

Otimização do planejamento de trajetórias de robôs móveis baseada em colônia de bactérias

Cezar Augusto Sierakowski e Leandro dos Santos Coelho
Pontifícia Universidade Católica do Paraná, Grupo Produtônica
Programa de Pós-Graduação em Engenharia de Produção e Sistemas
Rua Imaculada Conceição, 1155, CEP 80215-901, Curitiba, PR
E-mail: cezars@globocom; lscoelho@rla01.pucpr.br

Abstract

Foraging theory originated in attempts to address puzzling findings that arose in ethological studies of food seeking and prey selection among animals. The potential uses of biomimicry of social foraging strategies to develop advanced controllers and cooperative control strategies for autonomous vehicles is an emergent research topic. The activity of foraging can be focused as an optimization process. In this paper, a bacterial foraging approach for path planning of mobile robots is presented. Two cases study of static environment with obstacles are described and evaluated. Simulation results show the performance of the bacterial foraging procedure in different environments in the planned trajectories.

1. Introdução

A inteligência coletiva (*swarm intelligence*) é uma área emergente de pesquisa que apresenta características de população e evolução similares aos algoritmos evolutivos (ou evolucionários), contudo diferencia-se por enfatizar o comportamento de cooperação entre grupos de membros.

A inteligência coletiva é utilizada para resolver problemas de otimização e cooperação de agentes inteligentes, particularmente em redes de computadores, robótica coletiva [1] e controle cooperativo e/ou descentralizado [2]. A inteligência coletiva é inspirada na natureza, no fato de que a contribuição entre os diversos membros de um determinado grupo de seres vivos contribuem com as suas próprias experiências para o grupo, tornando-o mais forte perante os outros. Os representantes mais conhecidos da inteligência coletiva em problemas de otimização são abordagens de colônia de formigas [3], enxame de partículas [4] e sistema imunológico artificial [5].

A inteligência coletiva na natureza pode ser constituída de três princípios básicos: a avaliação, a comparação e a imitação.

A avaliação é a capacidade de analisar o que é positivo ou negativo na natureza, atrativo ou repulsivo. Até mesmo os menores seres vivos possuem esta capacidade, no caso das bactérias, estas têm a capacidade de perceber se o meio em que se encontram é nocivo ou não. O aprendizado não pode ocorrer a não ser que o organismo seja capaz de avaliar características atrativas e repulsivas do meio.

A comparação é a forma como os seres utilizam outros seres como um padrão para medir a si mesmos, e os resultados destas comparações podem vir a ser uma motivação para aprenderem e/ou mudarem.

A imitação, por sua vez, é uma forma efetiva de aprendizado. Entretanto, poucos animais na natureza são realmente capazes de realizar imitações, na realidade, apenas os seres humanos e algumas espécies de aves são capazes disto [4].

Estes três princípios básicos podem ser combinados de forma simplificada em programas de computadores, possibilitando que os mesmos se adaptem a problemas complexos. Os animais ou grupos destes, quando visam a busca por alimentos, tendem a tomar ações de forma a maximizar a energia obtida por unidade de tempo, levando em conta as suas restrições fisiológicas e também as restrições ambientais.

Neste artigo é considerada a aplicação da teoria de busca por alimentos relacionada às bactérias, que neste artigo adota a denominação de colônia de bactérias. A motivação para este estudo é devido ao fato das bactérias serem mais simples que diversos outros seres vivos e que podem utilizar-se da teoria de busca dos alimentos de forma a obter benefícios para o grupo. As bactérias possuem um sistema de controle que a permite controlar a sua busca por comida e evitar a presença de substâncias nocivas. Neste contexto, a atividade de cooperação de uma colônia de bactérias

pode ser utilizada em um procedimento de otimização baseado na busca de alimento, conforme proposto por Passino [6].

A contribuição deste artigo é apresentar um estudo da colônia de bactérias para o planejamento de trajetórias de robôs móveis. Neste contexto são avaliados dois casos simulados de mapeamentos em ambiente com a presença de obstáculos.

As próximas seções do artigo são organizadas da seguinte forma. Na seção 2, os fundamentos e o equacionamento do algoritmo de colônia de bactérias são descritos. Na seção 3, o problema de otimização de trajetórias de robôs móveis e dois estudos de caso são contextualizados. Na seção 4 são discutidos os resultados obtidos com a aplicação da colônia de bactérias. A conclusão e perspectivas de futuros trabalhos são apresentadas na seção 5.

