Heterogeneity, Stability, and Efficiency in Distributed Systems

James D Thomas *
Department of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
jthomas+@cs.cmu.edu

Katia Sycara
Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213
katia+@ri.cmu.edu

Abstract

This paper explores the increasing the heterogeneity of an agent population to stabilize decentralized systems by adding bias terms to each agent's expected payoffs. Two approaches are evaluated, corresponding to heterogeneous preferences and heterogeneous transaction costs; empirically, the transaction cost case provides stability with near optimal payoffs under certain conditions. Theoretically, in the idealized case of an infinite number of agents, it is proven that the system with added heterogeneous preferences has a fixed point different from that of the unbiased system, guaranteeing suboptimal performance, while the transaction cost case is demonstrated to have a fixed point identical to that of the unbiased system, and it is further shown to be a contraction mapping, guaranteeing convergence. This contraction mapping allows us to conceptualize the model with heterogeneous transaction costs as a decentralized root finding system.

Topic areas: decentralized systems, distributed search

1 Introduction

The growing interest in decentralized systems evinced by the fields of Multiagent Systems (MAS) and Distributed Artifical Intelligence (DAI) brings along with it a concern for the stability of such systems. When agents lack explicit coordination mechanisms, or act on incomplete or delayed information, actions that may appear locally optimal may create global instability at the system level. This can be a particular problem in systems where agents allocate resources among themselves with no central control. Problems with this characteristic include load balancing over multiple processors, the allocation of internet traffic over multiple network routes, and market-like control systems [2].

In such systems, when agents perceive that a resource is underutilized, they try to increase their utilization of it. But if all of the agents shift towards it, the other resources become underutilized – the agents see this, and try to switch back; this leads to unstable behavior. Such instabilities have a negative effect on system performance. Instability creates uncertainty, which further handicaps agent decision making. Additionally, if transaction costs for switching between resources are present in the system, the frequent switching between resources incurs unnecessary costs. Finally, instability can prevent these systems from settling into an equilibrium that provides the most efficient allocation of resources.

The idea that agent heterogeneity is a key to stabilizing these systems has been capturing increasing attention in recent work. The intuition is that since the instability is caused by too many agents wanting to shift, a heterogeneous agent population will increase the number of agents who don't want to shift, thus stabilizing the system. Kephart, Huberman & Hogg [5] noted the key to stability was increasing the heterogeneity of agent responses. Arthur [1] investigated similar issues in the context of the Santa Fe bar problem. He allowed the agents to make decisions based on predictions of the future state of the system; it turns out that for stability, the accuracy of the predictions were less important than their heterogeneity. Schaerf et al [7] studied a distributed load balancing system and discovered that communication between agents, by reducing heterogeneity, actually worsened system performance.

We explore the effects of creating heterogeneous agents by introducing bias terms to their perceptions of resource payoffs. We examine the effect of such heterogeneity both analytically and via simulations. We classify several approaches to applying heterogeneity, demonstrating that a transaction cost approach empirically provides stability close to the point of maximum payoffs. Furthermore, we analyze the application of heterogeneity and can prove, by

*Thanks to Matt Glickman, Bryan Routledge, John Miller, Onn Shehory, and Somesh Jha for helpful discussion. Sponsored by NSF grant IRI-9612131 and ONR grant N-00014-96-1-1222

means of a contraction mapping argument, guaranteed convergence to the proper equilibrium under certain conditions. In addition, we show that in order to apply heterogeneity efficiently in real systems, one should carefully examine and decide upon the amount of heterogeneity to be applied.

In section 2, we set up the basic model. Section 3 explains our two approaches to heterogeneity, conceptualized as heterogeneous preferences and heterogeneous transaction costs. The ability of these approaches to tame volatility in empirical simulations is reported in section 4. Section 5 contains a theoretical analysis of both approaches, proving that in a system with an infinite number of agents, the transaction cost approach is guaranteed to converge to the true equilibrium of the system. Finally, section **??** applies these techniques to the problem of allocating tasks across parallel processors to maximize throughput, and section 7 summarizes and discusses outstanding questions.

2 The Basic Model

For empirical modelling, we turned to a simplified version of the computational ecosystem model pioneered by Huberman, Hogg, and their colleagues at Xerox Parc [4, 5]. We created a simple model of resource utilization by agents. The agents want to maximize their payoffs, and decide which resource to utilize using publicly available information about each resource's payoffs. The resources' payoffs are decreasing functions in the number of agents utilizing them, so the more agents that utilize a resource the lower its payoffs.

