Balanced Matching of Buyers and Sellers in E-Marketplaces: The Barter Trade Exchange Model

Peter Haddawy
Namthip Rujikeadkumjorn
Khaimook Dhananaiyapergse
CSIM Program
Asian Institute of Technology
haddawy@ait.ac.th,
Namthip.Rujikeadkumjorn@ait.ac.th,
khaimook@chula.com

Christine Cheng
Dept. of EE&CS
University of Wisconsin-Milwaukee
ccheng@cs.uwm.edu

ABSTRACT

In this paper, we describe the operation of barter trade exchanges by identifying key techniques used by trade brokers to stimulate trade and satisfy member needs, and present algorithms to automate some of these techniques. In particular, we develop algorithms that emulate the practice of trade brokers by matching buyers and sellers in such a way that trade volume is maximized while the balance of trade is maintained as much as possible. We show that the buyer/seller matching and trade balance problems can be decoupled, permitting efficient solution as well as numerous options for matching strategies.

We model the trade balance problem as a minimum cost circulation problem (MCC) on a network. When the products have uniform cost or when the products can be traded in fractional units, we solve the problem exactly. Otherwise, we present a novel stochastic rounding algorithm that takes the fractional optimal solution to the trade balance problem and produces a valid integer solution. We then make use of a greedy heuristic that attempts to match buyers and sellers so that the average number of suppliers that a buyer must use to satisfy a given product need is minimized.

We present results on the empirical evaluation of our algorithms on test problems and simulations. Experiments show that our algorithm (MCC + stochastic rounding) runs in a fraction of the time of a commercial mixed integer programming (MIP) package while producing solutions that are always within 0.7% of the MIP solution. We evaluate the effectiveness of our algorithm on maintaining balance and on stimulating trade using two different simulation techniques, both based on transaction history data from a trade exchange. The simulation results support the barter trade exchange rule of thumb that maximizing single-period trade volume while maintaining balance of trade helps to maximize trade volume over the long run.

1. INTRODUCTION

With the movement of business to the Internet, one of the most popular e-commerce models to emerge has been that of the e-marketplace. An e-marketplace is an electronic intermediary that brings buyers and sellers together and provides various support services. These services have traditionally consisted of such things as e-catalogs, search capabilities, and transaction support. More recently researchers have sought to exploit the electronic infrastructure of e-marketplaces and the wealth of information that can be gathered in e-marketplaces to provide sophisticated methods of matching buyers and sellers, using agent-based, auction-based, and broker-based techniques. Of the work in this area that has addressed profitability of the e-marketplace, the overriding concern has been for maximization of single-period revenues [8, 17, 14], with less attention paid to how the techniques fit within a more strategic business model. But as the dramatic down-turn in the e-commerce sector demonstrated, e-business initiatives require solid business models that clearly relate the services provided to the overall profitability of the company [15]. In this paper, we take a particular but quite general e-marketplace business model as our point of departure and use that model to motivate the development of algorithms to support management of trade among buyers and sellers.

The model used in this paper is that of the barter trade exchange, also called retail or commercial barter. A barter trade exchange is a collection of businesses that trade their goods and services, managed by an intermediary. We call the collection of businesses the *barter pool* and call the intermediary the *trade exchange*. In modern barter trade exchanges, businesses do not exchange goods directly in the bilateral fashion of traditional barter. Rather, modern barter is multilateral, using a form of private label currency. The trade exchange issues trade dollars to the member businesses and acts as a neutral third party record keeper. When a company sells a good, they receive credit in trade dollars, which they can then use to purchase goods from other members. The value of the trade dollar is tied to the US dollar by not permitting businesses to charge more for their goods in terms of trade dollars than they do in US dollars in the open market, thus preventing devaluation of the currency.

The barter industry is interesting as a test bed for market design because a barter pool is a relatively closed economy about which we have very detailed information due to the book keeping function of the trade exchange. The trade exchange maintains a general profile for every member business, as well as complete records of all transactions between members. A barter pool has many similarities with a traditional economy, with the trade exchange playing a role analogous to that of the federal government in regulating the economy. The exchange controls such variables as monetary supply, interest rate, rate of commission (analogous to revenue tax), and even supply and demand through its ability to selectively recruit new member businesses. Interestingly, although it has control over all these parameters, the trade exchange works to stimulate the barter pool economy primarily by making referrals to member businesses through trade brokers.

The success and survivability of the barter business add to its attractiveness as a model to study. The barter trade exchange industry has existed for over forty years, surviving numerous changes in the economic landscape. The International Reciprocal Trade Association [4] estimated that the total value of products and services bartered by businesses through barter companies reached USD 7.87 billion in 2001. This number was an increase from USD 6.92 billion in 1999 and was the third consecutive year the industry saw over 12% growth. There were an estimated 719 trade companies active in North America in 1999 with some 471,000 client businesses [3]. Examples of active barter trade exchanges with a Web presence include BizXchange, ITEX, BarterCard, and Continental Trade Exchange.

The rest of this paper is organized as follows. In Section 2, we provide a description of the operation of barter trade exchanges, identifying key techniques used by trade brokers in order to stimulate trade and satisfy member needs. In this paper we focus on implementing techniques for maximizing single-period trade while maintaining balance of trade within the barter pool. In Section 3 we present a formalization of this problem and in Sections 4 and 5 we present novel efficient algorithms for its solution, using minimum cost circulations on networks and stochastic rounding techniques. Due to space limitations, some of the proofs are omitted. Omitted proofs can be found in the long version of this paper [10]. In Section 6, we present empirical evaluation of our algorithms. In Section 7 we discuss related work and show that the problem of maximizing trade while maintaining balance can be reduced to a multi-unit combinatoral auction problem. In Section 8 we present conclusions and directions for future work.

2. BARTER TRADE EXCHANGE MODEL

Given its important role in B2B commerce, there is a surprising lack of literature on the barter trade exchange industry. An exception is the work of Cresti, which examines theoretical economic rationale for development of the barter industry in industrialized countries [6], as well as investigating the macroeconomic variables influencing the industry in the United States [7]. But there exists no formal literature describing the barter trade exchange industry on an operational level. Our interest lies in understanding how managers and brokers in a trade exchange manage the operations of the exchange in order to maximize their company's profits. Thus our first step in conducting this work was to gather information through extensive interviews with industry experts. We also communicated with them periodically to verify the assumptions behind our models. We interviewed two executives at BizXchange (www.bizx.bz), a relatively new but rapidly growing trade exchange located in the San Francisco Bay and Seattle areas. Since its inception in January 2002, BizXchange has grown to include over 600 member businesses. The two executives we interviewed have over 28 years of combined industry experience, have founded and built several successful barter networks, and have served on the Boards of the International Reciprocal Trade Association and the National Association of Trade Exchanges.

A barter pool can be viewed as a carefully managed small-scale economy. Managers of trade exchanges attempt to recruit member businesses in such a way that supply and demand for each product category in the pool are approximately balanced.[1] Member businesses are typically small to medium size enterprises that offer products and/or services. They fall into the broad categories of operating expenses, employee benefits, and travel and entertainment. Henceforth we will use the term *goods* to refer to goods and services.

It is a common misconception that the primary benefit of barter is to avoid taxes. In fact, the US Tax Equity and Fiscal Responsibility Act, passed in 1982, legislated that barter income be treated as equivalent to cash income and taxed on the same basis. Cresti [7] shows empirical evidence that barter is adopted to increase profits and gain a competitive edge and that barter is, in fact, complementary to the cash economy.

