

# Geração de Regras Fuzzy com Pré-Seleção de Regras Candidatas

Marcos Evandro Cintra e Heloisa de Arruda Camargo

<sup>1</sup>Departamento de Computação - Universidade Federal de São Carlos (UFSCar)  
Rod. Washington Luís, Km 235 - Caixa Postal 676 - CEP 13565-905 - São Carlos-SP

{heloisa, marcos.cintra}@dc.ufscar.br

**Abstract.** *The definition of the Rule Base is one of the most important and difficult tasks for the design of Fuzzy Systems. The use of genetic algorithms to create Fuzzy Rule Bases has been widely researched. This paper describes the research on the creation of Fuzzy Rule Bases using genetic algorithms in association with a heuristic of pre-selection of candidate rules. This heuristic is related to the coverage degree of each possible rule for the problem, which allows the selection of a set of rules to be used by the AG, instead of working with all possible rules.*

**Resumo.** *A definição da Base de Regras é uma das tarefas mais importantes e difíceis para o projeto de um Sistema Fuzzy. O uso de algoritmos genéticos para criar BRFs tem sido amplamente pesquisado. Este artigo descreve a pesquisa feita sobre a criação de Bases de Regras Fuzzy usando algoritmo genético em associação com uma heurística de pré-seleção de regras candidatas. Esta heurística está relacionada ao grau de cobertura de cada regra possível para o problema, o que permite a seleção de um conjunto de regras a ser usado pelo algoritmo genético, ao invés de trabalhar com todas as regras possíveis.*

## 1. Introdução

Os sistemas baseados em lógica fuzzy têm sido usados com sucesso para a resolução de problemas de diversas áreas, incluindo classificação de padrões, otimização, controle de processos e modelagem de sistemas [Ichiba et al. 2006, Dumitrescu et al. 2000].

O tipo particular de Sistemas Fuzzy (SF) de interesse para este trabalho são os Sistemas Fuzzy Baseados em Regras (SFBR). Os SFBR são composto por dois componentes principais: a Base de Conhecimento (BC) e o Mecanismo de Inferência (MI).

A BC é composta pela base de dados, que contém as definições dos conjuntos fuzzy relacionados aos termos lingüísticos usados na regras fuzzy, e pela Base de Regras (BR), que armazena o conjunto de regras fuzzy que modelam um determinado problema. O MI é responsável pela aplicação de um processo de raciocínio que utiliza inferências para derivar a saída ou conclusão do sistema, baseado nas regras e fatos conhecidos.

Várias abordagens têm sido usadas para a geração automática da BC a partir de dados numéricos que representam amostras ou exemplos do problema. Dentre as técnicas mais bem sucedidas estão os algoritmos de agrupamento (*clustering*) [Liao et al. 1997], os métodos baseados em gradiente [Nomura et al. 1992], as redes neurais [Jang et al. 1997] e os algoritmos genéticos [Cordon et al. 2004].

Recentemente um esforço considerável tem sido empregado para explorar o uso de Algoritmos Genéticos (AG) no projeto de SFs. Esta iniciativa originou o campo dos Sistemas Fuzzy Genéticos (SFG), que tem atraído a atenção de pesquisadores de diferentes áreas. Um SFG é basicamente um SF acrescido de um processo de aprendizado proveniente de um AG [Cordon et al. 2004].

Este trabalho focaliza um tipo particular de SFG, que são aqueles sistemas que utilizam AG para gerar SFBR, chamados de Sistemas Fuzzy Genéticos Baseados em Regras (SFGBR). A classe dos SFG inclui também as Redes Neurais Fuzzy Genéticas e os Algoritmos de Agrupamento Fuzzy Genéticos.

Durante os últimos 15 anos, os AG têm sido usados na tarefa de definição da BC dos SFBR com diferentes objetivos, como por exemplo, definição das regras fuzzy, redução do número de regras fuzzy pré-estabelecidas, aperfeiçoamento dos conjuntos fuzzy por otimização de parâmetros, eliminação de redundâncias, entre outras.

