

Efficient Modeling and Simulation of Bacteria-based Nanonetworks with BNSim

Guopeng Wei, Paul Bogdan, and Radu Marculescu

Email: (guopengw, pbogdan, radum@ece.cmu.edu)

Supplementary File	Title
Supplementary Note 1	Validation of BNSim output with chemoreceptor adaptation
Supplementary Note 2	Validation of BNSim output with chemoreceptor dose response
Supplementary Note 3	Validation of BNSim output with fast phosphorylation cascade
Supplementary Note 4	Validation of BNSim output with two state flagella motor
Supplementary Note 5	Performance and memory scaling test with chemotaxis model
Supplementary Note 6	Overview of stiffness in biological systems and multiscale stochastic simulation algorithm
Supplementary Note 7	Model of bacteria swimming at Low Reynolds number
Supplementary Note 8	Overview of BNSim software structure
Supplementary Note 9	Brownian Motion and Levy walk
Supplementary Note 10	Equations for the genetic circuits
Supplementary Note 11	Modeling parameters

Supplementary Note 1

Validation of BNSim output with chemoreceptor precise adaptation

In order to capture the essential methylation and demethylation noise, we model 370 receptor clusters individually. Each receptor cluster has 13 Tsr receptors and 6 Tar receptors. The methylation level of each receptor has a range from 0 to 8. The methylation and demethylation dynamics is then simulated using exact stochastic simulation algorithm. More precisely, similar to the DOR reactions of NFSim¹, each receptor cluster participates in the methylation and demethylation reaction rules² with different rates, depending on their activity. In every step, one receptor inside one of the 370 receptor clusters is selected to be methylated or demethylated by CheR or CheB. However, we do not model CheR and CheB binding reactions explicitly. One reason for this is that these reactions are computationally too expensive to be explicitly modeled at population level. Another reason is because the high frequency component of the binding noise induced by CheR and CheB is filtered out by the slow methylation and demethylation kinetics due to its large relaxation time³. Instead, we use a Wiener process to describe the fluctuation of CheR⁴. More specifically, the fluctuating term is added as

$$dR = \beta \cdot d\eta(t) \quad (1)$$

where β is the noise parameter and $\eta(t)$ is the Wiener process. Simulation results with different value of β are available in Supplementary Note 3.

The activity of each individual receptor cluster depends on its methylation level and ligand occupancy. In this paper, it is calculated according to the “*assistance neighborhoods*” model⁵.

$$A = \frac{e^{-F^{on}}}{e^{-F^{on}} + e^{-F^{off}}} = \frac{1}{1 + e^F} \quad (2)$$

where F is the sum of the individual free-energy differences between the receptor *on* and *off* states. Since the entire receptor cluster is considered to be either in an *on* or *off* state, A represents the average activity of the receptors in the cluster. The *offset* energy of receptor at different methylation levels follows the original “*assistance neighborhoods*” model. Please refer to the BNSim simulation results of bacterial precise adaptation reproduced in Figure S1 from the original reference⁵.

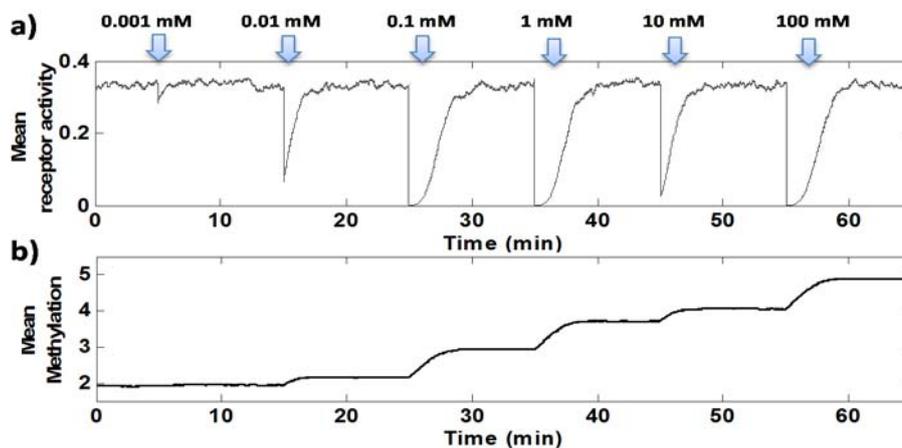


Figure S1. Bacteria precise adaptation

Supplementary Note 2

Validation of BNSim output with chemoreceptor dose response

Bacteria accurate response to various chemical gradients is of critical importance in chemotactic swimming. The dose response curve for wild-type *E. coli* is obtained by adding a certain level of MeAsp to the environment with a single bacterium. The chemical pathway of the bacterium is already in equilibrium in the nutrient-free environment before adding stimuli. The response curve shows how much the receptors are suppressed when adding a certain amount of stimuli. The BNSim simulation results (shown in Figure S2b below) shows the effectiveness of our simplified receptor cluster model and is consistent with the experimental result⁶ (shown in Figure S2).

