Chapter 1

SWARM ECONOMICS

SANZA KAZADI[1], JOHN LEE[2]

Abstract— The Hamiltonian Method of Swarm Design is applied to the design of an agent based economic system. The method allows the design of a system from the global behaviors to the agent behaviors, with a guarantee that once certain derived agent-level conditions are satisfied, the system behavior becomes the desired behavior. Conditions which must be satisfied by consumer agents in order to bring forth the "invisible hand of the market" are derived and demonstrated in simulation. A discussion of how this method might be extended to other economic systems and non-economic systems is presented.

Keywords: swarm engineering, Hamiltonian method of swarm design, swarm economics

1. Introduction

Economic systems are inherently difficult to predict and direct due in large part to the nonlinear nature of the system. Like most complex systems, small variations in the activity of a single behavior or characteristic of any the parts of the system can have very large effects in the whole. As a result, much of economic theory intended to explain what people are doing at the micro and macro levels is incapable of explaining much of what happens in economic systems.

The fact that economic systems are hard to understand can easily be seen in the patterns of the economy, as of today, in 2008. The economy both in the United States and in the world is undergoing huge fluctuations with causes that have their basis in human behavior. The current correction in the US housing market may be traced back to a great deal of enthusiasm in the development of the housing market during the early 2000s. The global system behavior is a result of millions of individual decisions by consumers, lenders, and governments. However, predicting that these decisions would cause such wide-ranging problems has been problematic at best, and disastrous at worst.

Economies are complex systems which encompass micro and macro behaviors, individual interaction, equilibriums, and, in most cases, some sense of self-regulation. Because of this overwhelming complexity, a quantitative form of economics has been difficult to observe. However, with more powerful computational power and the development of efficient control algorithms it is now possible to approach economics from a more quantitative, rather than theoretical, perspective . The use of these computational tools allows economists to examine many aspects of economic systems with remarkable flexibility and detail. Both local and global aspects of the economic system may be explored without the need for large data-collection enterprises or extensive human interaction. An accurate simulation can be run solely by itself and may be used to extensively examine basic economics laws and theories governing the physics of interaction of agents. An advantage of this method is the lack of the ceteris paribus[3] aspect of traditional economics. Observations qualified by ceteris paribus require that all other variables in a causal relationship are ruled out in order to simplify studies. Most economic systems are not simple enough to hold all other things constant, and examining them as though individual elements of the system can be isolated in order to be understood undermines an understanding of the true economic system. Computational studies do not require this kind of limitation, and the system may be examined in its full complexity.

Another salient advantage is an observer's ability to control the basic structure of interaction. Before a run, the simulation allows one to tinker with basic parameters of the system, such as sizes of budgets, rate of utility increase, and the magnitude of competition. By allowing such control, a user can predict results of economies in several types of real-life situations, which is key in understanding the scope of economic systems and the realistic range of our control.

Unlike traditional economic research tools, recent work in agent-based economic studies has begun to create mechanisms by which economic scenarios may be examined. These simulation tools allow millions or billions of interactions by simulated economic agents in relatively small amounts of time. By changing how these interactions work, the researcher can examine the short and long term effects of a multitude of interactions between agents and create an understanding of how these differing interactions cause varied global consequences.

Despite the power of the computational method of economic study, most computational studies have been limited in the sense that they've been more or less observational. That is, agents have been designed and the outcomes of repeated interactions have emerged over many iterations. The design has been changed and new observations made. This tends to give an idea of how changes in agent behaviors cause global changes in the system. What it doesn't address directly is how one might force a global property to emerge from the interactions of agents. As a result, economists have tried to understand what occurs in economies in order to develop predictive models. Economists have not generally developed requirements for global effects to emerge, designed agents within the system based on these requirements, and validated these requirements using computational models.

Complex system design is a challenging field of science in which some to many independent interacting parts are combined so as to create a machine or system with a particular desired function or property set. A subset of the general field of complex systems is swarms, which are groups of bidirectionally communicating autonomous agents. Swarms are interesting for a number of reasons, the most important of which is the tendency of swarms to exhibit emergence, which allows them to undertake actions that are not explicitly part of the control algorithm. The most challenging thing in complex system design is ensuring that the different parts will interact with each other in a such a way as to generate a desired system behavior. This is particularly true for systems of autonomous agents. Since each agent is independent, the interactions can be very difficult to predict, a priori.

In parallel to developing computational economic models, a new field called swarm engineering has been emerging over the past decade. This field is a subfield of engineering in which swarms are designed around global goals which have been determined prior to the swarm's design phase. The individual agents' behaviors can be shown theoretically to lead to the swarm's global behaviors, giving the swarm's design a much more robust flavor.

In the swarm literature, there is little in the way of generally applicable principled approach to swarm design. Some researchers have built preliminary systems for monitoring or understanding the emergent behaviors of agents. However, these studies do not yet generalize to a methodology that works for a large number of swarm systems. As a result, no particular method exists for generating swarms of particular design.

In this chapter, we examine what we call the Hamiltonian Method of Swarm Design (HMOSD). This method is a principled approach to swarm design consisting of two main phases. In the first phase, the global goal(s) is(are) written in terms of properties that can be sensed and affected by the agents. The resulting equation(s) can then be used to develop requirements for the behaviors of the agents that lead to the global goal. The second phase consists of creating behaviors that satisfy these swarm requirements provably. Once these have been created, it can be asserted that the resulting swarm will have the desired global goals.

Though swarm engineering has typically been applied to robotic design and computation design, we broaden the scope here by applying it to an economic system. Real economic systems are systems of autonomous agents with bidirectional communication, satisfying a broad definition of a swarm. Thus, it stands to reason that swarm engineering techniques might be able to be applied to such a system so as to generate a predefined global behavior of the system. Many studies have been made which use agent-based simulations in which interactions between agents define what the economy will do. However, though these studies extracted global behaviors from their systems, they did not develop or apply a method of generating the global behavior, and then designing the system around that behavior. This study, which might be termed a study in swarm economics, is meant to examine the design phase of an economic system using the swarm engineering methodology.

The remainder of the paper is organized as follows. Section 2 examines the theoretical application of the HMOSD to a simple economic model. This section focuses on the properties of the agents that will give the economy a particular behavior. Section 3 presents the performance of the model under different expected agent behaviors. Section 4 offers some discussion and concluding remarks.

2. Swarm engineering basics

In this section, we give an overview of swarm engineering theory. We begin with a set of definitions that clarify and make rigorous some of the concepts behind swarm engineering. We continue with a theoretical description of the steps behind swarm design and proof of design efficacy.

Definitions

We assume that a system can be thought of as a closed set of objects together with a set of consistent dynamic properties. These properties need not have closed form expressions, but we assume that they are consistent in the sense that measurements or combinations of measurements cannot produce differing numerical values for any measurable quantity. Because the system is closed, the objects are not affected by anything outside of the system. As a result, in simulations involving an outside controller of an agent in the simulation, the controller and everything that affects it must be viewed as part of the system.

We define a property of the system to be a characteristic of the system that can be measured using a process that is independent of the characteristic. In what follows, we'll represent a system's property as P_i where the subscript i serves to identify the property. As an example, the temperature of a processor may be measured using the radiative emmissions of the processor, even though the measurement cannot affect the processor's temperature [4].

We define an agent to be a situated subset of the system that exhibits autonomous control over at least one degree of freedom in the system. Autonomous control is control which does not exhibit a direct dependence on any part of the system other than the controlling element(s); the behavior of the agent also must not be attributable to the dynamic interactive equations that define the system.

An autonomous agent is an agent that acts without the direct control of any outside influence. This means that outside of the things that it can sense, no part of the outside world affects any part of the agent's controller. While the agent can be affected by other things that it can sense, the effect of the senses is expected to be independent of the cause of the sensory input to its controller. Anything failing to meet this metric cannot be thought of as autonomous.

We may quantify this idea. Let the controller of an agent be defined by the way in which the agent responds to its memory state M and its sensory state S_s. Then, given any outside property of the system P_S, it is true that if the current state of the agent is given by S_a then

$$\frac{dS_a}{dt} = \frac{\partial S_a}{\partial M}\frac{dM}{dt} + \frac{\partial S_a}{\partial S_s}\frac{dS_s}{dt} + \frac{\partial S_a}{\partial P_S}\frac{dP_S}{dt}. \tag{1.1}$$

That the final term is zero for all outside properties is a necessary and sufficient condition for autonomy. Now, this does not say that $\frac{dS_s}{dP_S}$ is zero. It simply means that the only way that this property may enter the controller is through the senses.

