18th World IMACS / MODSIM Congress, Cairns, Australia 13-17 July 2009
http://mssanz.org.au/modsim09

Tuning a modified Mamdani fuzzy rule base system with a genetic algorithm for travel decisions

Ricketts, J.H.[1]

[1]*CSIRO Marine and Atmospheric Research, Aspendale, Victoria, Australia; and Climate Change Research Group, Centre for Australian Weather and Climate Research, a partnership between CSIRO and the Bureau of Meteorology*
Email: jim.ricketts@csiro.au

Abstract: This paper presents work extending the modelling of behavioural rules for activity scheduling in the transport domain as previously published (Olaru and Smith 2005), by using genetic algorithms (GAs) to tune a modified Mamdani fuzzy knowledge base.

A Mamdani fuzzy knowledge base system is a fuzzy logic rule based system (FRBS) initially proposed by Mamdani in 1974 as a fuzzy logic controller. One uses a combination of fuzzification, fuzzy inference and defuzzification together with a knowledge base comprising database of fuzzy sets and a rule-base of fuzzy rules. This version substitutes fuzzy selection for defuzzification and, after training, outputs predicted travel schedule decisions given a coding of an individual's situation. The tuning of such a system is an open problem.

Two genetic algorithms were applied. One, the rule-base GA (rb-GA), taking the database as fixed, attempts to firstly maximize the classification rate and secondarily minimize the size of the rule base. The other, the fuzzy set GA (fs-GA) attempts to pre-tune the partitioning of the fuzzy sets for the rb-GA using an information entropy-like measure as a heuristic. The fs-GA is the focus of this paper, hence while the rb-GA is necessarily discussed, it is presented only briefly, and is the subject of a separate study.

The FRBS tuning problem has a high dimension error space with dimensions relating to the following partial order of choices; (a) the selection of axes, (b) the number and form of fuzzy-sets on the axes, these points relating to the database; (c) the grammar of the rules, and (d) the selection of the rules in the rule base. The high dimensionality of the problem requires design time constraint to be exercised.

Whilst a FRBS derived by the rb-GA (without the fs-GA) is more compact and better performed than the set derived by a human expert and published by Olaru and Smith, it improves further if performed after pre-tuning with the fs-GA. The entropy like measure served as a useful heuristic objective function.

Keywords: *Mamdani, Fuzzy Rule Base System (FRBS), Genetic Algorithm (GA), Trip Scheduling, Information Entropy*

1. INTRODUCTION

People are required to make trip-scheduling decisions in their daily lives. They deal with a rich set of uncertainties – changes of venue, priority, and time of activities – choices in mode, route, and time of trips – information horizons – delays and changes in current activities, and in trips. Populations both evolve and contain, a rich set of applicable strategies.

Rule based systems can be used both to explain and to predict responses of a population (or individuals) to perturbations. They can be constructed by human experts (either using examples sets, or interviews of subjects and domain experts) or by machine learning techniques from example sets. Rule based systems in turn use either logical systems or ad-hoc operations and learning systems construct them using many different heuristics. The machine learning approach is useful where it is desired to model group behaviour and/or to extract rules from the analysis that we can relate to plausible human behaviour. Fuzzy logic is useful for modelling inference under imprecision, and fuzzy logic can be used in a rule-based system. One system is a Mamdani Fuzzy Rule Based System (FRBS) with a database and a rule-base, and these have been previously adapted to classification problems although mostly using the Michigan approach (Cordon et al. 2004).

Genetic algorithms (GAs) form a well documented family of methods, useful amongst other things for optimisation (Goldberg 1989; Beasley et al. 1993). The essential feature of a GA is that a population of proposed solutions (coded using a "chromosome") is modified using biologically inspired operators (especially crossover and mutation), and incorporating a random component, to explore a solution space. GAs are useful for tuning both the database and the rule-base of FRBS (Maniadakis and Surmann 1999; Cordón et al. 2001). The subject of this paper is the impact of a GA (fs-GA) for tuning the database in a Mamdani FRBS on the outcomes of a GA which operates on the rule-base (rb-GA).

The rest of this paper first introduces a specific trip scheduling problem which has previously been analysed with a Mamdani FRBS. It then introduces both the standard Mamdani FRBS and the modifications made for this case including the structure of the rules and the database. It very briefly summarises the rb-GA (paper in preparation) since this is the test bed for the fs-GA. The fs-GA is described in greater detail, then follow the experiment and results, a discussion and conclusions.