Uma vertente muito promissora tem sido a de utilizar AG para gerar o conjunto de regras fuzzy usando conjuntos fuzzy previamente definidos e fixos. Assim, o espaço de busca do AG é formado por todas as regras possíveis. Entende-se por regras possíveis, neste contexto, todas as regras geradas por meio da combinação dos termos lingüísticos definidos para as variáveis do problema abordado. Entretanto, dependendo do número de variáveis e do número de conjuntos da partição definida para um determinado problema, o número de regras possíveis pode ser extremamente grande, o que dificulta muito a geração e codificação dos cromossomos e, conseqüentemente, a aplicação de operadores genéticos. Tendo em vista que apenas uma pequena parte do conjunto total de regras possíveis tem valor significativo na classificação de padrões, o objetivo deste trabalho é investigar o uso de uma heurística de pré-seleção de regras candidatas para a geração automática de bases de regras fuzzy através do uso de algoritmos genéticos, reduzindo o número de regras a ser usado pelo AG e, portanto, facilitando a definição e codificação dos cromossomos e aplicação de operadores genéticos.

Este artigo está organizado da seguinte maneira: na seção 2 são apresentadas as principais características dos SF e o Método de Classificação Fuzzy Clássico. Na seção 3 são apresentados os SFGBR. Na seção 4 é apresentada a proposta deste trabalho de geração genética de regras fuzzy com o uso de uma heurística de pré-seleção de regras candidatas, seguidos pelos resultados na seção 5. As conclusões e trabalhos futuros são apresentados na seção 6.

## **2. Sistemas Fuzzy**

Os sistemas baseados em lógica fuzzy, ou SF, usam um mecanismo baseado no raciocínio aproximado que possui grande habilidade para expressar a ambigüidade e subjetividade presentes no raciocínio humano [Pedrycz and Gomide 1998], e são sistemas de decisão adequados para a implementação de processos dedutivos que usam uma BR fuzzy.

A classificação é uma importante tarefa encontrada nas áreas de reconhecimento de padrões, tomada de decisão, mineração de dados e modelagem [Berg et al. 2002]. A seguir são descritos os Sistemas Fuzzy de Classificação (SFC).

## 2.1. Sistemas Fuzzy de Classificação

Dado um conjunto de objetos  $E = \{e_1, e_2, \dots, e_N\}$ , também chamados de *padrões*, o objetivo da classificação é atribuir uma classe  $C_j$  de um conjunto de classes  $C = \{C_1, C_2, \dots, C_M\}$  a um objeto  $e_p$ ,  $e_p = \{a_{p1}, a_{p2}, \dots, a_{pn}\}$  o qual é descrito por  $n$  atributos.

Uma típica regra fuzzy de classificação pode ser expressa por:

$$R_k : \mathbf{SE} X_1 \text{é} A_{i1} \mathbf{E} \dots \mathbf{E} X_n \text{é} A_{in} \mathbf{ENTAO} \text{ Classe} = C_j \quad (1)$$

na qual  $R_k$  é o identificador da regra,  $X_1, \dots, X_n$  são os atributos do padrão considerado no problema (representados por variáveis lingüísticas),  $A_{i1}, \dots, A_{in}$  são os valores lingüísticos usados para representar os valores de tais atributos e  $C_j$  é a classe, fuzzy ou não, a que pertence o padrão.

Um **Mecanismo de Inferência** aplica o conjunto de regras fuzzy ao padrão a ser classificado, determinando a classe a que ele pertence. A maioria dos sistemas fuzzy de classificação utiliza o *Método de Raciocínio Fuzzy Clássico* [Chi et al. 1996, González and Pérez 1999], que classifica um padrão usando a regra que possuir maior grau de compatibilidade com esse padrão.

A seguir o Método de Raciocínio Fuzzy Clássico é apresentado.