We define the behavior of a subset of the system to be the way in which properties of the subsystem change in time. Ie., if P_i is a property, then a behavior b_i is defined by

$$b_i = \frac{dP_i}{dt}. \tag{1.2}$$

Behaviors often involve the interplay between more than one property. In this case, we require a formalism for describing such behaviors. Let us suppose that a system is made up of elements whose behavior is defined in terms of measurables $A = \{P_1, \ldots, P_n\}$. Then this system can have a behavior which is composed of all of the behaviors of the different measurables. That is,

$$\overrightarrow{b_A} = \left(\frac{dP_1}{dt}, \ldots, \frac{dP_n}{dt} \right). \tag{1.3}$$

These properties can be most easily thought of as composite properties of many agents or objects in the system. For instance, a star has a discernable size which is defined as a combination of the positional properties of the atoms making it up. Any single element of the system would be insufficient to describe the system. Thus, the property must be described in terms of the properties of all (or at least many) of the atoms in the system.

In many physical systems, there are properties that are derivations of other properties. These properties are not basic in the sense that they do not depend on dynamics of other properties. As an example, consider a point mass in our universe. We may define its position in terms of a variable $\overrightarrow{x}$. However, another property, the velocity $\overrightarrow{v}$, is a derivative property whose relationship with the basic positional property is given by

$$\overrightarrow{v} = \dot{\overrightarrow{x}}. \tag{1.4}$$

It is possible to measure this property of the object, and so it is indeed a property of the system, as well as a behavior of the object. This duality of behavior and property can be resolved only by noting that behaviors are linked to properties, but the behaviors can only become properties if they, in fact, can be independently measured.

Emergence has been identified by many authors in the past in terms capturing the general idea that a system can have unintended global properties that are not explicitly built into its agents. The interest in swarm based systems seems to have come from this single observation. We now propose a rigorous definition of this property.

Suppose that we have a property P_j that is a function of another properties and behaviors of the system. That is, suppose that

$$P_j = f\left(b_1, \ldots, b_{n_b}, P_1, \ldots, P_{j-1}, P_{j+1}, \ldots, P_{n_P}\right), \tag{1.5}$$

where n_b is the number of systems behaviors, and n_P is the number of systems properties. The number of behaviors is not necessarily equal to the number of properties of the system. The property P_j is an emergent property of the subsystem i if

$$\frac{\partial b_i}{\partial P_j} = 0. \tag{1.6}$$

That is, the property P_j is not a factor in the defining function of behavior b_i for any of the behaviors of the elements of the system. This means that the agent or agents in the system are acting independently of the property, and so the property is not a deliberate result of the design of the agent's behaviors. As a result, it cannot be viewed as part of the design of the agent(s), and so it satisfies the meaning of emergence.

Given the distinction between behaviors and properties above, we can also define emergent behaviors to be emergent properties that are themselves behaviors.

These definitions may be used to formally define various types of swarms of agents. Firstly, we define a swarm of agents to be a set of interacting agents within a system in which one agent's change in state can be perceived by at least one other agent and effects the state of the agent perceiving the change. Moreover, the subset must have the property that every agent has at least one state whose change will initiate a continual set of state changes that affects every other agent in the swarm.

Let us more rigorously define a swarm. Suppose that the state of the agents is specified by a set of varibles $\{S_i\}$. Then the set of agents is a swarm if

$$\frac{\partial S_i|_{(t>t_0)}}{\partial S_j|_{(t=t_0)}} \neq 0 \tag{1.7}$$

[5] $\forall i \neq j$ for times t after some reference time t_0. That is, that the later states of agent i must depend on the current state of agent j.

Our definition of a swarm differs from others given in the literature in that it does not demand emergence from the system. However, emergent swarms are also interesting, and form the basis for most of the work in swarm engineering. Thus, we define a swarm of agents as an emergent swarm of agents with respect to property P_j if they exhibit an emergent behavior b_{P_j}. Note that this means that a swarm is defined only in terms of a specific property which yields the potential possibility that the group of agents is not a swarm with respect to another property P_k.

One of the unexpected results of this definition is that it does not exclude the potentiality of a centrally controlled swarm. The idea behind the swarm is that each element of the swarm is capable of initiating a cascade of state changes. How these are initiated is not important, and we can leave the possibility open that these go through a central controller, group of agents, or communication mechanism. Thus, we clarify these issues by defining a decentralized swarm to be a swarm that does not have a central communication or control mechanism. A centralized swarm is a swarm which is not decentralized.

The power of these definitions is that it is possible to test a set of agents in order to determine whether or not it is a swarm, if it is a centralized or decentralized swarm, and then whether or not it is an emergent swarm with respect to a specific property. For instance, it should be clear that a soccer team is a swarm, but it is not an emergent swarm with respect to, for instance, the team dispersion. Team members are very likely to use this information to affect their own behaviors. On the other hand, a swarm of ants is an emergent swarm with respect to food source exploitation, as it has the ability to exploit nearby food sources despite the absolute lack of knowledge on the part of the ants. This can be characterized by measuring the amount of exploitation of each food source when multiple food sources are available. Clearly this quantity is not part of the control algorithm of the agents.

Swarm engineering

Swarms are difficult to engineer primarily because groups of independent interacting agents can exhibit very complex and unexpected behaviors for a very large number of different reasons. Moreover, if the members of a group have specifications that are made independently, it is very difficult to guarantee that the specifications do not interact in an unexpected way. Moreover, proving that the interactions between the various agents have the desired outcome often requires the complete simulation of the group of agents. Finally, small perturbations to the system, which cause rather small changes in the behaviors of individual agents, can cause very large changes in the overall behavior of the system. This is, in fact, a foundational characteristic of the field of chaos.

It is important to create a new methodology for the generation of global behavior in a way that bypasses the difficulties presented here. We seek a method that is provable in the sense that the behaviors can be understood to generate the desired global behavior. The generated behaviors have well understood tolerances for perturbations within which the desired global behavior will still occur.

Swarm Engineering Equations

In this section, we explore differential equations which are relevant to swarm design. This method will assist in the determination of several things relevant to the overall global goal. The first thing needed is a set of different behaviors (also called castes) for the agents. The second thing is the set of sensors and actuators, with well-defined resolutions. Sensors with higher resolution (in the sense that they can measure the desired property with higher accuracy) can be used to affect the number of castes, as new ones can emerge at different times during the entire group action. This is the entire top-down portion of the design process.

Once the castes have been properly designed, the next step is to work out the specific sensory and actuation capabilities of the agents. This step consists of determining the actual hardware (either physical or virtual) that the agents will have, their computational capabilities, communication capabilities, etc. This hardware must make it possible for the agent to have the sensory and computational abilities determined in the top down part of the swarm design. Most notably, the agent must have the ability to determine what part of the phase space path it is on in order to properly determine its behavior (caste). The behaviors must also be developed at this step. If the behaviors have the ability to move the agent along the proper section of the path through phase space during the appropriate caste behavior, the global goal should be achieved.

Top down

As our starting point we choose the global goal. It is described in terms of a set of properties of the swarm $G = \{P_1, \dots, P_i, \dots, P_{n_P}\}$ and their corresponding initial and final characteristics $G^0 = \left\{P_1^0, \dots, P_i^0, \dots, P_{n_P}^0\right\}$ and $G^F = \left\{P_1^F, \dots, P_i^F, \dots, P_{n_P}^F\right\}$. The initial and final characteristics may be numerical values as in a count-based characteristic or they may be functional, as in a trajectory. They may also be sets of potential initial or final states of the two forms.

Once these initial and final conditions have been determined, it is important to specify conditions under which the final characteristics become consequences of the initial conditions and the system dynamics.

Assume that function f from (1.5) is a differentiable function of the properties P_i, $i = 1, \dots, n_p$ and the behaviors b_i, $i = 1, \dots, n_b$. Then, in general case the following holds

$$b_j = \frac{dP_j}{dt} = \sum_{i=1}^{n_b} \frac{\partial P_j}{\partial b_i}\frac{db_i}{dt} + \sum_{i \neq j}^{n_p} \frac{\partial P_j}{\partial P_i} b_i. \tag{1.8}$$

For simplicity we assume that each property correspond with only one behavior, i.e., $n_P = n_b$, then

$$b_j = \frac{dP_j}{dt} = \sum_{i \neq j}^{n_b} \left(\frac{\partial P_j}{\partial b_i}\frac{db_i}{dt} + \frac{\partial P_j}{\partial P_i} b_i \right) + \frac{\partial P_j}{\partial b_j}\frac{db_j}{dt}. \tag{1.9}$$

This expresses the idea that the change in the property is a function of the connectivity between other properties of the system and the behaviors which define this property. Thus, we wish to find a set of conditions such that

$$\lim_{\tau \to \infty} \int_0^\tau \sum_{i \neq j}^{n_b} \left(\frac{\partial P_j}{\partial b_i}\frac{db_i}{dt} + \frac{\partial P_j}{\partial P_i} b_i \right) dt + \lim_{\tau \to \infty} \int_0^\tau \frac{\partial P_j}{\partial b_j}\frac{db_j}{dt} dt + P_j^0 = P_j^F. \tag{1.10}$$

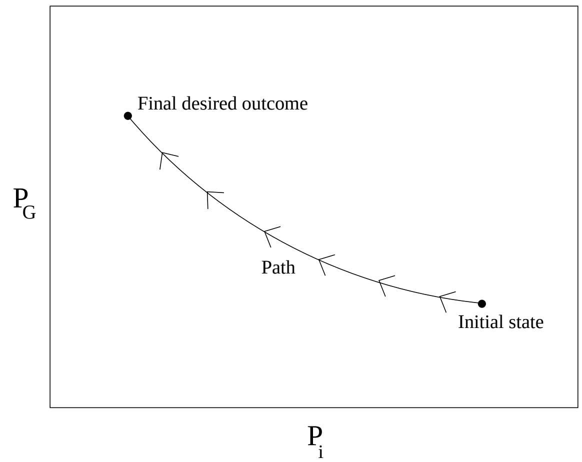
This is the general swarm engineering condition, and must be fulfilled by the behavior and sensor sets. Behaviors of the system depend on behaviors of agents in that system. Those, in turn, depend on agents' sensors, memory state, behavioral strategy, and position.