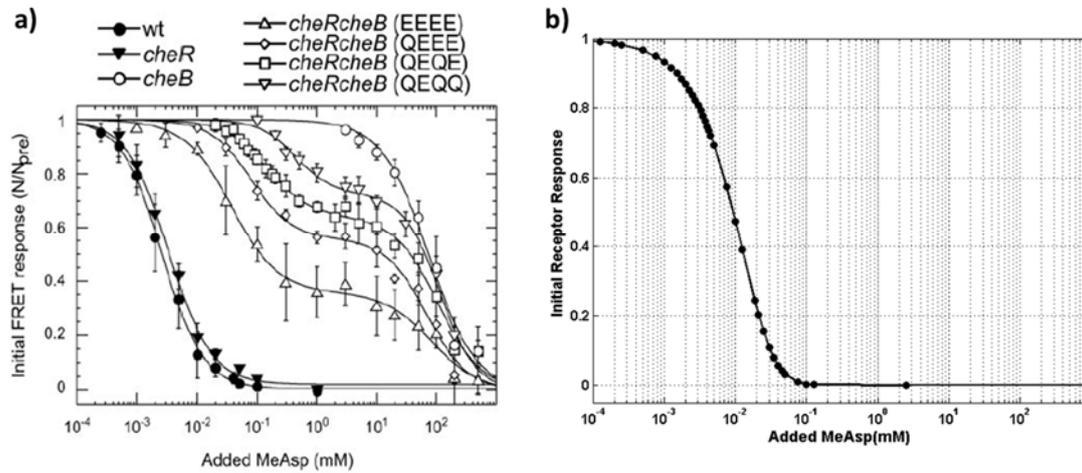


Figure S2. Chemoreceptor dose response

Supplementary Note 3

Validation of BNSim output with fast phosphorylation cascade

The phosphorylation relay in our work is similar to the one in the work of Sourjik and Berg⁶. The signaling protein CheY is phosphorylated by the kinase CheA, then diffuses away to downstream pathway flagella motor. The binding of CheY-P with motor complex regulates the motor rotation direction. The dephosphorylation of CheY-P is accelerated by CheZ.

As it can be observed from the Figure S3, the trend of CheY-P fluctuation faithfully reflects the fluctuation of CheR. The value of β in the Wiener process $dR = \beta \cdot d\eta(t)$ determines the fluctuations of CheR, and thus determines the switching activity of the flagella motor. As indicated in reference⁴, since the methylation process has a certain relaxation time, the chemotaxis signaling pathway functions like a low-pass filter of the CheR fluctuations. Therefore, to change the switching frequency significantly, the internal concentration of CheY-P must have enough time to adapt to a change in the level of CheR; this is only possible if the level of CheR stays above or below the average value for a longer time. Figure S3 shows the fluctuation of CheY-P with different CheR fluctuations, this translates directly into different exponential distributions of the bacteria run length. In Figure S3d below, the values of CheY-P are much lower in the time window 10min to 50min, which results in much lower CW bias as shown in the figure of **Supplementary Note 4**.