When a business joins a trade exchange, it typically pays a membership fee. This represents a small fraction of the revenues of the trade exchange. The primary revenue is made by charging a fee to the buyer and seller on each transaction. The fee is typically in the range of 6 - 7.5% and is payable in US dollars. When a business joins the trade exchange, they are issued a line of credit in trade dollars, which permits them to make purchases without first having to sell and also gives them flexibility in conducting transactions. The trade exchange charges interest on negative balances, usually at the same rate as major credit cards. In order to give a company some control over how much of their profits are accrued in terms of trade dollars, the trade exchange permits the member to set an upper limit on the amount of trade dollars they are willing to accumulate. The credit line and upper limit define the *financial operating range* of the business within the barter pool.

[1]Although managers *attempt* to keep supply and demand balanced, it is not the case that they are, in fact, balanced at any point in time. Therefore, in this paper we do not assume that supply and demand are equal or even near-equal.

Each member is assigned to a trade broker. A broker typically represents a set of 150 - 200 client businesses. The broker's job from the standpoint of the client is to help the client sell his goods and to inform him of goods he might like to buy. The broker's job from the standpoint of the trade exchange is to stimulate trade, since the exchange's revenues are directly tied to trade volume. The broker stimulates trade by working to help clients spend their trade dollars when they have positive balance and generate sales when they have negative balance. The broker's primary tool is the referral, referring potential buyers to suppliers. Note that member businesses are under no obligation to follow the broker's referrals, but experience from trade exchanges shows that they generally do. While the goods a business has to sell are stated explicitly, those that the business wants to buy may be explicitly stated or may be predicted by the broker based on things like the type of business and other goods that the business has purchased in the past.

In carrying out his job, the broker attempts to maximize single-period trade volume while maintaining balance of trade. Trade is *balanced* when the total dollar amount each member buys equals what it sells. For any supply/demand of a business there will typically be several different buyers/sellers to choose from. The question is then which of those buyers/sellers the broker should refer the business to. Most trade exchanges do not make this decision based on price; they leave price negotiation up to the members. Other factors such as convenience of location being equal, the broker maintains balance by basing his decision on the balances of the buyers/sellers. For example, if we have one supplier who has highly positive balance and one who has highly negative balance, the broker will refer the client to the supplier with highly negative balance.

This concludes our description of the operation of barter trade exchanges. In the remainder of the paper we present a mathematical formalization of the problem of maximizing trade while maintaining balance and develop efficient[2] algorithms for its solution. Algorithmic efficiency is important if we wish to be able to scale up to handle large real-world problems.

We assume that trade occurs in business cycles: first businesses' supplies and demands are determined, a matching is found, the businesses act on the resulting referrals, and the cycle repeats. The matching problem can be represented by a requirements matrix in which each row represents a member business, each column represents a category of goods, and matrix entries represent quantities to buy or sell. Each business can buy and/or sell multiple goods and we make no assumptions about the relationship between supply and demand in the barter pool. We assume a uniform unit cost for each product category. Since prices are naturally not uniform across suppliers, this unit cost can be an estimate based on the average retail price over suppliers in that category. The internal state of each business is characterized by its current balance, its credit line, and an upper limit on allowed balance. Once a business reaches its credit limit, then it can no longer buy without selling, so its row in the matrix will show no demand. Similarly, when a company reaches its upper limit, it can no longer sell without buying, so its row will show no supply.

[2] Algorithm efficiency, which refers to the algorithm's running time, should not be confused with economic efficiency.

3. THE BARTER UNIVERSE

In barter universe $\mathcal{B} = (C, P, \alpha, R)$, $C = \{c_i, i = 1, \ldots, m\}$ denotes the set of companies, and $P = \{p_j, j = 1, \ldots, n\}$ denotes the set of goods. For each j, the unit cost of good p_j is α_j barter dollars. The $m \times n$ *requirements matrix* R specifies the number of units each company is willing to supply or purchase for each good. We shall assume that R is an integer matrix, and use the convention that if c_i is a *seller* of p_j then $R_{ij} > 0$, and if c_i is a *buyer* of p_j then $R_{ij} < 0$, and if c_i is not interested in p_j then $R_{ij} = 0$.

A *trade set* in the barter universe is specified by an $m \times n$ matrix T, where T_{ij} indicates the number of units of p_j that c_i sold or bought in the trade. There are two conditions T must satisfy for each j:

i. If c_i is a seller of p_j then $0 \le T_{ij} \le R_{ij}$, and if c_i is a buyer of p_j then $0 \ge T_{ij} \ge R_{ij}$, and if c_i is not interested in p_j then $T_{ij} = 0$.

ii. $\sum_{i=1}^{m} T_{ij} = 0$; i.e., for each p_j, the total number of units sold equals the total number of units bought.

The *volume* of trade set T is $vol(T) = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{m}|T_{ij}|$, the total number of units of all the goods exchanged in the trade. Throughout this paper, we will consider *maximal trade sets*, which are trade sets with the largest volumes. For each j, let S_j and N_j denote, respectively, the *sellers* and *buyers* of p_j. If the supply of p_j exceeds its demand then a maximal trade set T will have $T_{ij} = R_{ij}$ for all $c_i \in N_j$. Similarly, if the supply of p_j is no more than its demand then $T_{ij} = R_{ij}$ for all $c_i \in S_j$. Hence, the number of units traded for each p_j is maximized, and every maximal trade set has volume equal to $\sum_{j=1}^{n} \min\{\sum_{i \in S_j} R_{ij}, \sum_{i \in N_j} |R_{ij}|\}$. We note that it is straightforward to find a maximal trade set for $\mathcal{B}$ in time $O(|P||C|)$.

4. FINDING THE MOST BALANCED MAXIMAL TRADE SET

After the trade specified by T takes place, we define the *balance of* c_i as $b_i = \sum_{j=1}^{n} \alpha_j T_{ij}$, and the *absolute balance due to* T as $ab_T = \sum_{i=1}^{m} |b_i|$. Our goal is to solve for the maximal trade set T^* that is *most balanced*; i.e., $ab_{T^*} \le ab_T$ for every other maximal trade set T. The next lemma follows directly from the fact that $\sum_{i=1}^{m} T_{ij} = 0$ for each j.

LEMMA 4.1. *For any trade set* T, $\sum_{i=1}^{m} b_i = 0$.

The problem of finding the most balanced maximal trade set can be formulated as an integer program. Thus, one approach we can take is to first relax the condition that the entries of a trade set T be integers. The integer program reduces to a linear program, and the latter can be solved optimally in polynomial time. We can then use some procedure to transform a fractional solution to an integer one. In this section, we shall take a similar approach. However, instead of solving the relaxed problem as a linear program, we shall show in the next subsection that it can be reduced to a minimum-cost circulation (MCC) problem on an appropriate network, and that it can be solved optimally by running a bounded number of breadth-first searches because of the network's simple structure. In the subsequent section, we will present a novel randomized rounding procedure that produces an integer maximal trade set T' from a fractional trade set T so that the expected balance of a company in T' is equal to its balance in T. By enforcing this constraint, we hope that $ab_{T'}$ will be close to ab_T. In Section 6, our simulations indicate that the MCC solutions combined with the randomized rounding procedure are as competitive as the solutions obtained from a commercial mixed-integer programming (MIP) package. Furthermore, in all but one instance, our approach is significantly faster than the MIP package.

