American Journal of Multidisciplinary Research & Development (AJMRD)
Volume 05, Issue 05 (May - 2023), PP 19-28
ISSN: 2360-821X
www.ajmrd.com

Research Paper **Open Access**

Motivation, Equality and Economy Collapse

Stanislaw Raczynski*
**Universidad Panamericana, Engineering Faculty, Augusto Rodin 498, Mexico City*

ABSTRACT
The egalitarian policies are attractive social ideas that are considered progressive and good social justice tendencies. However, we could observe that the ideologies related to social equality (used mainly as a propagandist issues) have not resulted in the welfare of the societies, and rather terminated with the collapse of economies. Here, we consider an agent-based model of an artificial society, where the individuals are optimizing their behavior patterns. They attempt to achieve better relation between their work and effort, and the expected return from the economy. In the agent-based simulation, the agents are created in the computer memory, and run their activities that obey simple behavioral rules. The inequality in the population is measured using the Gini coefficient. The results show the changes of the Gini measure, and the welfare parameters for different egalitarian modes.

Keywords: Social inequality, equalitarian policy, motivation, welfare

I. INTRODUCTION

Here, we are using an agent-based model to simulate an artificial society, where the individuals contribute to the common economy and get a return that determines the overall welfare. The individuals follow a simple behavior rules that make them look for a good relation between their effort and the returned goods. It should be noted that we do not pretend to prove if the egalitarian policies are "good" or "bad idea". We just show the simulated abstract population. On the other hand, the rules of the behavior of the "micro-models" (simulated individuals) are similar to those of real society members: they just look for the better return, obtained with less effort.

1.1 Agent-based vs System Dynamics models

Modeling and simulation (M&S) of social and economic system can be done, looking for the global properties of the modeled system. These may include the levels of the capital, workforce and welfare, and the flows of money, workforce and other variables. This leads to the *System Dynamics (SD)* models [14]. The SD models are represented by sets of ordinary differential equations or difference equations. This method had a great impact on modeling and simulation for several decades. It has been used to simulate industrial dynamics, urban development, economic and social systems. The easy use of SD and the model simplicity made the modeling and simulation available to great number of people who are not necessarily experts in M&S methodologies. The SD models are easy to create and to use. However, they caused a strange conviction among the users that everything in the real world can be described by the differential equations. These issues are addressed in Raczynski [36]. See also Perez and Dragicevic [33], and Obaidat and Papdimitrou [31].

When the capacity and speed of computing hardware have grown, the other, quite opposite M&S methods gained a considerable popularity. These are object- and agent based models and corresponding software implementations.

The object-oriented models consist in a number of software class declarations that are generic code segments. These declarations are used to create objects in computer memory. These are data structures linked to the methods or functions that use and process the data. In computer simulation, we use the software that, in addition to the object creation, manages the model time. This approach has been implemented to create simulation languages and packages that simulate mass-service systems, queuing models, birth-and-death processes, like GPSS, Arena, ProModel and similar. The common simulation paradigm of these models and software is the *discrete event simulation*. In few words, this consists in executing methods of the objects that may change the state of this and/or of other objects. The events are linked to certain model time instants.

In the agent-based models (ABM), the simulation mechanism is similar. The difference is that in ABM, the objects may be more complicated. They can take decisions, optimize their behavior and reveal some articfial intelligence.

In this kind of M&S, we describe and simulate the individuals. The global behavior of the whole model is the result of the behavior of these individual, "atomic" items, called agents.

More information about the ABM modeling can be found in Bandini et al. [4] or Railsback [35].

As for the AMB software, there are several packages, developed during the last three decades. Perhaps ~~one~~ the first was the SWARM package of the Santa Fe Institute developed in 1994 [37]. Other tools are SOARS (Tanuma et al.[39], MATSim (Bazzan et al.[2]), FLAME [9] [19]), MASS package [38], MASON [27], Cormas [5], Recursive Porous Agent Simulation Toolkit [30], Breve-2.7.2 [22] and Ascape [32]. The BLUESSS package, used in our simulations, is described in section 1.2.

1.2 BLUESSS simulation software

The model was implemented using the package BLUESSS (Blues Simulation System). This is an object-oriented simulation system that can be used to simulate the ABM models.

Shortly speaking, the BLUESSS code consists of a series of process definitions and the initializing code segment. The BLUESSS code is translated into C++ and then compiled and executed. The BLUESSS translator converts the process declarations into the C++ class definitions.

Each BLUESSS process includes a series of event code segments. At the runtime, one or more process instants (agents) are generated, equipped with the necessary parameters and activated. Then, the agents run, executing their events. The event execution is done through the internal clock mechanism that includes the event queue. The agent can modify its state or state of other objects, create or erase other objects and itself. This way, we can generate a population of agent objects that interact with each other and execute events over the model time.

This makes the BLUESSS system a good tool for ABM modeling. Note that the events are coded in C++ and compiled with a C++ compiler. This means that inside the event, we can code anything that is available in C++, from simple arithmetic to sophisticated behavior rules.

1.3 Soft system modeling

The artificial populations and human or animal group behavior are part of the soft system M&S. However, remember that to simulate human behavior is rather a Utopian issue. In models of soft systems, the human decision making is simplified to reflect only the issues of interest. Frequently, the human agents are modeled as rational agents that are people who perform optimal actions. This may reflect, to some extent, the behavior of real individuals. However, this approach can be questioned because the human actions and decisions are sometimes completely irrational.

Let us mention some, perhaps relevant, works on the soft system M&S.

