Computational Economics **12:** 1–24, 1998.

The WALRAS Algorithm: A Convergent Distributed Implementation of General Equilibrium Outcomes

JOHN Q. CHENG* and MICHAEL P. WELLMAN**
University of Michigan, Ann Arbor, MI 48109, USA

Abstract. The WALRAS algorithm calculates competitive equilibria via a distributed tatonnement-like process, in which agents submit single-good demand functions to market-clearing auctions. The algorithm is asynchronous and decentralized with respect to both agents and markets, making it suitable for distributed implementation. We present a formal description of this algorithm, and prove that it converges under the standard assumption of gross substitutability. We relate our results to the literature on general equilibrium stability and some more recent work on decentralized algorithms. We present some experimental results as well, particularly for cases where the assumptions required to guarantee convergence do not hold. Finally, we consider some extensions and generalizations to the WALRAS algorithm.

Key words: tatonnement, distributed computation, progressive equilibration.

1. Distributed Implementation of General Equilibria

1.1. Tatonnement

When a collection of interconnected markets achieves a perfect balance of supply and demand with respect to the maximizing behaviors of self-interested economic agents, we say that the economy is in *general equilibrium*. This idea of a general price equilibrium was originally articulated by Léon Walras (1954), who proposed that such equilibria could be realized through a price-adjustment process he called *tatonnement* (sometimes translated as 'groping', or 'tentative proceedings'), in which agents (that is, consumers and producers) respond to price signals for the individual goods. The agents' interactions are coordinated by a central 'auctioneer' who adjusts the general price levels toward a general balance, announcing interim prices to elicit responses from the agents.

More specifically, Walras envisioned the market clearing process working as follows.[‡] Starting with a set of prices arbitrarily given, the excess demand in each market may be positive, zero, or negative. For an arbitrary ordering of these markets, take the first and adjust the price so that supply and demand are equal, given all other prices. Of course, the change in the first price will normally change excess

[*] Department of Economics. e-mail: *jcheng@umich.edu*

[**] Department of Electrical Engineering and Computer Science. e-mail: *wellman@umich.edu*

[‡] See Lesson 12 of the original source (Walras 1954), or, for a more accessible exposition, Arrow and Hahn's account (1971, Chapter 1).

demand in all other markets. Next consider the second market and likewise adjust its price to clear; then repeat the process for all remaining markets. At the end of each round, only the last market is guaranteed to be in equilibrium, since a change of price in some later markets will normally destroy the equilibria established in previous markets. But Walras argued that the change in a good's own price will have a more significant impact on its excess demand than the change in other goods' prices. The own price adjustment goes directly to zero excess demand, whereas the indirect influences of other price changes may increase or decrease demand for the good, and may even cancel each other out. Hence, Walras argued, it is probable that after each round, the prices are closer to equilibrium than before. Eventually, in this story, all markets will clear.

Walras's intuition is correct only in particular circumstances, as a series of studies in the century following his original speculation have told us. In an early investigation of the problem, Hicks (1948) observed that the tendency toward equilibrium depends on the shapes of demand functions near the equilibrium point. Samuelson (1947) provided the first comprehensive mathematical formulation of the price-adjustment process and associated notion of equilibrium *stability*. However, this formulation differs from Walras's original vision in several respects. In Samuelson's now-standard version of tatonnement, competitive agents receive a price signal, and report their excess demands at these prices to the central auctioneer. The auctioneer then adjusts the prices *incrementally* in proportion to the magnitude of excess demands, and announces the new incrementally adjusted price level. In each round, agents recalculate their excess demands upon receiving the newly adjusted price signal, and report these to the central auctioneer. The process continues until the prices finally converge to an equilibrium. (Often this process is formalized as a continuous rather than a discrete adjustment, so that it can be described by differential equations and Lyapunov methods applied to prove stability results.)

This particular formulation of tatonnement, along with some minor variants, was extensively studied in the 1950s and 1960s (Arrow and Hahn, 1971). The results most relevant in our computational context are those on global stability. Of particular importance is the demonstration by Arrow et al. (1959) that *gross substitutability* (GS) is sufficient for a tatonnement process to be globally stable. Gross substitutability holds when the aggregate demand for each good is nondecreasing in the prices of other goods. Under this condition, the equilibrium is also unique.[⋆] A somewhat more general sufficient condition for global stability is that the aggregate excess demand functions satisfy the *weak axiom of revealed preference* (WARP) (see the aforementioned textbooks, or Takayama (1985), for example).

[⋆] For a broader discussion of existence, uniqueness, and stability of competitive equilibria, consult a good microeconomic theory textbook (Malinvaud 1985; Varian 1992; Mas-Colell, Whinston & Green, 1995).

1.2. Non-tatonnement Processes

Many economists (Rust (1996) is a recent example) criticize the tatonnement explanation of how to reach general equilibrium because it does not allow agents to trade until an equilibrium has been reached. In an alternate approach, the so-called *non-tatonnement process*, agents are allowed to trade before the economy has reached equilibrium (although irreversible consumption and production of goods are typically prohibited). Fisher (1983) provides an extensive discussion of non-tatonnement processes. Takayama (1985) presents Edgeworth and Hahn processes as two examples of non-tatonnement approaches.

The obvious rationale for allowing agents to trade out of equilibrium is simply that waiting an indefinite time to reach equilibrium is unreasonable, and that even within definite intervals, the world changes and thus the time delay is consequential. In the non-tatonnement case, an agent's excess demand function is typically a function of time, and the final price that clears all markets is generally path dependent. Therefore, the correspondence from initial endowments to final allocation is not as strong as that in the tatonnement process. One nice feature of the non-tatonnement processes is that they are usually more stable than the tatonnement processes, in part due to the fact that intermediate trading never decreases agents' utilities.

In defining non-tatonnement processes, it is important to specify the rules that allow the disequilibrium trading to occur. For the Edgeworth process, agents are simply assumed to trade whenever it is possible to improve their utilities. For the Hahn process, the short side of the market always gets satisfied: if there is excess demand of a certain good, then all sellers can sell but not all buyers can buy. Many other protocols are possible, as realized, for example, in particular artificial economies (Clearwater, 1995; Epstein and Axtell, 1996; Yamaki, Wellman and Ishida, 1996), or general dynamic exchange mechanisms such as continuous double auctions (Friedman and Rust, 1993).

If one is interested in general equilibrium for purposes of *implementing* decentralized resource allocation, then the objection that tatonnement is not an accurate description of observed price-formation processes is not crucial. Nevertheless, an ability to transact out of equilibrium can be advantageous if the time required to reach equilibrium is prohibitive. However, these advantages come with some drawbacks. Non-tatonnement processes may be subject to more strategic interactions, as transient prices become consequential to participating agents. This may reduce the modularity of the system, but the promise of more robust stability might prove to be a worthwhile tradeoff. Another serious concern is that irreversible decisions (particularly in production) before prices are in equilibrium may lead to inefficient resource allocations. The *tentative* quality of tatonnement makes it a more conservative, if somewhat less dynamic, approach.

1.3. Distributed Implementation

One important property common to both tatonnement and non-tatonnement processes is that they are amenable to straightforward distributed implementation. Although the traditional tatonnement process is often described as if there were a central auctioneer, the price update is typically separable across goods, and therefore nothing is lost by decentralizing the auctioneering process to this extent.

This is in contrast to other prevalent methods for computing competitive equilibria, namely fixed-point (Scarf, 1973; Shoven, Whalley and Wiegard, 1992) and mathematical programming approaches (e.g., (Zhao and Dafermos, 1991)), which generally require direct examination of the joint price space. Centralization is not necessarily a drawback for the problems to which these algorithms are usually applied: solution of a general-equilibrium model to examine issues such as tax or trade policy (Shoven and Whalley, 1992). In most applied general equilibrium modeling, the system being modeled is distributed, but the analysis itself need not be.