4.1 When maximal trade sets can be fractional

We can solve for the most balanced fractional maximal trade by first constructing a network, and then finding the minimum cost circulation on the network. (We refer the readers to [1] for an excellent introduction to the field of network flows.) Let us start with the following example. Suppose there are four companies and two goods which costs \$2 and \$3 respectively. Below is a requirements matrix R:

	p_1(\$2)	p_2(\$3)
c_1	50	20
c_2	70	−30
c_3	−20	15
c_4	−10	−40

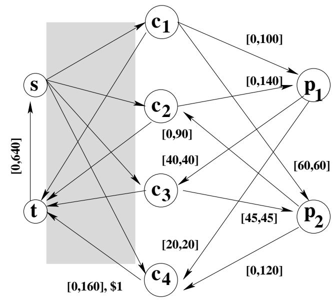

Figure 1: The network that corresponds to the barter universe in the example.

The corresponding network is shown in Figure 1. We only indicated the edge costs when they are non-zero. There is a directed edge from c_i to p_j (or p_j to c_i) if c_i is a seller (or buyer) of p_j. The lower and upper bound capacities of an edge between c_i and p_j indicate the minimum and maximum *total cost* of the units of p_j that c_i can sell or buy. The edges (s, c_i) and (c_i, t) will be used to indicate the value of b_i if $b_i > 0$ and if $b_i < 0$ respectively. The upper bound capacities on all edges between $\{s, t\}$ and nodes in C are set to 160 because this is the largest balance any of the companies can have at the end of a trade set. Finally, the edge (t, s) has an upper bound capacity of $160 \times 4 = 640$ because the total balance of all the companies can never exceed this value.

For an arbitrary $\mathcal{B}$, we shall create the network as follows. Let $G = (V, E)$ be a directed graph where

$$\begin{aligned} V &= C \cup P \cup \{s, t\}, \\ E &= \{(p_j, c_i), \forall c_i \in N_j, \forall j\} \cup \{(c_i, p_j), \forall c_i \in S_j, \forall j\} \\ &\quad \cup \{(s, c_i), \forall c_i \in C\} \cup \{(c_i, t), \forall c_i \in C\} \cup \{(t, s)\}. \end{aligned}$$

Let $l : E \to \mathbf{R}$ and $u : E \to \mathbf{R}$ denote the lower and upper bound capacities on the edges of G. For each j, if the supply of p_j exceeds its demand then let all edges $e = (p_j, c_i)$, where $c_i \in N_j$ have $l(e) = u(e) = \alpha_j |R_{ij}|$, but let all edges $e' = (c_{i'}, p_j)$ where $c_{i'} \in S_j$ have $l(e) = 0$ and $u(e) = \alpha_j R_{ij}$. The capacities are chosen because, in a fractional maximal trade set, all buyers of p_j will have to purchase all their demands for p_j, but the sellers of p_j need not sell all their supplies of p_j. When the supply for p_j is no more than its demand then we do the opposite. Let all edges $e' = (c_{i'}, p_j)$ where $c_{i'} \in S_j$ have $l(e) = u(e) = \alpha_j R_{ij}$, and let all edges $e = (p_j, c_i)$, where $c_i \in N_j$ have $l(e) = 0$ and $u(e) = \alpha_j |R_{ij}|$. Let the edges (s, c_i) and (c_i, t), for all c_i, and (t, s) have lower and upper bound capacities of 0 and ∞ (or a large number) respectively. Finally, let $\gamma : E \to \mathbf{R}$ be the cost function for the edges of G. For each c_i, set $\gamma(s, c_i) = \gamma(c_i, t) = \$1$, and set all other edge costs to \$0. Denote this network as (G, l, u, γ).

A *circulation* $f : E \to \mathbf{R}$ on the network assigns a *flow value* to each edge. We say that the circulation f is *feasible* if (a) for each $e \in E$, $l(e) \le f(e) \le u(e)$, and (b) for each $v \in V$, the flow into v equals the flow out of v. The *cost of* f is $\sum_{e \in E} \gamma(e) f(e)$. Let us now show that there is a correspondence between the fractional maximal trade sets for $\mathcal{B}$ and the feasible circulations for (G, l, u, γ).

Fractional maximal trade sets to feasible circulations. Let T be a fractional maximal trade set for $\mathcal{B}$. Construct the corresponding circulation f on (G, l, u, γ) as follows:

i. For each c_i and p_j, if $c_i \in S_j$, set $f(c_i, p_j) = \alpha_j T_{ij}$. If $c_i \in N_j$, set $f(p_j, c_i) = \alpha_j |T_{ij}|$.

ii. For each company c_i, if $b_i \ge 0$, let $f(s, c_i) = b_i$; otherwise, let $f(c_i, t) = |b_i|$.

iii. Let $f(t, s) = \sum_{b_i < 0} |b_i|$.

iv. Let all other edges have a flow of 0.

LEMMA 4.2. *The flow f has cost ab_T, and is a a feasible circulation on (G, l, u, γ).*

Proof: Step (i) ensures the that net flow out of p_j is

$$\sum_{i:T_{ij}<0} \alpha_j |T_{ij}| - \sum_{i:T_{ij}>0} \alpha_j T_{ij} = -\sum_{i=1}^{m} \alpha_j T_{ij} = 0$$

because T is a valid trade set for $\mathcal{B}$. Hence, flow is conserved at each p_j. From the same step, the net flow out of each c_i is $\sum_{j:T_{ij}>0} \alpha_j T_{ij} - \sum_{j:T_{ij}<0} \alpha_j |T_{ij}| = b_i$. Step (ii) forces $f(c_i, t) - f(s, c_i) = -b_i$ so flow is conserved at c_i, for each i. Since $\sum_{i=1}^{n} f(s, c_i) = \sum_{b_i > 0} b_i = \sum_{i=1}^{n} f(c_i, t)$ because of step (ii) and Lemma 4.1, step (iii) ensures flow conservation at nodes s and t. Finally, it is easy to verify that the flows on all the edges satisfy the upper and lower bound constraints so the circulation we have constructed is feasible. Its cost is $\sum_{i=1}^{m} f(s, c_i) + \sum_{i=1}^{m} f(c_i, t) = \sum_{i=1}^{m} |b_i| = ab_T$. □

Feasible circulations to fractional maximal trade sets. Given a feasible integer circulation f on (G, l, u, γ), for each c_i and p_j, if $(c_i, p_j) \in E$, set $T_{ij} = f(c_i, p_j)/\alpha_j$. If $(p_j, c_i) \in E$, set $T_{ij} = -f(p_j, c_i)/\alpha_j$. Otherwise, set $T_{ij} = 0$.

LEMMA 4.3. *The resulting fractional trade set T is maximal for $\mathcal{B}$.*

Proof: Because of the way the lower and upper bound capacities are assigned on the edges of the network, and because f is a feasible circulation, T is a valid trade set. When the supply of p_j exceeds its demand, for each $c_i \in N_j$, $T_{ij} = R_{ij}$ because $l(p_j, c_i) = u(p_j, c_i) = \alpha_j |R_{ij}|$. Similarly, when the supply of p_j does not exceed its demand then, for each $c_i \in S_j$, $T_{ij} = R_{ij}$. Thus, the volume of T is maximized so it is a fractional maximal trade set for $\mathcal{B}$. □

THEOREM 4.4. *Let f^* be a minimum cost circulation for (G, l, u, γ). The fractional maximal trade set T^* resulting from the transformation of f^* is a most balanced fractional maximal trade set for $\mathcal{B}$ whose cost is ab_{T^*}.*

Proof: At each node c_i, we have

$$\sum_{e'=(c_i,p_j)\in E} f^*(e') - \sum_{e=(p_j,c_i)\in E} f^*(e) = b_i,$$

the balance of company c_i in T^*. Since f^* is a feasible circulation, $f^*(c_i,t) - f^*(s,c_i) = -b_i$. If $b_i \geq 0$ then $f^*(c_i,t) = 0$ and $f^*(s,c_i) = b_i$; otherwise, we can decrease the flow on the three edges (c_i,t), (s,c_i) and (t,s) by $f^*(c_i,t)$, and the new circulation will have a cost less than that of f^*. Similarly, if $b_i < 0$ then $f^*(c_i,t) = -b_i$ and $f(s,c_i) = 0$. Consequently, the cost of f^* is $\sum_{b_i>0} b_i + \sum_{b_i<0} |b_i| = ab_{T^*}$.

