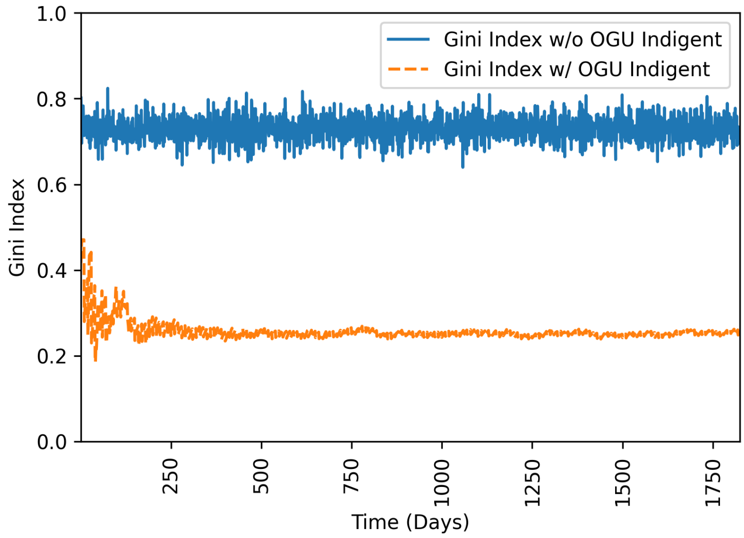

Figure 4.15: A rather large inequality within the community when 50% of people are indigent (no products and no savings) for the case without OGU, as well as with OGU.

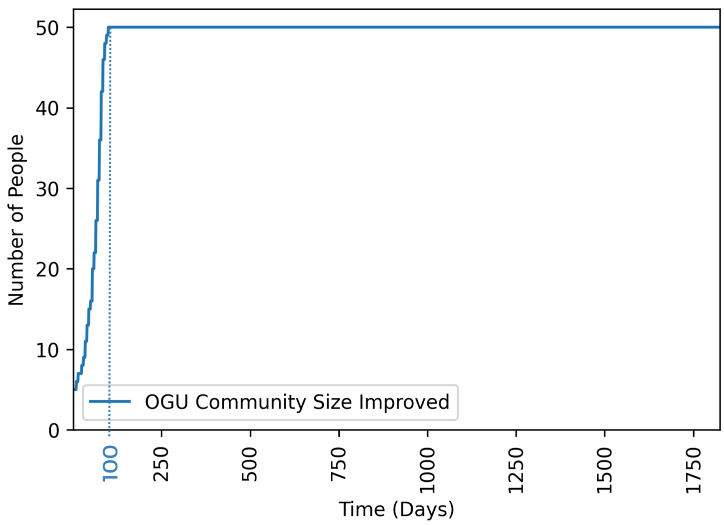

Figure 4.16: The halving of ceremony interval leads to a faster community growth, reaching its maximum size on day 100.

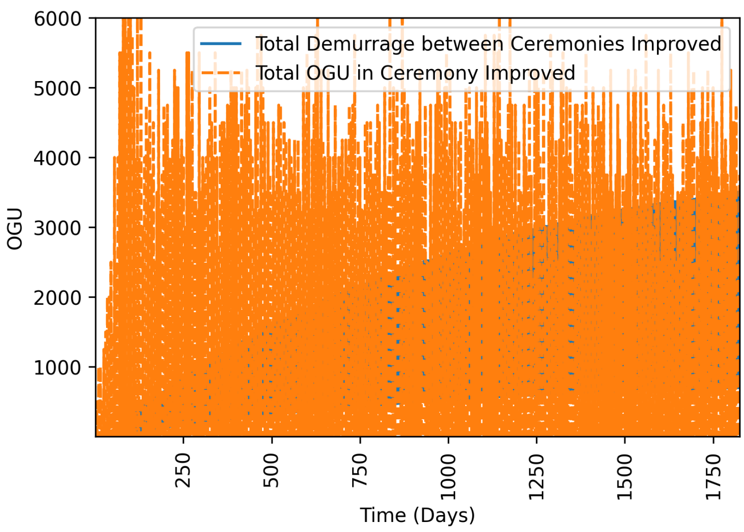

Figure 4.17: Increasing the base income by a factor of five also increases the supply of OGU tokens per ceremony by a factor of five. Shortening the ceremony interval results in a more concentrated discrete curve.

