

Algoritmo DL3: Uma abordagem de Clustering baseado em Auto-organização

Lutiele M. Godois¹, Luciano C. Marco¹, Diana F. Adamatti¹, Leonardo R. Emmendorfer¹

¹ Programa de Pós-Graduação em Modelagem Computacional - Centro de Ciências Computacionais - C3 – Universidade Federal do Rio Grande (FURG)
Caixa Postal 474 – 96.201-900 – Rio Grande – RS – Brazil

lutielegodois@furg.br, lucianomarco@msn.com,
leonardo.emmendorfer@gmail.com, dianaada@gmail.com

Abstract. *Clusters are considered as groups containing data objects that are similar to each other. Many clustering techniques have been proposed, most of them rely on a priori information to obtain the results, such as the desired number of clusters. This paper presents a clustering algorithm based on Multi-agent Systems, which automatically detects the clusters for a given dataset, which are emergent patterns from the interaction among agents. Results illustrate the convergence of the algorithm for a range of space dimensions.*

Resumo. *Clusters são considerados como grupos contendo objetos de dados que são semelhantes entre si. Muitas técnicas de Clustering foram propostas, a maioria delas se baseia em informações a priori para obter os resultados, como o número desejado de clusters. Assim, este artigo apresenta um algoritmo de agrupamento baseado em sistemas multiagentes, que detecta automaticamente os clusters para um determinado conjunto de dados, que são padrões emergentes da interação entre agentes. Os resultados ilustram a convergência do algoritmo para uma gama de dimensões espaciais.*

1. Introdução

Clustering tornou-se um tema cada vez mais importante nos últimos anos, devido ao grande número de dados utilizados nas diversas áreas: biologia, medicina, psicologia, processamento de imagens, etc. Os Algoritmos de Clustering particionam objetos de dados (padrões, entidades, instâncias, observações, unidades) em certo número de clusters (grupos, subconjuntos ou categorias) [Xu e Wunsch II 2009]. Devido a esse grande número de aplicações, dificilmente se encontrará uma técnica de clustering genérica e aplicável de forma satisfatória a todos os tipos de dados nos diferentes contextos. Logo, existem muitos algoritmos para este fim, entre os mais populares estão: K-means [MacQueen 1967], DBSCAN [Ester et al. 1996] e CURE [Guha, Rastogi and Shim 1998]. Apresenta-se nesse trabalho alguns resultados de uma nova proposta de algoritmo de clustering, baseados em Sistemas Multiagentes.

Os Sistemas Multiagentes são compostos por múltiplos elementos computacionais que se interagem, conhecidos como agentes, eles por sua vez, possuem a capacidade de ação autônoma, de decidir por si mesmos o que precisam fazer para satisfazer seus objetivos e a capacidade de interagir com outros agentes [Wooldridge 2002]. Assim, é justamente essa possibilidade de interação, cooperação e troca de

informações entre agentes e principalmente de autonomia que a técnica de clustering apresentada aqui se baseia em Sistemas Multiagentes, com o intuito de desenvolver um algoritmo capaz de realizar a auto-organização de diferentes conjuntos de objetos de dados. Nesse algoritmo proposto cada agente (objeto de dado) realiza certa tarefa com alguns critérios definidos para formar grupos com outros agentes, baseado apenas em sua localização espacial e sua capacidade de “enxergar” seus vizinhos. O critério principal utilizado é que o agente escolha o grupo com o maior número de agentes para se agrupar de acordo com um raio de percepção definido. Desta forma, o objetivo é a implementação do algoritmo e os testes foram realizados na plataforma NetLogo.

Esse artigo está estruturado da seguinte maneira: na Seção 2 apresentam-se alguns trabalhos relacionados com o tema do trabalho; na Seção 3 são apresentados os algoritmo de agrupamento proposto e a metodologia utilizada no processo de implementação; a Seção 4 apresenta os resultados alcançados; e por fim na Seção 5 estão as conclusões do trabalho.

