IAR0001 - 2017/1
Relatório Trabalho 2
Ant Clustering com Dados Heterogêneos

Alexandre Maros[1]

[1]Departamento de Ciência da Computação – Universidade do Estado de Santa Catarina
Centro de Ciências Tecnológicas – Joinville – SC – Brasil

alehstk@gmail.com

Resumo. *O Trabalho 2 estende a lógica do agrupamento de formigas homogêneas e aplica o conceito para o agrupamento de dados. Aqui as "formigas mortas" são na realidade dados heterogêneos, com diferentes atributos e são agrupados através de uma função de similaridade. Neste trabalho será discutido como essas funções são utilizadas e alguns testes serão realizados para verificar a efetividade do agrupamento.*

1. Introdução

Como visto no desenvolvimento do Trabalho 1, o agrupamento de formigas mortas por formigas vivas é um conceito interessante e eficaz quando há a necessidade de agrupar dados homogêneos, sem uma propriedade única. Entretanto, as aplicações que tal conceito pode resolver se tornam limitadas.

Visando resolver esse problema, diferentes métodos de calcular a similaridade de dados foram desenvolvidos. Com esses métodos distintos, há a possibilidade de agrupar dados heterogêneos, isto é, dados que possuem características diferentes uns dos outros. Um exemplo simples seria o de agrupar grupos de formigas com tamanhos diferentes. Formigas maiores devem ser agrupadas juntas e formigas menores devem estar em um outro grupo, mais próximas umas das outras.

Os dados podem ter diversas peculiaridades. Peso e tamanho de uma formiga; Tamanho de arquivo e data da última modificação; Peso, tamanho e idade de uma pessoa. A quantidade de características pode variar se criado uma função de similaridade que comporte esses dados. Dessa forma, é possível modelar diversas maneiras diferentes para se agrupar dados com pequenas modificações no algorítimo.

Aqui se estudará dois modelos propostos e estudados por [Jafar and Sivakumar 2010] e [Handl et al. 2003] onde se é utilizado a distância euclidiana para calcular a similaridade.

2. Problemática

Neste trabalho, as formigas mortas deixam de ser homogêneas, com as mesmas propriedades e passam a ser heterogêneas, com propriedades diferentes. Isso implica que agora as formigas se transformam em dados e o agrupamento se baseia em quão similar são esses dados uns dos outros. Um exemplo simples seria com formigas com pesos diferentes. Formigas mais pesadas, devem ser agrupadas juntas, enquanto as mais leves devem ser postas em um grupo separado.

Aqui, trabalharemos com dados que contem 2 características, que podem ser interpretas como posições. Formigas que são mais similares, isto é, estão mais próximas umas das outras, devem estar em um mesmo grupo. Deve-se notar que "próximas" não significa a posição do tabuleiro, mas sim, proximidade dos dados. Para ilustrar a problemática, abaixo encontra-se um exemplo simplificado da entrada.

−21.75161566	−20.00828819	1
−20.23733722	−19.36787841	1
+21.51552028	+17.66002862	2
+20.30472224	+19.60669745	2
−19.81298536	+19.76620021	3
−15.37010376	+19.17040501	3
+19.23332939	−20.21870089	4
+19.27999504	−16.97360944	4

O número da terceira coluna é apenas para fins visuais, ele indica o número do grupo que a formiga pertence (para pintar com uma cor diferente no tabuleiro e identificar se estão sendo agrupadas corretamente) e não é usado no cálculo da similaridade. Os 2 primeiros são as "posições" dos dados. Novamente, esses números não representam as posições do tabuleiro, mas são dados para calcular a similaridade. Quanto mais próximo esses números estão uns dos outros, mais similar eles são. Esses dados são chamados de posição pois a distância euclidiana é utilizado para o cálculo da similaridade.

3. Modelo implementado

O trabalho foi implementado utilizando a linguagem C++ e a biblioteca gráfica SFML (*Simple and Fast Multimedia Library*). A implementação segue o mesmo padrão do trabalho anterior com três grandes mudanças.

A primeira é como as formigas mortas, ou nesse caso os dados, são criados. O programa lê um arquivo de texto com n entradas. Para cada entrada, uma formiga morta é criada com os valores das posições e a que grupo ela pertence. Logo após isso, a formiga é colocada em uma posição aleatória no tabuleiro.