Let T be any fractional maximal trade and f be its corresponding circulation on (G, l, u, γ) as defined above. Since f^* is an MCC for (G, l, u, γ), $\gamma(f^*) \leq \gamma(f)$ so $ab_{T^*} \leq ab_T$. Therefore, T^* is a most balanced fractional maximal trade set in (G, l, u, γ). □

Solving MCC in (G, l, u, γ) efficiently. Given a network (G, l, u, γ) and a feasible circulation f on the network, its *residual network*, $R(G, f)$, has the same vertex set as G. The edges also have costs (specified by γ_R), and upper and lower bound capacities (specified by u_R and l_R). An edge e of G is in the residual network if and only if $f(e) < u(e)$. Furthermore, $u_R(e) = u(e) - f(e)$, $l_R(e) = 0$, and $\gamma_R(e) = \gamma(e)$. The reverse of e, which we shall denote as $\bar{e}$, is in the residual network if and only if $l(e) < f(e)$, and $u_R(\bar{e}) = f(e) - l(e)$, $l_R(\bar{e}) = 0$, and $\gamma_R(\bar{e}) = -\gamma(e)$.

To solve for the MCC in (G, l, u, γ), we make use of the minimum mean cycle-canceling algorithm. The algorithm starts with a feasible circulation f. At each iteration, it finds a minimum mean (directed) cycle[3] in the residual graph $R(G, f)$. If the cycle has negative cost, then the algorithm augments f using this cycle. (Step 2 of algorithm BALANCE specifies this step precisely.) Otherwise, $R(G, f)$ contains no negative-cost directed cycle, and the algorithm outputs f, which must be an MCC. Goldberg and Tarjan [9] showed that the number of augmentations is $O(|V||E|)$. We note that there are other algorithms for MCC whose theoretical guarantees are better than the minimum mean cycle-canceling algorithm (e.g., see reference in [1, 9]). We shall show, however, that this algorithm is very simple to implement for our network. Given a circulation f in (G, l, u, γ), the negative-cost directed cycles in $R(G, f)$ have a nice structure. As a result, the search for the minimum mean cycle in $R(G, f)$ becomes straightforward.

LEMMA 4.5. *All negative-cost directed cycles in $R(G, f)$ cost $-\$2$ and consist of a sequence of nodes of the form $t, c_{i_1}, p_{j_1}, c_{i_2}, p_{j_2}, \ldots, p_{j_k}, c_{i_{k+1}}, s, t$. (Proof: See [10].)*

We are now ready to present our algorithm.

BALANCE(R)

1. Find an initial maximal trade set T. Construct the network (G, l, u, γ) and the corresponding circulation f of T.

2. Construct the residual graph $R(G, f)$. Determine if $R(G, f)$ contains a negative-cost cycle as follows. In $R(G, f)$, remove (t, s) and all the edges with a cost of \$1, and find the shortest path from t to s using breadth first search.

   If no such path exists, go to Step 3. Else, append the edge (s, t) to the path to form a negative-cost cycle Y. Augment f using Y. (That is, let $\epsilon \leftarrow \min\{u(e) - f(e),$ for all forward edges e on $Y\} \cup \{f(e) - l(e)$ for all backward edges e on $Y\}$. For all edges e in Y, if e is a forward edge, $f(e) \leftarrow f(e) + \epsilon$; if e is a backward edge, $f(e) \leftarrow f(e) - \epsilon$.) Repeat Step 2.

3. Construct the trade set T equivalent to the circulation f. Return(T).

THEOREM 4.6. *BALANCE outputs a most balanced fractional maximal trade set in $O((|V| + |E|)|V||E|)$ time where $|V| = O(|P| + |C|)$ and $|E| = O(\sum_{j=1}^{n}(|S_j| + |N_j|))$.*

Proof: If we can show that Step 2 of BALANCE correctly finds the minimum mean cycle in $R(G, f)$ then BALANCE is essentially an implementation of the minimum mean cycle-canceling algorithm for (G, l, u, γ). Since it outputs the fractional maximal trade set that corresponds to the MCC in (G, l, u, γ), according to Theorem 4.4, the trade set must be most balanced.

Consider $R'(G, f)$, constructed by deleting edge (t, s) and all edges with cost \$1 in $R(G, f)$. From Lemma 4.5, $R(G, f)$ contains a negative-cost directed cycle if and only if $R'(G, f)$ has a directed path from t to s. Step 2 finds the shortest $t-s$ path in $R'(G, f)$ so the negative-cost directed cycle Y has the fewest number of edges. Since all negative-cost directed cycles in $R(G, f)$ have the same cost, Y is a minimum mean cycle in $R(G, f)$.

Doing a breadth-first search in $R(G, f)$ takes $O(|V| + |E|)$ time. Augmenting the flow on Y takes $O(|E|)$ time. According to Goldberg and Tarjan, the number of augmentations of BALANCE is $O(|V||E|)$ so step 2 takes $O((|V| + |E|)|V||E|)$ time where $|V| = O(|P| + |C|)$ and $|E| = O(\sum_{j=1}^{n}(|S_j| + |N_j|))$. Steps 1 and 3 take $O(|P||C|) = O(|V|^2)$ time. Therefore, the overall running time of BALANCE is $O((|V| + |E|)|V||E|)$ time. □

An extended example of the algorithm is contained in [10]. We note that, in practice, $|E|$ is likely going to be small compared to $|P| \times |C|$. BizXchange, for example, currently has

[3] If Y is a directed cycle, then its *mean cost* is $\sum_{e\in Y} \text{cost}(e)/|Y|$. The *minimum mean (directed) cycle* is the cycle with the smallest mean cost.

600 member companies and 125 product categories. Companies typically only sell between one to three goods and buy between five to 25 goods. Furthermore, as the trade exchange grows, the number of companies grows but the number of goods remains relatively constant.

4.2 When maximal trade sets must be integral

Since the flows on the edges between the company and good nodes need not be multiples of the goods' costs, the trade set T^* in Theorem 4.4 may contain non-integer values except when all the goods have $\alpha_j = 1$. But the maximal trade set that minimizes ab_T when $\alpha_j = 1$ for all j also minimizes ab_T when $\alpha_j = \alpha$, α a constant, so the statement below is true.