2. Trabalhos Relacionados

Provavelmente, agrupamento de dados é um dos problemas mais amplamente estudados pelas comunidades de mineração de dados e aprendizagem de máquina por causa de suas numerosas aplicações em diversas áreas: biologia, medicina, psicologia, processamento de imagens, etc. O problema básico de agrupamento de dados pode ser afirmado da seguinte forma: Dado um conjunto de pontos de dados, divida-os em um conjunto de grupos que são tão semelhantes quanto possível. Porém essa definição de problema pode sofrer algumas alterações dependendo do modelo específico utilizado e também o tipo de dado. Por exemplo, um modelo generativo pode definir similaridade com base em um mecanismo generativo probabilístico, enquanto que uma abordagem baseada em distância utilizará uma função de distância tradicional para quantificação [Aggarwal e Reddy 2014].

Na literatura são encontrados muitos métodos para mostrar a homogeneidade e heterogeneidade entre grupos de dados. Talvez o mais tradicional de todos seja o K-means. Esse método consiste em particionar um conjunto de objetos de dados em k grupos, definido a priori, além disso, é definido um centroide para cada cluster. É a partir desse centroide que se buscam os objetos mais próximos a ele, obtendo-se assim os clusters. A atualização desses centros é realizada sempre que um novo objeto é atribuído ao cluster, sendo o critério de parada desse algoritmo quando não há mais alterações. As escolhas iniciais de número de grupos e centroides se constituem com uma das desvantagens desse algoritmo, pois o agrupamento final fica condicionado a esses critérios.

Contrário a propostas de informar muitos dados de entradas ou de adicionar elementos que forcem a quantidade de grupos a serem formados, se propõe neste trabalho um algoritmo de agrupamento que possui como principal característica a auto-organização de objetos em grupos segundo o critério determinado para este fim. Além disso, o algoritmo também se embasa na concepção de Sistemas de Multiagentes, onde cada objeto de dado é compreendido com um agente que deve realizar uma tarefa para se organizar em grupos com os outros agentes que pertencem ao conjunto.

Algoritmos de Agrupamentos baseados em Sistemas Multiagentes já são temas de muitos estudos. Santos e Bazan (2012) propõe em seu trabalho um algoritmo de

agrupamento inspirado em técnicas de inteligência de enxames, como a organização de colônias de abelhas e a alocação de tarefas entre insetos sociais, partindo também do pressuposto contrário das técnicas tradicionais que necessitam de algumas orientações para a alocação de dados em grupos. O processo de tomada de decisão dos agentes é inspirado no trabalho de Agogino e Tumer (2006). Esses autores, por sua vez, apresentam uma alternativa quando se deseja trabalhar com numerosos domínios. Nesta abordagem, para cada agente é atribuído um pequeno subconjunto dos dados e esses dados escolhem em qual grupo final seus pontos de dados devem pertencer. O agrupamento final é então avaliado por uma função utilitária global, calculada de forma distribuída. Os resultados obtidos nesse trabalho mostram que essa metodologia específica para agente pode obter um melhor desempenho do que os métodos tradicionais não baseados em agentes.

Já Takacs e Demiris (2010) descrevem um novo algoritmo que utiliza subdivisões iterativas onde os clusters são pré-processados em cada etapa para combater o dimensionamento espacial, a rotação, a velocidade de repetição e a frequência de amostragem variável. Além disso, apresentam um método para equilibrar a segmentação espacial e temporal com base no tamanho de grupo esperado, e uma medida de validade é introduzida para determinar o número ótimo de clusters. A aplicação foi realizada em jogos e os resultados comparados com o K-means e o tradicional Clustering Espectral.

Chaimontree, Atkinson e Coenen (2012) descrevem um framework para agrupamento baseado em multi-agentes onde cada agente individual representa um cluster individual. Uma característica particular do framework é que, após a criação de uma configuração de cluster inicial, os agentes podem negociar com o objetivo de melhorar esse cluster inicial. O framework pode ser usado no contexto de um número de paradigmas de agrupamento, dois são investigados: K-means e KNN. A avaliação relatada demonstra que a negociação pode servir para melhorar uma configuração de cluster inicial.

Gueleri (2013) desenvolveu técnicas de agrupamento baseadas em comportamento coletivo e auto-organização. Nessas técnicas cada objeto do conjunto de dados corresponde a um indivíduo do sistema, os quais interagem um com os outros, de forma que os grupos apareçam a partir de uma organização promovida por próprios. Os objetos são mantidos fixos em seu espaço de atributos, mas carregam certo tipo de "energia". Tal energia será trocada gradualmente entre eles. Logo os grupos serão formados por objetos que possuem energias semelhantes.

