Recebido: 28/01/10
Aprovado: 09/04/12

Proposta de melhorias a um algoritmo para agrupamento de padrões via Colônia de Formigas

Rosangela Villwock (UNIOESTE; PR) – rosangela.villwock@unioeste.br
Rua Universitária, 2069, Bairro Universitário, CEP: 85.819-110, Cascavel, PR
Maria Teresina Arns Steiner (UFPR; PUCPR; PR) – tere@ufpr.br
Paulo Henrique Siqueira (UFPR; PR) – paulohs@ufpr.br.

RESUMO Métodos inspirados em formigas são uma grande promessa para problemas de agrupamento. No algoritmo de Agrupamento baseado em Formigas, proposto por Deneubourg *et al.* (1991), os padrões são espalhados em uma grade e a cada formiga é atribuído um padrão. As formigas são responsáveis por carregar, transportar e descarregar os padrões na grade. Após a convergência do algoritmo, a recuperação dos grupos é feita, usando-se as posições dos padrões na grade. O objetivo do presente artigo é propor melhorias a este algoritmo, doravante denominado de algoritmo proposto, avaliando o seu desempenho comparativamente ao Método de Ward, aos Mapas de Kohonen Unidimensionais e ao algoritmo ACAM (*Ant-based Clustering Algorithm Modified*), proposto por Boryczka (2009). No algoritmo proposto, as principais modificações foram: a introdução de uma comparação da probabilidade de descarregar um padrão na posição escolhida aleatoriamente, com a probabilidade de deixar este padrão em sua posição atual; a introdução de uma avaliação da probabilidade de uma posição vizinha, quando a decisão de descarregar um padrão for positiva e a célula em que o padrão deveria ser descarregado estiver ocupada; e a substituição do padrão carregado por uma formiga, caso este padrão não seja descarregado em 100 iterações consecutivas. Para a avaliação do desempenho do algoritmo proposto, foram utilizadas três bases de dados reais e públicas (ÍRIS, WINE e PIMA Indians Diabetes). Os resultados mostraram que houve superioridade no desempenho do algoritmo proposto, em relação ao ACAM para duas das três bases de dados e igualdade, em relação aos outros dois métodos.

Palavras-chave Mineração de Dados; Metaheurística; Agrupamento Baseado em Formigas.

ABSTRACT *Methods inspired by ants are a great promise for clustering problems. In the Ant-based clustering algorithm, proposed by Deneubourg et al. (1991), the patterns are spread in a grid and each ant is assigned a pattern. The ants are responsible for loading, unloading and transport patterns in the grid. After the convergence of the algorithm, the recovery of the groups is done using the positions of the patterns on the grid. The aim of this paper is to propose improvements to this algorithm, hereinafter called the proposed algorithm, evaluating its performance compared to the method of Ward, to One-dimensional Self-Organizing Kohonen maps and to ACAM (Ant-based Clustering Algorithm Modified), proposed by Boryczka (2009). In the proposed algorithm, the main changes were: the introduction of a comparison of the probability of dropping a pattern in a location chosen at random with the probability of dropping this pattern in its current position; the introduction of an assessment of the probability of position likelihood, where the decision to dropping a pattern is positive and the cell in which the pattern should be dropped is busied; and the replacement of the pattern carried by an ant, if this pattern has not been dropped in 100 consecutive iterations. To evaluate the performance of the proposed algorithm, three real and public databases were used (IRIS, WINE and PIMA Indians Diabetes). The results showed the proposed algorithm it is superior to ACAM for two of the three databases and equal when compared to the other two methods.*

Keywords *Data Mining; Metaheuristic; Ant-based Clustering.*

VILLWOCK, R.; STEINER, M. T. A.; SIQUEIRA, P. H. Proposta de melhorias a um algoritmo para agrupamento de padrões via Colônia de Formigas. **GEPROS. Gestão da Produção, Operações e Sistemas**, Ano 7, nº 2, abr-jun/2012, p. 123-137.

1. INTRODUÇÃO

Sociedades de insetos sociais são sistemas distribuídos que apresentam uma organização social altamente estruturada, apesar da simplicidade dos seus indivíduos. Como resultado desta organização, colônias de formigas podem realizar tarefas complexas que, em alguns casos, excedem a capacidade individual de uma única formiga. Na área de pesquisa sobre "algoritmos de formigas", estudam-se modelos inspirados na observação do comportamento de formigas reais e usam-se estes modelos como fonte de inspiração para o desenvolvimento de novos algoritmos, para a solução de problemas de otimização e de controle distribuído (DORIGO e STÜTZLE, 2004).

Otimização por colônia de formigas (*Ant Colony Optimization – ACO*) é uma metaheurística em que a colônia de formigas artificiais coopera para encontrar boas soluções para problemas de otimização discretos difíceis (DORIGO e STÜTZLE, 2004). Dorigo, Caro e Gambardella (1999) apresentam uma avaliação de trabalhos recentes em algoritmos de formiga para a otimização discreta e introduzem a metaheurística *ACO*. Dorigo e Blum (2005) apresentam uma pesquisa sobre resultados teóricos em otimização por colônia de formigas.

Segundo Dorigo, Maniezzo e Colorni (1996), na escolha de um trajeto, uma formiga é influenciada pela intensidade dos rastros de feromônio. Um nível mais alto de feromônio dá para uma formiga um estímulo mais forte e, assim, uma probabilidade mais alta para escolhê-lo. O resultado é que uma formiga encontrará um rastro mais forte em caminhos mais curtos. Como consequência, o número de formigas que seguem estes caminhos será mais alto. Isto fará com que a quantidade de feromônio no caminho mais curto cresça mais rápido do que no mais longo e, então, a probabilidade com que qualquer formiga escolhe um caminho para seguir, é rapidamente tendenciada para o mais curto. O resultado final é que muito depressa todas as formigas escolherão o caminho mais curto.

Entre os comportamentos dos insetos sociais, o mais amplamente reconhecido é a habilidade das formigas para trabalhar em grupo, com o intuito de desenvolver uma tarefa que não poderia ser executada por um único agente. Também vista na sociedade humana, esta habilidade das formigas é um resultado de efeitos cooperativos. O efeito cooperativo recorre ao fato de que o efeito de dois ou mais indivíduos ou partes coordenadas é mais alto do que o total dos efeitos individuais. O número alto de indivíduos em colônias de formigas e a abordagem descentralizada para tarefas coordenadas (executadas de forma simultânea), significam que colônias de formigas mostram altos graus de paralelismo, auto-organização e tolerância a falhas. Estas características são desejadas em técnicas de otimização modernas (BORICZKA, 2009).