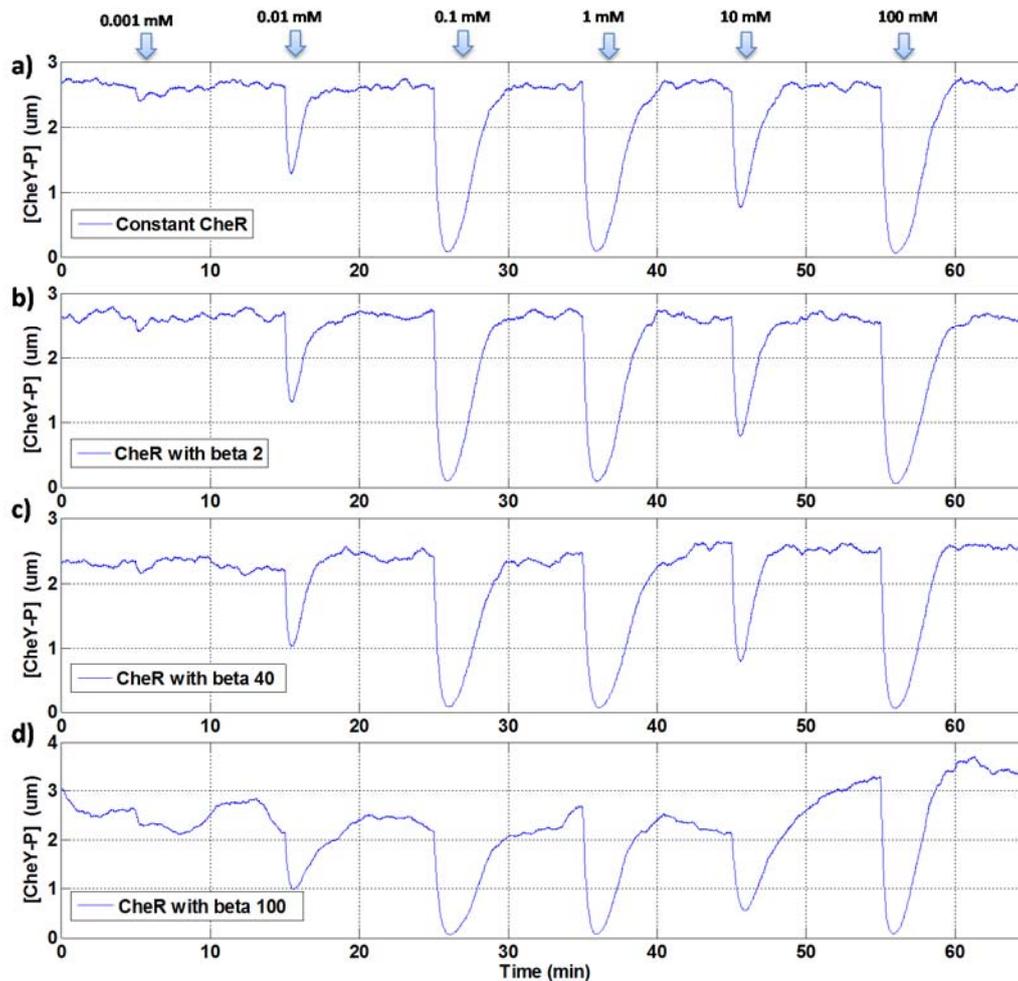


Figure S3. CheY-P level as a function of CheR fluctuation

Supplementary Note 4

Validation of BNSim output with two state flagella motor

Thermal fluctuations and upstream signaling cause the flagella motor to spontaneously transit between alternating rotational states. In this work, we use a coarse-grained, two-state model that captures the key dynamic behavior of the motor response. In other words, we treat the flagella motor as a two-state model with each state, namely clockwise (CW) and counterclockwise (CCW) sitting in a potential well⁷. The energy barriers for CCW to CW and CW to CCW are $\Delta G_0(Y_p(t))$ and $-\Delta G_0(Y_p(t))$, respectively. Here we use the definition of $\Delta G_0(Y_p(t))$ from reference¹ (Note that other definitions of $\Delta G_0(Y_p(t))$ are possible, such as reference⁸), specifically

$$\Delta G_0(Y_p(t)) = \frac{g_0}{4} - \frac{g_1}{2} \left(\frac{Y_p(t)}{K_D + Y_p(t)} \right) \quad (3)$$

The switching rates from CW to CCW and from CCW to CW are then

$$k^+ = w_0 \times \exp(\Delta G_0(Y_p(t))) \quad (4)$$

$$k^- = w_0 \times \exp(-\Delta G_0(Y_p(t))) \quad (5)$$

Figure S4 shows the simulation results for CW bias and motor switching frequency as function of CheY-P concentration. The parameters are tuned to match the experimental results⁹ shown in the figure.

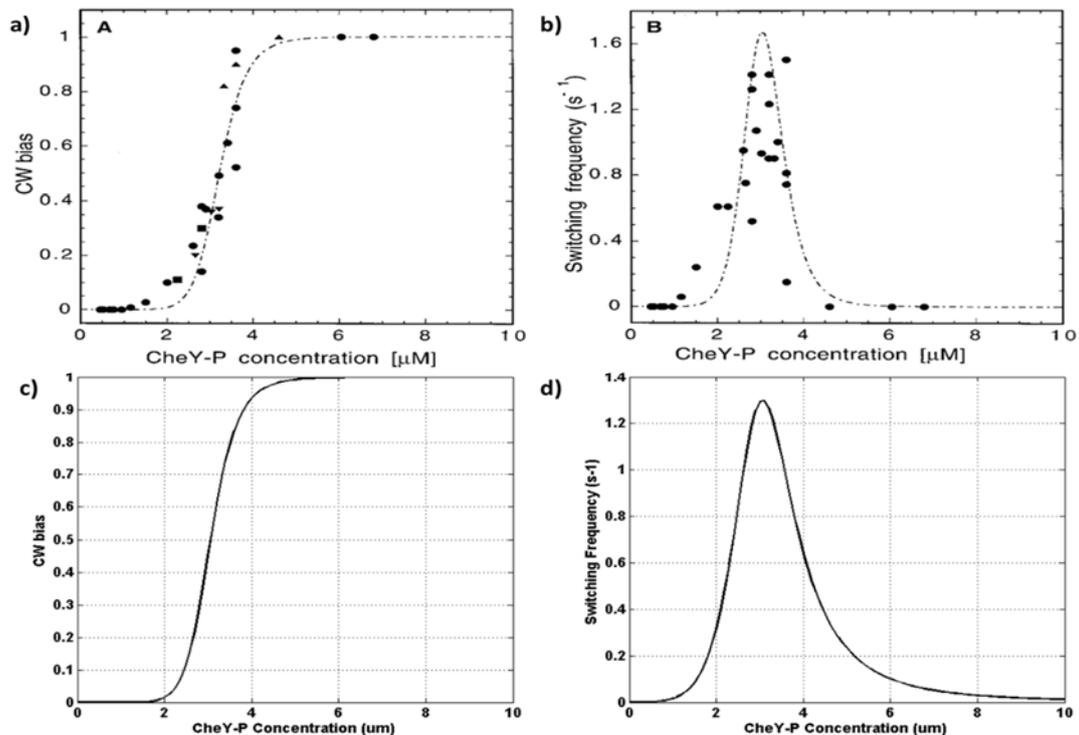


Figure S4. CW bias and motor switching frequency as function of CheY-P concentration

