Biomimicry of Bacterial Foraging
for Distributed Optimization and Control

Kevin M. Passino[1]
Presented by: Alexander Van de Kleut[2]

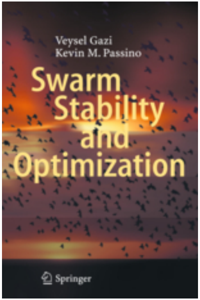
[1]The Ohio State University
Electrical and Computer Engineering

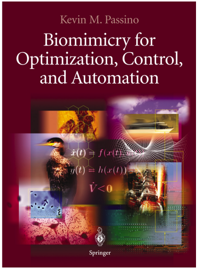
[2]University of Waterloo
Centre for Theoretical Neuroscience

IEEE Control Systems Magazine, 2002

Table of Contents

About the Author





About the Author

Fuzzy control KM Passino, S Yurkovich, M Reinfrank Addison-wesley 42, 15-21, 1998	3599
Biomimicry of bacterial foraging for distributed optimization and control KM Passino IEEE control systems magazine 22 (3), 52-67, 2002	3023
Stability analysis of swarms V Gazi, KM Passino IEEE transactions on automatic control 48 (4), 692-697, 2003	1125
Stable adaptive control using fuzzy systems and neural networks JT Spooner, KM Passino IEEE Transactions on Fuzzy Systems 4 (3), 339-359, 1996	728
Stability analysis of social foraging swarms V Gazi, KM Passino IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 34 …, 2004	710

Foraging

Foraging

- searching for nutrients
- avoiding noxious stimuli (toxins, predators, etc)

Social Foraging

- increases likelihood of finding nutrients
- better detection and protection from noxious stimuli
- gains can offset cost of food competition

Foraging as Optimization

How can we view foraging as an Optimization Process?

- We have some parameters θ and a loss function $J(\theta)$ that we want to minimize

Foraging as Optimization

How can we view foraging as an Optimization Process?

- We have some parameters θ and a loss function $J(\theta)$ that we want to minimize
- θ can represent the position of an organism in its environment

Foraging as Optimization

How can we view foraging as an Optimization Process?

- We have some parameters θ and a loss function $J(\theta)$ that we want to minimize
- θ can represent the position of an organism in its environment
- J can represent the concentration of nutrients and noxious stimuli
  - smaller values of J = more nutrients, less noxious stimuli
  - higher values of J = more noxious stimuli, less nutrients

Foraging as Optimization

How can we view foraging as an Optimization Process?

- We have some parameters θ and a loss function $J(\theta)$ that we want to minimize
- θ can represent the position of an organism in its environment
- J can represent the concentration of nutrients and noxious stimuli
  - smaller values of J = more nutrients, less noxious stimuli
  - higher values of J = more noxious stimuli, less nutrients
- In general, J and θ can be arbitrary
  - $\theta \in \mathbb{R}^p$
  - $J : \mathbb{R}^p \to \mathbb{R}$

E. coli

- Model organism

E. coli

- Model organism
  - Highly studied

E. coli

- Model organism
  - Highly studied
  - Well-characterized foraging behaviour

E. coli

- Model organism
  - Highly studied
  - Well-characterized foraging behaviour
  - Probably won't feel bad about simplifying its behaviour

E. coli

- Model organism
  - Highly studied
  - Well-characterized foraging behaviour
  - Probably won't feel bad about simplifying its behaviour
- Social organism

E. coli

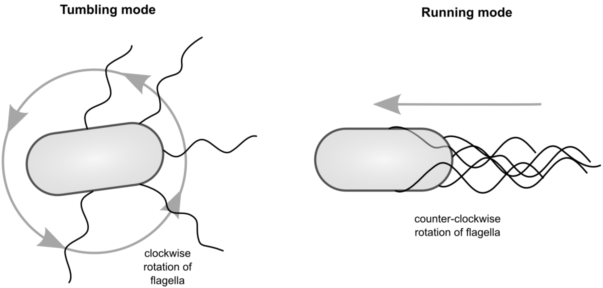
- Model organism
  - Highly studied
  - Well-characterized foraging behaviour
  - Probably won't feel bad about simplifying its behaviour
- Social organism
  - Secretes signals to attract others nearby