2. THE TRIP SCHEDULING PROBLEM

Olaru and Smith (2005) published a study based on transport decisions in hypothetical situations made by 126 respondents in a follow-up survey after a larger survey of students and academics in Bucharest November 1998. They showed that a Mamdani FRBS allowed them to "examine a broader and richer field than traditional methods" and "represents the real system in a form close to human perception". Their system had 56 rules and achieved 82-87% successful classification rates. This problem has been further analysed, concentrating on the rule sets, the formal grammar of the fuzzy rules, and the composition rules. The problem is now cast as a learning/classification challenge, where the inputs are coded situations, presented as real numbered tuples, and outputs of the FRBS are discrete values representing predicted responses of individuals to their situation.

3. MAMDANI FRBS

A Mamdani fuzzy knowledge base system is a fuzzy logic rule based system (FRBS) initially proposed by Mamdani in 1974 as a controller. One uses a combination of fuzzification, fuzzy inference and defuzzification together with a knowledge base comprising a database of fuzzy membership functions implemented as fuzzy-sets and a rule-base of fuzzy rules. Input is mapped onto a set of fuzzy partitions (overlapped fuzzy-sets) from the database by the Fuzzification interface. These are mapped by the rule set in the rule base onto a single fuzzy partition using the fuzzy inference methods defined in the inference system. This fuzzy partition is fed into the de-fuzzification system which produces real output.

3.1. The Modified Mamdani FRBS

This version of the Mamdani FRBS substitutes selection for defuzzification, since the application is not a control one but a classification one with disjoint classes. The modified FRBS is fed tuples of real values. The Selection interface takes the fuzzy partition from the Inference system, and outputs a discrete value, as illustrated in Figure 1.

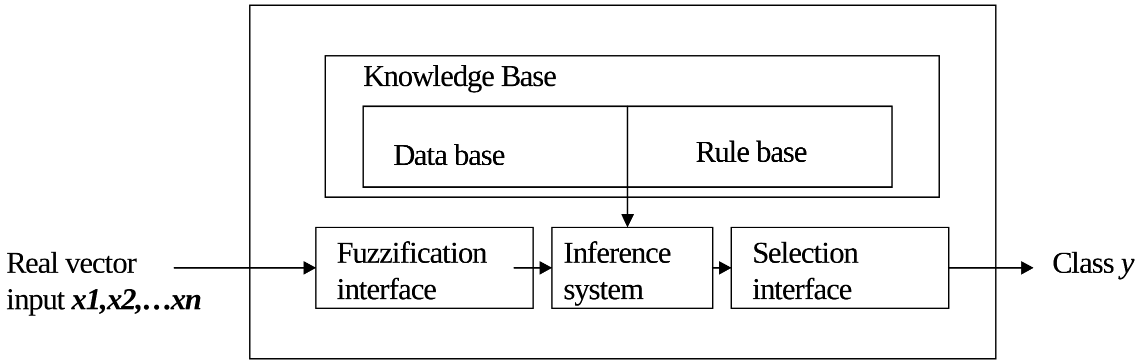


Figure 1. The modified Mamdani FRBS with Selection replacing de-fuzzification. (After Cordón, Herrera et al. 2001)

As per Olaru and Smith (2005), the problem is expressed in terms of four fuzzy partitions representing conditions or aspects of a change in schedule; schedule flexibility, time saving (or loss), time of day, and duration of next activity. Each fuzzy set in the partition represents a different linguistic adjective. To this there is associated a set of possible decisions or actions taken by travellers.

By distinction with Olaru and Smith (2005) the overall grammar of the rule set and inference is disjunctive normal form (DNF) using monotone monomials. Monotone means that there is no negation and monomials are conjunctions of literals. E.g. “Time of day is not noon” is not monotone, and “Time of day is noon or afternoon” is not monomial. Within rules fuzzy conjunction is implemented as a min, and disjunction as a max. Each rule consists of an implication with monotone monomials in the antecedent, and a single literal consequent. E.g. “if flexibility=flexible ^ `time_saving=small` ^ `time_of_day= night` ^ `duration_next_activity=short` then `action=remove`”. I.e. if the schedule is flexible and the time saving is small and the time of day is night and the duration of the next activity is short, then remove the next item from the schedule. The list of antecedents and consequents is shown in Figure 2.

