DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO

Relatório Técnico

RT-MAC-9606

REAL: Real Attribute Learning for Strategic Market Operation

Julio M. Stern
Fabio Nakano
Marcelo Lauretto

Agosto 96

REAL: Real Attribute Learning for Strategic Market Operation

Julio M. Stern * Fábio Nakano
Marcelo Lauretto
Computer Science Department, University of São Paulo
Technical Report RT-MAC-9606

August 22, 1996

Abstract

This paper presents REAL, a Real-Valued Attribute Classification Tree Learning Algorithm. The end users demands for a REAL based decision support tool to be used at the Brazilian stock market explain several of the algorithm's unique features. Compared to competing algorithms, in our applications, the REAL algorithm presents major advantages, like:
1- The REAL classification trees usually have smaller error rates.
2- A single "conviction" measure at each leave is more convenient than the traditional (probability, confidence-level) pair.
3- No need for an external pruning criterion.

*jstern@ime.usp.br The authors were supported by grants from CNPq, 30138/91-7, and FAPESP, 96/2341-2, and also by grants from the Industrial Engineering Department, and from The São Paulo Commodities and Futures Exchange, *BM&F*.

1 Introduction

This paper presents the Machine Learning REAL algorithm, for automatic construction of classification trees with real-valued attributes [Breiman-93], [Quinlan-86], [Unger-81]. The REAL project started as an application to be used at the Brazilian stock market, looking for a good algorithm for predicting the adequacy of operation strategies. In this context the success or failure of a given operation strategy corresponds to different classes, and the attributes are real-valued technical indicators (Real is also the name of the new Brazilian currency). The users demands for a REAL based decision support tool also explain several of the algorithm's unique features.

The project began testing several Machine Learning tools presented at the ESPRIT-StatLog project [Michie-94]; The TDIDT - Top Down Induction Decision Tree - software CAL5 [Mueller-94] proved to be specially useful for the application we had in mind. The CAL5 algorithm had a strong influence on our project, and we use it as our main performance benchmark. In our applications the REAL algorithms presented some major advantages:

1. Usually the classification trees have a smaller error rate.
2. The single "conviction" measure at each leave is more convenient, than the traditional (probability, confidence-level) pair.
3. REAL interval partition procedure stops naturally, eliminating the need of an additional pruning procedure.

2 Problem Formulation

Our examples for the classification problem are given as a $n \times (m+1)$ matrix A. Each row, $A(i,:)$, represents a different example, and each column, $A(:,j)$, a different attribute. The first m columns in each row are real-valued attributes, and the last column , $A(i,m+1)$ is the example's class. Part of this samples, the training set, are used by the algorithm to generate a classification tree, which is then tested with the remaining examples. The error rate in the classification of the examples in the test set is the simplest evaluation of the classification tree.

3 Tree Construction

Each main iteration of the REAL algorithm corresponds to the branching of a terminal node in the tree. The examples at that node are classified according to the value of a selected attribute, each branch corresponding to a specific interval. The partition of a real-valued attribute's domain in adjacent non-overlapping (sub) intervals is the discretization process. Each main iteration of REAL includes:

1. The discretization of each attribute, and its evaluation by a loss function.
2. Selecting the best attribute, and branching the node accordingly.
3. Merging adjacent intervals that fail to reach a minimum conviction threshold.

4 Conviction and Loss Functions

Given a node of class c with n examples, k of which misclassified and $(n-k)$ of which correctly classified, we needed a single scalar parameter, cm, to measure the probability of misclassification and its confidence level. Such a simplified "conviction" measure was a demand of REAL application users operating at the stock market.

Let q be the misclassification probability for an example at a given node, $p = (1-q)$ the probability of correct classification, and assume we have a Bayesian distribution for q ,

$$D(c) = Pr(q \leq c) = Pr(p \geq 1-c)$$

We define the conviction measure: $100 * (1-cm)\%$, where

$$cm = \min\ c \ \mid \ Pr(q \leq c) \geq 1 - g(c)$$

and $g(\,)$ is a monotonicaly increasing bijection of $[0,1]$ onto itself. From our experience in the stock market application we learned to be extra cautious about making strong statements, so we usually make $g(\,)$ a convex function.