### 2.1.1. Método de Raciocínio Fuzzy Clássico

Seja  $e_p = \{a_{p1}, a_{p2}, \dots, a_{pn}\}$  um padrão a ser classificado e  $R_1, R_2, \dots, R_S$ ,  $S$  regras fuzzy para um sistema fuzzy de classificação, cada uma dessas regras com  $n$  antecedentes. Seja  $A_{ki}(a_{pi})$ , com  $i = 1, \dots, n$ , o grau de pertinência do valor do atributo  $a_{pi}$ , do padrão  $e_p$  a ser classificado, no  $i$ -ésimo conjunto fuzzy da regra fuzzy  $R$  definida em 1. O método de Raciocínio Fuzzy Clássico segue os seguintes passos para classificação deste padrão:

1. Calcular o grau de compatibilidade entre o padrão  $e_p$  e cada regra  $R_k$ .

$$Compat(R_k, e_p) = \mathbf{t}(A_{k1}(a_{p1}), A_{k2}(a_{p2}), \dots, A_{kn}(a_{pn}))$$

na qual  $\mathbf{t}$  denota uma *t-norma*.

2. Encontrar a regra que possui maior grau de compatibilidade com o padrão:

$$\max\{Compat(R_k, e_p)\}, k = 1, 2, \dots, n$$

3. Atribuir a classe  $C_j$  ao padrão  $e_p$ , tal que  $C_j$  é a classe da regra  $R_k$  que possui o maior grau de compatibilidade com o padrão.

## 3. Sistemas Fuzzy Genéticos Baseados em Regras

O tópico dos SFGBR, apesar de razoavelmente recente, já deu origem a um número expressivo de trabalhos e, como área de pesquisa, continua em crescente atividade nos dias de hoje. Apesar da dificuldade em classificar as abordagens recentes que surgem a cada ano trazendo enfoques inovadores, já é possível identificar algumas consideradas clássicas, bem como novas tendências.

A organização das abordagens de SFGBR existentes em categorias está fundamentada na definição de dois aspectos básicos: qual parte da BC é o objeto de otimização por AG e qual classe de problemas está sendo abordada - problemas de ajuste ou adaptação de

elementos da base de conhecimento previamente definidos ou problemas de construção ou projeto de partes da BC sem definição prévia.

No contexto dos SFGBR os métodos que combinam as abordagens fuzzy e genética para geração de BCs podem ser divididos em dois grupos principais: métodos que fazem o ajuste ou sintonia de componentes da BC (Adaptação Genética) e métodos que constroem componentes da BC (Construção Genética).

Estão incluídos no grupo de Adaptação Genética os métodos que partem da BR Fuzzy ou BD Fuzzy existentes e usam AG para aperfeiçoar o desempenho do sistema fazendo ajuste ou adaptação de uma ou mais partes da BC. Esses métodos podem ser subdivididos em grupos de acordo com o enfoque adotado: Adaptação Genética dos Conjuntos Fuzzy [Cordón and Herrera 1997, Gurocak 1999, Casillas et al. 2005] e Otimização Genética do número de regras [Ishibuchi et al. 1997, Cordón and Herrera 1997].

O grupo chamado aqui de construção genética engloba os métodos que usam AG para construir (ou projetar) efetivamente um ou mais componentes da BC. A definição prévia de um dos componentes da BC pode ser necessária se o objetivo do processo genético for a geração do outro componente da BC. Esse grupo de métodos foi o que gerou o maior número de pesquisas e pode ainda ser subdividido em três subgrupos: Construção Genética da Base de Regras [Spiegel and Sudkamp 2003, González and Pérez 1999, Hoffmann 2004], Construção Genética da BD [Cordón et al. 2001], e Construção Genética da BC [Cordón and Herrera 2001, Rivas et al. 2003, Angelov 2003, Kim et al. 2006].