For other applications, however, distribution of the computation is important. Such is the case when modeling the actual price-formation process of an economy, or (as in our case), when the purpose is to solve resource-allocation problems within a distributed computational environment. For these purposes, two levels of decentralization are important. First, at the *auction level* (or more generally, the *institution* level), we would like to decompose the market across commodities. For tatonnement and non-tatonnement processes, this means operating separate markets for each good. Decomposition obviously reduces dimensionality and simplifies each auction's task.

Second, at the *agent level*, we would like to bound the information scope of each agent to its own preferences or technology, and the decision scope of each agent to its own consumption or production. In the framework of *mechanism design*, decentralization constraints of this sort are typically formalized in terms of privacy preservation and informational efficiency (Campbell, 1987; Hurwicz, 1977), and the competitive mechanism is characterized as a maximally distributed mechanism in these terms. Agent-level decentralization generally enhances modularity of the overall system, and is often necessary to reflect an existing authority or information structure.

We might also consider a third level of decentralization–distribution of the auction process for each good. Such distribution might be appropriate when communication costs depend on locality, in which case multiple auctions might mediate exchange in various neighborhoods. If it is possible to arbitrage between the local auctions, then such configurations would have the same equilibria as a single-auction-per-good setup. If the markets are not so closely coupled, then the result may be multiple prices for the same good. An extreme version of this is *bilateral exchange*, where agents engage exclusively in pairwise trades with their neighbors. This approach has been explored extensively in the *Sugarscape* model (Epstein and Axtell, 1996), revealing systematic divergence from competitive equilibrium results.[⋆]

[⋆] Bilateral exchange in Sugarscape is characterized as an *unmediated* mechanism, where no auctioneer per se is involved in the exchange process. We regard the role of the auction to be

In the next section, we describe an approach to distributed resource allocation based on general equilibrium. We have embodied this approach in a system, called WALRAS, that is designed to facilitate the construction and implementation of computational economies based on the competitive mechanism. Our focus in this article is on the underlying algorithm that WALRAS employs to derive general equilibria.

2. The WALRAS Algorithm

The WALRAS system is designed to provide a computational environment based on the basic concepts of general equilibrium theory. The system supports the two standard types of agents, consumers and producers. Consumers are defined by their preferences and endowments, and producers by their technology. Both types are assumed to be rational, competitive agents, maximizing utility or profits subject to feasibility at the given price level.

The system supports these agent types by providing computational building blocks for programming agent behaviors, based on maximization with respect to specified preferences or technologies. At its core the system also provides a distributed bidding protocol and price adjustment procedure, which together comprise the WALRAS *algorithm*. In this section we describe this algorithm in detail, and its connection to some related work.

2.1. Overview

In the WALRAS system the actions taken by consumers and producers are coordinated by auctioneers whose jobs are to clear their respective markets. There is a one-to-one correspondence between auctions and goods.

Upon receiving a randomly chosen initial price vector, each agent computes the demand function for each of its goods of interest. Consumers are interested in any good mentioned in their utility functions (including any with nonzero endowments), and producers are interested in any good mentioned in their technologies. Each demand function specifies the net quantity demanded of a good (which for a net supply is negative) as a function of its price, assuming that the prices for the remaining goods are constant. The agents then send these single-good demand functions – the *bids* – to the respective auctioneers for each good.

Each auction, upon receiving the bids from all agents, computes the clearing price, which is the point at which the aggregate excess demand is zero. The auctions then notify the bidders of the new price. Upon seeing new prices, the consumers and producers compute revised demand functions as necessary based on these new prices, and send these updated bids to the auctions.

that of enforcing the information revelation and allocation properties of the exchange protocol. Implementation of any protocol requires this function, whether or not it is realized in a mediating institution.

This process continues until the prices no longer change (or the changes are within a pre-specified threshold). Then the process terminates, and WALRAS reports the final state of the price vector as the equilibrium.

2.2. The Asynchronous Bidding Protocol

An important feature of the WALRAS bidding process is that it is *asynchronous*. That is, at any particular time, agents are not necessarily bidding on the same goods or computing their demands based on the same state of price information. Rather, each agent maintains its own queue of goods for which to compute revised bids, and processes new price information as it is received from the auctions.

In a distributed computing environment, the asynchrony of agents' bidding behavior is completely unconstrained. In our implementation of WALRAS on a single computational process,[⋆] we realize a more restricted form of asynchrony by introducing randomness into the bidding process. Because it takes time to compute and transmit a bid, simultaneous transmission of all bids by all agents would not be an accurate model. Therefore, in our simulated system we poll the agents, allowing each to submit a random number of bids on each iteration.

Formally, let there be k goods and n agents. At any particular time t, agent i submits some number of bids for a subset of the goods. We assume that the number of new bids in any period is determined by a random draw.[⋆⋆] There is a distinct draw for each agent at each time period. The draws are independent across time and across agents. When an agent does not submit a bid for a good, its demand is simply carried over by the auction from the last period.

If it has a chance to submit, the bid agent i submits to auction j at time t is $x_i^j(p_j, p_{-j}^{t-1})$, a function of p_j, the price of good j, parametrized by p_{-j}^{t-1}, the price of the other goods according to information available to the agent at time $t-1$.

If agent i has not submitted a new bid to auction j for $s_i^j(t)$ periods prior to the current time t, we denote the *effective bid* for this agent, good, and time as $x_i^j(p_j, p_{-j}^{t-s_i^j(t)})$. For the general case where the agent might or might not submit a new bid, we can express this effective bid as

$$\begin{aligned} & x_i^j(p_j, p_{-j}^{t-s_i^j(t)}) \\ & \quad \equiv x_i^j(p_j, p_{-j}^{t-1}) + b_i^j(t)[x_i^j(p_j, p_{-j}^{t-1-s_i^j(t-1)}) - x_i^j(p_j, p_{-j}^{t-1})]. \end{aligned}$$

In this expression, the coefficient $b_i^j(t)$ is an indicator that takes the value 0 if i submits a new bid for good j at t, and 1 otherwise. The first term in the square

[⋆] We have implemented both distributed and single-machine versions of the system. Although our target applications generally run in distributed environments, we maintain the single-machine version to facilitate experimentation with the system and the algorithm.

[⋆⋆] The current implementation of WALRAS chooses with equal probability from the set $\{0, 1, 2\}$. Note that with this setting, each agent submits an average of one bid per cycle. As we see below, the particular distribution is inessential to our analysis.

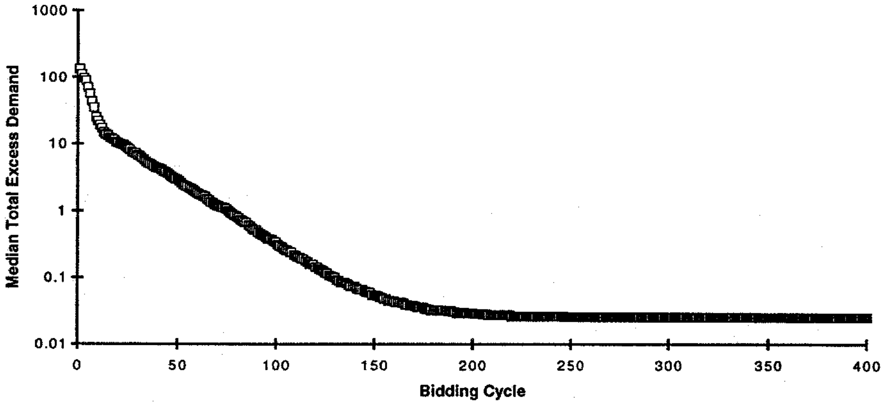

Figure 1. Convergence of the WALRAS algorithm on one hundred randomly generated exchange economies. Note that with seven goods, it takes seven cycles on average for each agent to complete an entire round of bidding.

bracket is the bid that agent i would submit to auction j based on the information $s_i^j(t-1)+1$ periods before t (or $s_i^j(t-1)$ periods before $t-1$). This is the old bid retained by the auction if no update is called for by the system.