Os exemplos anteriores expressam a eficácia de trabalhar com Sistemas Multiagentes quando se deseja agrupar objetos de dados ou quando se faz necessário a utilização de algoritmos de clustering para agrupar agentes. A proposta apresentada neste artigo pretende salientar ainda mais essa importância, dando ênfase nas propriedades que os agentes possuem, como a autonomia e a capacidade de reação. Nas próprias seções são expostas as etapas e como ocorre a interação do agente no processo de agrupamento.

3. Algoritmo proposto e procedimentos metodológicos

Nessa seção é detalhado o algoritmo, conforme apresentado no pseudocódigo representado na figura 1, o qual é denominado como DL3. Para realizar uma análise experimental, implementou-se uma versão do algoritmo utilizando o software NetLogo.

```
Entrada: Um conjunto de objetos de dados X e um raio de visão r
1  $g_i$  conta qual setor circular possui mais objetos
2  $M_a$  é o  $g_i$  com mais objetos
Saída: Uma partição K de X em k grupos
3 início
4   repita
5     para cada  $x_i \in X$  faça
6       Contar a quantidade de objetos em cada setor ( $g_i$ )
7       Analisar qual setor possui mais objetos ( $M_a$ )
8       if  $M_a \neq 0$  then
9         | Mover  $x_i$  na direção do maior  $g_i$ 
10      end
11      if  $M_a == 0$  then
12        | Escolher caminho aleatório
13      end
14    fim
15  até Até não houver mais mudanças nos clusters;
16 fim
17 retorna K
```

Figura 1. Pseudo-código do Algoritmo Proposto

O algoritmo proposto é iniciado com a entrada de um conjunto de objetos de dados X qualquer e de um raio de visão r . Apesar de ainda ter que fornecer informações a priori, o DL3 ainda se destaca dos demais quando não se faz necessário informar o número de grupos antes da execução do algoritmo. O número de grupos será formado automaticamente durante o processo de agrupamento, de acordo com o raio de visão pré-estabelecido. Os g_i são os valores obtidos nas oito direções de um único objeto, ou seja, todos os objetos tem uma visão de 360° graus dividido de 45° graus em 45° graus e para cada uma dessas direções é contado o número de objetos que cada objeto enxerga, o maior desses valores é chamado de M_a . Este valor de M_a é quem determina a direção para onde cada ponto deve seguir salvo quando este número for empate ou nulo. Assim, o objeto escolhe uma direção de um dos g_i iguais em caso de empate ou qualquer direção quando for nulo.

3.1. Fase Inicial

Primeiramente o usuário informa o número de pontos que são aleatoriamente espalhados no *grid* e um raio de visão. Cada ponto possui um rótulo distinto e característico (cor). Nas simulações, cada agente é um ponto que representa um objeto a ser agrupado.

3.2. Fase de Agrupamento

Cada ponto enxerga em um ângulo de 45° com o raio de visão, definida pelo usuário. O objetivo principal do algoritmo é agrupar cada ponto ao grupo que possuir o maior

número de pontos em um ângulo de 360° em relação ao seu raio de visão, adquirindo assim um novo rótulo característico do grupo onde se agrupou. Caso dois ou mais grupos possuam o mesmo número de pontos, o ponto optará por um dos grupos de forma aleatória. Essa propriedade de aleatoriedade também vale quando o ponto não avistar nenhum grupo. Dessa forma, ele escolhe uma direção e anda um passo de maneira que possa encontrar novos grupos em sua volta.

3.3. Critério de Parada

O critério de parada do algoritmo é estabilidade dos agrupamentos, ou seja, verificar quando o número de grupos não se modifica significativamente em relação ao raio de visão adotado. Para este fim, se criou uma lista a partir da movimentação dos objetos, cada vez que um deles se movimenta, que não seja aleatoriamente, o objeto assume a cor de outro ponto e sua identificação é colocada em uma lista, para cada tick (iteração do NetLogo) essa lista é atualizada, removendo-se os objetos repetidos e comparado com o número de objetos determinado no início do procedimento. Se esse número de pontos da lista for maior ou igual ao número de objetos o processo é finalizado.

Logo, foram realizadas várias simulações com diferentes números de objetos em espaços de tamanhos distintos, com o intuito de observar o comportamento do algoritmo testado. Os resultados das simulações serão discutidos de melhor forma na seção a seguir.

