

# ANNOUNCING THE OPENING OF THE ARIZONA TOKEN EXCHANGE (AZTE)

December 1, 1992

Earn cash profits by competing against computerized program traders

**THE CHALLENGE** Is "artificial intelligence" superior to human intelligence? In some domains such as chess, computer programs now outperform all but the very best human players. However in other domains such as speech, handwriting, and other kinds of pattern recognition, computers lag far behind human beings. On Wall Street computer "program traders" are becoming increasingly common, yet there is substantial controversy over their performance -- they have even been blamed as a factor in the October 1987 stock market crash. The purpose of this study, co-sponsored by the University of Arizona's Economic Science Laboratory and the Santa Fe Institute, is to compare the performance of human and program traders to see whether humans can learn to exploit the limitations and idiosyncracies of computers in repeated interactions.

**THE ARIZONA TOKEN EXCHANGE** To compare the performance of human and program traders we have created a computerized market, the Arizona Token Exchange (AZTE), in which a fictional commodity, "tokens", are traded. The market is a simplified version of commodity exchanges such as the Chicago Board of Trade where buyers and sellers are able to call out bids and asks to buy or sell units of the commodity. In each trading session on AZTE traders are assigned the role of buyer or seller and are given an allocation of tokens. A seller's objective is to sell their tokens for as much as possible above the token cost and a buyer's objective is to buy tokens as cheaply as possible below their redemption value. By ranking the token costs and redemption values, well-defined supply and demand curves can be constructed. The intersection of these curves defines the so-called competitive equilibrium (CE) price and quantity, at which neoclassical economic theory predicts all trading will occur. The complication is that in the AZTE, each trader's token costs and redemption values are private information and differ from trader to trader. Thus traders in the AZTE face a complex sequential decision problem: how much should they bid or ask for

their own tokens, how soon should they place a bid or ask, and under what circumstances should they accept an outstanding bid or ask of some other trader? An additional complication is that each trading session runs for a fixed amount of time. This creates a difficult trade-off, for if traders spend too much time looking for a good deal, they may find themselves locked out of the market without trading anything.

**HOW IS TRADER PERFORMANCE EVALUATED?** In the AZTE there is a well-defined performance measure: trading efficiency, EFF. This is the ratio of profits a trader actually earns divided by the profits it would have made if all trades took place at the competitive equilibrium level. Thus, if a trader's EFF is greater than 100% they are earning more than their "fair" share of the profits. The use of EFF is more desirable performance measure than simply using trading profits, since profits depend on the token allocations which are allocated at random from a known distribution. After each trading session, participants will earn cash profits equal to the following linear function of their efficiency:  $\$ \text{ payments} = a + b(\text{EFF} - 100)$  The term  $a$  represents a fixed fee paid for participating in the trading session and the term  $b(\text{EFF} - 100)$  represents a bonus (penalty) for trading above (or below) 100% efficiency. Thus, it is possible to lose money in any particular trading session. Dollar earnings are cumulated over successive trading sessions and subjects are eligible to "cash out" at any time after participating in a minimum number of trading sessions (provided cumulative earnings are positive).

**THE OPPONENTS: COMPUTER PROGRAM TRADERS** Unlike real commodities markets where most traders are humans, in the AZTE all of your opponents will be computer programs. The opponent programs will be selected from a field of over 30 different trading strategies including winners of the **Santa Fe Institute's Double Auction Tournament** held in March, 1990. The program traders range in sophistication from simple rules of thumb (such as Gode-Sunder "Zero-Intelligence" strategy) to sophisticated optimizing/learning algorithms (such as neural nets and genetic algorithms) developed from the recent literature on artificial intelligence. The identities of your opponents will (usually) be revealed to you at the start of each trading session. You will also be informed about other market characteristics such as the number of buyers and sellers, the number of tokens, and the joint distribution from which token values are drawn.

**SETTING UP AN ACCOUNT** To trade on the AZTE you will need a Unix or PC-compatible computer linked to the Internet computer network. We provide the trading interface software that allows you to log on and trade at any time you like and for as long as you like (subject to general restrictions). To qualify for an AZTE trading account you need to file an application providing information on your address, phone, and email

address, and a release form stating whether or not you want to remain anonymous in published analyses of the outcome of this experiment. Upon receipt of the application we will set up a trading account and access password. Your dollar earnings will cumulate in your account until you decide to cash out, at which time we will close your account and mail you a check for the total amount of your earnings. The software and ASCII traders' manual (including the application form) is available via anonymous ftp on "fido.econ.arizona.edu", in the azte sub-directory. The manual (azte.man) explains how the software works and what is required to use it. We suggest you ftp this first and read through it, then get the appropriate trading interface for your system. The DOS interface requires VGA graphics resolution and the use of Clarkson packet drivers for the network interface card. The Clarkson drivers are also available via ftp on "fido.econ.arizona.edu". If you don't have access to anonymous ftp, we will mail you a diskette containing the software and trader's manual.

To cover the costs of a diskette and surface mail, send \$5.00 to:

Shawn LaMaster                    Manager,  
Economic Science Systems Development  
Economic Science Laboratory  
McClelland Hall, Room #116  
University of Arizona  
Tucson, Arizona 85719                    (602) 621-6218  
Internet: lamaster@ziggy.econ.arizona.edu

We will assist in ftp and setting up the Clarkson packet drivers, just give us a call. The AZTE software was co-developed by: Sean Coates  
Economic Science Laboratory, University of Arizona  
Shawn LaMaster  
Economic Science Laboratory, University of Arizona  
Richard G. Palmer  
Duke University  
John Rust                    University of Wisconsin  
Vernon L. Smith  
Economic Science Laboratory, University of Arizona

ANDERSON

David Anderson  
University of Minnesota

This strategy attempts to buy at prices better than the average sale price in a period. If the average of the current bid and the current offer is above the average sale price my program bids aggressively (in my case bidding aggressively is bidding  $2*cbid/3+coffer/3$ .) I also bid aggressively if all other buyers have sold one more unit than me. If neither of these conditions hold I bid nonaggressively (in my case this means bidding  $10*cbid/11+coffer/11$ .) Since this is what I expect to happen in most cases my strategy amounts to attempting to colude with the other buyers to

bring prices down. I only bid aggressively when a favorable counteroffer has been made or if others are doing all the trading. Finally, if I'm running out of time or if I haven't bought in a long time I bid coffer. My strategy for selling is symmetric. CLASSIFICATION: simple, nonadaptive, nonstochastic, nonpredictive, nonoptimizing Program rank in 1990 SFI Double Auction Tournament: 8th out of 29

ANON1:

Strategy: delay as long as you dare before moving towards the opponent's price. This works GREAT if all your teammates do it also. If they don't, it's still nice for them, but not so nice for me. How noble. In order not to miss trades by running overtime, I define my waiting period as a fraction of the remaining ticks. CLASSIFICATION: simple nonadaptive nonstochastic nonpredictive nonoptimizing Rank in 1990 SFI DA tournament: 9th out of 29

ANON2

This program does the following: - keeps an average transaction price - goes quickly to somewhat below the average - goes slowly otherwise - speeds up toward the end of the period - preserves the average price across periods of the same round CLASSIFICATION: simple nonadaptive nonstochastic nonpredictive nonoptimizing Rank in 1990 SFI DA tournament: **4th out of 29**

BOLCER

Author(s) name: Gregory Alan Bolcer

Phone: (714) 856-4047

Institution: University of California at Irvine

Mailing address: Department of Information and Computer Science

University of California, Irvine

Irvine, CA 92717

e-mail address: gbolcer@ics.uci.edu

This program attempts to predict specific trading prices which can then be used to evaluate bid/offer values by utilizing concepts from an acceleration method in economics which assumes that as prices near equilibrium the velocity with which they approach this point decreases, and a method of encoding feature vectors into a scalar in such a way that is meaningful, (incompletely) similar to the method used for sensor-fusion in robotics. CLASSIFICATION: simple nonadaptive nonstochastic nonpredictive nonoptimizing

Rank in 1990 SFI DA tournament: 29th out of 29.