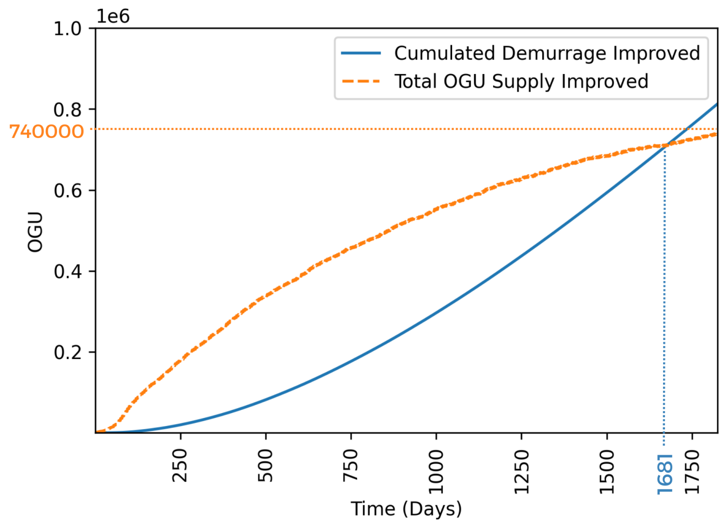

Figure 4.18: Increase of the basic income results that the OGU supply saturates at 740,000 tokens. Cumulated demurrage meets the total OGU in circulation on day 1,681. Note that the scale of the y-axis is changed.

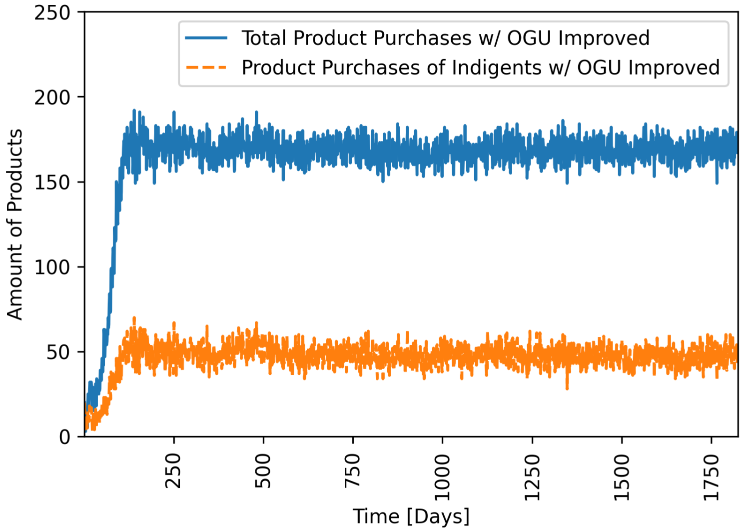

Figure 4.19: Increasing the basic income, shortening the ceremony interval or lowering the demurrage fee, does not only lead to a higher purchasing power, but also lets the 50% lower class population purchase 50% of the goods.

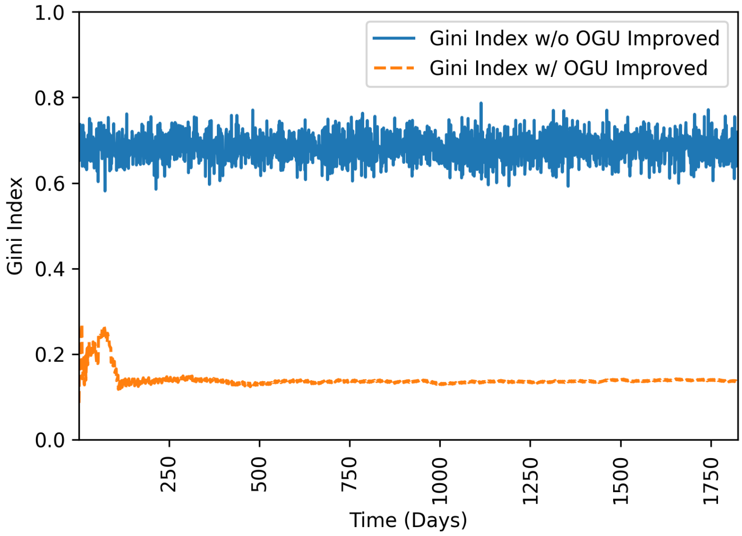

Figure 4.20: Increasing the basic income, shortening the ceremony interval or lowering the demurrage fee, brings our community closer to equality when OGU is introduced.

Increasing the basic income, lowering the ceremony interval or decreasing the demurrage fee (by increasing the demurrage half-time) leads to an increase of purchasing power of the initial lower class population. This results in an overall shift towards more equality in the population.