Além das abordagens clássicas descritas nos grupos apresentados, um grande número de outras propostas surgiu nos últimos anos e continuam a surgir até os dias de hoje. Dentre eles destaca-se a proposta [Ishibuchi and Yamamoto 2004] para uso de AG para selecionar as regras que vão compor a BRN para problemas de classificação a partir de um conjunto de regras candidatas. O método consiste de duas fases: primeiro as regras candidatas são geradas a partir de conjuntos de dados numéricos pelo uso das medidas de avaliação de regras da mineração de dados (confiança e suporte); depois um número pequeno de regras é selecionado a partir desse conjunto inicial, usando um algoritmo genético multi-objetivos.

Neste trabalho seguimos a abordagem de geração prévia de um conjunto de regras candidatas pelo uso do grau de cobertura (GdC).

#### **4. Geração Genética de Regras Fuzzy**

Quando a BR de um SF é gerada automaticamente pelo uso de um algoritmo genético, o espaço de busca é formado por combinações de um certo número de regras a partir de todas as regras possíveis, considerando as variáveis do problema e os conjuntos fuzzy já definidos. Os cromossomos devem então codificar toda a base de regras ou apenas uma regra, dependendo da abordagem adotada. Na formulação adotada neste trabalho, um cromossomo codifica uma base de regras.

Dependendo do número de variáveis do problema e da partição definida, o conjunto de combinações possíveis de regras que formarão a BC pode ser muito grande, o que pode comprometer o resultado do processo de aprendizado ou mesmo inviabilizá-lo.

O conceito básico associado à proposta apresentada neste trabalho é o uso de um

conhecimento heurístico para a pré-seleção de regras candidatas a serem usadas nos AG, de forma que a dimensão do espaço de busca seja reduzida e que a codificação da base de regras seja simplificada. Esse conhecimento heurístico proposto está associado ao GdC das regras e fundamenta-se na evidência de que, embora o GdC por si só não seja um parâmetro de seleção para apontar quais as regras que devem compor a BR, permite que se descarte boa parte das regras possíveis sem prejuízo da qualidade da BR encontrada. Essas regras a serem descartadas são as que possuem grau de cobertura muito baixo ou nulo. O cálculo do GdC é apresentado em seguida.

#### 4.1. Grau de Cobertura

O GdC de uma regra é definido como sendo a somatória dos graus de compatibilidade dos exemplos do conjunto de treinamento quando classificados somente pela regra em questão. O GdC de cada regra é calculado uma única vez, antes do início da execução dos AG. Assim, regras que não classificam nenhum exemplo do conjunto de treinamento podem ser descartadas na definição do espaço de busca de soluções para os AG.

A seguir é descrito o processo para o cálculo do GdC e classificação das regras, usados neste trabalho.

Seja  $R = \{R_1, R_2, \dots, R_m\}$  o conjunto de  $m$  regras possíveis para o problema. Seja a regra  $R_t = \{a_{p_1}, a_{p_2}, \dots, a_{p_n}\}$ , formada por  $n$  antecedentes, uma das regras de  $R$ . Seja  $E = \{e_1, e_2, \dots, e_q\}$  o conjunto com  $q$  exemplos do problema a ser tratado contendo valores numéricos associados às variáveis lingüísticas definidas para o problema. O grau de cobertura (GdC) de cada regra é calculado seguindo os seguintes passos:

1. Calcular o grau de compatibilidade  $GdC_{(R_i, e_i)}$  do exemplo  $e_1$  com a regra  $R_i$  avaliando e agregando, por meio de uma *t-norma*, os graus de pertinência dos valores dos atributos de  $e_1$  nos conjuntos fuzzy determinados por  $R_i$
2. Para a mesma regra  $R_i$  repetir o passo anterior, calculando os graus de compatibilidade da regra  $R_i$  em relação a cada exemplo de entrada de  $e_2$  a  $e_q$ ;
3. O grau de cobertura da regra  $R_i$  em relação ao conjunto de dados será a soma dos graus de compatibilidade de cada um dos exemplos de entrada em relação a regra  $R_i$ :

$$GdC_{R_i} = \sum \{GdC_{(R_i, e_1)}, GdC_{(R_i, e_2)}, \dots, GdC_{(R_i, e_q)}\}$$