2. Colônia de bactérias

A seleção natural tende a eliminar animais com estratégias de procura por alimentos (*foraging strategies* – métodos de localização, manipulação e ingestão de alimentos) fracas e a favorecer a propagação dos genes que possuem estratégias de procura por alimentos favoráveis, uma vez que os mesmos possuem maiores possibilidades de obter sucesso na reprodução. Estes princípios evolucionários levaram os cientistas a desenvolverem a teoria da busca por alimento, de forma a desenvolver a hipótese de que é apropriado modelar a atividade de busca por alimentos como um processo de otimização [6].

A presença de flagelos nas bactérias permite a sua locomoção e isto é obtido pela sua rotação em uma mesma direção, a uma velocidade de 100 a 200 rotações por segundo. As bactérias podem se mover de duas formas diferentes, estas podem correr (nadar por um período em uma mesma direção), este movimento é obtido quando os flagelos são rotacionados no sentido anti-horário, ou então, elas podem sofrer uma rotação, quando os flagelos são rotacionados no sentido horário. As bactérias operam numa alternância entre estes dois métodos de locomoção durante a sua vida.

Após um período de corrida, a bactéria sofre uma rotação, o período de rotação é de $0,14 \pm 0,19s$, segundo Passino [6]. Após esta rotação, a bactéria estará voltada para uma direção aleatória. Quando os flagelos são rotacionados no sentido anti-horário, a bactéria se desloca em uma direção, a uma velocidade média de $10-20\mu m/s$, ou seja, cerca de 10 vezes o seu comprimento por segundo, por um período médio de corrida de $0,86 \pm 1,18s$.

É possível que os ambientes onde se localizam as bactérias se modifiquem, tanto de forma gradual quanto de forma repentina. Com isso, as bactérias podem sofrer um processo de eliminação, através do surgimento de uma substância nociva, ou então, se dispersem, através da ação de uma outra substância, com isso gerando os efeitos de eliminação e dispersão.

As posições das bactérias em um determinado instante de tempo podem ser determinadas através da equação (1), onde a posição de determinado instante é obtida usando a posição no instante anterior mais um valor do tamanho do passo $C(i)$ a ser aplicado em uma direção aleatória $\phi(j)$, gerada pela rotação da bactéria,

$$\theta'(j+1) = \theta'(j) + C(i) * \phi(j) \quad (1)$$

Para a utilização de tal estratégia como uma forma de resolver problemas de otimização, deve existir uma equação que determine o custo de cada uma das posições, de forma a determinar qual a melhor posição. O custo é determinado segundo a equação,

$$J(i, j) = J(i, j) + J_{cc}(\theta'(j), P(j)). \quad (2)$$

Através da equação (2) nota-se que o custo de uma determinada posição $J(i, j)$ é também afetado pela atratividade e pela repulsão existente entre as diversas bactérias existentes na população $J_{cc}(\theta'(j), P(j))$.

Após um determinado número de passos *chemotactic* (passos que compreendem a movimentação e determinação do custo da posição de cada bactéria), um passo de reprodução ocorre. Neste passo de reprodução ordenam-se as bactérias através do custo acumulado, de forma decrescente. Durante os passos quimiotáticos (*chemotactic*), a metade inferior desta lista morre, ou seja, as bactérias que não conseguiram juntar nutrientes suficientes para se reproduzirem, e a metade superior dividem-se em duas novas bactérias, localizadas na mesma posição.

Na figura 1 (apresentada no final do artigo) é detalhado o algoritmo desta teoria em pseudocódigo. Como pode ser visto através do pseudocódigo, o algoritmo para a colônia de bactérias é composto inicialmente por um laço (*loop*) de eliminação e dispersão, dentro deste laço, existe um outro que é responsável pela reprodução das bactérias. Dentro deste existe ainda um terceiro laço que é responsável por determinar a direção para a qual cada bactéria se deslocará e a determinação de por quanto tempo esta permanecerá se deslocando e, portanto, determinando sua posição após a execução deste laço e também por determinar o custo (*fitness*) destas posições. O laço de reprodução é responsável por determinar quais das bactérias devem se reproduzir e quais delas devem ser

exterminadas após as movimentações exercidas no laço 3, através da análise dos custos das mesmas ao longo da sua movimentação. O primeiro laço é responsável por eliminar algumas bactérias, regido por uma probabilidade de eliminação e posicioná-las em outra posição do espaço analisado.