Note that in equation (1.9), each of the entities b_i and $\frac{db_i}{dt}$ represent the behavior associated with P_i and its rate of change. These behaviors are changes in the property P_i, which can only happen through the action of the agents. This is a very powerful equation, as it indicates precisely which behaviors might be used to effect the global change. Clearly, equation (1.10) can be satisfied in a number of ways, with respect to the various individual properties. The stronger condition requires specific changes in all properties. I.e.

$$\lim_{\tau \to \infty} \int_0^{\tau} \frac{dP_i}{dt} dt = P_i^F - P_i^0. \tag{1.11}$$

In this case, each of the individual properties changes in a specific way, causing the overall change, assuming that P_j is single-valued.

We can imagine the change happening in a phase space, of sorts. In this phase space, each point represents a set of values for each of the properties. In order to ensure that the global goal is achieved, the system must follow a path through the state space. We can imagine a state space made up of n_b-dimensional vectors such that each point represents a different system state. The initial state would then be a point, and the motion of the system through state space would be achieved by the behaviors of the agents. That is, every action of the agents will move the system. The trick is to direct the agents' behaviors so that the path through state space will connect the initial point to the end point in a stable way. A system in which the final state is the outcome of any initial state is a system in which the final state is an attractor.

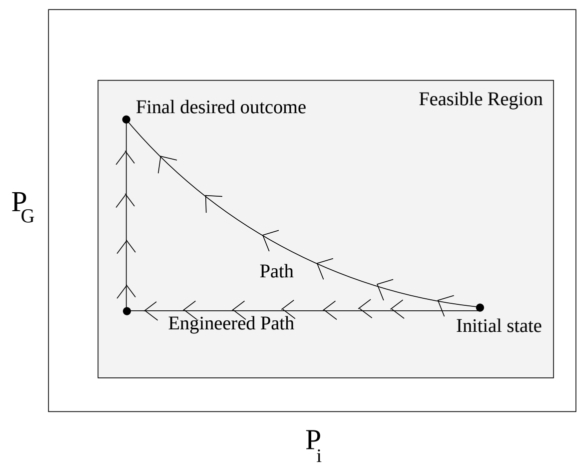


In general, not every configuration in phase space is allowed. If any group of properties is connected to one another, the connection may preclude certain areas of phase space. For instance, if one property is the distance between two objects, and a second is the distance between one of these objects and a third object, certain restrictions occur in the feasible points in phase space. In this case, the triangle inequality must apply, and this limits the range in phase space of the system.

Let us examine the possibility that the global property is constructed from other properties that are independent of one another. Then, the feasible region of the phase space is the cross product of the feasible regions of each of the individual properties. That is, if A_i is the set of all feasible choices for P_i, then

$$A = \bigotimes_{i=1}^{n_b} A_i \tag{1.12}$$

is the set of all feasible points in the space. Suppose that each A_i is continuous. Then a piecewise linear path will connect the starting and ending points. Moreover, each of these linear segments may focus on a single property, indicating a specific task for the swarm. The situation is depicted in the next Figure in which a feasible region is clearly graphed, along with the engineered path through the phase space.

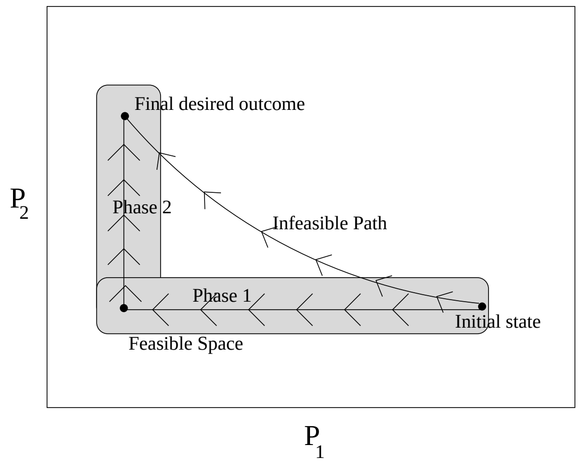


In this Figure, the path has two segments. The first segment illustrates the change of one of the properties, while the second illustrates the second. The agents that carry out this evolution of the system must therefore be able to sense the current state of both properties, determine when the system has reached the desired endpoint, and modify each of the system's components indicated by these properties.

As the system has two independent properties, our choice of how these actions can be achieved is quite open. We can have a single individual that works on one property or the other according to opportunity. On the other hand, one could build a system with two distinct castes of agents that each affect different properties simultaneously. Finally, we might have a single caste of agents that completes work on one property and then works on a second. This freedom is available to us once we realize that the properties are independent and therefore don't require synchrony. The engineer is free to choose how this is done.

Let us try to understand what this means. Each of the path segments is independent of one another, and so modification of this property only requires that some agent is capable of doing the modification. This, in turn, tells us a few practical things. First, it gives us an idea of the sensory capability of the agent. It must be able to discern under what conditions it should act with enough specificity to avoid changing any of the other properties and ending at the appropriate endpoint. Second, it must be able to carry out the behavior changing the system state. This gives us an idea of its physical requirements. Finally, the number of different behavioral states or different castes of agents is indicated by the number of different independent properties. These can work independently and in sequence or in tandem.

On the other hand, sometimes the properties are not independent of one another. In this case, the feasible region may not include the entire cross product space. This means that the path through phase space is much more constrained. The situation is depicted in the Figure below.



In this case, the path must necessarily result from the modification of property one first, and then property two. As a result, the swarm must consist of at least two disparate behavior sets. The first behavior set must move the system through phase space along property one, while the second behavior set must move the system through phase space along property two. This indicates that at different times, the swarm must behave in different ways. This is extremely important. It indicates a few things. First, it indicates two different behavioral castes. These castes are disparate in their behavior, and they act independently. This may be achieved practically by two different means. In the first, there are physically two different sets of agents which become active at the appropriate times. In the second, there is one set of agents, capable of discriminating between the two situations, and deciding how to behave based on the specific situation. Of course, in both situations, it is necessary that the agents carry sufficient computational and sensory machinery as to be able to discern which state the system is in, so that the correct behavior is achieved. This is identical to the previous situation, though in this case, the constraint on the behaviors is that they cannot happen in tandem. In this case the behaviors must occur independently and sequentially. Thus, the planning and possibly the behavior must be more precise.

If the feasible region does not include a complete path from the initial point to the desired final point, then the final completed task is impossible. In this case, the swarm may not be constructed using the sensors (properties) specified, though a path through a larger phase space enhanced by a new property might be able to connect disconnected phase spaces.

What these considerations allow us to do is to determine from the global property the agents' sensory requirements including resolution, the agents' physical capabilities, and the types of agents. This is a very useful set of information and can be used to determine how to build the set of agents for the task. This is the top-down portion of the swarm engineering methodology. Next, we consider the bottom up portion.

Bottom up

Once we've worked out the top-down considerations, the remainder is relatively straightforward. The top-down considerations should clearly indicate how many behavioral castes there are, what sensory and/or computational capabilities are required, including the resolution of these sensory and computational capabilities, and how the different castes should be deployed. The remainder of the job consists of developing agents which meet these requirements. In this subsection, however, we'll examine the effect of

Let us assume that our swarm consists of $\{N_A\}$ agents. First, we may assume that the lth agent's state may be completely described by its memory state m_s^l, its internal state in_s^l, its sensor state s_s^l, its positional state p_s^l (which expresses its position and higher derivatives of position), and its behavioral strategy k^l. Note that k^l may be a function of time and it may be able to take on one of multiple states. Moreover, transitions may be triggered by sensor states. Then, we may express the global behavior b_j as a function of a number of things. First, the coupling between a global property and an agent behavior is defined, in part, by the positional state of the agent. We define the coupling between agent l and the global behavior b_j by $C_{jk}^l(p_s^l, in_s^l)$. Secondly, we describe the individual behavior of the agent by $AB_{k^l}^l(m_s^l, in_s^l, s_s^l)$. Then, the overall behavior may be expressed as

$$b_j = \sum_{l}^{N_A} C_{jk}^l(p_s^l, in_s^l) AB_{k^l}^l(m_s^l, in_s^l, s_s^l). \qquad (1.13)$$

[6] The trick, then, is to create behaviors that are dependant on realistic sensor states and internal states which provably satisfy equation (1.10). In many studies, equations $(1.13 - 1.35)$ are converted to average behavioral equations, greatly simplifying the required analysis.