A segunda é a forma como as formigas se movimentam. No trabalho anterior, a direção do próximo movimento era definida aleatoriamente a cada "passo", com a mesma probabilidade para todas as direções. Aqui, essa movimentação foi modificada. A formiga escolherá um ponto aleatório do mapa e seguirá naquela direção até chegar nele. Ao chegar nesse ponto, uma outra localização será decidida. Caso a formiga seja impossibilitada de fazer um movimento (encontrou outra formiga no caminho ou está carregando uma formiga morta e encontrou outra morta no caminho) a formiga também escolherá outro ponto aleatório.

A última modificação é em relação as fórmulas para calcular a probabilidade de pegar ou soltar uma formiga morta. Neste trabalho foi usado as formulas descritas no trabalho [Jafar and Sivakumar 2010] com algumas modificações e alterações descritas no trabalho de [Handl et al. 2003].

Aqui, primeiro se é calculado a similaridade da vizinhança em relação a uma formiga i. Essa similaridade é então usada como parâmetro para duas outras fórmulas que decidem a probabilidade da formiga pegar ou soltar uma formiga morta.

A fórmula para calcular a similaridade da vizinhança ou medida de densidade de uma formiga i é a seguinte:

$$f(i) = \begin{cases} \frac{1}{s^2} \sum_j (1 - \frac{d(i,j)}{\alpha}) & \text{, se } f(i) > 0 \wedge \forall_j (1 - \frac{d(i,j)}{\alpha}) > 0 \\ 0 & \text{, caso contrário} \end{cases} \quad (1)$$

onde:

- s é o número de quadrantes vazios na vizinhança;
- $d(i, j)$ é a distância euclidiana dos dois dados sendo analisados, definida pela Fórmula 2
- α é o fator que define a escala para a dissimilaridade.

$$d(x_i, x_j) = \sqrt{\sum_{k=1}^{n} (x_{i,k} - x_{j,k})^2} \quad (2)$$

onde:

- n é o número de características do dado

A Fórmula 1 é utilizada para identificar quão similar é a formiga i em relação aos dados à sua volta. O parâmetro α da fórmula não deve ser nem muito maior nem muito menor que o valor médio de distância esperado pela Equação 2 e é definido empiricamente. As fórmulas para decidir se a formiga deve carregar ou soltar uma formiga morta são definidas da seguinte forma:

$$P_c(x_i) = \left(\frac{k_1}{k_1 + f(x_i)} \right)^2 \qquad P_s(x_i) = \left(\frac{f(x_i)}{k_2 + f(x_i)} \right)^2 \quad (3)$$

onde:

- $P_c(x_i)$ é a probabilidade da formiga viva carregar a formiga x_i
- $P_s(x_i)$ é a probabilidade da formiga viva soltar a formiga x_i
- k_1 e k_2 são fatores de correção e são definidos empiricamente

Caso $P_c(x_i)$ ou $P_s(x_i)$ seja 0% o valor será 0.05%. Se $P_c(x_i)$ ou $P_s(x_i)$ forem 100% o valor será 99.95%. Isso é utilizado para simular comportamentos que não sempre explicados ou para não deixar a escolha completamente determinística quando esses valores absolutos acabam sendo gerados.

Um número fixo de iterações são calculadas a cada execução. Quando o número de iterações máximas é atingida, todas as formigas vivas, caso estejam carregando algo, continuam sua movimentação até largar o dado. O trabalho continua implementado de maneira serial, isto é, cada formiga executa sua movimentação de forma sequencial.

Para o calculo do tamanho do tabuleiro foi utilizado o método proposto por [Handl et al. 2003], onde se tenta atingir uma proporção $\frac{1}{10}$ na fórmula $\frac{N_{itens}}{N_{celulas}}$, onde N_{itens} é o número de dados dispostos no tabuleiro e $N_{celulas}$ é o número total de posições do tabuleiro.

4. Experimentos, resultados e análises

Foram disponibilizados duas entradas. A primeira contendo 4 grupos com 400 formigas e o segundo contendo 15 grupos com 600 formigas. As variáveis que não foram modificadas dependendo da entrada são as seguintes:

- número de formigas vivas = 50
- raio de visão = 1
- $k1 = 0.02$
- $k2 = 0.03$

4.1. Entrada 1 - 400 dados e 4 grupos

Aqui, 400 dados pertencentes a 4 grupos, sendo que cada grupo contém 100 dados, foram dispostos aleatoriamente no tabuleiro. Os dados foram gerados a partir de uma distribuição normal. As Figuras 1, 2 e 3 mostram como foi feita as disposições dos dados inicial e final dado a entrada 1. É importante notar que aqui, os grupos possuem valores bem distintos um do outro, sendo possível assim um melhor agrupamento.