In this paper $D(c)$ is the posterior distribution for a sample taken from the Bernoulli distribution, with a uniform prior for q:

$$\begin{aligned} B(n,k,q) &= comb(n,k) * q^k * p^{n-k} \\ D(c,n,k) &= \int_{q=0}^{c} B(n,k,q) \ / \ \int_{q=0}^{1} B(n,k,q) \\ &= \text{betainc}(c, k+1, n-k+1) \end{aligned}$$

Also in this paper, we focus our attention on

$$g(c) = g(c,r) = c^r, \quad r \geq 1.0$$

we call r the convexity parameter.

With these choices, the posterior is the easily computed incomplete beta function, and cm is the root of the monotonicaly decreasing function

$$\begin{aligned} cm(n,k,r) &= c \ \mid \ f(c) = 0 \\ f(c) &= 1 - g(c) - D(c,n,k) \\ &= 1 - c^r - \text{betainc}(c, k+1, n-k+1) \end{aligned}$$

On other papers (in preparation) the study the behavior of the REAL algorithm for other choices of $D(c,n,k)$ and $g(c)$.

Finally we want a loss function for the discretizations, based on the conviction measure. In this paper we use the overall sum of each example classification conviction, that is, the sum over all intervals of the interval's conviction measure times the number of examples in the interval.

$$loss = \sum_i n_i * cm_i$$

5 Discretization Procedure

The first step of the discretization procedure, for a selected attribute, is to order the examples in that node by the attribute's value, and then to join together the neighboring examples of the same class. So, at the end of this first step, we have the best ordered discretization for the selected attribute with uniform class clusters.

In the subsequent steps, we join intervals together, in order to decrease the overall loss function of the discretization. The gain of joining J adjacent intervals, $I_{h+1}, I_{h+2}, \ldots I_{h+J}$, is the relative decrease in the loss function

$$gain(h,j) = \sum_j loss(n_j, k_j, r) \quad - loss(n, k, r)$$

where $n = \sum_j n_j$ and k counts the minorities' examples in the new cluster (at the second step $k_j = 0$, because we begin with uniform class clusters).

At each step we join the clusters with maximum gain, but we can limit our search to a small neighborhood, i.e., $3 \leq J \leq jmax$. The discretization procedure stops when there are no more clusters with positive gain. This strict greedy 1-best approach can be modified to use, at each step, a b-best clusters to be joined list. Doing so will expedite the algorithm without any noticeable consistent degradation.

The next examples show some clusters that would be joined together at the first step of the discretization procedure. The notation $(n, k, m, r, \pm)$ means the we have two uniform clusters of the same class, of size n and m, separated by a uniform cluster of size k of a different class; r is the convexity parameter, and + (−) means we would (not) join the clusters together.

(2,1, 2,2,+)			
(6,2, 7,2,-)	(6,2, 8,2,+)	(6,2,23,2,+)	(6,2,24,2,-)
(7,2, 6,2,-)	(7,2, 7,2,+)	(7,2,42,2,+)	(7,2,43,2,-)
(23,3,23,2,-)	(23,3,43,2,-)	(23,3,44,2,+)	
(11,3,13,3,-)	(11,3,14,3,+)	(11,3,39,3,+)	(11,3,40,3,-)
(12,3,12,3,-)	(12,3,13,3,+)	(12,3,54,3,+)	(12,3,55,3,-)

In these examples we see that it takes extreme clusters of a balanced and large enough size, n and m, to "absorb" the noise or impurity in the middle cluster of size k. A larger convexity parameter, r, implies a larger loss at small clusters, and therefore makes it easier for sparse impurities to be absorbed.

6 Branching and Merging

For each terminal node in the tree, we perform the discretization procedure for each available attribute, measure the loss function of the final discretization, select the minimum loss attribute, and branch the node according to the selected attribute discretization. If no attribute discretization decreases the loss function by a threshold $\epsilon > 0$, no branching takes place.

A premature discretization by a parameter selected at a given level may preclude further improvement of the classification tree by the branching process. For this reason we establish a conviction threshold, ct, and after each branching step we merge all adjacent intervals that do not achieve $cm < ct$. To prevent an infinite loop, the loss function value assigned to the merged interval is the merging intervals' losses sum. At the final leaves, this merging is undone. The conviction threshold naturally stops the branching process, so there is no need for an external pruning procedure, like in most TDIDT algorithms.

7 Computer Implementation

REAL was implemented as a portable code in *C++*, and the final application uses a Microsoft VB-4.0 graphical user interface.