Interesting remarks on organizational psychology models can be found in Crowder [11] and Hughes [21] who discusses the advantages of ABM models. Models of terrorist groups like the Osama Bin Laden organization can be found in [25], by Vetch Corporation and Defiant [12]. The self-organization in soft systems is addressed by Latane and Nowak [24].

Holland [20], Epstein [13], Axelrod [1], Gotts [17], Cioffi-Revilla [8], Bak et al. [3] and Macy [29] discuss various aspects of the "computational sociology". Chatterjee and Seneta [7] and Cohen [10] consider models of the opinion interactions in the populations. Other approach, also related to opinion interactions, called the "BC" model is discussed by Krause [23]

The models of violence in riots, and aggression are discussed by Casilli et al. [6]. The models of extortion can be found in the works of Platas-Lopez et al. [34]. Younger [40] deals with the models of social structures related to food and material storage.

The interactions between agents over a landscape is considered by Lustick [28]. In that paper, the model includes the influence of a small number of "exclusivist" individuals in the population.

II. THE SOCIAL INEQUALITY

Commonly used measures of the social inequality are the *Lorenz curve* and the Gini coefficient.

Max Otto Lorenz [26] proposed a method to express the social inequality, known as the Lorenz curve.

The curve is defined as follows. Let x_i be the income or welfare of the individual or of a group number i. First, we sort the x_i values so that $x_i < x_i + 1$. The value of the Lorenz curve is given as follows.

$$G_i = \frac{\sum_1^i x_i}{\sum_1^N x_i} \quad (1)$$

where N is the total number of individuals or groups. If the groups have income equal to each other (egalitarian society), then the Lorenz curve becomes a straight line, as shown in figure 1.

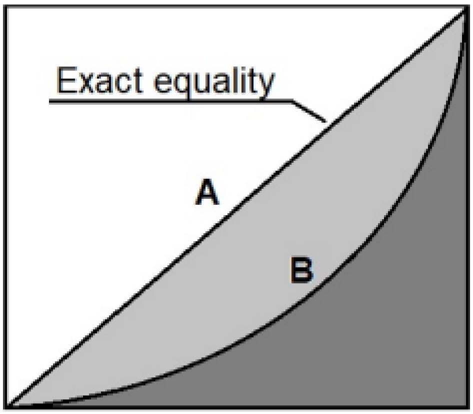

Figure 1: The Lorenz curve

The Gini coefficient proposed by Corrado Gini [16] is another measure of the inequality. It is defined by the following equation.

$$G = \frac{\sum_{i=1}^{N}\sum_{j=1}^{N}|x_i - x_j|}{2N\sum_{j=1}^{N} x_i} \qquad (2)$$

where x_i is the income or welfare of a member number i of a society, and N is the size of the population. Observe that in the perfect egalitarian society the differences $|x_i - x_j|$ are all equal to zero, and the coefficient G=0. If there are differences between members welfare, then the Gini coefficient is greater than zero. In the real societies, the "normal" value of Gini coefficient is more likely about 0.65.

III. MODEL DESCRIPTION

It is a common opinion that more egalitarian society can achieve better economic development. The present article can be treated as a counterexample of this belief. We use the simplest possible model that, however, reflects, to some extent. the behavior of the individuals. Remember that the simplest counterexample is always the best one.

a. Agent objectives and behavior

The model is of agent-based (ABM) type. The society is simulated as a set of members (agents or individuals). We will refer to them as agents. The agents are activated and execute their actions (model events). Each agent has its own objective that may not coincide with the general "common benefit".

The notation and model parameters are as follows

U - Minimal salary
Z - Actual average income in the population
v - Objective function coefficient for the agent
m - Accumulated goods
I - goods increment of the agent per time unit
c - Additional return
g - Return coefficient
W - Amount of work per time unit (default 1)
r - total common goods
N - Population size

If not stated otherwise, the model variables will be expressed in relative units, between zero and one, in relation to maximal or default reference value.

The agents are created in the computer memory, according to a generic agent class (**BLUESSS** process) declaration. Though they have the same data structures and methods, the agent parameters can vary ~~between~~ over agent instances. The main activity of an agent is to work and contribute to the common poll of goods. Agents receive a salary return. The pool of goods grows according to the following equation.

$$\frac{dr}{dt} = \sum_{k=0}^{N}(W_k - I_k) \qquad (3)$$

Here, I_k is the amount of goods received by the agent.

$$I_k = r(g + (W_k - 1)c)/N, \qquad (4)$$

In this model, the income and work variables are understood in a generalized sense. The amount of work is not only the working time, but represents the effective work that includes the agent effort and the quality

of work he/she is doing. We will consider two simulation modes: A, when c=0, and B, when c>0. Looking at eq.(4) we can see that if $c=0$ (mode A), then the agent income is proportional to the amount of common goods r. In case B, with c>0, the income depends also on the agent effort and work effectiveness. Supposing "1" as the reference of W (a "standard" working effort), it can be seen that those who work harder ($W_k > 1$), receive greater income. This is a source of the social inequality in the model.
Variable m is the amount of goods accumulated by the agent. Independently from the received income, this amount decreases with a given constant rate, due to the good consumption.