O algoritmo de Agrupamento baseado em Colônia de Formigas foi escolhido para estudo, análise e novas propostas, devido a diversos fatores. Primeiramente, é uma metaheurística relativamente nova e tem recebido atenção especial, principalmente porque ainda exige muita investigação, para melhorar seu desempenho, estabilidade e outras características consideradas "chave", que fariam de tal algoritmo uma ferramenta madura para mineração de dados (BORYCZKA, 2009). Ainda, a quantidade de grupos de padrões vai se definindo no decorrer da execução do algoritmo, ou seja, tal informação não é necessária para a execução do mesmo.

O objetivo do presente artigo é apresentar melhorias ao algoritmo de Agrupamento baseado em Formigas, originalmente proposto por Deneubourg *et al.* (1991), doravante denominado de algoritmo proposto, avaliando o seu desempenho comparativamente ao Método de Ward e aos Mapas de Kohonen Unidimensionais e ao algoritmo ACAM (*Ant-based Clustering Algorithm Modified*), proposto por Boryczka (2009). O método da área de Estatística Multivariada (Método de Ward) foi utilizado por ser um dos métodos mais consagrados na literatura (JOHNSON e WICHERN, 1998); já os Mapas de Kohonen Unidimensionais foram utilizados porque, assim como o Agrupamento baseado em Formigas, executam as tarefas de agrupamento e mapeamento topográfico simultaneamente.

Este trabalho está organizado da seguinte forma: na seção 2, é apresentada uma revisão bibliográfica sobre o Agrupamento baseado em Formigas; na seção 3, são apresentadas as bases de dados utilizadas, assim como as principais contribuições (modificações e melhorias) para o Agrupamento baseado em Formigas; na seção 4, são apresentados os resultados e discussões e, finalmente, na seção 5, são apresentadas as considerações finais.

2. REVISÃO BIBLIOGRÁFICA

No Agrupamento baseado em Formigas, proposto por Deneubourg *et al.* (1991), as formigas foram representadas como agentes simples, que se moviam aleatoriamente em uma grade quadrada. Os padrões foram dispersos dentro desta grade e poderiam ser carregados, transportados e descarregados pelos agentes (formigas). Estas operações são baseadas na similaridade e na densidade dos padrões distribuídos dentro da vizinhança local dos agentes; padrões isolados ou cercados por dissimilares são mais prováveis de serem carregados e então, descarregados numa vizinhança de similares.

As decisões de carregar e descarregar padrões são tomadas pelas probabilidades P_{pick} e P_{drop}, dadas pelas equações (1) e (2), a seguir, respectivamente.

$$P_{pick} = \left(\frac{k_p}{k_p + f(i)} \right)^2 \quad (1)$$

$$P_{drop} = \left(\frac{f(i)}{k_d + f(i)} \right)^2 \quad (2)$$

Nestas equações, *f(i)* é uma estimativa da fração de padrões localizados na vizinhança que são semelhantes ao padrão atual da formiga e k_p e k_d são constantes reais. Este valor é utilizado no cálculo da probabilidade de um padrão ser carregado ou descarregado pela formiga. A cada iteração, a cada formiga, o padrão carregado por ela, bem como seus vizinhos, são avaliados para o cálculo de *f(i)*. No trabalho de Deneubourg *et al.* (1991), os autores usaram $k_p = 0{,}1$ e $k_d = 0{,}3$. Os autores obtiveram a estimativa *f*, através de uma memória de curto prazo de cada formiga, onde o conteúdo da última célula da grade analisada é armazenado. Esta escolha da função de vizinhança *f(i)* foi essencialmente motivada pela sua facilidade de realização por robôs simples.

Lumer e Faieta (1994, *apud* HANDL *et al.*, 2006) introduziram um número de modificações ao modelo, que permitiu a manipulação de dados numéricos e melhorou a qualidade da solução e o tempo da convergência do algoritmo. A ideia era definir uma medida de similaridade ou dissimilaridade entre os padrões, já que no algoritmo proposto inicialmente, os objetos eram similares, se os objetos fossem idênticos; e dissimilares, se não fossem idênticos. No referido trabalho, aparece pela primeira vez, o mapeamento topográfico.

Segundo Vizine *et al.* (2005), a ideia geral deste algoritmo é ter dados semelhantes no espaço n-dimensional original, em regiões vizinhas da grade, ou seja, dados que são vizinhos na grade indicam padrões semelhantes no espaço original.

No trabalho de Lumer e Faieta (1994, *apud* HANDL *et al*., 2006), a decisão de carregar padrões é baseada na probabilidade P_{pick}, dada pela equação (1) anterior e a decisão de descarregar padrões é baseada na probabilidade P_{drop}, dada pela equação (3) a seguir, onde *f(i)* é dada pela equação (4).

$$P_{drop} = \begin{cases} 2f, \text{ se } f(i) < k_d \\ 1, \text{ se } f(i) \geq k_d \end{cases} \quad (3)$$

$$f(i) = \max\left\{0, \frac{1}{\sigma^2} \sum_{j \in L} \left[1 - \frac{d(i,j)}{\alpha}\right]\right\} \quad (4)$$

Na equação (4), *d(i, j)* é uma função de dissimilaridade entre padrões *i* e *j* (que pode ser, por exemplo, a Distância Euclidiana entre os vetores *i* e *j*), pertencentes ao intervalo [0, 1] (ou seja, a dissimilaridade é calculada entre os padrões e, caso necessário, é padronizada no intervalo [0, 1]); α é um parâmetro escalar dependente dos dados (padrões) e pertencente ao intervalo [0, 1]; *L* é a vizinhança local de tamanho igual a σ^2, onde σ é o raio de percepção (ou vizinhança). Em Lumer e Faieta (1994, *apud* HANDL *et al*., 2006), $k_p = 0{,}1$, $k_d = 0{,}15$ e $\alpha = 0{,}5$. Segundo Handl, Knowles e Dorigo (2006), σ^2 é geralmente pertencente ao intervalo [9, 25], e este parâmetro pode ser ajustado e representa o tamanho da vizinhança (quadrada σ x σ).