3. Otimização de trajetória de robôs móveis

A literatura é rica em abordagens para resolução de problemas de planejamento de trajetórias de robôs móveis na presença de obstáculos estáticos e/ou móveis [7]. Uma dos mais populares métodos de planejamento é o de campos potenciais artificiais [8]. Entretanto, este método dá somente uma solução de rota que pode não ser a menor trajetória quando em um ambiente estático. A principal dificuldade na determinação de uma trajetória ótima é devida ao fato que os métodos analíticos são muito complexos de serem utilizados em tempo real e os métodos de busca enumerativos são muito afetados pelo tamanho do espaço de busca.

Recentemente, tem sido difundido o interesse na utilização de algoritmos evolutivos, geralmente algoritmos genéticos para o planejamento de trajetórias para robôs móveis quando o espaço de busca é grande [9],[10],[11].

O aspecto central da movimentação de um robô móvel é o planejamento de sua trajetória. O problema de planejamento da trajetória de um robô móvel é tipicamente formulado como segue: dado um robô e a descrição de um ambiente, planeja-se uma trajetória entre duas localizações específicas que é livre de colisão e satisfaz um critério de desempenho [10].

A abordagem adotada neste artigo é que o problema de planejamento de trajetória seja visto como um problema de otimização. Neste caso, uma seqüência de configurações que movem o robô móvel de um ponto inicial (origem) até um ponto final (destino).

Um planejador (otimizador) de trajetórias deve localizar uma série de configurações que evita colisões entre o(s) robô(s) e o(s) obstáculo(s) presente(s) no espaço de trabalho. Para ser eficiente o planejador deve tentar também minimizar o comprimento da trajetória encontrada. O espaço de busca é o conjunto de todas as possíveis configurações.

No presente estudo, é considerado um problema de planejamento de trajetória de um robô móvel pontual e bidimensional, onde a posição de um robô móvel R são representados por coordenadas cartesianas (x, y) no plano xy . O ponto inicial e ponto meta do robô são $(x_0,$

$y_0)$ (x_{np}, y_{np}) , onde n_p é um parâmetro de projeto. O ponto inicial adotado é sempre $(100,100)$.

Somente o problema de planejamento de trajetória foi enfatizado neste artigo. O problema de controle do robô não foi enfocado neste artigo. No entanto detalhes das equações de movimentação do robô podem ser encontrados em Fujimori *et al.* [9].

É assumido que os obstáculos são circulares no plano de movimentação do robô. Além disso, é realizada a hipótese de que o espaço bidimensional livre é conectado e os obstáculos são finitos em tamanho e não se sobrepõem com o ponto meta.

O problema de otimização formulado consiste de um problema de otimização discreta, onde a função objetivo $f(x,y)$, que é a ligação entre a colônia de bactérias e o entorno, visa a minimização da trajetória total a ser percorrida pelo robô móvel e é regida por

$$f(x, y) = \alpha d_{obj} + \lambda n_o \quad (3)$$

$$d_{obj} = \sum_{i=0}^{n_p} \sqrt{(x(i+1) - x(i))^2 + (y(i+1) - y(i))^2} \quad (4)$$

onde α e λ são fatores de ponderação, d_{obj} representa a distância euclidiana entre o ponto inicial e o ponto meta e n_o denota o número de obstáculos atingidos pela movimentação do robô se a trajetória planejada fosse d_{obj} , e n_p é o número de pontos de mudança de trajetória do robô (parâmetro de projeto do usuário, neste artigo). Nota-se pela equação (3) que existe um termo λ ponderação (penalização) de soluções ineficazes, ou seja, as trajetórias que interceptam obstáculos. Neste caso, a função *fitness* a ser avaliada pela colônia de bactérias visa a maximização de

$$fitness = \frac{K_c}{f(x, y) + \varepsilon} \quad (5)$$

onde K_c e ε são constantes de escala.

4. Resultados de simulação

Para a iniciar o sistema, os parâmetros p (dimensão do problema de otimização), S (tamanho da população), N_c (número de passos quimiotáticos), N_s (número máximo de passos seguidos nadando), N_{re} (número de reproduções), N_{ed} (número de eliminações e dispersões), p_{ed} (probabilidade de eliminação e dispersão) e $C(i)$, $i=1,2,...,S$ (velocidade de movimentação regida por um tamanho de passo). Os valores iniciais de θ , $i=1,2,...,S$ (posições) foram escolhidos de forma aleatória com distribuição uniforme. Adota-se, neste artigo, $S=50$, $N_c=500$, $N_s=4$,

$N_{re}=4$, $N_{ed}=2$, $p_{ed}=0.3$ e $C(i)=2.5$, $i=1,2,...,S$. Estes parâmetros foram escolhidos por tentativa e erro.