4.2 Limitations

4.2.1 Consumer Buying Behavior and Community Growth

Our simulator uses S-curves to decide who buys a product and who attends a ceremony. The probability with which an agent decides is based on the wealth of that person, but also on the money the richest citizen has. In our example community of Ouagadougou, this approach works well because we consider community members with similar settings. However, if we look at a city like Zurich, where the wage band is very wide, this can distort our simulation. For example, suppose that Rich is the richest person in Zurich. In our simulator, Rich is much more likely to shop every day than the middle and lower classes. The decision of who Rich buys from depends only on who he has not purchased from in the last few days. Therefore, money flows from the upper class to the lower class, and equality is created without introducing an Encointer currency. In reality, however, this is not the case. Analyzing consumer buying behavior is complex and beyond the scope of this paper. Our model, nevertheless, shows meaningful results for low-income communities.

Another constraint that arises with the use of S-curves is when all bootstrappers have exactly the same amount of money. Since the probability of participating in a ceremony is very low if one is the agent with the highest wealth, all bootstrappers have almost zero probability of attending the ceremony. If the number of bootstrappers is small, it requires a lot of fortune for enough bootstrappers to participate in the ceremony so that the newbie ratio is satisfied. This can cause to no one joining the community. To simulate the case where all agents have no national currency, an additional line of code was written to give them a high probability of attending the ceremony. We believe that in the other cases it is not very likely that all agents have the same amount of money in their accounts. Nevertheless, this must be kept in mind.

4.2.2 RAM Memory

One issue that became apparent towards the end of this work is that the cadCAD simulation engine requires an excessive amount of RAM memory. This is because cadCAD stores every variable for each substep in the RAM memory, so that the size of the occupied space increases linearly with the number of time steps. Thus, even with a cloud-based environment, it was not possible to run the simulation for large time steps and many Monte Carlo runs simultaneously. However, we found that the Monte Carlo runs produced similar results for small time steps.

Chapter 5

Conclusion and Future Work

5.1 Conclusion

Looking at developing countries with low income and prosperity, our simulator showed that community currencies with basic income and demurrage have a lasting positive impact. For the scenario where there is little or no government-issued money in circulation, the Encointer framework increased the daily and general purchasing power of citizens, leading to a higher local economy. Moreover, the given maximum purchasing power was fully covered by the community currency, leaving the national currency available for saving. As the community functioned without the use of government-issued currency, a stable micro-economy was created. Even, when the little national currencies in circulation were unequally distributed among members at first, the introduction of the community currency has led a more balanced ecosystem.

It was also observed that the increase the basic income or the shortening of the intervals between successive basic income payments not only led to an increase in purchasing power overall, but also gave the lowest stratum of society the opportunity to satisfy their basic needs. This led to an overall shift toward more equality in the population.

Our simulator, however, is limited with basic parameters and should be enriched to represent more realistic world economy. This is primarily because people's purchasing behavior in developed countries, like Switzerland, is not determined by their basic needs, and their decision to join an Encointer community is influenced by many more factors and drivers than their current financial situation.

5.2 Future Work

The big challenge next is to realistically implement people's buying behavior and the motivation to join an Encointer community. Since the first Encointer communities are being launched, a survey could be carried out in order explore the drivers of customers. Then, the results could be implemented into our framework.

Due to time limitations of this thesis some parts had to be omitted in the simulation, like letting agents have more types of products or distinguishing between services and products. Also agents could be given designated roles, such as a vendor. Furthermore, in a real Encointer community, people can buy the community currency to become part of the community instead of signing up for a ceremony. However, the experiment going on in Zurich has shown that this does not happen often. In the ceremony, there is usually a small transaction fee which was omitted. Moreover, we assumed our sales prices do not change over time, however, in the real Encointer protocol, everyone is free to set prices at their own discretion. Product manufacturing can be implemented dynamic. The list goes on and on.

Last but not least, as mentioned in Chapter 4, it is necessary to overcome the issue with RAM memory. One possibility is to replace the cadCAD simulator with a self-written simulator that writes each time step to the hard disk. Thus, the space used in the RAM memory would be independent of the number of simulation steps and the total capacity would not be exceeded.