E. coli

- Model organism
  - Highly studied
  - Well-characterized foraging behaviour
  - Probably won't feel bad about simplifying its behaviour
- Social organism
  - Secretes signals to attract others nearby
  - Encourages "swarming" or "clumping"

E. coli Behaviour

- Swims using left-handed helical flagella ("propellers")
  - **Tumble**: flagella all rotate clockwise → pull on cell in all directions → random movement
  - **Run**: flagella all rotate counterclockwise → flagella form a bundle → push on cell in one direction → directed movement



E. coli Behaviour

- If during a tumble *E. coli* swims down a nutrient concentration gradient:
  - Prolongs time spent on a run
  - Continues moving in the same direction
- Otherwise:
  - Tends to switch to a tumble (search for more)
  - Moves randomly which searching for more nutrient gradients to exploit
- Call a tumble followed by a run a "chemotaxis step"

Algorithm for a Single Bacterium

for $j \leftarrow 1 \ldots N_c$ **do**:

  $\phi \sim \mathcal{U}$

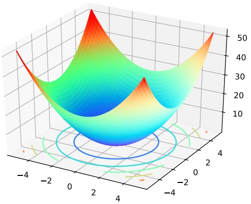
  $\theta \leftarrow \theta + c\phi$

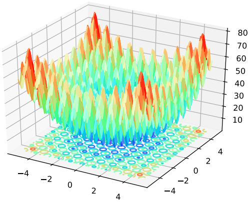
  while $J(\theta + c\phi) < J(\theta)$ **do**:

    $\theta \leftarrow \theta + c\phi$

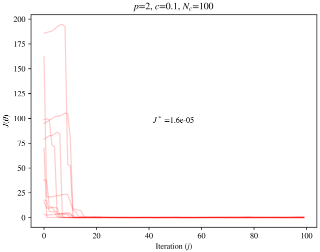
- θ: p-dimensional vector (randomly initialized)
- N_c: number of chemotaxis steps
- $\phi \sim \mathcal{U}$: a random unit vector
- c: a step-size

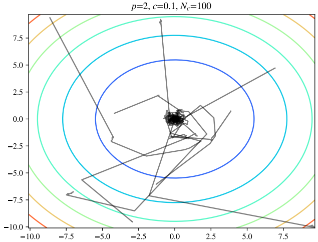
Loss Function to Optimize





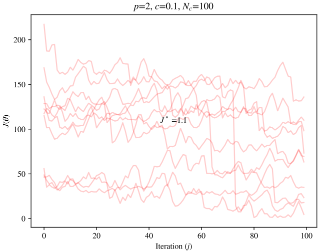
Results of Single Bacterium

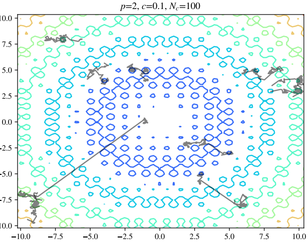




- Relatively consistent performance for a convex function.

Results of Single Bacterium





- Relatively inconsistent performance for a highly nonconvex function.

Algorithm for a Colony

for $j \leftarrow 1 \ldots N_c$ **do**:
  for $i \leftarrow 1 \ldots S$ **do**:
    $\phi \sim \mathcal{U}$
    $\theta_i \leftarrow \theta_i + c_i\phi$
    while $J(\theta_i + c_i\phi) + J_{cc}(\theta_i + c_i\phi) < J(\theta_i) + J_{cc}(\theta_i)$ **do**:
      $\theta_i \leftarrow \theta_i + c_i\phi$

