APPENDIX IV

PSEUDO CODE FOR BACTERIAL FORAGING OPTIMIZATION ALGORITHM

Step 1  Initialize the following parameters

\(p\) as dimension of the search space

\(s\) as the number of bacteria in the population

\(N_c\) as the number of chemotactic steps per bacterium

lifetime between reproduction steps

\(N_s\) as maximum number of swim of bacteria in the same direction \(N_{re}\) as the number of reproduction steps

\(N_{ed}\) as the number of elimination and dispersal events

\(P_{ed}\) as the probability that each bacterium will be eliminated/dispersed

\(i = 1,2,...,S\) as the index for the bacterium

\(J = 1,2,.....N_{C}\) as the index for chemotactic step

\(K = 1,2,.....N_{re}\) as the index for reproduction step

\(l = 1,2,.....,N_{ed}\) as the index of elimination and dispersal event

\(m_s = 1,2,.....,N_s\) as the index for number of swim

Step 2  Elimination dispersal loop: for \(l = 1, 2,....., N_{ed}\), do \(l = l+1\)
Step 3  Reproduction loop: for \( k=1,2,\ldots,N_{re} \), do \( k=k+1 \)

Step 4  Chemo taxis loop; for \( j=1,2,\ldots,N_c \), do \( j=j+1 \)

a. For \( i=1,2,\ldots,s \), take a chemotactic step for bacterium \( i \):

b. Compute the nutrient media (cost function) value \( J(i,j,l) \).
   Calculate \( J(i,j,l) = (j(i,j,l) + J_c(\theta(j,k,l),P(j,k,l)) \) if there is
   no swarming effect than \( J_c(\theta(j,k,l),P(j,k,l)) = 0 \)

c. Put \( J_{last} = J(i,j,k,l) \) to save this value since a better cost via
   run my be found

d. Tumble: generate a random vector \( \Delta(i) \in \mathbb{R}^p \) which each
   element \( \Delta_{mp}(i), m_p 1,2,\ldots,p \), a random number on the
   range \([-1,1]\]

e. Move: compute \( \theta'(j+1,k,l) = \theta'(j,k,l) + C(i) \times \frac{\Delta(i)}{\sqrt{\Delta'(i) \times \Delta(i)}} \)
   This result in a step of size \( C(i) \) in the direction of the tumble for
   bacterium \( i \).

f. Compute the nutrient media (cost function) value \( J(i,j+1,k,l) \),
   and calculate
   \( J(i,j+1,k,l) = J(i,j+1,k,l) + J_c(\theta'(j+1,k,l),P(j+1,k,l)) \).
   If there is no swarming effect then \( J_c(\theta'(j+1,k,l),P(j+1,k,l)) = 0 \)

g. swim
   i. Put \( m_s = 0 \) (counter for swim length)
   ii. While \( m_s < \) (if have not climbed down too long)
      count \( m_s = m_s + 1 \)

If \( J(i,j+1,k,l) < J_{last} \) then \( J_{last} = J(i,j+1,k,l) \) and
calculate $\theta^i(j+1,k,l) = \theta^i(j,k,l) + C(i) \times \frac{\Delta(i)}{\sqrt{\Delta^T(i) \times \Delta(i)}}$ 

This result in a step of size C (i) in the direction of the tumble for bacterium i. Use this $\theta^i(j+1,k,l)$ as in sub step f above.

Else, m=Ns

h. Go to next bacterium (i+1) if $i \neq S$ to process the next bacterium.

Step 5 If $J < N_c$, go to step 4.

Step 6 Reproduction :

a. For the given k and I, and for each $i=1,2,3\ldots\ldots S$, let

$$J_{health}^i = \sum_{j=1}^{N_{ij}} J(i,j,l)$$

Be the health of bacterium i. Sort bacteria and chemotactic parameter C (i) in order of ascending cost $J_{health}$

b. the $S_r$ bacteria with the highest $J_{health}$ values die and the other $S_r$ bacteria with the best values split

Step 7 If $k < N_{re}$, go to step 3.

Step 8 Elimination - dispersal: for $i=1,2,3\ldots\ldots S$.

For $m=1:S$

If $p_{ed} > \text{rand}$ (Generate random number for each bacterium and if the generated number is smaller than $p_{ed}$ then eliminate positions for bacterium)
Generate new random position bacteria

else

Bacteria keep their current position (bacteria are not dispersed)

end

end

Step 9 If $1 < N_{cd}$, then go to step 2; otherwise end.