

Estratégias de coordenação em enxames robóticos para minimizar o congestionamento de grupos

Gustavo P. Mitsuchi Oliveira¹, Douglas Macharet¹

¹Departamento de Ciência da Computação – Universidade Federal de Minas Gerais (UFMG)

{gustavo.oliveira, doug}@dcc.ufmg.br

Abstract. *The use of Multi-Robot Systems presents several advantages like increased robustness and, in most cases, time reduction to accomplish a task. In this context, Robotic Swarms is generally related to very large groups of robots. The congestion problem within the mobile robotics field is a well-known field of research. The problem occurs when a large number of autonomous robots tries to reach a common target in the environment at the same time. This paper addresses the congestion problem considering a robotic swarm formed by distinct groups of agents, and we aim to minimize the period of time agents of the same group take to reach the target. We propose four different approaches to optimize the navigation process of the entire group, however, prioritizing members of the same group. The first strategy gives to each group a priority based on the average distance between its members and the target. This team priority will define which robots will move into the target first. The second one states which robots will move based on a randomized group priority. The third will segregate the groups in different points before allowing them to reach the goal. The fourth will also segregate the groups but, in the shape of circular queues. The proposed methodology was evaluated through several experiments in a simulated environment, providing statistical insight on the performance.*

Resumo. *O uso de sistemas multi-robôs apresenta diversas vantagens como o aumento da robustez e, na maioria dos casos, redução do tempo para realizar uma tarefa. Neste contexto, os enxames robóticos geralmente estão relacionados a grupos muito grandes de robôs. O problema do congestionamento no campo da robótica móvel é um campo de pesquisa bem conhecido. O problema ocorre quando um grande número de robôs autônomos tenta atingir um alvo comum no ambiente, ao mesmo tempo. Este artigo aborda o problema de congestionamento considerando um enxame robótico formado por grupos distintos de agentes, e buscamos minimizar o período de tempo os agentes do mesmo grupo levam para atingir o alvo. Nós propomos quatro abordagens diferentes para otimizar o processo de navegação de todo o grupo, no entanto, priorizando membros de um mesmo grupo. A primeira estratégia dá a cada grupo uma prioridade com base na distância média entre seus membros e alvo. Esta prioridade da equipe definirá quais robôs se moverão primeiro em direção ao alvo. O segundo afirma quais robôs vão se mover com base em uma prioridade de grupo randomizada. O terceiro irá segregar os grupos em diferentes pontos antes de permitir que eles se movam ao objetivo. O quarto também irá segregar os grupos porém, na forma de filas circulares. A metodologia proposta foi avaliada através de vários experimentos em um ambiente simulado, fornecendo informações estatísticas sobre o desempenho.*

1. Introdução

No mundo atual, especialmente no contexto da tecnologia e inovação, um assunto que está cada vez mais sendo discutido e ganhando espaço no cotidiano é a robótica. A popularização desse tópico se deve ao fato de que incontáveis problemas de diversas áreas diferentes podem ser solucionados de alguma forma com o auxílio de robôs agindo individualmente ou em cooperação com outros robôs.

Porém, com os avanços na área, crescem os problemas e impedimentos relacionados às novas tecnologias e estratégias de resolução de problemas associados à robótica. Nesse trabalho será abordado o problema de congestionamento em situações envolvendo múltiplos robôs organizados em sub conjuntos.

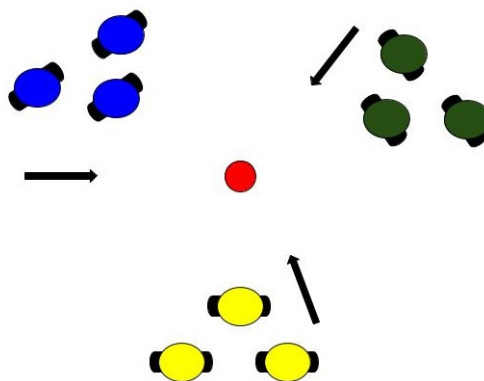


Figura 1. Robôs com alvo em comum agrupados

O problema consiste em evitar o congestionamento e a colisão de um conjunto de robôs autônomos que têm como objetivo chegar um ponto em comum. Dentro desse conjunto de robôs, os indivíduos estão agrupados em sub conjuntos que correspondem a times de robôs, podendo representar times de robôs do mesmo tipo ou com uma mesma função.

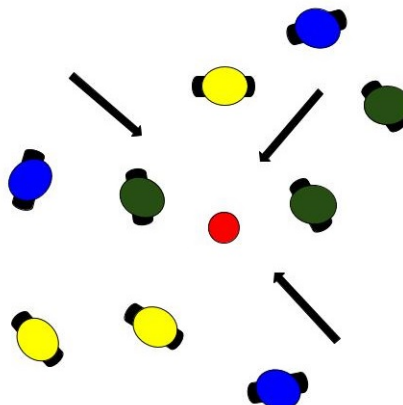


Figura 2. Robôs com alvo em comum desagrupados

As figuras 1 e 2 representam duas situações possíveis do problema do congestionamento. Em cada uma delas, o ponto vermelho representa o alvo em comum dos robôs e a cor de cada robô representa seu sub grupo. Na primeira, os robôs estão dispostos de

modo que cada indivíduo do sub grupo fique próximo uns aos outros, já na segunda, não há nenhuma organização prévia.