Os algoritmos de Agrupamento baseados em Formigas estão principalmente baseados nas versões propostas por Deneubourg *et al*. (1991) e Lumer e Faieta (1994, *apud* HANDL *et al*., 2006). Várias modificações foram introduzidas para melhorar a qualidade do agrupamento e, em particular, a separação espacial entre os grupos na grade (BORICZKA, 2009).

Mudanças que melhoram a separação espacial dos grupos e permitem que o algoritmo seja mais robusto foram introduzidas por Handl, Knowles e Dorigo (2006). Uma delas é a restrição na função f(i) dada pela equação (5), a seguir, que serve para penalizar dissimilaridades elevadas.

$$f^*(i) = \begin{cases} \frac{1}{\sigma^2} \sum_{j \in L} \left[1 - \frac{d(i,j)}{\alpha}\right], \text{ se } \forall j \left(1 - \frac{d(i,j)}{\alpha}\right) > 0 \\ 0 \qquad\qquad , \text{caso contrário} \end{cases} \quad (5)$$

Segundo Vizine *et al*. (2005), uma dificuldade na aplicação do algoritmo de Agrupamento por Formigas em problemas complexos é que, na maioria dos casos, eles geram um número de grupos muito maior que o real. Além disso, estes algoritmos normalmente não estabilizam em uma solução de agrupamento, ou seja, eles constantemente constroem e desconstroem grupos durante o processo. Para superar estas dificuldades e melhorar a qualidade dos resultados, os autores propuseram um Algoritmo de Agrupamento por Formigas Adaptável (*Adaptative Ant Clustering Algoritthm* – A^2CA). Uma modificação incluída nesta abordagem é um programa de resfriamento para o parâmetro que controla a probabilidade de formigas apanharem objetos da grade.

2.1. Parâmetros da função de vizinhança

A separação espacial dos grupos na grade é crucial para que grupos individuais sejam bem definidos, permitindo a sua recuperação automática. A proximidade espacial, quando ocorrer, pode indicar a formação prematura do agrupamento (HANDL *et al.*, 2006).

A definição dos parâmetros da função de vizinhança é um fator decisivo na qualidade do agrupamento. No caso do raio de percepção σ, é mais atrativo empregar vizinhanças maiores para melhorar a qualidade do agrupamento e da distribuição na grade. Porém, este procedimento é mais caro computacionalmente, já que o número das células a serem consideradas para cada ação cresce quadraticamente com o raio, e ainda inibe a formação rápida dos grupos durante a fase de distribuição inicial. Um raio da percepção que aumenta gradualmente com o tempo, acelera a dissolução de grupos pequenos preliminares (HANDL *et al.*, 2006). Um raio de percepção progressivo também foi utilizado por Vizine *et al.* (2005).

Além disso, depois de uma fase inicial de agrupamento, Handl *et al.* (2006) substituíram o parâmetro escalar $\frac{1}{\sigma^2}$ por $\frac{1}{N_{occ}}$, na equação (5), onde N_{occ} é o número de células da grade ocupadas, observadas dentro da vizinhança local. Assim, somente a semelhança e não a densidade, foi levada em conta.

Segundo Boryczka (2009), a função de vizinhança depende do raio de percepção σ^2. Segundo a autora, o valor estável deste parâmetro pode causar comportamentos inadequados, porque não é possível distinguir as diferenças entre grupos de tamanhos diferentes. Por outro lado, um raio de percepção grande pode ser útil no começo do algoritmo, quando os padrões são espalhados aleatoriamente na grade. Para superar esta dificuldade, a autora propôs a substituição do escalar $\frac{1}{\sigma^2}$ na equação (5) pelo escalar $\frac{\sigma_0^2}{\sigma^2}$, onde σ_0 é o raio de percepção inicial, em seu algoritmo ACAM (*Ant-based Clustering Algorithm* ou Algoritmo de Agrupamento Baseado em Formigas Modificado).

Segundo Handl *et al.* (2006), α determina a porcentagem de padrões na grade classificados como semelhantes. A escolha de um valor muito pequeno para α, impede a formação de grupos na grade; por outro lado, a escolha de um valor muito grande para α, resulta na fusão de grupos.

Definir o parâmetro α não é simples e a sua escolha é altamente dependente da estrutura do conjunto de dados. Um valor inadequado é refletido por uma excessiva ou extremamente baixa atividade na grade. A quantidade de atividade é refletida pela frequência de operações com sucesso da formiga em carregar e descarregar. Com base nestas análises, Handl *et al.* (2006) propuseram uma adaptação automática de α. Já Boryczka (2009) propôs um novo esquema de adaptação para o valor de α.

Tan *et al.* (2007) examinam o parâmetro escalar de dissimilaridade em abordagens de Colônia de Formigas, para agrupamento de dados. Os autores mostram que não há necessidade de se usar uma adaptação automática de α. Os mesmos propõem um método para calcular um α fixo para cada base de dados. O valor α é calculado independentemente do processo de agrupamento.

Para medir a similaridade entre os padrões, diferentes métricas são utilizadas. Handl *et al.* (2006) utilizam distância Euclidiana para dados sintéticos e Co-seno para dados reais. Boryczka (2009) testou diferentes medidas de dissimilaridade: Euclidiana, Co-seno e medidas de Gower.

2.2. O algoritmo básico proposto por Deneubourg *et al*. (1991)

O procedimento proposto por Deneubourg *et al.* (1991) consta, basicamente, das seguinte ideias: numa fase inicial, todos os padrões são aleatoriamente espalhados na grade. Depois, cada formiga escolhe aleatoriamente um padrão para carregar e é colocada em uma posição aleatória na grade. Na próxima fase, chamada de fase de distribuição, em um laço (*loop*) simples, cada formiga é selecionada aleatoriamente. Esta formiga se desloca na grade, executando um passo de comprimento L, numa direção determinada aleatoriamente. Segundo Handl *et al.* (2006), o uso de um tamanho de passo grande acelera o processo de agrupamento. A formiga então decide, probabilisticamente, se descarrega seu padrão nesta posição.

Se a decisão de descarregar o padrão for negativa, escolhe-se aleatoriamente outra formiga e recomeça-se o processo. No caso de decisão positiva, a formiga descarrega o padrão em sua posição atual na grade, se esta estiver livre. Se esta célula da grade estiver ocupada por outro padrão, o mesmo deve ser descarregado numa célula imediatamente vizinha desta, que esteja livre, por meio de uma procura aleatória.