In the specific model used in this section, there are two resources, both with payoffs that are linear decreasing functions of the number of agents utilizing them. Letting p denote the proportion of agents using resource one, the payoffs of resource one and resource two are $r_1 = 10 - 5p$, $r_2 = 10 - 10 \cdot (1 - p)$ respectively. Figure 1 shows the payoff for each resource plotted against p. When the payoffs from each resource is equal at $p = 2/3$, the system is in equilibrium, since no agent has incentive to switch; this point also provides maximum aggregate payoffs. However, this equilibrium is unstable; if p is smaller than $2/3$ by just a little bit, then the payoffs for resource one are higher than those of resource two, and every agent has incentive to switch over to using resource one.

Following Kephart et al [5], our agents act myopically on delayed information: they expect the next payoff to be the same as the last payoff they've seen, but there is a lag time before they can see the payoffs, and must form expectations based on payoff information a few turns old. In addition, not every agent acts every turn; each has some fixed probability of being able to act. A small Gaussian noise term is added to all payoffs.

Figure 2 shows example dynamics for this model (50 agents, two resources, lag times uniformly distributed between one and five turns, each agent has a $.3$ chance of acting per turn, noise term standard deviation of $.05$ – these values are used for all simulations unless noted otherwise). The pattern is obvious; p oscillates around the unstable equilibrium of $p = 2/3$. If $p < 2/3$, then every agent sees that resource one pays more than resource two, and tries to shift. This shift inevitably overshoots, and p rises to above 2/3, leading to higher payoffs for resource two, which starts another shift back towards resource two. Since the aggregate payoffs are highest at $p = 2/3$, the normalized payoffs plotted on the graph show that these system oscillations give rise to sub-optimal aggregate payoffs.

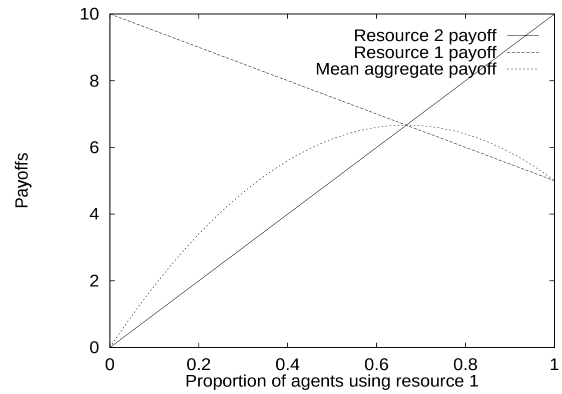

Figure 1. Resource payoffs as a function of p

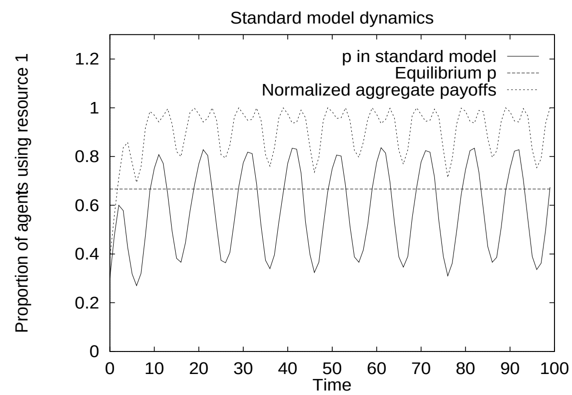

Figure 2. Standard Model Dynamics

3 Adding Heterogeneity

The basic problem this model presents is how to stabilize the system, specifically, how to get the agents to distribute themselves among resources such that none of them have incentive to switch.

There are many possible approaches to this problem. Game theory [3] offers up an easy answer: a symmetric mixed strategy where every agent chooses resource one with probability 2/3, and resource two with probability 1/3. However, this solution assumes that the agents know (or can learn) the true equilibrium, and would incur horrendous costs if there was a transaction cost to switch between resources. Similarly, adding noise to the system stabilizes the dynamics [5], at the cost of moving the system away from the true equilibrium (however, as delays in the system increase, this negative effect becomes less of an issue). One might try to build agents that observe the system dynamics, learn to predict the oscillations, and act in ways that stabilize the system. Kephart [5] showed that allowing some agents to predict the system did not necessarily improve the dynamics; with small numbers of predicting agents, system stability improved, but as the proportion of these predicting agents grew, the system grew unstable again.

Using heterogeneity for stability has the advantage of not imposing an additional computational burden on the agents, and potentially stabilizing the system near the true equilibrium. Hogg & Huberman [4] approach this problem by increasing the heterogeneity of the system: by manipulating payoffs, they effectively increase the heterogeneity in agents' lag times, which is enough to stabilize the system. In addition, they briefly explore the idea of introducing classes of agents that systematically err in estimating the payoffs of resources; the resulting heterogeneity in decision making calms the system. It is this approach that we investigate in this paper.