## BRETON

Marc Breton  
University of Minnesota

This program attempts to allow for flexible initial bidding /offering (at time 1, period 1) and a fairly aggressive strategy for achieving the low offer/high bid. In contrast, the strategy for closing in on a deal is less aggressive in that it requires reciprocation on the part of by the agents in the other half of the market. It responds to percentage changes as well as interval changes in the bid/offer from the other side of the market. The acceptance criteria for offers is simply enough to guarantee a reasonable return (i.e., ratio of bid to offer) and probably will be very susceptible to other competitors "stealing" the deal. C'est la vie! Rank in 1990 SFI DA tournament: 7th out of 29

## BROMILEY

Philip Bromiley, Mgmt. Dept., U of Minnesota  
271 - 19th Av S #835  
Minneapolis, MN 55455

ASP sets an aspiration level and adjusts it on the basis of last time's activity. It does a simple forecast of next time's offers and bids and then decides to enter the market if trading is expected to reach the aspiration level. By initially being quiet, ASP hopes to avoid counter-productive competition with traders on its own side. ASP takes care of a couple of odd contingencies: time running out and no messages from its own side, but leaves other, rather obvious contingencies uncovered (e.g. monopoly for or against, or only one unit to trade). ASP requires little computational ability and little memory. It is an outgrowth of work here to try to simulate some of the experimental market (human) results with minimal "expense" in terms of computational ability and memory. Rank in 1990 SFI DA tournament: 18th out of 29

## BURCHARD

Paul Burchard  
Institute for Advanced Study, ECP-8  
Princeton NJ 08540

This program uses an adaptive curve fitter based on "cell division" to predict the trading behavior of its fellow players. (For more information on the curve fitter, see "cells.h".) Four curve fitters are used in the current version of the program. They predict: [1] the highest bid to be made by

others on the next time step (assuming no trade takes place this time), [2] the same (but assuming a trade does take place this time), [3] the lowest offer to be made by others on the next time step (assuming no trade takes place this time), the same (but assuming a trade does take place this time). Given its predictions, it bids just high enough or offers just low enough that it should win on the next timestep. In addition to obviously acceptable deals, during a buy-sell step it will also accept deals (not objectionable in any obvious way) which come at a better than average price. At the very end of a period it will accept anything with positive profit. Rank in 1990 SFI DA tournament: 10th out of 29

## EXP

Roland Michelitsch, Shawn LaMaster, Mark VanBoening  
Economic Science Laboratory, University of Arizona, Tucson, Arizona

Exp computes an expected equilibrium price based not only on the information we gain by gametype, but also taking into consideration our own token values. Expected eq is updated if we are a seller if a trade occurs above the expected eq and if we are buyer if a trade occurs below the expected eq. At the end of every period (in the last quarter) Exp goes below this expected eq as far as a markup above/below the current token value. This markup depends on how much time is left. Rank in 1990 SFI DA tournament: 27th out of 29

## ESL Free

Designed by Dr. Corinne Bronfman, Finance Department, University of Arizona Programmed by Shawn LaMaster, Economic Science Lab, University of Arizona Economic Science Laboratory, University of Arizona, Tucson, Arizona Free uses the information in the price process to update the estimate of the equilibrium price (free rides on other peoples updating). As the number of trades increase, the weights on the current price information increase. Rank in 1990 SFI DA tournament: 23rd out of 29.

## GAMER

Shawn LaMaster, Mark Van Boening, Kevin McCabe Economic Science Laboratory, University of Arizona, Tucson, Arizona ESL Gamer simply bids/offers a 5% markup of its value to win the bid/offer step, and then accepts any bid/offer that provides non negative profits. Because of the discreet steps involved, the bid/offer and buy/sell steps, and the manner in which ties for both sides accepting the winning bid or offer are broken, we believe this simple strategy will work. At worst, ESL Gamer would receive 5% of its value, if a winner of the bid/offer step. At best, ESL Gamer would receive the difference between its value and the offer that won the bid/offer

step. This strategy would not work in an auction market where any trader can accept the standing bid/offer. Rank in 1990 SFI DA tournament: 24th out of 29

## SILVER BUFFALO

Greg Fullerton 4771 Tantra Boulder CO. 80303  
Mark Cronshaw CU, Dept of Econ. Box 256, Boulder CO. 80309-0256  
BJ Lee CU, Dept of Econ. Box 256, Boulder CO. 80309-0256  
Jamie Kruse CU, Dept of Econ. Box 256, Boulder CO. 80309-0256

This program bids based on risk calculated by time and tokens left. It starts by giving a random increment above the last token or cbid then when available it uses the past increment and estimated equilibrium calculated by a regression on past bids and offers. Rank in 1990 SFI DA tournament: 15th out of 29

## JACOBSON

Guy Jacobson, Carnegie Mellon University

A simple speed-up strategy that maintains an estimate of the equilibrium price for this round based on a weighted average of previous trade prices, and moves to that estimate at rate proportional to its confidence in the estimate's accuracy. (As more trades are made, this confidence increases.) Once there, it is pretty stubborn about getting its price, and only capitulates when time is running out relative to the rate at which the gap between cbid and coffer is closing in the current period, and when the profit is a relatively large fraction of the profit + gap. Rank in 1990 SFI DA tournament: 17th out of 29

## KAPLAN

Todd R. Kaplan  
University of Minnesota  
Mpls, Mn. 55406

This routine involves several strategies.

1. it stays out of the bidding war hoping that by doing so the market becomes more favorable to buyers
2. It waits until the bids and offers are within a certain range, then it jumps in and buys before someone else does. It also makes sure that before jumping in that the transaction is somewhat profitable
3. If its been a while since the last transaction, it decides to buy preventing the period from ending with incomplete transactions.

4. When it is stuck with a one on one, (No transactions can take place without it participating) it just uses (3) to wait until the seller saturates and offers its lowest possible price.
5. It tries to improve in future periods and not get 'taken' twice.

The main idea is to make the market favorable to buyers and get its fair share of the profits. This is a semi-collusive strategy.. If the other buyers/sellers keep there prices favorable, then they all can win big. In practice, it needs the majority of one side of the market to be non aggressive. Problems: the range it decides to jump in might be too big or too small so it might be to ambitious and buy too expensive, or wait too long and get shut out of the market. It also relies heavily on the other buyers being smart if they are non collusive. An aggressive side member can also hurt it. It also assumes that its opponents are computers and won't recognize its obvious patterns. This strategy takes no account of past history or performance. It also doesn't allow for tokens being outside the minprice-maxprice range. Rank in 1990 SFI DA tournament: **first place out of 29.**

KINDRED

Darrell Kindred  
Duke University

Basis of strategy: Similar to the simple one in the skeleton, but we adjust our bids/offers and propensity to sell based on previous sales, potential profits, and time left in the period.