In a straightforward implementation, REAL spends most of the execution time computing the function $cm(n, k, r)$. We can greatly accelerate the algorithm by using precomputed tables of $cm(n, k, r)$ values for small n, and precomputed tables of $cm(n, k, r)$ polynomial interpolation coefficient for larger n. The most straightforward implementation of REAL, used for the numerical tests in the next section, takes about 5 minutes to run each problem, training and testing, on a Pentium-100 IBM-PC.

8 Market Operation Strategies

A market operation strategy is a predefined set of rules determining an operator's actions in the market; the strategy shall also have a predefined criterion for classifying a strategy application as success or failure. As an example, let us define the strategy $buysell(t, d, l, u, c)$:

- At time t buy a given asset A, at its price $p(t)$.
- Sell A as soon as:
  1. $t' = t + d$, or
  2. $p(t') = p(t) * (1 + u/100)$, or
  3. $p(t') = p(t) * (1 - l/100)$.
- The strategy application is successful if $c \leq 100 * p(t')/(p(t) \leq u$

The parameters u, l, c and d can be interpreted as the desired and worst accepted returns (low and upper bound), the strategy application cost, and a time limit. Figure 1 displays $buysell(t, d, l, u, c)$ applications.

8.1 Technical Indicators

Market analysts use several tools to forecast asset prices: some are based on accounting records, input-output or macro-economic analysis, etc; these are known as fundamentalist analysis tools. Other frequently used tools are "technical indicators". A technical indicator is function of one or more observed market variables. Several providers distribute daily and "intraday" data for all the important stock, commodities and

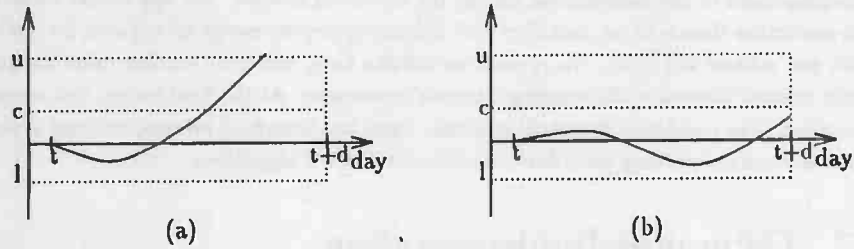

Figure 1: *buysell*(t, d, l, u, c) strategy application examples.

derivative markets. At BOVESPA, the São Paulo Stock Exchange, available daily data include, $H(t)$, $L(t)$, $O(t)$, $C(t)$, $M(t)$ and $V(t)$, respectively, the asset highest, lowest, opening, closing and mean price, and the asset transaction volume at day t. An example of technical indicator is an asset opening price relative the its highest price in the previous r days:

$$OH_r(t) = O(t)/\max\{H(t), H(t-1), \ldots H(t-r)\}$$

Technical indicators are traditional and widely used market analysis tools [Colby-88] (In this paper we are not concerned with the comparative value or optimality of the classical technical indicators as statistical models for time series analysis). The familiarity of market operators with technical indicators explains why they feel confident with decision support sistems providing logical decision rules based on them. This attitude translates in a more expedite and effective use of the system, and was the main motivation for our client requirement for a TDIDT trading decision support system for strategy classification and selection based on technical indicators attributes.

The core of the trading decision support system is a TDID strategy classification procedure based on the REAL algorithm. Let us consider *buysell*(t, d, l, u, c) strategy classification examples. The attributes of example $Ex(t)$ are the technical indicators values at time $t-1$, and $Ex(t)$ class is the strategy application at time t success or failure based on the information available at time $t+d$. The examples are taken from non overlapping segments of historical time series.

8.2 Objective Functions

A classification procedure, applied to a given test set of examples, give us a classification (or confusion) matrix like table 1, where $n11$ and $n22$ are the number of correctly classified successful and failed applications of the strategy, $n12$ are the successful applications misclassified as failures, and $n21$ the opposite misclassifications. A market operator expects from the decision support system advice to help him detect almost all good opportunities to apply the strategy, and rarely fail when he does so. So our