Specifically, in this paper heterogeneity will mean making the decision process of each agent slightly different from that of every other agent's. We will accomplish this by modifying each agent's expected payoffs by bias terms that vary between agents. These bias terms remain constant over time. Intuitively, this works by increasing the diversity in agent responses. In the standard model, as soon as p drops below 2/3, every agent wants to switch to resource 1. With heterogeneity, the perceived payoffs of some agents will be distorted enough that some will still prefer to stay with resource two, hopefully dampening the oscillations enough for the system to stabilize.

Instead of creating classes of agents, we opted for a 'distributional' approach, creating a positive bias term for each agent/resource pair by drawing from uniformly distributed[1], interval.

We wanted to explore two issues: first, is there a right amount of heterogeneity? Very small bias terms will not have much of an effect; very large ones might dominate the decisions of the agents enough to drive the system far from optimality. Since we draw our bias terms from a uniformly distributed interval $[0, 2n]$, where n is the mean of this distribution, this n provides a good measure of the amount of heterogeneity we add to the agents. The higher n is, the higher the average bias term is, and the more importance it plays in the agent's actions.

We also wanted to examine exactly how these bias terms should be applied. The obvious approach is to apply them to every expected payoff an agent sees. Since these biases are constant over time, this can be interpreted in economic terms as heterogeneous preferences.

An alternate approach applies the bias terms only to those resources an agent is not currently using – these biases, when all positive, could then be interpreted as heterogeneous transaction costs. Each agent factors in a small cost to switch resources, and this cost is different for each agent/resource pair.

To implement these approaches in our model, we create a bias term for each agent/resource pair (or agent class/resource pair). If we let r_i stand for the true payoffs of resource i, $bias_{i,j}$ represent the constant bias term for each agent/resource pair, then using the heterogeneous preferences approach, the perceived payoffs $r'_{i,j}$ of each agent (or agent class) j for resource i is:

$$r'_{i,j} = r_i - bias_{i,j}$$

And in the heterogeneous transaction cost approach:

$$\begin{aligned} r'_{i,j} &= r_i - bias_{i,j} && \text{agent j is not currently using resource i} \\ r'_{i,j} &= r_i && \text{agent j is currently using resource i} \end{aligned}$$

4 Empirical Results

We examined these issues by running simulations, using both the preference approach and the transaction cost approach. We performed 20 trials, each time running our model for 200 turns with levels of heterogeneity varying from 0 to 5. The results are presented in figures 3,4,5.

Figure 3 addresses the issue of volatility, by plotting the average standard deviation of p over the 200 turns against the level of heterogeneity. This is a good measure of the volatility of the system; low standard deviation means that p is more or less motionless. In the transaction cost approach, at a heterogeneity level of about $.75$ the volatility of p drops almost to zero; the system is stabilized. The preferences approach is not quite as effective; there is a slower drop in the volatility of p that never goes below a standard deviation of $.02$.

Figures 4,5 address the issue of optimality. Figure 4 plots the payoffs of the two approaches against the level of heterogeneity. Figure 5 plots the mean 'error' of p, that is the average difference between p in the simulation and the optimal p of $2/3$. Since being close to the optimal p produces close to optimal payoffs, the two graphs show similar results.

[1] We experimented with both using classes of agents and the 'distributional' approach used in this paper; the distributional approach proved universally superior, but for concerns of brevity those results are not presented here.