Rank in 1990 SFI DA tournament: 22nd out of 29

LEE

Bernard P. Lee  
University of British Columbia

The strategy employed is a passive "observe your competition" strategy. An assumption is made that the competition has very similar token values to what was given to the program. As each competitor (buyer or seller) makes a bid or offer, their approximate 'margin' is calculated by dividing the amount of the bid or offer by the approximated value of their token. For instance, of buyer #2 has made an offer of 500, and had already bought one in the same period, this program approximates that buyer #2's cost will be the same as the value of the second token given. All the calculated margins of the most recent offers for each competitor with each token is saved, and the program's next bid and offer for the nth token will be based on the average margin on the nth token of all other competitors, instead of the current bid or offer. This strategy isolates the fact that different competitors



with different token values will be bidding or offering at similar amounts, each with different levels of profit. The strategy also allows for setting of a fixed goal that whatever the bid or offer will be, it must be equal to or greater than the average profit gained by all other competitors. To prevent the strategy from being stubborn, data on how much the program is 'ahead of' or 'behind' the competition is also collected. If the program is 'ahead of' the competition, the 'requirement' on the profit will be greater. If the program is 'behind', the 'requirement' on the profit will drop in order to meet the furious competition. Buying and selling are also very simple strategies. A ratio between the profits if the current bid becomes the price and if the current offer becomes the price is made. Accept the bid/offer if the ratio falls to a certain limit. If the ratio is too high, do not accept, and hope that the other side accepts. If the held current bid or offer is bettered, let the bid/offer strategy beat it again. If the other side refuse to accept for a fixed number of times, assume the other side has reached its lower limit on profit and accept the offer/bid. If it is about the last time in the period, accept it because minimal profit is better than none. Both the ratio and the maximum number of times to reject is adjusted by the position relative to the competition, i.e. how much ahead of or behind the competition. No assumption on consistent behaviour on other players is made, and parameters are cleared and recalculated each period. Rank in 1990 SFI DA tournament: 13th out of 29

LEINWEBER

David Leinweber  
MJT Advisers  
800 West 6th Street  
LA, CA 90066

This is a simple strategy based on the fact that it is better to take a smaller profit at the end of a period than to be stuck with token that "expire" worthless. The basic strategy is the same as the skeleton provided by SFI, modified as follows: The maximum number of trades possible is calculated at the start of each period. When more than half of this number have actually occurred, the program becomes less "greedy" in its buy/sell decisions. Rank in 1990 SFI DA tournament: 14th out of 29

LIN

Prof. Kuan-Pin Lin, Department of Economics  
Portland State University, Portland, OR 97207

My strategy in playing DA game is very simple. First the "bid" is a weighted average of the standing bid (if any) and a reservation (target) price drawn from a normal distribution with mean and standard error of historical prices in the market. The weight used in computing this bid depends on number of sellers and buyers in the market, the number of total and current time units

in each period, and the number of total and current trades. Next the "buy" decision is based on the profitability rule that current offer must be lower than the mean market price plus one standard error. The seller's "offer" and "sell" strategies are constructed similarly. Rank in 1990 SFI DA tournament: 26th out of 29

## MAX

Vernon Smith, Shawn LaMaster, Mark Van Boening, Roland Michelitsch, Robert Dorsey, Steve Rassenti, Kevin McCabe, Corinne Bronfman, Eric Lu  
Economic Science Laboratory, University of Arizona, Tucson, Arizona  
Max is an expected utility maximizing buyer/seller that uses the cumulative distribution function of offers/bids to calculate the probability that a seller/buyer will accept our bid/offer. Until the number of observations of offers+accepted bids/ bids+accepted offers are equal to  $(2 \times \text{ntimes} \times \text{sellers}) / (2 \times \text{ntimes} \times \text{buyers})$ , or 20, whichever is smaller, Max uses an exponential decay based on the expected equilibrium price computed from the gametype to determine a bid/offer. Risk in this model is held at 1 for a risk neutral player. Max uses a linear decay rule for minimum acceptable profit when deciding to accept a offer/bid once he wins the bid/offer step. This decay starts with the maximum amount of profit and then decays to zero. Rank in 1990 SFI DA tournament: 19th out of 29

## MAX-R

Vernon Smith, Shawn LaMaster, Mark Van Boening, Roland Michelitsch  
Robert Dorsey, Steve Rassenti, Kevin McCabe, Corinne Bronfman, Eric Lu  
Economic Science Laboratory, University of Arizona, Tucson, Arizona  
Max is an expected utility maximizing buyer/seller that uses the cumulative distribution function of offers/bids to calculate the probability that a seller/buyer will accept our bid/offer. Until the number of observations of offers+accepted bids/ bids+accepted offers are equal to  $(2 \times \text{ntimes} \times \text{sellers}) / (2 \times \text{ntimes} \times \text{buyers})$ , or 20, whichever is smaller, Max uses an exponential decay based on the expected equilibrium price computed from the gametype to determine a bid/offer. Risk in this model is held at 1.5 for a risk loving player. Max uses a linear decay rule for minimum acceptable profit when deciding to accept a offer/bid once he wins the bid/offer step. This decay starts with the maximum amount of profit and then decays to zero. Rank in 1990 SFI DA tournament: 20th out of 29

## GA-NN-1

Richard Dallaway  
richardd@cogs.susx.ac.uk

Inman Harvey  
inmanh@cogs.susx.ac.uk

School of Cognitive & Computing Sciences

University of Sussex  
Falmer BRIGHTON UK

Genetic algorithm has been used to evolve recurrent neural network. Program is described in more detail in Rust, Miller, Palmer "Characterizing Effective Trading Strategies: Insights from a Computerized Double Auction Tournament" Journal of Economic Dynamics and Control (forthcoming) also available by anonymous ftp from fido.econ.arizona.edu in azte/da2.ps. This program reads the result of the best player (garnet.dna) we had at the time we stopped evolution. Garnet.dna is a specification of a fixed (i.e., no learning) recurrent neural network. Disqualified for excessive losses in 1990 SFI DA tournament Program modified to yield average performance in subsequent "scientific tournaments" held at SFI.

PERRY

Stan Perry  
Portland State University

This trader uses a statistical bidding strategy based on the average price and the price variance. In addition, a measure of expected efficiency is included and is used to adjust a tuneable parameter at the end of each trading period. The Statistical Player uses a fairly conservative non-statistical strategy until the first 3 trades are completed in the first period. In what follows the market strategy is described from the buyer's viewpoint, the seller's viewpoint is generally just the inverse. The Statistical Player computes the average price and standard deviation from the beginning of the round, and for the most recent period within the round. The player weights the statistics from the previous period and combines this with the round statistics. These simple statistics are combined with information on the number of players, the seller/buyer ratio, the apparent efficiency and the time remaining within the round to develop a bid. This player opens bidding at about 2 standard deviations below the average price at the beginning of a period. (The precise amount the bid is below average price depends on the tuneable parameter which includes some random noise.) As the period progresses, the player moves the bid closer to the average price, and at the end of the period it may bid up to 0.2 standard deviations above the average price. The buying strategy is very simple, the Statistical Player sets an acceptance region at the beginning of the period at approximately 1 standard deviation below the average price. As the period evolves, the player moves the acceptance region closer to the average price. By the end of the period, the player will accept an offer that is 0.2 standard deviations above the average price. At the end of each period an evaluation function is called to assess the efficiency of the player. The possible profit is computed by comparing the number of trades that were feasible at the average price with the player's token values. The apparent efficiency is

calculated from actual profit and the possible profit at the average price. If the efficiency is below an acceptable level a tuneable parameter is adjusted in proportion to the size of the error. Rank in 1990 SFI DA tournament: **6h out of 29**