Combining equations (1.10) and (1.13), we obtain the general combined agent-swarm equations:

$$\lim_{\tau\to\infty}\int_0^\tau \sum_{i\neq j}^{n_b}\left(\frac{\partial P_j}{\partial b_i}\frac{db_i}{dt}+\frac{\partial P_j}{\partial P_i}b_i\right)dt$$
$$+\lim_{\tau\to\infty}\int_0^\tau \frac{\partial P_j}{\partial b_j}\frac{db_j}{dt}dt+P_j^0$$
$$=\lim_{\tau\to\infty}\int_0^\tau \left[\sum_l^{N_A} C_{jk}^l(p_s^l,in_s^l)AB_{k^l}^l(m_s^l,in_s^l,s_s^l)\right]dt$$
$$+P_j^0=P_j^F. \tag{1.14}$$

Swarm engineering is concerned with balancing these equations linking the agent behaviors and the global desired behaviors.

3. Swarm engineering applied to economic systems

In this section, we will theoretically explore the application of the principles of swarm engineering to economic systems. In swarm engineering, we are primarily interested in generating group behaviors by utilizing careful examination of the desired global behavior and using this analysis to guide the design of agent-level behaviors capable of producing the desired global behavior [6]. While this method still requires considerable input from the engineer, we have been able to use it to solve previously unsolved problems in deployment of swarms. In the present study, this means that we are interested in examining one or more global economic measurables and putting together a method of directly manipulating these by designing specific agent behaviors.

In economic systems, there are many global measurables. Each one is tied to local variables in a complicated and non-linear way. This makes the prediction of the global effect of a specific local behavior very difficult. As a result, it is often times simpler to utilize agent-based systems to get an idea of the effect of specific behaviors. The difficulty with utilizing agent-based systems in this way derives from the difficulty in creating a new system with specific desired qualities; the nonlinearity of complex systems makes this a very difficult thing to do. As a result, we utilize the swarm engineering methodology, which draws its initial motivation from the desired global outcome.

As our global property, we choose a truly dispersed property – that of the average cost of a commodity across vendors for sales of specific commodities. This property is interesting because it measures how much a consumer pays for goods and services that are worth a specific amount. If all vendors tend to end up with similar prices, this indicates that either the system is designed to enforce a specific price, or that there is some kind of communication between vendors that allows them to collude. We shall see that there are specific system designs that allow the former to occur without collusion or any communication between vendors.

We examine the design of consumer behavior as a method of controlling the average price. Vendors are modelled as profit maximizers who will increase their prices when all else is kept constant. The reaction of the consumers must be made in such a way that slow creeping price increase does not occur. We shall see that specific agent behaviors, designed properly, can limit the average prices to prices that very closely match the cost of vending the product.

Vendors and consumers

We begin by modelling the main factors that affect the vendors in their decision to alter prices of commodities that they are selling. We begin with the assumption that all vendors will choose a price for a commodity that equals or exceeds his or her costs incurred during the sale of the commodity. The question then is what factors affect the change in the price?

We begin by assuming that the price function used by a vendor is a complicated function of several different values. That is, let the price be represented as

$$p_{v,c} = f\left(m_1, m_2, \ldots, m_n\right). \tag{1.15}$$

Then, each of these values m_i represents a factor in determining the price of the commodity.

There are many factors one might include in a decision about the cost of a commodity or in a decision about whether or not to increase the cost of a commodity. Among these factors are the demand for the commodity (D), the vendor's account balance (b), the total cost of the commodity to the vendor including the cost to put it on the shelf (space, cost outlays, and personnel) (c_c), any memorized or recorded data of the past l cycles ($\{m_j\}_{j=1}^{l}$), and the current income of the vendor (i). We assume, for the moment, that these are the main effectors of the cost of the commodity.

As we stated above, our goal is to examine the dynamics of the average price of specific commodities. This is the average price over all vendors of the commodity. I.e.,

$$P_{a,c} = \frac{1}{N_v} \sum_{v=1}^{N_v} p_{v,c} \tag{1.16}$$

where $P_{a,c}$ is the average price for commodity c, N_v is the number of vendors, and $p_{v,c}$ is vendor v's price for the commodity c.

In real economic systems, the average price of a specific commodity typically remains stable or increases over time. However, theoretical prices should actually decrease or remain stable over time as the cost of production decreases. Moreover, the market is assumed to produce corrections to initially poorly priced items (i.e. items whose prices are much higher than the cost to produce it). We are interested in discovering what the minimal conditions are for consumers which will result in commodity prices that decrease or stabilize over time. This can be written mathematically as

$$\frac{dP_{a,c}}{dt} = \frac{1}{N_v}\sum_{i=1}^{N_v}\frac{dp_{v,c}}{dt} \leq 0. \tag{1.17}$$

If a single vendor's prices start decreasing, then under competitive conditions, all vendors' prices should start decreasing. This being the case, we don't expect one vendor's price to increase while any of the other vendors' prices decrease. As a result, we can replace the requirement of (1.17) with

$$\frac{dp_{v,c}}{dt} \leq 0. \tag{1.18}$$

If we begin by assuming that the vendors have a systematic method to their pricing choices, then we may write the prices faced by consumers as (1.15). Utilizing the various measurements indicated above, this means that

$$\frac{dp_{v,c}}{dt} = \frac{\partial f}{\partial D}\frac{dD}{dt} + \frac{\partial f}{\partial b}\frac{db}{dt} + \frac{\partial f}{\partial c_c}\frac{dc_c}{dt} + \sum_{j=1}^{l}\frac{\partial f}{\partial m_j}\frac{dm_j}{dt} + \frac{\partial f}{\partial i}\frac{di}{dt}. \tag{1.19}$$

The term in (1.19) $\frac{\partial f}{\partial c_c}\frac{dc_c}{dt}$ would seem to have little to do with the consumers, and so cannot be directly affected by a behavioral change among consumers. We therefore ignore it as a potential design point. On the other hand, it is interesting to note that $\frac{db}{dt}$ is the rate at which the bank account changes. Thus, we identify this with the profit. If profit is Pr then,

$$Pr\,(t) = \frac{db}{dt} = D\,(t)\,(f\,(t) - c_c\,(t))\,. \tag{1.20}$$

where $D\,(t)$ represents the number sold per time period. Moreover, this profit/loss may be memorized by the agent, affecting behavior. For each vending agent, the behavior can be different, but in general

$$m_k\,(t) = Pr\,(t - kt_p) = D\,(t - kt_p)\,(f\,(t - kt_p) - c_c\,(t - kt_p)) \tag{1.21}$$

where t_p represents a time period and k represents the specific memory element being stored. k typically runs from 1 through N_m, the number of memory elements used in the function.

Since we are examining conditions that make $\frac{dp_{v,c}}{dt}$ a non-increasing function of time in the absence of inflation and supply variations, we want

$$0 \geq \frac{\partial f}{\partial D}\frac{dD}{dt} + \frac{\partial f}{\partial b}\frac{db}{dt} + \frac{\partial f}{\partial c_c}\frac{dc_c}{dt} + \frac{\partial f}{\partial m_{p/l}}\frac{dm_{p/l}}{dt} + \frac{\partial f}{\partial i}\frac{di}{dt}. \tag{1.22}$$

As a result, we have that

$$\frac{\partial f}{\partial D}\frac{dD}{dt} \leq -\left(\frac{\partial f}{\partial b}\frac{db}{dt} + \frac{\partial f}{\partial c_c}\frac{dc_c}{dt} + \frac{\partial f}{\partial m_{p/l}}\frac{dm_{p/l}}{dt} + \frac{\partial f}{\partial i}\frac{di}{dt}\right) \tag{1.23}$$

Inserting the results of (1.20) and (1.21) reveals that the actual form of this equation becomes

$$\begin{aligned}\frac{\partial f}{\partial D}\frac{dD}{dt} \leq &-\left(\frac{\partial f}{\partial b}\left(D\left(t\right)\left(f\left(t\right) - c_c\left(t\right)\right)\right) + \frac{\partial f}{\partial c_c}\frac{dc_c}{dt} + \frac{\partial f}{\partial i}\frac{di}{dt}\right) \\ &-\left(\sum_k \left[\frac{\partial f}{\partial m_{p/l}} D\left(t - kt_p\right)\left(f\left(t - kt_p\right) - c_c\left(t - kt_p\right)\right)\right]\right)\end{aligned} \tag{1.24}$$

In the case that the vendor simply reacts to current conditions, the relation takes the form

$$\frac{\partial f}{\partial D}\frac{dD}{dt} \leq -\left(\frac{\partial f}{\partial b}\left(D\left(t\right)\left(f\left(t\right) - c_c\left(t\right)\right)\right) + \frac{\partial f}{\partial c_c}\frac{dc_c}{dt} + \frac{\partial f}{\partial i}\frac{di}{dt}\right). \tag{1.25}$$

Now, we examine (1.24) to determine the form of f.