Uma vez calculados os graus de cobertura de todas as regras possíveis e definida a ordenação das regras em ordem decrescente desse grau, foram utilizados dois critérios diferentes para a pré-seleção de regras candidatas:

1. Considerar o conjunto de regras ordenadas até incluir todas as regras que apareceram na BC gerada pelo método de Wang & Mendel (WM) [Wang and Mendel 1992, Wang 2003]. Os experimentos realizados utilizando este critério de pré-seleção de regras são aqui identificados como AG I;
2. Considerar o conjunto de regras ordenadas contendo todas as regras com GdC não-nulo. Os experimentos realizados utilizando este critério de pré-seleção de regras são aqui identificados como AG II;

Além disso foi investigado também o caso em que todas as regras possíveis são consideradas pelo AG, ou seja, sem a pré-seleção de regras, cujos experimentos foram identificados como AG III.

As bases de regras usadas para comparação foram geradas segundo o método de WM por ser um método muito usado e amplamente aceito para a geração de bases de regras a partir de dados numéricos de um problema.

#### **4.2. Codificação dos Cromossomos**

Foram usadas duas codificações diferentes para os cromossomos, porém, ambas as codificações continham em cada cromossomo uma BR para o problema abordado. Para a geração de regras com pré-seleção de regras candidatas, foi utilizada uma codificação binária, que tornou a criação e manipulação dos cromossomos bastante simples. Para a execução com todas as regras possíveis, os cromossomos foram codificados com valores inteiros, cada valor representando uma regra do conjunto com todas as regras possíveis, e o valor 0 representando uma regra inativa.

Para a codificação binária, todas as regras possíveis foram previamente classificadas de acordo com o GdC. No início da execução dos AG defini-se o tamanho do cromossomo e o número inicial de regras ativas no cromossomo. A escolha desse número é discutida na próxima subseção.

Todos os experimentos tiveram 250 iterações, taxa de elitismo de 5%, taxa de cruzamento de 70% e taxa de mutação de 5%.

#### **4.3. Definição do tamanho e número de regras ativas de cada cromossomo**

Para a codificação binária, o tamanho do cromossomo representa o tamanho do espaço de busca e cada cromossomo é inicializado com uma percentagem de regras ativas baseada no número de regras na BR gerada pelo método de WM. O valor 1 (um) em um determinado gene  $k$  indica que a regra  $k$  da lista de regras contendo o espaço de busca está ativa no cromossomo. O valor 0 (zero) indica que aquela regra está inativa.

Para a codificação inteira, o tamanho do cromossomo foi definido como sendo igual ao número de regras da BR gerada pelo método de WM, já que nesta codificação os genes contêm um índice para uma regra do espaço de busca.

Para ambas as codificações os cromossomos foram gerados aleatoriamente, sendo verificada a existência de regras conflitantes, que eram eliminadas.

#### **4.4. Cálculo do Valor de Aptidão**

Na codificação binária os cromossomos podem ser muito grandes, para espaços de busca muito grandes. Desta forma, visando atingir altos níveis de acuidade, o define-se muitas regras ativas na BR, e, geralmente, usa-se um segundo processo para redução da BR induzida [Castro and Camargo 2004].

Com o objetivo de aumentar a interpretabilidade das bases de regras geradas, o Valor de Aptidão (VA) utilizado para os cromossomos com codificação binária considerou, além da taxa de classificação correta, o número de regras nas bases. Para isso, depois de calculada a Taxa de Classificação Correta (TCC) de cada cromossomo, os mesmos recebem uma penalização de acordo com seu número de regras ativas.