Supplementary Note 5

Performance and memory scaling tests for the chemotaxis model

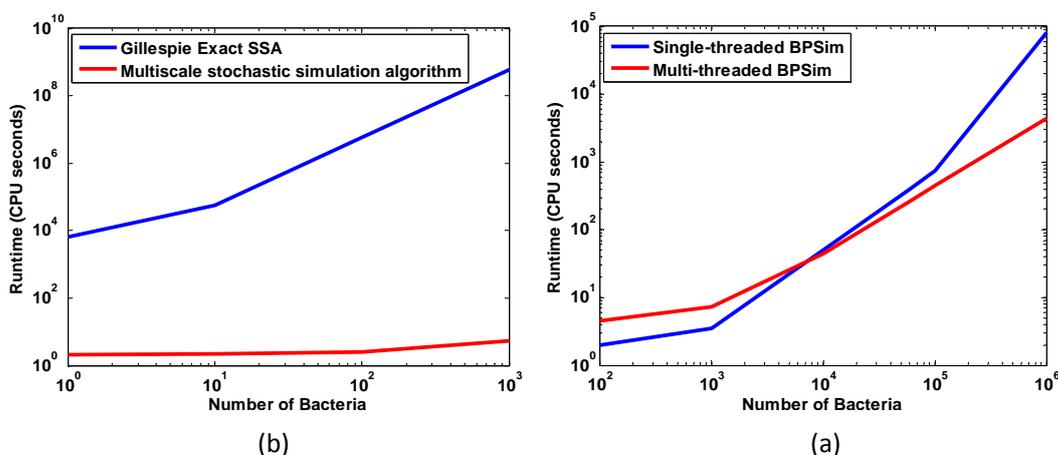
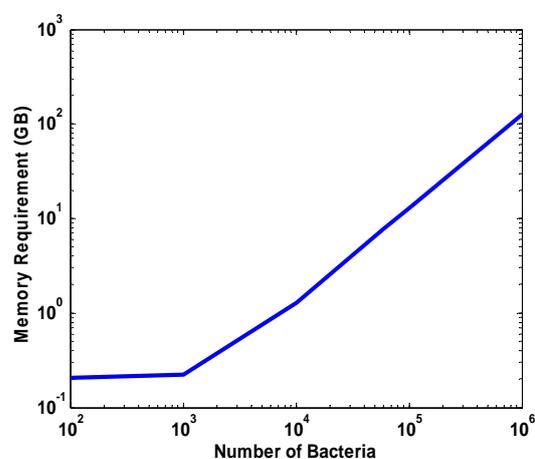


Figure S5. Performance scaling as a function of bacteria number

Figure S5 shows the runtime (CPU seconds) of Gillespie's exact stochastic simulation versus multiscale stochastic simulation as function of bacteria population size. Since the simulation using Gillespie's Exact SSA is extremely time consuming, we compare simulations involving up to 10^3 bacteria for the first experiment. As expected, the multiscale stochastic simulation scheme accelerates the time evolution of the chemical system by several orders of magnitude. At the same time, the Gillespie's Exact SSA algorithm scales much slower than the multiscale stochastic simulation.

On the other hand, the multithreading feature of BNSim allows simulation of a large population of bacteria efficiently, while the performance of single-threaded BNSim starts to drop dramatically after considering only 10^3 bacteria. However, we also observe that the multithreaded BNSim does not simulate a small number of bacteria efficiently mainly because of the synchronization overhead introduced when compared to single-threaded BNSim. Therefore, BNSim allows users to enable or disable the multithreading feature as needed in order to get the highest efficiency in simulation. Also, BNSim allows users to specify the number of threads to be used. However, the maximum number of threads that can be used depends on how much CPU is available for BNSim.



Memory is another important issue when simulating a large population of bacteria. However, BNSim does not model every molecule as an independent object for non-key subsystems, but rather relies on continuous variables. This feature makes BNSim consume much less memory compared to exact stochastic simulation. In practice, an average desktop computer with 4GB memory supports approximately $4 \cdot 10^4$ bacteria population simulation. To simulate up to 10^7 bacteria requires resources that

can be provided only by a super computer center, or by cloud computing services.

Supplementary Note 6

Overview of stiffness in biological systems and multiscale stochastic simulation algorithm

Complex biochemical systems typically involve tens or even hundreds of reactants participating in a multitude of reactions, which occur on a broad spectrum of time scales ranging from microseconds to seconds¹⁰. In such systems involving multiple time scales, the fastest reactions are often found to be dependent on the slower ones. More precisely, the theory of near decomposability¹¹ indicates that a complex system consists of small subsystems that can perform specific functions without too much interaction with the rest of the system for short interval of times. In other words, the short-run behavior of each subsystem is approximately independent of the short-run behavior of the other components. Consequently, the complex systems satisfying the near decomposability property reduce the interdependence of components, enhance robustness, facilitate adaptation and evolution^{12, 13}. Such hierarchy is clearly visible in biological systems¹⁴, and has been regarded as one of the biology's first laws¹⁵.