1 If the cost to the vendor increases, it is reasonable to expect the vendor to either increase or hold steady its prices. That is

$$\frac{dc_c}{dt} > 0 \Rightarrow \frac{\partial f}{\partial c_c} > 0. \tag{1.26}$$

2 If the income increases, one can infer that the demand at a particular price has increased. Therefore, by increasing the price, the profit will increase. Thus, we expect that

$$\frac{\partial f}{\partial i} > 0. \tag{1.27}$$

3 If profit increases, one can infer that the demand at a particular price has increased. Therefore, by increasing the price, the profit will increase. Thus, we expect that

$$\frac{\partial f}{\partial b} > 0.$$

4 If the demand increases, typically the price increases. Therefore we expect that

$$\frac{\partial f}{\partial D} > 0$$

These results together give us that

$$\frac{dD}{dt} \leq -\frac{1}{\frac{\partial f}{\partial D}} \left(\frac{\partial f}{\partial b} \left(D\left(t\right) \left(f\left(t\right) - c_c\left(t\right)\right)\right) + \frac{\partial f}{\partial c_c}\frac{dc_c}{dt} + \frac{\partial f}{\partial i}\frac{di}{dt} \right)$$
$$- \left(\sum_k \left[\frac{\partial f}{\partial m_{p/l}} D\left(t - kt_p\right)\left(f\left(t - kt_p\right) - c_c\left(t - kt_p\right)\right)\right]\right) \qquad (1.28)$$

or in the case that the agents are purely reactive

$$\frac{dD}{dt} \leq -\frac{1}{\frac{\partial f}{\partial D}} \left(\frac{\partial f}{\partial b} \left(D\left(t\right) \left(f\left(t\right) - c_c\left(t\right)\right)\right) + \frac{\partial f}{\partial c_c}\frac{dc_c}{dt} + \frac{\partial f}{\partial i}\frac{di}{dt} \right). \qquad (1.29)$$

We have just proved the following theorem.

THEOREM 3.1 If the condition in equations (1.28) or (1.29) continually holds, then the price will be bounded above.

These last two equations give the limits of the behavior of the consumer agents in a system composed of the vendor and consumer agents only. It indicates that the consumer agents must respond with a decrease in the demand for a commodity which is greater in magnitude than the magnitude of the right hand side of equations (1.28) and (1.29). This is a severe design requirement on the consumer agents. However, as we will see in the next section, systems containing consumer agents which follow these restrictions do tend to have the desired global characteristics, while those that do not tend to have significantly higher to run-away prices.

Design of consumer agents

Our primary concern is that the consumer agents provably behave in such a way that the global average price remains bounded above. We have already seen in section 2.1 that if the conditions in equations (1.28) and (1.29) are obeyed, the goal will be achieved. That completes the top-down portion of the design problem. We now have an engineering requirement with which to work. We can now begin the bottom-up phase.

In this new phase, we must generate agents that satisfy this requirement. The general solution to the general equation given in (1.29) if $\frac{\partial f}{\partial i} = \frac{dc_c}{dt} = 0$, $\frac{\partial f}{\partial D} = \alpha, and \frac{\partial f}{\partial b} = \gamma$, the general solution is

$$D = e^{-\int_0^t -\frac{\gamma}{\alpha}(f(t') - c_c(t'))dt'}. \tag{1.30}$$

As a result of this general solution, it is clearly the case that, in order to react correctly in the next time frame, our agents must have the following capabilities.

1 The agents must be able to measure the price of the commodity.

2 The agents must be able to measure the demand for the commodity. In our simulations, it is a good estimate to know one's own probability of purchasing the commodity and multiplying by the population size.

3 The agents must be able to accurately estimate the cost to the vendor.

Thus, all agents must have this capability, and their behavior must be one of this family of behaviors. We can write this as an update rule. This becomes

$$D_{i+1} = D_i \left(1 - \frac{\gamma}{\alpha}(f_i - c_{c_i})\right). \tag{1.31}$$

This equation underscores the idea that the demand will remain constant when the price is near the cost. However, as the vendors will constantly be trying to increase the price, and the consumers will be reacting to increases, the actual average price will be greater than the cost to vendors. It is worth noting, of course, that in the real world, this cost is replaced by a very poorly defined notion of the "value" of an object. Since consumers have no idea, in general, how much a specific object actually costs in real terms, they must guess about it's value. However, despite this ignorance-driven inflation, the prices, once equilibrated, must respond to the same type of force.

In the next section, we describe our simulation and the behaviors of the agents carrying out repeated cycles of interactions between consumers and vendors. We generate a family of behaviors parametrized by a small number of parameters. Some values of the parameters generate behaviors that obey the requirements of (1.29) and some do not. We explore the effects of these parameters and demonstrate that they yield the expected global behaviors.

4. Simulation Design

We examine our theoretical results using a computer simulation that centers around the interactions between two types of agents: consumers and vendors. Our simulation functions by creating repeated interactions between the consumers and vendors as they learn and react to certain situations [19]. Vendors have commodities to sell, and are designed to maximize profit. Consumers purchase commodities from vendors using money provided to them by jobs, and attempt to maximize consumption. The simulation proceeds by repeated "sessions" during which consumers visit vendors, evaluate what the vendors have to offer, and decide whether or not to buy. Vendors respond to changes in their products' marketability by changing prices in an attempt to increase their profit.

In our simulation, many details come into play. Both consumers and vendors have memory which help them decide on things such as which of the other class of agents to do business with, how to change prices, etc., and how to respond to current offerings [10]. In the coming subsections, we explain these in detail, including motivating assumptions borrowed from economic theory. Our goal is to test our method of designing agents whose interactions produce a desired global goal, namely the control of the average price of a commodity. We describe, in addition to the agents' designs, the tools used to evaluate the function of the system.

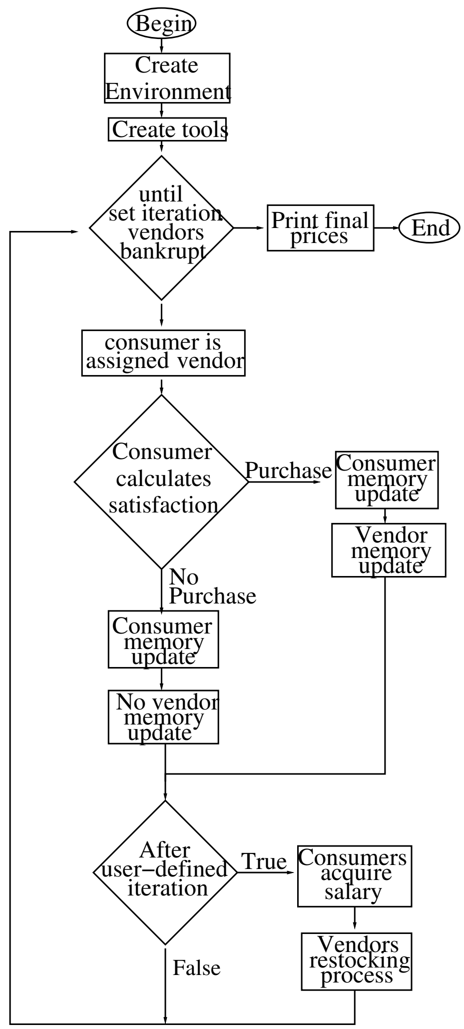


Figure 4.1: This is a general flowchart of ABES

Vendors

As soon as ABES is executed, the products are assigned a random cost. Each vendor sells a single commodity, and so must assign and manage the price of the single commodity. Each vendor calculates its own minimum price. Initially, the price is set at twice the cost to the vendor. All the profits made from a completed exchange is directly added to the vendor's bank account, the total amount of money that the vendor has. The vendor will restock its inventory when the number of products it holds reaches a user defined number if there is enough money in the bank to purchase more products. If the vendor fails to restock using the amount of money in the bank, then that vendor is considered bankrupt and is removed from the pool of vendors. As a result, the bankrupt vendor no longer participates in the interactions between vendors any consumer.

Each vendor's goal is to maximize its profit by any means. After a user-defined number of iterations, if the vendor has made more profit than it did in the previous period, the prices of the vendor's product are incremented by a constant, user defined percentage of the product's cost. This price update rule comes from the assumption that vendors will expect the same number (or nearly the same number) of products to sell the next period. A slight increase of price will increase the total profit. Conversely, if the vendor has made less profit, it reduces its prices by the same percentage. This behavior of decreasing the price derives from the assumption that the vendor will sell more the next period by slightly decreasing the price. This should increase the total profit.