4. Resultados

A linguagem de programação do NetLogo se consolidou como uma boa ferramenta para as simulações do algoritmo proposto, propiciando assim a visualização do comportamento dos agentes e as trocas de informações entre eles no decorrer das iterações, com um baixo custo de memória, que gerou uma economia de tempo e rapidez para verificar a existência de uma convergência. A comunicação dos agentes é realizada de forma visual, como apresentado anteriormente na descrição do algoritmo. Vale salientar também que um dos fatores positivos do algoritmo DL3 é a não necessidade de fornecer muitas informações a priori como, por exemplo, a escolha inicial dos centroides que representará cada cluster como no caso do popular *K-means*. Logo, a quantidade de grupos será definida no decorrer da execução do algoritmo, ou seja, não existe a necessidade de informar o número como um dado de entrada.

No algoritmo proposto nesse trabalho, cada ponto escolhe avançar em direção do grupo que no seu raio de visão possui mais pontos, quando não são perceptíveis esses grupos, a direção do passo é aleatória para que em iterações futuras possa se unir com algum grupo. Sobre o tamanho de passo que cada agente anda no processo optou-se que todos avançam um passo, pois valores maiores podem acelerar o progresso do agrupamento, podendo gerar resultados não esperados. Contudo, esse aspecto ainda precisa ser melhor analisado.

Dessa forma, para fins de simulação foram utilizados grids de tamanhos 10x10, 20x20, 50x50 e 100x100, com 10%, 50% e 100% do seu espaço ocupado por pontos, gerando assim 12 cenários diferentes. A Figura 2 mostra um exemplo de simulação realizada: nesse caso temos 50 objetos espalhados em um grid de tamanho 10x10 com

raio de visão igual a 4 (Figura 2 – I). No final da execução obteve-se 4 grupos distintos (Figura 2 – II).

Considere a Figura 3, ela mostra os gráficos provenientes dos resultados das simulações para os grids de menor tamanho (10x10 e 20x20) e com valores absolutos apresentados nas Tabelas de 1 a 2, sendo que foram realizadas dez rodadas de simulação para cada cenário proposto (percentual de objetos nos diferentes grids e com raios de visão de tamanhos distintos) e calculada a média aritmética para cada caso. Os resultados obtidos são semelhantes para ambos os grids na comparação das porcentagens de ocupação. Para 10% de ocupação se obteve a diminuição do número de grupos formados com o aumento da visão e a diminuição do tempo (ticks) nos primeiros aumentos de raio de visão, o que leva os objetos a se agruparem mais rápido, porém a partir de uma visão de tamanho aproximado 6 voltam a aumentar o tempo para finalizar o processo de clustering. Já para os outros cenários, 50% e 100 de ocupação, se observa também um comportamento análogo com o crescimento do número de ticks e decréscimo do número de grupos formados com o aumento da visão.

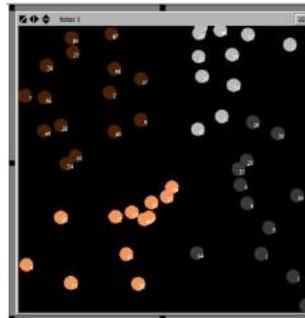
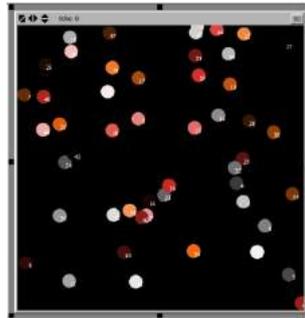
A Figura 4 exibe os gráficos provenientes dos resultados das simulações para os grids de maior tamanho (50x50 e 100x100) e com valores absolutos apresentados nas Tabelas de 2 a 3, sendo também realizadas dez rodadas de simulação para cada cenário proposto (percentual de objetos nos diferentes grids e com raios de visão de tamanhos distintos) e calculada a média aritmética para cada caso. Pode-se perceber pelos resultados obtidos que quanto mais objetos espalhados no grid, maior o tempo para a finalização do processo, independente do raio de visão adotado. Além disso, conjuntos com muitos objetos convergem para um único grupo com raio de visão menor, se comparados com conjuntos com menos objetos.