The objective function that the agent k intents to maximize, is as follows.

$$J_k = v_k I_k - (1 - v_k)W_k, \quad 0 \le v_k \le 1. \tag{5}$$

This is a weighted difference between the income and the effort. Here, $0 \le V_K \le 1$ is the agent parameter, maybe different for different agents. If $V_K = 1$, then the agent looks for maximal income with any effort. If $V_K = 0$ then the agent minimizes the amount of his work, and does not care about the income. As will be explained further on, each agent receives a minimal wage, anyway. The minimal salary U changes, and it is always equal to 0.1Z (average salary). From eq. (4) we obtain (we omit the agent index)

$$J = \frac{vr}{N}(g + (W - 1)c) - (1 - v)W, \tag{6}$$

and after rearranging.

$$J = W\left[v\left(\frac{rc}{N} + 1\right) - 1\right] + \frac{vr}{N}(g - c) \tag{7}$$

Let denote

$$s = v\left(\frac{rc}{N} + 1\right) - 1$$

To maximize the object function, the agent increases the work effort if s>0, and decreases it otherwise. Changing the amount of work W, the agent intents to maximize the object function J. During the simulation, the agent changes the variable W each model time-step, equal to h time units. The change of W, denoted as dW, is as follows.

$$\begin{cases} dW = 0.03s \text{ for } I > U \\ dW = 0.01 \text{ for } I < U \end{cases} \tag{8}$$

It can be seen that if the agent noted that his salary reached the minimal value U, then he/she changes the strategy.

b. Population parameters "v" and "c"

Let us summarize the meaning of the main model parameters. Parameter c is the global property of the model. As a consequence of eqs. (6) and (7), we can see that if c = 0 then the agent objective function is

$$J = W(v - 1) + v\frac{rg}{N},$$

where r is the amount of common goods, and g is the "return coefficient" that tells how much goods are returned to the individual as the remuneration. Recall that $0 \le v \le 1$. This means the agent will always minimize his/her effort to maximize the object function. On the other hand, if c = 1, then we have

$$J = W\left(\left(\frac{r}{N} + 1\right)v - 1\right) + \frac{vr}{N}(g - 1)$$

This means that, depending of v, the agent may increase or decrease his effort. Parameter v ~~is the agent~~ is different for different agents.
Parameter v. This parameter defines how the agent is "predetermined" to work. If v = 0, he/she minimizes the effort, if v = 1, he/she maximizes the effort.

IV. THE IMPLEMENTATION

Below, parameter h is the default time step for execution of most of the events. The agents are implemented in the BLUESSS package as *process agent*. The process has the following events:
Process agent
Init event is used to initiate the agent. Here, the agent parameters are defined, most of them different for different agent instances. The most important is the parameter v, used in the object function as the weight for the income variable. We simulate two cases:
1. $v \in (0.2, 0.8)$, uniformly distributed
2. $v \in (0.5, 1.0)$

Observe that in case 1, v may be less than one. This means that the income may have lower weight than the corresponding effort.

The goods decay rate for the agent is set equal to $S = 0.02(1 + d)$, where $d \in (-0.2, 0.2)$ is a random variable. S defines the decay of the agent accumulated goods, that is, the rate with which the agent consumes its earned goods. Both v and S are different for different agents. The **Init** event activates other events of the agent ~~object~~.

Event **Works.** This event describes the working effort. The agent contributes to the common pool of goods with amount hW. This event is executed repeatedly each h units of model time.

Event **Gets**. Executing this event the agent receives the amount I of goods (see eq.(4). The value of the common pool of goods decreases by the same amount. This event repeats each with time-step h.

Event **Consumes**. The agent decrements (consumes) the amount hS of its own accumulated goods.

Event **Obfunc.** Here, the agent intents to maximize his object function, changing the work variable W, see eq.(8).

Event **Init** is executed only once for each created agent. Each other event re-schedules itself to be executed after h time units. This makes the events execute repeatedly.

Process System

This process has only one instance, created in the initial segment of BLUESSS code. It sets up the global model parameters, ~~and~~ creates 400 agent instances and activates their **Init** events.

V. SIMULATION RESULTS

In the following experiments, we simulate actions of 400 agents, over the model time interval 0-1000, with default time-step for event execution $h = 1$. In all simulations, there is a fixed lower limit for agent's work amount and for the minimal income.

a. Experiment A

In this experiment, the parameters are as follows. $g = 0.2$ (see eq. (4) and (6). Parameter v is different for different agents, randomly (uniformly) distributed in (0.2,0.9). The variables M (accumulated goods) and W (work), at the early stage of simulation are shown in figure 2. W is the amount of work, and M is the agent income. Each vertical line represents one agent.

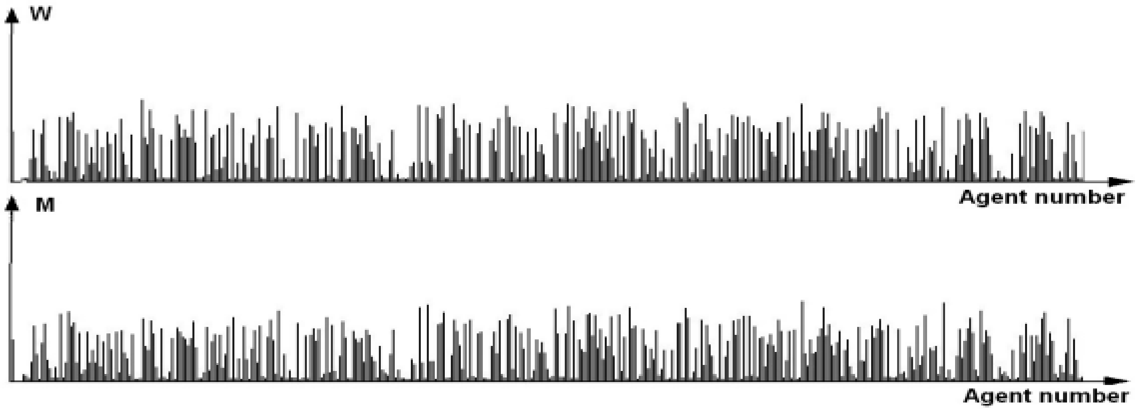

Figure 2: Inequality, agent work and income at early stage of simulation. c = 0.2

In figure 3 we can see the situation when model time approaches final simulation time. It can be seen how the inequality in the population has grown during the simulated period.