PERRY

Stan Perry  
Portland State University

This trader uses a statistical bidding strategy based on the average price and the price variance. In addition, a measure of expected efficiency is included and is used to adjust a tuneable parameter at the end of each trading period. The Statistical Player uses a fairly conservative non-statistical strategy until the first 3 trades are completed in the first period. In what follows the market strategy is described from the buyer's viewpoint, the seller's viewpoint is generally just the inverse. The Statistical Player computes the average price and standard deviation from the beginning of the round, and for the most recent period within the round. The player weights the statistics from the previous period and combines this with the round statistics. These simple statistics are combined with information on the number of players, the seller/buyer ratio, the apparent efficiency and the time remaining within the round to develop a bid. This player opens bidding at about 2 standard deviations below the average price at the beginning of a period. (The precise amount the bid is below average price depends on the tuneable parameter which includes some random noise.) As the period progresses, the player moves the bid closer to the average price, and at the end of the period it may bid up to 0.2 standard deviations above the average price. The buying strategy is very simple, the Statistical Player sets an acceptance region at the beginning of the period at approximately 1 standard deviation below the average price. As the period evolves, the player moves the acceptance region closer to the average price. By the end of the period, the player will accept an offer that is 0.2 standard deviations above the average price. At the end of each period an evaluation function is called to assess the efficiency of the player. The possible profit is computed by comparing the number of trades that were feasible at the average price with the player's token values. The apparent efficiency is calculated from actual profit and the possible profit at the average price. If the efficiency is below an acceptable level a tuneable parameter is adjusted in proportion to the size of the error. Rank in 1990 SFI DA tournament: 6h out of 29

RINGUETTE

Marc Ringuette, Carnegie Mellon University, Comp Sci Dept.

This is a simple-minded strategy which does surprisingly well in local run-offs. I'll describe it from the buyer's point of view. It has two phases. For a long time, it just watches, making no significant bids; but if a buyer and seller converge within a trigger margin, it jump-bids and makes the transaction. In other words, it waits for fellow buyers to do the negotiating, then "steals" the transaction. The surprising thing is that it doesn't check what the selling price is as long as it's not a loss -- it trusts its fellow buyers to do the negotiating. For robustness, it has a second phase for when no one else is bidding. In that case, I chose to have the program bid as slowly as possible. It can get away with slow bidding because (a) when phase 2 is invoked, there is usually no competition from other buyers, and (b) because it speeds up as the deadline approaches, so as not to lose transactions. Given this, slower is better, to pull the selling price down. General comments: I conjecture that, since it is fairly difficult to write a computer program that recognizes the behavior of other programs, most programs will be almost oblivious to the behavior of the other players. Given this, there are two possible tactics: (1) make a non-oblivious program that beats up on oblivious ones (2) choose the most robust oblivious strategy you can manage. I chose the second option, mainly because it's easier. There's a tricky tradeoff: bid quickly so as to put yourself in the drivers' seat and make all the trades possible, or bid slowly to pull equilibrium prices downwards. I try to strike a middle ground, by not bidding at all, but by snatching transactions so as not to be left out. Rank in 1990 SFI DA tournament: **2nd out of 29**

ANDERSON

David Anderson  
University of Minnesota

This strategy attempts to buy at prices better than the average sale price in a period. If the average of the current bid and the current offer is above the average sale price my program bids aggressively (in my case bidding aggressively is bidding  $2*cbid/3+coffer/3$ .) I also bid aggressively if all other buyers have sold one more unit than me. If neither of these conditions hold I bid nonaggressively (in my case this means bidding  $10*cbid/11+coffer/11$ .) Since this is what I expect to happen in most cases my strategy amounts to attempting to collude with the other buyers to bring prices down. I only bid aggressively when a favorable counteroffer has been made or if others are doing all the trading. Finally, if I'm running out of time or if I haven't bought in a long time I bid coffer. My strategy for selling is symmetric.

CLASSIFICATION: simple, nonadaptive, nonstochastic, nonpredictive, nonoptimizing Program rank in 1990 SFI Double Auction Tournament: 8th out of 29

ANON1:

Strategy: delay as long as you dare before moving towards the opponent's price. This works GREAT if all your teammates do it also. If they don't, it's still nice for them, but not so nice for me. How noble.

In order not to miss trades by running overtime, I define my waiting period as a fraction of the remaining ticks.

CLASSIFICATION: simple nonadaptive nonstochastic nonpredictive nonoptimizing

Rank in 1990 SFI DA tournament: 9th out of 29

ANON2

This program does the following:

- keeps an average transaction price
- goes quickly to somewhat below the average
- goes slowly otherwise
- speeds up toward the end of the period
- preserves the average price across periods of the same round

CLASSIFICATION: simple nonadaptive nonstochastic nonpredictive nonoptimizing

Rank in 1990 SFI DA tournament: 4th out of 29

BOLCER

Author(s) name: Gregory Alan Bolcer

Phone: (714) 856-4047

Institution: University of California at Irvine

Mailing address: Department of Information and Computer Science

University of California, Irvine

Irvine, CA 92717

e-mail address: gbolcer@ics.uci.edu

This program attempts to predict specific trading prices which can then be used to evaluate bid/offer values by utilizing concepts from an acceleration method in economics which assumes that as prices near equilibrium the velocity with which they approach this point decreases, and a method of encoding feature vectors into a scalar in such a way that is meaningful, (incompletely) similar to the method used for sensor-fusion in robotics.

CLASSIFICATION: simple nonadaptive nonstochastic nonpredictive nonoptimizing

Rank in 1990 SFI DA tournament: 29th out of 29.  
BRETON

Marc Breton  
University of Minnesota

This program attempts to allow for flexible initial bidding /offering (at time 1, period 1) and a fairly aggressive strategy for achieving the low offer/high bid. In contrast, the strategy for closing in on a deal is less aggressive in that it requires reciprocation on the part of by the agents in the other half of the market. It responds to percentage changes as well as interval changes in the bid/offer from the other side of the market. The acceptance criteria for offers is simply enough to guarantee a reasonable return (i.e., ratio of bid to offer) and probably will be very susceptible to other competitors "stealing" the deal. C'est la vie!

Rank in 1990 SFI DA tournament: 7th out of 29  
BROMILEY

Philip Bromiley, Mgmt. Dept., U of Minnesota  
271 - 19th Av S #835  
Minneapolis, MN 55455

ASP sets an aspiration level and adjusts it on the basis of last time's activity. It does a simple forecast of next time's offers and bids and then decides to enter the market if trading is expected to reach the aspiration level.

By initially being quiet, ASP hopes to avoid counter-productive competition with traders on its own side.

ASP takes care of a couple of odd contingencies: time running out and no messages from its own side, but leaves other, rather obvious contingencies uncovered (e.g. monopoly for or against, or only one unit to trade).

ASP requires little computational ability and little memory. It is an outgrowth of work here to try to simulate some of the experimental market (human) results with minimal "expense" in terms of computational ability and memory.

Rank in 1990 SFI DA tournament: 18th out of 29  
BURCHARD

Paul Burchard  
Institute for Advanced Study, ECP-8  
Princeton NJ 08540

This program uses an adaptive curve fitter based on "cell division" to predict the trading behavior of its fellow players. (For more information on the curve fitter, see "cells.h".) Four curve fitters are used in the current version of the program. They predict: [1] the highest bid to be made by others on the next time step (assuming no trade takes place this time), [2] the same (but assuming a trade does take place this time), [3] the lowest offer to be made by others on the next time step (assuming no trade takes place this time), the same (but assuming a trade does take place this time). Given its predictions, it bids just high enough or offers just low enough that it should win on the next timestep.