In a Mamdani FRBS, all rules are fired during inferencing, and their output is aggregated or composed using any of a variety of operators, often ones called “and also”. The composition operator here used is the single-best operator used by Olaru and Smith.

```
Antecedent conditions and their adjectives

•  frb_flexibility = (frb_f_none, frb_f_rigid, frb_f_not_rigid, frb_f_flexible);
•  frb_time_saving = (frb_ts_none, frb_ts_neg_large, frb_ts_neg_small, frb_ts_small,
   frb_ts_medium, frb_ts_large);
•  frb_time_of_day = (frb_tod_none, frb_tod_night, frb_tod_early_am, frb_tod_am,
   frb_tod_noon, frb_tod_early_pm, frb_tod_later_pm, frb_tod_evening);
•  frb_duration_next_activity = (frb_dna_none, frb_dna_short, frb_dna_medium,
   frb_dna_long);

Consequents
•  frb_action = (frb_a_none, frb_a_do_nothing, frb_a_change_time,
   frb_a_change_duration, frb_a_location, frb_a_remove, frb_a_new,
   frb_a_change_transport);
```

Figure 2. The four antecedent conditions, each of which is represented by a partition of fuzzy sets. Each of the fuzzy sets except the **_none* represents the degree of membership applicable to one “linguistic adjective”. The consequents are by contrast not a linear partition, but disjoint fuzzy sets.

4. THE RULES GA

This GA uses a population of 100 variable length chromosomes. The system is a Pittsburgh GA (Cordón et al. 2001; Cordon et al. 2004) with a population consisting of individuals encoding rule sets. The chromosome is translated into a rule set of a Mamdani FRBS, *Fitness* is the proportion of test cases correctly classified by a FRBS using that rule set with an additional weighting factor preferring shorter chromosome lengths. *Chromosome* Coding the consequent conditions by position and thus storing only their adjectives, there are only (768) different possible antecedent rule sets each with a single consequent (one of 8). Each gene is a pair of integers; the antecedent part and the consequent part. Each individual chromosome consists of a variable length string of up to 100 genes. *Crossover* is asymmetric single point where a random crossover point is computed for each parent separately and two new strings are composed from the head of one and the tail of the other after duplicate antecedents are deleted. Thus the two offspring are generally of different lengths. A number of mutation operators are defined. No mutator is allowed to create a chromosome that has duplicate antecedent parts. *Point Mutation*: A randomly selected locus in the antecedent part *or* the consequent of one gene is incremented or decremented so that it refers to one of the overlapping, neighbouring fuzzy-sets. *Delete*: Delete a sequence of up to a quarter of genes. *Extend:* extend the chromosome with a random gene. *Inversion*: a strategy for regrouping genes so that crossover has a chance to test different building blocks. *Breeding* consists of selection of pairs from the population, and reproduction using crossover and mutation producing new pairs of offspring. *Selection* is *elitist*, using a form of ranking selection. Individuals are first ranked by fitness and then working from fittest to least fit, individuals mate with selected members of the old population, offspring replacing least fit individuals. The majority of the population are subjected to a bout of mutation using a random selection of mutation operators. One feature of ranking selection is that all objective functions which are monotonic on each other are equivalent in effect.

5. THE FUZZY-SET GA

5.1. Genome and operators

This was implemented as an ordering GA (Goldberg 1989) by contrast with the rb-GA. The reasons are (a) there is a fixed number of objects to be constructed (the partitions), (b) it is simple to encode all required constructs and avoid redundancy, (c) there is no need to eliminate duplication or repair illegal codings. By contrast the rb-GA requires code in the mutators and crossover to prevent illegality.

Chromosome

The chromosome is a fixed length string of 200 bytes. Each value represents an operator which selects a partition and then applies one of two operations. There are two families of operators, *set-value* and *extend*, and eight different operator types in total. The first operator type sets the value of one of the four corners of the trapezoid to be the current axis value, and the second operator type increments the axis value. There are three *set -value* operators and four increment operators per fuzzy set in each partition.

The sets are designed to be normalised. Hence setting the third point for a trapezoid signals the start of the next one and setting the fourth point of one also defines the second point of the next one.

Advantages of this coding are that it is quick and does not produce illegal chromosomes. Disadvantages are mainly that it is crude and coarse grained.

Decoding and maturation

The goal of the maturation is the construction of a set of four partitions. The input to the construction routine is parsed one gene at a time. First the partition of interest is identified then the operator is executed.