O mapa para o planejamento das trajetórias é uma área de 100x100 metros. O intervalo de busca de parâmetros usando a colônia de bactérias é $x_i \in [0,100]$ metros e $y_i \in [0,100]$ m, onde $i=1,...,np$. Em relação ao *fitness* adotou-se $\alpha=1$, $\lambda=200$, $K_c=100$ e $\varepsilon=1 \times 10^{-6}$.

A seguir são apresentados dois casos simulados e uma análise dos resultados para 5 experimentos com a otimização por colônia de bactérias.

4.1. Caso 1: Ambiente com 4 obstáculos

Na tabela 1 são apresentadas as posições dos centros (x_c, y_c) dos obstáculos circulares e seu respectivo raio (em metros) do estudo de caso 1. Os resultados obtidos com a colônia de bactérias são resumidos para $np=3$.

Tabela 1: Obstáculos para o estudo de caso 1.

obstáculo nº	raio	posição (x_c, y_c)
1	18	(40, 20)
2	10	(80, 50)
3	10	(80, 80)
4	15	(40, 70)

O melhor (maior) resultado de *fitness* para $p=3$ foi obtido pela colônia de bactérias com a solução:

- $(x_1, y_1) = (71,2323; 57,6478)$,
- $(x_2, y_2) = (49,6346; 87,2888)$ e
- $(x_3, y_3) = (19,0272; 33,5094)$.

Na figura 2 é apresentado o melhor resultado dos 5 experimentos realizados. Neste caso, avaliando-se o *fitness* tem-se [máximo, média, mínimo, desvio padrão]=[0,6861; 0,6571; 0,6129; 0,017].

4.2. Caso 2: Ambiente com 12 obstáculos

Na tabela 2 são apresentadas as posições dos centros (x_c, y_c) dos obstáculos circulares e seu respectivo raio (em metros) para o estudo de caso 2. Os resultados com a colônia de bactérias são resumidos para $np=4$.

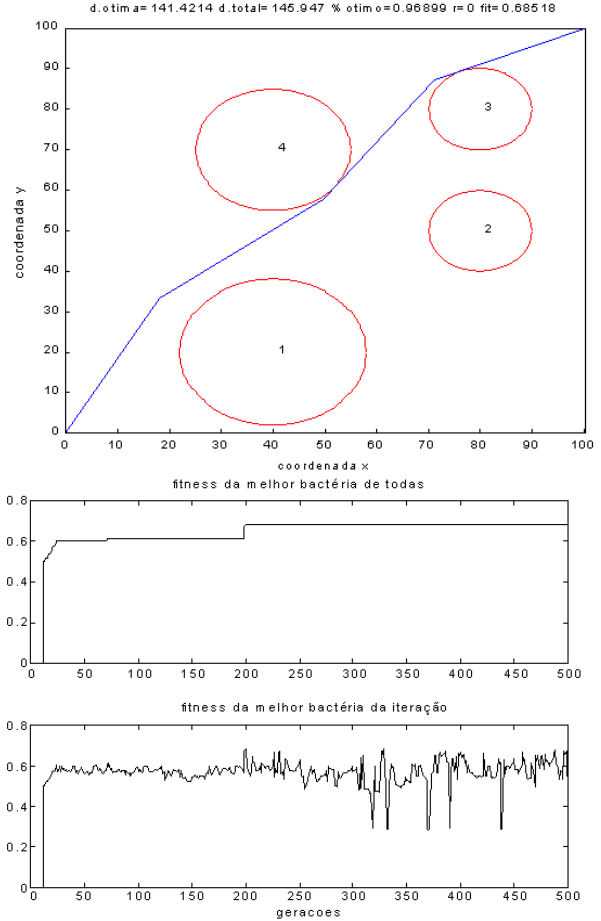


Figura 2. Melhor resultado com a colônia de bactérias para o caso 1 (sendo que a distância mínima sem obstáculos seria de 141,4214).