Another way to write this is

$$x_i^j(p_j, b_i^j(t)p_{-j}^{t-1-s_i^j(t-1)} + [1-b_i^j(t)]p_{-j}^{t-1}),$$

or simply $x_i^j(p_j, p_{-j}^{t-s_i^j(t)})$, where

$$s_i^j(t) = b_i^j(t)(s_i^j(t-1)+1) + (1-b_i^j(t)).$$

Note that throughout we take one period as the minimum possible delay.

One virtue of introducing asynchrony into the price adjustment process is that it tends to reduce price oscillations caused by simultaneous reactions by agents to the same delayed information (Wellman, 1993). By randomizing the bidding, different agents tend to bid on different goods in a given period. That is, we typically have $b_i^j \neq b_{i'}^j$ for $i \neq i'$, and consequently different delays, $s_i^j(t) \neq s_{i'}^j(t)$.

Asynchrony *among auctions* is a feature that is buried in the continuous version of the tatonnement process. Although it is usually not stressed, in the translation from a difference-equation formalization of tatonnement to a differential formalization, the simultaneity (among auctions) is lost (see Huberman and Glance (1993) for a discussion).

2.3. Experimental Results

We have implemented this algorithm in the WALRAS system and have run it on numerous computational economies. Figure 1 presents some representative experimental results. We ran one hundred randomly generated exchange economies, each with seven goods and seven consumers. The consumers were given randomly generated endowments and preferences represented by the CES (constant elasticity of substitution) utility function,

$$u(x) = \left(\sum_{j=1}^{k} \alpha_j \left(x^j \right)^{\rho} \right)^{\frac{1}{\rho}}. \tag{1}$$

For these experiments we set $\rho = 0.5$ and randomly generated the α_j coefficients from a uniform distribution. Note that these settings ensure that the gross substitutes assumption is satisfied for this example.

Figure 1 plots the median total excess demand, as a function of t, the bidding cycle index. Our measure of total excess demand is the sum of absolute excess demand for each good. (For comparison, note that the economies averaged approximately 220 units of endowed goods.) As we can see from the figure, the excess demand drops precipitously in the first few cycles, as the agents submit their initial bids. Convergence continues rapidly in the early iterations (a constant slope on the log-scale plot), beginning to level off at around 150 cycles.

We have also examined scalability of the algorithm with respect to the numbers of goods and agents (Wellman, 1995). The main experimental evidence seems to suggest that the number of iterations required to reach equilibrium does not rise with the number of agents in the economy. We have also observed manageable increases in convergence time with up to a dozen goods (see Wellman, 1995a; Figure 2).

To calculate the clearing price for any particular market, we have generally used binary search over the aggregate demand function. As pointed out by Ygge and Akkermans (1996), Newton methods could be employed to speed up this computation, by exploiting derivatives of the aggregate demand.

2.4. Relation to Tatonnement and Other Algorithms

The process generated by WALRAS is similar to the tatonnement process, and particularly resemblant of Walras's original vision. In WALRAS, there is one auctioneer for each good's market. Each auctioneer receives information from consumers and producers on one good, and it adjusts the price only for that particular good to reduce the excess demand for it. No auctioneer has to worry about interactions among different markets, such as how the changes in the price of one good will affect the excess demand of another good. It is in this sense that WALRAS is decentralized at the good or auction level. It is of course also decentralized at the agent level, as agents are concerned exclusively with their own consumption or production. Both forms of decentralization are useful (often essential) in practice, providing the two sources of distributivity and modularity in the system (Wellman, 1995).

As we have previously (Wellman, 1993) pointed out, WALRAS differs from the traditional tatonnement procedure in two basic ways:

(1) agents send excess demand *functions* rather than single quantity points at a particular price, and

(2) the auction adjusts individual prices to clear each market, rather than just adjusting the whole price vector incrementally.

In practice, sending a whole curve is not much more complicated than sending just a point, since a curve can be effectively approximated by a number of points. And, as we discuss in Section 5, we can also accommodate hybrid methods, where curves are built up incrementally from a sequence of point bids. The analysis in terms of entire demand functions which we pursue below avoids some complications that occur when dealing with points accumulated from different time periods.

The market-by-market approach employed by WALRAS is called *progressive equilibration* by Dafermos and Nagurney (1989), in which it was applied to a transportation network equilibrium problem. This method was also used in Eydeland and Nagurney (1989), where a collection of non-interacting markets with linear demand and cost structures were studied. These are essentially partial equilibrium models which could be reformulated as quadratic programming problems, as pointed out by the authors. WALRAS uses this method to approach a collection of markets which are inherently interdependent. The progressive equilibration approach also bears some similarity to Hicks's notion of perfect stability (Hicks, 1948), which itself is patterned more after Walras's tatonnement concept than are more modern formalizations.

Reiter and Simon (1992) studied a decentralized approach for the case where the equations that describe the equilibrium are linear. For this special case, their method requires a particularly small message space.

Murphy and Mudrageda (1994) investigated the problem of decomposing large-scale equilibrium models composed of linear-program submodels. Their work focuses on issues arising from models made up of heterogeneous components. In subsequent work (Murphy and Mudrageda, 1995), they describe a method of constructing a series of approximate supply curves that, when combined with exact demand models determines a sequence of points that converge to the equilibrium.

Bertsekas and Tsitsiklis (1989) have studied asynchronous dynamic programming and optimization in a more general setting. Using a model of asynchrony based on parallel processors subject to random delays, the authors have identified conditions under which convergence of an asynchronous algorithm can be derived as a consequence of the convergence of the synchronous version of the same algorithm. They have applied such results to problems in dynamic programming, network flow, and constrained optimization.

Some other related work is described in previous writings (Wellman, 1993).

3. Convergence of the WALRAS Algorithm

In this section, we show that the price adjustment process generated by WALRAS converges to the unique competitive equilibrium, under conditions of an exchange economy with strictly convex preferences and gross substitutability (GS) between goods. The result here also extends to productive economies. The GS assumption is essential for the proof, although we have found experimentally that convergence is often obtained even when it is violated.

3.1. Basic Setting

Our argument is based on the adaptive-learning framework of Milgrom and Roberts (1991). We start by formulating the price-adjustment process as a game, with auctions as players and agents mechanically following the rules of competitive behavior.

Let the k auctions play a game where each sets the price of its own good, that is, auction j sets p_j. Let p be the k-vector of prices, where each component p_j is restricted (without loss of generality) to the interval $[0, 1]$. Let $Q^j(p)$ be the true aggregate excess demand function for good j, that is, the total demand summed over all n agents. As mentioned above, we assume strictly convex preferences and GS.

We define the payoff function, π_j, for auction j as follows:

$$\pi_j \equiv -|Q_j(p_j, p_{-j})|.$$

We assume $Q_j(p_j, 0) < 0$ for $p_j > 0$, and $Q_j(0, p_{-j}) > 0$ for $p_{-j} > 0$.

With this payoff function, a competitive equilibrium of the n agents is also an equilibrium for this game among the k auctions. By our assumption that all agents' preferences are strictly convex, such a competitive equilibrium exists. Under this same assumption, any equilibrium for the game must have zero excess demand for each good, and so the game equilibria and competitive equilibria of the economy coincide exactly.

