Simulation of trust in client-wealth management adviser relationships

Terry Bossomaier*

CRiCS Centre for Research in Complex Systems,
Charles Sturt University,
Panorama Avenue, Bathurst, NSW, Australia
Fax: +61 2 6338 469
E-mail: tbossomaier@csu.edu.au
*Corresponding author

Russell K. Standish

School of Mathematics and Statistics,
University of New South Wales,
Sydney, 2052 NSW, Australia
Fax: +61 2 6338 469
E-mail: hpcoder@hpcoders.com.au

Michael Harré

CRiCS Centre for Research in Complex Systems,
Charles Sturt University,
Panorama Avenue, Bathurst, NSW, Australia
Fax: +61 2 6338 469
E-mail: mike@centreforthemind.com

Abstract: This paper describes the components of a two-phase model for simulating trust amongst clients and their Wealth Management Advisers (WMAs). In phase one, an artificial life model was used to assess the dynamics of trust. In phase two, the model is extended to utilise real data from a corporate database of client information. The Alife model highlighted the need for information not captured directly, requiring sophisticated inference techniques. Fuzzy logic is used to describe client behaviour with rules found through evolutionary optimisation. Analysis of mutual information between time series of clients' investments is used to determine links between clients.

Keywords: agent-based model; trust; financial adviser; fuzzy logic; mutual information; social network.

Reference to this paper should be made as follows: Bossomaier, T., Standish, R.K. and Harré, M. (2010) 'Simulation of trust in client-wealth management adviser relationships', *Int. J. Simulation and Process Modelling*, Vol. 6, No. 1, pp.40–49.

Biographical notes: Terry Bossomaier is a Professor of Computer Systems and Director of the Charles Sturt University Centre for Research in Complex Systems. His research covers many areas of complex systems, ranging from biological vision to agent-based models of socio-economic systems.

Russell K. Standish runs a Consultancy Business specialising in high-performance computing and computational science. His academic research is in the domains of complex systems (theoretical descriptions of life, cellular automata, chaos, information theory, etc.), ontological theories of everything. His applied interests cover high-performance computing, computational science and agent-based modelling.

Michael Harré is a Research Fellow at the Centre for Research in Complex Systems at Charles Sturt University. He recently completed his PhD at the University of Sydney in sub-rational game theory. Currently, he is researching phase transitions in financial markets and other complex systems.

1 Introduction

Today's companies operate in a very complicated and sometimes turbulent environment. Unexpected changes in global resources such as the food crises in 2008 and the escalating oil price can dramatically change market positions. On the other hand, the sub-prime mortgage meltdown has revealed just how complicated and fragile financial systems can be.

Decision-support systems and the mining of the vast quantities of consumer data in corporate data warehouses are valuable but their capacity to predict future requirements is often limited to narrow extrapolation from the past. More powerful scenario planning systems are needed, which can explore new trends, such as, for example, the rapid switch of agricultural land from food to biofuel production, a powerful influence with little prior history.

Agent-Based Models (ABMs) attempt to model people, companies and external forces and to go beyond simple extrapolation. From early minimalist models, some ABMs now incorporate many millions of agents, such as the large Epicast models for studying pandemics in the USA.

Until a few years ago, Australian workers were, in general, locked into the superannuation fund provided by their employer. But, following deregulation, people were allowed to choose to which of many available funds they belonged. Inevitably, this created a significant market for financial advisers. At first, such services were poorly monitored and advice was not always sound. Worse, there were a number of concerns about conflict of interest, where advisers were given hidden trailing commissions or promoted a fund owned by their parent organisation. This issue is still making news in early 2009!

Thus, trust in financial services became an important issue for banks and other providers to confront. This project focused on building an ABM with two distinct goals

1 To examine the dynamics of trust, to look for phase transitions and indicators of declining trust. This was essentially an abstract of artificial life model.

2 To build a realistic ABM derived from real data.

The artificial life model (Bossomaier et al., 2005) consisted of sets of clients and WMAs and an artificial stock market. The advisers compete with one another to maximise their wealth, whereas the clients share trust information with one another.

The stock market simulates shares or share portfolios, which are owned by the client and are paid for with upfront consultancy fees, and funds, aggregates of shares from the same market, which pay commissions. The growth in value of the shares follows a deterministic equation with added noise. Advisers may purchase access to the growth parameters of this equation, thus trading off costs against better advice to clients.

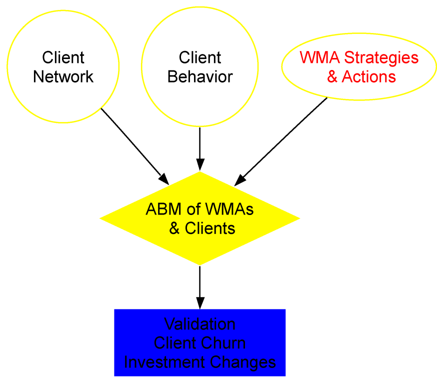
Clients' trust evolves according to their returns on investment, using insights from the neuroeconomics literature. Each client is connected to other clients on one of a number of different types of network, ranging from lattice and small world to scale-free.

Moving to the full-scale model involves analysing a large corporate database of several million clients. Figure 1 shows the overall design of the model. This paper addresses only the determination of client networks and behaviour shown as circles in the figure. Client behaviour is, of course, very difficult to fully capture. The model here focuses on risk-taking propensity, as a building block of investment decisions.

Analysis of the client network is not possible from any fields in the database itself. Thus, indirect methods are needed to infer the connectivity. A novel approach has been adopted, which involves looking for common patterns in investment over time among clients.

Section 2 outlines the Alife model. Sections 3–5 discuss the data sets, the client behaviour and the network analysis, respectively.