É fácil visualizar o problema no mundo real, sendo possível citar como exemplo robôs tentando chegar a um ponto para realizar um carregamento de bateria, robôs mineradores autônomos tentando minerar uma mesma mina ou até mesmo quadro-rotorez que devem chegar a um determinado alvo para transportá-lo.

Esse trabalho tem como objetivo propor quatro algoritmos que solucionam o problema de maneira descentralizada, ou seja cada indivíduo será responsável pela sua tomada de decisão, realizar uma série de experimentos utilizando cada um deles e apresentar de maneira comparativa os resultados. As métricas para avaliar o desempenho de cada um deles será descrita em um seção posterior.

2. Referencial teórico

O problema do congestionamento e tráfego de indivíduos já foi abordado por diversos autores, inclusive de outras áreas diferentes da robótica. Nessa seção serão citados trabalhos que abordam o problema específico da robótica.

Grossman (1988)[Grossman 1988], é um dos pioneiros a tratar do problema de tráfego de robôs, ele propõe uma solução centralizada, ou seja, uma unidade central é responsável pela tomada de decisão de cada indivíduo e cada robô age de maneira independente, não havendo sub-grupos, nem uma alvo em comum para os robôs.

Em Hinoshi (2011)[Hoshino 2011], o problema do congestionamento é abordado de maneira simplificada, onde os robôs possuem um mesmo alvo em comum porém podem trafegar em apenas um determinado trajeto cada um, nesse projeto, além de estarem dispostos em sub grupos, os robôs poderão vir de qualquer direção e trafegar em todo o espaço.

Os algoritmos que serão construídos nesse projeto terão grande influência de Marcolino et al.(2009) [Marcolino and Chaimowicz 2009] e Marcolino et al.(2016)[Marcolino et al. 2016], em ambos trabalhos são propostos algoritmos descentralizados para o tratamento do problema e considerando que os indivíduos podem estar em qualquer posição do espaço de trabalho. Marcolino et al.(2009) [Marcolino and Chaimowicz 2009], apresenta um algoritmo probabilístico que trata o problema do congestionamento mas apresenta problemas no momento que os robôs estão deixando o alvo. No segundo, os autores propõem uma modificação nesse algoritmo a fim de resolver essa questão.

Vale ressaltar nessa seção que, não foi encontrado um trabalho prévio que trate do problema do congestionamento em grupos ou swarms de robôs subdivididos.

3. Objetivos

Os algoritmos foram desenvolvidos com o objetivo de otimizar dois aspectos específicos do problema que serão descrito a seguir. As métricas são atribuídas a cada um dos sub-grupos

1. **Tempo de Trânsito.** É o tempo que leva para que um indivíduo do subgrupo alcance a área objetivo, ou seja, o tempo gasto para que o grupo comece o processo

de alimentação (entrada e saída da área objetivo). Durante esse período, é esperado que cada indivíduo do subgrupo realize movimentos que tendem a levá-lo a área objetivo.

2. **Tempo de alimentação** Essa métrica diz respeito ao tempo que um subgrupo leva para que todos os seus indivíduos alcancem a área objetivo. É esperado que durante esse período de tempo os indivíduos de um subgrupo entrem e saiam da área objetivo.
3. **Distância Percorrida** Essa métrica é a distância percorrida por todos os robôs, incluindo todos os grupos.

4. Estratégias

4.1. Algoritmo da Distância Média

O primeiro método desenvolvido foi o chamado Algoritmo da Distância Média. Nele, cada subgrupo de robôs possui uma prioridade P , que determina qual pode avançar em direção a área objetivo. A prioridade P de cada subgrupo é definida com base na média das distâncias entre cada robô j e a área objetivo, dentro de um grupo i com n robôs, como mostrado na equação 1. O subgrupo que possuir a menor distância média, possuirá a maior prioridade P e por consequência poderá se encaminhar em direção a área objetivo.

$$D_i = \frac{\sum_{j \in i} dist(j, Alvo)}{n} \quad (1)$$

Os subgrupos com menor prioridade podem apenas desviar dos demais, até que o último robô do subgrupo com maior prioridade entre e saia da área objetivo. Quando isso ocorre, o próximo subgrupo recebe permissão para se encaminhar a área objetivo.

Algorithm 1 Distância Média

```

for all  $i \in \text{grupos}$  do
   $D_i = \frac{\sum_{j \in i} dist(j, Alvo)}{n}$ 
  if  $D_i \leq min\_dist$  then
     $min\_dist = D_i$ 
     $grupo\_prioridade = i$ 
while  $existe\_grupo\_entrando()$  do
  for all  $j \in grupo\_prioridade$  do
     $move(j)$ 
  if  $grupo\_prioridade\_saindo()$  then
    for all  $i \in grupos\_restantes$  do
       $D_i = \frac{\sum_{j \in i} dist(j, Alvo)}{n}$ 
      if  $D_i \leq min\_dist$  then
         $min\_dist = D_i$ 
         $grupo\_prioridade = i$ 

```

Para a navegação, o método utiliza da técnica de campos potenciais, que será descrita em uma seção posterior, pois o outro algoritmo também faz uso desse tipo de navegação.

4.2. Algoritmo Probabilístico

O segundo método desenvolvido nesse trabalho foi livremente inspirado no trabalho de Marcolino et al.(2009) [Marcolino et al. 2016]. Nesse segundo algoritmo, a permissão de ir em direção a área alvo de cada grupo é dada por uma variável aleatória.