Consumers

Each consumer interacts with its vendor in the same way: the consumer buys from the vendor if all of the conditions are met each time the consumer randomly chooses a vendor to buy the commodity from. We assume the commodity is something the consumer eventually must buy, like water. If the consumer waits long enough, it will be forced by necessity to purchase the commodity at any price. If the consumer has enough money, the item is in stock, the vendor is not bankrupt, and the consumer is "satisfied" with the product, the consumer will purchase the product. The consumer's satisfaction with the vendor's products is represented by a number from 0 to 100, and is affected by the length of time since the last purchase, the consumer's memory of the prices, and the vendor's profit margin. Along with the information in the consumer's memory, the consumer calculates its satisfaction toward the product. A random number from 0 to 100 is generated, and if the calculated satisfaction is higher than the generated number, then the consumer will be considered "satisfied" enough to buy the product. Thus, the higher the satisfaction of the consumer is, the more likely the consumer is to purchase an item from the vendor. Each consumer's cache of money is incremented by a user defined salary after some number of iterations, and decremented by the amount of each purchase.

The goal of the consumer in our simulation is maximize consumption at the lowest price and at the highest possible satisfaction. Our consumers are sensitive to the vendors' profit margin and will not purchase a product if the profit margin is too large. Whenever a vendor increases its price, consumer satisfaction decreases. As a result, consumers are less likely to purchase from the vendor. At some point in the simulation this will so aversely affect consumer satisfaction that very few of them will purchase the commodity. Once consumers cease purchasing, vendors react to a decrease in their income. Vendors, in turn, have no choice but to lower their price. Once the price has been lowered sufficiently, satisfaction returns to a high enough level for consumers to begin buying again. This consumer behavior keeps the vendors from constantly increasing their price and will result in a stablized price. However, as we will see in the next section, there are strict limits on even this behavior which yield control on global price levels.

In our simulation, we model the consumer satisfaction as

$$S = smax[1 - (\frac{1}{e^{t-[\alpha(profit)+(price-\gamma(pricemem))]}})] \qquad (1.32)$$

Here S is the satisfaction, α is a constant that controls the consumer's aversion to profit margin, and γ is a constant that affects competition among vendors. Both of these variables can be initially assigned different values. Profit is the amount of money the vendors make after an exchange is complete. Price is the current price of the commodity and pricemem is the running average of the prices paid by the individual consumer during the last several interactions for the same commodity. The higher the exponent value, higher the satisfaction. Clearly, changing the value of α will alter the consumer's sensitivity toward the profit. Likewise, γ affects the consumer's sensitivity to prices much higher than those recently paid. This indirectly affects competition between vendors.

5. Simulation data

In section 2, we examined the theoretical basis for the design of consumer agents which, we expect, are capable of causing the "invisible hand of the market" to appear, limiting the prices of commodities. Section 3 described our simulation. This simulation consisted of two kinds of agents – consumers and vendors. The two types of agents interact with each other, and have conflicting goals. Moreover, the consumers have a limitation that they must have the commodity that is being sold, eventually. Such a commodity might be like water. The consumer agents have the limitation that the longer they go without the commodity in question, the more they're willing to tolerate to get it. As a result, there is potential for price gouging, leading to runaway prices.

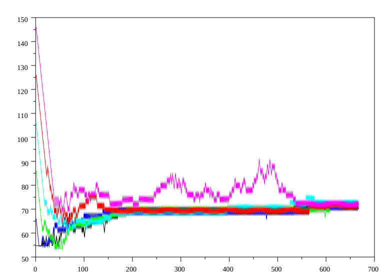
In this section, we examine the behavior of the system under the action of the consumer agents. The agents' behaviors are controlled by the equation (1.32). In this equation, there are two main parameters, α and

γ. By changing the values of these parameters, we can produce differing agents behaviors. Some of these behaviors satisfy equation (1.29) and some don't. We shall see that the desired outcome is achieved when equation (1.29) is satisfied.

The effects of γ

In equation (1.32), we have two parameters, γ and α. γ primarily controls the effect of a high price with respect to previous experienced prices. A high value of γ indicates a high sensitivity to higher prices while a low γ value indicates little or no effect. The overall effect is akin to competition between individual vendors. With a high value of γ, the prices tend to stabilize near those of the agent with the lowest prices, while lower values do not tend to reinforce this.

We can understand this in terms of equation (1.29). The demand does not change on the left hand side if the prices are all the same. However, the first term on the right hand side is large enough that the equation does not hold. As a result, the price does not reduce, but rather stays constant once all vendors have synchronized their prices. The situation is depicted in Figure 5.1.



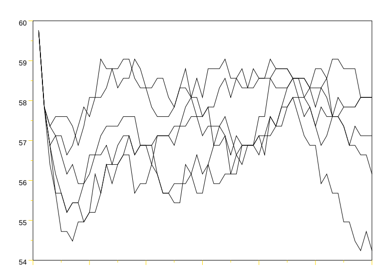


Figure 5.1: With a high value of γ the prices are limited to the lowest price of all consumers. However, if this lowest price is itself high, the prices will not rebound, as can be seen in these figures.

While γ tends to cause competition among vendors, it is not strong enough to cause the control of runaway prices. Consumers are generally stuck with the lowest of the vendor prices. We have seen that the failure of the system to satisfy the theoretical conditions translates to a failure of overall system to produce the desired property. If all of the vendors tend to increase their prices at the same rate (colluding or not), the effect on equation (1.32) is negligible, and so the condition is still not met. In this case, we can have runaway prices as well. This situation is depicted in Figure 5.2.

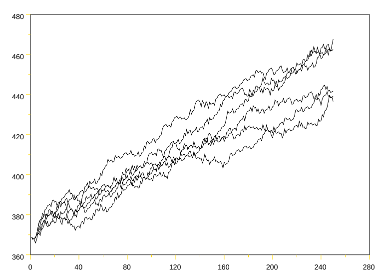


Figure 5.2: Even with a high value of γ the prices can increase unboundedly if vendors continually increase their prices at similar rates.

Adding in α

It is clear that the competition between vendors is enough to hold most prices equal, but not strong enough to stabilize the cost of the commodities at prices that reflect their actual cost. This is interesting for a great many reasons, not the least of which is that this seems to contradict the "invisible hand of the market" that underlies much of economic theory. Clearly, more than simple competition is required to restore this property.

Satisfying equation (1.29) requires that another, stronger term become active. In equation (1.32), the parameter α controls the sensitivity of the consumer to the profit margin that the vendor is receiving. Very high values for α make the consumer intolerant of even small amounts of profit. On the other hand, small values for α make the consumer very tolerant of profits. We examine the effect of this.

The immediate effect is that the decrease in demand as a function of time becomes inextricably tied to the rate of increase of profit. If the profit increases, then the demand decreases. If α is high enough, the decrease exceeds any increase in overall profit associated with increasing the price. As a result, the condition in equation (1.29) is satisfied, and the price is controlled. The situation is depicted in Figure 5.3.

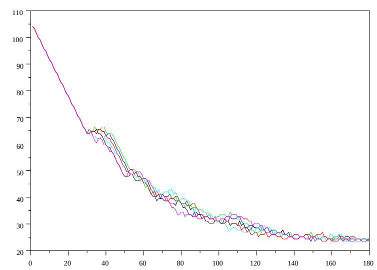


Figure 5.3: With γ high or low, a high value of α is sufficient to control the prices of the commodity. This is expected due to equation (1.29), and confirmed in this simulation.

Note that in subsection 5.1, we kept α low, and the simulation had a global price increase over time. Only adding this very strong affector seems to hold prices low over time. The effect of this design element is so strong that it can take hold long after the price increase has begun, as illustrated in Figure 5.4.

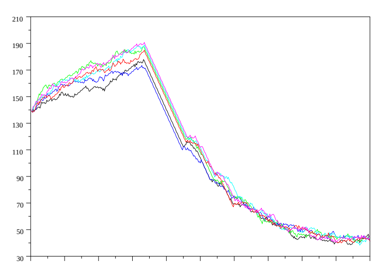


Figure 5.4: If α is initially small, and γ high, the system exhibits slow price increase over time. However, if α is "turned on" at some later time, the system recovers its low-price configuration.

Examining (1.29)

One of the main guiding principles of this study has been the need to satisfy equation (1.29) in generating the consumer behavior. The reason is that we showed in section 2 that if (1.29) is satisfied, then the behavior will lead to the desired global behavior. We now examine how closely our simulations adhere to this equation in generating the behaviors that limit commodity prices.

We can graph both sides of equation (1.29) as a function of the simulation iteration number. When we do this for both cases in which the price is controlled and cases in which the price is not controlled, we find that when a vast majority of the data follows equation (1.29), the prices are controlled. If this is not obeyed, even a bit more than intermittently, the prices are not controlled.