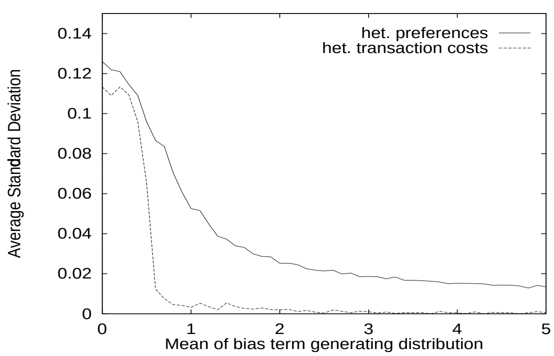

Figure 3. Standard deviation of $'p'$

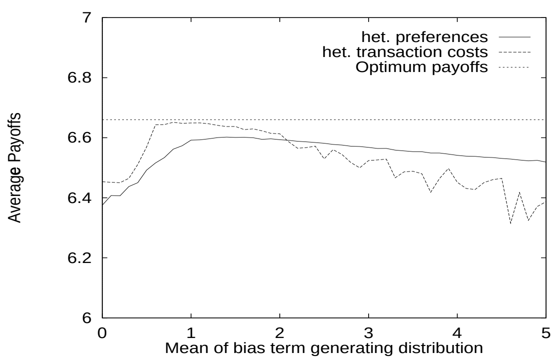

Figure 4. Aggregate payoffs

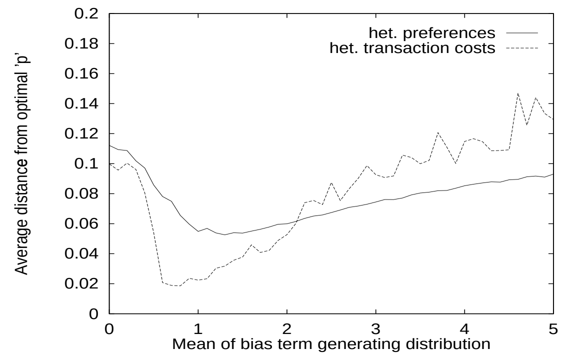

Figure 5. Distance from optimal $'p'$

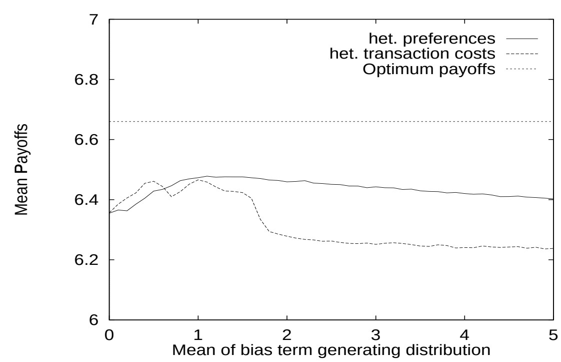

Figure 6. Moving Equilibrium

In the transaction cost approach, corresponding to the rapid drop in volatility around heterogeneity level of $.75$ there is a rapid rise in payoffs and a rapid decrease in error. But, as the level of heterogeneity increases, the error rises and payoffs decrease, indicating that the system has stabilized away from the optimal p of $2/3$. The preferences approach shows a more gradual reduction of error, but also a more gradual decay of performance as the heterogeneity level rises.

In conclusion, there is a 'sweet spot' for levels of heterogeneity between $.75$ and 2.25 for which the transaction cost approach is clearly superior to the preferences approach, both in reduction of volatility and nearness to optimal payoffs. However, past the 'sweet spot', although the transaction cost approach still controls volatility, the payoffs is provides is inferior to those of the preferences approach.

We also tested these techniques on a 'moving target', where the resource payoffs shifted over time. This would require the ability to not only initially stabilize the system, but to move with it as it shifted. To test this, we re-ran the simulations, but every fifty turns we flipped the payoffs of the two resources, so that the system alternated between the two equilibria $p = 2/3$ and $p = 1/3$. Figure 6, plots the payoffs against level of bias for both cases.

The results here are similar to the static case; the same sharp rise and decline in payoffs for the transaction cost in approach, and the more gradual improvement and delay of the preferences approach. Here, however, the preferences approach provides comparable if not superior payoffs; it is possible that the preferences approach 'moves' better when confronted with a new equilibrium since it has never fully stabilized the system, balancing out the transaction costs approach's advantage in static situations.

5 A Theoretical Interlude

We would like to have a theory that explains the results above. For purposes of tractability some simplifying assumptions will be needed, but we hope that the conclusions reached here will generalize to similar domains. We will examine the model with no noise or delays, and we assume the existence of an infinite number of agents – the law of large numbers will allow us to represent the heterogeneity induced by the bias terms as a frequency distribution. We can then treat the system as a 'representative agent' and analyze the system's behavior by looking at the difference equation that describes the evolution of p, the proportion of agents using resource one, over time. Our general approach will be to write down the difference equations that represent the idealized system, look for a fixed point, and examine convergence properties.

Here we let a stand for the proportion of agents that act each turn, $p(t)$ the proportion of agents at time t using resource one, and $r1$, $r2$ the payoffs of resources one and two, respectively. Let $f(p(t))$ stand for $r1(p(t)) - r2(p(t))$, the payoffs from resource one minus the payoffs from resource two. This represents a sort of 'net incentive'; when $f(p(t))$ is positive, resource one pays more than resource two, and all agents want to shift to resource one.

In our simplified model without heterogeneity, the equations describing the evolution of $p(t)$ are:

$$\mathrm{p(t+1)} = \begin{cases} p(t) + a \cdot (1 - p(t)) & f(p(t)) > 0 \\ p(t) & f(p(t)) = 0 \\ p(t) - a \cdot p(t) & f(p(t)) < 0 \end{cases}$$

If $f(p(t)) > 0$, then every agent wants to use resource one, and so of the agents using resource two, $(1 - p(t))$, all those that can act this turn move, shifting p by $a \cdot (1 - p(t))$. If $f(p(t)) = 0$, then no one wants to move, and the system is stable. We will denote this fixed point of the underlying system p'. Since this is the point of maximum aggregate payoffs, we would like the addition of heterogeneity to stabilize as close to this point as possible.