The bacterial chemotaxis is a typical stiff chemical system which contains slow and fast dynamics. More precisely, the slow dynamics are the methylation and demethylation kinetics of receptor clusters by CheR and CheB. The fast dynamics is the phosphorylation relay in the signaling pathway, which involves several thousands of molecules and has a high reaction rate. Hence, it becomes impractical to simulate the full system using exact stochastic simulation for bacteria population. In practice, the exact method needs to keep track of up to 10^6 reaction events for chemotaxis in every second of simulation time. The time interval chosen by exact stochastic simulation algorithm for methylation and demethylation kinetics only is around $10^{-2} \sim 10^{-3}$ s, while the time interval for the full chemotaxis system is around $10^{-5} \sim 10^{-9}$ s. On the other hand, the memory requirements to represent the molecules, reactions, and the internal data-structure for a single bacterium may take up to 10^4 bytes. Thus, $10^4 \sim 10^7$ bacteria population size also makes this approach impractical in terms of space complexity.

Therefore, if we can take advantage of the hierarchy existing in biological system, identify the key subsystem, and simulate the key subsystem according to the exact stochastic simulation method¹⁶ (while use accelerate approximation methods for other fast subsystems), then we can get a speed up of orders of magnitudes compared to exact stochastic simulation algorithm¹⁶.

Take chemotaxis as an example, at each time step, we choose time interval τ using (6)

$$\tau = \frac{1}{a_0(x)} \ln \frac{1}{r} \quad (6)$$

where r is a random number and $a_0(x)$ is the propensity function of all reactions of slow dynamics (adaptive chemoreceptor subsystem), as defined in the Gillespie's algorithm. In each time interval τ , one methylation and demethylation related reaction takes place. We use *tau-leaping* or Langevin approximation¹⁷ to simulate the phosphorylation relay, and a course-grain function to describe the motor switching activity. To approximate the phosphorylation relay, we check whether the expected number of occurrences of reaction $n = \tau \times a$ is much larger than 1, where a is the propensity

function of fast-dynamics according to (7).

$$a_j(x) = k_j x_1 x_2 \quad (7)$$

where $a_j(x)$ is the propensity for reaction channel R_j , k_j is the reaction constant for R_j , x_1 is the copy number of the first reactant, x_2 is the copy number of the second reactant.

If the reaction number is much larger than 1, then we use Langevin method¹⁷ (as shown in eq. (9) below) to approximate the number of reactions that is likely to happen in time τ ; otherwise, we use Poisson approximation¹⁸ as in (8):

$$X(t + \tau) = x + \sum_{j=1}^M P_j(a_j(x)\tau) v_j \quad (8)$$

where $X(t)$ is the state vector, $P_j(a_j(x)\tau)$ is a statistically independent Poisson random variable

with mean $a_j(x)\tau$. v_j is the state change vector for reaction channel R_j .

$$X(t + \tau) = x + \sum_{j=1}^M v_j (a_j(x)\tau) + \sum_{j=1}^M v_j \sqrt{a_j(x)} N_j(0,1) \sqrt{\tau} \quad (9)$$

where $N_j(0,1)$ is a normal (Gaussian) random variable.

Table 1 Multiscale stochastic simulation

<p>Multiscale stochastic simulation algorithm</p> <p>a. Choose time step $\tau = \frac{1}{a_0(x)} \ln \frac{1}{r}$ for key subsystem according to exact stochastic simulation algorithm</p> <p>b. At each time step, calculate the propensity function for other subsystems according to</p> $a_j(x) = k_j x_1 x_2$ <p>Then the number of expected reactions is $n = \tau \times a_j(x)$</p> <p>c. If the reaction number is much larger than 1, use <i>Langevin</i> method to simulate the fast dynamics</p> $X(t + \tau) = x + \sum_{j=1}^M v_j (a_j(x)\tau) + \sum_{j=1}^M v_j \sqrt{a_j(x)} N_j(0,1) \sqrt{\tau}$ <p>d. If the reaction number is not much larger than 1, use <i>tau-leaping</i> method to simulate the fast dynamics</p> $X(t + \tau) = x + \sum_{j=1}^M P_j(a_j(x)\tau) v_j$ <p>e. Go to step a if the simulation is not finished</p>

Supplementary Note 7

Model of Swimming

Bacteria use runs and tumbles to effectively swim in liquids characterized by low Reynolds numbers¹⁹. Berg and Brown in 1972 tracked individual cells in 3D²⁰. To be consistent with their experimental results, we assume that bacteria speed is constant during runs and the direction of a run is affected by the rotational Brownian diffusion. When bacteria tumble, the direction change is chosen from a Gamma distribution which fits the distribution of tumbling angles measured by Berg and Brown²⁰ (see figure below).