Tabela 2: Obstáculos para o estudo de caso 2.

obstáculo nº	raio	posição (x_c, y_c)
1	05	(50, 40)
2	10	(75, 75)
3	10	(50, 70)
4	05	(20, 20)
5	10	(40, 15)
6	10	(70, 10)
7	05	(75, 50)
8	10	(20, 60)
9	10	(30, 40)
10	08	(85, 50)
11	05	(20, 80)
12	08	(20, 80)

Na figura 3 é apresentado o melhor resultado de 5 experimentos. Neste caso, avaliando-se o *fitness* tem-se [máximo, média, mínimo, desvio padrão]=[0,6129; 0,5871; 0,5649; 0,038]. Em síntese, o melhor resultado dos experimentos foi com:

- $(x_1, y_1) = (99, 9817; 86, 4327)$,
- $(x_2, y_2) = (56, 4813; 26, 7221)$,
- $(x_3, y_3) = (57, 8743; 17, 0184)$ e
- $(x_4, y_4) = (33, 22227; 42, 2410)$.

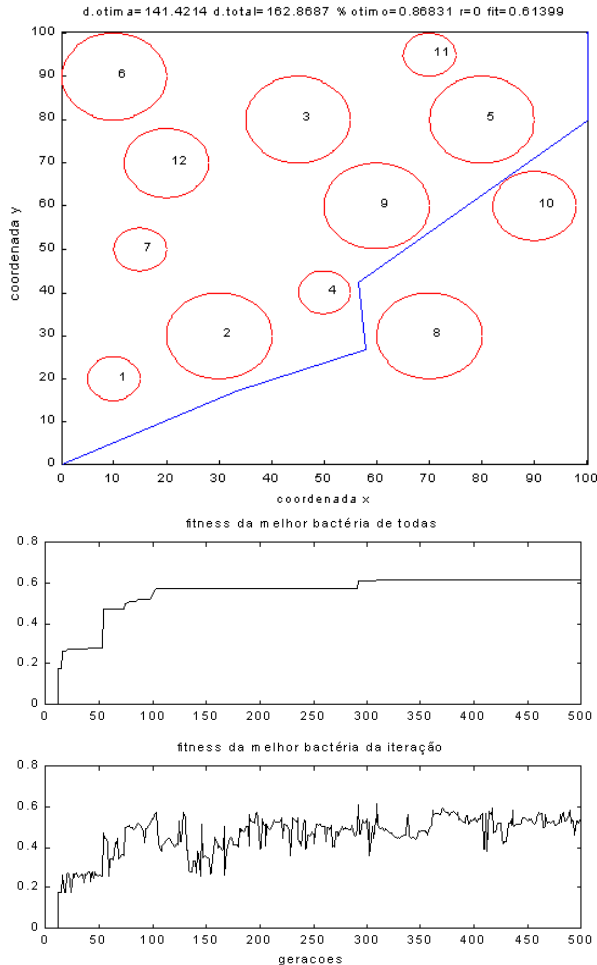


Figura 3. Melhor resultado com a colônia de bactérias para o caso 2 (sendo que a distância mínima sem obstáculos seria de 141,4214).

5. Conclusão e futuros trabalhos

Neste artigo é explorada a aplicação da teoria de busca por alimentos de uma colônia de bactérias na forma de um algoritmo de otimização. Deve-se enfatizar que as bactérias possuem um sistema de controle que a permite controlar a sua busca (otimizar) por comida e evitar a presença de substâncias nocivas (comportamento de repulsão).

A possibilidade de explorar a eficiência da colônia de bactérias foi apresentada neste artigo e com sucesso, pois nos dois casos simulados os resultados do planejamento da trajetória dos robôs móveis foram bons indicativos da eficiência da abordagem.

Entretanto, estudos mais aprofundados relacionados ao projeto do tamanho da população, número de passos quimiotáticos, número de passos seguidos de natação da bactéria, número de eliminações e dispersões e a velocidade de movimentação são necessários. Além disso, posteriormente será realizado um estudo comparativo com técnicas clássicas, algoritmos genéticos e colônia de bactérias para o planejamento de trajetórias para robôs móveis.