Os valores usados para a penalização foram definidos de forma empírica. Todos os cromossomos com um número de regras ativas igual ou menor a *Melhor\_NR*, inicializado com o número de regras da BR gerada pelo método de WM, recebem como

VA a própria TCC, sem alterações. Cromossomos com um número de regras maior que *Melhor\_NR*, recebem como VA o resultado da aplicação de uma redução do valor original da TCC em função da porcentagem de regras adicionais. Para cromossomos com até 50% a mais de regras que *Melhor\_NR*, a penalização é de 20%. Cromossomos com até o dobro de regras de *Melhor\_NR* recebem penalização de 33%. Cromossomos com até o triplo do número de regras de *Melhor\_NR* são penalizados em 50%, e cromossomos ainda maiores são penalizados em 100%.

A cada iteração dos AG o valor de *Melhor\_NR* é atualizado se surgir um indivíduo com um número de regras ativas menor que *Melhor\_NR* que tenha uma TCC maior que *Melhor\_TCC*, que é inicializada com a TCC da BR gerada pelo método de WM.

O VA dos cromossomos com codificação inteira igual à TCC dos mesmos, sem penalizações.

## 5. Resultados

Nesta seção são apresentados os conjuntos de treinamento utilizados e os resultados obtidos com os testes.

### 5.1. Conjuntos de Treinamento Utilizados

Os quatro domínios utilizados estão disponíveis no repositório da UCI Machine Learning [Merz and Murphy 1998]. Foram escolhidos domínios com atributos numéricos. Os exemplos com valores incompletos foram eliminados e, em alguns casos, houve redução de atributos, que foram selecionados aleatoriamente. A Tabela 1 resume as características dos domínios dando o total de instâncias, assim como o número de atributos de cada um (classe incluída), considerados neste trabalho após seleção.

**Tabela 1. Características dos domínios utilizados.**

Domínio	Instâncias	Atributos(incluindo classe)
BUPA	209	5
Carros	392	5
Dia (Diabetes)	724	5
Iris	150	5

Para cada um dos conjuntos foram definidas 3 partições distintas, com 3, 5 e 7 conjuntos fuzzy para cada atributo de entrada.

### 5.2. Resultados Obtidos

A comparação entre os resultados dos experimentos executados com pré-seleção de regras (seleção das melhores regras, em comparação com as regras geradas pelo método de WM), pré-seleção de regras com GdC não-nulo, e uso de todas as regras possíveis demonstra que quando há pré-seleção das melhores regras temos taxa de classificação correta maior, número reduzido de regras nas bases de regras geradas, número menor de iterações até o algoritmo convergir para um resultado ótimo, redução do tempo de processamento devido à redução do tamanho dos cromossomos.

Os resultados mostram que há ganho na TCC para todos os experimentos executados com pré-seleção de regras candidatas, além de redução no número de regras das bases de regras geradas pelos AG em comparação às bases de regras geradas pelo método de WM. Além disso, a pré-seleção de regras candidatas associada à codificação inteira dos cromossomos permitem o uso de AG para geração de bases de regras mesmo quando o número de regras possíveis é muito grande.

A Tabela 2 apresenta o número de regras possíveis para cada partição definida (coluna 2), o número de regras nas bases de regras geradas com pré-seleção de regras pelos critérios do AG I (coluna 3), do AG II (coluna 4) e sem pré-seleção de regras pelo AG III e pelo método de WM (coluna 5).

**Tabela 2. Número de regras.**

Domínio	Total	AG I	AG II	AG III / WM
BUPA 3	243	6.2	12.8	13.8
BUPA 5	3125	21.6	25.4	29
BUPA 7	16807	26	27.8	33
Carros 3	243	13	16.4	20
Carros 5	3125	36.8	46	46.2
Carros 7	16807	34.4	51.2	76.6
Dia 3	162	14.2	28.4	24
Dia 5	1250	51	67.4	84.4
Dia 7	4802	38.6	75.4	159.2
Iris 3	243	7.6	13.8	15
Iris 5	1875	15	41.2	44.8
Iris 7	7203	54.4	61.4	67.2

A Tabela 3 apresenta o VA das bases de regras geradas pelo método de WM na coluna 2, na coluna 3 o VA das bases geradas pelo AG I (pré-seleção de regras pelo critério 1), na coluna 4 o VA das bases geradas pelo AG II (pré-seleção de regras pelo critério 2) e na coluna 5 o VA das bases geradas pelo AG III (sem pré-seleção de regras).