THEOREM 4.7. *Assuming all the goods in $\mathcal{B}$ have the same cost α, a most balanced (integer) maximal trade set can be found in $O((|V|+|E|)|V||E|)$ time where $|V| = O(|P|+|C|)$ and $|E| = O(\sum_{j=1}^{n}(|S_j|+|N_j|))$.*

Rounding Fractional Maximal Trade Sets. Given a fractional maximal trade set T, we wish to transform it into an integer maximal trade set T' so that $ab_{T'}$ is approximately equal to ab_T. In this section, we shall try to achieve this goal by employing a randomized rounding procedure that sets each T'_{ij} equal to $\lceil T_{ij} \rceil$ or $\lfloor T_{ij} \rfloor$ so that (i) for each j, $\sum_{i=1}^{m} T'_{ij} = 0$, and (ii) for each i, $E[b'_i] = E[\sum_{j=1}^{n} \alpha_j T'_{ij}] = b_i$. In the process of rounding T, condition (i) guarantees that the number of units of p_j sold remains equal to the number of units of p_j bought, while condition (ii) states that the *expected balance* of each company in T' is equal to its balance in T. It is, of course, possible for $|b'_i|$ to differ significantly from $|b_i|$ so that $ab_{T'} = \sum_{i=1}^{m} |b'_i|$ is much larger than $\sum_{i=1}^{m} |b_i| = ab_T$. We hope, however, that if we generate several integer maximal trade sets, one of them will have an absolute balance close to ab_T. First, we make the following observation:

LEMMA 4.8. *Let T be a fractional maximal trade set for the barter universe $\mathcal{B} = (C, P, \alpha, R)$. Suppose T' is an integer matrix obtained by setting each T'_{ij} equal to $\lceil T_{ij} \rceil$ or $\lfloor T_{ij} \rfloor$ so that $\sum_{i=1}^{m} T'_{ij} = \sum_{i=1}^{m} T_{ij} = 0$ for each j. Then T' is an integer maximal trade set for $\mathcal{B}$ and $ab_{T'} \leq ab_T + m\sum_{j=1}^{n} \alpha_j$. (Proof: See [10].)*

Without loss of generality, assume that, in the jth column of the fractional maximal trade set T, exactly k entries are not integers. The fractional entries have to be all positive if the supply of p_j exceeds its demand, and all negative otherwise. For $i = 1, \ldots, m$, let $x_i = T_{ij} - \lfloor T_{ij} \rfloor$ if $T_{ij} \geq 0$, and $\lceil T_{ij} \rceil - T_{ij}$ if $T_{ij} < 0$, so that $0 \leq x_i \leq 1$. Since $\sum_{i=1}^{m} T_{ij} = 0$, the fractional parts of the T_{ij}'s must also sum up to an integer; i.e., $\sum_{i=1}^{m} x_i = z$ for some positive integer $z < k$. Setting $T'_{ij} = \lceil T_{ij} \rceil$ or $\lfloor T_{ij} \rfloor$ so that $\sum_{i=1}^{m} T'_{ij} = 0$ is equivalent to setting $x'_i = \lceil x_i \rceil$ or $\lfloor x_i \rfloor$ so that $\sum_{i=1}^{m} x'_i = z$. The latter can be done by choosing exactly z out of the k fractional x_i's, and rounding them all to 1, and rounding the remaining $k - z$ x_i's to 0. Thus, creating an integer maximal trade set T' with the properties described in the previous lemma can be done in $O(|P||C|)$ time. We would like to choose a rounding of the x_i's, however, in a randomized manner so that $Prob(x'_i = 1) = x_i$ for each i. To do so, we make use of the following lemma which is very likely known.

LEMMA 4.9. *Let m and z be positive integers with $z \leq m$. Let $\mathcal{V}(m, z)$ be the set of all vectors in $\mathbf{R}^m$ that have z components equal to 1 and $m - z$ components equal to 0. (Note that $\mathcal{V}(m, z)$ contains $\binom{m}{z}$ vectors.) Let $\vec{x} = (x_1, \ldots, x_m) \in [0,1]^m$ so that $\sum_{i=1}^{m} x_i = z$. Then $\vec{x}$ can be expressed as a convex combination of the elements in $\mathcal{V}(m, z)$. That is, $\vec{x} = \sum_{\vec{v}_r \in \mathcal{V}(m,z)} \lambda_r \vec{v}_r$ so that $0 \leq \lambda_r \leq 1$ for each r, and $\sum_r \lambda_r = 1$. (Proof: See [10].)*

In the constructive proof of Lemma 4.9 given by Volkmer [18], the convex combination of $\vec{x}$ contains at most k vectors from $\mathcal{V}(m, z)$, where k is the number of fractional entries in $\vec{x}$. The construction takes $O(km)$ time. We note that $\mathcal{V}(m, z)$ contains all the possible roundings of $(x_1, \ldots, x_m)$. Let us use the coefficients in the convex combination of $\vec{x}$ as probabilities over the vectors in $\mathcal{V}(m, z)$. That is, for each $\vec{v}_r \in \mathcal{V}(m, z)$, define $Prob(\vec{v}_r) = \lambda_r$. Choose one vector $\vec{u} = (u_1, u_2, \ldots, u_m)$ in $\mathcal{V}(m, z)$ at random using this probability distribution. For each i, $Prob(u_i = 1) = \sum_{v_{r,i}=1} Prob(\vec{v}_r) = \sum_{r:v_{r,i}=1} \lambda_r = x_i$, where the last equality is true because $x_i = \sum_{r:v_{r,i}=1} \lambda_r v_{r,i}$. We are now ready to present our randomized rounding procedure.

RANDOM_ROUND(T)
For $j = 1$ to n, do:

1. For $i = 1$ to m, if $T_{ij} \geq 0$, set $x_i = T_{ij} - \lfloor T_{ij} \rfloor$. Else, set $x_i = \lceil T_{ij} \rceil - T_{ij}$. Set $z = \sum_{i=1}^{m} x_i$.

2. Express $\vec{x} = (x_1, \ldots, x_m)$ as a convex combination of the vectors in $\mathcal{V}(m, z)$ using CONVEX_COMB [4]. For each $\vec{v}_r \in \mathcal{V}(m, z)$, set $Prob(\vec{v}_r)$ equal to the coefficient of $\vec{v}_r$ in the convex combination of $\vec{x}$.

3. Pick one vector $\vec{u}$ randomly from $\mathcal{V}(m, z)$ using the probability distribution in step 2. For $i = 1$ to m, if $T_{ij} \geq 0$, set $T'_{ij} = \lfloor T_{ij} \rfloor + u_i$. Else, set $T'_{ij} = \lceil T_{ij} \rceil - u_i$.

Return(T').

THEOREM 4.10. *RANDOM_ROUND(T) outputs an integer maximal trade set T' such that for each i and j, $E[T'_{ij}] = T_{ij}$. Thus, for each i, $E[b'_i] = b_i$ and $ab_{T'} \leq ab_T + m\sum_{j=1}^{n} \alpha_j$. The procedure runs in $O(K|C|+|P||C|)$ time where K is the number of fractional entries in T. (Proof: See [10].)*

5. MATCHING BUYERS AND SELLERS

Once a desirable maximal trade set T has been found, buyers and sellers for each good have to be matched, and the number of units traded between them specified. This step generates the actual referrals. We make use of a greedy heuristic that attempts to minimize the average number of sellers matched to a buyer per good by always matching the company with the largest supply with the company with the largest demand. Minimizing the number of sellers is important since most businesses would rather not deal with too many suppliers to fulfill a particular need.

GREEDY_MATCHING (T)
For $j = 1$ to n, do:

1. Initialize $S_j \leftarrow \{c_i : T_{ij} > 0\}$, $N_j \leftarrow \{c_i : T_{ij} < 0\}$, and $M_j \leftarrow \emptyset$.