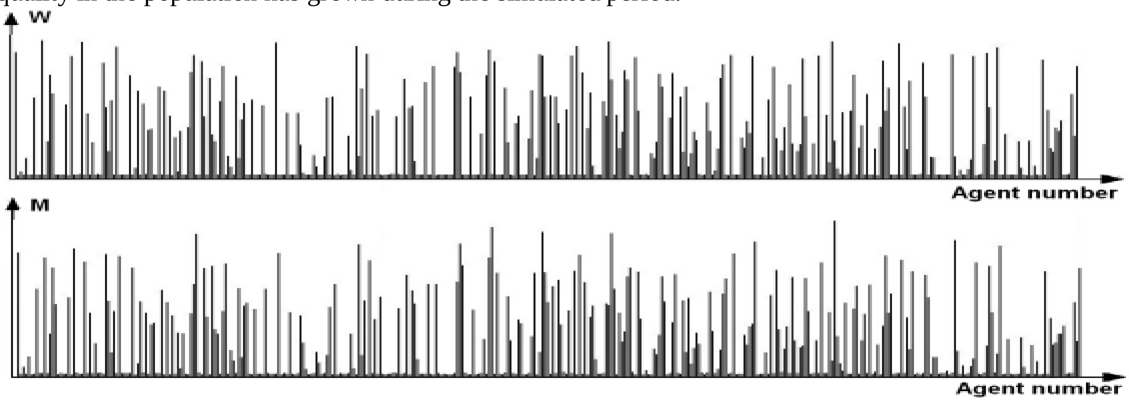

Figure 3: Inequality, agent work and income near the final simulation time. c = 0.2

b. Experiment B

In figures 4, 5 and 6 there are plots of the total amount of work done by all agents, goods accumulated in the common pool, and the average agent income, respectively. Here, $v \in (0.2, 0.9)$. The variables are plotted for different values of c = 0.15, 0.18, 0, 21, 0.24, 0.27 and 0.3.

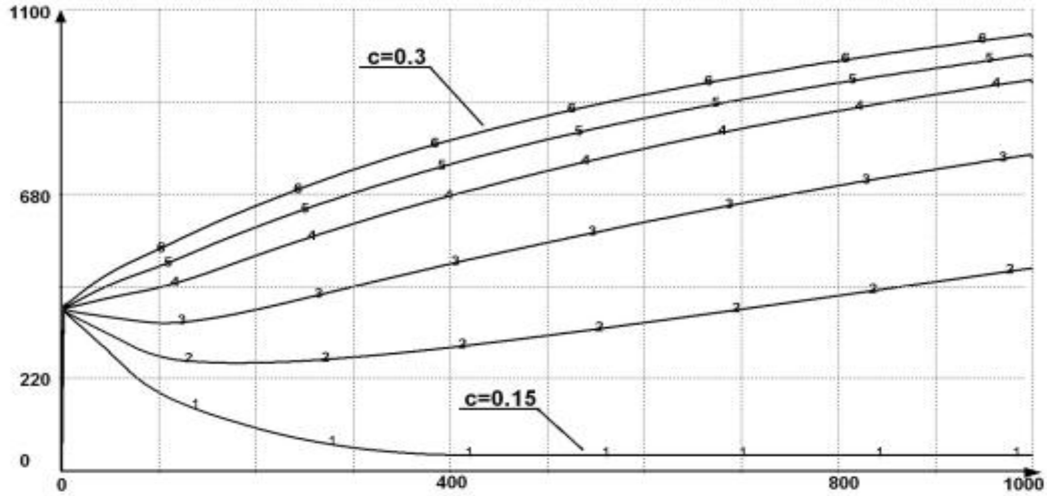

Figure 4: Total amount of work done by all agents.

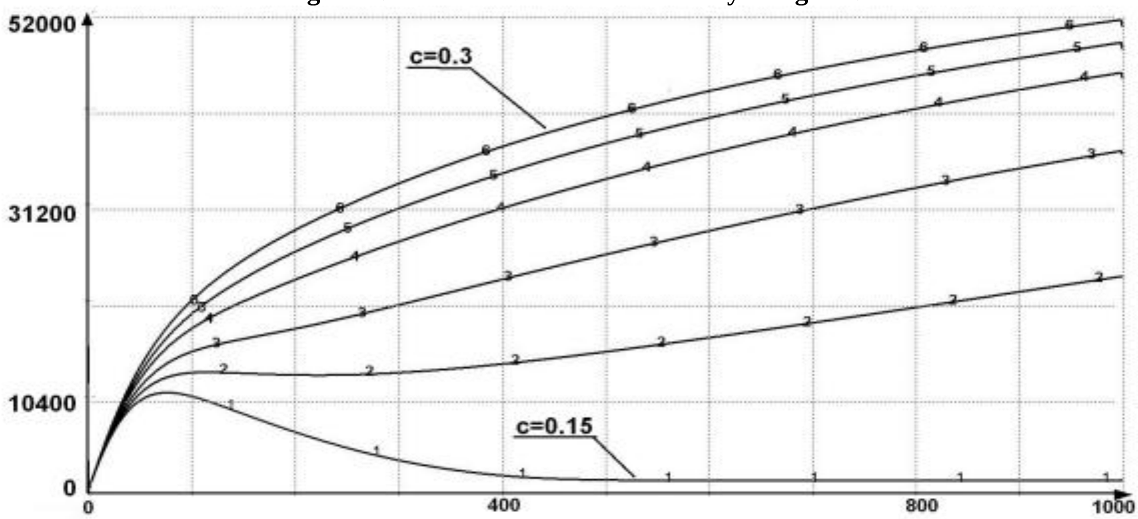

Figure 5: Goods accumulated in the common pool.