A formiga procura, então, por um novo padrão para carregar. Dentre os padrões livres na grade, ou seja, dentre os padrões que não estão sendo carregados por nenhuma formiga, ela seleciona aleatoriamente um, vai para a sua posição na grade, faz a avaliação da função de vizinhança e decide probabilisticamente se carrega este padrão. Este processo de escolha de um padrão livre na grade é executado até que a formiga encontre um padrão que deva ser carregado. Só então esta fase é reiniciada, escolhendo-se outra formiga até que um critério de parada seja satisfeito.

2.3. Recuperação do agrupamento

Para a recuperação do agrupamento, o processo inicia-se com cada padrão, formando um grupo. Depois de calcular as distâncias entre todos os grupos, deve-se fundir (ligar) os dois grupos com menor distância. Os tipos de ligações mais comuns são: Ligação Simples, Ligação Completa, Ligação Média e Método de Ward (JOHNSON e WICHERN, 1998). As distâncias entre os grupos são definidas em termos de sua distância na grade. Cada padrão é agora composto por apenas dois atributos, que o posicionam na grade bidimensional. A distância entre cada dois padrões é então a distância Euclidiana entre dois pontos da grade. Este processo se repete até que um critério de parada seja satisfeito.

Quando padrões em torno das bordas dos grupos estão isolados, Handl *et al.* (2006) introduziram um peso que incentiva a fusão destes padrões com os grupos.

2.4. Avaliação do agrupamento

Na avaliação de grupos, diferentes aspectos podem ser observados: determinação da tendência de agrupamento de um conjunto de dados, comparação dos resultados de uma análise de grupos com resultados externamente conhecidos, avaliação de quão bem os resultados de uma análise de grupos se ajustam aos dados sem referência à informação externa, comparação dos resultados de dois diferentes conjuntos de análise de grupos para determinar qual deles é melhor ou, ainda, determinação do número correto de grupos (TAN *et al.*, 2005).

Segundo Tan *et al.* (2005), as medidas numéricas aplicadas para julgar vários aspectos de avaliação de grupos são classificadas em três tipos: os índices externos são usados para medir até que ponto rótulos de grupos correspondem a rótulos de classes externamente fornecidos; os índices internos são usados para medir quão boa é a estrutura de agrupamento sem relação com informação externa e os índices relativos são usados para comparar dois grupos ou agrupamentos diferentes.

Boryczka (2009) utilizou, em seu trabalho, dois índices internos (a Variância Intra-Grupos e o Índice *Dunn*) e dois índices externos (a medida *F* e o Índice Aleatório). Estas medidas são descritas a seguir, e utilizadas também, neste trabalho.

A medida F usa a ideia de precisão e memória da recuperação da informação. Cada classe *i* é um conjunto de n_i padrões desejados; cada grupo *j* (gerado pelo algoritmo) é um conjunto de n_j padrões; n_{ij} é o número de padrões da classe *i*, pertencentes ao grupo *j*. Para cada classe *i* e grupo *j*, a precisão *p* e a memória *r* são definidas como $p(i,j) = \frac{n_{ij}}{n_j}$ e $r(i,j) = \frac{n_{ij}}{n_i}$, respectivamente. O valor da medida *F* é dado pela equação (6).

$$F = \sum_i \frac{n_i}{n} \max_j \{F(i,j)\} \qquad \text{onde:} \quad F(i,j) = \frac{(b^2+1)\,.\,p(i,j)\,.\,r(i,j)}{b^2\,.\,p(i,j) + r(i,j)} \tag{6}$$

O valor de *b* deve ser "1", para proporcionar pesos iguais para a precisão *p* e a recordação *r*. Na equação (6), *n* é o tamanho do conjunto de dados; *F* é limitada ao intervalo [0, 1] e deve ser maximizada. Já o Índice Aleatório (*R*) é dado pela equação (7), onde *a*, *b*, *c* e *d* são calculados para todos os possíveis pares de padrões *i* e *j* e seus respectivos grupos *U* (classificação correta - $c_U(i)$ e $c_U(j)$) e *V* (solução gerada pelo algoritmo de agrupamento – $c_V(i)$ e $c_V(j)$). O valor de *R* está limitado no intervalo [0, 1] e deve ser maximizado.

$$R = \frac{a+d}{a+b+c+d} \qquad \text{onde:} \quad \begin{aligned} a &= \left|\{i,j \mid c_U(i) = c_U(i) \wedge c_V(i) = c_V(i)\}\right| \\ b &= \left|\{i,j \mid c_U(i) = c_U(i) \wedge c_V(i) \neq c_V(i)\}\right| \\ c &= \left|\{i,j \mid c_U(i) \neq c_U(i) \wedge c_V(i) = c_V(i)\}\right| \\ d &= \left|\{i,j \mid c_U(i) \neq c_U(i) \wedge c_V(i) \neq c_V(i)\}\right| \end{aligned} \tag{7}$$

2.5. Outros métodos de agrupamento utilizados

Neste trabalho, conforme já comentado, foram selecionados três métodos para serem comparados com o algoritmo aqui proposto: Método Ward (método estatístico clássico), encontrado, por exemplo, em Johnson e Wichern (1998); Redes Neurais de Kohonen Unidimensional (realizam o agrupamento e o mapeamento topográfico simultaneamente) encontrado, por exemplo, em Fausett (1994) e o *ACAM* (método análogo ao aqui proposto), devido a Boryczka (2009) que propôs modificações no algoritmo de Lumer e Faieta. Para aumentar a robustez do agrupamento baseado em formigas, a autora incorporou duas principais modificações em relação à abordagem clássica: 1. *an adaptive perception scheme occurred in the density function*, ou seja, um esquema de percepção adaptativa, ocorrido na função densidade e 2. *a cooling scheme of α-adaptation*, ou seja, um esquema de resfriamento para a adaptação do parâmetro α, modificações já comentadas na seção 2.1.

3. MATERIAL E MÉTODOS DA PESQUISA

As bases de dados utilizadas neste trabalho foram: Iris, Wine e Pima Indians Diabets[1]. A Tabela 1 mostra o número de padrões, o número de atributos e a quantidade de grupos para cada uma destas bases de dados. Os dados foram padronizados antes da aplicação dos métodos para agrupamento. A padronização foi feita por dimensão.

1 As bases de dados utilizadas estão disponíveis em http://mlearn.ics.uci.edu/databases.

O Método de Ward (JOHNSON e WICHERN, 1998) foi aplicado às três bases de dados, com o auxílio do *software* computacional MATLAB 2008. A medida de dissimilaridade utilizada foi a distância euclidiana por ser a mais conhecida entre as medidas de dissimilaridade e por ter sido empregada em trabalhos anteriores para todos os métodos, aqui, utilizados.