Ainda sobre as conclusões observadas nas Figuras 4 e 5 e Tabelas 1, 2, 3 e 4 é identificada a existência da convergência do número de grupos no decorrer do aumento do raio de visão. De tal modo que um espaço muito ocupado os seus objetos tendem a se agrupar em único grupo com um raio de visão menor se o mesmo for comparado com o raio que se obtém o mesmo efeito quando o espaço estiver menos ocupado. Logo o raio de visão se caracteriza como um fator muito importante que determinará um resultado satisfatório ou não do agrupamento, dessa forma novas análises devem ser feitas com o intuito da formalização desse parâmetro fundamental para a validação do algoritmo de clustering proposto nesse trabalho.

A Figura 5, já ilustra um resultado interessante sobre esse assunto, mostrando a média do número de grupos e ticks para 10% (Figura 4 – I), 50% (Figura 4 – II) e 100% (Figura 4 – III) de pontos, onde pode-se analisar que nos três gráficos quando o campo de visão está próximo de 10 é onde tem-se maior impacto, pois não importando o tamanho do grid nem o número de pontos que é adicionado a este, verifica-se que todas as médias estão próximas. Desta forma, pode-se aconselhar que seja adotado um raio de visão entre 1 e 10 em todos os casos, quando se pretende dividir o conjunto de objetos entre mais de um grupo.

Conjunto de objetos considerado

Agrupamento Final



I

II

Figura 2. Exemplo de simulação realizada. Considerando um grid 10x10, 50% do seu espaço ocupado e raio de visão 4.