A cada n iterações, um valor aleatório é gerado e se o valor for menor que uma variável ρ , o grupo ganha a permissão de avançar, caso contrário, o grupo tem apenas a permissão de desviar dos demais indivíduos.

No início da execução, todos os grupos estão no estado *esperando*, ao final de n iterações, cada um dos grupos tem chance ρ de transitar para o estado *entrando*. Quando todos os robôs do grupo já entraram na área alvo e estão se movendo em direção oposta à mesma, o grupo está no estado *saindo*. Como no método anterior, a execução finaliza quando o último robô deixa a área objetivo.



Figura 3. Máquina de estados finitos do algoritmo probabilístico

Algorithm 2 Probabilístico

```

for all  $i \in \text{grupos}$  do
   $\text{random} = \text{random}()$ 
  if  $\text{random} \leq \rho$  then
     $\text{tem\_permissao}(i)$ 
while  $\text{existe\_grupo\_entrando}()$  do
  for all  $j \in \text{grupo\_prioridade}$  do
     $\text{move}(j)$ 
     $\text{iteracoes}++$ 
  if  $\text{iteracoes} \% n == 0$  then
    for all  $i \in \text{grupos}$  do
       $\text{random} = \text{random}()$ 
      if  $\text{random} \leq \rho$  then
         $\text{tem\_permissao}(i)$ 
  
```

4.3. Estratégia das filas circulares

As estratégias apresentadas nessa subseção e na imediatamente seguinte, se baseiam no algoritmo descrito na subseção 4.1, sendo diferente quanto a organização dos grupos não prioritários.

Na estratégia das filas circulares, os subgrupo com menor prioridade se organizam em círculos concêntricos com a área objetivo. Cada círculo possui um raio r que varia de modo inversamente proporcional à prioridade do grupo que vai se agrupar nele, ou seja, grupos com maior prioridade (baseada na distância média), vão se dispor em círculos de

menor raio, se localizando mais perto da área objetivo. Conforme as prioridades se alteram, os círculos se alterarão, causando um deslocamento de todos os grupos em direção ao alvo, a cada momento em que mudam as prioridades.

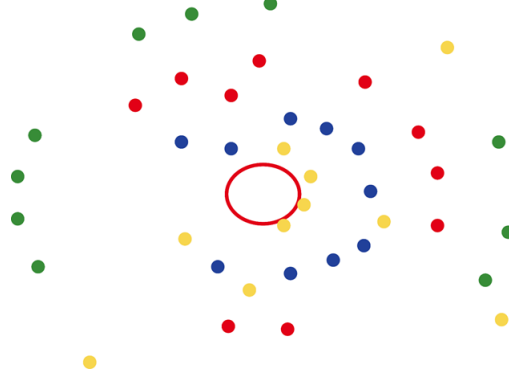


Figura 4. Ilustração da estratégia de filas circulares

A figura 4 apresenta uma ilustração da organização dos grupos se posicionando nas chamadas filas circulares. A movimentação dos indivíduos em direção às suas respectivas filas, se dá de modo a se afastar ou se aproximar do alvo, dependendo da sua posição corrente e seguindo o método dos campos potenciais, que será descrito na próxima seção.

Algorithm 3 Algoritmo das Filas Circulares

```

for all  $i \in \text{grupos}$  do
   $D_i = \frac{\sum_{j \in i} \text{dist}(j, \text{Alvo})}{n}$ 
  if  $D_i \leq \text{min\_dist}$  then
     $\text{min\_dist} = D_i$ 
     $\text{grupo\_prioridade} = i$ 
while  $\text{existe\_grupo\_entrando}()$  do
  for all  $j \in \text{grupo\_prioridade}$  do
     $\text{move}(j)$ 
  for all  $j \notin \text{grupo\_prioridade}$  do
     $\text{move\_para\_fila}(j)$ 
  if  $\text{grupo\_prioridade\_saindo}()$  then
    for all  $i \in \text{grupos\_restantes}$  do
       $D_i = \frac{\sum_{j \in i} \text{dist}(j, \text{Alvo})}{n}$ 
      if  $D_i \leq \text{min\_dist}$  then
         $\text{min\_dist} = D_i$ 
         $\text{grupo\_prioridade} = i$ 

```

O algoritmo 4.3 descreve a estratégia. É importante notar que o comportamento é bem semelhante a estratégia da priorização pela distância média, sendo diferente apenas no momento em que os subgrupos estão em espera, como visto na linha 10.

4.4. Estratégia da segregação em pontos

Como na estratégia descrita anteriormente, na estratégia da segregação em pontos, os grupos que não tem permissão de avanço ao alvo, vão se agrupar em pontos predefinidos

no mapa. Cada ponto $p(x, y)$ associado a um grupo j é calculado no início do algoritmo utilizando as equações 2 e 3, onde θ é 2π dividido pelo número de subgrupos e o ponto do grupo 0 é definido como (5,0).

$$x_p = x_{p-1} \cos \theta - y_{p-1} \sin \theta \quad (2)$$

$$y_p = y_{p-1} \cos \theta + x_{p-1} \sin \theta \quad (3)$$

Com os pontos definidos, cada grupo com baixa prioridade se move em direção ao seu respectivo ponto, utilizando a navegação por campos potenciais que será apresentada na próxima seção. Assim que o grupo recebe a permissão para se deslocar em direção ao alvo, ele assim o faz, abandonando a região do seu ponto de segregação.