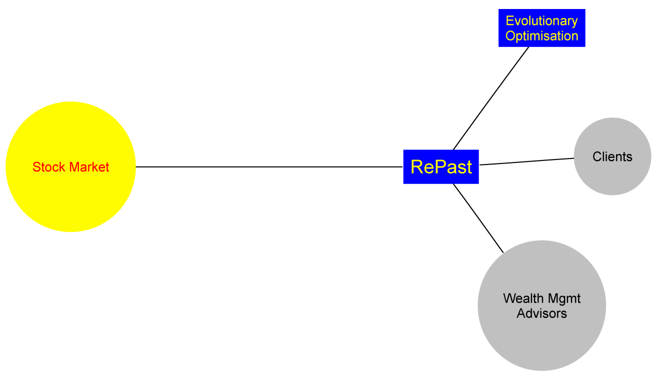
Figure 1 Flow chart illustrating the components and information flow in the ABM (see online version for colours)



2 The Alife model

The Alife model consists of several components as shown in Figure 2.

Figure 2 Architecture of the trust Alife model (see online version for colours)



Wealth Management Advisers invest money on behalf of their clients taking some profit along the way. They invest the clients' money in a mixture of shares and funds.[1] Shares return a percentage of the investment as a one-off fee. Funds provide an ongoing trailing commission.

Clients split their wealth into a fraction invested with their WMA and leave the rest in the bank, the fraction being determined by their trust level. If their trust falls significantly below the trust of their neighbours, they will churn to another adviser.

The stock market is not intended to represent the real stock market in any detail and the many diverse investment options it provides. It merely provides an investment framework.

Simulations of this model allowed the study of the evolution of trust under various conditions. One important issue is the nature of the client networks. If the clients are unconnected, then their trust will go up and down with WMA performance and the WMA can trade off the additional investment they get with increased trust against the potential loss of fees and commissions. But, if the clients can talk to each other, then the WMAs now have to outperform each other to avoid loss of clients to others who provide better net returns.

Much recent interest in networks has led to three common types in social systems: simple local connectivity, such as a lattice; small world networks in which additional long-range connections are added (Watts, 1999) scale-free networks characterised by a power law distribution in the connectivity of nodes (Barabási, 2002). Different social networks could lead to different trust flow behaviours, hence modelling these networks is important. However, this information may not be readily available.

2.1 The stock market

There are many stock market models in the literature and this paper tries neither to improve on them, nor to even select the best. Two factors drive the approach:

- the model must be efficient in computing resources
- it must have a natural transition to real financial instruments or products and must reflect investment in research activity into the value of different sorts of investment.

A simple, illustrative, model satisfying these requirements used data from the New York Stock Exchange obtained via Yahoo finance (Yahoo, 2008). An exponential fit to this data is perturbed by an additional noise term using Brownian motion as in equation (3). Stocks are evolved according to a multiplicative stochastic process

$$\mathrm{d}y = \mu y \mathrm{d}t + \sigma y \mathrm{d}W, \tag{1}$$

where W is a standard Wiener process. This has an analytic solution

$$y(t) = y_0 \exp((\mu - \sigma^2/2)t + \sigma W). \tag{2}$$

The explicit algorithm is

$$y(t) = A_0 \exp(a_1 * t + a_2 * c), \tag{3}$$

where A_0, a_1, a_2 are constants and c is a cumulative uniformly distributed random number in the range [0, 1].

The full bank model uses the financial instruments constructed by the bank and their variation over time as measured on the real stock market.

3 The data sets

A large data set of 456 million records describing 14 million customers over a five-year period, and a smaller more detailed data set of 42 million records describing the investment profiles of 1.5 million customers over a three-year period were supplied to the project.

The larger data set was collected over a 60-month period, whereas the smaller one was collected over a shorter time period of 24. However, it had the advantage that the balances for each of 20 financial instruments and the market performance were available. Thus, the analysis used the smaller data set, referred to hereafter as just the data set.

From the data set, it was possible to establish the investment profiles of the 1.5 million investment customers. Whilst product performance was not available in this database, it was possible to match the internal product identifiers to published product information, and to download the relevant product performance from the bank's website. By defining a product's risk as the variance of its performance (historical volatility), one can estimate a customer's risk profile by taking the product balance weighted average of the product's risk. This will fluctuate over time as the product balances change (unless the customer is invested in a single product only), but if the customer performs active portfolio balancing, this will reasonably accurately reflect the customer's risk preference. The resulting time series has six independent variables (age, length of customer relationship, gender, marital status, deceased and number of dependents), and one dependent variable (risk).

4 Modelling client behaviour

Human behaviour may be represented in several ways, each extremely diverse (Fulcher, 2008). Artificial Neural Networks (ANNs) are loosely linked to the structure and operation of the human brain but there are many architectures and training or learning algorithms from which to choose. At the other extreme to ANNs are formal rule-based systems. But, representing human behaviour with rules is tricky, often requiring very large rule sets. Yet, these two extremes ultimately have to converge to the same outcomes since both are capable of arbitrary accurate representations.

Fuzzy logic falls somewhere in between. Its advantage in ABMs arises from the interdisciplinary nature of socio-economic modelling. Qualitative research outcomes and judgements from domain experts can be readily transcribed into fuzzy logic and its conception was in part motivated by the semi-quantitative style of much human thinking. Thus, fuzzy logic is the methodology used herein.

4.1 Fuzzy Inference Systems

Fuzzy sets are sets whose elements have a degree of membership in the range [0, 1]. More precisely, a fuzzy set F is a pair $F = (A, \phi)$, where A is a set, and ϕ: $A \rightarrow [0, 1]$ is the membership function. If $\phi(x) = 0$, then x is not considered to be a member of the fuzzy set F, and if $\phi(x) = 1$ then x is considered fully included in the fuzzy set.