Bibliography

[1] A. Thériault and J. Tenwick, "Terrifying prospect of over a quarter of a billion more people crashing into extreme levels of poverty and suffering this year," 4 2022. [Online]. Available: https://www.oxfam.org/en/press-releases/terrifying-prospect-over-quarter-billion-more-people-crashing-extreme-levels-poverty

[2] M. Sauvante and V. Macneil, "A Model Complementary Currency Used to Reduce Poverty & Help Build Healthy Local Economies," *National Commonwealth Group*. [Online]. Available: https://nationalcommonwealthgroup.net/doc/UNOwhite.paper.pdf

[3] A. Brenzikofer, "Towards a Decentralized Global Universal Basic Income," 4 2020. [Online]. Available: https://medium.com/@funny_mango_mouse_595/towards-a-decentralized-global-universal-basic-income-995651b2ceb6

[4] C. M. Macal and M. J. North, "Agent-based modeling and simulation," *Proceedings - Winter Simulation Conference*, pp. 86–98, 2009. [Online]. Available: https://www.researchgate.net/publication/216813135_Agent-based_modeling_and_simulation

[5] D. Helbing, "Agent-based modeling," *Understanding Complex Systems*, pp. 25–70, 2012. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-642-24004-1_2

[6] M. Sauvante and V. Macneil, "The UNO A Model Complementary Currency Used to Reduce Poverty & Help Build Healthy Local Economiesc)(3) non-profit corporation." [Online]. Available: https://commonwealthgroup.net/doc/CommonwealthDevelopmentOrganizations.pdf

[7] K. Hart, J.-L. Laville, and A. D. Cattani, "Community and complementary currencies," *The Human Economy*, pp. 303–312, 2010. [Online]. Available: https://www.researchgate.net/publication/51014991_Community_and_complementary_currencies

[8] "BIEN | About Basic Income," 2021. [Online]. Available: https://basicincome.org/about-basic-income/

[9] H. Godschalk, "Does Demurrage matter for Complementary Currencies?" *International Journal of Community Currency Research*, vol. 16, pp. 58–69, 2012. [Online]. Available: https://www.academia.edu/29711843/Does_Demurrage_matter_for_Complementary_Currencies

[10] "Gini Coefficient - Definition, Principles and Limitations," 2 2021. [Online]. Available: https://corporatefinanceinstitute.com/resources/knowledge/economics/gini-coefficient/

[11] T. Sitthiyot and K. Holasut, "A simple method for measuring inequality," *Palgrave Communications*, vol. 6, no. 1, 12 2020.

[12] "Economics Gini coefficient." [Online]. Available: https://en.wikipedia.org/wiki/File:Economics_Gini_coefficient.svg#/media/File:Economics_Gini_coefficient2.svg

[13] G. Pyatt, "On the Interpretation and Disaggregation of Gini Coefficients," *The Economic Journal*, vol. 86, no. 342, p. 243, 6 1976.

[14] "Logistische Funktion." [Online]. Available: https://de.wikipedia.org/wiki/Logistische_Funktion

[15] M. Zargham and J. Gabler, "cadCAD: Filling a Critical Gap in Open-Source Data Science," 2019. [Online]. Available: https://medium.com/block-science/cadcad-filling-a-critical-gap-in-open-source-data-science-fcd0d3faa8ed

[16] R. Hipkins and C. Bronwen, "The Sigmoid Curve as a Metaphor for Growth and Change." [Online]. Available: https://files.eric.ed.gov/fulltext/EJ1123357.pdf

[17] R. Torres, "The Discrete Lorenz Curve." [Online]. Available: http://ciep.itam.mx/~rtorres/microapl/lorenz.pdf

[18] "Encointer - how to promote your local economy with a digital basic income." [Online]. Available: https://www.youtube.com/watch?v=80SdkjtHcr4

[19] "Cost of Living in Ouagadougou," 2022. [Online]. Available: https://www.numbeo.com/cost-of-living/in/Ouagadougou-Burkina-Faso

[20] "cadCAD Bootcamp (Beginner Level)." [Online]. Available: https://www.cadcad.education/course/bootcamp

Appendix A

cadCAD Structure

The standard structure of cadCAD consists of a modeling and a simulation section and is composed as follows:

Modeling

1. **State Variables:** Sufficiently describe the state of a system and change over time
2. **System Parameters:** Impact and control the behavior of the system
3. **Policy Functions:** Describe the logic or behavior of a component of a system
4. **State Update Functions:** Responsible for updating a single state variable
5. **Partial State Update Blocks (PSUB):** A series of PSUBs is a structure for serial or parallel compilation of State Update Functions and Policy Functions as a representation of the system model. A single PSUB represents a substep, while all PSUBs form a time step.

Simulation

6. **Configuration:** Select the number of times the simulation is run 'N' (also called "Monte-Carlo runs"), the number of time steps the simulation will run for 'T' and the system parameters 'M'
7. **Execution:** Take the model and configuration, create and runs the simulation object, and store the raw results in as a list of Python dictionaries
8. **Output Preparation:** Convert the raw data to analyze the results

The structure of cadcad is explained in detail in a certified beginner's course offered by cadCAD Edu.[20] This course is recommended for anyone new to the cadCAD engine.