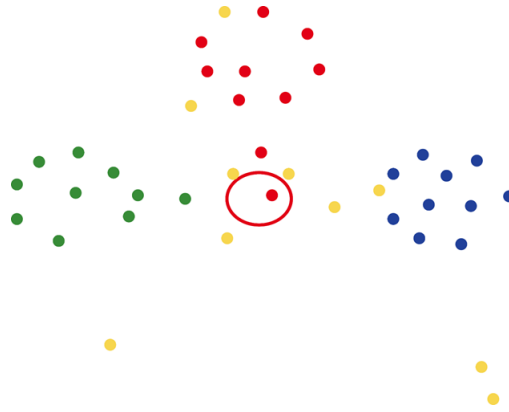


Figura 5. Ilustração da estratégia de segregação em pontos

Algorithm 4 Algoritmo da segregação em pontos

```

for all  $i \in \text{grupos}$  do
     $D_i = \frac{\sum_{j \in i} \text{dist}(j, \text{Alvo})}{n}$ 
    if  $D_i \leq \text{min\_dist}$  then
         $\text{min\_dist} = D_i$ 
         $\text{grupo\_prioridade} = i$ 
while  $\text{existe\_grupo\_entrando}()$  do
    for all  $j \in \text{grupo\_prioridade}$  do
         $\text{move}(j)$ 
    for all  $j \notin \text{grupo\_prioridade}$  do
         $\text{move\_para\_ponto}(j)$ 
    if  $\text{grupo\_prioridade\_saindo}()$  then
        for all  $i \in \text{grupos\_restantes}$  do
             $D_i = \frac{\sum_{j \in i} \text{dist}(j, \text{Alvo})}{n}$ 
            if  $D_i \leq \text{min\_dist}$  then
                 $\text{min\_dist} = D_i$ 

```

Assim como a estratégia de filas circulares, a segregação em pontos tem como

algoritmo originário a priorização pela distância média, sendo possível notar a semelhança no algoritmo 4.4, que descreve a estratégia aqui apresentada.

5. Prevenção de Colisões

A técnica escolhida para a navegação de cada robô em todos os métodos desenvolvidos foi a modelagem por campos potenciais.

Nesse tipo de navegação, cada robô é modelado de modo a se comportar como cargas elétricas de mesmo sinal e a área objetivo como uma carga de sinal oposto. Isso faz com que exista um campo de força potencial ao redor de cada robô que "repele" outros indivíduos e o atrai para a área objetivo.

$$F_{att}(q) = -k_{att} \cdot (q - q_{objetivo}) \quad (4)$$

$$F_{rep}(q) = \begin{cases} k_{rep} \left(\frac{1}{\rho(q)} - \frac{1}{\rho_0} \right) \frac{1}{\rho^2(q)} & \text{se } \rho(q) \leq \rho_0 \\ 0 & \text{se } \rho(q) > \rho_0 \end{cases} \quad (5)$$

As equações 4 e 5 mostram como são calculadas as forças de atração e repulsão respectivamente. A incógnita q representa um robô e função $\rho(q)$ é a distância do robô para um obstáculo (outro robô).

De maneira bem simples, a força resultante em cada robô é calculada a cada iteração e sua nova posição é computada. A força resultante é dada pelo somatório das forças de repulsão mais a força de atração em direção a área objetivo.

A técnica é interessante para esse tipo de problema principalmente por funcionar em ambientes dinâmicos. Porém ainda podem ocorrer problemas de ótimo local quando as forças que atuam sobre um indivíduo resultam em um vetor nulo. A fim de tentar evitar esse problema, foi acrescentada uma variação aleatória na direção da força de repulsão em um robô. A direção que em princípio poderia somente ser totalmente oposta ao obstáculo, agora pode sofrer leves alterações, somando valores de $\frac{\pi}{2}$ até $\frac{\pi}{8}$.

6. Metodologia

Para a etapa de experimentação do trabalho, houve a implementação de ambos os métodos descritos anteriormente, utilizando a linguagem Python 3. A linguagem oferece uma biblioteca que permite a visualização gráfica de cada iteração do algoritmo, o que facilita a compreensão.

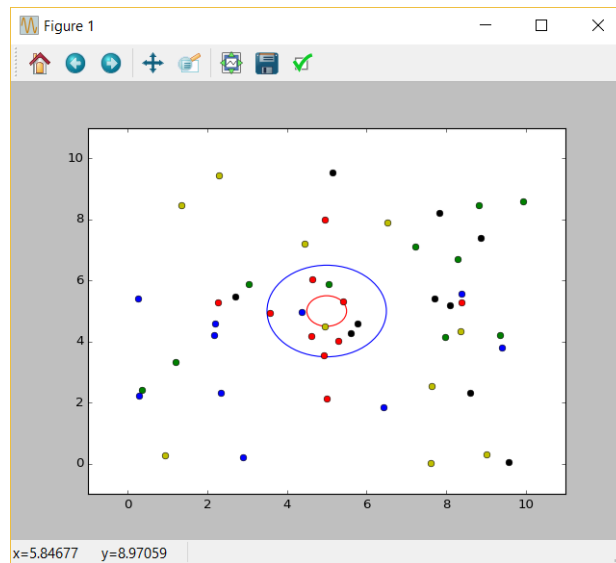


Figura 6. Ambiente de execução dos experimentos

Foram testadas as variações para o seguinte conjunto de parâmetros:

- **Quantidade de robôs em cada subgrupo**
- **Quantidade de subgrupos**

Os indivíduos em cada um dos experimentos aparecem aleatoriamente no mapa, portanto, foi necessário realizar cada um dos experimentos 100 vezes, para melhores resultados.