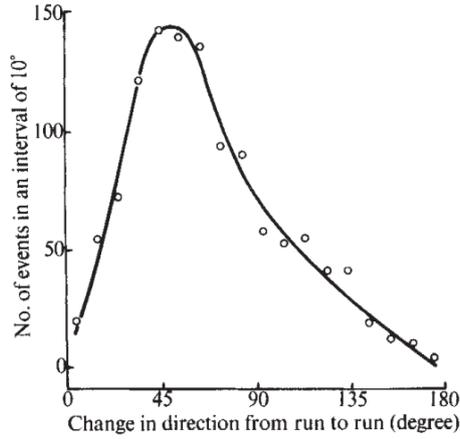


Fig. 3 Distribution of changes in direction from the end of one run to the beginning of the next for the wild type bacteria of Table 1. The distribution was constructed from 1,166 events by summing the numbers falling in successive 10° intervals. If the analysis is confined to the shortest twiddles, the distribution is skewed even farther toward small angles (mean and standard deviation $62 \pm 26^\circ$).

The rotational frictional drag coefficient²¹ for a sphere of radius R is calculated as,

$$f_{r,sphere} = 8\pi\eta R^3 \quad (10)$$

where η is the viscosity of the environment.

The rotation angle \mathcal{G} is therefore

$$\mathcal{G} = 4kT/f_{r,sphere} \quad (11)$$

where k is Boltzmann constant, and T is the temperature of the environment.

Besides rotational diffusion, we also consider collisions between bacteria, and collisions between bacteria and the reflective boundaries of the environment. This is important because the collisions between bacteria themselves and bacteria with reflective boundaries may change their forwarding direction dramatically. Typically, the bundled flagella filaments of a bacterium fall apart, hence the bacterium tumbles and chooses a new direction.

Supplementary Note 8

Extended description of the multithreading multiscale time evolution

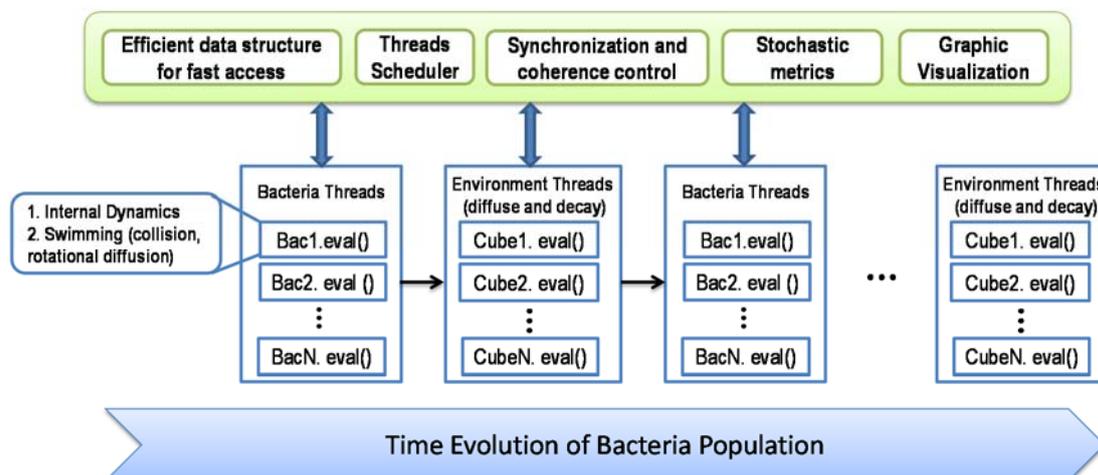


Figure S8. Parallel structure of BNSim

As shown in the figure, the time evolution of the stochastic simulation follows a tick-tock fashion. In “tick”, bacteria sense the environment, update their internal dynamic, and swim in the environment. In “tocks”, the chemicals diffuse and decay in the environment. This is done by exploiting the multithreading feature of BNSim. More precisely, at “ticks”, the threads scheduler uses two cyclic barriers to synchronize all the threads therefore ensuring that all bacteria are active in the same “time space”. The “cubes” of the environment (explained below) are updated in a similar manner at “tocks”.

In order to capture bacteria interaction with nearby bacteria and environment efficiently, we consider cube-based partitions. The 3D space is therefore tessellated in cubes of a typical size $100 \times 100 \times 100 \mu\text{m}$, while *E. coli* is a sphere with a radius of $1 \mu\text{m}$. One data element storing all references of living and non-living objects in the cube local environment is associated with each cube; these data elements can therefore be accessed very efficiently according to their indexes as a function of their absolute position in the global environment. This way, we can ensure that interactions between bacterium and environment, also interactions among bacteria themselves can be efficiently simulated. Of note, to ensure the correctness of the result, access to each cube needs to happen in a mutually exclusive manner so each bacterium needs to obtain the corresponding mutex beforehand.

BNSim collects various statistical metrics during simulation and provides a better insight into the dynamics of bacteria population. More specifically, BNSim records the distribution of the bacteria, concentration of certain chemical species in the 2D/3D space, or trajectories of particular bacteria. BNSim also records statistics such as the mean square displacement, hitting time to a certain targeted region, etc. These statistics are processed automatically and finally plotted using Matlab scripts. Moreover, in order to show the time evolution of the bacteria population more intuitively during simulation, we also provide means to visualize the motion of bacteria population in a 3D GUI.