5.1 Heterogeneous Preferences

Let us now add heterogeneous preferences by adding a term drawn from a uniform distribution across the interval $[0, 2n]$ to each agent's payoffs for each resource (here, n represents the 'level' of heterogeneity as used in section 4, the mean of the distribution from which the bias terms are generated). Since f represents resource one's payoffs subtracted from resource two's payoffs, this is identical to adding a term drawn from uniform distribution over $[-2n, 2n]$ to f. Remember that agents switch from resource two to resource one if they perceive $(f(p(t))) < 0$, and vice versa. The key to understanding the effect of the bias terms is to understand for a given $(f(p(t)))$, what proportion of agents have bias terms strong enough to shift their behavior. Given $f(p(t))$, let $P(f(p(t)) - bias > 0)$ represent the proportion of agents for whom $f(p(t))$ plus bias terms is less than zero. For example, if we let let $2n = 1$, and $f(p(t)) = .5$, given uniformly distributed bias, $P(f(p(t)) - bias > 0) = .75$ and $P(f(p(t)) - bias < 0) = .25$. These terms would represent the proportion of agents switching from resource two to one, and the proportion switching from one to two, respectively. Adding these terms gives us the more complicated single equation:

$$\begin{aligned} p(t+1) &= p(t) + a \cdot (1 - p(t)) \cdot P(f(p(t)) - bias > 0) \\ &\quad - a \cdot (p(t)) \cdot P(f(p(t)) - bias < 0) \end{aligned}$$

Since $P(f(p(t)) - bias > 0) + P(f(p(t)) - bias < 0) = 1$, this simplifies to

$$\begin{aligned} p(t+1) &= p(t) + a \cdot (1 - p(t)) \cdot P(f(p(t)) - bias > 0) \\ &\quad - a \cdot (p(t)) \cdot (1 - P(f(p(t)) - bias > 0)) \\ &= p(t) + a \cdot P(f(p(t)) - bias > 0) \\ &\quad + a \cdot p(t) \cdot P(f(p(t)) - bias > 0) \\ &\quad - a \cdot p(t)) + a \cdot p(t) \cdot P(f(p(t)) - bias > 0) \end{aligned}$$

The two $a \cdot p(t) \cdot P(f(p(t)) - bias > 0)$ terms cancel, leading to:

$$\begin{aligned} p(t+1) &= p(t) + a \cdot P(f(p(t)) - bias > 0) - a \cdot p(t) \\ &= (1 - a)p(t) + a \cdot P(f(p(t)) - bias > 0) \end{aligned}$$

Assuming there is a fixed point p'' such that $f(p'') \in [-2n, 2n]$, to find it we need a p'' such that

$$\begin{aligned} p'' &= (1 - a)p'' + a \cdot P(f(p'') - bias > 0) \\ a \cdot p'' &= a \cdot P(f(p'') - bias > 0) \\ p'' &= P(f(p'') - bias > 0) \end{aligned}$$

Note that this fixed point is not the fixed point of the unbiased system. At the unbiased system fixed point $p' = 2/3$, $f(p') = 0$, and so $P(f(p') - bias > 0) = 1/2$. Let us examine the $P(f(p(t)) - bias > 0)$ term. Given that the bias is drawn from a uniform distribution,

$$\mathrm{P(f(p(t))} = \begin{cases} 1 & f(p(t)) > 2n \\ 1/2 + f(p(t))/(4 \cdot n) & f(p(t)) \in [-2n, 2n] \\ 0 & f(p(t)) < -2n \end{cases}$$

so, given that $f(p'') \in [-2n, 2n]$,

$$p'' = 1/2 + f(p'')/4n$$

Intuitively, this makes sense. The fixed point is a mixture of two terms. The first, $1/2$, is the equilibrium that would be induced by the heterogeneous preferences $1/2$ of the agents would prefer each resource. The second depends on the payoffs functions and represents the true underlying fixed point. Note that this second term grows less important the larger n is; increasing the amount of heterogeneity pushes us closer and closer to random chance.

We can examine this fixed point in the case where both resources have linear payoffs (as in the simulations above) of form $a + bx$; $r1 - r2$ gives an f of the form $(p(t)) = c1 + d1 \cdot (p(t)) - (c2 + d2 \cdot (1 - p(t))) = (c1 - c2 + d2) + (d1 - d2) \cdot p(t)) = c + d \cdot p(t)$.

$$\begin{aligned}
p'' &= 1/2 + (c + d \cdot p'')/(4n) \\
p'' \cdot (1 - d/(4n) &= 1/2 + c/(4n) \\
p'' \cdot ((4n - d)/4n) &= (2 \cdot n + c)/(4n) \\
p'' &= (2 \cdot n + c)/(4n - d)
\end{aligned}$$

In the payoffs used for the empirical simulations, $c = 10$ and $d = -15$, giving:

$$p'' = (n + 10)/(4n + 15)$$

Here we can see that if $n = 0$, $p'' = 2/3$, but as n grows p'' approaches $(1/2) \cdot n = 1/2$. So, the stronger the heterogeneous preferences, the farther away we stabilize from the underlying equilibrium.