Actual/Attributed	Success	Failure
Success	$n11$	$n12$
Failure	$n21$	$n22$

Table 1: Classification (confusion) matrix

objectives are to, respectively, maximize and minimize the yield and application failure rates, ry and rf:

$$ry = n11/(n11 + n12) \quad \text{and} \quad rf = n21/(n11 + n21)$$

To conciliate these conflicting multiple objectives we define a merit function that can be interpreted as a conservative estimate of the cumulative gain for $buysell(t, d, l, u, c)$ strategy:

$$merit = c * n11 - l * n21$$

9 Numerical Tests

We tested the classification procedures of the previous sections on $buysell(t, d, l, u, c)$, with $d = 5$ $days$, $l = 1\%$, $c = 1\%$ and $u = 3\%$, on non overlapping time series segments of two of the most liquid (traded) stocks at BOVESPA: Telebras-PN (TEL4), with aprox. 300 examples 45% successful, and Petrobras-PN (PET4), with aprox. 400 examples 40% successful. We divided our examples in 10 subsets, for a 10-fold cross validation procedure [Hjorth-94]. At each run the algorithm generated a tree using the data in a 9-subsets learning-set, and tested it in the remaining test-(sub)set. The procedure was repeated for the algorithm's parameters in a discrete grid:

REAL: $(r, cl) \in \{1.0, 1.5, \ldots, 4.0\} \times \{0.1, 0.15, 0.2, \ldots, 0.45\}$.

Cal5: $(S, \alpha) \in \{0.05, 0.10, \ldots, 0.90\} \times \{0.05, 0.10, \ldots, 0.90\}$.

NewID: $\phi \in \{0\%, 2\%, 4\%, \ldots, 28\%, 30\%\}$.

We also included in the benchmark NewID, a classic TDIDT algorithm, primarily developed for categorical classification, but also capable of handling real attributes. At tables 2 and 3 we show the parameters optimizing the merit function mean, with corresponding mean and standard deviation for rf and ry.

For a sensitivity analysis we show similar results in a neighborhood of the optimal parameters in tables 4 and 5.

10 More Numerical Tests

We also tested REAL and Cal5 using van Cutsem's "emergency voltage conditions" dataset, where Cal5 had the previous best published performance [Mueller-94], [Cutsem-

Algorithm	P^*	*merit*	rf	ry
REAL	$r = 3.5, ct = 0.4$	3.8 ± 1.9	0.22 ± 0.12	0.44 ± 0.12
Cal5	$S = 0.3, \alpha = 0.1$	3.3 ± 1.5	0.35 ± 0.09	0.63 ± 0.10
NewID	$\phi = 6\%$	2.9 ± 2.3	0.25 ± 0.09	0.45 ± 0.22

Table 2: Optimal parameters for TEL4

Algorithm	P^*	*merit*	rf	ry
REAL	$r = 2, ct = 0.30$	4.3 ± 2.3	0.24 ± 0.10	0.39 ± 0.16
Cal5	$S = 0.4, \alpha = 0.2$	3.8 ± 2.4	0.23 ± 0.15	0.35 ± 0.16
NewID	$\phi = 8\%$	2.9 ± 2.6	0.42 ± 0.05	0.28 ± 0.19

Table 3: Optimal parameters for PET4

ct / r	0.35	$ct^* = 0.40$	0.45
3.0	$merit = 3.5 \pm 2.1$ $rf = 0.18 \pm 0.15$ $ry = 0.38 \pm 0.15$	$merit = 3.5 \pm 3.0$ $rf = 0.26 \pm 0.18$ $ry = 0.47 \pm 0.16$	$merit = 3.7 \pm 2.5$ $rf = 0.30 \pm 0.14$ $ry = 0.51 \pm 0.15$
$r^* = 3.5$	$merit = 3.5 \pm 2.2$ $rf = 0.14 \pm 0.16$ $ry = 0.34 \pm 0.13$	$merit = 3.8 \pm 1.9$ $rf = 0.22 \pm 0.12$ $ry = 0.44 \pm 0.12$	$merit = 3.5 \pm 3.1$ $rf = 0.31 \pm 0.16$ $ry = 0.51 \pm 0.15$
4.0	$merit = 3.2 \pm 2.6$ $rf = 0.27 \pm 0.29$ $ry = 0.38 \pm 0.18$	$merit = 3.5 \pm 2.2$ $rf = 0.24 \pm 0.12$ $ry = 0.41 \pm 0.15$	$merit = 3.2 \pm 3.0$ $rf = 0.27 \pm 0.20$ $ry = 0.45 \pm 0.20$

Table 4: REAL - Sensitivity analysis for TEL4

91]. We optimized the parameters over the same grid of the last section.
Each generated tree is tested in two ways,

1. Eliminating the test examples that fall outside a node's interval as out-of-range, and the examples in leaves that do not achieve the targeted conviction or probability-confidence, and then counting the *Hits* and *Misses* (correct and incorrect classifications).
2. Extending the intervals to cover the attributes' domains, and then counting the *hits* and *misses*, even for the intervals not achieving the desired conviction or probability-confidence.