Supplementary Note 9

Brownian motion and Levy walk

In a nutrient free environment, the fluctuations of CheR propagate to the regulatory protein CheY, and results in a power-law distribution of run intervals and an exponential distribution of the tumble intervals^{4, 7, 22}. The extended runs allow bacteria to perform a Levy walk (see Figure S10b), instead of Brownian motion (see Figure S10a). Recent work⁴ shows that Levy walk turns out to be a good strategy over the Brownian movement when the food is located randomly and sparsely in an environment. See figures below for BNSim simulation result of single bacterium trajectory with and without CheR fluctuation in nutrient free environment.

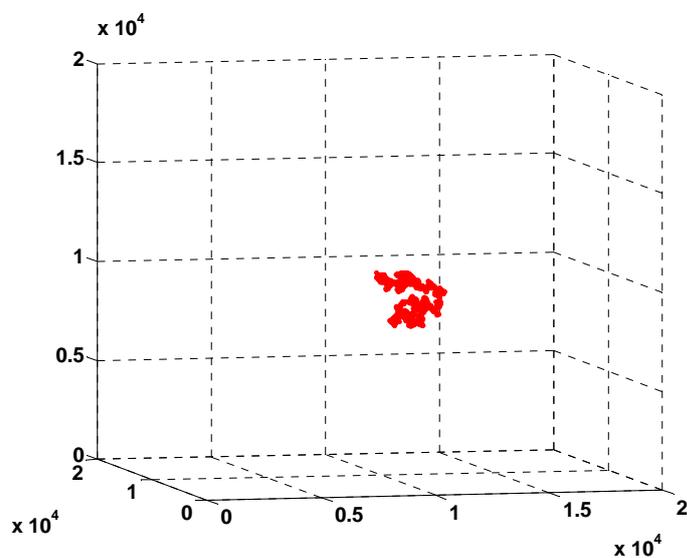


Figure S10a. Single bacterium with constant CheR perform Brownian Motion in nutrient free environment

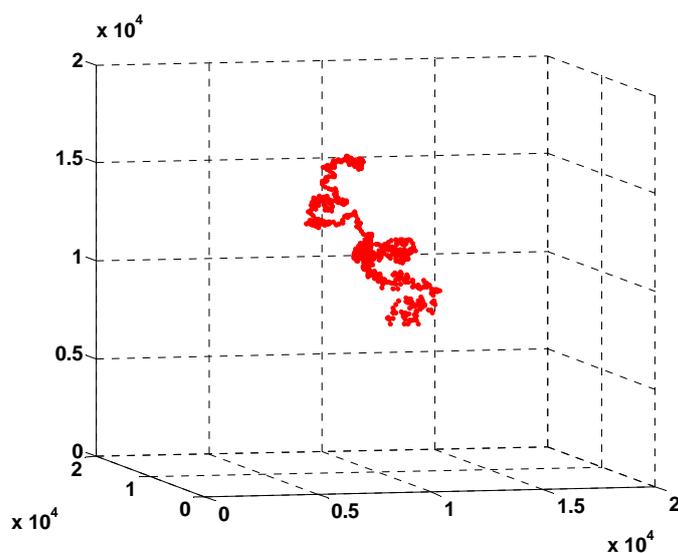


Figure S10b. Single bacterium with fluctuate CheR perform Levy walk in nutrient free environment

Supplementary Note 10

Equations for the genetic circuits

As shown in [23][24], for the AND gate in the USC, the authors provide a steady-state transfer function to model the behavior of the AND gate, namely,

$$\frac{A}{A_{max}} = \frac{I_1 I_2^2}{a(b + I_2)^2 + I_1 I_2^2}$$

where A_{max} is the maximum concentration of AHL for the output, I_1 and I_2 are the activity of input promoters T7ptag and supD tRNA, respectively.

Then, AHL molecules diffuse into the DSC through the environment, and the equilibrium concentration of bound transcription factor in DSC is

$$C = C_0 \frac{A^n}{K_d^n + A^n}$$

where C is the concentration of bound transcription factor, C_0 is the total concentration of transcription factor, K_d is the dissociation constant, and n is hill coefficient. Therefore, the activity of P_{luxrep} is,

$$P_{luxrep} = \frac{I}{1 + C/k}$$

where I is the strength of the promoter P_{luxrep} , k is the binding affinity of LuxR to P_{luxrep} . Therefore, for the AND gate in DSC, the two inputs are $I'_1 = P_{luxrep}$ and $I'_2 = I_1 + I_2$.

Finally, the fluorescence level from the output is,

$$\frac{B}{B_{max}} = \frac{I'_1 I'^2_2}{a(b + I'_2)^2 + I'_1 I'^2_2}$$

where B_{max} is the maximum blue-green light intensity observed from the bacterial luciferase (lux system) at the output of DSC.