Note that the consumers do not enter this game explicitly; they simply behave as price takers and submit their bids accordingly. There is no strategic interaction between these consumer agents and auctions in our game here. The interesting interaction arises in how auctions formulate their *expectations* about other auctions' strategies. Our goal is to prove that if auctions behave exactly according to the rules in WALRAS, their interaction will yield a convergent price adjustment process leading to an equilibrium for the game, and hence a competitive equilibrium for the economy.

3.2. Adaptive Learning

To proceed with the argument, we must first introduce and discuss several useful concepts from Milgrom and Roberts (1991). The following terminology and notation is drawn from that work.

Let S_j be the set of pure strategies of player j, and let $\Delta(S_j)$ denote the set of mixed strategies over S_j. The joint strategy space, S, is the cross product of the player strategy spaces, $\prod_j S_j$.

DEFINITION 1. For $\epsilon > 0$, a strategy $\sigma_j \in S_j$ is *ϵ-dominated* by another strategy $\sigma'_j \in \Delta(S_j)$, if for all $\sigma''_{-j} \in S_{-j}, \pi_j(\sigma_j, \sigma''_{-j}) + \epsilon < \pi_j(\sigma'_j, \sigma''_{-j})$.

DEFINITION 2. Given $T \subseteq S$, let T_j denote the projection of T onto S_j, and let $T_{-j} = \prod_{l \neq j} T_l$. The set $U^{\epsilon}_j(T)$ is the set of pure strategies in S_j that are *not* ϵ-dominated when the other players are limited to pure or mixed strategies over T_{-j}. $U^{\epsilon}(T) = \prod_j U^{\epsilon}_j(T)$.

DEFINITION 3. $U^{\epsilon m}(S)$ is the mth iterate of $U^{\epsilon}(S)$, that is,

$$U^{\epsilon m}(S) = \begin{cases} U^{\epsilon}(S) & m = 0 \\ U^{\epsilon}(U^{\epsilon(m-1)}(S)) & m \geq 1. \end{cases}$$

DEFINITION 4. $U^{0\infty}(S) \equiv \bigcap_{m=1}^{\infty} U^{0m}(S)$ is the *serially undominated strategy set.*

DEFINITION 5. A sequence of *strategy profiles* $\sigma(t)$ is *consistent with adaptive learning* if and only if (iff) the following property holds for all players $j = 1, \ldots, k$:

$$\forall \epsilon > 0, \quad \forall \hat{t}, \quad \exists \bar{t}, \quad \forall t \geq \bar{t}, \quad \sigma_j(t) \in U^{\epsilon}_j(\{\sigma(s) \colon \hat{t} \leq s \leq t\})$$

where

$$\{\sigma(s) \colon \hat{t} \leq s \leq t\} \equiv \prod_{j=1}^{k} \{\sigma_j(s) \colon \hat{t} \leq s \leq t\}$$

This definition roughly means that a sequence $\sigma(t)$ is consistent with adaptive learning if each player can *eventually* justify its choice in terms of other players' *past* play. Milgrom and Roberts (1991) point out that this definition of *adaptive learning* is very inclusive. It includes best-response dynamics, fictitious play, and many other policies.

Now let us take a closer look at auction j's payoff at time t:

$$\begin{aligned} \pi_j(t) &= -\left|Q_j(p^t_j, p^t_{-j})\right| \\ &= -\left|\sum_{i=1}^{n} x^j_i(p^t_j, p^t_{-j})\right| . \end{aligned}$$

From the simultaneous-play nature of our game, auction j does not know p^t_{-j} at time t. Since it does know p^{t-1}_{-j}, one natural way for auction j to play is to use $x^j_i(p^t_j, p^{t-1}_{-j})$ to forecast $x^j_i(p^t_j, p^t_{-j})$. However, since in the WALRAS algorithm agents do not submit new bids every iteration, auction j does not have all the demand information $x^j_i(p^t_j, p^{t-1}_{-j})$ either.

Alternately, we could assume that WALRAS auctions *know* the functional forms $x^j_i(p_j, p_{-j})$, that is, the agents' true demand functions, but not the price vector (p_j, p_{-j}). Then the task for the auctions is to forecast the price vector p^t at time t, and determine their optimal response. (This price vector p^t will be common knowledge only at time $t+1$ or later.)

Before considering the fully asynchronous WALRAS algorithm, let us enumerate some simpler cases:

- Auction j believes that other auctions will play at any time t the vector price $p^{t-s_0}_{-j}$, where s_0 is a constant.
- Auction j believes that other auctions will play at any time t the vector price $p^{t-s_j}_{-j}$, where s_j is a constant for j and $s_j < t$.
- Auction j at any time t uses a stationary probabilistic forecast, for example, at any time t, auction j believes that with probability μ^j_h, other auctions will play $p^{t-s^j_h}_{-j}$, where $\sum_{h=1}^k \mu^j_h = 1$, and $s^j_h < t$ for all h.

These forecasting methods can be shown to induce strategies consistent with adaptive learning. One common feature across all of these algorithms is that although different auctions may use different information, *there is no informational discrepancy among the agents communicating demands to the auctions*. That is, the same τ enters each term $x^j_i(p_j, p^\tau_{-j})$ for all i. This condition is violated by the WALRAS algorithm, which is asynchronous at the agent level.

3.3. Asynchrony: A Limited Case

A very important feature of the WALRAS algorithm is that within each auction, the demand functions collected from different agents might be based on price information from different periods of time. Because the agents do not typically have the opportunity to send their most up-to-date bids to all the auctions, some of the bids that the auctions receive are not based on the updated information. More importantly, bids from different agents do not typically reflect the *same* amount of delay. Put another way, agents behave as if they have discrepancies in price information.

In WALRAS, auction j sets p^t_j, so that

$$\sum_{i=1}^n x^j_i(p^t_j, p^{t-s^j_i(t)}_{-j}) = 0.$$

It is easy to see that different agents (indexed by i) will, in general, have different perceptions of p_{-j}.

The processes generated by WALRAS are indeed consistent with adaptive learning, as we show in Section 3.5. But before considering the full-blown algorithm, we consider a simpler case with only two agents. This allows us to focus on the basic issue of asynchrony, without bringing in all of the details required for the general situation.

Consider a simple economy with k goods, but only two agents. Let the *perceived excess demand* function Z^j be defined by

$$Z^j(p_j, t) \equiv x_1^j(p_j, p_{-j}^{t-1}) + x_2^j(p_j, p_{-j}^{t-2}).$$

Here we allow agents to have different perceptions of p_{-j}. Agent 1's information has a time lag of one, agent 2's a time lag of two. (Neither the number of agents nor these arbitrarily chosen time lags are crucial to our result.)

We would like to show that if each auction sets prices such that perceived excess demand $Z^j(p_j, t)$ is zero, the sequence of vector prices $\{p^t\}$ will converge to the unique equilibrium price vector. The first step, which we accomplish in this section, is to show that the bounds of such a sequence are consistent with adaptive learning.

At time t, auction j selects $\hat{p}_j(t)$ such that

$$x_1^j(\hat{p}_j(t), p_{-j}^{t-1}) + x_2^j(\hat{p}_j(t), p_{-j}^{t-2}) = 0.$$

DEFINITION 6. If y and z are two k-dimensional vectors, then

$$y \vee z \equiv \langle \max(y_1, z_1), \max(y_2, z_2), \ldots, \max(y_k, z_k) \rangle,$$

$$y \wedge z \equiv \langle \min(y_1, z_1), \min(y_2, z_2), \ldots, \min(y_k, z_k) \rangle.$$

We can now specify two additional sequences, $\{a(t)\}$ and $\{b(t)\}$, defined for $t \geqslant 1$.[⋆] For $t = 1$ and $t = 2$, and for all j, let $a_j(t) = p_j^t \vee p_j^{t-1}$ and $b_j(t) = p_j^t \wedge p_j^{t-1}$.