For a sensitivity analysis we show similar results in a neighborhood of the optimal parameters in tables 7 and 8.

r \ ct	0.25	$ct^* = 0.30$	0.35
1.5	$merit = 2.7 \pm 2.9$ $rf = 0.30 \pm 0.30$ $ry = 0.24 \pm 0.16$	$merit = 2.2 \pm 2.7$ $rf = 0.36 \pm 0.17$ $ry = 0.35 \pm 0.16$	$merit = 1.8 \pm 3.4$ $rf = 0.40 \pm 0.18$ $ry = 0.43 \pm 0.16$
$r^* = 2.0$	$merit = 3.0 \pm 2.7$ $rf = 0.35 \pm 0.30$ $ry = 0.23 \pm 0.18$	$merit = 4.3 \pm 2.3$ $rf = 0.24 \pm 0.10$ $ry = 0.39 \pm 0.16$	$merit = 3.1 \pm 2.5$ $rf = 0.36 \pm 0.09$ $ry = 0.45 \pm 0.13$
2.5	$merit = 0.1 \pm 0.3$ $rf = 0.94 \pm 0.17$ $ry = 0.02 \pm 0.06$	$merit = 3.2 \pm 3.2$ $rf = 0.36 \pm 0.35$ $ry = 0.26 \pm 0.19$	$merit = 3.4 \pm 3.1$ $rf = 0.26 \pm 0.19$ $ry = 0.32 \pm 0.14$

Table 5: REAL - Sensitivity analysis for PET4

Algorithm	P^*	*hit*	*miss*	*Hit*	*Miss*
REAL	$(r, ct) = (1.5, 0.20)$	241.7	8.3 ± 6.85	230.6 ± 13.5	4.1 ± 3.51
Cal5	$(S, \alpha) = (0.65, 0.15)$	240.6	9.4 ± 6.88	236.2 ± 9.62	6.9 ± 6.28

Table 6: Optimal parameters for van Cutsem's dataset

r \ ct	0.15	$ct^* = 0.02$	0.25
1.0	$hits = 240.5 \pm 6.67$	$hits = 239.0 \pm 9.53$	$hits = 236.4 \pm 10.9$
$r^* = 1.5$	$hits = 240.7 \pm 6.20$	$hits = 241.7 \pm 6.85$	$hits = 240.3 \pm 7.50$
2.0	$hits = 240.1 \pm 7.36$	$hits = 239.7 \pm 7.75$	$hits = 239.8 \pm 7.27$

Table 7: REAL - Sensitivity analysis for van Cutsem's dataset

S \ α	0.10	$\alpha^* = 0.15$	0.20
0.6	$hits = 237.8 \pm 7.66$	$hits = 238.3 \pm 7.35$	$hits = 239.2 \pm 7.16$
$S^* = 0.65$	$hits = 238.5 \pm 9.13$	$hits = 240.6 \pm 6.88$	$hits = 239.8 \pm 7.50$
0.7	$hits = 239.9 \pm 7.39$	$hits = 238.8 \pm 7.07$	$hits = 239.6 \pm 7.31$

Table 8: Cal5 - Sensitivity analysis for van Cutsem's dataset

11 Conclusions

In the project's application the end users wanted to "understand" each classification with "explanations" based on already familiar attributes. Classification trees provided

such a tool, also allowing the users to dismiss a classification if suspicious of an attribute in the classification tree path exhibiting anomalous behavior. The convenient "conviction" measure of each classification was greatly appreciated by the end users taking real time decisions.

REAL performed better then any other decision tree algorithm we had access to, and proved to be a useful tool for predicting the success of stock market operation strategies, based on real-valued attributes (technical indicators) computed from stocks price and transaction volume time series. We also conjecture REAL advantages over Cal5 to be greater for noisyer data, but that statement requires more extensive numerical testing.

Acknowledgments

We are grateful for the support received from DCC-IME-USP - the Computer Science Department of the Mathematics and Statistics Institute of the University of São Paulo, from CNPq - Conselho Nacional de Desenvolvimento Científico e Tecnológico, from FAPESP - Fundação de Amparo à Pesquisa do Estado de São Paulo, and from The São Paulo Commodities and Futures Exchange - *BM&F*. The software used to compute tecnical indicators was developed by Celma O. Ribeiro, from the Industrial Egineering Department of The Polytechnic School of USP.