Fuzzy logic extends the notion of propositional logic to fuzzy sets with the fuzzy logic operators *AND*, *OR* and *IS*. Fuzzy logic rules are of the form:

$$\begin{aligned} &IF\ x_1\ IS\ I_{11} \\ &AND(x_2\ IS\ I_{21}\ OR\ x_2 IS\ I_{22})\ldots \\ &\quad THEN\ y\ IS\ O_1. \end{aligned} \tag{4}$$

There are a variety of Fuzzy Inference Systems (FISs). The *Mamdani* type, used here, consists of a number of input variables x_i, whose ranges are partitioned into fuzzy sets I_{ij}, an output variable y whose range is partitioned into fuzzy sets O_j, and a set of fuzzy rules of the form (4). The FIS takes a vector of input values, and outputs an inferred value $\hat{y}$.

The *matching degree* $w(x_1, x_2, \ldots, x_n)$, or *weight*, for a rule is constructed from the antecedent, where the *IS* operator is replaced by the membership function, *AND* is replaced by a binary operator $\wedge$ called a *t-norm*, and *OR* is replaced by its *t*-conorm, $\vee$, where $a \vee b = 1-(1-a) \wedge (1-b)$. Simple examples of *t*-norms are the minimum of the two argument and the product of its arguments, e.g., with rule (4):

$$w(x_1, x_2, \ldots) = \phi_{I_{11}}(x_1) \wedge (\phi_{I_{21}}(x_2) \vee \phi_{I_{22}}(x_2))\ldots \tag{5}$$

For the purposes of this work, $\wedge = \min$ and $\vee = \max$ were used to initially generate the FIS, but the Evolutionary Algorithm (EA) was allowed to mutate these to the product *t*-norm or the Łukasiewicz *t*-norm $(x \wedge y = \max(0, x + y - 1))$.

For converting the weight value w computed in equation (5) into the output value $\hat{y}$ of the inference system, one needs to aggregate the weights of all rules with the same consequent (the part following THEN in rule (4)), and then apply a *defuzzification operator*. Aggregation involves taking either the maximum, or the sum of the weights, of all rules with the same consequent. In this work, we used the sum aggregation rule, i.e.:

$$W_j = \sum_{\{r|c_r = "y\ IS\ O_j"\}} w_r(x) \tag{6}$$

where c_r is the consequent of the rth rule and w_r is the weight of the rth rule.

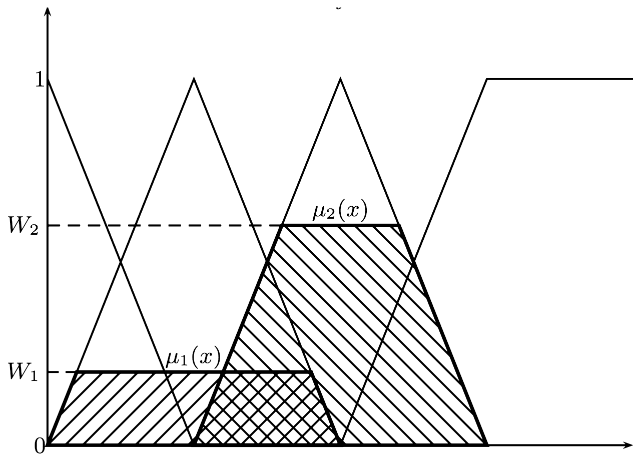
Finally, to produce an inferred output $\hat{y}$, one needs to defuzzify the aggregate weights W_j. A number of possible operators can be employed for this task, but the one we use herein is known as *area defuzzification*. Form new functions

$$\mu_j(x) = \begin{cases} m_{O_j}(x) & \text{if } m_{O_j}(x) < W_j, \\ W_j & \text{otherwise} \end{cases} \tag{7}$$

as shown in Figure 3. Then, the inferred output is the centroid of the area underneath the sum of the $_j$ curves:

$$\hat{y} = \frac{\int x \sum_j \mu_j(x)\mathrm{d}x}{\int \sum_j \mu_j(x)\mathrm{d}x}. \tag{8}$$

Figure 3 Membership functions and area defuzzification



4.1.1 Fuzzy logic source code

FISPRO (2008) is an open-source software package available from the French Government, Institut National de la Recherche Agronomique (INRA). Implementing FIS, allowing a wide variety of membership functions, different forms of weight computation, aggregation and defuzzification to be specified. It is implemented as a C++ library, with a Java interface provided through JNI that enables an interactive Java programme that users can use to design FISs, and experiment with the inference engine.

Additionally, FISPRO provides a number of functions for learning rules from training datasets. These were used to initially seed the EAs.

Various performance enhancements were added to FISPRO 3.0 during this project and submitted to the FISPRO maintainers for inclusion into the next release of FISPRO.

4.2 Parameterising the FIS from real-world data

Since around 1990, people have sought to combine the knowledge representation power of FISs with the learning power of EA, particularly genetic algorithms. Alander (1997) noted that some 280 papers have been published on the topic by early 1996. A ten-year survey by Cordón et al. notes the different types of approaches taken to evolving FIS (Cordón et al., 2004). Different aspects of the FIS are available to be evolved: the type of FIS (whether Mamdani, or Takagi-Sugeno), the *t*-norm used in the calculation of the rule weight (5), the rules and their consequents, the number and shapes of membership functions for the inputs and outputs, and the actual parameters of the membership functions. Most commonly, the parameters of the membership functions are evolved, or the rule base is evolved. The EA used is usually a genetic algorithm (parameters converted into a bitstring representation, which is evolved), although evolutionary strategies (working directly with floating point representations) are also deployed (e.g., Cordón and Herrera, 1999), as we do here. When evolving the rule base, individuals of the EA may either be complete FIS (called the Pittsburgh approach) as used in this work, or individual rules (Michigan or Iterative approaches).

In the EvoNF framework (Abraham, 2002), all of these aspects can be tuned, but in practice evolving all levels of this framework is computationally prohibitive. Normally, domain knowledge is used to constrain the optimisation search space. We had initially hoped to evolve just the membership function parameters, with the fuzzy rule base being given by domain knowledge. However, the domain knowledge turned out to be insufficient for the task, so we chose to inform the rule base from the data, by seeding the evolving population using the Fast Prototyping Algorithm (FPA) (Glorennec, 1996) and then further evolving the rule base.