2. While ($N_j \neq \emptyset$), do

   Find a and b so that $T_{aj} = \max\{T_{ij}, c_i \in S_j\}$ and $T_{bj} = \min\{T_{ij}, c_i \in N_j\}$. Let $\epsilon \leftarrow \min\{T_{aj}, |T_{bj}|\}$. Add $\{(a, b, \epsilon)\}$ to M_j.

[4] The pseudocode for CONVEX_COMB can be found in [10].

If $T_{aj} - \epsilon = 0$, set $S_j \leftarrow S_j - \{c_a\}$. Else, set $T_{aj} \leftarrow T_{aj} - \epsilon$. If $|T_{bj}| - \epsilon = 0$, set $N_j \leftarrow N_j - \{c_b\}$ Else, set $T_{bj} \leftarrow T_{bj} + \epsilon$.

3. Return(M_j).

It is straightforward to verify the following lemma:

LEMMA 5.1. *GREEDY_MATCHING outputs a valid matching for each good p_j. It runs in $O(\sum_{j=1}^{n}(|S_j| + |N_j|)^2)$ time.*

6. EMPIRICAL EVALUATION

We performed three different sets of experiments in order to evaluate the effectiveness and efficiency of our algorithms. The experiments included (i) evaluation of running time and solution quality on a variety of requirements matrices, (ii) comparison of trade volumes using our algorithm with those from an operating trade exchange, and (iii) evaluation of the effectiveness of our algorithm in increasing trade volume by using a simulator, where we were able to vary parameters of the barter pool.

First, we evauated the quality of solutions and running time from the combination of our BALANCE and RANDOM_ROUND algorithms. We will refer to the combination of the two algorithms as B+RR. We compared the results from B+RR to those from a commercial mixed integer programming (MIP) package on 100 randomly generated matrices. We varied two parameters: product price range and current balance range, taking uniformly distributed random values over the ranges and producing ten matrices for each combination of parameter settings. Since a barter pool represents a closed economy, the sum of the starting balances was always zero. Each matrix represented 100 companies and 50 products. This problem size was chosen because its specification was the largest file size that the MIP package would accept. Supply and demand were randomly generated in such a way that their distributions mirrored those in the BizXchange barter pool. The RANDOM_ROUND algorithm was run 10 times for each matrix and the best value was selected. Increasing the number of runs to 50 made little difference in the results. Results are shown in Table 1. For each matrix, the degree of sub-optimality of the B+RR solution was computed as the difference between the value produced by B+RR and the value produced by the MIP package, and expressed as a percentage of the range of possible values of absolute balance for that matrix. It is necessary to express the degree of sub-optimality relative to the range of possible values since a small absolute difference in solutions is more siginicant if the range of solutions is small than if it is large. Since we have no algorithm for exactly solving the integer problem, we had to use our best estimate of the range of possible values of the absolute balance. For the minimum value, we took the lesser of the MIP and B+RR solutions. For the maximum value, we used the MIP package, setting the objective to maximize absolute balance. Note that this approach may underestimate the width of the range but will not overestimate it since all solutions are valid but not guaranteed to be optimal. Thus we are never overestimating the quality of the B+RR solution. Negative values for degree of sub-optimality indicate that B+RR produced a better solution than MIP. The solution from B+RR is always within 0.7% of the MIP solution, with little variance. Running times for B+RR and for the MIP package, as well as their standard deviations are shown in the last four columns of the table. Running times for B+RR are the sum of the running times for BALANCE and ten runs of RANDOM_ROUND. In every case but one, B+RR has significantly better running time and smaller variance in running time than MIP. To illustrate how well our algorithm scales up, we ran it on ten matrices with 1000 companies and 200 products, with product price range \$10 to \$100 and current balance -\$4,000 to \$10,000. On these large problems, BALANCE took on average 68.05 sec and RANDOM_ROUND took 0.55 sec, for a total running time of only 68.6 sec.

For our second and third sets of experiments, we obtained 66 weeks of transaction history data from BizXchange. The data specified the buyer, seller, date, amount, and product category for each transaction. Each member business supplied only one product category. The data included each company's credit line, which ranged from \$500 to \$20,000, and the company's upper bound on balance, which ranged from \$10,000 to \$50,000. The number of companies started at only 4 in week one and steadily grew to 264 by week 66. The total number of transactions was 1,887. The average number of suppliers per product category in which there was at least one buyer was 4.8 and the average number of buyers per category was 1.9.

The purpose of the second set of experiments was to compare trade volume following the buyer/seller matching produced by our optimizer with that in the BixXchange barter pool. Since we did not know the exact demand in a given week, we took the actual purchases companies made in a given week as representing their demand. This provided each company's demand in the weekly requirements matrix. The supply for each company in its product category was determined by the difference between the company's current balance and it's upper bound. We then used the optimizer to match buyers and sellers in each week and assumed that the companies made purchases following the determined optimal matching. We did the same thing without optimization, simply randomly choosing any maximal trade set, not necessarily the most balanced one. Comparison of the trade volumes in the optimized and non-optimized simulations with actual trade volumes are shown in Table 2. Comparison over the entire 66 weeks shows the optimized simulation closer to the actual trade volume than the non-optimized by 6%. Analysis of the trace of weekly trade volumes shows that the trade volumes are identical or very close for all three through week 48. (By week 48, the number of companies in the barter pool has grown to 239.) So we also compared the trade volumes over only weeks 48 - 66, shown in the second row. Here we see a greater difference (16%) between optimized and non-optimized trade volumes. Presumably if we had yet more data, this trend would continue. As the table shows, there is a notable difference between the actual trade volume and the trade volume of the optimized simulation. This is likely due to the fact that the trade brokers are considering more factors than balance when matching buyers and sellers, e.g. the rate at which companies tend to spend their trade dollars. It should be noted that in this simulation experiment it is not possible for the simulated trade volume to exceed the actual trade volume due to the way that demand is generated.

Description		Ave. Degree of sub-opt. of B+RR (%)	Std. deviation of deg. of sub-opt. of B+RR (%)	MIP Ave. Runtime (msec)	B+RR Ave. Runtime (msec)	Std. deviation of MIP Runtime	Std. deviation of B+RR Runtime
Price Range ($)	Current Balance ($)						
[10, 10]	0	0.00	0.00	400.00	1,600.00	0.52	0.84
[10, 10]	[-4000, 10,000]	0.01	0.01	45,250.00	1,750.00	120.00	0.50
[10, 100]	0	0.14	0.10	41,100.00	1,500.00	54.59	0.53
[10, 100]	[-4000, 10,000]	0.07	0.09	29,600.00	1,100.00	39.32	0.32
[10, 200]	0	0.21	0.26	12,700.00	900.00	4.24	0.32
[10, 200]	[-4000, 10,000]	0.21	1.15	7,400.00	400.00	7.34	0.70
[10, 500]	0	0.41	0.58	11,400.00	500.00	8.68	0.53
[10, 500]	[-4000, 10,000]	0.39	0.59	5,600.00	600.00	5.82	0.52
[10, 1000]	0	-0.21	1.73	12,500.00	600.00	24.74	0.52
[10, 1000]	[-4000, 10,000]	0.63	0.02	1,000.00	500.00	1.25	0.53

Table 1: Comparison of BALANCE+RANDOM_ROUND versus mixed integer programming.

Weeks	Actual Trade Volume	Optimized Simulation	Non-Optimized Simulation	Increase: Optimized vs Non-Optimized
1 - 66	$3,007,051	$2,573,403	$2,393,576	7%
48 - 66	$1,151,853	$878,800	$740,883	16%

Table 2: Comparison of optimized and non-optimized simulations with actual trade volumes.