We are also grateful to Junior Barrera, Flávio S. C. da Silva, Jacob Zimbarg Sobrinho and Carlos A. B. Pereira, from IME-USP, for many useful insights, to Wolfgang Mueller, from the Fraunhofer Institut, for a CAL5 SUN-Sparc executable code, and to Gerd Kock, from GMD-FIRST Berlin, for all the help at Germany.

References

[Colby-88] R.W.Colby, A.Mayers. *The Encyclopedia of Technical Market Indicators.* Dow Jones - Irwin, Homewwod, Illinois.

[Michie-94] D.Michie, D.J.Spiegelhalter, C.C.Taylor. *Machine Learning, Neural and Statistical Classification.* Ellis Horwood.

[Breiman-93] L.Breiman, J.H.Friedman, C.J.Stone. *Classification and Regression Trees.* Chapman & Hall.

[Cutsem-91] T.van Cutsem. *Decision Trees for Detecting Emergency Voltage Conditions.* Proc. 2nd Workshop on Power System Voltage, pp.229-240.

[Hjorth-94] J.S.U.Hjorth. *Computer Intensive Statistical Methods.* Chapman & Hall.

[Mueller-94] W.Mueller, F.Wysotzki. *Automatic Construction of Decision Trees for Classification.* Ann.Oper.Res.52, pp.231-247.

[Quinlan-86] J.R.Quinlan. *Induction of Decision Trees.* Machine Learning, 1 221-234.

[Unger-81] S.Unger, F.Wysotzki. *Lernfachige Klassifizierungssysteme.* Akademie Vrlg.

RELATÓRIOS TÉCNICOS

DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO

Instituto de Matemática e Estatística da USP

A listagem contendo os relatórios técnicos anteriores a 1993 poderá ser consultada ou solicitada à Secretaria do Departamento, pessoalmente, por carta ou e-mail(mac@ime.usp.br).

Imre Simon
THE PRODUCT OF RATIONAL LANGUAGES
RT-MAC-9301, Maio 1993, 18 pp.

Flávio Soares C. da Silva
AUTOMATED REASONING WITH UNCERTAINTIES
RT-MAC-9302, Maio 1993, 25 pp.

Flávio Soares C. da Silva
ON PROOF-AND MODEL-BASED TECHNIQUES FOR REASONING WITH UNCERTAINTY
RT-MAC-9303, Maio 1993, 11 pp.

Carlos Humes Jr., Leônidas de O.Brandão, Manuel Pera Garcia
A MIXED DYNAMICS APPROACH FOR LINEAR CORRIDOR POLICIES
(A REVISITATION OF DYNAMIC SETUP SCHEDULING AND FLOW CONTROL IN MANUFACTURING SYSTEMS)
RT-MAC-9304, Junho 1993, 25 pp.

Ana Flora P.C.Humes e Carlos Humes Jr.
STABILITY OF CLEARING OPEN LOOP POLICIES IN MANUFACTURING SYSTEMS (Revised Version)
RT-MAC-9305, Julho 1993, 31 pp.

Maria Angela M.C. Gurgel e Yoshiko Wakabayashi
THE COMPLETE PRE-ORDER POLYTOPE: FACETS AND SEPARATION PROBLEM
RT-MAC-9306, Julho 1993, 29 pp.

Tito Homem de Mello e Carlos Humes Jr.
SOME STABILITY CONDITIONS FOR FLEXIBLE MANUFACTURING SYSTEMS WITH NO SET-UP TIMES
RT-MAC-9307, Julho de 1993, 26 pp.

Carlos Humes Jr. e Tito Homem de Mello
A NECESSARY AND SUFFICIENT CONDITION FOR THE EXISTENCE OF ANALYTIC CENTERS IN PATH FOLLOWING METHODS FOR LINEAR PROGRAMMING
RT-MAC-9308, Agosto de 1993

Flavio S. Corrêa da Silva
AN ALGEBRAIC VIEW OF COMBINATION RULES
RT-MAC-9401, Janeiro de 1994, 10 pp.

Flavio S. Corrêa da Silva e Junior Barrera
AUTOMATING THE GENERATION OF PROCEDURES TO ANALYSE BINARY IMAGES
RT-MAC-9402, Janeiro de 1994, 13 pp.