The FPA consists of generating all possible combinations of antecedents, and computing the average matching degree *w* from the weight of each antecedent over all rows of the training dataset. Rules with matching degree less than a certain threshold (0.01) were dropped from the rule base. A conclusion for a particular rule was computed by the matching-degree weighted average over all data rows of the fuzzy set index corresponding to the output field of the training dataset, rounded to the nearest integer.

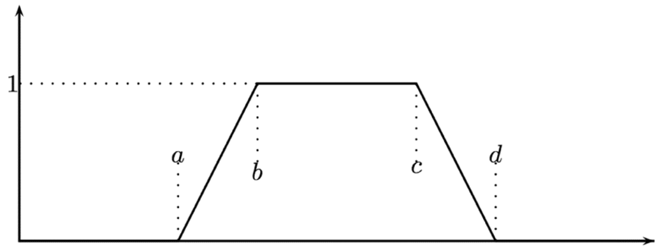
4.3 Representation of the FIS

An EA requires a representation of the solution, a sequence of evolutionary operators (genetic operators) to generate variation and a selection criterion for removing unsuccessful solutions from the pool.

In this work, we use a direct representation in the form of a list of the parameters for all the membership functions of the input and output fuzzy sets. We also differentiate which rules are active, and what their consequents are.

For computational efficiency (avoiding the extensive computation of exponentials or other such functions), we use piecewise linear trapezoidal membership functions. This includes triangular and semi-trapezoidal membership functions as a special case. Figure 4 shows the general form of the membership function, and defines the term *core*, where the membership function is 1, and the *support* where the membership function is greater than 0.

Figure 4 A piecewise linear *trapezoidal* membership function. The area between b and c is known as the *core* and the area between a and d the *support*. If $b = c$, the function is called *triangular*, if $a = b = -\infty$, or $c = d = \infty$ it is known as *infimum* or *supremum semi-trapezoidal* respectively



4.4 Evolutionary operators

The operators we implemented were mutation, insertion (splitting), deletion (merging) and crossover, each controlled by a separate parameter.

In the case of mutation, with probability given by the mutation probability parameter `mutConc` (0.01), a number was drawn from $\{0, \ldots, N_c\}$, where N_c is the number of fuzzy sets in the FIS output. If the number drawn was N_c, the rule was toggled between active and inactive, otherwise the rule's conclusion was set to the fuzzy set corresponding to the number drawn. Similarly, with probability `mutConj` (0.01), the conjunction operator $\wedge$ was mutated between minimum, product and Łukasiewicz *t*-norms (which are the *t*-norms supported by FISPRO).

Mutating membership function parameters is a little more complex. We need to ensure that the fuzzy sets do not expand to engulf neighbouring fuzzy sets, so we require that

a the support of a membership function does not overlap the core of its neighbours

b the support to overlap the support of its neighbour to ensure complete coverage

c $a \le b \le c \le d$.

We also limit the maximum variation to an evolutionary parameter `mutrange` (=0.1). In practice, this means that we calculate a range that maximally satisfies all those constraints, and chose a value randomly from that range. More precisely, if a_i, b_i, etc., represent the parameters of the ith membership function, the core is given by $[b_i, c_i]$ and the support by $[a_i, d_i]$. Let μ = `mutrange`. Then, we form:

$$\begin{aligned}
a_i^- &= \max\{a_i - \mu, c_{i-1}\}\\
a_i^+ &= \min\{a_i + \mu, b_i, d_{i-1}\}\\
a_i' &\in [a_i^-, a_i^+]\\
b_i^- &= \max\{b_i - \mu, a_i'\}\\
b_i^+ &= \min\{b_i + \mu, c_i\}\\
b_i' &\in [b_i^-, b_i^+]\\
c_i^- &= \max\{c_i - \mu, b_i'\}\\
c_i^+ &= \min\{c_i + \mu, d_i\}\\
c_i' &\in [c_i^-, c_i^+]\\
d_i^- &= \max\{d_i - \mu, c_i', a_{i+1}\}\\
d_i^+ &= \min\{d_i + \mu, b_{i+1}\}\\
d_i' &\in [d_i^-, d_i^+].
\end{aligned} \tag{9}$$

The a_i', b_i', c_i' and d_i' become the new mutated value of the parameters. If any of the $x_i^- > x_i^+$, it is not possible to draw a new value for parameter x_i, so the parameters are left unchanged.

The *insertion operator* was implemented by replacing a membership function with two new membership functions that split the original core between them. If *a*, *b*, *c*, *d* are the original function's parameters, the new functions' parameters are:

$$\begin{aligned}
a_1 &= a\\
b_1 &= c_1 = b\\
d_1 &= c\\
a_2 &= b\\
b_2 &= c_2 = c\\
d_2 &= d.
\end{aligned} \tag{10}$$

The resulting membership functions are triangular, but need not remain that way after further mutation.

The *deletion operator* was implemented by replacing a pair of consecutive membership functions with a single merged function:

$$\begin{aligned}
a &= a_1\\
b &= b_1\\
c &= c_2\\
d &= d_2.
\end{aligned} \tag{11}$$

These operators were applied with probability `split` (0.01) and `merge` (0.01), which are parameters of the EA. Because these operations change the number of membership functions describing an input variable, we end up with oddities such as rule bases with three genders, so in practice we also specified a Boolean input array to indicate which inputs, and whether the output could have split/merge applied to them. Furthermore, changing the numbers of membership functions invalidates the ruleset. Rather than renumbering the ruleset (which would require resolving the issue of which of the two previous fuzzy sets maps to the new fuzzy set when a merge has happened), we took the approach of reapplying the FPA to regenerate a new rule base from scratch.