7. Experimentos

Nessa seção serão apresentados e analisados os resultados dos experimentos. Ela será dividida em sub seções que tratarão de experimentos com um determinado valor de alcance do sensor.

Antes de cada um dos experimentos, o ambiente de simulação é preparado com valores estáticos para alguns parâmetros, como é possível observar na tabela 1. Os valores utilizados surgiram de observações e experimentos prévios.

Parâmetro	Valor
Região de surgimento de robôs	[[0,10],[0,10]]
Posição da área objetivo	[5,5]
Raio da área objetivo	0.5
Força máxima (módulo)	0.08
K_ATT	0.1
K_REP	0.04
Alcance do sensor	1

Tabela 1. Parâmetros de execução

Para fins comparativos, a simulação foi realizada também para uma estratégia base, que denominamos de "Sem Regra". Nessa estratégia não há nada que faça qualquer grupo entrar no modo de espera, a simulação inicia e todos os indivíduos avançam em direção

ao alvo ao mesmo tempo, utilizando o mesmo método de navegação que as estratégias anteriormente apresentadas.

7.1. Resultados

A figura 10 mostra o gráfico com os resultados das execuções, exibindo a informação sobre tempo de trânsito. As informações que dizem respeito a tempo, sempre serão exibidas como número de iterações a fim de desconsiderar o tempo de processamento computacional.

É possível notar que, no algoritmo probabilístico e no sem regra, os grupos levam menos iterações para começar a chegar a área objetivo. Isso era esperado já que, no algoritmo da distância, os grupos de menor prioridade ficam estáticos, esperando o grupo com maior prioridade terminar de atingir a área objetivo, ao contrário do probabilístico em que os grupos podem avançar ao objetivo a qualquer momento e do sem regra, onde todos obrigatoriamente se movem em direção ao alvo ao mesmo tempo, o que faz com que o tempo médio de espera de cada grupo seja menor e consequentemente diminuindo o tempo de trânsito.

Quanto às estratégias que realizam uma ordenação prévia dos subgrupos, nota-se que a estratégia de filas circulares melhora o tempo médio de espera em relação ao algoritmo da distância média puro. Isso se deve ao fato de que ele organiza os subgrupos em círculos próximos à área objetivo, permitindo a cada subgrupo uma menor navegação até atingir o alvo, reduzindo a métrica. A estratégia da segregação em pontos, por sua vez, gerou um tempo médio maior do que a estratégia da distância média, provavelmente porque os pontos em que cada subgrupo se segrega são relativamente distantes da área objetivo quando se compara a menor distância que um indivíduo pode surgir no mapa, dado que eles aparecem em pontos aleatórios.

No quesito tempo de alimentação, cujo resultado é exibido na figura 14, percebe-se uma vantagem considerável do algoritmo da distância média sobre o algoritmo probabilístico e o base. O que causa essa discrepância é que o congestionamento pode ocorrer de modo severo no algoritmo probabilístico já que, mais de um grupo pode tentar atingir a área objetivo no mesmo período de tempo, fato que aumenta o tempo de alimentação de cada um desses subgrupos.

O tempo de alimentação é ainda melhor nas estratégias que utilizam de organizações iniciais, fato já era esperado pois, tais algoritmos além de utilizar a priorização pela menor distância média, otimizam a navegação em direção ao alvo quando organizam os indivíduos de mesmo grupo em posições semelhantes. Tal característica permite uma movimentação mais eficiente e livre de obstáculos.

Na métrica da distância percorrida pelos robôs, o método da distância média se mostrou menos eficiente, como se observa na figura 18. Esse comportamento se explica pelo movimento de repulsão entre robôs que ocorre quando os grupos estão em estado de espera que é somado com o movimento de atração ao alvo e movimento de repulsão que os indivíduos descrevem nessa estratégia. Na estratégia base não a repulsão em espera e no probabilístico quase não ocorre, o que diminui a distância percorrida. As estratégias com segregação prévia apresentam o mesmo problema de seu "algoritmo originário", porém tal efeito parece ser atenuado, no caso da segregação em pontos e diminuído consideravelmente utilizando estratégia de filas circulares.