Parameter	Description	Value
A	Parameter in AND gate, see [24].	50±10
B	Parameter in AND gate, see [24].	3000±1000
K_d	Binding affinity of AHL to LuxR.	0.50
K	Binding affinity of LuxR to plux_rep.	0.03±0.01
N	Hill coefficient to describe the cooperative effect of AHL	2.0±0.3
I	Strength of P_{lux_rep}	100
C₀	Translation strength of <i>luxI</i>	1

Supplementary Note 11
Modeling parameters

Parameter	Value	Parameter	Value
ϵ_0	1.0	ϵ_1	-0.45
$K_{TAR_{offstate}}^{Serine}$	1e5 mM	$K_{TAR_{onstate}}^{Serine}$	1e6 mM
$K_{TSR_{offstate}}^{Serine}$	0.0025 mM	$K_{TSR_{onstate}}^{Serine}$	1 mM
$K_{TAR_{offstate}}^{Aspartate}$	0.02 mM	$K_{TAR_{onstate}}^{Aspartate}$	0.1 mM
$K_{TSR_{offstate}}^{Aspartate}$	100 mM	$K_{TSR_{onstate}}^{Aspartate}$	1e6 mM
σ_m	3	τ	60s
w_0	$1.3s^{-1}$	g_0, g_1	$40k_bT$
K_D	$3.06\mu m$		

TABLE I
PARAMETERS OF THE DETAILED BACTERIA MODEL

References

1. Sneddon, M.W., Faeder, J.R. & Emonet, T. Efficient modeling, simulation and coarse-graining of biological complexity with NFsim. *Nature methods* **8**, 177-183 (2011).
2. Faeder, J.R., Blinov, M.L. & Hlavacek, W.S. Rule-based modeling of biochemical systems with BioNetGen. *Methods Mol Biol* **500**, 113-167 (2009).
3. P Sartori, Y.T. Noise Filtering Strategies in Adaptive Biochemical Signaling Networks. *Journal of Statistical Physics* (2011).
4. Matthaus, F., Mommer, M.S., Curk, T. & Dobnikar, J. On the origin and characteristics of noise-induced Levy walks of E. coli. *PLoS one* **6**, e18623 (2011).
5. Endres, R.G. & Wingreen, N.S. Precise adaptation in bacterial chemotaxis through "assistance neighborhoods". *Proceedings of the National Academy of Sciences of the United States of America* **103**, 13040-13044 (2006).
6. Sourjik, V. & Berg, H.C. Receptor sensitivity in bacterial chemotaxis. *Proceedings of the National Academy of Sciences of the United States of America* **99**, 123-127 (2002).
7. Tu, Y. & Grinstein, G. How White Noise Generates Power-Law Switching in Bacterial Flagellar Motors. *Physical Review Letters* **94**, 208101 (2005).
8. Park, H., Oikonomou, P., Guet, C.C. & Cluzel, P. Noise underlies switching behavior of the bacterial flagellum. *Biophysical journal* **101**, 2336-2340 (2011).
9. Zagaris, A., Kaper, H.G. & Kaper, T.J. Analysis of the Computational Singular Perturbation Reduction Method for Chemical Kinetics. *Journal of Nonlinear Science* **14**, 59-91 (2004).
10. Budrene, E.O. & Berg, H.C. Dynamics of formation of symmetrical patterns by chemotactic bacteria. *Nature* **376**, 49-53 (1995).
11. Simon, H.A. The Architecture of Complexity. *Proceedings Of The American Philosophical Society* **106**, 467-482 (1962).
12. Wagner, G.P.A., L. Complex Adaptations and the Evolution of Evolvability. *Evolution*, **50(203)**, 1-24 (1996).
13. Felix, M.A. & Wagner, A. Robustness and evolution: concepts, insights and challenges from a developmental model system. *Heredity* **100**, 132-140 (2008).
14. Bassett, D.S. *et al.* Dynamic reconfiguration of human brain networks during learning. *Proceedings of the National Academy of Sciences of the United States of America* **108**, 7641-7646 (2011).
15. McShea, D.W. & Brandon, R.N. *Biology's first law : the tendency for diversity and complexity to increase in evolutionary systems.* (University of Chicago Press, Chicago ; London; 2010).
16. Gillespie, D.T. Exact Stochastic Simulation of Coupled Chemical-Reactions. *J Phys Chem-US* **81**, 2340-2361 (1977).
17. Gillespie, D.T. The chemical Langevin equation. *Journal of Chemical Physics* **113**, 297-306 (2000).
18. Gillespie, D.T. Approximate accelerated stochastic simulation of chemically reacting systems. *Journal of Chemical Physics* **115**, 1716-1733 (2001).
19. Berg, H.C. Motile behavior of bacteria. *Phys Today* **53**, 24-29 (2000).
20. Berg, H.C. & Brown, D.A. Chemotaxis in Escherichia coli analysed by three-dimensional tracking. *Nature* **239**, 500-504 (1972).
21. Berg, H.C. *Random Walks in Biology.* (Princeton University Press, 1993).
22. Korobkova, E., Emonet, T., Vilar, J.M.G., Shimizu, T.S. & Cluzel, P. From molecular noise to

behavioural variability in a single bacterium. *Nature* **428**, 574-578 (2004).

23. W. Ji, H. Shi, H. Zhang, R. Sun, J. Xi, D. Wen, J. Feng, Y. Chen, X. Qin, Y. Ma et al., "A formalized design process for bacterial consortia that perform logic computing," *PLOS ONE*, vol. 8, no. 2, p. e57482, 2013
24. J. C. Anderson, C. A. Voigt, and A. P. Arkin, "Environmental signal integration by a modular and gate," *Molecular systems biology*, vol. 3, no. 1, 2007.