Note that the preceding analysis can only say something about the system when $f(p(t)) \in [-2n, 2n]$, where n is the level of heterogeneity explained above; if $f(p(t))$ never goes into this range, then the heterogeneous preferences or transaction costs are not large enough to affect the decisions of any agents. In terms of the stability of the system, understanding when the falls into this zone, allowing the heterogeneity to have it's stabilizing effect, is crucial to stability. Unfortunately, in the general case, the only rule of thumb is that large values of n create a zone big enough to guarantee that the system will step into it.

5.2 Transaction costs

Adding non-heterogeneous transaction costs of n give us:

$$\text{p(t + 1)} = \begin{cases} p(t) + a \cdot (1 - p(t)) & f(p(t)) > n \\ p(t) & f(p(t)) \in [-n, n] \\ p(t) - a \cdot p(t) & f(p(t)) < n \end{cases}$$

Here, if the payoff difference is smaller than the transaction cost, no agents will switch and the system will freeze. So, high transaction costs stabilize the system, but at a fixed point potentially far away from the underlying equilibrium.

What happens if we make the transaction costs heterogeneous? The basic equations are:

$$p(t+1) = \begin{cases} p(t) + a \cdot (1 - p(t)) \cdot P(f(p(t)) - bias > 0) & f(p(t)) > 0 \\ p(t) & f(p(t)) = 0 \\ p(t) - a \cdot p(t) \cdot P(f(p(t)) - bias < 0) & f(p(t)) < 0 \end{cases}$$

Here, if $f(p(t)) > 0$, meaning that resource one pays more than resource two, the same number of agents switch from resource two to resource one as in the heterogeneous preferences case. However, no agents switch from resource one to resource two, since the transaction costs only makes resource two's payoffs look less favorable; they have no effect on resource one's payoffs. This is the key difference between the two approaches. Assuming a uniform distribution, we get

$$p(t+1) = \begin{cases} p(t) + a \cdot (1 - p(t)) & f(p(t)) - n > 0 \\ p(t) + a \cdot (1 - p(t)) \cdot (f(p(t))/2n) & f(p(t)) \in (0, 2n] \\ p(t) & f(p(t)) = 0 \\ p(t) - a \cdot p(t) \cdot (-f(p(t))/2n) & f(p(t)) \in [-2n, 0) \\ p(t) - a \cdot p(t) & f(p(t)) + n < 0 \end{cases}$$

Note that if there exists a p' in the original system such that $f(p') = 0$, this p' is still a fixed point in the system with added transaction costs. So a system heterogeneous transaction costs has a fixed point that is identical to the equilibrium of the underlying system. We'd like to show that this system necessarily converges to p', which would entail showing that

$$|p' - p(t+1)| < k \cdot |p' - p(t)|$$

for some $k < 1$. We will examine this by cases. First, assume that $p(t) < p'$, which implies that $f(p(t)) > 0$

$$\begin{aligned}
|p' - p(t+1)| &= |p' - (p(t) + a \cdot (1 - p(t)) \cdot (f(p(t))/n))| \\
&= |(p' - p(t)) - a \cdot (1 - p(t)) \cdot (f(p(t))/2n)|
\end{aligned}$$

As above, assume the linear payoff case (the general case will be dealt with later). We already know that there exists a p' where payoffs are equal, ie. $r_1 = r_2$ and by extension $f(p') = 0$. This implies that $p' = -c/d$.

$$\begin{aligned}
|p' - p(t+1)| &= |(p' - p(t)) - \\
&\quad a \cdot (1 - p(t)) \cdot (c + d \cdot (t))/2n|
\end{aligned}$$

Since $c + d \cdot p' = 0$, $c = -d \cdot p'$.

$$\begin{aligned}
|p' - p(t+1)| &= |(p' - p(t)) \\
&\quad -(a \cdot (1 - p(t)) \cdot (-d \cdot p' + d \cdot p(t))/2n)| \\
&= |(p' - p(t)) \\
&\quad +a \cdot (1 - p(t)) \cdot d(p' - p(t))/2n)| \\
&= |(p' - p(t) \cdot (1 + a \cdot (1 - p(t)) \cdot d/2n)| \\
&\leq |p' - p(t)| \cdot |1 + a \cdot (1 - p(t)) \cdot d/2n)| \\
&= |p' - p(t)| \cdot |(1 + (1 - p(t)) \cdot d \cdot (a/2n)|
\end{aligned}$$

In the case above, where $a = .3$ and $d = -15$, picking n to be greater than 1.25 ensures that $d \cdot (a/2n) > -2$, and $|1 + (1 - p(t)) \cdot d \cdot (a/2n)| < 1$, thus guaranteeing that we have a contraction mapping that converges to p'.