In addition to obviously acceptable deals, during a buy-sell step it will also accept deals (not objectionable in any obvious way) which come at a better than average price. At the very end of a period it will accept anything with positive profit.

Rank in 1990 SFI DA tournament: 10th out of 29

EXP

Roland Michelitsch, Shawn LaMaster, Mark VanBoening  
Economic Science Laboratory, University of Arizona, Tucson, Arizona

Exp computes an expected equilibrium price based not only on the information we gain by gametype, but also taking into consideration our own token values. Expected eq is updated if we are a seller if a trade occurs above the expected eq and if we are buyer if a trade occurs below the expected eq. At the end of every period (in the last quarter) Exp goes below this expected eq as far as a markup above/below the current token value. This markup depends on how much time is left.

Rank in 1990 SFI DA tournament: 27th out of 29  
ESL Free

Designed by Dr. Corinne Bronfman, Finance Department, University of Arizona  
Programmed by Shawn LaMaster, Economic Science Lab, University of Arizona  
Economic Science Laboratory, University of Arizona, Tucson, Arizona



Free uses the information in the price process to update the estimate of the equilibrium price (free rides on other peoples updating). As the number of trades increase, the weights on the current price information increase.

Rank in 1990 SFI DA tournament: 23rd out of 29.

GAMER

Shawn LaMaster, Mark Van Boening, Kevin McCabe  
Economic Science Laboratory, University of Arizona, Tucson, Arizona

ESL Gamer simply bids/offers a 5% markup of its value to win the bid/offer step, and then accepts any bid/offer that provides non negative profits. Because of the discreet steps involved, the bid/offer and buy/sell steps, and the manner in which ties for both sides accepting the winning bid or offer are broken, we believe this simple strategy will work. At worst, ESL Gamer would receive 5% of its value, if a winner of the bid/offer step. At best, ESL Gamer would receive the difference between its value and the offer that won the bid/offer step. This strategy would not work in an auction market where any trader can accept the standing bid/offer.

Rank in 1990 SFI DA tournament: 24th out of 29  
SILVER BUFFALO

Greg Fullerton 4771 Tantra Boulder CO. 80303  
Mark Cronshaw CU, Dept of Econ. Box 256, Boulder CO. 80309-0256  
BJ Lee CU, Dept of Econ. Box 256, Boulder CO. 80309-0256  
Jamie Kruse CU, Dept of Econ. Box 256, Boulder CO. 80309-0256

This program bids based on risk calculated by time and tokens left. It starts by giving a random increment above the last token or cbid then when available it uses the past increment and estimated equilibrium calculated by a regression on past bids and offers.

Rank in 1990 SFI DA tournament: 15th out of 29  
JACOBSON

Guy Jacobson, Carnegie Mellon University

A simple speed-up strategy that maintains an estimate of the equilibrium price for this round based on a weighted average of previous trade prices, and moves to that estimate at rate propotional to its confidence in the estimate's accuracy. (As more trades are made, this confidence

increases.) Once there, it is pretty stubborn about getting its price, and only capitulates when time is running out relative to the rate at which the gap between bid and offer is closing in the current period, and when the profit is a relatively large fraction of the profit + gap.

Rank in 1990 SFI DA tournament: 17th out of 29  
KAPLAN

Todd R. Kaplan  
University of Minnesota  
Mpls, Mn. 55406

This routine involves several strategies.

1. it stays out of the bidding war hoping that by doing so the market becomes more favorable to buyers
2. It waits until the bids and offers are within a certain range, then it jumps in and buys before someone else does. It also makes sure that before jumping in that the transaction is somewhat profitable
3. If its been a while since the last transaction, it decides to buy preventing the period from ending with incomplete transactions.
4. When it is stuck with a one on one, (No transactions can take place without it participating) it just uses (3) to wait until the seller saturates and offers its lowest possible price.
5. It tries to improve in future periods and not get 'taken' twice.

The main idea is to make the market favorable to buyers and get its fair share of the profits. This is a semi-collusive strategy.. If the other buyers/sellers keep their prices favorable, then they all can win big. In practice, it needs the majority of one side of the market to be non aggressive. Problems: the range it decides to jump in might be too big or too small so it might be too ambitious and buy too expensive, or wait too long and get shut out of the market.

It also relies heavily on the other buyers being smart if they are non collusive. An aggressive side member can also hurt it. It also assumes that its opponents are computers and won't recognize its obvious patterns. This strategy takes no account of past history or performance. It also doesn't allow for tokens being outside the minprice-maxprice range.

Rank in 1990 SFI DA tournament: first place out of 29.  
KINDRED

Darrell Kindred  
Duke University

Basis of strategy: Similar to the simple one in the skeleton, but we adjust our bids/offers and propensity to sell based on previous sales, potential profits, and time left in the period.

Rank in 1990 SFI DA tournament: 22nd out of 29  
LEE

Bernard P. Lee  
University of British Columbia

The strategy employed is a passive "observe your competition" strategy. An assumption is made that the competition has very similar token values to what was given to the program. As each competitor (buyer or seller) makes a bid or offer, their approximate 'margin' is calculated by dividing the amount of the bid or offer by the approximated value of their token.

For instance, if buyer #2 has made an offer of 500, and had already bought one in the same period, this program approximates that buyer #2's cost will be the same as the value of the second token given. All the calculated margins of the most recent offers for each competitor with each token is saved, and the program's next bid and offer for the nth token will be based on the average margin on the nth token of all other competitors, instead of the current bid or offer. This strategy isolates the fact that different competitors with different token values will be bidding or offering at similar amounts, each with different levels of profit. The strategy also allows for setting of a fixed goal that whatever the bid or offer will be, it must be equal to or greater than the average profit gained by all other competitors.

To prevent the strategy from being stubborn, data on how much the program is 'ahead of' or 'behind' the competition is also collected. If the program is 'ahead of' the competition, the 'requirement' on the profit will be greater. If the program is 'behind', the 'requirement' on the profit will drop in order to meet the furious competition.

Buying and selling are also very simple strategies. A ratio between the profits if the current bid becomes the price and if the current offer becomes the price is made. Accept the bid/offer if the ratio falls to a certain limit. If the ratio is too high, do not accept, and hope that the other side accepts.

If the held current bid or offer is bettered, let the bid/offer strategy beat it again. If the other side refuse to accept for a fixed number of times, assume the other side has reached its lower limit on profit and accept the offer/bid. If it is about the last time in the

period, accept it because minimal profit is better than none. Both the ratio and the maximum number of times to reject is adjusted by the position relative to the competition, i.e. how much ahead of or behind the competition.

No assumption on consistent behaviour on other players is made, and parameters are cleared and recalculated each period.

Rank in 1990 SFI DA tournament: 13th out of 29  
LEINWEBER

David Leinweber  
MJT Advisers  
800 West 6th Street  
LA, CA 90066

This is a simple strategy based on the fact that it is better to take a smaller profit at the end of a period than to be stuck with token that "expire" worthless. The basic strategy is the same as the skeleton provided by SFI, modified as follows:

The maximum number of trades possible is calculated at the start of each period. When more than half of this number have actually occurred, the program becomes less "greedy" in its buy/sell decisions.

Rank in 1990 SFI DA tournament: 14th out of 29  
LIN

Prof. Kuan-Pin Lin, Department of Economics  
Portland State University, Portland, OR 97207

My strategy in playing DA game is very simple. First the "bid" is a weighted average of the standing bid (if any) and a reservation (target) price drawn from a normal distribution with mean and standard error of historical prices in the market. The weight used in computing this bid depends on number of sellers and buyers in the market, the number of total and current time units in each period, and the number of total and current trades. Next the "buy" decision is based on the profitability rule that current offer must be lower than the mean market price plus one standard error. The seller's "offer" and "sell" strategies are constructed similarly.