Breeding

Breeding is via only two operators, mutation and crossover. Selection policy is the same as the previous GA. Elites are retained, and the same ranking and replacement schemes are used. The crossover operator used is the single locus order-based crossover similar to OX (Goldberg 1989), but with only a single crossover point. The only mutation operator simply exchanges the values at two randomly selected points on a chromosome.

5.2. Fitness and objective function

The aim of the fitness function is to assess how well the partitioning of the fuzzy sets also partitions the data. There are four axes each with a number of partitions which thus forms a partitioned hyper-space with multiple “hyper-regions”. Intuitively one would like all of the classified situations with a particular class to fall within one hyper-region. In this case just eight hyper-regions would be filled, each with a single class and just eight rules would suffice to classify the data fully. For each class represented within a hyper-region, a *specificity* and a *selectivity* of that region for that class can be assigned. *Specificity* of a situation S in region R and having class C is the probability of S being of class C given having non-zero membership of the region R.. *Selectivity* of a situation S in region R and having class C is the probability of S having non-zero membership of region R given class C. S may have a non-zero degree of membership within more than one region, so to compute fitness we first compute the specificity of S multiplied by log of selectivity of S within each of the regions for which it has a degree of membership, take the maximum of absolute value of all these as the contribution of the situation to the fitness function and sum over all situations.

$$Specificity_S^R = \Pr(C_S \mid R_S) \tag{1}$$

$$Selectivity_S^R = \Pr(R_S \mid C_S) \tag{2}$$

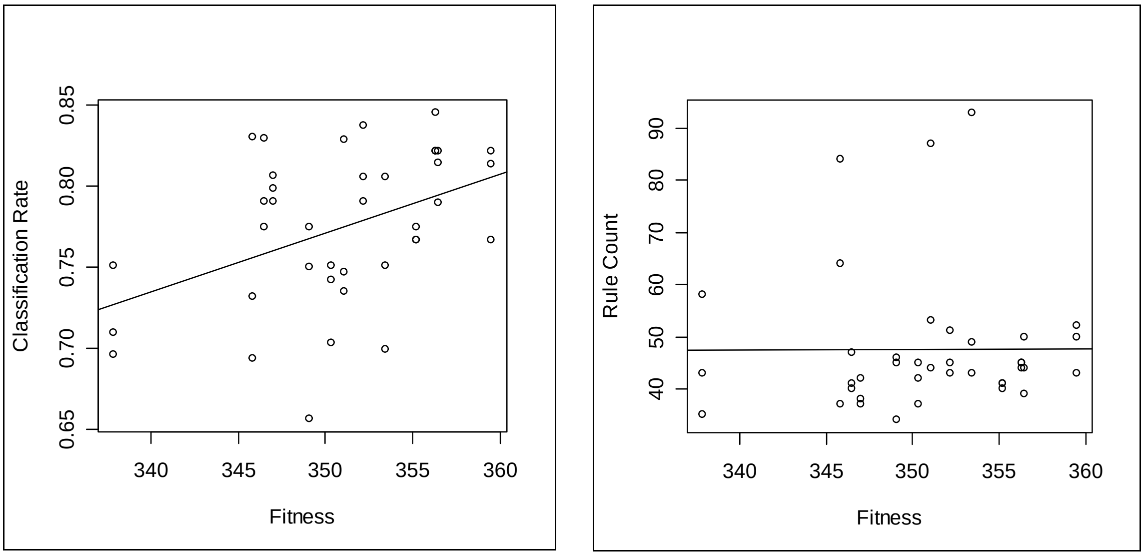
$$F = \sum_{S=1}^{NumberSituations} \max_R \left(\mathrm{abs}\left(Specificity_S^R \cdot \log\left(Selectivity_S^R\right)\right)\right) \tag{3}$$

6. EXPERIMENTS AND RESULTS

The fs-GA was run 13 times for 100 generations. The elite from each run was retained and the value of its associated objective function was recorded. This produced a spectrum of fuzzy sets, each with a different value. This constituted a test set of fuzzy-sets.

Using each of the obtained fuzzy sets in turn, the rb-GA was then run against the full travel decisions data set. The rb-GA ran with single-best composition, and was halted at 10,000 generations. The process was repeated in triplicate to give 39 results, each yielding a collection of rules sets, each of these having a classification rate, a rule count, and an associated value for the objective function assigned to the fuzzy-set by the fs-GA.

Pooling the three runs, and performing a linear regression of the fuzzy-set fitness v classification rate and rule counts obtained, we find that there is a significant positive trend for the classification rates and no trend for the rule counts as seen in Figure 3.