The other case, where $p(t) > p'$, which implies that $f(p(t)) < 0$, follows as well:

$$\begin{aligned}
|p' - p(t+1)| &= |p' - (p(t) - a \cdot p(t) \cdot (-f(p(t))/2n))| \\
&= |(p' - p(t)) + a \cdot p(t) \cdot (f(p(t))/2n)| \\
&= |(p' - p(t)) \\
&\quad +a \cdot p(t) \cdot (-d \cdot p' + d \cdot p(t)/2n)| \\
&= |(p' - p(t)) \\
&\quad -a \cdot p(t) \cdot d \cdot (p' - p(t))/2n))| \\
&= |(p' - p(t)) \cdot (1 - (a \cdot p(t) \cdot d \cdot /2n))| \\
&= |(p' - p(t))| \cdot |(1 - (p(t) \cdot d \cdot (a/2n)))|
\end{aligned}$$

The same choice for n will work here.

We can extend the result to situations where $r1$ and $r2$, and by extension f, are not linear. We assume that f is continuous, and that df/dp is bounded above and below by d' and d'', the minimum and maximum derivatives, both finite and both strictly negative.

Again, let us go by cases, and assume that $p(t) < p'$. By our assumptions on f,

$$\begin{array}{lll} f(p) & >= & -d' \cdot (f(p') - f(p(t))) \\ f(p) & <= & -d'' \cdot (f(p') - f(p(t))) \end{array}$$

We can use this to bound the $a \cdot (1 - p(t)) \cdot f(p(t))$ term.

$$\begin{array}{lll} a \cdot (1 - p(t)) \cdot (f(p(t))/n) & => & a \cdot (1 - p(t)) \\ & & \cdot - d' \cdot (p' - p(t))/2n) \\ a \cdot (1 - p(t)) \cdot (f(p(t))/n) & =< & a \cdot (1 - p(t)) \\ & & \cdot - d'' \cdot (p' - p(t))/2n) \end{array}$$

If we pick n large enough, $(a \cdot (1 - p(t)) \cdot -d'' \cdot (p' - p(t))/2n) < p' - p(t)$, so

$$\begin{array}{lll} |p' - p(t+1)| & >= & |(p' - p(t)) \\ & & +a \cdot (1 - p(t)) \cdot -d' \cdot (p' - p(t))/2n)| \\ |p' - p(t+1)| & <= & |(p' - p(t)) \\ & & +a \cdot (1 - p(t)) \cdot -d'' \cdot (p' - p(t))/2n)| \\ |p' - p(t+1)| & >= & |(p' - p(t)) \cdot (1 - p(t)) \cdot d' \cdot (a/2n)| \\ |p' - p(t+1)| & <= & |(p' - p(t)) \cdot (1 - p(t)) \cdot d'' \cdot (a/2n)| \\ |p' - p(t+1)| & >= & |(p' - p(t))| \cdot |(1 - p(t)) \cdot d' \cdot (a/2n)| \\ |p' - p(t+1)| & <= & |(p' - p(t))| \cdot |(1 - p(t)) \cdot d'' \cdot (a/2n)| \end{array}$$

Again, by picking n large enough, we can easily establish a minimum and a maximum contraction factor, guaranteeing convergence.

So, in the case of an infinite number of agents and no delays in information, we have shown that adding properly chosen uniformly distributed heterogeneous transaction costs guarantees convergence to the equilibrium of the underlying system, guaranteeing maximum payoffs.

It is important to take a step back to think of the implications of this. With guaranteed convergence to the point where $f = 0$, the system is functioning as a decentralized root-finding algorithm. Root finding algorithms [6] are numerical methods used to solve equations, usually by moving all terms to one side and finding the point where $f(x) = 0$. One must usually make assumptions about the form of f, namely that it is monotonic, and sometimes continuous. But given these assumptions, root finding algorithms proceed by 'bracketing' the solution, and iteratively closing in on it by moving closer to it on each step; if the sign of $f(x)$ changes, then we have passed the root and must step backwards. The system here converges slowly (many root finding algorithms promise quadratic convergence), but conceptually it is very similar. This opens the question as to whether we can take more sophisticated root finding algorithms, and try to implement them in a decentralized manner by playing with the decision making processes of individual agents. For example, bisection, which merely halves the distance to the root each turn, could be implemented by merely halving the number of agents that can act each turn.

So in theory, if we have a problem that can be represented as solving an equation, we can set up a decentralized system that solves the problem with no loss in accuracy and guaranteed convergence.

5.3 Noise vs. Heterogeneity

There is nothing in the preceding analysis that distinguishes heterogeneous preferences and heterogeneous transaction costs from homogeneous preferences and transaction costs with an appropriate amount of uniformly distributed noise. The only difference is that in the heterogeneous case, the bias terms are generated once and fixed permanently. In the noise case, a new set of bias terms is generated each turn – but, because of the law of large numbers, the proportion of agents whose decisions are affected by the bias terms is the same.