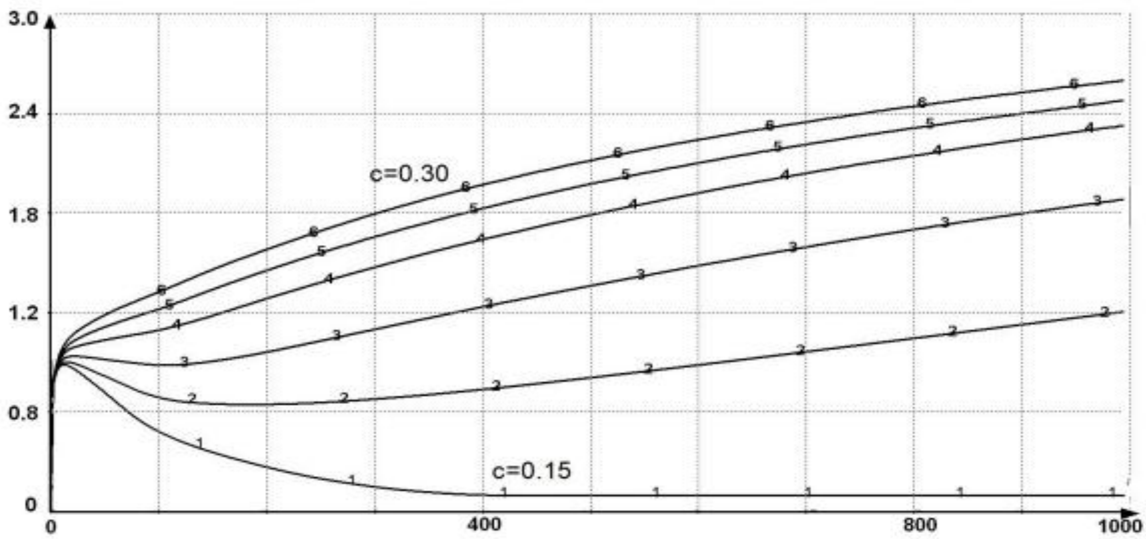

Figure 6: The average agent income, experiment B.

In figure 7, we can see the changes of the Gini coefficient for different *c*. From the above figures we can see that for greater values of *c* the Gini coefficient grows in time and there is more inequality in the population. On the other hand, if *c* is relatively low, the inequality disappears. Unfortunately, in this case the total accumulated goods result to be low, and so is the average income of the individuals.

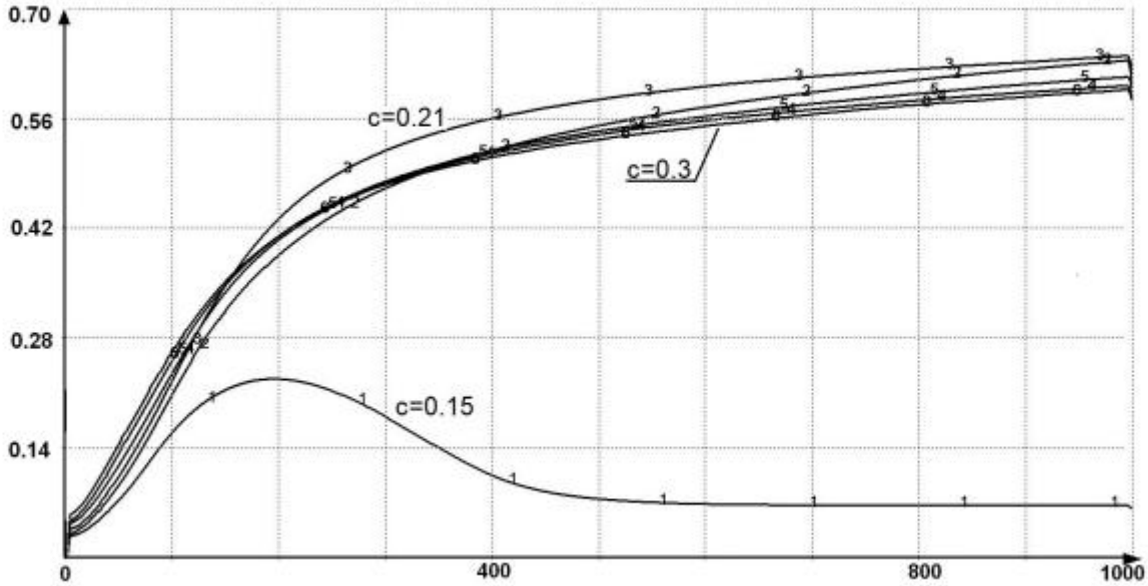

Figure 7: Changes of the Gini coefficient, experiment B.

c. EXPERIMENT C

In this experiment, we simulate the same population, with greater agent parameter *v*. It is supposed that $v \in (0.5, 1.0)$. This means that the agents tend to maximize their income, instead of minimizing the effort *W*. Comparing the plots of figures 8 and 5 we can see that the common pool of goods grows in this case, for wide range of values of *c*, and hardly depends on *c*. The average agent income grows in the similar way.
In this case, the Gini coefficient grows, as shown in figure 9.