The third set of experiments was carried out in order to examine the effect of balance optimization with more active trading and over a longer period of time than in the transaction history data and also to see how changing the financial operating ranges of the businesses effects the outcome of optimization. Using the 66 weeks of transaction data, we built a barter pool trade simulator[5] by learning Bayesian network models to predict company product demands. We tested a number of different models and found a naive Bayes model to give the best results, with an average ROC predictive value over all companies of 0.82. The simulator used the Bayesian networks to generate the demand for each company in each week. Again each company sold only one product category, with supply determined by the difference between the current balance and the upper bound. We ran a number of simulations to compare the absolute balance and the trade volume with and without balance optimization. In the simulations without balance optimization, maximal trade sets were determined as in the previous experiment. Each simulation used the same set of 130 companies and 26 product categories, with 3 - 7 suppliers per product category. Company balances all started at zero trade dollars. Simulations were run for 100 trade cycles, each cycle being one week. To observe the effectiveness of the optimization for various financial operating ranges, we varied the credit limits and upper limits. The credit limit ranged from that given in the data (CL) to four times that value (4CL). The upper bound ranged from twice the credit limit (2CL) to eight times the credit limit (8CL). Companies were not allowed to exceed their credit limit or upper limit. As the results of the simulations displayed in Table 3 show, using the optimizer results in a reduction in the average absolute balance over companies and in an increase in trade volume. The smallest increase in trade volume is for the [CL, 2CL] case, with the difference increasing as the credit limit is held constant and the upper bound increases to 8CL. The last two rows in the table show the results for the case of the credit limit and the upper bound equal to the values in the BizXchange barter pool, i.e. a credit limit of CL and high upper bounds. This results in the largest difference between the trade volumes of optimized and non-optimized simulations: 40%. These results are not surprising since tight credit limits and high upper bounds make it easy for the companies to hit their credit limits, making maintaining balance important. Although our simulation is far from incorporating all the complexities of trade dynamics, it does strongly suggest that maintaining balance of trade helps to increase trade volume over the long run.

[5]Details of the simulator are described in a related paper [11].

Description	Average ab_T per week ($)	Decrease in ab_T O vs NoO (%)	Average Trade Volume per week ($)	Increase in Trade Volume per week O vs NoO (%)
[CL, 2CL] NoO	1,455,448		79,752	
[CL, 2CL] O	1,030,860	29	93,870	18
[CL, 4CL] NoO	1,742,210		78,258	
[CL, 4CL] O	1,432,848	18	95,965	23
[CL, 8CL] NoO	1,903,018		68,731	
[CL, 8CL] O	1,481,754	22	95,696	39
[2CL, 4CL] NoO	3,193,685		106,184	
[2CL, 4CL] O	2,451,769	23	122,444	15
[4CL, 4CL] NoO	5,649,643		145,350	
[4CL, 4CL] O	4,634,927	18	176,389	21
BizXchange NoO	1,874,709		69,121	
BizXchange O	1,421,266	24	97,037	40

Table 3: Results of simulation runs with (O) and without (NoO) balance optimization.

7. RELATED WORK

Segev and Beam [16] point out that while there exists a large body of literature on auction theory, there is no equivalent theory for brokered marketplaces. They examine the effect of search costs on brokerage costs where negotiation is only on price. Their model assumes a single commodity where each seller has one unit and each buyer desires one unit and supply equals demand. They assume a single broker who receives bids from buyers and asks from sellers, each consisting of a single price point. The broker matches buyers and sellers by finding a seller whose asking price is below the bid of a buyer. Through simulation, they examine combinations of search and brokerage cost that make use of the broker a more attractive option than direct searching.

Tewari and Maes [17] describe the MARI project, which aims at developing an agent-mediated e-marketplace infrastructure for matching buyers and sellers. Each buyer and each seller is represented by an agent, which represents that buyer or seller's preferences via a multi-attribute utility function. The approach is illustrated with a market for translation services, where each buyer is looking for a certain number of words to be translated and each seller has a certain translation capacity to sell. They present an algorithm for matching buyers and sellers so that the aggregate surplus of all transaction parties is maximized. The amount a buyer is willing to pay for a given translation service is determined by his utility function. The algorithm works by representing the problem as a bipartite graph in which each buyer and each seller is represented by a node and there is a link between potential buyer-seller pairs, based on compatibility between hard buyer constraints and seller characteristics. Each link is assigned a reward equal to the bid-ask spread between the buyer and seller at the ends of the link. The objective is to satisfy all the buyers' needs by matching with sellers such that the sum of the rewards is maximized. They solve the weighted bipartite graph matching problem by transforming it to a linear program. Tewari and Maes are working with a single good characterized by multiple attributes and are performing matching to optimize aggregate surplus, while we are working with multiple goods and are matching based on more complex optimization criteria, resulting in more complex minimum cost flow problems. The work of Tewari and Maes focuses less on the combinatorial optimization aspects and more on providing a general framework for agent mediated buying and selling of non-tangible goods and services.

Our balance problem is closely related to the combinatorial auction problem (CAP), in which buyers may bid on combinations of goods, and the value of a good to a buyer may be a function of the other goods that he wins. In the typical formulation of the CAP, the auction house is faced with a set of price offers for various bundles of goods and its objective is to allocate the goods so as to maximize its own revenue. While early work on the CAP dealt only with single-unit CAPs, more recent work has dealt with multi-unit CAPs, in which there are multiple units of some goods available [13, 12].

Theorem 7.1. *The balance problem can be reduced to a sealed-bid multi-unit CAP in time linear in the size of the requirements matrix.*

Proof: For the formal framework, we generalize the single-unit model of Nisan [14] to cover multi-unit CAPs. We assume a sealed-bid auction in which each bidder has his own private valuation function v_i, which specifies his valuation for each possible set of items he may receive.

We have m bidders who wish to bid on a set of n goods. Let $q(j)$ be the number of available units of good j. An *allocation* is an $m \times n$ matrix G where G_{ij} is the number of units of good j allocated to buyer i such that $\sum_i G_{ij} \leq q(j)$. A *full allocation* [12] is an allocation in which $\sum_i G_{ij} = q(j)$. Each bidder i has a valuation function v_i associated with it, where $v_i(a_1, a_2, \ldots, a_n)$ denotes the value of the bundle of goods in which a_j units of good j are allocated to bidder i. The allocation problem is then to find a full allocation G so that $\sum_{i=1}^{m} v_i(G_{i1}, G_{i2}, \ldots, G_{in})$ is maximized. We shall consider the *constrained allocation problem*, which has the same objective, except that some of the entries in G have upper and lower bounds.