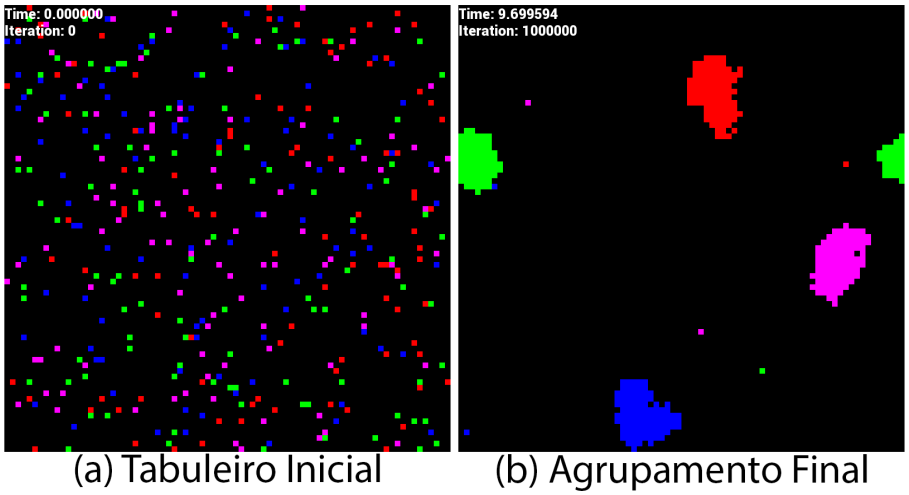

Figura 1. Entrada 1 – Execução 1

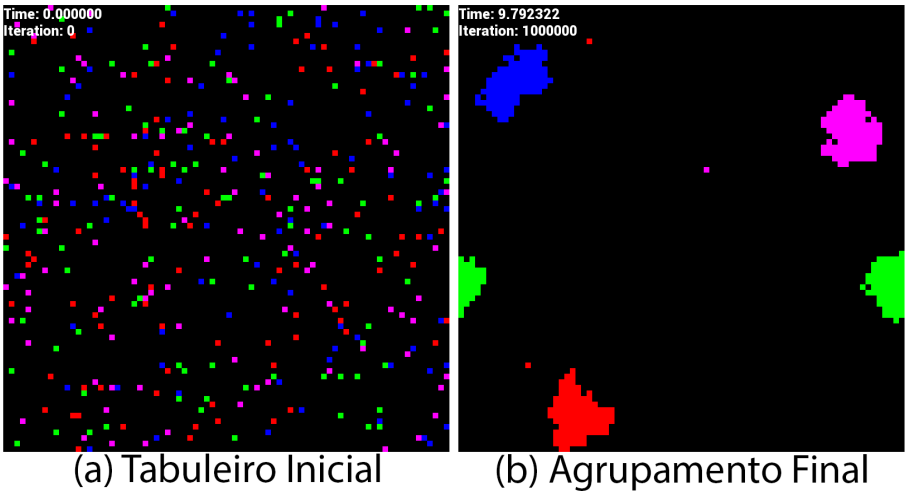

Figura 2. Entrada 1 – Execução 2

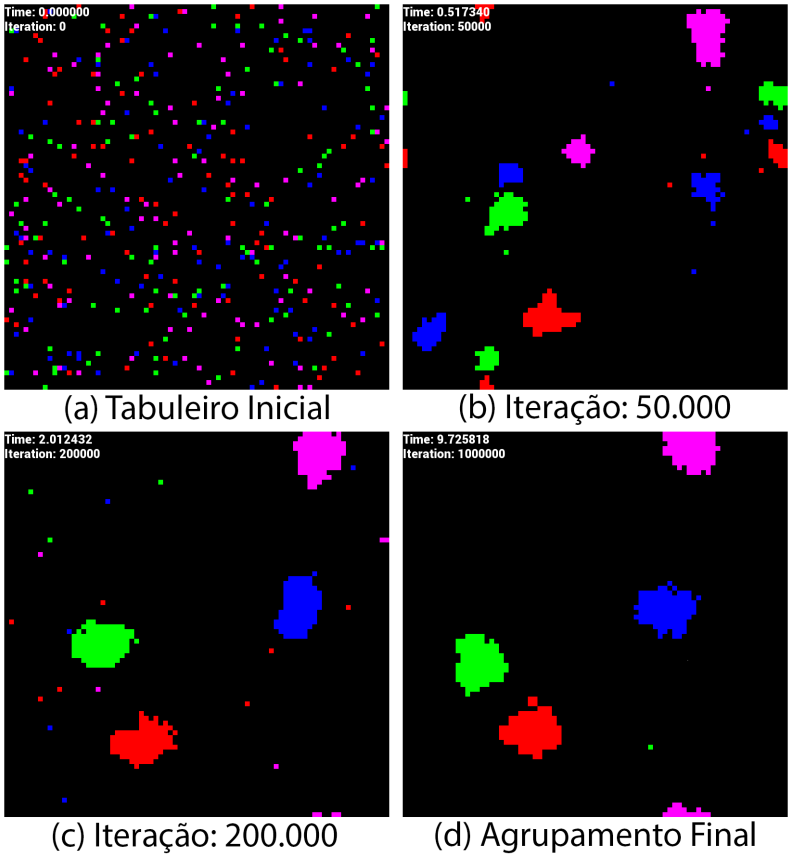

Figura 3. Entrada 1 – Execução 3 com o estado do sistema em 4 momentos distintos

Para essa entrada o valor do α foi de 10 e as dimensões do tabuleiro foi de 63×63.

Nas três execuções é possível notar que o agrupamento foi eficiente. Formaram-se quatro grupos bem definidos e com poucos buracos. Na Figura 1, quatro dados ficaram fora de algum grupo, na Figura 2, três dados ficaram fora de grupos e na Figura 3 apenas um dado ficou desconexo.

É possível notar na Figura 3 que a formação de grupos é rápida. Em menos de 1 segundo (50 mil iterações) já é possível notar claramente os grupos sendo formados. Na iteração 200 mil os grupos já estão bem definidos e apenas alguns dados estão espalhados.

4.2. Entrada 2

Nessa entrada há 600 dados pertencentes a 15 grupos distintos, sendo que cada grupo contém 40 dados. Aqui, os grupos são mais parecidos entre si, possuindo uma diferença pequena nos valores dos dados entre cada um deles. O valor do α utilizado foi de 1.6 e a dimensão do tabuleiro foi de 77×77.

Essa entrada provou ser mais complicada de se agrupar corretamente devido a similaridade entre os grupos. É possível notar que alguns dados são postos em outros grupos similares. Isso é ilustrado na Figura 6.