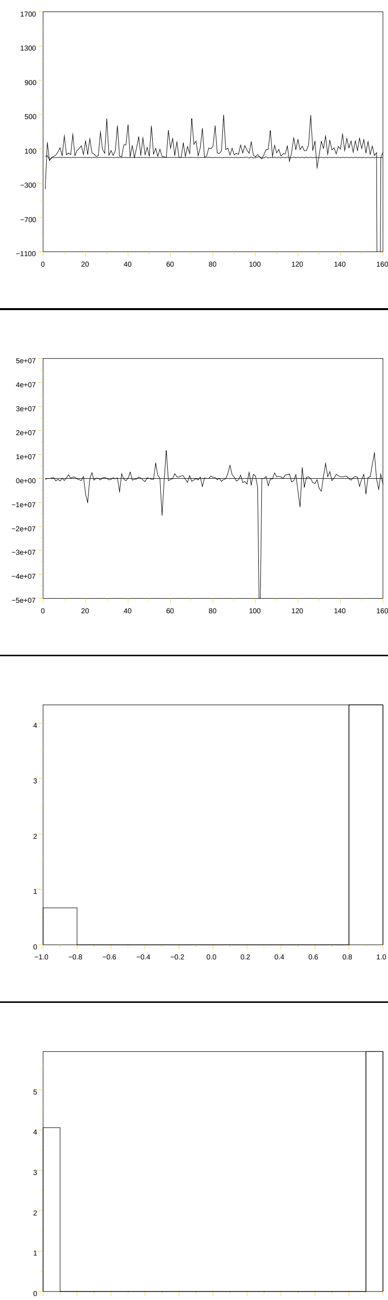


Figure 5.5: These graphs illustrate the values of equation (1.29) as the simulation is run (top two) and histogram the number of times it is obeyed and not obeyed (bottom two).

We find that when the equation is obeyed most of the time (first and third), the prices are controlled. However, when the equation is obeyed considerably less than all the time (second and fourth), the prices are not controlled. This supports our theoretical derivation of this condition.

6. Perturbations of the economic swarm

In the previous sections, we saw that we are able to stabilize the prices using behaviors mediated by two behavioral parameters, alpha and gamma. Behaviors generated when alpha and gamma are relatively large tend to satisfy equation (1.29). We find that, as expected, the resultant behaviors cause the average prices to be limited. Other parameters, however, do not affect the agent behavior in a way that exerts much influence over the swarm's ability to achieve the desired global goal. As a result, these two parameters seem to be critical in determining whether or not the swarm will yield the global goal.

In this section, we examine a perturbation to the initial swarm developed in earlier sections. We examine the income effect in heterogeneous swarms. This refers to the effect of consumers having extra money to spend. Consumers tend to increase their consumption as their income increases. The question is whether or not this has an effect on the global behavior. If there is an effect on the global outcome, we examine how much of the swarm is required to see a change in the global outcome.

The income effect implies that people with the means will often purchase more items than those with lesser means. In previous simulations all consumers received identical wages and therefore exhibited equal purchasing power. In real systems, this is obviously not going to be the case because income differs depending on the consumers' professions. So we create a heterogeneous swarm by assigning different incomes to each agents. Moreover, we also examine the influence of the income effect applied to the swarm.

The income effect allows some consumers to purchase greater quantities of commodities than other consumers. The increase in purchasing power may create a leeway for vendors to subtly raise price. Since consumers are now capable of purchasing much larger quantities, the vendors might find it possible to increase their prices without losing profit, and therefore choose to leave the prices higher. Such an effect might destabilize the swarm, increasing prices over long period of time. We are interested in determining whether the swarm's design requirement from equation (1.29) is strong enough to offset the income effect in a heterogeneous swarm.

In order to examine the income effect in our swarm, we have extended ABES to execute the income effect with same parameters that produced the prior global goal. In the previous version of ABES, each consumer had only one opportunity to purchase a commodity per iteration regardless of his income. This limitation has been replaced with multiple purchase opportunities per consumers per iteration. Those consumers with higher incomes may purchase more per iteration than those with lower incomes. The amount that a consumer may purchase is proportional to his income. This simple modification emulates the income effect in the simulation leaving all other details unchanged.

As indicated above, the income effect may provide vendors a leeway to subtly inflate prices. As before, an increase in profits spurs vendors to raise prices. Wealthier consumer may still purchase at higher prices; the system might then produce various outcomes. Of interest to us is whether or not the effect of a greater income is overcome by the swarm's behavior as indicated by our prior analysis. Intuitively, one could make the argument that with more money available, prices might still be able to stabilize, but would tend to stabilize above the prior prices. Our analysis, however, indicates that as long as (1.29) is satisfied, the prices will not only remain stable, but they will stabilize at the same price.

In our simulations, we implement the income effect by giving consumers the opportunity to purchase multiple commodities in a single cycle. There are two types of interactions that cause the income effect. In the first interaction, consumers buy same commodities from multiple vendors; in the second interaction, consumers buy large numbers of the same commodity from a single vendor. The first type of interaction may occur, for example, when consumers purchase clothes. Many consumers purchase clothes from a wide range of clothing vendors, though they may be classified as interchangable products. The second interaction, however, is mostly likely to occur when consumers purchase consumables such as food or water. Consumers may purchase large quantities at once from a single vendor as there is no real reason to purchase it from many vendors at once.

Effect	Mean
First Income Effect	$55,629.53 \pm 2,664.39$
Second Income Effect	$82,282.64 \pm 15,092.99$
No Income Effect	$35,649.37 \pm 25,956.33$

Table 6.1: As a result of the income effects, consumers purchase and consume significantly more of the commodity than simulations without the income effects.

As indicated in the Table 6.1, the income effect does result in an increase of purchasing power. The number of products purchased also increases. According to traditional economic analysis, this increase in demand should lead to an increase in the price of the commodity.

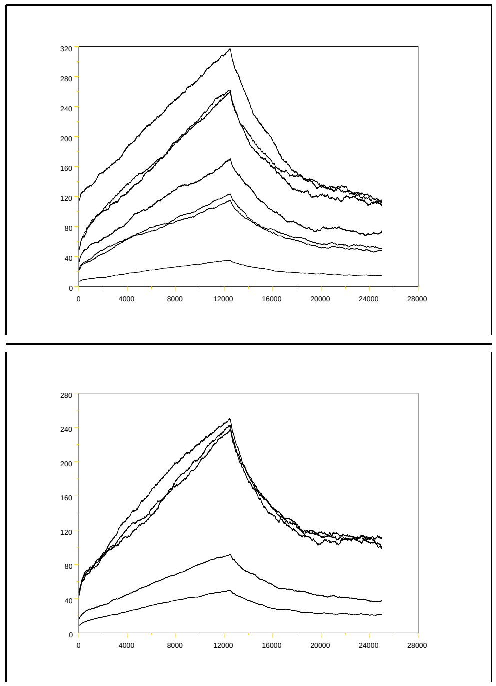


Figure 6.1: Despite the income effect, with α high, the price stabilization takes place. In these graphs, the price of one commodity sharply drops when α is increased despite the strong income effect of the agents with greater salaries.

However, as we can see in Graph 6.1 this is not the case. Despite the increase in the quantities of commodities purchased, however, the prices remain controlled. Initially, the agents are not cognizant of the profits (α is low). However, when they begin to take profit into account in their decisions, the prices drop dramatically. In fact, not only are the prices controlled, but they return to the same levels obtained when the income effect was not part of the simulation.

These data may be interpreted as indicating that the income effect, though changing the nature of the agents and increasing the consumption, does not affect the basic decision-making process that provides vendors with greater profits when prices are controlled. In terms of the consumer behavioral model, the parameter most responsible for this effect is alpha. Alpha mediates the sensitivity of consumers to unnecessary increases in price. When consumers are very sensitive to profit margin, vendors cannot overprice commodities. This local behavioral attribute is strong enough to stabilize prices independent of the volume of sales.

This data is significant because it indicates that, although one effect might be expected from a change in the conditions of the swarm's agents, the design of the swarm can be created so as to provide the opposite effect. This indicates a capability for the design of economic swarms and interactions that has heretofore been impossible.

7. Discussion and conclusion

Designing swarms of agents is a very tricky business, owing to the nonlinear interactions of the various agents. As with all complex systems, swarms of a particular design might have a particular global behavior, but swarms with a very slight difference in behavior may have completely different global behaviors. As a result, predictive design has largely been avoided in the swarm literature.

In this chapter, we've explored a method of swarm design in which a specific global swarm behavior is developed prior to the design of the agents. The desired behavior, it has been shown, can be made to order once a set of requirements for agent design is worked out which will mathematically guarantee that the swarm accomplishes the task [6]. Mathematical guarantee, which has eluded swarm researchers previously, is achieved by utilizing the global goal written in terms of the senses and actuators that the agents can be expected to have access to. Once the swarm condition has been met, the global goal may be achieved with agents meeting this condition.