For $t > 2$ and all j, we define $a_j(t)$ and $b_j(t)$ recursively:

$$x_1^j(a_j(t), a_{-j}^H(t-1)) + x_2^j(a_j(t), a_{-j}^H(t-1)) = 0, \tag{2}$$

$$x_1^j(b_j(t), b_{-j}^L(t-1)) + x_2^j(b_j(t), b_{-j}^L(t-1)) = 0, \tag{3}$$

where

$$a_{-j}^H(t-1) \equiv a_{-j}(t-1) \vee a_{-j}(t-2),$$
$$b_{-j}^L(t-1) \equiv b_{-j}(t-1) \wedge b_{-j}(t-2).$$

[⋆] We assume that $t = 0$ is the starting point for series $\{\hat{p}(t)\}$.

Note that $a_j(t)$ is a best response to $a^H_{-j}(t-1)$, and $b_j(t)$ is a best response to $b^L_{-j}(t-1)$. Let us consider the sequence $\{a_j(t)\}$. Although $a^H_{-j}(t-1)$ does not represent the collective actions taken by the other auctions at any time in the past, each component of $a^H_{-j}(t-1)$ is an actual past play by one of the auctions other than j at a particular time in the past. Therefore, the components of $a^H_{-j}(t-1)$ are within the *past plays set* defined by

$$\{a(s) : s < t\} \equiv \prod_{j=1}^{k} \{a_j(s) : s < t\},$$

which is the cartesian product of all players' actual past plays sets. It must be the case that

$$a_j(t) \in U^\epsilon_j(\{a(s) : s < t\}),$$

since otherwise $a_j(t)$ would be an ϵ-dominated strategy, but an ϵ-dominated strategy cannot be a best response to any strategy profile of other players. Therefore we conclude that $\{a_j(t)\}$ as a price-setting strategy sequence is consistent with adaptive learning.

More rigorously, $\{a_j(t)\}$ is consistent with adaptive learning iff

$$\forall \epsilon > 0, \quad \forall \hat{t}, \quad \exists \bar{t}, \quad \forall t \geqslant \bar{t}, \quad a_j(t) \in U^\epsilon_j(\{a(s) : \hat{t} \leqslant s \leqslant t\}).$$

We verify that $\{a_j(t)\}$ indeed satisfies this definition. Fix any $\epsilon > 0$, for any $\hat{t}$, let $\bar{t} = \hat{t} + 2$. By definition (2) $a_j(t)$ is a best response to $a^H_{-j}(t-1)$, which is constructed from components of $a_{-j}(t-1)$ and $a_{-j}(t-2)$. Hence

$$\forall t \geqslant \bar{t} = \hat{t} + 2, a_j(t) \in U^\epsilon_j(\{a(s) \colon t-2 \leqslant s \leqslant t\}) \subset U^\epsilon_j(\{a(s) \colon \hat{t} \leqslant s \leqslant t\}).$$

The last inclusion comes from the fact that $t - 2 \geqslant \bar{t} - 2 = \hat{t}$. Thus by definition, $\{a_j(t)\}$ is consistent with adaptive learning. A similar argument establishes that $\{b_j(t)\}$ is consistent with adaptive learning as well.

Finally, we show that these two price sequences bound the actual price sequence, $\{\hat{p}_j(t)\}$. The argument is by induction.

— For $t = 1$ and $t = 2, b_j(t) \leqslant \hat{p}_j(t) \leqslant a_j(t)$ directly follows their definitions.

— Suppose for $t \leqslant t'$, we have $b_j(t) \leqslant \hat{p}_j(t) \leqslant a_j(t)$. We need to show that $b_j(t'+1) \leqslant \hat{p}_j(t'+1) \leqslant a_j(t'+1)$ follows.

— Since

$$a^H_{-j}(t') \equiv a_{-j}(t') \vee a_{-j}(t'-1) \geqslant p^{t'}_{-j} \vee p^{t'-1}_{-j} \geqslant p^{t'}_{-j},$$

$$a^H_{-j}(t'-1) \equiv a_{-j}(t'-1) \vee a_{-j}(t'-2) \geqslant p^{t'-1}_{-j} \vee p^{t'-2}_{-j} \geqslant p^{t'-1}_{-j},$$

and from the definition of $\hat{p}_j(t'+1)$,

$$x_1^j(\hat{p}_j(t'+1), p_{-j}^{t'}) + x_2^j(\hat{p}_j(t'+1), p_{-j}^{t'-1}) = 0,$$

by gross substitutability, we have

$$x_1^j(\hat{p}_j(t'+1), a_{-j}^H(t')) + x_2^j(\hat{p}_j(t'+1), a_{-j}^H(t'-1)) \geqslant 0.$$

Since $a_j(t'+1)$ satisfies

$$x_1^j(a_j(t'+1), a_{-j}^H(t')) + x_2^j(a_j(t'+1), a_{-j}^H(t'-1)) = 0,$$

gross substitutability again implies

$$\hat{p}_j(t'+1) \leqslant a_j(t'+1).$$

— By a strictly similar argument we obtain

$$\hat{p}_j(t'+1) \geqslant b_j(t'+1).$$

— Therefore for all t and j,

$$b_j(t) \leqslant \hat{p}_j(t) \leqslant a_j(t). \tag{4}$$

3.4. Strategic Complementarity

The assumption of gross substitutability serves three purposes in our argument. First, it guarantees the uniqueness of competitive equilibrium (Varian, 1992). Second, it ensures that the price sequences $\{a_j(t)\}$ and $\{b_j(t)\}$ are consistent with adaptive learning. Third, it establishes that the price-setting game exhibits *strategic complementarity* (Milgrom and Shannon, 1994).

DEFINITION 7. A game exhibits *strategic complementarity* iff for every player j,

(1) the strategy space S_j is a compact lattice,
(2) the payoff function $\pi_j(p_j, p_{-j})$ is upper semi-continuous in p_j and continuous in p_{-j}, and
(3) $\pi_j(p_j, p_{-j})$ is quasi-supermodular in p_j and satisfies the *single crossing property* in (p_j, p_{-j}).

For our game, the strategy space for each player is the unit interval, and hence the first condition is easily satisfied. The second condition follows from the continuity of the excess demand functions given strictly convex preferences.

The third condition relies on two concepts – quasi-supermodularity and the single crossing property – defined by Milgrom and Shannon (1994). In our case these properties hold by virtue of the total ordered strategy space for each player and GS.[⋆] The technical details are omitted for brevity.

[⋆] GS is actually a stronger condition than the single crossing property. Both are assumptions on global behavior. However, it can be shown that the single crossing property implies that if there is a competitive equilibrium, then in at least a small region near the equilibrium, all goods are gross substitutes.

The following two theorems are from Milgrom and Roberts (1991).

THEOREM 8 (Milgrom and Roberts Theorem 11). *Consider a game with strategic complementarities. Let* PNE *denote the set of pure Nash equilibrium profiles. Then,*

$$[U^{0\infty}(S)] = [PNE],$$

where

$$[T] \equiv \{\sigma \in S | \inf(T) \leq \sigma \leq \sup(T)\}.$$

THEOREM 9 (Milgrom and Roberts Theorem 7). *Suppose* $U^{0\infty}(S) = \{\bar{\sigma}\}$*. Then* $\| \sigma(t) - \bar{\sigma} \| \rightarrow$ 0 iff $\{\sigma(t)\}$ *is consistent with adaptive learning.*

First, our game is one with strategic complementarities, and it has a unique Nash equilibrium, which is also the unique competitive equilibrium of our exchange economy, due to the GS assumption. By Theorem 8, the serially undominated set is a singleton that contains only the unique Nash equilibrium.