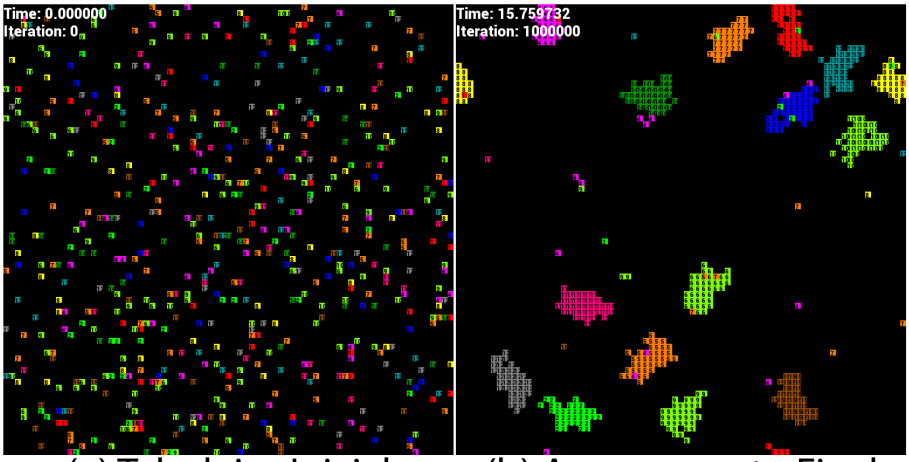

Figura 4. Entrada 2 – Execução 1

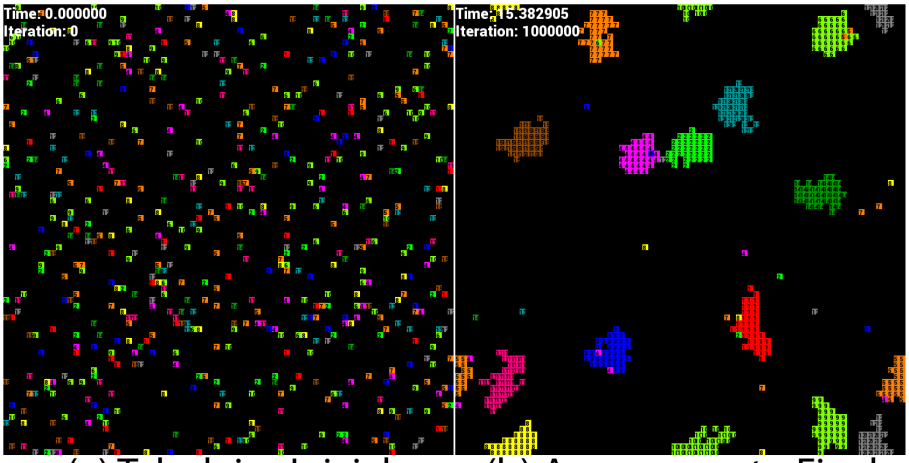

Figura 5. Entrada 2 – Execução 2

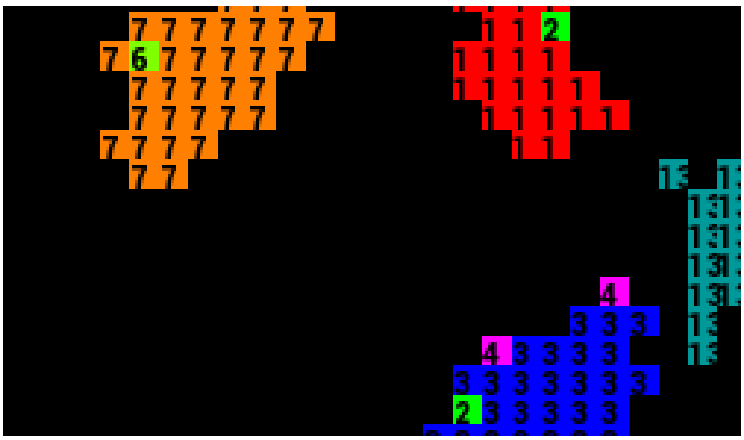

Figura 6. Ampliação de uma área da Execução 1 - Entrada 2, onde é possível notar que alguns dados foram erroneamente agrupados devido a sua similaridade.

Alguns dados ficaram em grupos errados e muitos ficaram fora de grupos. Isso sugere que há a necessidade de fazer uma análise mais específica para essa entrada, ajustando os valores de α, k_1 e k_2 de uma maneira mais precisa. Outro caso a se testar é o aumento do número de iterações executadas.

Os tempos de execução foram os seguintes:

Tempo (segundos)		
	Entrada 1	Entrada 2
Execução 1	9.699s	15.758s
Execução 2	9.792s	15.382s
Execução 3	9.725s	–

Tabela 1. Tempos de execução de 1 milhão de iterações com base nas entradas

Os tempos de agrupamento foram relativamente rápidos. Houve um aumento na segunda entrada devido ao aumento do tabuleiro e também das formigas presentes no tabuleiro (há mais cálculos de similaridade sendo feitas).

5. Conclusão

Novamente, o agrupamento se mostrou eficiente, principalmente para quando a diferença entre os grupos de dados eram maiores, como foi o caso da Entrada 1. Em um intervalo de tempo curto (Tabela 1) é possível agrupar uma grande quantidade de dados.

Como sugestão de trabalhos futuros está a paralelização das ações das formigas, a melhora da função de movimentação e uma análise mais refinada das variáveis empíricas das Equações 1 e 3.

Também há algumas melhoras propostas por [Handl et al. 2003], onde pode-se implementar um α e um raio de visão dinâmico para cada formiga.

Referências

Handl, J., Knowles, J., and Dorigo, M. (2003). Ant-based clustering: a comparative study of its relative performance with respect to k-means, average link and id-som. In *Proceedings of the Third International Conference on Hybrid Intelligent Systems. IOS Press*.

Jafar, O. M. and Sivakumar, R. (2010). Ant-based clustering algorithms: A brief survey. *International journal of computer theory and engineering*, 2(5):787.