Tabela 1 – Bases de dados utilizados para avaliação do algoritmo.

Base de Dados	N° de Padrões	N° de atributos	N° de grupos
Íris	150	4	3
Wine	178	13	3
Pima Indians Diabetes	768	8	2

Fonte: Os autores.

O agrupamento por Mapas de Kohonen, aplicado às bases de dados, foi implementado no *software* computacional MATLAB 2008 e foi executado por 10 vezes para cada base de dados. Detalhes de implementação podem ser obtidos em Villwock (2009).

O método *ACAM*, diferentemente dos demais (Ward, Kohonen e o proposto), não foi implementado. A comparação foi realizada diretamente com os resultados apresentados em Borycska (2009).

3.1. Algoritmo proposto

Este algoritmo é aqui detalhado, pelo fato de ser a contribuição deste trabalho. O algoritmo proposto, baseado no algoritmo básico de Deneubourg *et al.* (1991), apresentado na seção 2.1, foi implementado no *software* computacional MATLAB 2008. Para tanto, foram utilizados recursos da grade computacional do LCPAD: Laboratório Central de Processamento de Alto Desempenho/UFPR, parcialmente financiado pela FINEP, projeto CT-INFRA/UFPR/Modelagem e Computação Científica.

O algoritmo implementado utilizou como critério de parada o número de iterações e o algoritmo foi executado por 10 vezes. Sendo n o número de padrões e m o número de atributos, o número de iterações N_{max} foi definido como $N_{max} = 500.n.m$. Para definir o número máximo de iterações, vários testes foram realizados. No algoritmo implementado, foram definidas duas fases (inicial e final), nas quais os ajustes de parâmetros são modificados. A fase inicial foi definida como $t_{inicial} = 0{,}2.N_{max}$.

Na definição do tamanho da grade, escolheu-se o número de células, como sendo 10 vezes o número de padrões e foram utilizadas 10 formigas (p=10), como em Handl *et al.* (2006). Observou-se que a alteração destes valores não é imprescindível no processo de agrupamento e, por este motivo, foram utilizados os mesmos valores. Foi utilizada vizinhança quadrada na busca dos padrões vizinhos.

Como em Handl *et al.* (2006), o raio de vizinhança inicial ficou definido como sendo igual a "1", com a utilização de incremento deste valor durante a fase inicial. Como não foi encontrada explicitamente, em outros trabalhos da literatura, uma equação para o aumento deste valor, isto foi feito segundo a equação (8), onde t é a iteração atual da fase inicial. Durante a fase final, este valor decresce em 0,05 a cada 100 substituições do padrão carregado por uma formiga (modificação sugerida e que é detalhada mais adiante). O valor do raio de vizinhança é sempre o valor inteiro menor ou igual ao definido em qualquer uma das fases. Esta adaptação automática, durante a fase final, tem a finalidade de "relaxar" o tamanho da vizinhança, quando as formigas não estão conseguindo descarregar os padrões que carregam.

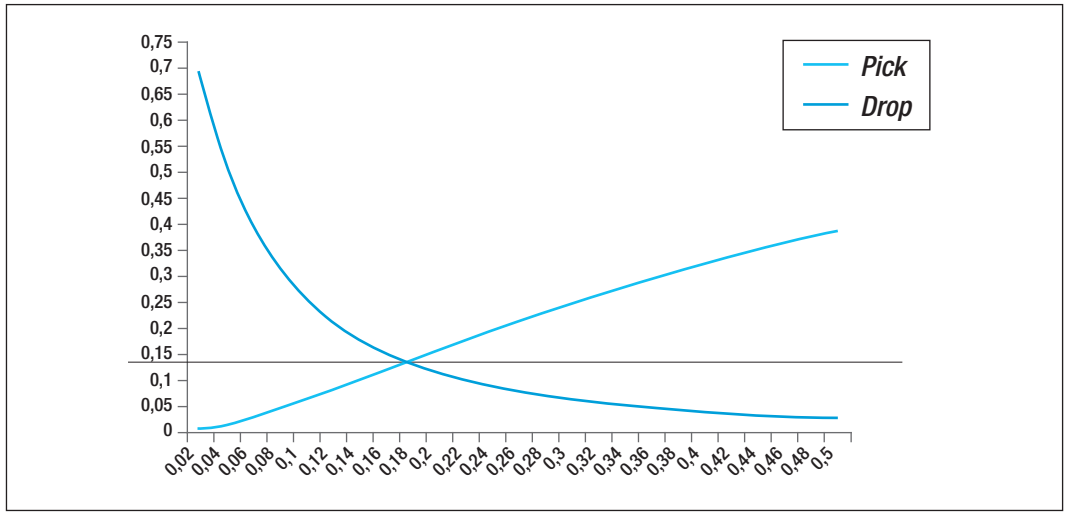
$$\sigma = \frac{t}{4^{t_{inicial}}} \tag{8}$$

Na definição da vizinhança para o cálculo da probabilidade de descarregar um padrão em sua posição atual e para o cálculo da probabilidade de carregar um padrão, considerou-se o raio de vizinhança sempre igual a "1". Na busca de uma nova posição, a direção do passo é aleatória. Definida a direção, calcula-se o tamanho máximo possível do passo. Um número aleatório pertencente ao intervalo [0, 1] foi utilizado para determinar este tamanho, multiplicando-se este número pelo tamanho máximo do passo.

As probabilidades de carregar (p_{pick}) e descarregar (p_{drop}) utilizadas, são as descritas pelas equações (1) e (2), respectivamente, onde $k_p = 0{,}1$ e $k_d = 0{,}3$, (mesmos valores adotados por Deneubourg *et al.*, 1991). Um padrão é carregado se a probabilidade p_p for maior que um valor mínimo para carregamento ($pick_{min}$). Um padrão é descarregado se a probabilidade p_d for maior que um valor mínimo para descarregamento ($drop_{min}$).

Os valores de $drop_{min}$ e $pick_{min}$ foram definidos como 0,13397, durante a fase inicial. Este valor ficou definido, fazendo a probabilidade de carregar (p_{pick}) igual à probabilidade de descarregar (p_{drop}). A Figura 1 mostra o gráfico das probabilidades de carregar e descarregar. A definição de um valor aleatório maior que 0,13397, durante a fase final, foi feita para restringir a mudança de posição durante esta fase, sem, no entanto, "engessar" o processo. A definição de um valor que aumentasse com o tempo, a longo prazo, impediria que as formigas movessem os padrões.