The final evolutionary operator was the crossover operator. This simply selected two parents at random from the pool, and crossed the membership functions for each input with 50% probability, and also crossed rules with matching antecedents with 50% probability (or performed an insertion of a parent 2 rule if its antecedent does not exist in parent 1). A later step in the EA removes identical FIS from the pool.

4.5 Evolutionary optimisation procedure

A pool of FIS is seeded with an FIS generated using FPA (Glorennec, 1996). The FIS pool is iterated over, with the evolutionary operators described in Section 4.4 applied according to the controlling probabilities. Once the number of FIS in the pool reached `maxPop` (10), selection is applied.

The primary indicator of fitness of the FIS is Root Mean Square Error (RMSE) *P*, which is the RMSE of the FIS with respect to the training dataset σ and has to be *minimised*. Some of the items in the training dataset may not match any of the rules well, particularly as the input fuzzy partitions evolve. If the maximum *w* computed according to equation (5) is less than ζ (`matchThresh` = 0.01), then the item is dropped from the training set. Let $\sigma' = \{j \in \sigma: w(j) \geq \zeta\}$, then RMSE is calculated from

$$P = \sqrt{\frac{1}{|\sigma'|}\sum_{i\in\sigma'}(\hat{y}_i - y_i)^2}, \tag{12}$$

and the coverage *C* as

$$C = \frac{|\sigma'|}{|\sigma|}. \tag{13}$$

The advantage of using this is that the FIS can report when its predictive ability is poor, and one can substitute an alternative inference rule (such as random selection from a probability table).

Using RMSE (equation (12)) directly as a fitness function encourages the algorithm to find solutions that diminish coverage. By eliminating difficult to predict conclusions, *P* can be made arbitrarily small, even zero. If one sets `matchThresh` to zero (ensuring all of the training set is used), then *P* is dominated by the poorly performing rules, and the EA has difficulty finding improvements.

An improvement to using *P* directly is to combine coverage, for instance as a ratio *P/C* or as a linear combination $P + \alpha/C$. The former fitness function is particularly prone to finding an FIS that reduces coverage to the point that $P = 0$, which then dominates the evolutionary pool. The latter fitness function has the α parameter, and the algorithm stagnates once $P \leq \alpha$.

This is a problem of multi-objective optimisation (simultaneously minimising *P* at the same time as keeping *C* as high as possible). An alternative approach to multi-objective EA is Pareto optimisation (Abbass, 2006), whereby only Pareto-dominated FIS candidates are eliminated (those for which other FIS in the population are better at both RMSE and coverage). In practice, this algorithm worked the best of all.

4.6 Implementation and results

The EA was coded in C++ as a model running under `EcoLab` (Standish and Leow, 2003), available from the `EcoLab` website.[2] OpenMP (OpenMP, 2002) was used to parallelise the computation of RMSE and coverage, as well as repopulating the rule base after a change in the number of fuzzy sets describing the inputs or output. The source code is the NCR.D5 release and relies on the modified version of FISPRO 3.0 (fispro.3.0.D10 see Section 4.1.1). Both are available from the same (`EcoLab`) website.

The data set was further downsized by sampling the customers with a frequency 0.01 and of 0.001. This resulted in 285,660 records for the 0.01 sampling frequency, and 30,627 for the 0.001 sampling frequency. Various rule induction options available within FISPRO were tried, but only the FPA option could handle such large data sets.

Using a single objective function, P/C could, in exceptional cases, give good coverage but tended to achieve a coverage of around 80–85%.

With the matching threshold parameter set to 0, coverage is 100% by definition. The EA still improved the starting ruleset found by FPA, but it does not produce as good a solution as the multi-objective methods. In the multi-objective case, the Pareto front of the best solutions occurs near the edge of the 'cliff' where coverage drops off precipitously. The best solution at the end of the run has $P = 3.39 \times 10^{-11}$ and $C = 1.0$. Extending the data set to the 0.01 sample rate achieves an RMSE rate of 0.077 with complete coverage.

Thus, optimising just on RMSE was not nearly as effective as optimising both P and C.

The actual evolved FIS had 53 rules, of the form "single middle-aged living female that is a medium term customer has a moderate risk preference". The fuzzy membership functions define what each of these terms mean, but the evolved membership functions appeared non-intuitive. On the gender axis, for instance, the evolved membership functions were such that both male and female rules were activated for all customers, but for male customers the female rule was about 10% of the male rule and for female customers the male rule had about a 35% weighting of the female rule. For some of the input variables, such as age or length of relationship with the bank, the membership functions separated into non-overlapping fuzzy sets, indicating a need for constraints to be applied to the algorithm to ensure that the fuzzy sets overlap.

Finally, it is hoped to scale the algorithm up to analysing the full dataset of approximately 2.5 million records, using half as a training set and the second half to test coverage and RMSE. This is still work-in-progress.

5 Network analysis

The trust model requires understanding the social networks of clients. Although there is plenty of empirical evidence for small world (Watts, 1999) or scale-free connectivity (Barabási, 2002), there are no fields in the data warehouse, which capture the links between clients. Furthermore, there are no obvious proxies, e.g., children attending the same school, membership of the same sports clubs might all be harbingers of interactions, but no data of this kind is available. Hence, a deeper inference system was needed. The solution is in looking at the time series of investments and determining how one client's investments correlates with another. Correlation between time series is a well-understood metric but it can sometimes miss non-linear interactions completely. Mutual information is a more powerful, although computationally more demanding technique (Cellucci et al., 2005).