Junior Barrera, Gerald Jean Francis Banon e Roberto de Alencar Lotufo
A MATHEMATICAL MORPHOLOGY TOOLBOX FOR THE KHOROS SYSTEM
RT-MAC-9403, Janeiro de 1994, 28 pp.

Flavio S. Corrêa da Silva
ON THE RELATIONS BETWEEN INCIDENCE CALCULUS AND FAGIN-HALPERN STRUCTURES
RT-MAC-9404, abril de 1994, 11 pp.

Junior Barrera; Flávio Soares Corrêa da Silva e Gerald Jean Francis Banon
AUTOMATIC PROGRAMMING OF BINARY MORPHOLOGICAL MACHINES
RT-MAC-9405, abril de 1994, 15 pp.

Valdemar W. Setzer; Cristina G. Fernandes; Wania Gomes Pedrosa e Flavio Hirata
UM GERADOR DE ANALISADORES SINTÁTICOS PARA GRAFOS SINTÁTICOS SIMPLES
RT-MAC-9406, abril de 1994, 16 pp.

Siang W. Song
TOWARDS A SIMPLE CONSTRUCTION METHOD FOR HAMILTONIAN DECOMPOSITION OF THE HYPERCUBE
RT-MAC-9407, maio de 1994, 13 pp.

Julio M. Stern
MODELOS MATEMATICOS PARA FORMAÇÃO DE PORTFÓLIOS
RT-MAC-9408, maio de 1994, 50 pp.

Imre Simon
STRING MATCHING ALGORITHMS AND AUTOMATA
RT-MAC-9409, maio de 1994, 14 pp.

Valdemar W. Setzer e Andrea Zisman
*CONCURRENCY CONTROL FOR ACCESSING AND COMPACTING B-TREES**
RT-MAC-9410, junho de 1994, 21 pp.

Renata Wassermann e Flávio S. Corrêa da Silva
TOWARDS EFFICIENT MODELLING OF DISTRIBUTED KNOWLEDGE USING EQUATIONAL AND ORDER-SORTED LOGIC
RT-MAC-9411, junho de 1994, 15 pp.

Jair M. Abe, Flávio S. Corrêa da Silva e Marcio Rillo
PARACONSISTENT LOGICS IN ARTIFICIAL INTELLIGENCE AND ROBOTICS.
RT-MAC-9412, junho de 1994, 14 pp.

Flávio S. Corrêa da Silva, Daniela V. Carbogim
A SYSTEM FOR REASONING WITH FUZZY PREDICATES
RT-MAC-9413, junho de 1994, 22 pp.

Flávio S. Corrêa da Silva, Jair M. Abe, Marcio Rillo
MODELING PARACONSISTENT KNOWLEDGE IN DISTRIBUTED SYSTEMS
RT-MAC-9414, julho de 1994, 12 pp.

Nami Kobayashi
THE CLOSURE UNDER DIVISION AND A CHARACTERIZATION OF THE RECOGNIZABLE Z-SUBSETS
RT-MAC-9415, julho de 1994, 29pp.

Flávio K. Miyazawa e Yoshiko Wakabayashi
AN ALGORITHM FOR THE THREE-DIMENSIONAL PACKING PROBLEM WITH ASYMPTOTIC PERFORMANCE ANALYSIS
RT-MAC-9416, novembro de 1994, 30 pp.

Thomaz I. Seidman e Carlos Humes Jr.
SOME KANBAN-CONTROLLED MANUFACTURING SYSTEMS: A FIRST STABILITY ANALYSIS
RT-MAC-9501, janeiro de 1995, 19 pp.

C.Humes Jr. and A.F.P.C. Humes
STABILIZATION IN FMS BY QUASI- PERIODIC POLICIES
RT-MAC-9502, março de 1995, 31 pp.

Fabio Kon e Arnaldo Mandel
SODA: A LEASE-BASED CONSISTENT DISTRIBUTED FILE SYSTEM
RT-MAC-9503, março de 1995, 18 pp.

Junior Barrera, Nina Sumiko Tomita, Flávio Soares C. Silva, Routo Terada
AUTOMATIC PROGRAMMING OF BINARY MORPHOLOGICAL MACHINES BY PAC LEARNING
RT-MAC-9504, abril de 1995,16 pp.

Flávio S. Corrêa da Silva e Fabio Kon
CATEGORIAL GRAMMAR AND HARMONIC ANALYSIS
RT-MAC-9505, junho de 1995,17 pp.

Henrique Mongelli e Routo Terada
ALGORITMOS PARALELOS PARA SOLUÇÃO DE SISTEMAS LINEARES
RT-MAC-9506, junho de 1995, 158 pp.