Rank in 1990 SFI DA tournament: 26th out of 29  
MAX

Vernon Smith, Shawn LaMaster, Mark Van Boening, Roland Michelitsch, Robert Dorsey, Steve Rassenti, Kevin McCabe, Corinne Bronfman, Eric Lu  
Economic Science Laboratory, University of Arizona, Tucson, Arizona

Max is an expected utility maximizing buyer/seller that uses the cumulative distribution function of offers/bids to calculate the probability that a seller/buyer will accept our bid/offer.

Until the number of observations of offers+accepted bids/bids+accepted offers are equal to  $(2 \cdot n_{\text{times}} \cdot \text{sellers}) / (2 \cdot n_{\text{times}} \cdot \text{buyers})$ , or 20, whichever is smaller, Max uses an exponential decay based on the expected equilibrium price computed from the gametype to determine a bid/offer.

Risk in this model is held at 1 for a risk neutral player. Max uses a linear decay rule for minimum acceptable profit when deciding to accept a offer/bid once he wins the bid/offer step. This decay starts with the maximum amount of profit and then decays to zero.

Rank in 1990 SFI DA tournament: 19th out of 29

MAX-R

Vernon Smith, Shawn LaMaster, Mark Van Boening, Roland Michelitsch, Robert Dorsey, Steve Rassenti, Kevin McCabe, Corinne Bronfman, Eric Lu  
Economic Science Laboratory, University of Arizona, Tucson, Arizona

Max is an expected utility maximizing buyer/seller that uses the cumulative distribution function of offers/bids to calculate the probability that a seller/buyer will accept our bid/offer.

Until the number of observations of offers+accepted bids/bids+accepted offers are equal to  $(2 \cdot n_{\text{times}} \cdot \text{sellers}) / (2 \cdot n_{\text{times}} \cdot \text{buyers})$ , or 20, whichever is smaller, Max uses an exponential decay based on the expected equilibrium price computed from the gametype to determine a bid/offer.

Risk in this model is held at 1.5 for a risk loving player. Max uses a linear decay rule for minimum acceptable profit when deciding to accept a offer/bid once he wins the bid/offer step. This decay starts with the maximum amount of profit and then decays to zero.

Rank in 1990 SFI DA tournament: 20th out of 29

GA-NN-1

Richard Dallaway

Inman Harvey

richardd@cogs.susx.ac.uk

inmanh@cogs.susx.ac.uk

School of Cognitive & Computing Sciences  
University of Sussex  
Falmer BRIGHTON UK

Genetic algorithm has been used to evolve recurrent neural network. Program is described in more detail in Rust, Miller, Palmer "Characterizing Effective Trading Strategies: Insights from a Computerized Double Auction Tournament" Journal of Economic Dynamics and Control (forthcoming) also available by anonymous ftp from fido.econ.arizona.edu in azte/da2.ps. This program reads the result of the best player (garnet.dna) we had at the time we stopped evolution. Garnet.dna is a specification of a fixed (i.e., no learning) recurrent neural network.

Disqualified for excessive losses in 1990 SFI DA tournament  
Program modified to yield average performance in subsequent "scientific tournaments" held at SFI.  
PERRY

Stan Perry  
Portland State University

This trader uses a statistical bidding strategy based on the average price and the price variance. In addition, a measure of expected efficiency is included and is used to adjust a tuneable parameter at the end of each trading period. The Statistical Player uses a fairly conservative non-statistical strategy until the first 3 trades are completed in the first period.

In what follows the market strategy is described from the buyer's viewpoint, the seller's viewpoint is generally just the inverse. The Statistical Player computes the average price and standard deviation from the beginning of the round, and for the most recent period within the round. The player weights the statistics from the previous period and combines this with the round statistics. These simple statistics are combined with information on the number of players, the seller/buyer ratio, the apparent efficiency and the time remaining within the round to develop a bid.

This player opens bidding at about 2 standard deviations below the average price at the beginning of a period. (The precise amount the bid is below average price depends on the tuneable parameter which includes some random noise.) As the period progresses, the player moves the bid closer to the average price, and at the end of the period

it may bid up to 0.2 standard deviations above the average price.

The buying strategy is very simple, the Statistical Player sets an acceptance region at the beginning of the period at approximately 1 standard deviation below the average price. As the period evolves, the player moves the acceptance region closer to the average price. By the end of the period, the player will accept an offer that is 0.2 standard deviations above the average price.

At the end of each period an evaluation function is called to assess the efficiency of the player. The possible profit is computed by comparing the number of trades that were feasible at the average price with the players token values. The apparent efficiency is calculated from actual profit and the possible profit at the average price. If the efficiency is below an acceptable level a tuneable parameter is adjusted in proportion to the size of the error.

Rank in 1990 SFI DA tournament: 6h out of 29  
PRICETAKER

John Rust  
University of Wisconsin  
Madison, WI 53706

Pricetaker emulates "naive price-taking" behavior, treating market prices as good estimates of CE prices. If PTAKER is a buyer, he will accept any outstanding offer that yields a profit if time is running out: i.e. if there are fewer than STPLFT steps remaining, where STPLFT is a random fraction of the total number of available time steps (see precise definition below). If there are more than STPLFT steps remaining, Pricetaker won't buy until it sees a "good deal". A good deal is defined as an offer which is no more than a certain percentage MRKUP over a running average of the transaction price in the current and previous periods. However even if it detects a good deal, Pricetaker will still wait to buy if offer prices are coming down and there are more the STPLFT steps remaining in the period.

Pricetaker makes bids according to a simple rule: if there is a current bid outstanding and this bid is not higher than a certain percentage MRKUP over a running average of the transaction prices in the current and previous periods, then Pricetaker overbids the current bid by a random percentage PREM of the profits earned on the purchase of its token at the current bid price, or passes if profits at the current bid are negative. If the current bid is "too high", i.e. more than MRKUP percent over a running average of transaction prices in the current and previous periods,

then Pricetaker waits for bid prices to come down before trying to bid. If there is no current bid then Pricetaker overbids a random percentage PREM of the running average transaction price, provided it is positive. However if the running average transaction price is zero (as it is in the start of period 1 in each round), then Pricetaker starts by making an "outrageous bid", i.e. a random fraction of his lowest redemption value. Such an outrageous bid guarantees that Pricetaker will earn a handsome profit if accepted by a "sucker" seller at the beginning of the game, reducing the risk of being committed to buy at too high a price while Pricetaker is getting a sense of what kinds of bids and offers the other traders will be making.

The strategy Pricetaker follows as a seller is simply a mirror image of the strategy it follows as a buyer.

Program did not participate in 1990 SFI DA tournament, but ranks as an average player in subsequent "scientific tournaments".  
RINGUETTE

Marc Ringuette, Carnegie Mellon University, Comp Sci Dept.

This is a simple-minded strategy which does surprisingly well in local run-offs. I'll describe it from the buyer's point of view. It has two phases. For a long time, it just watches, making no significant bids; but if a buyer and seller converge within a trigger margin, it jump-bids and makes the transaction. In other words, it waits for fellow buyers to do the negotiating, then "steals" the transaction. The surprising thing is that it doesn't check what the selling price is as long as it's not a loss -- it trusts its fellow buyers to do the negotiating.