Second, Theorem 9 states that if the serially undominated set is a singleton, then any adaptive learning process will eventually converge to the element in that singleton. Since both $\{a_j(t)\}$ and $\{b_j(t)\}$ are consistent with adaptive learning for any j, $\{a(t)\}$ and $\{b(t)\}$ are also consistent with adaptive learning, hence *both* converge to the unique equilibrium price vector.

Since $\{\hat{p}(t)\}$ is bounded by $\{a(t)\}$ and $\{b(t)\}(4), \{\hat{p}(t)\}$ must therefore converge to the unique equilibrium price vector as well.

3.5. THE WALRAS ALGORITHM

Finally, we consider the full WALRAS algorithm. Define

$$Z^j(p_j, t) \equiv \sum_{i=1}^{n} x_i^j(p_j, p_{-j}^{t-s_i^j(t)}),$$

where $s_i^j(t) \geq 0$ is the delay of information for agent i in market j at time t. If agent i's most recent update for its jth bid is based on the information it has at time $t - s$, then the delay is simply s. Let

$$r_i^j(t) \equiv t - s_i^j(t).$$

We can then write

$$Z^j(p_j, t) = \sum_{i=1}^{n} x_i^j(p_j, p_{-j}^{r_i^j(t)}).$$

Note that

$$r_i^j(t) \leq t - 1,$$

since the smallest possible delay is 1.

At time t, auction j selects $\hat{p}_j(t)$ such that,

$$\sum_{i=1}^{n} x_i^j(\hat{p}_j(t), p_{-j}^{r_i^j(t)}) = 0.$$

Let $a_j(t)$ and $b_j(t)$ be defined as in Section 3.3 as a best response to $a_{-j}^H(t-1)$ and $b_{-j}^L(t-1)$, where

$$a_{-j}^H(t-1) \equiv \bigvee_{i=1}^{n} a_{-j}(r_i^j(t)),$$

$$b_{-j}^L(t-1) \equiv \bigwedge_{i=1}^{n} b_{-j}(r_i^j(t)).$$

PROPOSITION 10. $\forall \hat{t}, \forall i, \exists \dot{t}_i, \quad r_i^j(\dot{t}_i) > \hat{t}.$

Proof. Fix i. Pick any $\hat{t}$, let $\tilde{t} = \hat{t} + 2$. At $\tilde{t}$, there are two possibilities. One is $r_i^j(\tilde{t}) = \tilde{t} - 1 = \hat{t} + 1$. In this case, our proposition is true immediately.

The second situation is $r_i^j(\tilde{t}) < \tilde{t} - 1$. In this case, the bid to auction j by agent i has not been updated. Suppose that this bid is in the queue with m steps left before it will be called. At each time v WALRAS draws a number d_v randomly from $\{0, 1, 2\}$. The position of our bid in the queue will move forward d_v steps. Since m is a finite positive integer, it is conceivable that eventually the bid's position in the queue will be zero and then it will be updated, unless WALRAS draws 0 every time, which is an event with probability zero. When our bid is updated after t_0 periods – that is, at time $\tilde{t} + t_0$ – we have $r_i^j(\tilde{t} + t_0) = \tilde{t} + t_0 - 1 > \tilde{t} - 1 > \hat{t}$. Again, our proposition holds with probability one.

In fact, if WALRAS draws from a set such as $\{1, 2, 3\}$ instead of $\{0, 1, 2\}$, we can say that this proposition is true *for sure*.

Let us check that $\{a_j(t)\}$ is consistent with adaptive learning. Let $\hat{t} = \max_i\{\hat{t}_i\}$. Now, $\forall \epsilon > 0, \forall \hat{t}$, there exists a $\bar{t}$, such that $\forall t > \bar{t}$, $\forall i$, we have $r_i^j(t) > \hat{t}$. Hence all components of $a_{-j}^H(t-1)$ (when $t > \bar{t}$) occurred between time $\hat{t}$ and t. Therefore, since $a_j(t)$ is a best response to $a_{-j}^H(t-1)$, it belongs to the set $\{a(s) : \hat{t} \leq s \leq t\}$. This is the same as saying

$$\forall \epsilon > 0, \quad \forall \hat{t}, \quad \exists \bar{t}, \quad \forall t > \bar{t}, \quad a_j(t) \in U_j^\epsilon(\{a(s) : \hat{t} \leq s \leq t\}),$$

so $\{a_j(t)\}$ is consistent with adaptive learning. A similar argument holds for $\{b_j(t)\}$.

Thus, both $\{a_j(t)\}$ and $\{b_j(t)\}$ are consistent with adaptive learning for all $j = 1, \ldots, k$. Again by the theorems in the last subsection, $\{a(t)\}$ and $\{b(t)\}$ both converge to the unique equilibrium price with probability one. Since $b_j(t) \leqslant \hat{p}_j(t) \leqslant a_j(t)$ for all j and t, $\{\hat{p}(t)\}$ also converges to the unique equilibrium price with probability one. Thus practically every actual price adjustment process run by WALRAS converges to the unique equilibrium price.

4. Relaxing Gross Substitutability

4.1. Gross Substitutability

As we have shown above, when commodities are gross substitutes, the processes generated by WALRAS converges to a competitive equilibrium. However, the GS requirement is quite restrictive. We would like to be able to establish more relaxed conditions under which the algorithm converges.

First, let us consider a very simple special case. Suppose we have k goods and n consumers in an exchange economy, all of them with the *same* CES preferences. For simplicity, we accord all the goods equal weight, setting the α_j coefficients from (1) to unity,

$$u(x) = \left(\sum_{j=1}^{k} \left(x^j\right)^\rho \right)^{\frac{1}{\rho}}.$$

We restrict the range of ρ to the interval $(-\infty, 1)$, to ensure convexity of preferences.

The budget constraint that consumer i faces is

$$\sum_{j=1}^{k} p_j x_i^j = \sum_{j=1}^{k} p_j e_i^j,$$

where p_j is the price of good j, and e_i^j is agent i's endowment of good j. The solution to the consumer's maximization problem is the demand function

$$x_i^j(p, e) = \frac{\left(\sum_{m=1}^{k} p_m e_i^m\right) p_j^{r-1}}{\sum_{m=1}^{k} p_m^r}, \quad \text{where } r = \frac{\rho}{\rho - 1}.$$

Note that here x_i^j is a function of price and endowment vectors.

The cross-price derivative of this demand function, for $j \neq l$, is

$$\frac{\partial x_i^j}{\partial p_l} = \frac{p_j^{r-1} \left[e_i^l \sum_{m=1}^{k} p_m^r - r \left(p_l^{r-1}\right) \sum_{m=1}^{k} p_m e_i^m \right]}{\left(\sum_{m=1}^{k} p_m^r\right)^2}. \tag{5}$$

If $\frac{\partial x_i^j}{\partial p_l} > 0$, then good j is a gross substitute for good l. From (5), it is clear that this relationship holds given the following:

$$e_i^l \sum_{m=1}^{k} p_m^r - r\left(p_l^{r-1}\right) \sum_{m=1}^{k} p_m e_i^m > 0. \tag{6}$$

If $r < 0$, then (6) holds, provided that all prices are positive, and each consumer starts with strictly positive endowment in at least one good. Since $r = \frac{\rho}{\rho-1}, r < 0$ is equivalent to $0 < \rho < 1$, given that $\rho \in (-\infty, 1)$. Thus, $r < 0$ (or $\rho > 0$) gives us a sufficient condition for gross substitutability for CES agents. However, even if $r > 0$ (or $\rho < 0$), it is still possible that the first term in (6) dominates the second term, especially when $|r|$ is small, which implies that ρ is negative but close to zero.