5.1 Methods

The investment timeseries of approximately 180,000 customers investing in products with known performance data was computed using the technique described in Section 3. Since the absolute balances of the product investment do not carry meaningful information, and even relative balance (monthly investment divided by product balance) is distorted, the time series was converted to the range $\{-1, 0, 1\}$, representing withdrawal, no investment and investment, respectively, in a product. Mutual information, M, was calculated in the usual way between these reduced investment timeseries $I(c, pr, t)$ for customer c, product pr at time t:

$$\begin{aligned} M(c_1, c_2; \Delta) &= \sum_{x.y} p(x, y) \ \mathrm{In} \frac{p(x, y)}{p_1(x) p_2(y)} \\ p(x, y) &\equiv p(I(c_1, pr, t) = x \\ &\qquad I(c_2, pr, t + \Delta) = y), \\ p_1(x) &\equiv p(I(c_1, pr, t) = x) \\ p_2(y) &\equiv p(I(c_2, pr, t + \Delta) = y) \\ (x, y) &\in \{-1, 0, 1\}^2 \mid (x, y) \neq (0, 0). \end{aligned} \tag{14}$$

Here, the probability distributions were computed from the histograms of the 20 investment products and all timeseries points t (January 2002–August 2003 inc).

Under certain circumstances, it is possible to infer the causal direction of a relationship between two customers from the mutual information. If we measure the mutual information between series A at time a and series B at time b, where $a < b$, and assuming a causative relationship between the processes that generate A and B, then there will be a $\Delta = b - a$, which maximises the mutual information between the two time series. If $\Delta > 0$, then A causes B and if $\Delta < 0$ then B causes A. Formally, offsetting the time series by Δ in the range $[-3, +3]$ months allows for possible causality to be inferred. By maximising the mutual information over Δ,

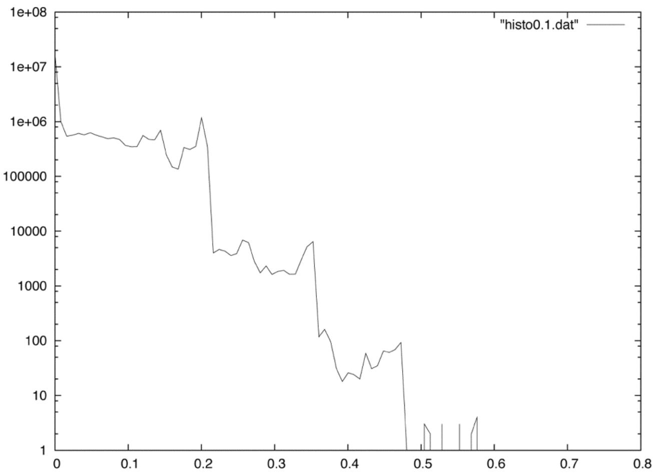
$$M(c_1, c_2) = \max_{\Delta} M(c_1, c_2; \Delta) \quad (15)$$

the sign of Δ for which the M is maximised induces a direction on the network link, pointing from c_1 to c_2 if Δ is positive, and vice versa.

The customers in the data set could be classified according to whether their investment patterns were static 'set-and-forget' (either no investment, or a regular monthly amount set up as an autopay), or actively managed. Therefore, the data set was further reduced to just those customers classified as active investors more than 5% of the time (around 120,000 customers) and then decimated to the final collection of 10,656 customers.

With a continuous measure of distance such as this one, the network is effectively fully connected. Hence, some thresholding is necessary. The approach here is to examine the histogram of M values as shown in Figure 5.

Figure 5 Histogram of mutual information values



The account balances against each of these are given monthly for a period of 20 months, making each investment series 420 points in length. Using this approach, the mutual information will show sharp changes as the number of products mutually invested in by the clients increases. In fact, few clients invest in more than a couple of products, leading to plateau-like characteristics. Note that if a client shows similar temporal investment behaviour, but with a different financial instrument, this will show up as a *low* mutual information value. Furthermore, the two time series are shifted relative to each other by up to 3 months, and the maximum mutual information taken. The sign of the offset value indicates a possible causal influence between the two clients, whereas a zero offset might indicate that both clients are under the influence of a third party (who may or may not be in the data set).

5.2 Network statistics

Figure 5 shows the histogram of the mutual information values. The regions of lower average slope suggest fundamental changes in client behaviour or relationships and possible thresholds for which to examine the networks. At a threshold of 0.4 or above, the networks essentially disappear, leading us to examine thresholds of 0.2 and 0.3.

Displaying graphs with very large numbers of nodes and edges is not easy. The approach used in this paper is to display the *adjacency matrix.* This is a matrix with rows and columns representing the nodes of the graph and each entry is one if an edge exists between two nodes and zero otherwise. To see structure in the network, some reordering of the rows and columns is usually necessary. The reordering process here is known as Reverse Cuthill McKee Ordering (RCMO) (Cuthill and McKee, 1969). The idea here is to order the nodes such that the *bandwidth* is minimised. The bandwidth is the largest path, in number of edges, between each pair of nodes. In effect, this means permuting the labels of the nodes (the clients) such that the edges (connections between them) lie as close to the diagonal as possible.

As a reference, Figure 6 shows the RCMO plot of a graph in which the edges are selected at random.

Figure 7 shows the network at a mutual information threshold of 0.2. This network has a very large number of connections. The RCM graph shows that a significant number of these nodes are highly connected on this metric.

So, although there are highly connected 'hub' nodes, the density and distribution of these is quite different to a scale-free network. These very highly connected nodes imply that the topological dimension of the network is high, with a high genus number, unlike the stock market data analysed by Tumminello et al. (2005). Figure 8 shows the network with the higher threshold of 0.3. About half the clients have disappeared, i.e., had very weak connections to any other client. Figure 9 zooms in on the most connected nodes. Again, we see clusters of nodes with very high levels of interconnectivity.