For robustness, it has a second phase for when no one else is bidding. In that case, I chose to have the program bid as slowly as possible. It can get away with slow bidding because (a) when phase 2 is invoked, there is usually no competition from other buyers, and (b) because it speeds up as the deadline approaches, so as not to lose transactions. Given this, slower is better, to pull the selling price down.

General comments: I conjecture that, since it is fairly difficult to write a computer program that recognizes the behavior of other programs, most programs will be almost oblivious to the behavior of the other players. Given this, there are two possible tactics:

- (1) make a non-oblivious program that beats up on oblivious ones



(2) choose the most robust oblivious strategy you can manage.  
I chose the second option, mainly because it's easier.

There's a tricky tradeoff: bid quickly so as to put yourself in the drivers' seat and make all the trades possible, or bid slowly to pull equilibrium prices downwards. I try to strike a middle ground, by not bidding at all, but by snatching transactions so as not to be left out.

Rank in 1990 SFI DA tournament: 2nd out of 29  
SILVER BUFFALO

Greg Fullerton 4771 Tantra Boulder CO. 80303  
Mark Cronshaw CU, Dept of Econ. Box 256, Boulder CO. 80309-0256  
BJ Lee CU, Dept of Econ. Box 256, Boulder CO. 80309-0256  
Jamie Kruse CU, Dept of Econ. Box 256, Boulder CO. 80309-0256

This program bids based on risk calculated by time and tokens left. It starts by giving a random increment above the last token or cbid then when available it uses the past increment and estimated equilibrium calculated by a regression on past bids and offers.

Rank in 1990 SFI DA tournament: 15th out of 29  
SKELETON

Richard Palmer, Duke University  
John Miller, Carnegie Mellon University

This strategy is the simple example strategy distributed to all potential entrants to the 1990 SFI double auction tournament. A block diagram of this strategy appears in the paper "Characterizing Effective Trading Strategies: Insights from a Computerized Double Auction Tournament" forthcoming in the Journal of Economic Dynamics and Control 1993 and also available by anonymous ftp from /pub/azte/papers at fido.econ.arizona.edu.

The bidding strategy of the Skeleton program (the strategy for offering is symmetric) depends on whether or not there is a current bid on the table. If  $cbid > 0$ , skeleton bids a randomly weighted average of  $cbid+1$  and the variable

MOST defined as the minimum of the current ask and the program's redemption value minus one (provided the current ask  $> 0$ ) or the redemption value minus one otherwise. If there is no current bid ( $cbid=0$ ) then skeleton bids a value of MOST less  $U(\text{first token} - \text{last token})$  where  $U$  is a uniform  $(0,1)$  random variable and first token is the redemption value of Skeleton's most valuable token and last token is the redemption value of skeleton's least valuable token. When  $cbid=0$

MOST is set to the minimum of the current ask and the redemption value of the least valuable token minus (provided the current ask > 0), or just the value of the least valuable token less one otherwise.

The buy/sell decision is based on target value defining a "good price" which is raised towards Skeleton's redemption value as the time since the last trade increases. The current offer is only accepted if it is below this target value. The target value is initially set at 1.3 times the least valuable token less .3 times the most valuable token. Subsequent target values are given by  $a \cdot \text{target} + (1-a) \cdot \text{next token value}$ , where  $a = 1/(t - \text{lasttime})$  where  $t$  is the current time and lasttime is the last time any trade occurred. Skeleton's selling strategy is symmetric.

Skeleton did not participate in the 1990 SFI DA tournament, but performed above average in subsequent "scientific tournaments".

SLIDE

Mark Van Boening, Shawn LaMaster  
Economic Science Laboratory, University of Arizona, Tucson, Arizona

This program collects bids/offers and computes a prediction of price (bid or offer) based on a OLS regression of a sliding window of 15 observations. This strategy is basically computing a trend, estimating the coefficients from the regression and computing a prediction to bid or offer.

CLASSIFICATION: simple,nonadaptive,stochastic,predictive,optimizing

Rank in 1990 SFI DA tournament: 16th out of 29  
STAECKER

MARK STAECKER  
UNIVERSITY OF WESTERN ONTARIO  
LONDON ONTARIO CANADA

Strategy: The basic strategy of this program is to predict what will occur in the next step, then attempting to maximize my profits by using this information. The program attempts to predict the next low offer, the next high bid, and the equilibrium trading price by using general statistics on past behavior. Then, using these three predictions, it decides if a transaction is likely to occur on the next step. If it is, it attempts to beat the best price, otherwise, it does not attempt to beat the best price. Further details on Staecker's strategy can be found in paper "Characterizing Effective Trading Strategies:

Insights from a Computerized Double Auction Tournament" by Rust, Miller and Palmer (forthcoming Journal of Economic Dynamics and Control, 1993) and available via anonymous ftp from fido.econ.arizona.edu in /azte/papers/da2.ps

Classification: complex, adaptive, non-stochastic, predictive, optimizing

Rank in 1990 SFI DA tournament: 2nd out of 29

TERMINATOR

Shawn LaMaster, Mark Van Boening, Stephen Rassenti, Robert Dorsey

Original design by Dr. Robert Dorsey, University of Mississippi (see "Estimation and Simulation of the Double Auction" a discussion paper for the Economic Science Association meetings, 1989, available from Robert Dorsey, Dept. of Economics, University of Mississippi, University MS 38677.)

The Terminator strategy is derived from an empirical analysis of human behavior in experimental DA markets.

Terminator uses a 3 part linear decay rule for setting bids and asks. As a starting point, it uses the minimum seller value/maximum buyer value computed from the game type. Current parameters have Terminator decay 30 percent over the first .5 ntimes, 50 percent over the next .3 ntimes, and 20 percent over the last .2 ntimes. The buy/sell decision is based on a straight linear decay of a measure of acceptable profit.

The original design uses an exponential decay. We have modified the design to use a piecewise linear decay.

Rank in 1990 SFI DA tournament: 11th out of 29

TRUTHTELLER

John Rust  
University of Wisconsin  
Madison, WI 53706

This program implements naive truth-telling strategy in which the program bids a value equal to its current token value and accepts any current ask for which it can make a profit. The selling strategy is symmetric.

Classification: simple, nonadaptive, nonstochastic, nonoptimizing, nonpredictive

Program did not participate in 1990 SFI DA tournament, but did very poorly in subsequent scientific tournaments.

UTGOFF

Paul E. Utgoff  
Lederle Graduate Research Center  
University of Massachusetts  
Amherst, MA 01003

Strategy is based on fitting two quadratics (one for bids, other for offers) to the three most recent bids/offers, using time as the x-axis. One can then extrapolate, drawing inferences about whether, when, and where the two curves might intersect in the future, or (more likely) when and where they come closest. See comments below for details. This could be improved quite a bit with some more effort.

Classification: simple, nonadaptive, stochastic, predictive, nonoptimizing

Rank in 1990 SFI DA tournament: 21st out of 29

WENDROFF

Harvey Rose and Burton Wendroff  
Los Alamos National Laboratory  
Los Alamos NM 87545

Bid-Offer: We determine an equilibrium price by averaging over previous trades. Our bid-offers are constrained to be within two standard deviations of that price (random in that interval).

Buy-Sell: If offer is too much higher than the equilibrium we reject it, otherwise we accept.

CLASSIFICATION: simple, adaptive, nonstochastic, predictive, nonoptimizing

Rank in 1990 SFI DA tournament: 25th out of 29

ZI

This is a version of the "Zero Intelligence" strategy of Gode and Sunder (see chapter 7 by Gode and Sunder in "The Double Auction Market: Institutions, Theories and Evidence" Friedman and Rust (eds.)