Figura 1 – Gráfico das probabilidades de carregar e descarregar padrões



Fonte: Os autores.

No cálculo da função *f*, foi utilizada a função *f**, definida pela equação (5), já apresentada, proposta por Handl *et al.* (2006), substituindo-se o parâmetro escalar $\frac{1}{\sigma^2}$ por $\frac{1}{N_{occ}}$, onde N_{occ} é o número de células da grade ocupadas, observadas dentro da vizinhança local, conforme apresentado em (9).

$$f^*(i) = \begin{cases} \frac{1}{N_{occ}} \sum_{j \in L} \left[1 - \frac{d(i,j)}{\alpha}\right], \text{ se } \forall j \left(1 - \frac{d(i,j)}{\alpha}\right) > 0 \\ 0 \qquad\qquad , \text{caso contrário} \end{cases} \tag{9}$$

O parâmetro α_0, depois de alguns testes preliminares, ficou definido como 0,8. A sua atualização, durante a fase inicial, ficou definida, conforme a equação (10); para a fase final, este valor decresce em 0,001 a cada 100 substituições do padrão carregado por uma formiga. Este decréscimo, durante a fase final, foi feito para evitar que as formigas não conseguissem mais mover seus padrões.

$$\alpha = \alpha_0 + \frac{2t}{p.t_{inicial}} - 0{,}01 \tag{10}$$

Observa-se que qualquer alteração nos valores de k_p, k_d e α influenciam diretamente o processo de agrupamento. Optou-se por manter os valores de k_p e k_d e utilizar somente uma adaptação para α. Se os valores de k_p e k_d forem alterados, a adaptação para α, bem como os valores $pick_{min}$ e $drop_{min}$, deverão ser revistos.

Quando um padrão é descarregado na grade, um novo padrão deverá ser carregado. A busca deste padrão é aleatória, porém, cada padrão livre é avaliado somente uma vez, até que todos sejam avaliados. Caso nenhum padrão apresente probabilidade p_{pick} maior que $pick_{min}$, o padrão que apresentar a maior probabilidade p_{pick} é carregado.

Quando um padrão não tem vizinhos, definiu-se a função f igual a zero. Isso faz com que a probabilidade p_d seja igual a "0", ou seja, o padrão não deve ser descarregado naquela posição e a probabilidade p_p foi igual a "1", ou seja, o dado deverá ser carregado e futuramente deixar esta posição.

A medida de dissimilaridade utilizada foi a distância Euclidiana. A matriz de distâncias foi calculada segundo a equação (11) e depois, foi padronizada. Nesta equação, o peso se refere ao atributo e é calculado dividindo-se o desvio-padrão pela média, calculado para cada atributo da matriz dos dados já padronizada (Q).

$$d(i,j) = \sum_{a=1}^{m} \left[(Q(a,i) - (Q(a,j) \,.\, \text{peso}(a,1) \right]^2 \tag{11}$$

Para a recuperação dos grupos, foi utilizado o Método de Ward e foi definido um número máximo de grupos. Já para a avaliação dos resultados, foram utilizados dois índices externos (Medida F e Índice Aleatório), conforme apresentado na seção 2.4 e o percentual de classificação errada.

3.2. Modificações propostas para o agrupamento baseado em Formigas

Durante o estudo do Agrupamento baseado em Formigas, conforme procedimento apresentado na seção 3.1 anterior, foi observado que muitas das mudanças de posição dos padrões ocorrem desnecessariamente. Considera-se uma mudança desnecessária, quando um padrão está entre similares na grade e, neste caso, não há necessidade da mudança deste padrão para outra posição. Com o objetivo de evitar estas mudanças desnecessárias, uma comparação da probabilidade de descarregar um padrão na posição escolhida aleatoriamente, com a probabilidade de descarregar este padrão em sua posição atual, foi introduzida. A decisão de descarregar um padrão na posição escolhida aleatoriamente só ocorre, se esta probabilidade for maior que a probabilidade de descarregar este padrão em sua posição atual.

Também foi observada a ocorrência de fusão de grupos próximos na grade. Quando a decisão de descarregar um padrão for positiva e a célula em que o padrão deveria ser descarregado, está ocupada, busca-se aleatoriamente uma posição vizinha a esta, que esteja livre. Porém, esta nova posição pode estar próxima também, a outro grupo de padrões na grade. Este pode ser um motivo para a fusão de grupos próximos. Como uma alternativa para evitar a fusão de grupos próximos na grade, foi proposta, neste trabalho, uma avaliação da probabilidade para a nova posição. O padrão só é descarregado na célula vizinha, se a probabilidade de descarregar o padrão, nesta posição, for maior que a probabilidade de descarregar este padrão em sua posição atual. Todas as posições vizinhas livres são avaliadas. Se em nenhuma posição vizinha livre a probabilidade de descarregar o padrão for maior que a probabilidade de descarregar este padrão em sua posição atual, o padrão não é descarregado e o processo se reinicia escolhendo-se outra formiga.

Outra questão observada no Agrupamento baseado em Formigas, é que uma formiga pode carregar um padrão que está entre similares na grade. Uma formiga só carrega um padrão, quando este não está entre similares na grade, porém, desde que a formiga carrega um padrão até ela ser sorteada para tentar descarregar o padrão, mudanças ocorrem na vizinhança deste, podendo deixá-lo então entre similares. Sendo assim, esta formiga fica inativa, pois a operação de descarregar o padrão não é executada. Neste caso, foi proposta a substituição do padrão carregado por uma formiga, caso este padrão não seja descarregado em 100 iterações consecutivas. O novo padrão foi escolhido por sorteio, mas ele só foi carregado pela formiga se a probabilidade de carregar este padrão for maior que 0,13397. O valor 0,13397 foi definido fazendo a probabilidade de carregar (p_{pick}) igual à probabilidade de descarregar (p_{drop}). Caso não exista nenhum padrão com probabilidade de carregar maior que 0,13397, o último padrão sorteado é carregado pela formiga. Este também, poderia ser um critério de parada.

3.3. Pseudo-código do algoritmo proposto

Considerando-se o algoritmo apresentado na seção 3.1 e as modificações propostas na seção 3.2, é apresentado, a seguir, um pseudo-código que reúne todas as ideias.