Appendix B

Source Code

B.1 GitHub Repository

The source code of our Encointer simulator can be found on the following GitHub Repository:

https://github.com/dwelti/encointer_simulations

B.2 Simulation Parameters

The following four pages contain our simulation parameters used in Chapter 4.

Key	Value
SIMULATION_TIMESTEPS	1825
demurrage	demurrage(365), 0
ceremony_interval	10, SIMULATION_TIMESTEPS+1
basic_income	50, 0
initial_population	5, 50
max_population	50, 50
lower_class_national_currency	(500, 1500)
middle_class_national_currency	(0, 0)
upper_class_national_currency	(0, 0)
lower_middle_boundary	0
middle_upper_boundary	0
percentage_of_classes	(1, 0, 0)
newbie_ratio	1/3
max_nr_of_purchases_per_person	5
preference_change_big	10
preference_change_small	1
max_product_stock	200
exchange_rate	0.01

Table B.1: System parameters for the *OGU Community*

Key	Value
person_product_price	randint(5, 10)
person_product_amount	randint(50, 100)
person_production_speed	randint(10, 20)
person_prefer	randint(1, 10)

Table B.2: Agent parameters for the *OGU Community*

Key	Value
SIMULATION_TIMESTEPS	1825
demurrage	demurrage(365), 0
ceremony_interval	10, SIMULATION_TIMESTEPS+1
basic_income	50, 0
initial_population	5, 50
max_population	50, 50
lower_class_national_currency	(0, 0)
middle_class_national_currency	(0, 0)
upper_class_national_currency	(0, 0)
lower_middle_boundary	0
middle_upper_boundary	0
percentage_of_classes	(1, 0, 0)
newbie_ratio	1/3
max_nr_of_purchases_per_person	5
preference_change_big	10
preference_change_small	1
max_product_stock	200
exchange_rate	0.01

Table B.3: System parameters for the *OGU Community without National Currency*

Key	Value
person_product_price	randint(5, 10)
person_product_amount	randint(50, 100)
person_production_speed	randint(10, 20)
person_prefer	randint(1, 10)

Table B.4: Agent parameters for the *OGU Community without National Currency*

Key	Value
SIMULATION_TIMESTEPS	1825
demurrage	demurrage(365), 0
ceremony_interval	10, SIMULATION_TIMESTEPS+1
basic_income	50, 0
initial_population	5, 50
max_population	50, 50
lower_class_national_currency	(0, 0)
middle_class_national_currency	(500, 1500)
upper_class_national_currency	(0, 0)
lower_middle_boundary	0
middle_upper_boundary	0
percentage_of_classes	(0.5, 0.5, 0)
newbie_ratio	1/3
max_nr_of_purchases_per_person	5
preference_change_big	10
preference_change_small	1
max_product_stock	200
exchange_rate	0.01

Table B.5: System parameters for the *OGU Community with Indigent Members*

Key	Value
person_product_price	0
person_product_amount	0
person_production_speed	0
person_prefer	randint(1, 10)

Table B.6: Parameters for agents with intial status 0 for the *OGU Community with Indigent Members*

Key	Value
person_product_price	randint(5, 10)
person_product_amount	randint(50, 100)
person_production_speed	randint(10, 20)
person_prefer	randint(1, 10)

Table B.7: Parameters for agents with intial status 1 for the *OGU Community with Indigent Members*

Key	Value
SIMULATION_TIMESTEPS	1825
demurrage	demurrage(730), 0
ceremony_interval	5, SIMULATION_TIMESTEPS+1
basic_income	250, 0
initial_population	5, 50
max_population	50, 50
lower_class_national_currency	(0, 0)
middle_class_national_currency	(500, 1500)
upper_class_national_currency	(0, 0)
lower_middle_boundary	0
middle_upper_boundary	0
percentage_of_classes	(0.5, 0.5, 0)
newbie_ratio	1/3
max_nr_of_purchases_per_person	5
preference_change_big	10
preference_change_small	1
max_product_stock	200
exchange_rate	0.01

Table B.8: System parameters for the *OGU Community with Improved Welfare for Indigent Members*

Key	Value
person_product_price	0
person_product_amount	0
person_production_speed	0
person_prefer	randint(1, 10)

Table B.9: Parameters for agents with intial status 0 for the *OGU Community with Improved Welfare for Indigent Members*

Key	Value
person_product_price	randint(5, 10)
person_product_amount	randint(50, 100)
person_production_speed	randint(10, 20)
person_prefer	randint(1, 10)

Table B.10: Parameters for agents with intial status 1 for the *OGU Community with Improved Welfare for Indigent Members*