Another possibility is that condition (6) be violated for some consumers, but still hold in the aggregate. This aggregate substitutability condition is what we really need for the algorithm's convergence.

Let $E^j = \sum_{i=1}^{n} e_i^j$ be the aggregate endowment level in good j. We then have, for $j \neq l$,

$$\begin{aligned}
\sum_{i=1}^{n} \frac{\partial x_i^j}{\partial p_l} &= \frac{p_j^{r-1} \sum_{i=1}^{n} \left[e_i^l \sum_{m=1}^{k} p_m^r - r\left(p_l^{r-1}\right) \sum_{m=1}^{k} p_m e_i^m\right]}{\left(\sum_{m=1}^{k} p_m^r\right)^2} \\
&= \frac{p_j^{r-1} \left[E^l \sum_{m=1}^{k} p_m^r - r\left(p_l^{r-1}\right) \sum_{m=1}^{k} p_m E^m\right]}{\left(\sum_{m=1}^{k} p_m^r\right)^2} \\
&= \frac{p_j^{r-1} \left[p_l E^l \sum_{m=1}^{k} p_m^r - r\left(p_l^{r}\right) \sum_{m=1}^{k} p_m E^m\right]}{p_l \left(\sum_{m=1}^{k} p_m^r\right)^2}.
\end{aligned}$$

Now it is clear that

$$\sum_{i=1}^{n} \frac{\partial x_i^j}{\partial p_l} > 0$$

if the following is true:

$$p_l E^l \sum_{j=1}^{k} p_j^r - r p_l^r \sum_{j=1}^{k} p_j E^j > 0. \tag{7}$$

Again, we can see that $r < 0$ guarantees this inequality. When $r > 0$ but $|r|$ is small, (7) is still likely to hold. The inequality (7) is equivalent to

$$\frac{p_l E^l}{\sum_{j=1}^{k} p_j E^j} > \frac{p_l^r r}{\sum_{j=1}^{k} p_j^r}. \tag{8}$$

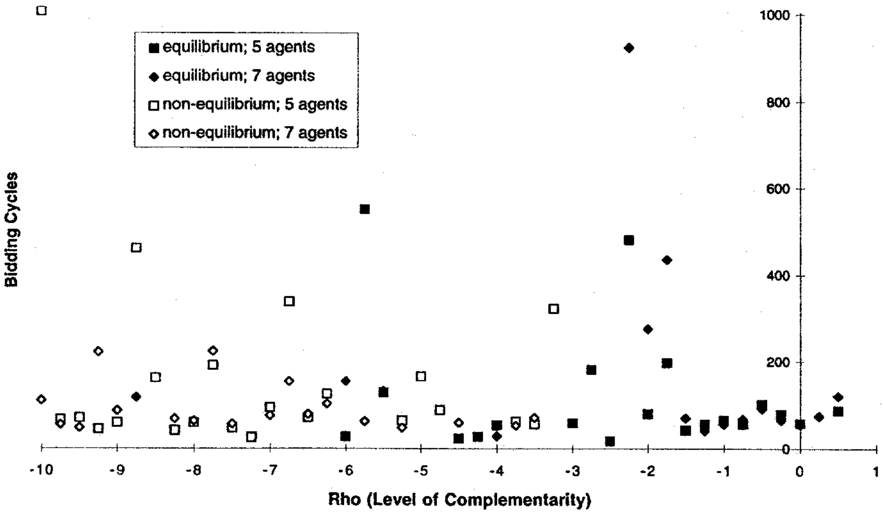

Figure 2. Equilibrium as a function of degree of complementarity; all economies have five goods.

The first term is the value of good l divided by the value of all goods; the second term does not appear to have such a simple interpretation. It is interesting to note that if we sum the above inequality across *all* goods, we get $r < 1$, which is implied by $\rho \in (-\infty, 1)$. So the *aggregation* of (8) across all goods must be true, even though it might not hold for individual goods.

4.2. More Experimental Evidence

The experimental results presented in Section 2.3 were based on CES consumers with a fixed substitution coefficient $\rho = 0.5$. From the CES utility function (Equation (1)), we know that when $0 < \rho < 1$, both the individual and the market demand functions exhibit gross substitutability. Therefore, WALRAS always converges for these examples.

In order to explore situations where convergence is not guaranteed, we randomly generated a range of CES exchange economies with negative ρ values. Specifically, we varied ρ from $+0.5$ to -10, with a step size of 0.25. For each ρ value, we randomly generated two five-good economies, one with five consumers and the other with seven. The α_j coefficients and consumer endowments were randomly generated from uniform distributions.

Figure 2 presents our results. The vertical axis indicates the number of bidding cycles for each economy to converge. The horizontal axis indicates the values of ρ. There were nine economies that did not converge at all after 5000 cycles (not shown in the figure). All of these were seven-consumer economies with $\rho \leq -2$. In the figure, black dots indicate that the experimental run converged to the equilibrium for the economy, and white dots indicate that the run converged, but to a *non-equilibrium* state.

The experiment suggests that for small negative values of ρ, the randomly generated CES exchange economies are very likely to converge to an equilibrium. In fact, all economies with a $\rho > -2$ converge to an equilibrium in our experiments. For large negative values of ρ, however, few of the economies converge to equilibria.

These results suggest possible convergence when ρ is negative but with a small absolute value. Our theoretical investigation in Section 4.1 also suggests this possibility, as the inequality (6) is more easily satisfied for smaller degrees of complementarity.

4.3. Notes on Aggregate Demand Functions

While the experimental results are certainly encouraging, formal results establishing convergence under a class of utility functions wider than GS is still out of reach at this time. That this is difficult is perhaps not surprising. The Sonnenschein-Mantel-Debreu (SMD) theorem (Shafer and Sonnenschein 1982) tells us that utility maximization by each individual agent does not impose any restriction on the aggregate demand function. Even if restrictions stronger than utility maximization are imposed on agents – such as homothetic preferences – there are typically still no restrictions on market demand functions.

The SMD theorem suggests that if the market demand functions are to exhibit particular properties, we are likely to need restrictions on how individuals' preferences are related *to each other*. For example, one property of interest is *exact aggregation*, where the market demand function behaves as if it were generated by some utility maximization individual. For discussions, see Muellbauer (1976), Chipman (1974), Hildenbrand (1983), and Kirman (1992). The typical conditions either call for all agents to have proportional incomes or require all agents to have identical preferences.

5. Incremental Bidding

Recall that bids in the WALRAS algorithm consist of single-good demand functions, specifying the quantity demanded as the own price changes, keeping the remaining prices fixed. This notion of bid differs from that often employed elsewhere, where agents announce only a particular demand quantity at a particular point price. Some may be concerned that the WALRAS algorithm therefore places an undue burden on the agents, who are required to compute a one-dimensional curve rather than a zero-dimensional point.

If calculating demand curves is a serious burden, agents may in fact participate in the protocol by calculating and submitting approximate demand functions, perhaps based on fitting a small number of points. We have explored a particular version of this, called *incremental bidding*. In incremental bidding, agents are allowed to update only one point on their bid curve at a time, rather than revising their whole demand function all at once.