The difference comes in two areas. For real systems with a finite number of agents, the exact terms produced by adding noise will vary each turn, potentially causing instability; in contrast, the heterogeneous bias terms do not vary over time. The law of large numbers makes this issue grow theoretically less important with the number of agents in the system, although it is probably a concern in any real world problem. Also, in applications where agents should stay with the same resources over time – for example, when there are real transaction costs – the noise case incurs suboptimal performance. Although the noise case maintains the equilibrium, it does so because the number of agents who switch from resource one to resources two exactly balances out those who switch the other way. In the heterogeneity case, no agents switch – at equilibrium, their inherent biases give them no incentive to switch, and no transaction costs are observed.

6 Theory & Practice

How do the theoretical and empirical results mesh together? For both the heterogeneous preferences and transaction cost cases, low levels of heterogeneity had little effect on the system. This probably corresponds to a situation where $f(p(t))$ never entered the stabilizing $[-2n, 2n]$ range. As n, and thus the stabilizing interval, grows big enough, the oscillations of the system dropped dramatically – in the heterogeneous transaction cost case, converging to a fixed point almost identical to that of the underlying system.

As n rises, in both cases the systems fell away from optimality; in the heterogeneous preferences case, this was because the system was stabilizing farther and farther away from the equilibrium, as predicted. The heterogeneous transaction cost case is more mysterious – the theory predicts that the value of n should have no effect on the fixed point of the biased system. This is probably an issue of granularity – since we don't really have a continuum of agents, there is some smallest transaction cost c'; and the system can potentially stabilize at any point such that $f(p(t)) < c'$. As n grows, this c' grows with it, thus allowing the system to stabilize farther and farther away from the true equilibrium. As the number of agents grows, thus better approximating a continuum, hopefully this would be less of an issue.

For empirical success, it seems that the goal is to pick an n big enough for the system to fall into the zone where $f(p(t)) \in [-2 \cdot n, 2 \cdot n]$, but not so big that the system stabilizes away from the underlying equilibrium. Fortunately, in the heterogeneous transaction cost case, the range of useful n is reasonable – anywhere in $[.8, 1.85]$ gives near-optimal results.

7 Conclusions & Future Work

This paper has explored the use of heterogeneity to stabilize decentralized systems, both in terms of preferences over resources, and transaction costs to switch between resources. In was shown that although in both cases heterogeneity can produce stability, in both practice and theory using heterogeneous transaction costs produces stability with less suboptimality than heterogeneous preferences. Given an continuum of agents, it was proved that biased transaction costs guarantee convergence to the true underlying equilibrium of the system, allowing us to conceptualize these systems as a simple decentralized root finding algorithms.

This work leaves many unanswered questions. The use of a uniform distribution to represent heterogeneity was motivated largely by concerns of ease of analysis; an exploration of the effects of Gaussian or other distributions of the exponential family might find them either theoretically or practically superior. The conceptualization of this sort of system as a decentralized root finder suggests tantalizing possibilities; could we design an algorithm to mimic Newton's method or even simulated annealing?

In economics terms, the knowledge that heterogeneous transaction costs stabilize these systems so well has intriguing potential policy implications. The recent discussion of using the Tobin tax [9, 8] (a small transaction cost on international currency trades) to encourage stability in global currency markets has raised the profile of transaction costs as a policy method; the work here suggests that making them heterogeneous may make them more effective at a lower social cost, although issues of fairness would need to be addressed.

References

[1] Brian Arthur. Inductive reasoning and bounded ratinality. *AEA Papers and Proceedings*, 84(2):406–411, 1994.

[2] John Q. Cheng and Michael P Wellman. The walras algorithm: A convergent distributed implementation of general equilibrium outcomes. Technical report, University of Michigan, 1995.

[3] Drew Fudenberg and Jean Tirole. *Game Theory*. MIT Press, 1991.

[4] Tad Hogg and Bernardo A. Huberman. Controlling chaos in distributed systems. *IEEE Trans. on Systems, Man and Cybernetics*, 21(6):1325–1332, November/December 1991.

[5] J. O. Kephart, Tad Hogg, and Bernardo A. Huberman. Dynamics of computational ecosystems. *Physical Review A*, 40, 1989.

[6] William H. Press, Saul A. Tukolsky, William T. Vertterling, and Brian P. Flannery. *Numerical Recipies in C*. Cambridge University Press, 1992.

[7] Andrea Schaerf, Yoav Shoham, and Moshe Tennenholtz. Adaptive load balancing: A study in multi-agent learning. *Journal of Artificial Intelligence Research*, (2):475–500, 1995.

[8] Staff. Contagious speculative attacks. Technical Report 22, Bank for International Settlements, 1994.

[9] James Tobin. A proposal for international monetary reform. *Eastern Economic Journal*, 4, 1978.