It is interesting to note that this method of designing swarms is similar in form and function to the design of mechanical systems using the Lagrangian method. The power of this method lies in the ability of the engineer to create one or more properties whose numerical values are unique to the state that the system is in. The engineer, then, needs only chart a path through the allowed phase space of the system to the final desired value, hopefully utilizing behaviors which individual agents can accomplish on their own, with or without guidance from a central controller. The method can be applied to single properties or to vectors of properties, provided that the desired vector is well-defined in the same way a single property might be. We believe that the method is so powerful, in fact, that we now coin a term for this method: The Hamiltonian Method of Swarm Design.

This study, which examines the design of an agent based economic system, has demonstrated that in such systems, the achievement of global goals is possible when specific agent traits are required of the agents. It is interesting that such systems can exhibit control that typically comes from "the invisible hand of the market" or from a command economy. In fact, we have unmasked the "invisible hand of the market" in this study, revealing not only where it comes from but under what conditions it functions. It is interesting to ask, in light of the new method of controlling these swarms, what other economic indicators, trends, etc. can be commanded by the agents within the system.

Another interesting aspect of this study is just how fragile the system seems to be in terms of destabilizing under the improper behavior of one or a few agents. As we have seen in section 4, when the inequality (1.29) is not obeyed, even a little, the prices become uncontrolled. It is interesting, then, to ask whether or not this system is stable in the sense that a few agents do not have the ability to drive the system into this uncontrolled region. This may give us insight into some of the interesting trends seen in recent years in economic systems including overvaluing of various commodities including .com stocks and housing prices. More research on this is clearly indicated.

One unexpected result of the current research is that we seem to have discovered that the income effect alone is not sufficient to raise prices of commodities. This unintutive result may be seen from the stability of prices in the simulations that utilized the income effect. This would seem to be in contradiction to traditional economic theory, which implies that prices would tend to rise. The stability may be seen to be caused by the engineered behaviors of the agents, rather than other market forces.

The current work also underscores the vast power that is enjoyed but not normally readily understood by consumers. It is clear that if consumers actually behaved in the way that our agents behave, the resulting economic system would be much more stable, in terms of commodity prices. The fact that this is now known might be used by policy makers in search of methods of stabilizing economies. With a properly implemented educational system, the public could bring its vast power to bear on economies without the need for governmental intervention or regulation. Whether or not people will actually adopt these behavioral norms is an entirely different question, though it is worth noting that the current method of swarm design brings this possibility to light.

In the future, we intend to apply this method to swarms of greater complexity than this one. We expect that this method of not only swarm design, but complex system design, may be applied to a large number of different systems including, but not limited to, systems of autonomous mechanical agents, computing systems, economic systems, and social systems. While some of this research is currently under way, we expect that the exploration of all fields to which this methodology might be applied will reveal an extraordinarily vast scope. Moreover, we expect that an extension to this work will be able to solve the problem originally posed ten years ago which led us to these results: "Is it possible that the global specification of a problem is enough to yield the basic requirements of the solution including all actuators, sensors, processing, and other capabilities of agents in the solution?" We believe the answer is yes.

Notes

1 JISAN RESEARCH INSTITUTE EMAIL: SKAZADI@JISAN.ORG TO WHOM CORRESPONDENCES SHOULD BE ADDRESSED.

2 JISAN RESEARCH INSTITUTE EMAIL: JOHN@JISAN.ORG

3 (Latin for “all other things unchanged”)

4 The radiative emmissions of the measuring device are likely to be much less important in determining the temperature of the device than internal processes. Thus, the effect of these emmissions is assumed to be negligible.

5 This can be rigorously defined as follows:

$$\frac{\partial S_i|_{(t>t_0)}}{\partial S_j|_{(t=t_0)}} = \lim_{\delta S_j \to 0} \frac{S_i\left(t, S_j + \delta S_j\right) - S_i\left(t, S_j\right)}{\delta S_j} \neq 0 \tag{1.33}$$

for any time $t > t_0$.

6 In the case that there is only one behavior, this is simplified to

$$b_j = \sum_{l}^{N_A} C_j^l(p_s^l, in_s^l) AB^l(m_s^l, in_s^l, s_s^l). \tag{1.34}$$

In the additional case that there is no memory, the expression simplifies further to

$$b_j = \sum_{l}^{N_A} C_j^l(p_s^l, in_s^l) AB^l(in_s^l, s_s^l). \tag{1.35}$$

This is the equation for uniform reactive agent swarms.

References

E. Bonabeau, M. Dorigo, and G. Theraluz. Swarm IntelligenceNew York: Oxford University Press, 1999.

M. Chang, J. Harrington, Jr. Agent-Based Models of Organizations. Handbook of Computational Economics II: Agent-Based Conputational Economics. March 24, 2005.

S. Das. On Agent-Based Modeling of Complex Systems: Learning and Bounded Rationality. Department of Computer Science and Engineering. La Jolla, CA, 92093-0404.

M. Fisher, C. Ghidini. The ABC of Rational Agent Modeling. Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems: Part 2, 849 - 856, Bologna, Italy, 2002.

C. Hommes. Heterogeneous Agent Models in Economics and Finance. Department of Quantitative Economics, University of Amsterdam, March 2005.

K.Judd, F. Kubler, K. Schmudders. Computational Methods for Dynamic Equilibria with Heterogeneous Agents. World Congress, Cambridge University Press, pp. 243 - 290, 2003.

S. Kazadi. Swarm Engineering. Ph.D. Thesis, California Institute of Technology, 2000.

S. Kazadi. The Genesis of Swarm Engineering. Proceedings of the SCI2003 Conference, Special Session on Swarm Engineering, Orlando, Florida, USA, 2003.

S. Kazadi. On the Development of a Swarm Engineering Methodology. Proceedings of IEEE Conference on Systems, Man, and Cybernetics, Waikoloa, Hawaii, USA, pp. 1423 - 1428, October 2005.

S. Kazadi, P. Kim, J. S. Lee, and J. Lee. Swarm Economics. World Congress on Engineering and Computer Science 2007, San Francisco, California, USA, October 24-26, pp. 602-610, 2007.

S. Kazadi, J. R. Lee, J. Lee. Artificial Physics, Swarm Engineering, and the Hamiltonian Method. World Congress on Engineering and Computer Science 2007, San Francisco, California, USA, October 24-26, pp. 623-632, 2007.

S. Kazadi, J. Chang, M. Hwang, D. Lee, and R. Lee. Hijackability in swarms, in prep., 2008.

G. Klein, A Fallah-Seghrouchni, P. Taillibert. An Agent-Based Programming Method. Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems: Part 1, 4 - 7, Bologna, Italy, 2002.

H. Knublauch. Extreme Programming of Multi-Agent Systems. Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems: Part 2, 704 - 711, Bologna, Italy, 2002.

B. LeBaron. Agent Based Computational Finance. Brandeis University, April 21st, 2005.

P. Maes. How to Do the Right Thing. Connection Science Journal, 1(3), 1989

J. Mackie-Mason, M. Weldman. Automated Markets and Trading Agents. University of Michigan, Ann Arbor, MI 48109, USA.

R. Marks. Market Design Using Agent-Based Models. The Universities of Sydney and New South Wales Sydney. May 17, 2005.

P. Massonet, Y. Deville, C. Neve. From AOSE Methodology to Agent Implementation. Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems: Part 1, 27 - 34, Bologna, Italy, 2002.

S. Mellouli, G. Mineau, D. Pascot. The Integrated Modeling of Multi-Agent Systems and Their Environment. Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems. Bologna, Italy, 2002.

P. Reitsma, P. Stone, J. Csirik, M. Littman. Randomized Strategic Demand Reduction - Getting More by Asking Less. Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems: Part 1, 162 -163, Bologna, Italy, 2002.

L. Said, T. Bouron, A. Drogoul. Agent-Based Interaction Analysis of Consumer Behavior. Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems: Part 1, 184 - 190, Bologna, Italy, 2002.

Spears, William M. and Diana F. Gordon (1999). Using Artificial Physics to Control Agents.IEEE International Conference on Information, Intelligence, and Systems, November, 1999.

J. Stiglitz. Economics, W. W. Norton & Company: Inc, New York City, NY, pp. 214 - 218, 2002.

L. Tesfatsion. Agent-Based Computational Economics: A Constructive Approach to Economic Theory. Economics Department, Iowa Sate University, Ames, IA 50011-1070.

K. Tumer, A. Ayogino, D. Wolpert. Learning Sequence of Actions in Collectives of Autonomous Agents. Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems: Part 1, 378 - 385, Bologna, Italy, 2002.

Winfield, A., Sa, J., Gago, M.C., Dixon, C., Fisher, M. On Formal Specification of Emergent Behaviours in Swarm Robotics Systems. Int. J. Advanced Robotic Systems, 2(4),363-370, 2005.

S. Wolfram. A New Kind of Science. Champaigne: Wolfram Press, 2002.

G. Zhong, K. Takahashi, S. Amamiya. KODAMA Project: From Design to Implementation of a Distributed Multi-Agent System. Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems. Bologna, Italy, 2002.