The details of the scheme are as follows. A demand schedule is denoted by an ordered set of price-quantity pairs, with linear interpolation between the specified prices. At each bidding opportunity t for good j, agent i selects a price $\bar{p}_j$ to bid on, and adds the point $\langle \bar{p}_j, x_i^j(\bar{p}_j, \hat{p}_{-j}^{t-1})\rangle$ to its demand schedule, where $\hat{p}^t$ is the price announced by auctions at time t. Note that at any time, the demand curve contains points based on different information about the other goods' prices. On adding a new point, we delete any previous points that are inconsistent with the new point and an assumption of downward-sloping demand.

If the bid prices $\bar{p}_j$ are chosen systematically, agents will eventually update all the relevant information under this incremental bidding scheme. In our experiments with incremental bidding, we chose $\bar{p}_j = \hat{p}_j^{t-1} + \eta$, where η is a random variable with a mean value of zero. On one hundred randomly generated exchange economies with five goods and five CES consumers (with $\rho = 0.5$), all converged to equilibrium.[⋆]

Since one bid in the incremental bidding protocol adds considerably less information than a usual WALRAS bid (i.e., an entire demand function), it is not surprising that it takes WALRAS longer to converge under incremental bidding. Specifically, the incremental bidding scheme took on average about 70 times more cycles than the regular scheme to reach a total excess demand below a particular threshold.[⋆⋆]

6. Conclusion

In describing and analyzing the WALRAS algorithm, we have extended the standard tatonnement convergence results to a setting that admits asynchrony and facilitates distributed implementation. Although the rigorous convergence guarantees are limited to the classical conditions with gross substitutability, we have found empirically that the method often works when GS is violated to some degree.

By employing the general machinery of Milgrom and Roberts (1991) in our proofs, we also suggest how a variety of related asynchronous price-adjustment algorithms can be shown to converge. In continuing work we intend to explore some of the more interesting variants in bidding protocols (e.g., incremental bidding, discussed above), as well as other generalizations of the WALRAS algorithm.

[⋆] Without the random perturbation η (i.e., $\bar{p}_j = \hat{p}_j^{t-1}$), the process typically gets stuck in a non-equilibrium state.

[⋆⋆] As mentioned in Section 2.3, our measure of total excess demand is the sum of absolute excess demand for each good. The threshold we used was 0.1 units, which is approximately 1/1000 of the total endowed quantity.

Acknowledgements

We are grateful to Hal Varian for suggestions about this work, and to several Michigan students for assistance. In particular, Philip Haar initiated some of the experimental work with the WALRAS algorithm, and Brian Forney generated most of the data included in this paper. This work was supported in part by an NSF National Young Investigator award.

References

Arrow, K.J. and Hahn, F.H. (1971). *General Competitive Analysis*. San Francisco: Holden-Day.

Arrow, K.J., Block, H.D. and Hurwicz, L. (1959). 'On the stability of competitive equilibrium II'. *Econometrica* **27**, 82–109.

Bertsekas, D.P. and Tsitsiklis, J.N. (1989). *Parallel and Distributed Computation: Numerical Methods*. Prentice Hall.

Campbell, D.E. (1987). *Resource Allocation Mechanism*. Cambridge University Press.

Chipman, J.S. (1974). 'Homothetic preferences and aggregation'. *Journal of Economic Theory* **8**, 26–38.

Clearwater, S., ed. (1995). *Market-Based Control: A Paradigm for Distributed Resource Allocation*. World Scientific.

Dafermos, S. and Nagurney, A. (1989). 'Supply and demand equilibration algorithms for a class of market equilibrium problems'. *Transportation Science* **23**, 118–124.

Epstein, J.M. and Axtell, R. (1996). *Growing Artificial Societies: Social Science from the Ground Up*. MIT Press.

Eydeland, A. and Nagurney, A. (1989). 'Progressive equilibration algorithms: The case of linear transaction costs'. *Computer Science in Economics and Management* **2**, 197–219.

Fisher, F.M. (1983). *Disequilibrium Foundations of Equilibrium Economics*. Cambridge, Cambridge University Press.

Friedman, D. and Rust, J. (eds) (1993). *The Double Auction Market*. Addison-Wesley.

Hicks, J.R. (1948). *Value and Capital*. London: Oxford University Press, second, edition.

Hildenbrand, W. (1983). 'On the law of demand'. *Econometrica* **52**, 997–1019.

Huberman, B. and Glance, N. (1993). 'Evolutionary games and computer simulations'. In *Proceedings of the National Academy of Sciences*.

Hurwicz, L. (1977). 'The design of resource allocation mechanisms'. In K.J. Arrow and L. Hurwicz (eds), *Studies in Resource Allocation Processes*. Cambridge: Cambridge University Press. 3–37. Reprinted from *American Economic Review Papers and Proceedings*, 1973.

Kirman, A.P. (1992). Whom or what does the representative individual represent? *Journal of Economic Perspectives* **6**(2), 117–136.

Malinvaud, E. (1985). *Lectures on Microeconomic Theory*. North-Holland, revised edition. English translation by A. Silvey.

Mas-Colell, A., Whinston, M.D. and Green, J.R. (1995). *Microeconomic Theory*. New York: Oxford University Press.

Milgrom, P. and Roberts, J. (1991). Adaptive and sophisticated learning in normal form games. *Games and Economic Behavior* **3**, 82–100.

Milgrom, P. and Shannon, C. (1994). Monotone comparative statics. *Econometrica* **62**, 157–180.

Muellbauer, J. (1976). Community preferences and the representative consumer. *Econometrica* **44**(5), 979–999.

Murphy, F.H. and Mudrageda, M.V. (1994). Using the cobweb algorithm to decompose large-scale economic equilibrium models. Manuscript, Temple University.

Murphy, F.H. and Mudrageda, M.V. (1995). A decomposition approach for a class of economic equilibrium models. Manuscript, Temple University.

Reiter, S. and Simon, C. (1992). Decentralized dynamic processes for finding equilibrium. *Journal of Economic Theory* **56**, 400–425.

Rust, J. (1996). Dealing with the complexity of economic calculations. Technical report, Yale University.

Samuelson, P.A. (1974). *Foundations of Economic Analysis*. Cambridge, Mass.: Harvard University Press.

Scarf, H. (1973). *The Computation of Economic Equilibria*. New Haven, CT: Yale University Press.

Shafer, W. and Sonnenschein, H. (1982). Market demand and excess demand functions. In K.J. Arrow and M.D. Intriligator (eds), *Handbook Of Mathematical Economics*, volume 2. Amsterdam: North Holland.

Shoven, J.B. and Whalley, J. (1992). *Applying General Equilibrium*. Cambridge University Press.

Shoven, J.B., Whalley, J. and Wiegard, W. (1992). Computing general equilibria. In: J.B. Shoven and J. Whalley (eds), *Applying General Equilibrium*. Cambridge University Press. Chapter 3.

Takayama, A. (1985). *Mathematical Economics*. Cambridge: Cambridge University Press.

Varian, H.R. (1992). *Microeconomic Analysis*. New York: W.W. Norton and Company, 3rd edition.

Walras, L. (1954). *Elements of Pure Economics*. Allen and Unwin. English translation by William Jaffé, originally published in 1874.

Wellman, M.P. (1993). A market-oriented programming environment and its application to distributed multicommodity flow problems. *Journal of Artificial Intelligence Research* **1**, 1–22.

Wellman, M.P. (1995). Market-oriented programming: Some early lessons. Clearwater.

Yamaki, H., Wellman, M.P. and Ishida, T. (1996). A market-based approach to allocating QoS for multimedia services. In *Second International Conference on Multiagent Systems*, 385–392.

Ygge, F. and Akkermans, H. (1996). Power load management as a computational market. In *Second International Conference on Multiagent Systems*, 393–400.

Zhao, L. and Dafermos, S. (1991). General economic equilibrium with variational inequalities. *Operations Research Letters* **10**, 369–376.