Figure 6 RCMO plot of random graph (see online version for colours)

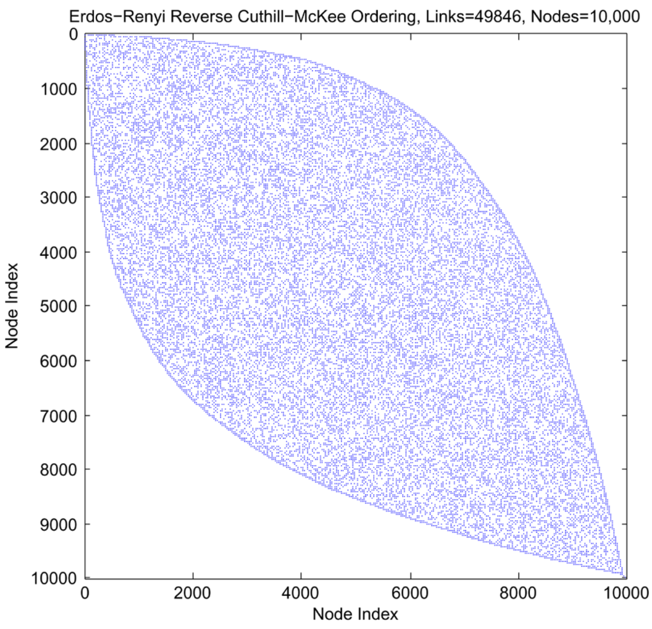


Figure 7 RCMO of mutual information threshold of 0.2. Some nodes are seen to be very highly connected (see online version for colours)

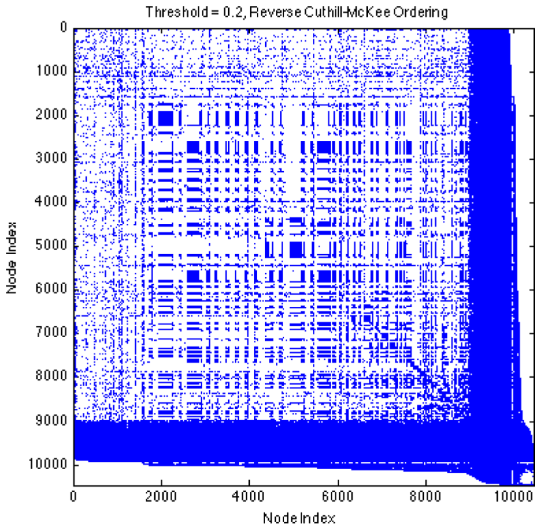


Figure 8 RCMO of mutual information threshold of 0.3 (see online version for colours)

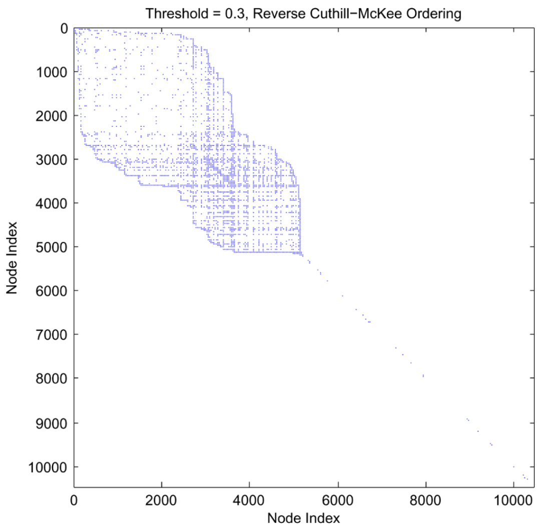
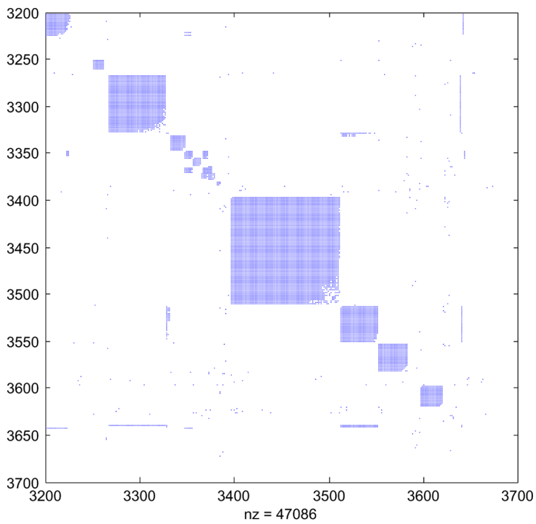


Figure 9 RCMO of mutual information of 0.3 expanded to show structure in most connected nodes (see online version for colours)



Much recent work in networks has realised two common types of network in social systems, scale free and small world. Scale-free networks have a power law distribution of node degree distributions (Barabási, 2002) according to equation (16)

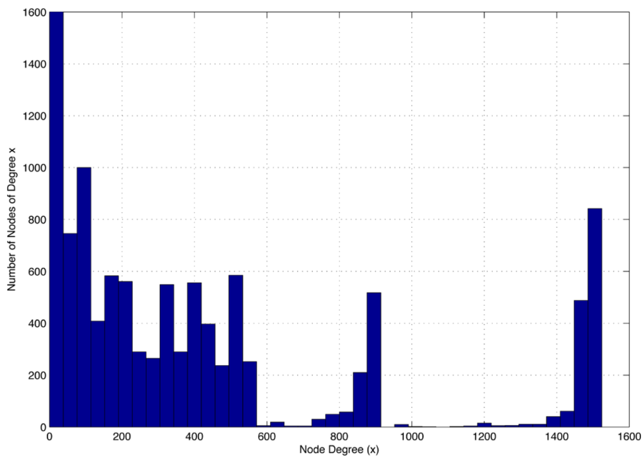
$$p(k) \propto k^{-\gamma} \tag{16}$$

where k is the number of connections and γ is a constant. Such networks exhibit increasingly large 'hub' nodes as the total network size increases, rather akin to the hubs used by airlines in their routing. The highly connected hubs make the graph diameter low, i.e., the shortest path between any two nodes is low. So to go from Bathurst in regional Australia to the small community of Santa Fe in New Mexico, requires just Sydney, Los Angeles and Albuquerque as intermediate nodes.

Small world networks, on the other hand, have low graph diameters, but do not have the large hubs. The low diameter is achieved through short cuts between distant nodes.