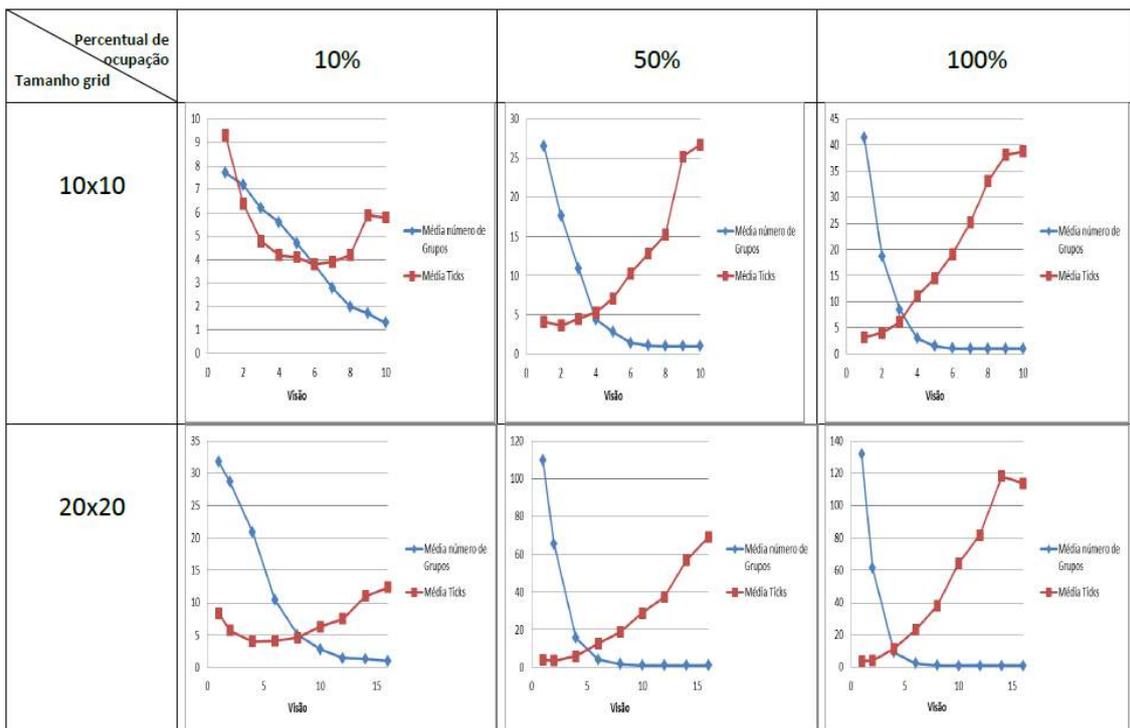


Figura 3. Gráficos do comportamento para os grids 10x10 e 20x20

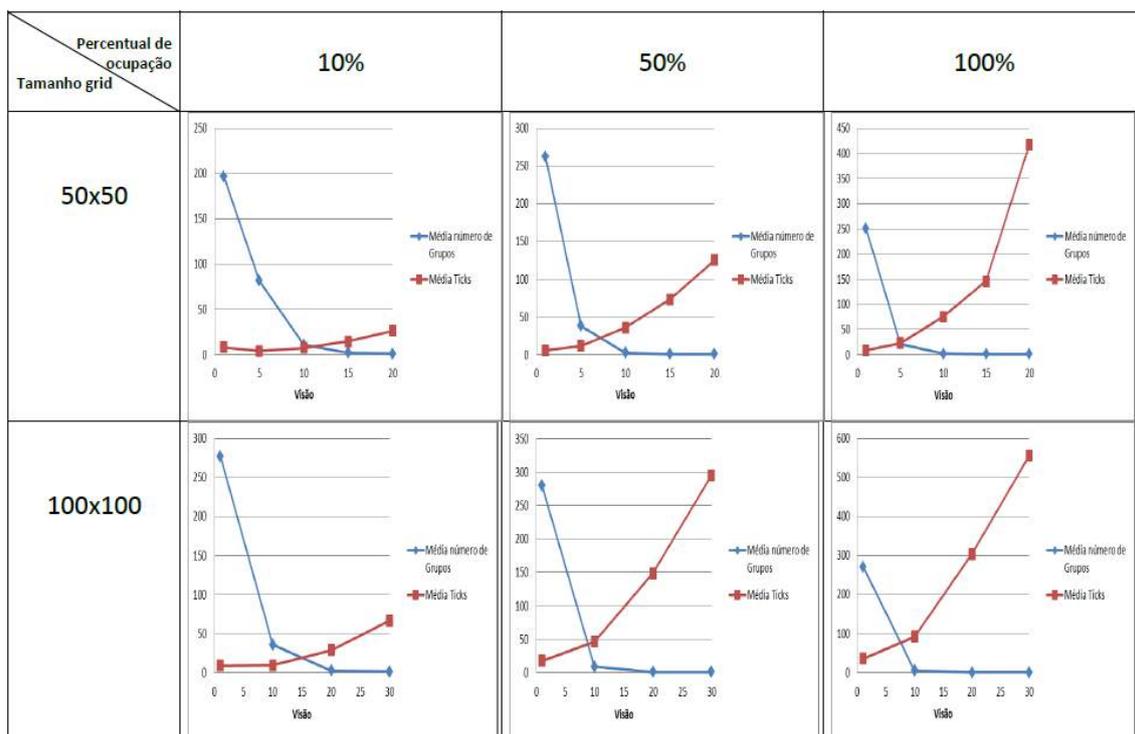


Figura 4. Gráficos do comportamento para os grids 50x50 e 100x100

Tabela 1. Valores Absolutos Grid 10x10 (MG e MT representam respectivamente a média do número de grupos e média do número de ticks)

Visão	MG 10%	MG 50%	MG 100%	MT 10%	MT 50%	MT 100%
1	7,7	26,5	41,4	9,3	4,1	3,2
2	7,2	17,6	18,7	6,4	3,6	4
3	6,2	10,9	8,5	4,8	4,5	6,1
4	5,6	4,4	3	4,2	5,3	11,1
5	4,7	2,8	1,5	4,1	7,1	14,5
6	3,8	1,4	1,1	3,8	10,3	19,1
7	2,8	1,1	1	3,9	12,8	25,2
8	2	1	1	4,2	15,3	33,1
9	1,7	1	1	5,9	25,2	38,1
10	1,3	1	1	5,8	26,7	38,8

Tabela 2. Valores Absolutos Grid 20x20 (MG e MT representam respectivamente a média do número de grupos e média do número de ticks)

Visão	MG 10%	MG 50%	MG 100%	MT 10%	MT 50%	MT 100%
1	31,8	109,7	131,7	8,4	4,1	3,9
2	28,7	65,5	61,4	5,7	3,5	4,1
4	20,9	15,6	9,3	4	6	11,5
6	10,4	4,1	2,4	4,1	12,7	23,2
8	5	1,8	1,2	4,6	18,9	38

10	2,8	1	1	6,3	28,8	64,3
12	1,4	1	1	7,5	37,4	81,9
14	1,3	1	1	11	56,8	118,2
16	1	1	1	12,4	69,2	113,6

Tabela 3. Valores Absolutos Grid 50x50 (MG e MT representam respectivamente a média do número de grupos e média do número de ticks)