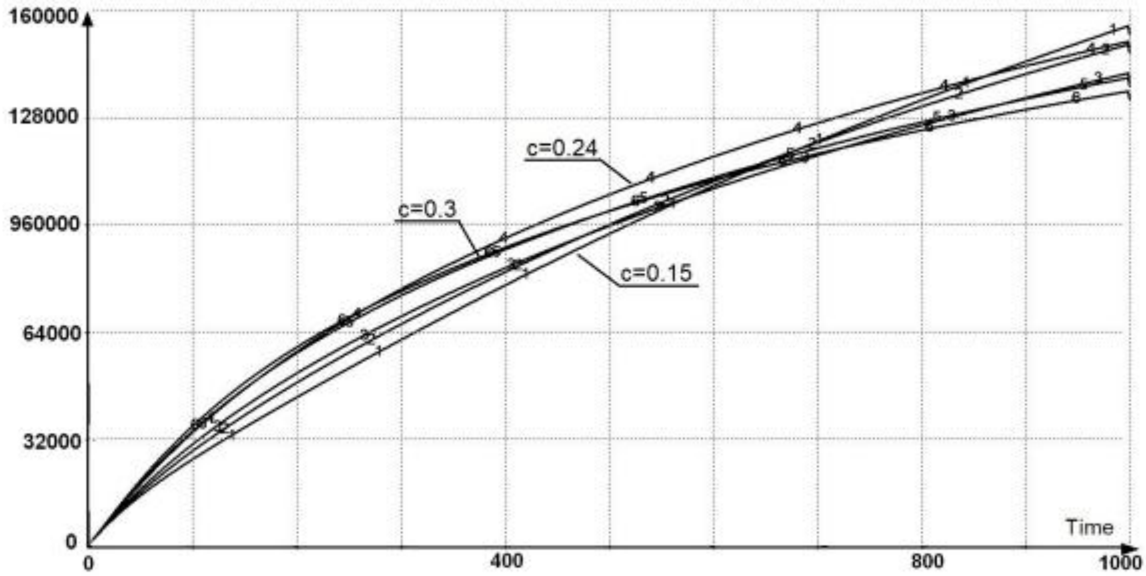

Figure 8: Total goods in common pool, experiment C.

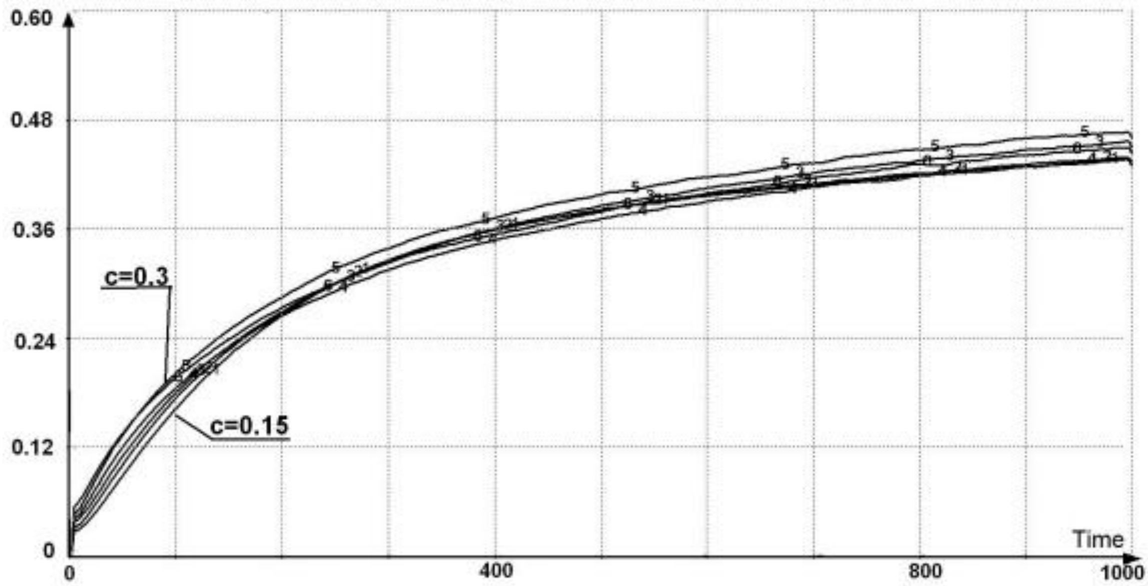

Figure 9: Gini coefficient, experiment C.

d. Experiment D

In this experiment, the global parameter "*c*" is being changed. The final simulation time is equal to 2000. Due to a social discontent caused by high inequality (Gini coefficient approaching 0.6), the social policy is changed. The value of "*c*" changes from 0.2 to zero, at model time instant equal to 1000. This reduces the social inequality, as shown in figure 10, part A. However, this policy makes the economy collapse: The total accumulated goods decrease, and the average agents income decreases to very low level.