**Tabela 3. Valor de aptidão.**

Domínio	WM	AG I	AG II	AG III
BUPA 3	100	100	100	100
BUPA 5	100	100	100	100
BUPA 7	100	100	100	100
Carros 3	85.21	95.41	87.05	79.15
Carros 5	83.20	92.73	85.75	72.19
Carros 7	80.10	85.79	80.45	44.70

Domínio	WM	AG I	AG II	AG III
Dia 3	95.70	100	100	100
Dia 5	98.20	100	100	99.45
Dia 7	97.93	99.30	94.45	93.11
Iris 3	100	100	100	100
Iris 5	98.66	97.33	98.68	97.24
Iris 7	100	100	97.18	77.33

Os resultados obtidos indicam que o uso da heurística proposta para pré-seleção de regras fuzzy para a geração automática de bases de regras gera bases de regras com TCC maiores que as taxas de classificação corretas das bases geradas sem pré-seleção de regras, além de reduzir bastante o número de regras nas bases de regras produzidas.

Os resultados validam o uso da heurística de pré-seleção de regras, permitindo o uso de AG mesmo para problemas com um número bem maior de variáveis de entrada e



de conjuntos fuzzy para cada variável, casos em que o número de regras possíveis para o problema é extremamente grande e torna o uso da codificação binária inviável.

O uso do algoritmo auto-adaptativo para definir os valores de *Melhor\_NR* e *Melhor\_TCC* permitiu a geração de bases de regras menores que as geradas pelo método de WM, contribuindo para uma redução no tempo de convergência dos AG.

## 6. Conclusão

Este trabalho apresentou uma heurística para pré-seleção de regras candidatas para geração de bases de regras a partir de um conjunto de exemplos.

O processo de pré-seleção é dividido em duas etapas: na primeira, todas as regras possíveis para um determinado problema são geradas. Na segunda etapa essas regras são classificadas e ordenadas de acordo com seu GdC, permitindo a seleção das regras mais representativas a serem usadas como espaço de busca por AG.

Os resultados dos experimentos evidenciam a melhora obtida no desempenho do método de geração automática das regras. Foram obtidas bases de regras compactas e com altas TCC.

## Referências

- Angelov, P. P. (2003). An evolutionary approach to fuzzy rule-based model synthesis using indices for rules. *Fuzzy Sets and Systems*, 137:325–338.
- Berg, J., Kaymak, U., and Bergh, W. (2002). Fuzzy classification using probability-based rule weighting. In *Proceedings of the 11th IEEE International Conference on Fuzzy Systems*. Hawaii - USA.
- Casillas, J., Cordon, O., Jesus, M., and Herrera, F. (2005). Genetic tuning of fuzzy rule deep structures preserving interpretability and its interaction with fuzzy rule set reduction. *IEEE Transactions on Fuzzy Systems*, 13:13–29.
- Castro, P. A. D. and Camargo, H. A. (2004). Learning and optimization of fuzzy rule base by means of self adaptive genetic algorithms. *IEEE International Conference on Fuzzy Systems*, 2:1037–1042. Budapeste - Hungria.
- Chi, Z., Yan, H., and Pham, T. (1996). *Fuzzy Algorithms with Applications to Image Processing and Pattern Recognition*. World Scientific.
- Cordon, O. and Herrera, F. (1997). A three stage evolutionary process for learning descriptive and approximate fuzzy logic controller knowledge bases from examples. *International Journal of Approximate Reasoning*, 7-4:369–407.
- Cordon, O. and Herrera, F. (2001). Hybridizing genetic algorithms with sharing scheme and evolution strategies for designing approximate fuzzy rule-based systems. *Fuzzy Sets and Systems*, 118:235–255.
- Cordon, O., Herrera, F., and Villar, P. (2001). Generating the knowledge base of a fuzzy-based system by the genetic learning of the data base. *IEEE Transactions on Fuzzy Systems*, 9(4):667–674.
- Cordon, O., Gomide, F. A. C., Herrera, F., Hoffmann, F., and Magdalena, L. (2004). Special issue on genetic fuzzy systems. *Fuzzy Sets and Systems*, 141.