6. Referências

- [1] Liu, Y., and K.M. Passino, "Stable social foraging swarms in a noisy environment", *IEEE Transactions on Automatic Control*, vol. 49, no. 1, pp. 30-44, 2004.
- [2] Baras, J.S., X. Tan, and P. Hovareshti, "Decentralized control of autonomous vehicles", *Proceedings of the 42nd IEEE Conference on Decision and Control*, Maui, Hawaii, USA, pp. 1532-1537, 2003.
- [3] Dorigo M., and G. Di Caro, "The ant colony optimization meta-heuristic", in D. Corne, M. Dorigo, and F. Glover (editors), *New Ideas in Optimization*, McGraw-Hill, pp. 11-32, 1999.
- [4] Kennedy, J.F., R.C. Eberhart, and R.C. Shi, *Swarm intelligence*. San Francisco: Morgan Kaufmann Pub., 2001.
- [5] Silva, L.N.C., *Engenharia imunológica: desenvolvimento e aplicação de ferramentas computacionais inspiradas em sistemas imunológicos artificiais*, Tese de doutorado, DCA/UNICAMP, Campinas, SP, 2001.
- [6] Passino, K.M., "Biomimicry of bacterial foraging for distributed optimization and control", *IEEE Control Systems*, vol. 22, no. 3, pp. 52-67, 2002.
- [7] Tu, J., and S.X. Yang, "Genetic algorithm based path planning for a mobile robot", *Proceedings of the IEEE International Conference on Robotics & Automation*, Taipei, Taiwan, pp. 1221-1226, 2002.
- [8] Tsuji, T., Y. Tanaka, P.G. Morasso, V. Sanguineti, and M. Kaneko, "Bio-mimetic trajectory generation of robots via artificial potential field with time base generator", *IEEE Transactions on Systems, Man and Cybernetics, Part C*, vol. 32, no. 4, pp. 426- 439, 2002.
- [9] Fujimori, A., P.N. Nikiforuk, and M.M. Gupta, "Adaptive navigation of mobile robots with obstacle avoidance", *IEEE Transactions on Robotics and Automation*, vol. 13, no. 4, pp. 596-602, 1997.
- [10] Xiao, J., Z. Michalewicz, L. Zhang, and K. Trojanowski, "Adaptive evolutionary planner/navigator for robots", *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 18-28, 1997.

```

FAÇA {  $l = l + 1$ ;
  FAÇA {  $k = k + 1$ ;
    FAÇA {  $j = j + 1$ ;
      PARA CADA bactéria  $i$  {
        Calcular  $J(j, k, l)$ ;
        Assumir  $J(j, k, l) = J(j, k, l) + J_{cc}(\phi'(j, k, l), P(j, k, l))$ ;
        Salvar  $J_{last} = J(i, j, k, l)$ ;
        Gerar um vetor aleatório  $\Delta(i) \in \mathbb{R}^p$ , com valores no intervalo  $[-1, 1]$ ;
        Mover  $\theta'(j+1, k, l) = \theta'(j, k, l) + C(i) * \frac{\Delta(i)}{\sqrt{\Delta^T(i) * \Delta(i)}}$ ;

        Calcular  $J(i, j+1, k, l)$ ;
        Assumir  $J(i, j+1, k, l) = J(i, j+1, k, l) + J_{cc}(\theta'(j+1, k, l), P(j+1, k, l))$ ;
        Assumir  $m = 0$ ;
        FAÇA {
          Assumir  $m = m + 1$ ;
          SE  $J(i, j+1, k, l) < J_{last}$  (problema de maximização)
            ENTÃO
              Assumir  $J_{last} = J(i, j+1, k, l)$ ;
              Calcular  $\theta'(j+1, k, l) = \theta'(j+1, k, l) + C(i) * \frac{\Delta(i)}{\sqrt{\Delta^T(i) * \Delta(i)}}$ ;

              Calcular  $J(i, j+1, k, l)$ ;
              Assumir
                 $J(i, j+1, k, l) = J(i, j+1, k, l) + J_{cc}(\theta'(j+1, k, l), P(j+1, k, l))$ ;
            SENÃO
              Assumir  $m = N_s$ ;
        } ENQUANTO  $m < N_s$ ;
      }
    } ENQUANTO  $j < N_c$ ;
    PARA CADA bactéria  $i$  {
      Calcular  $J_{Health}^i = \sum_{j=1}^{N_s-1} J(i, j, k, l)$ ;
    }
    Ordenar as bactérias de acordo com o valor de  $J_{Health}$ ;
    Matar as bactérias com os maiores valores de  $J_{Health}$ ;
    Duplicar as bactérias com os menores valores de  $J_{Health}$ ;
  } ENQUANTO  $k < N_{re}$ ;
  PARA CADA bactéria  $i$  {
    Eliminar e dispersar a bactéria com a probabilidade  $p_{ed}$ ;
  }
} ENQUANTO  $l < N_{ed}$ .

```

Figura 1. Pseudocódigo da busca de alimento de uma colônia de bactérias.