Visão	MG 10%	MG 50%	MG 100%	MT 10%	MT 50%	MT 100%
1	196,9	262,2	251	8,3	6	9
5	81,5	37,9	21,3	4	11,5	23,2
10	10,7	2,7	1,8	7,3	35,8	75,7
15	2,2	1	1	14,8	72,8	146,4
20	1	1	1	26,4	125	417

Tabela 4. Valores Absolutos Grid 100x100 (MG e MT representam respectivamente a média do número de grupos e média do número de ticks)

Visão	MG 10%	MG 50%	MG 100%	MT 10%	MT 50%	MT 100%
1	277,1	279,3	271	9,1	18	36
10	35,7	8,7	5,4	9,2	46,4	92
20	2,7	1	1	28,9	148,7	303,8
30	1,1	1	1	66,5	295	556

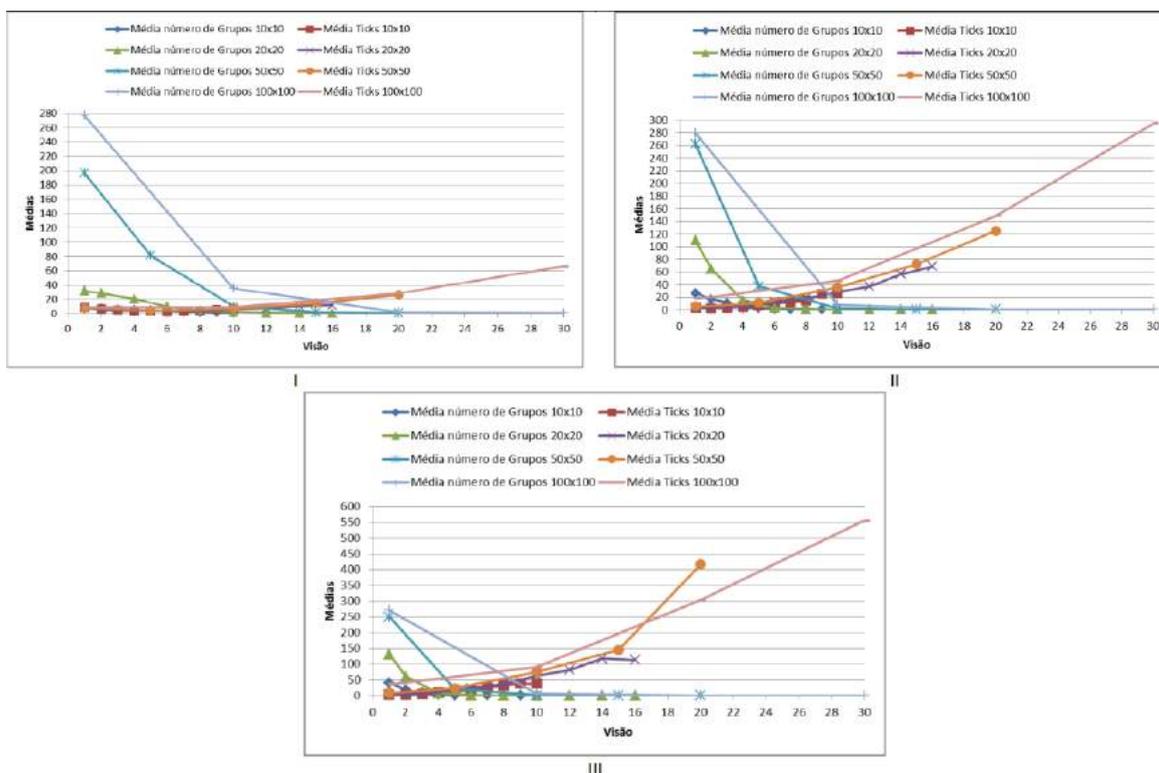


Figura 5. Gráficos da Média de número de Grupos e Ticks para 10% (I), 50% (II) e 100% (III) dos grids ocupados.

Por fim, após as simulações, é possível afirmar que os agrupamentos gerados pelo algoritmo testado são fortemente dependentes da distribuição dos pontos no grid, do tamanho escolhido para o grid e do raio de percepção. De fato, poucos pontos em um grid de tamanho muito grande, pode gerar, de acordo com a sua disposição, muitos grupos, quando se considera um raio de percepção muito pequeno. Caso o contrário, a escolha de um valor muito grande para o raio, pode ter como resultado poucos grupos. Logo, um valor impróprio irá refletir em uma atividade inadequada no grid.