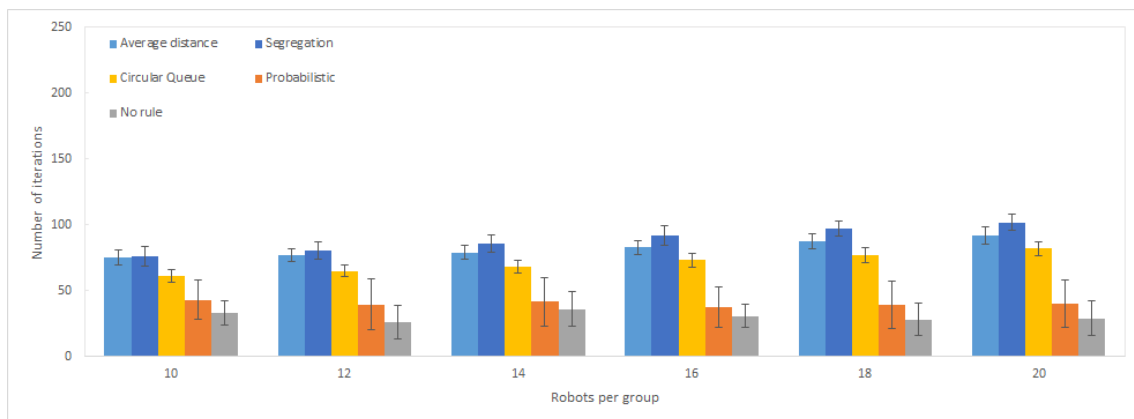


Figura 7. Tempo de Trânsito médio para 3 grupos, variando o número de indivíduos

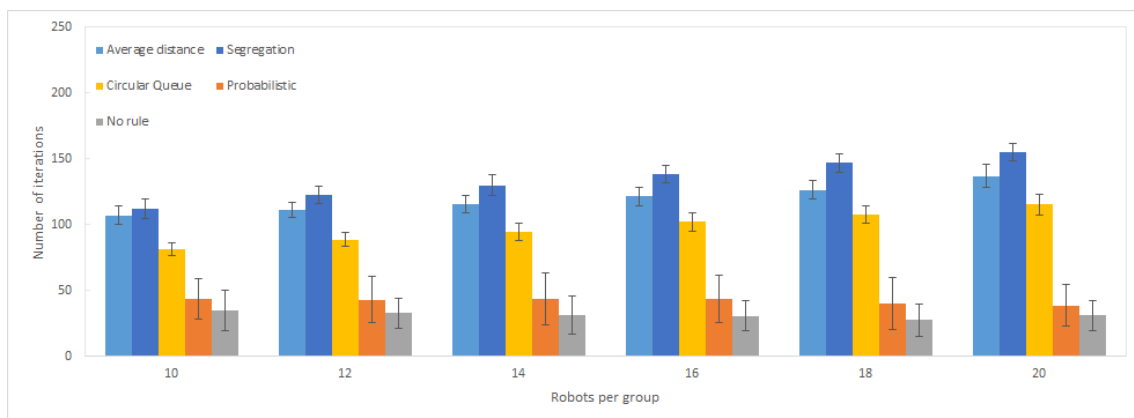


Figura 8. Tempo de Trânsito médio para 4 grupos, variando o número de indivíduos

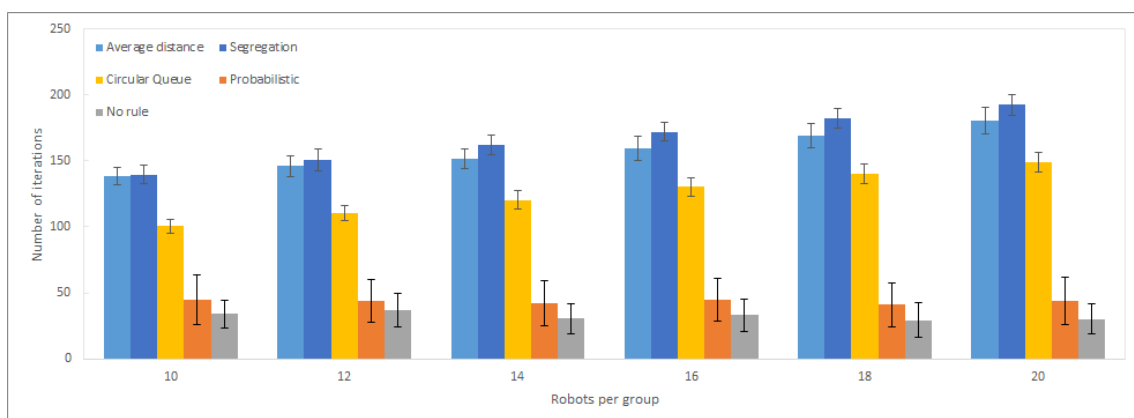


Figura 9. Tempo de Trânsito médio para 5 grupos, variando o número de indivíduos

Figura 10. Tempo de Trânsito

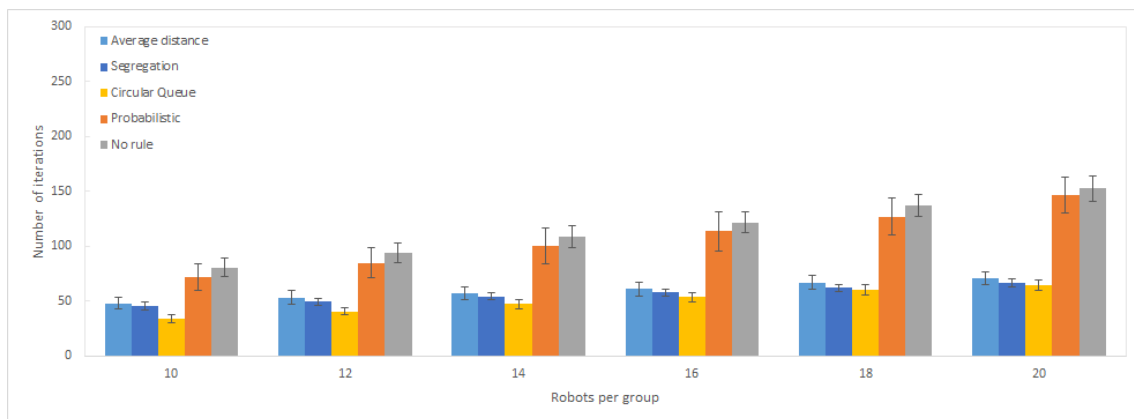


Figura 11. Tempo de Alimentação médio para 3 grupos, variando o número de indivíduos.