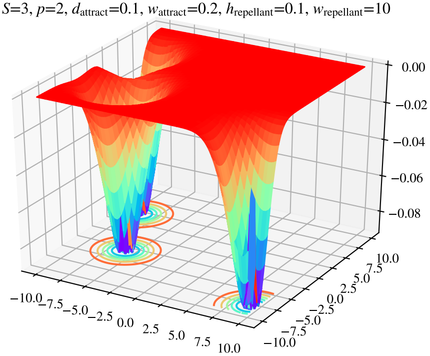
- θ_i: ith p-dimensional vector (randomly initialized)
- S: number of bacteria in the colony
- c_i: a step-size for bacterium i

J_{cc} and swarming behaviour

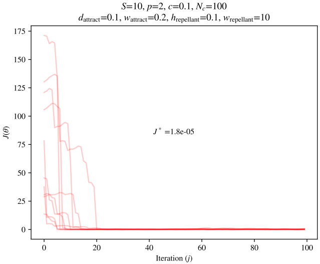
- *E. coli* do social foraging
- Secrete a substance to indicate to attract nearby *E. coli* and encourage swarming
- Strength of signal diffuses over space
- Use gaussian distribution to model this

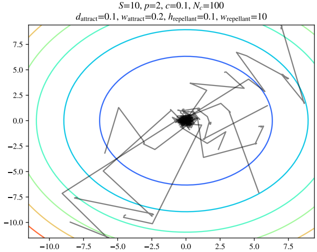
$$J_{cc}(\theta) = \sum_{i=1}^{S} - d_{\text{attract}} \exp\left(-w_{\text{attract}}(\theta - \theta_i)^T(\theta - \theta_i)\right) \\ + h_{\text{repellant}} \exp\left(-w_{\text{repellant}}(\theta - \theta_i)^T(\theta - \theta_i)\right)$$

J_{cc} and swarming behaviour



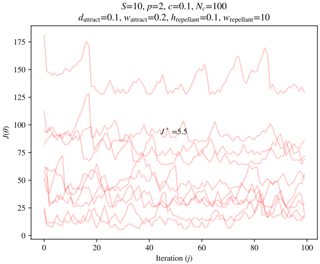
Results of Single Bacterium

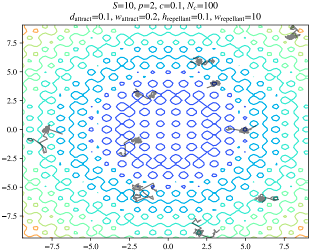




- Still relatively consistent performance for a convex function.
- Achieves similar performance to single bacterium.

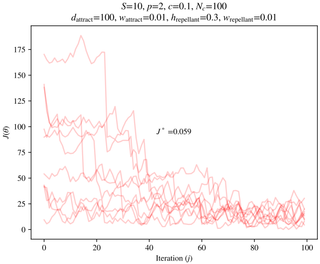
Results of Single Bacterium

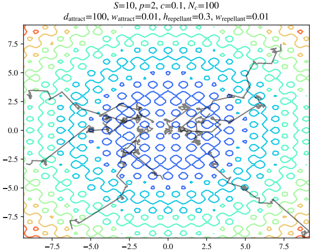




- Still relatively inconsistent performance for a highly nonconvex function.
- But wait... What if the problem is just the hyperparameters?

Results of Single Bacterium





- By trying out different combinations of hyperparameters we can improve overall performance.
- Here we increased the depth and width of attraction as well as the depth and width of repellance to increase "global" behaviour.

Algorithm for a Reproducing Colony

for $k \leftarrow 1 \dots N_{re}$ **do**:
  for $j \leftarrow 1 \dots N_c$ **do**:
    for $i \leftarrow 1 \dots S$ **do**:
      $\phi \sim \mathcal{U}$
      $\theta_i \leftarrow \theta_i + c_i\phi$
      while $J(\theta_i + c_i\phi) + J_{cc}(\theta_i + c_i\phi) < J(\theta_i) + J_{cc}(\theta_i)$ **do**:
        $\theta_i \leftarrow \theta_i + c_i\phi$
  delete worst $S/2$ and reproduce best $S/2$

- N_{re}: number of reproduction steps

Algorithm for a Reproducing Colony