We considered the number of links each node has to other nodes for two different threshold values of the mutual information between nodes 0.25 and 0.35. The resultant histogram of node connectivity for the 0.35 threshold is shown in Figure 10. The first bin of the histogram has been truncated at 1600. The distribution of node connectivity does not follow the scale-free distribution often associated with social networks. This is most likely because of the very particular nature of the networks between MWAs and their clients.

Figure 10 The histogram of node connectivity for the network with a threshold of 0.35. Only approximately 16,000 nodes have non-zero connectivity at this threshold (see online version for colours)



6 Discussion and conclusions

This paper described an ABM for studying trust in financial adviser-client relationships and focused on the extraction of client parameters from data warehouse records of client activity.

Fuzzy logic was used to model client behaviour. A multi-objective EA was found to significantly improve the performance and coverage of the FIS trained on a large data set over that was obtained by the FPA of FISPRO. The original data set was downsized by a factor of 100 to yield a training dataset of 285,660 records, but we have some doubts as to whether sufficient of the customer behaviour is captured in the resulting evolved FIS. Future work will resolve the technical issues needed to scale the algorithms up to using a large fraction of the full dataset, and to analyse the resulting FIS.

The data warehouse did not contain any information about client networks and thus a new inferential approach was explored. The mutual information in the investment time series between clients was calculated and client networks were inferred for different threshold values.

The threshold approach used here exploits the characteristics of the histogram. But, there are other approaches that could be investigated. Tumminello et al. (2005) examine the topology of graphs for stock-market data. They find that much of the edge weight of their stock-market data can be captured by a *genus* 0 subgraph – i.e., the graph that captures as many edges as possible whilst still planar (i.e., can be drawn in the plane with no edges crossing over). The significance of topological simplicity is not fully understood, but this seems worthy of further investigation for our data.

The mutual information may be influenced by common external factors influencing client investment patterns. A computationally more demanding measure advocated by Schreiber (2000) is the *transfer entropy,* in which the joint probability in the mutual information is replaced by a conditional probability. Further work will study the networks based on transfer entropy.

Acknowledgements

This work was supported by grant LP0453657 from the Australian Research Council and a grant of computer time from the Australian Centre for Advanced Computing and Communications.

References

Abbass, H. (2006) 'Pareto-optimal approaches to neuroensemble learning', in Jin, Y. (Ed.): *Multi-Objective Machine Learning*, Vol. 16 of *Studies in Computational Intelligence*, Chapter 18, Springer, Berlin, pp.407–427.

Abraham, A. (2002) 'EvoNF: a framework for optimization of fuzzy inference systems using neural network learning and evolutionary computation', *Proceedings of the 2002 IEEE International Symposium on Intelligent Control*, arXiv:cs/0405032, The Coast Plaza Suite Hotel, Vancouver, British Columbia, Canada, pp.327–332.

Alander, J.T. (1997) 'An indexed bibliography of genetic algorithms with fuzzy logic', in Pedrycz, W. (Ed.): *Fuzzy Evolutionary Computation*, Kluwer Academic, Boston, pp.299–318.

Barabási, A-L. (2002) *Linked: The New Science of Networks*, Perseus, Cambridge, Massachusetts.

Bossomaier, T., Jarratt, D., Anver, M., Thompson, J. and Cooper, J. (2005) 'Optimisation of client trust by evolutionary learning of financial planning strategies in an agent based model', *Proc., IEEE Conf. on Evolutionary Computing*, University of Edinburgh, Edinburgh, United Kingdom, pp.856–863.

Cellucci, C., Albano, A. and Rapp, P. (2005) 'Statistical validation of mutual information calculations: comparison of alternative numerical algorithms', *Physical Review E*, Vol. 71, No. 6, p.71.

Cordón, O. and Herrera, F. (1999) 'A two-stage evolutionary process for designing task fuzzy rule-based systems', *IEEE Transactions on Systems, Man, and Cybernetics – Part B: Cybernetics*, Vol. 29, pp.703–715.

Cordón, O., Gomide, F., Herrera, F., Homann, F. and Magdalena, L. (2004) 'Ten years of genetic fuzzy systems: current framework and new trends', *Fuzzy Sets and Systems*, Vol. 141, pp.5–31.

Cuthill, E. and McKee, J. (1969) 'Reducing the bandwidth of sparse symmetric matrices', *Proceedings of the 24th National Conference of the ACM*, New York, USA, pp.157–172.

FISPRO (2008) URL, http://www.inra.fr/internet/Departments/MIA/M//fispro/indexen.html

Fulcher, J. (2008) 'Computational intelligence: an introduction', *Studies in Computational Intelligence*, Vol. 115, p.1.

Glorennec, P-Y. (1996) 'Quelques aspects analytiques des systémes d'inférence floue', *Journal Européen des Systèmes Automatisés*, Vol. 30, pp.231–254.

OpenMP (2002) *OpenMP C and C++ Application Program Interface*, OpenMP Architecture Review Board, Version 2.0, http://www.openmp.org/mp-documents/cspec20.pdf

Schreiber, T. (2000) 'Measuring information transfer', *Phys. Rev. Lett.*, Vol. 85, pp.461–464.

Standish, R.K. and Leow, R. (2003) 'EcoLab: agent based modeling for C++ programmers', *Proceedings SwarmFest 2003*, arXiv:cs.MA/0401026, University of Notre Dame, Notre Dame, Indiana, USA, pp.1–5.

Tumminello, M., Aste, T., Di Matteo, T. and Mategna, R. (2005) 'A tool for filtering information in complex systems', *Proc. Natl. Acad. Sci.*, Vol. 102, pp.10421–10426.

Watts, D. (1999) *Small Worlds*, Princeton University Press, Princeton, New Jersey, USA.

Yahoo (2008) *Yahoo Finance*, URL, http://www.yahoo.com.au

Notes

[1]To make the simulation computationally tractable, portfolios are relatively small and each share may be thought of as more an asset class than an individual equity.

[2]http://ecolab.sourceforge.net