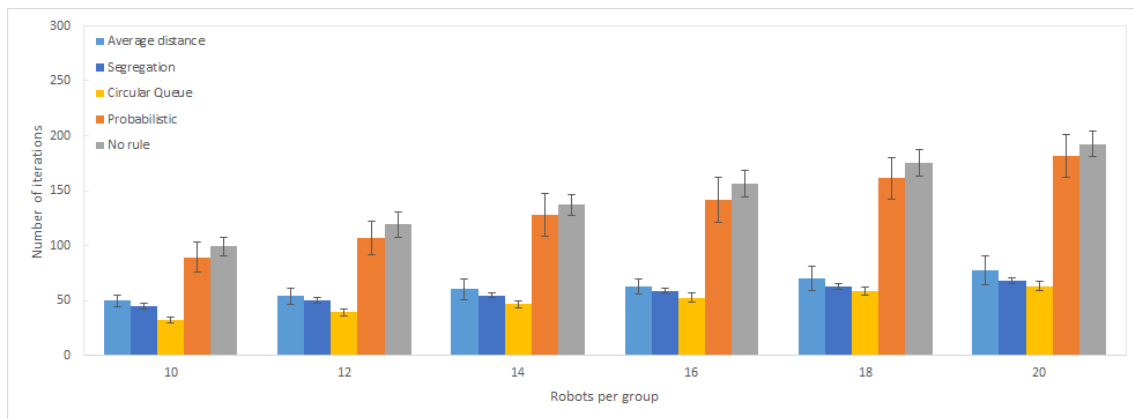


Figura 12. Tempo de Alimentação médio para 4 grupos, variando o número de indivíduos.

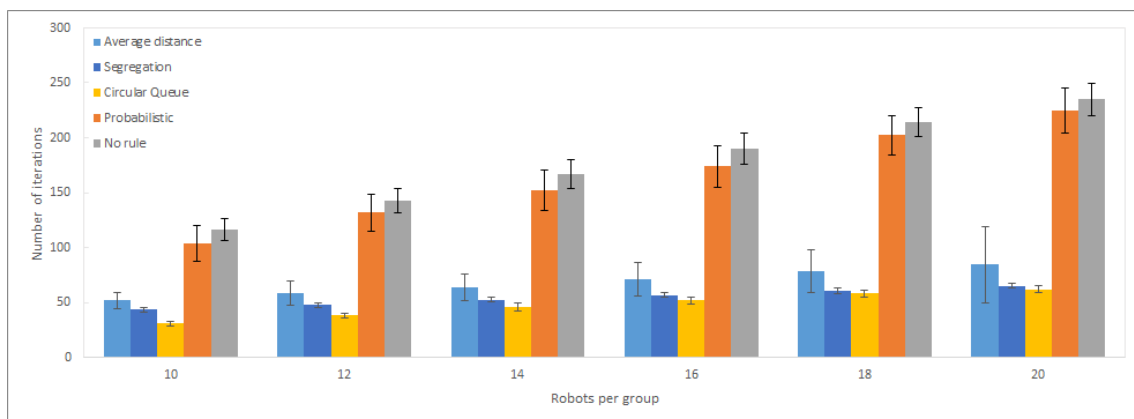


Figura 13. Tempo de Alimentação médio para 5 grupos, variando o número de indivíduos.

Figura 14. Tempo de Alimentação

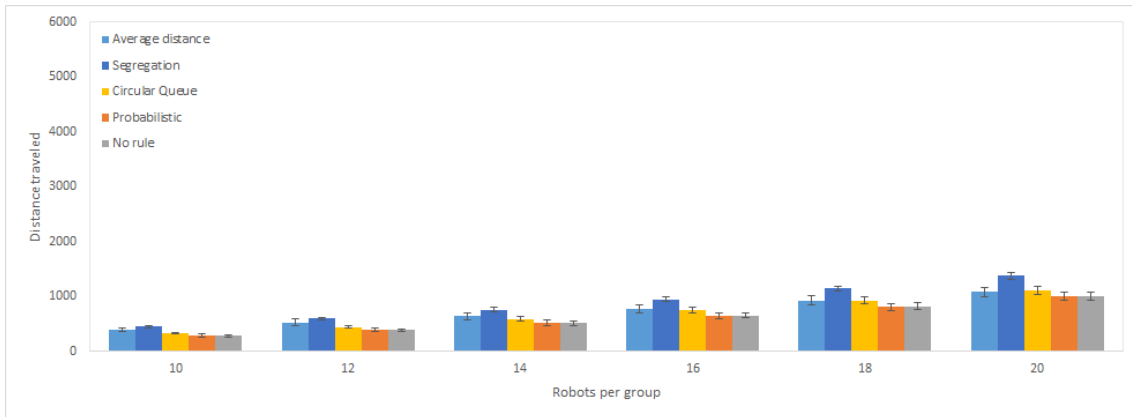


Figura 15. Distância percorrida média para 3 grupos, variando o número de indivíduos.