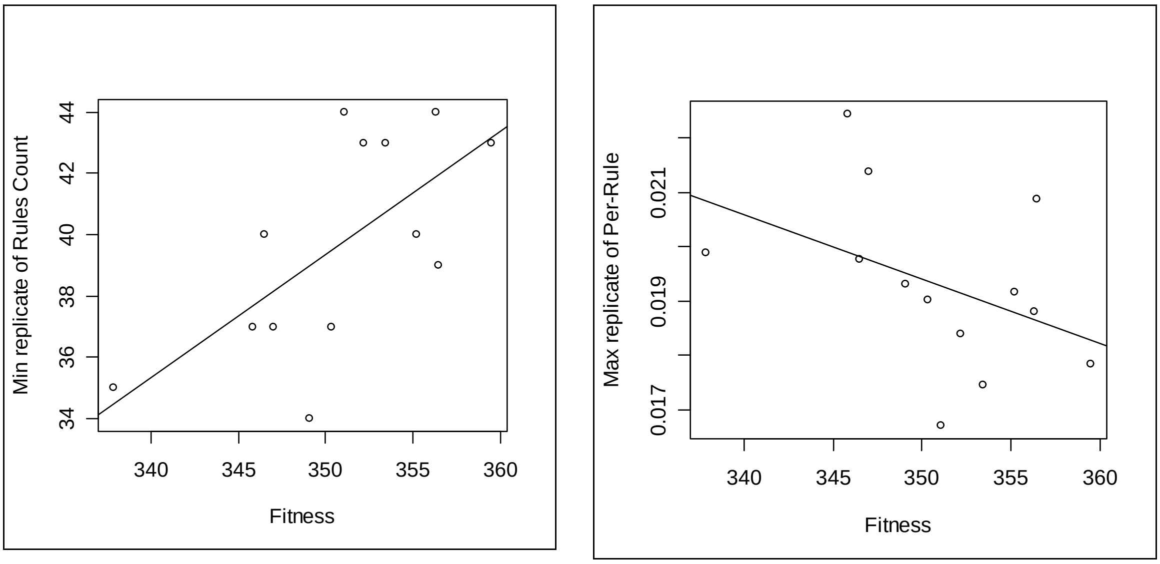

```
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.504031   0.438784  -1.149  0.25805
Fitness      0.003644   0.001251   2.914  0.00602*
Residual standard error: 0.04302 df 37
Multiple R-Squared: 0.1867
```

```
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  41.90556  137.29823   0.305    0.762
Fitness       0.01627    0.39130   0.042    0.967
Residual standard error: 13.46 df 37
Multiple R-Squared: 4.675e-05
```

Figure 3. Regressions of Classification Rate and of Rule Count against the fs-GA fitness achieved by each of thirteen independent runs of the fs-GA. The classification rate shows a significant trend, whereas the rule count does not.

There is considerable variance. Furthermore fitness and rule counts are not independent. The rb-GA optimises for high classification rates and only reduces rule counts within classification rate. That, plus the fact that the rb-GA is terminated after a fixed number of generations, means that there is less significance to values obtained for rule counts, than for either the minimum rule counts obtained for each fitness level or the maximum ratio of classification rate to rule count. This latter measure is important since higher classification rate per rule is a measure of the generality of the rule-set.

Therefore these two measures (smallest replicate of rule count, and maximum classification rate per rule) were tested as well. As shown in Figure 4, there is a non-significant positive trend between fs-GA objective function value and the smallest per-replicate size of the rule set obtained, indicating that the GA may have proceeded faster (since all runs of the rb-GA halted after the same number of generations). There is a more significant negative trend with the maximum per-replicate ratio of classification rate to rule count, indicating that less compact rule sets obtained running with the higher values of the fs-GA fitness, are likely less generalized.




```
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  6.095e-02  2.675e-02   2.278   0.0437*
Fitness     -1.187e-04  7.625e-05  -1.557   0.1478
Residual standard error: 0.001514 df 11
Multiple R-Squared: 0.1805
```

```
               Estimate Std. Error t value Pr(>|t|)
(Intercept) -101.2506    48.5548  -2.085   0.0611
Fitness        0.4017     0.1384   2.903   0.0144*
Residual standard error: 2.749 df 11
Multiple R-Squared: 0.4338
```

Figure 4. The second analysis tests fuzzy-set fitness against the minimum rule count, and maximum ratio of classification rate to rule count, for the replicates at each fitness level.