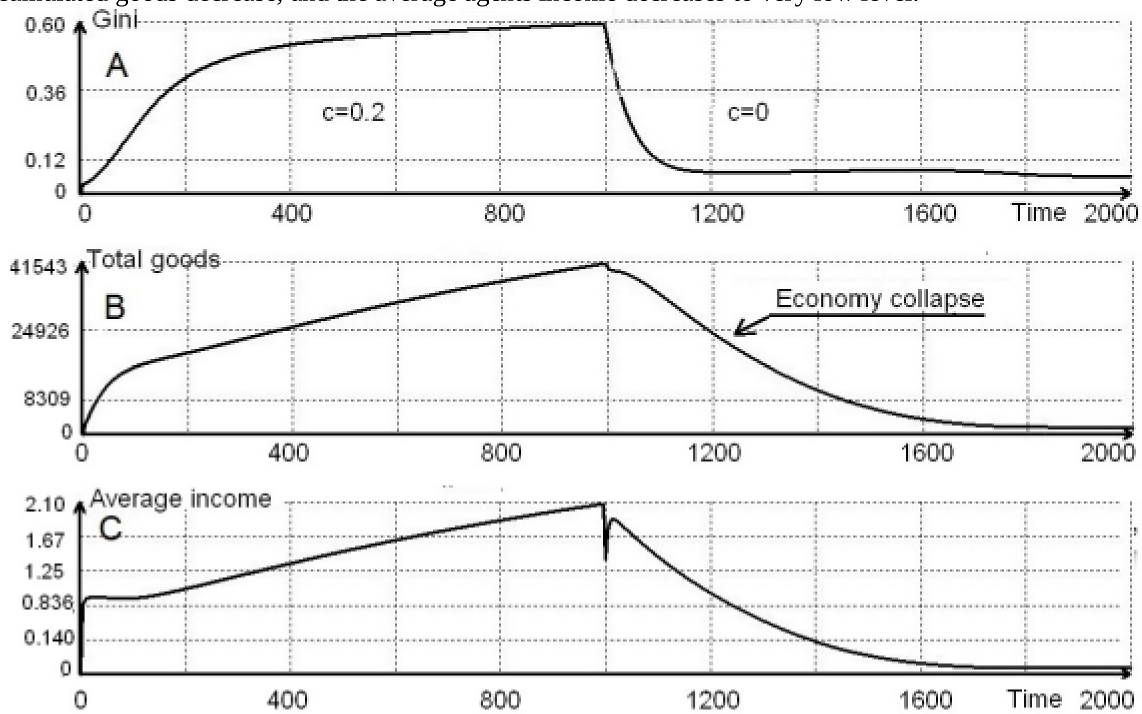

Figure 10: Changing the coefficient "c".

VI. CONCLUSIONS

The model discussed here is very simple, compared to any real socio-economic system. What we simulate is an abstract, artificial population. However, the main actions of population members (agents) resemble what real, rational individuals are doing: they maximize their income, changing the amount (including intensity and effectiveness) or work. However, the objective function is a weighted average of the income and work. This way, if the income does not grow enough while increasing the effort, the amount of work is being reduced.

The agents of the population are similar to each other, but their parameters are different. The results show how the Gini inequality coefficient changes in different simulation scenarios; One of the conclusions is that the egalitarian economic policy may cause the total economy collapse.

REFERENCES

[1]. Axelrod R. (1997) The Complexity of Cooperation: Agent-Based Models of Competition and Collaboration. Princeton University Press.

[2]. Bazzan A.I.C. (2009) Opportunities for multiagent systems and multiagent reinforcement learning in tra-c control, Autonomous Agents and Multi-Agent Systems. 18(3):342-375, DOI: 10.1007/s10458-008-9062-9

[3]. [3] Bak P. (1997) How Nature Works: The Science of Self-Organized Criticality. Oxford University Press.

[4]. Bandini S., Vizzani G. (2009) Agent Based Modeling and Simulation: An Informatics Perspective. Journal of Articial Societies and Social Simulation, 12(4), ISBN/ISSN 1460-7425

[5]. Bommel P, Becu N, Le Page C, Bousquet F (2015) Cormas, an agent-based simulation platform for coupling human decisions with computerized dynamics. In: Hybrid simulation and gaming in the network society series. Translational Systems Scienc

[6]. Casilli, Antonio A. and Tubaro, Paola, Why Net Censorship in Times of Political Unrest Results in More Violent Uprisings: A Social Simulation Experiment on the UK Riots (August 14, 2011). Available at SSRN: https://ssrn.com/abstract=19094

[7]. Chatterjee S., Seneta E. (1977) Towards consensus: some convergence theorems on repeated averaging. Journal of Applied Probability, 14:89-97.

[8]. Cioffi-Revilla C. (1998) Politics and Uncertainty: Theory, Models and Applications. Cambridge University Press, Cambridge, UK.

[9]. Coakley S, Smallwood R, Holcombe M (2006) From molecules to insect communities how formal agent based computational modeling is undercovering new biological facts. http://www. jams.or.jp/scm/contents/e-2006-7/2006-69.pdf , Scientiae M

[10]. Cohen J., Kejnal J., Newman C. (1986) Approaching Consensus can be Delicate when Positions Harden. Stochastic Processes and their Applications, 22:315-322.

[11]. Crowder R.M., Robinson M.A., Hughes H.P.N., Sim Y.W. (2012) The development of an agent-based modeling framework for simulating engineering team work. IEEE Transactions on Systems Man, and Cybernetics, Part A: Systems, 42(6):1426-1439.

[12]. Deffiruant G., Amblard F., Weisbuch G., Faure T. (2002) How can Extremism Prevail? A Study based on the Relative Agreement Interaction Model. Journal of Articial Societies and Social Simulation, 5(4).