Fase inicial

Os padrões são aleatoriamente espalhados na grade.

Cada formiga escolhe aleatoriamente um padrão para carregar e é colocada em uma posição aleatória na grade.

Fase de distribuição

Cada formiga é selecionada aleatoriamente.

Esta formiga se desloca aleatoriamente na grade e avalia sua função de vizinhança $f(i)$ (equação 9).

A formiga decide probabilisticamente se descarrega seu padrão nesta posição (equação 2). O padrão só é descarregado na posição escolhida aleatoriamente, se esta probabilidade for maior que a probabilidade de descarregar este padrão em sua posição atual.

Se a decisão for negativa, escolhe-se aleatoriamente outra formiga e recomeça-se a fase de distribuição. Se a decisão for positiva, a formiga descarrega o padrão em sua posição atual na grade, se esta estiver livre.

Se esta célula da grade estiver ocupada, o mesmo deve ser descarregado numa célula vizinha desta, que esteja livre, por meio de uma busca aleatória. A avaliação da probabilidade de descarregar o padrão na nova posição é feita e o padrão só é descarregado na célula vizinha se a probabilidade de descarregar o padrão nesta posição continuar maior que a probabilidade de descarregar este padrão em sua posição atual. Se em nenhuma posição vizinha livre a probabilidade de descarregar o padrão for maior que a probabilidade de descarregar este padrão em sua posição atual, o padrão não é descarregado e o processo se reinicia, através da escolha de outra formiga.

A formiga procura aleatoriamente por um novo padrão para carregar (dentre os padrões livres), vai para a sua posição na grade, faz a avaliação da função de vizinhança (equação 9) e decide probabilisticamente se carrega este padrão (equação 1).

Este processo de escolha de um padrão livre na grade é executado até que a formiga encontre um padrão que deva ser carregado.

O padrão carregado por uma formiga será substituído, caso este padrão não seja descarregado em 100 iterações consecutivas. Outro padrão é escolhido aleatoriamente, mas ele só é carregado pela formiga se a probabilidade de carregar este padrão for maior que 0,13397, valor este discutido na seção 3.4. Caso não exista nenhum padrão com probabilidade de carregar maior que 0,13397, o último padrão sorteado é carregado pela formiga.

Fase de recuperação do agrupamento

O processo inicia com cada padrão formando um grupo.

Depois de calcular as distâncias entre todos os grupos, deve-se fundir (ligar) os dois grupos com menor distância.

4. ANÁLISE DE DADOS E RESULTADOS

O algoritmo de Agrupamento baseado em Formigas proposto foi aplicado às três bases de dados reais e públicas, apresentadas na Tabela 1. Por se tratar de uma metaheurística, este método foi aplicado a cada base de dados por 10 vezes, conforme já comentado.

4.1. Resultados da aplicação do algoritmo proposto às bases de dados

A Tabela 2 apresenta a média e o desvio-padrão das medidas de avaliação para as bases de dados, utilizando o algoritmo proposto, além das medidas de avaliação do agrupamento para o melhor resultado.

Como se pode observar, os resultados foram bastante satisfatórios para as bases de dados IRIS e WINE (11,9% e 12,7%, em média de classificações erradas). Já para a base de dados PIMA, os resultados não foram tão bons; mais adiante, se mostra que os outros métodos também, não apresentaram resultados satisfatórios para esta base de dados.

Tabela 2 – Resultados da aplicação do algoritmo proposto, médias da execução de 10 vezes, para as bases de dados reais (IRIS, WINE e PIMA).

Resultados		R	F	Classificação errada (%)
IRIS	Média	0,871	0,877	11,9
	Desvio-padrão	0,039	0,050	4,6
	Melhor resultado	0,927	0,940	6,0
WINE	Média	0,843	0,871	12,7
	Desvio-padrão	0,019	0,021	1,9
	Melhor resultado	0,871	0,899	10,1
PIMA	Média	0,510	0,583	43,6
	Desvio-padrão	0,010	0,022	4,0
	Melhor resultado	0,531	0,623	37,5

Fonte: Os autores.

A Tabela 3 (matriz de confusão) mostra a distribuição dos padrões para a base de dados IRIS, onde se pode observar os padrões atribuídos aos grupos corretamente e os padrões atribuídos aos grupos erroneamente. Nesta base de dados, são apenas nove padrões (*) em grupos errados, de um total de 150 padrões. O grupo 1 contém todos os padrões atribuídos a ele.

Tabela 3 – Matriz de Confusão apresentando a distribuição dos Padrões para a base de dados IRIS – melhor resultado.

IRIS	Solução Gerada		
Agrupamento Correto	Grupo 1	Grupo 2	Grupo 3
Classe 1	50	0	0
Classe 2	0	48	2*
Classe 3	0	7*	43

Fonte: Os autores.

Analogamente, a Tabela 4 mostra a distribuição dos padrões para a base de dados WINE. Nesta base de dados, são 18 padrões (*) em grupos errados, de um total de 178 padrões.

Tabela 4 – Matriz de Confusão apresentando a distribuição dos Padrões para a base de dados WINE – melhor resultado.

WINE	Solução Gerada		
Agrupamento Correto	Grupo 1	Grupo 2	Grupo 3
Classe 1	55	4*	0
Classe 2	4*	64	3*
Classe 3	2*	5*	41

Fonte: Os autores.

4.2. Avaliação do algoritmo proposto em relação aos métodos de Ward, de Kohonen Unidimensional e ACAM

Na Tabela 5 são apresentadas as comparações das medidas médias de avaliação para os três métodos (algoritmo proposto, Ward e Kohonen), para as bases de dados IRIS, WINE e PIMA. Os melhores resultados encontram-se em negrito.

Na base de dados IRIS, o Método de Ward foi melhor para as três medidas de avaliação (cerca de 3% de erros); na base de dados WINE, o algoritmo proposto foi melhor para duas das três medidas de avaliação (cerca de 12% de erros) e na base de dados PIMA, a técnica de Redes de Kohonen Unidimensional foi melhor para duas das três medidas de avaliação (cerca de 34% de erros).

Vale salientar que Handl *et al*. (2006) também afirmam que nenhum algoritmo domina os outros sempre. Segundo Ho e Pepune (2002), pelo teorema "*NO-FREE-LUNCH*", se não há nenhuma suposição anterior sobre o problema de otimização que se tenta resolver, é de se esperar que nenhuma estratégia tenha melhor desempenho que outra, quando testada em um conjunto grande de bases de dados com características diversas.