The ZI program simply bids a random amount below its current redemption value and MINPRICE, and a ZI seller asks a random amount above its token cost and MAXPRICE. A ZI trader will accept any bid or ask if it can make a positive profit.

CLASSIFICATION: simple, nonadaptive, stochastic, nonpredictive, nonoptimizing

Program did not participate in 1990 SFI DA tournament, but did poorly in subsequent "scientific tournaments".

### SILVER BUFFALO

Greg Fullerton 4771 Tantra Boulder CO. 80303  
Mark Cronshaw CU, Dept of Econ. Box 256, Boulder CO. 80309-0256  
BJ Lee CU, Dept of Econ. Box 256, Boulder CO. 80309-0256  
Jamie Kruse CU, Dept of Econ. Box 256, Boulder CO. 80309-0256

This program bids based on risk calculated by time and tokens left. It starts by giving a random increment above the last token or cbid then when available it uses the past increment and estimated equilibrium calculated by a regression on past bids and offers.

Rank in 1990 SFI DA tournament: 15th out of 29

### SKELETON

Richard Palmer, Duke University  
John Miller, Carnegie Mellon University

This strategy is the simple example strategy distributed to all potential entrants to the 1990 SFI double auction tournament. A block diagram of this strategy appears in the paper "Characterizing Effective Trading Strategies: Insights from a Computerized Double Auction Tournament" forthcoming in the Journal of Economic Dynamics and Control 1993 and also available by anonymous ftp from /pub/azte/papers at fido.econ.arizona.edu.

The bidding strategy of the Skeleton program (the strategy for offering is symmetric) depends on whether or not there is a current bid on the table. If  $cbid > 0$ , skeleton bids a randomly weighted average of  $cbid + 1$  and the variable

MOST defined as the minimum of the current ask and the program's redemption value minus one (provided the current ask  $> 0$ ) or the redemption value minus one otherwise. If there is no current bid ( $cbid = 0$ )

then skeleton bids a value of MOST less  $U(\text{first token} - \text{last token})$  where  $U$  is a uniform  $(0,1)$  random variable and first token is the redemption value of Skeleton's most valuable token and last token is the redemption value of skeleton's least valuable token. When  $\text{cbid}=0$  MOST is set to the minimum of the current ask and the redemption value of the least valuable token minus (provided the current ask  $> 0$ ), or just the value of the least valuable token less one otherwise.

The buy/sell decision is based on target value defining a "good price" which is raised towards Skeleton's redemption value as the time since the last

trade increases. The current offer is only accepted if it is below this target value. The target value is initially set at 1.3 times the least valuable token less .3 times the most valuable token. Subsequent target values are given by  $a \cdot \text{target} + (1-a) \cdot \text{next token value}$ , where  $a = 1/(t - \text{lasttime})$  where  $t$  is the current time and lasttime is the last time any trade occurred. Skeleton's selling strategy is symmetric.

Skeleton did not participate in the 1990 SFI DA tournament, but performed above average in subsequent "scientific tournaments".

SLIDE

Mark Van Boening, Shawn LaMaster  
Economic Science Laboratory, University of Arizona, Tucson, Arizona

This program collects bids/offers and computes a prediction of price (bid or offer) based on a OLS regression of a sliding window of 15 observations. This strategy is basically computing a trend, estimating the coefficients from the regression and computing a prediction to bid or offer.

CLASSIFICATION: simple,nonadaptive,stochastic,predictive,optimizing

Rank in 1990 SFI DA tournament: 16th out of 29

STAECKER

MARK STAECKER  
UNIVERSITY OF WESTERN ONTARIO  
LONDON ONTARIO CANADA

Strategy: The basic strategy of this program is to predict what will occur in the next step, then attempting to maximize my profits by using this information. The program attempts to predict the next low offer, the next high bid, and the equilibrium trading price by using

general statistics on past behavior. Then, using these three predictions, it decides if a transaction is likely to occur on the next step. If it is, it attempts to beat the best price, otherwise, it does not attempt to beat the best price. Further details on Staecker's strategy can be found in paper "Characterizing Effective Trading Strategies:

Insights from a Computerized Double Auction Tournament" by Rust, Miller and Palmer (forthcoming Journal of Economic Dynamics and Control, 1993) and available via anonymous ftp from fido.econ.arizona.edu in /azte/papers/da2.ps

Classification: complex, adaptive, non-stochastic, predictive, optimizing

Rank in 1990 SFI DA tournament: 2nd out of 29

TERMINATOR

Shawn LaMaster, Mark Van Boening, Stephen Rassenti, Robert Dorsey

Original design by Dr. Robert Dorsey, University of Mississippi (see "Estimation and Simulation of the Double Auction" a discussion paper for the Economic Science Association meetings, 1989, available from Robert Dorsey, Dept. of Economics, University of Mississippi, University MS 38677.)

The Terminator strategy is derived from an empirical analysis of human behavior in experimental DA markets.

Terminator uses a 3 part linear decay rule for setting bids and asks. As a starting point, it uses the minimum seller value/maximum buyer value computed from the game type. Current parameters have Terminator decay 30 percent over the first .5 ntimes, 50 percent over the next .3 ntimes, and 20 percent over the last .2 ntimes.

The buy/sell decision is based on a straight linear decay of a measure of acceptable profit.

The original design uses an exponential decay. We have modified the design to use a piecewise linear decay.

Rank in 1990 SFI DA tournament: 11th out of 29

TRUTHTELLER

John Rust

University of Wisconsin  
Madison, WI 53706

This program implements naive truth-telling strategy in which the program bids a value equal to its current token value and accepts any current ask for which it can make a profit. The selling strategy is symmetric.

Classification: simple, nonadaptive, nonstochastic, nonoptimizing, nonpredictive

Program did not participate in 1990 SFI DA tournament, but did very poorly in subsequent scientific tournaments.

UTGOFF

Paul E. Utgoff  
Lederle Graduate Research Center  
University of Massachusetts  
Amherst, MA 01003

Strategy is based on fitting two quadratics (one for bids, other for offers) to the three most recent bids/offers, using time as the x-axis. One can then extrapolate, drawing inferences about whether, when, and where the two curves might intersect in the future, or (more likely) when and where they come closest. See comments below for details. This could be improved quite a bit with some more effort.

Classification: simple, nonadaptive, stochastic, predictive, nonoptimizing

Rank in 1990 SFI DA tournament: 21st out of 29

WENDROFF

Harvey Rose and Burton Wendroff  
Los Alamos National Laboratory  
Los Alamos NM 87545

Bid-Offer: We determine an equilibrium price by averaging over previous trades. Our bid-offers are constrained to be within two standard deviations of that price (random in that interval). Buy-Sell: If offer is too much higher than the equilibrium we reject it, otherwise we accept.

CLASSIFICATION: simple, adaptive, nonstochastic, predictive, nonoptimizing Rank in 1990 SFI DA tournament: 25th out of 29

ZI



This is a version of the "Zero Intelligence" strategy of Gode and Sunder (see chapter 7 by Gode and Sunder in "The Double Auction Market: Institutions, Theories and Evidence" Friedman and Rust (eds.)

The ZI program simply bids a random amount below its current redemption value and MINPRICE, and a ZI seller asks a random amount above its token cost and MAXPRICE. A ZI trader will accept any bid or ask if it can make a positive profit.

CLASSIFICATION: simple, nonadaptive, stochastic, nonpredictive, nonoptimizing

Program did not participate in 1990 SFI DA tournament, but did poorly in subsequent "scientific tournaments".