We now show how to represent the balance problem in the form of the above multi-unit CAP. Consider an $m \times n$ requirements matrix R. For each good p_j, set $q(j) = 2 \times \min(\sum_{i:c_i \in N_j} |R_{ij}|, \sum_{i:c_i \in S_j} R_{ij})$. If the supply of p_j exceeds its demand, then twice the total demand is auctioned (half of which will be assigned to all the buyers of p_j and the other half to all the sellers of p_j) and vice versa. Recall that α_j is the cost of good p_j in our barter universe $\mathcal{B}$. For each i and j, define the function $f_{ij}(a)$ as $-\alpha_j a$ if $c_i \in N_j$, and $\alpha_j a$ if $c_i \in S_j$, and 0 otherwise. For each i, we specify the valuation function as:

$$v_i(a_1, a_2, \ldots, a_n) = -\left| \sum_{j=1}^{n} f_{ij}(a_j) \right|.$$

(This is an asymmetric valuation function in the terminology of Nisan.) Our goal is to find the full allocation G that maximizes $\sum_{i=1}^{m} v_i(G_{i1}, G_{i2}, \ldots, G_{in})$ subject to the condition that for each j, (i) if supply of p_j exceeds its demand then $G_{ij} = |R_{ij}|$ for each $c_i \in N_j$ and $0 \leq G_{ij} \leq R_{ij}$ for each $c_i \notin N_j$; (ii) if the supply of p_j is no more than its demand then $0 \leq G_{ij} \leq |R_{ij}|$ for each $c_i \notin S_j$ and $G_{ij} = R_{ij}$ for each $c_i \in S_j$. It is straightforward to verify that the optimal full allocation G^* is also the most balanced maximal trade set. Finally, computing $q(j)$ and the valuation functions from the requirements matrix takes at most $O(|P||C|)$ time so our theorem follows. □

The problem of finding the optimal allocation in a combinatorial auction can be represented as an integer programming problem and is intractable in general. Researchers have identified numerous tractable special cases, typically expressed in terms of constraints on the bidding language [14]. When fractional solutions are admitted, the problem reduces to a linear programming problem and can be solved in polynomial time, as in our case as well. Archer et al [2] approximate the solution to the integer case by applying random rounding to the fractional solution produced by linear programming. Their random rounding algorithm works in two phases. The first phase has a small probability of over-selling some goods. When goods are over-sold, a second phase randomly deallocates goods from some bidders, resulting in a feasible allocation.

Our work is related to work on automated supply chain formation, the problem of assembling a network of agents that can transform basic goods into composite goods of value. The main problem in automated supply chain formation is to make sure that agents to not purchase more supplies than needed to satisfy the demand for their product and at the same time that they do not commit to prodviding more product than they can produce, given limited supplies. This should be done while maximizing efficiency, i.e. total value to all agents in the supply chain. Babaioff and Walsh [5] present one of the most general models for automated supply chain formation and show how to solve it as a combinatorial auction. There are interesting similarities and differences between their work and ours. They are interested in maintaining balance of flow of materials between consumption and production for each company. Analogously, we are interested in maintaining balance of cash flow for each company. But whereas they, like all other work in supply chain formation, assume an acyclic network of producers and consumers, the supply/demand relations in the barter pool are generally not acyclic. In their work, agents only receive positive value for obtaining the entire bundle of needed supplies. This is not the case in the barter pool since agents are free to obtain some or even all of their supplies outside the pool.

8. CONCLUSIONS

We have presented efficient algorithms for matching buyers and sellers in such as way that single-period trade volume is maximized and balance is maintained as much as possible. Our approach produces solutions that are within 0.7% of those produced by MIP but runs in a fraction of the time and can scale up to handle very large problems. Empirical evaluation shows the effectiveness of the optimization and supports the rule of thumb that maintaining balance of trade helps to maximize trade volume over the long run.

A number of questions remain open to further research. First, trade brokers not only try to maintain balance but also try to put trade dollars into the hands of businesses that tend to spend them. We are examining how to incorporate this by learning estimates of the rate at which different businesses tend to spend as a function of their current balance. Second, our algorithm for matching buyers and suppliers may still recommend that some companies go to many suppliers to satisfy a single product need. To be more realistic, we need to be able to incorporate a bound on the number of suppliers. Finally, we would like to make our simulator more realistic by adding parameters such as the probability that a business will follow a recommendation and the tendency of businesses to stick with suppliers.

Acknowledgements

We thank Bob Bagga and Chris Haddawy at BizXchange for their time and patience in helping us to understand the operations of trade exchanges and for generously providing the transaction data. We thank Sumanta Guha for help in formalizing an earlier qualitative version of this problem. The convex hull lemma is due to Hans Volkmer. Thanks to Yongyos Kaewpitakkun and Thai Bui for help in implementing the stochastic rounding algorithm.

9. REFERENCES

[1] R. Ahuja, T. Magnanti, and J. Orlin. *Network Flows.* Prentice Hall, 1993.

[2] A. Archer, C. Papadimitriou, K. Talwar, and E. Tardos. An approximate truthful mechanism for combinatorial auctions with single parameter agents. In *Proc. of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'03)*, pages 205–214, Baltimore, Maryland, January 2003.

[3] I. R. T. Association, 1999. http://www.irta.net/barterstatistics.html.

[4] I. R. T. Association, 2001. www.irta.net.

[5] M. Babaioff and W. Walsh. Incentive-compatible, budget-balanced, yet highly efficient auctions for supply chain formation. In *ACM Conference on Electronic Commerce (EC'03)*, pages 64–75, San Diego, California, 2003.

[6] B. Cresti. Welfare improving barter in imperfect competition. Discussion Paper N. 2002/71, Center for Operations Research and Econometrics (CORE), 2002.

[7] B. Cresti. U.S. domestic barter: An empirical investigation. Discussion Paper N. 2003-5, Institut de Recherches Economiques et Sociales (IRES), 2003. http://www.ires.ucl.ac.be/DP/IRES_DP/2003-5.pdf.

[8] A. Dailianas, J. Sairamesh, V. Gottemukkala, and A. Jhingran. Profit-driven matching in e-marketplaces: Trading composable commodities. In *Agent Mediated Electronic Commerce (IJCAI Workshop)*, pages 153–179, 1999.

[9] A. Goldberg and R. Tarjan. Finding minimum-cost circulations by canceling negative cycles. *Journal of the ACM*, 36(4):873–886, 1989.

[10] P. Haddawy, C. Cheng, and N. Rujikeadkumjorn. Balanced matching of buyers and sellers: The barter trade exchange model. Technical Report AIT-CSIM-TR-26-12-03, Asian Institute of Technology, December 2003. www.cs.ait.ac.th/~haddawy/Papers.htm.

[11] P. Haddawy, K. Dhananaiyapergse, Y. Kaewpitakkun, and T. Bui. Data-driven agent-based simulation of commercial barter trade. In *Proc. IEEE/WIC/ACM International Conference on Intelligent Agent Technologies (IAT'04)*, Beijing, September 2004.

[12] A. Kothari, D. Parkes, and S. Suri. Approximately-strategyproof and tractable multi-unit auctions. In *ACM Conference on Electronic Commerce (EC'03)*, pages 166–175, 2003.

[13] K. Leyton-Brown, Y. Shoham, and M. Tennenholtz. An algorithm for multi-unit combinatorial auctions. In *AAAI/IAAI*, pages 56–61, 2000.

[14] N. Nisan. Bidding and allocation in combinatorial auctions. In *ACM Conference on Electronic Commerce (EC'00)*, pages 1–12, 2000.

[15] M. Rossi, T. Saarinen, and V. Tuunainen. Introduction to special issue on new business models for electronic commerce. *ACM SIGMIS Database*, 34(2):7, 2003.

[16] A. Segev and C. Beam. Brokering strategies in electronic commerce marketplaces. In *ACM Conference on Electronic Commerce (EC'99)*, pages 167–176, 1999.

[17] G. Tewari and P. Maes. Design and implementation of an agent-based intermediary infrastructure for electronic marketplaces. In *ACM Conference on Electronic Commerce (EC'00)*, pages 86–94, 2000.

[18] H. Volkmer. Personal communication. Department of Mathematical Sciences, University of Wisconsin-Milwaukee, 2003.