Over the 39 repetitions of the fs-GA, the median rule count obtained by the rb-GA was 44, the mean was 47.6, ranging from 34 to 93, and the mode was 45. The median and the mean classification rate was 0.77 (77%) ranging from 0.66 to 0.85, but the modal value was 0.82. In terms of parsimony as given by the ratio of the classification rate to rule count, the rule set with 93 rules gave a classification rate of 0.70 (70%) and the most parsimonious had 37 rules for a classification rate of 0.83 (83%).

7. DISCUSSION AND CONCLUSIONS

The entropy like measure served as a useful heuristic objective function for the purpose of pre-tuning the database of the FRBS in this problem. There is a positive relationship between the fitness of the fuzzy-sets and the classification success of the rb-GA. There is a suggestion that the rule counts increase with fitness, and there is evidence that classification rate per rule decreases.

The FRBS tuning problem has a high dimension error space with dimensions relating to the following partial order of choices; (a) the selection of axes, (b) the number and form of fuzzy-sets on the axes, these points relating to the database; (c) the grammar of the rules, and (d) the selection of the rules in the rule base. The high dimensionality of the problem requires design time constraint to be exercised. Both the experience of Olaru and Smith with their hand coding (personal communication, and (Olaru and Smith 2005)), and the results of the original experiments with the rb-GA show that the partitioning of the fuzzy sets was both critical and difficult.

The database can be tuned prior to the tuning of the rule-base, in parallel with, or after it. The decision to pre-tune relates to the specific case, where both the axes of interest and the partition count were determined by the requirements of the survey, but the exact partitioning was difficult. Pre-tuning gives two further choices. One can use a heuristic objective function, or one can use an objective function based on an approximate rule-base. This paper shows some benefits to pre-tuning using a simple heuristic.

It has been well established that GAs in general execute a more robust search than pure hill-climbing. I now consider the choice of heuristic for the fs-GA. The hypothesis was that finding natural partitions in the data-set would reduce the optimal rule-count by reducing the number of rules dealing with exceptions. Secondarily it was anticipated that it would increase progress by flattening the error topography, essentially allowing the GA to proceed up valleys. The heuristic measure used is monotonic on the entropy of the data-set, and is a measure of the ability of the partitioning to classify the data-set with only one rule per classification.

The decision to change the syntax from Olaru and Smith, and to restrict atomic rules to monotone monomials simplifies the error landscape. The decoding scheme in the fs-GA limits the shapes of the fuzzy sets and again simplifies the error landscape.

The primary experiment with the rb-GA shows several important things. The GA was halted after a set number of runs, and there is a high level of variance of all performance measures. The performance measures of the fs-GA, which are the same measures, thus are subject to high variance. One can interpret higher variance in the rb-GA to be a measure of roughness of topography, but have I not attempted to partition the variance between the fs-GA and rb-GA. The effectiveness of the rb-GA is demonstrated by the following metrics. The modal rule count obtained from the rb-GA after tuning was 45 compared with the 56 rules obtained by Olaru and Smith (2005) from their hand tuned sets. The modal classification rate was 83%, comparable with the range of 82-87% published. The most parsimonious rule set contained only 37 rules in DNF with a classification rate of 83%.

Another study with a full analysis of the rb-GA has been done and the results will submitted under the title "GA Optimized FRBS for Travel Decisions" by Ricketts, J.H. and D. Olaru.

ACKNOWLEDGEMENTS

Doina Olaru provided the initial problem, the data and encouragement. Comments and questions during the review process also were useful.

REFERENCES

Beasley, D., D. R. Bull and R. R. Martin (1993). An overview of genetic algorithms: Part 1, Fundamentals. *University Computing* 15(2): 58-69.

Cordon, O., F. Herrera, F. Gomide, F. Hoffmann and L. Magdalena (2004). Ten years of genetic fuzzy systems: current framework and new trends. *Fuzzy sets and systems* 141(2004): 5–31.

Cordón, O., F. Herrera, F. Hoffmann and L. Magdalena (2001). Genetic fuzzy systems : evolutionary tuning and learning of fuzzy knowledge bases. Singapore, World Scientific.

Goldberg, D. E. (1989). Genetic algorithms in search, optimization, and machine learning, Addison-Wesley Publishing Company.

Olaru, D. and B. Smith (2005). Modelling behavioural rules for daily activity scheduling using fuzzy logic. *Transportation* 32(4): 423-441.