- Dumitrescu, D., Lazzerini, B., and Jair, L. (2000). *Fuzzy Sets and Their Application to Clustering and Training*. International Series on Computational Intelligence. CBC Press.
- González, A. and Pérez, R. (1999). SLAVE: A genetic learning system based on an iterative approach. *IEEE Transactions on Fuzzy Systems*, 7:176–191.
- Gurocak, H. B. (1999). A genetic-algorithm-based method for tuning fuzzy logic controllers. *Fuzzy Sets and Systems*, 108-1:39–47.
- Hoffmann, F. (2004). Combining boosting and evolutionary algorithms for learning of fuzzy classification rules. *Fuzzy Sets and Systems*, 141:47–58.
- Ichiba, D., Hara, K., and Kanoh, H. (2006). Spatial interpolation of traffic data by genetic fuzzy system. *International Symposium on Evolving Fuzzy Systems*, pages 280–285. Ambleside - Inglaterra.
- Ishibuchi, H., Murata, T., and Turksen, I. B. (1997). Single-objective and two-objective genetic algorithms for selecting linguistic rules for pattern classification problems. *Fuzzy Sets and Systems*, 89:134–150.
- Ishibuchi, H. and Yamamoto, T. (2004). Fuzzy rule selection by multi-objective genetic local search algorithms and rule evaluation measures in data mining. *Fuzzy Sets and Systems*, 141:59–88.
- Jang, S. R., Sun, C. T., and Mizutani, E. (1997). *Neuro-Fuzzy and Soft Computing*. Prentice Hall.
- Kim, M., Kim, C., and Lee, J. (2006). Evolving compact and interpretable takagi-sugeno fuzzy models with a new encoding scheme. *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics*, 36:1006–1023.
- Liao, T. W., Celmins, A. K., and Hammell, R. J. (1997). A fuzzy C-Means variant for the generation of fuzzy term sets. *Fuzzy Sets and Fuzzy Systems*, 135:241–257.
- Merz, C. J. and Murphy, P. M. (1998). UCI repository of machine learning databases, [<http://www.ics.uci.edu>]. Irvine, CA, UCLA.
- Nomura, H., Hayashi, L., and Wakami, N. (1992). A learning method of fuzzy inference rules by descent method. In *IEEE International Conference on Fuzzy Systems*, pages 203–210. San Diego - USA.
- Pedrycz, W. and Gomide, F. (1998). *An Introduction to Fuzzy Sets*. MIT Press.
- Rivas, V. M., Merelo, J. J., Rojas, I., Romero, G., Castillo, P. A., and Carpio, J. (2003). Evolving two-dimensional fuzzy systems. *Fuzzy Sets and Systems*, 138:381–398.
- Spiegel, D. and Sudkamp, T. (2003). Sparse data in the evolutionary generation of fuzzy models. *Fuzzy Sets and Systems*, 138:363–379.
- Wang, L. (2003). The WM method completed: a flexible fuzzy system approach to data mining. *IEEE Trans. on Fuzzy Systems*, 11:768–782.
- Wang, L. and Mendel, J. (1992). Generating fuzzy rules by learning from examples. *IEEE Trans. on SMC*, 22:414–427.