5. Conclusões

Com os resultados obtidos, principalmente aos que se referem a aleatoriedade, pode-se afirmar que o algoritmo DL3 poderá servir para diversos problemas, não se limitando a uma única questão ou solução, sendo essa uma de suas principais vantagens. A proposta apresenta aqui é de um algoritmo genérico, que apenas analisa a localização espacial entre os objetos. Todavia, pode-se incluir nas características dos agentes outras variáveis, que podem influenciar na tomada de decisão de fazer ou não parte de um grupo. Além disso, DL3 não necessita da definição de centroides aos grupos formados, pois eles se auto-organizam, formando seus próprios grupos sem informação inicial de quantos se deseja. Assim, o algoritmo cumpre algumas propriedades de agentes, através da autonomia que colabora com a tomada de decisões durante o processo de agrupamento.

Porém, o algoritmo proposto ainda é muito dependente de certos parâmetros pré-definidos em sua execução, principalmente a definição do raio de visão. Logo, necessita ainda de novos estudos, análises e propostas para aperfeiçoar seu desempenho e estabilidade e formalização do melhor raio de visão a ser adotado, com a finalidade de fazer o algoritmo uma boa ferramenta para mineração de dados. Dessa forma, as próximas etapas do trabalho consistem na avaliação ampla do algoritmo utilizando diferentes índices de validação encontrados na literatura que se apropriam de conceitos de grafos [Pal e Biswas 1997] e densidade [Moulavi et al. 2014], além da comparação com outros métodos de agrupamento como o K-means.

6. Agradecimentos

Os autores agradecem especialmente a Capes (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) pelo apoio financeiro colaborando assim para a realização da pesquisa.

Referências

Aggarwal, C. C. and Reddy, C. K. (2014) “Data Clustering: Algorithms and Applications”. CRC Press, NY, USA.

- Agogino, A. and Tumer, K. (2006) “Efficient agent-based cluster ensembles”. In Proc. 5th Int. Conf. on Autonomous agents and multiagent systems. AAMAS '06, ACM, New York, , p. 1079–1086.
- Chaimontree, S.; Atkinson, K. and Coenen F. (2012) “A Multi-Agent Based Approach To Clustering: Harnessing The Power of Agents”. In Agents and Data Mining Interaction, Vol. 7103 of the series Lecture Notes in Computer Science, p. 16-29.
- Ester, M.; Kriegel, H.-P.; Sander, J.; XU, X. (1996) “A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise.” In KDD-96 Proceedings, p.226–231.
- Gueleri, R. A.; (2013) “Agrupamento de dados baseado em comportamento coletivo e auto-organização”. Programa de Pós-Graduação em Ciências de Computação e Matemática Computacional - Mestrado em Ciências da Computação e Matemática Computacional — Universidade de São Paulo, São Carlos.
- Guha, S.; Rastogi, R.; Shim, K. (1998) “CURE: An Efficient Clustering Algorithm for Large Databases”. In SIGMOD, Vol.27, p.73–84.
- MacQueen, J. (1967). “Some methods for classification and analysis of multivariate observations”. In *Proceedings of the Fifth Berkeley Symposium*, Vol. 1, p. 281-297.
- Moulavi, D.; Jaskowiak, P. A.; Campello, R. J. G. B.; Zimek, A.; Sander, J. (2014) “Density-Based Clustering Validation”. In Proceedings of the 14th SIAM International Conference on Data Mining, p.839–847.
- Pal, N.; Biswas, J. (1997) “Cluster validation using graph theoretic concepts”. In Pattern Recognition, Vol.30, p.847–857.
- Santos, D. S. and Bazzan, A. L. C. (2012) “Distributed clustering for group formation and task allocation in multiagent systems: A swarm intelligence approach”. In Appl. Soft Comput., Vol. 8, p. 2123-2131.
- Takacs, B. and Demiris, Y. (2010) “Spectral clustering in multi-agent systems”. In Knowl Inf. Syst., Vol. 25, p. 607–622.
- Wooldridge, M. (2002) “An Introduction to Multiagent Systems”, JOHN WILEY & SONS, LTD, 1st edition.
- Xu, R. and Wunsch II, D. C. (2009) “Clustering”, IEEE Press, 1st edition.