[13]. Epstein J.M., Axtell R. (1996) Growing Articial Societies: Social Science from the Bottom Up. Brookings Institution Press, Washington, DC.

[14]. Forrester J.W. (1961) Industrial dynamics. Pegasus Communications, Waltham, MA.

[15]. Galam S., Wonczak S. (2000) Dictatorship from Majority Rule Voting. European Physical Journal B, 18:183-186.

[16]. Gini, C. (1936). "On the Measure of Concentration with Special Reference to Income and Statistics", Colorado College Publication, General Series No. 208, 7379.

[17]. Gotts N.M., Polhill J.G., Law A.N.R. (2003) Agent-based simulation in the study of social dilemmas. Articial Intelligence Review, 9(1):3-92.

[18]. H, Deguchi H, Shimizu T (2006) SOARS: Spot Oriented Agent Role Simulator design and implementation. In: Agent-based simulation: from modeling methodologies to real-world applications. Springer, Tokyo, ISBN 9784431269250

[19]. Holcombe M, Coakley S, Kiran M (2013) Large-scale modelling of economic systems. Compl Syst 22(2):175191. http://www.complex-systems.com/pdf/22-2-3.pdf

[20]. Holland J.H. (1998) Emergence: From Chaos to Order. Helix Books: Addison-Wesley Publishing Company.

[21]. Hughes H.P.N., Clegg C.W., Robinson M.A., Crowder R.M. (2012) Agent-based modelling and simulation: The potential contribution to organizational psychology. Journal of Occupational and Organizational Psychology, 85:487-502, DOI: Journal of

[22]. Klein J (2002) Breve: a 3D environment for the simulation of decentralized systems and arterials life. Conference paper: ICAL 2003 Proceedings of the eighth international conference on arterial life, MIT Press, Cambridge, MA. ISBN/ISSN 0

[23]. Krause U. (2000) A Discrete Nonlinear and Non-Autonomous Model of Consensus Formation. In: In: Elaydi S., Ladas G., Popenda J and Rakowski, Communications in dierence equations, Gordon and Breach, Amsterdam.
[24]. Latane B., Nowak A. (1997) Self-organizing Social Systems: Necessary and su-cient conditions for the emergence of Clustering, Consolidation and Continuing Diversity. In: In Barnett F.J. and Boster F.J., Progress in Communication Sciences
[25]. Long J.E. (2002) Systems Analysis: A Tool to Understand and Predict Terrorist Activities. Internet communication Vitech Corporation, URL: http://www.umsl.edu/ sauterv/analysis/62S-Long-INTEL.pdf.
[26]. Lorenz, M. O. (1905). Methods of measuring the concentration of wealth. Publications of the American Statistical Association. Publications of the American Statistical Association, Vol. 9, No. 70. 9 (70): 209219.
[27]. Luke S, Cio--Revilla C, Panait L, Sullivan K (2005) MASON: a multiagent simulation environment. Simulation 81(7):517527.
[28]. Lustick S. (2000) Agent-Based Modeling of Collective Identity. Journal of Articial Societies and Social Simulation, 3(1), URL: http://jasss.soc.surrey.ac.uk/3/1/1.html.
[29]. Macy M.W., Willer W, (2002) From Factors to Actors: Computational Sociology and Agent-based Modeling. Annual Review of Sociology, 28:143-166.
[30]. Michael JN, Nicholson T, Collier JR, Vos JR (2006) Experiences creating three implementations of the repast agent modeling toolkit. ACM Trans Model Comput Simul 16(1):125. https://doi.org/10.1145/1122012.1122013.
[31]. Obaidat M.S., Papadimitriou G.I. (2003) Applied System Simulation Methodologies and Applications. Springer, ISBN: 978-1-4613-4843-6.
[32]. Parker MT (2001) What is ascape and why should you care? J Artif Soc Soc Simul. http://jasss. soc.surrey.ac.uk/4/1/5.html.
[33]. Perez L., Dragicevic S. (2009) An agent-based approach for modeling dynamics of contagious disease spread. BioMed Central 8(50), DOI: 10.1186/1476-072X-8-50.
[34]. Platas-Lopez, Alejandro, Guerra-Hernandez, Alejandro and Grimaldo, Francisco (2021).
[35]. Railsback S.F., Lytinen S.L., Jackson S.K. (2006) Agent-based simulation platforms: Review. Simulation, 82(9):609-623, DOI: 10.1177/0037549706073695.
[36]. Raczynski Stanislaw (2000) Alternative mathematical tools for modeling and simulation: Metric space of models, Uncertainty, Dierential Inclusions and Semi-discrete Events. In European Simulation Symposium ESS2000, Hamburg Hamburg,
[37]. SWARM Development Group (2001) Swarm simulation system. Electronic citation. Electron 8(110). http://digitalcommons.usu.edu/nrei/vol8/iss1/2.
[38]. Tatai G, Gulyas L, Laufer L, Ivanyi M (2005) Articial agents helping to stock up on knowledge. Conference paper: 4th International Central and Eastern European Conference on Multi-Agent System, Budapest, Hungary, ISBN:3-540-29046-X 978-3-
[39]. Tanuma H, Deguchi H, Shimizu T (2005) Agent-based simulation: from modeling methodologies to real-world applications, vol 1. Springer, Tokyo.
[40]. Younger S.M. (2003) Discrete agent simulations of the eect of simple social structures on the benets of resource. J Artif Soc Soc Simul 6(3).

*Stanislaw Raczynski**
**Universidad Panamericana, Engineering Faculty, Augusto Rodin 498, Mexico City*