Kunio Okuda
PARALELIZAÇÃO DE LAÇOS UNIFORMES POR REDUCÃO DE DEPENDÊNCIA
RT-MAC-9507, julho de 1995, 27 pp.

Valdemar W. Setzer e Lowell Monke
COMPUTERS IN EDUCATION: WHY, WHEN, HOW
RT-MAC-9508, julho de 1995, 21 pp.

Flávio S. Corrêa da Silva
REASONING WITH LOCAL AND GLOBAL INCONSISTENCIES
RT-MAC-9509, julho de 1995, 16 pp.

Julio M. Stern
MODELOS MATEMÁTICOS PARA FORMAÇÃO DE PORTFÓLIOS
RT-MAC-9510, julho de 1995, 43 pp.

Fernando Iazzetta e Fabio Kon
A DETAILED DESCRIPTION OF MAXANNEALING
RT-MAC-9511, agosto de 1995, 22 pp.

Flávio Keidi Miyazawa e Yoshiko Wakabayashi
POLYNOMIAL APPROXIMATION ALGORITHMS FOR THE ORTHOGONAL Z-ORIENTED 3-D PACKING PROBLEM
RT-MAC-9512, agosto de 1995, pp.

Junior Barrera e Guillermo Pablo Salas
SET OPERATIONS ON COLLECTIONS OF CLOSED INTERVALS AND THEIR APPLICATIONS TO THE AUTOMATIC PROGRAMMINIG OF MORPHOLOGICAL MACHINES
RT-MAC-9513, agosto de 1995, 84 pp.

Marco Dimas Gubitoso e Jörg Cordsen
PERFORMANCE CONSIDERATIONS IN VOTE FOR PEACE
RT-MAC-9514, novembro de 1995, 18pp.

Carlos Eduardo Ferreira e Yoshiko Wakabayashi
ANAIS DA I OFICINA NACIONAL EM PROBLEMAS COMBINATÓRIOS: TEORIA, ALGORITMOS E APLICAÇÕES
RT-MAC-9515, novembro de 1995, 45 pp.

Markus Endler and Anil D'Souza
SUPPORTING DISTRIBUTED APPLICATION MANAGEMENT IN SAMPA
RT-MAC-9516, novembro de 1995, 22 pp.

Junior Barrera, Routo Terada,
Flávio Corrêa da Silva and Nina Sumiko Tomita
*AUTOMATIC PROGRAMMING OF MMACH'S FOR OCR**
RT-MAC-9517, dezembro de 1995, 14 pp.

Junior Barrera, Guillermo Pablo Salas and Ronaldo Fumio Hashimoto
SET OPERATIONS ON CLOSED INTERVALS AND THEIR APPLICATIONS TO THE AUTOMATIC PROGRAMMING OF MMACH'S
RT-MAC-9518, dezembro de 1995, 14 pp.

Daniela V. Carbogim and Flávio S. Corrêa da Silva
FACTS, ANNOTATIONS, ARGUMENTS AND REASONING
RT-MAC-9601, janeiro de 1996, 22 pp.

Kunio Okuda
REDUÇÃO DE DEPENDÊNCIA PARCIAL E REDUÇÃO DE DEPENDÊNCIA GENERALIZADA
RT-MAC-9602, fevereiro de 1996, 20 pp.

Junior Barrera, Edward R. Dougherty and Nina Sumiko Tomita
AUTOMATIC PROGRAMMING OF BINARY MORPHOLOGICAL MACHINES BY DESIGN OF STATISTICALLY OPTIMAL OPERATORS IN THE CONTEXT OF COMPUTATIONAL LEARNING THEORY.
RT-MAC-9603, abril de 1996, 48 pp.

Junior Barrera e Guillermo Pablo Salas
SET OPERATIONS ON CLOSED INTERVALS AND THEIR APPLICATIONS TO THE AUTOMATIC PROGRAMMINIG OF MMACH'S
RT-MAC-9604, abril de 1995, 66 pp.

Kunio Okuda
CYCLE SHRINKING BY DEPENDENCE REDUCTION
RT-MAC-9605, maio de 1996, 25 pp.

Julio Stern, Fabio Nakano e Marcelo Lauretto
REAL: REAL ATTRIBUTE LEARNING FOR STRATEGIC MARKET OPERATION
RT-MAC-9606, agosto de 1996, 16 pp.