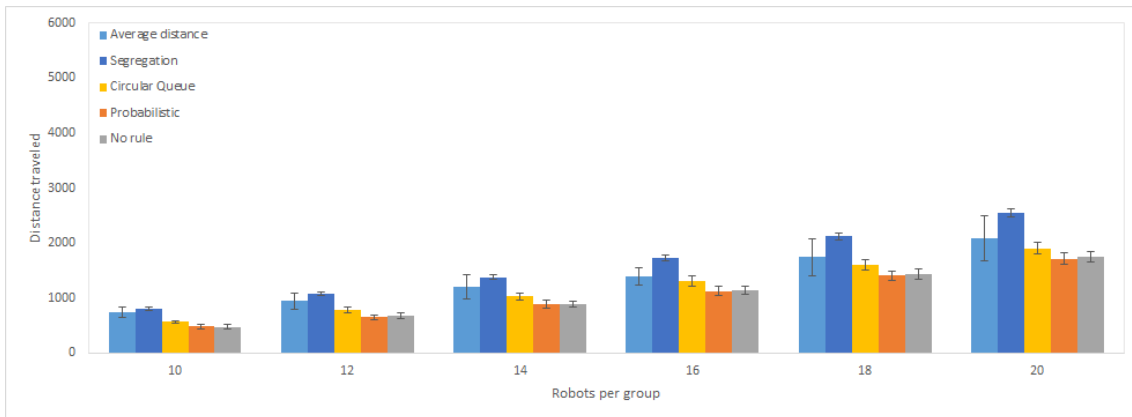


Figura 16. Distância percorrida média para 4 grupos, variando o número de indivíduos.

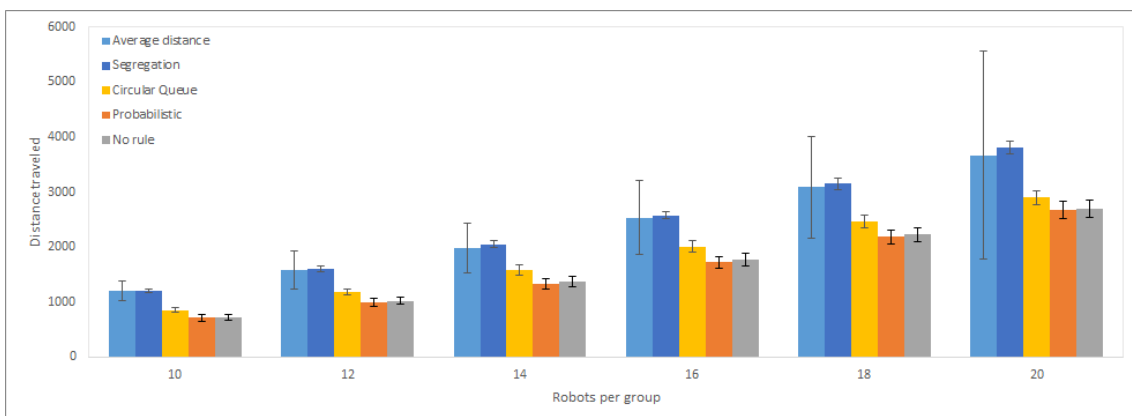


Figura 17. Distância percorrida média para 5 grupos, variando o número de indivíduos.

Figura 18. Distância Percorrida

8. Conclusão

Os métodos aqui apresentados mostraram desempenhos um tanto diferentes. No método pirobalístico, o tempo gasto para o grupo começar a entrar na área alvo é bem pequeno, sendo pouco afetado pelo número total de robôs. Entretanto para o tempo total médio de alimentação, ele apresenta um desempenho bem ruim quando comparado com o método da distância média. Isso acontece porque o método do probabilístico permite o congestionamento, já que mais de um grupo pode tentar chegar ao mesmo tempo a área objetivo. Com o congestionamento, também aumenta a métrica da distância percorrida pelos indivíduos, já que eles vão se movimentar devido a influência de outros robôs muito próximos. Os métodos que realizam uma segregação prévia e se baseiam na estratégia de priorização pela distância média se mostraram mais eficiente que seu predecessor em todas as métricas, com exceção da estratégia de segregação em pontos que foi pior, em média, na métrica da distância percorrida.

Para trabalhos futuros, a utilização de uma técnica de navegação que não seja a de campos potenciais é sugerida, visto que essa tem problemas e é bem dependente da posição inicial dos indivíduos. Também como melhoramento necessário às estratégias é a otimização dos parâmetros pois, em todos os experimentos foram utilizados valores empíricos.

Referências

- Grossman, D. D. (1988). Traffic control of multiple robot vehicles. *IEEE Journal on Robotics and Automation*, 4(5):491–497.
- Hoshino, S. (2011). Multi-robot coordination methodology in congested systems with bottlenecks. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2810–2816. IEEE.
- Marcolino, L. S. and Chaimowicz, L. (2009). Traffic control for a swarm of robots: Avoiding target congestion. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1955–1961. IEEE.
- Marcolino, L. S., dos Passos, Y. T., de Souza, Á. A. F., dos Santos Rodrigues, A., and Chaimowicz, L. (2016). Avoiding target congestion on the navigation of robotic swarms. *Autonomous Robots*, pages 1–24.