Na comparação das médias das medidas de avaliação do agrupamento para o algoritmo proposto e para o algoritmo *ACAM*, os resultados mostram que o algoritmo proposto é melhor que o *ACAM* para duas das três bases de dados (IRIS E WINE). Os melhores resultados são apresentados com (+). Como já comentado, para a base de dados PIMA, os resultados mostram que nenhum dos métodos apresentou resultado satisfatório.

Tabela 5 – Comparação dos resultados médios da aplicação dos métodos de agrupamento Ward, Redes de Kohonen Unidimensional, ACAM e Algoritmo Proposto para as bases de dados IRIS, WINE e PIMA.

Bases de dados		Ward	1D-SOM	Formiga	
				Proposto	*ACAM*
IRIS	**R (quanto maior melhor)**	**0,96**	0,86	0,87+	0,82
	F (quanto maior melhor)	**0,97**	0,86	0,88+	0,81
	Classificação errada (%) (quanto maior melhor)	**3,33**	12,8	11,9+	18,7
WINE	**R (quanto maior melhor)**	0,82	0,76	0,843	**0,85+**
	F (quanto maior melhor)	0,85	0,76	**0,87+**	0,87
	Classificação errada (%) (quanto maior melhor)	15,169	22,42	**12,7+**	13,9
PIMA	**R (quanto maior melhor)**	0,53	**0,55**	0,51	0,52+
	F (quanto maior melhor)	0,62	**0,66**	0,58+	0,57
	Classificação errada (%) (quanto maior melhor)	37,370	34,57	43,6	**33,7+**

Fonte: Os autores.

5. CONSIDERAÇÕES FINAIS

O algoritmo proposto para o Agrupamento baseado em Formigas foi aplicado em três bases de dados e, para avaliação do seu desempenho, foi comparado com outros três métodos: Ward, Kohonen unidimensional e *ACAM*.

Comparando-se o algoritmo proposto aos métodos de Ward e de Kohonen (colunas 2, 3 e 4 da tabela 5), os resultados não mostram superioridade de algum deles. Já na comparação das médias das medidas de avaliação do agrupamento, através do algoritmo proposto e do algoritmo *ACAM* (colunas 4 e 5 da tabela 5), os resultados mostram que o algoritmo proposto apresenta um desempenho melhor para duas das três bases de dados.

Tem-se, assim, que embora o algoritmo proposto não tenha apresentado superioridade, em relação aos métodos já consagrados de Ward e de Kohonen, apresentou melhorias em relação a uma das últimas abordagens envolvendo agrupamento por colônia de formigas (*ACAM*, de BORYCZKA, 2009) certificando, desta forma, a importância do presente artigo.

Pretende-se dar prosseguimento a este trabalho, utilizando-se bases de dados adicionais para testes, bem como a utilização de índices adicionais para a avaliação do agrupamento para, então, propor-se melhorias adicionais ao algoritmo aqui proposto.

6. AGRADECIMENTOS

À FINEP, pelo apoio financeiro ao projeto de pesquisa CT – INFRA / UFPR / Modelagem e Computação Científica e à CAPES, pela bolsa concedida à primeira autora.

7. REFERÊNCIAS BIBLIOGRÁFICAS

BORYCZKA, U. Finding groups in data: Cluster analysis with ants. **Applied Soft Computing**, v. 9, p. 61-70, 2009.

DENEUBOURG, J. L.; GOSS, S.; FRANKS, N.; SENDOVA-FRANKS, A.; DETRAIN, C.; CHRÉTIEN, L. The dynamics of collective sorting: Robot-like ants and ant-like robots. *In*: **Proceedings of the First International Conference on Simulation of Adaptive Behaviour**: From Animals to Animats 1 (pp. 356–365). Cambridge, MA: MIT Press, 1991.

DORIGO, M.; BLUM, C. Ant colony optimization theory: A survey. **Theoritical Computer Science**, v. 344, p. 243-278, 2005.

DORIGO, M.; CARO, G. D.; GAMBARDELLA L. M. Ant algorithms for discrete optimization. **Artificial Life**, v. 5, p. 137-172, Belgium, 1999.

DORIGO, M.; MANIEZZO, V.; COLORNI, A. Ant System: Optimization by a colony of cooperating agents. **IEEE Transactions on Systems, Man, and Cybernetics** – Part B, v. 26, n. 1, p. 1–26, 1996.

DORIGO, M.; STÜTZLE, T. **Ant colony optimization**. Cambridge: MIT Press, 2004.

FAUSETT, L. **Fundamentals of Neural Networks – Architectures, Algorithms, and Applications**. New Jersey: Prentice Hall, 1994.

HANDL, J.; KNOWLES, J.; DORIGO, M. Ant-Based Clustering and Topographic Mapping. **Artificial Life**, v. 12, n. 1, p. 35-61, 2006.

HO, Y. C.; PEPUNE, D. L. Simple Explanation of the No-Free-Lunch Theorem and Its Implications. **Journal of Optimization Theory and Applications**, v. 115, n. 3, p. 549-570, 2002.

JOHNSON, R. A.; WICHERN, D. W. **Applied Multivariate Statistical Analysis**. Fourth Edition. New Jersey: Prentice Hall, 1998.

MATLAB R2008b – The MatWorks, MATLAB (R2008b), The MathWorks Inc., Natick, 2008.

TAN, P. N.; STEINBACH, M.; KUMAR, V. **Introduction to Data Mining**. Inc. Boston, MA, USA: Addison-Wesley Longman Publishing Co., 2005.

TAN, S. C.; TING, K. M.; TENG, S. W. Examining Dissimilarity Scaling in Ant Colony Approaches to Data Clustering. *In*: ACAL, 2007. ACAL 2007. Springer-Verlag, 2007.

VILLWOCK, R. Técnicas de Agrupamento e de Hierarquização no Contexto de Kdd – **Aplicação a Dados Temporais de Instrumentação Geotécnica-Estrutural da Usina Hidrelétrica de Itaipu**. 125 f. Tese (Doutorado em Métodos Numéricos em Engenharia) – Setor de Ciências Exatas, Universidade Federal do Paraná, Curitiba, 2009.

VIZINE, A. L.; DE CASTRO, L. N.; HRUSCHKA, E. R.; GUDWIN, R. R. Towards improving clustering ants: an adaptive ant clustering algorithm. **Informatica**, v. 29, p. 143–154, 2005.