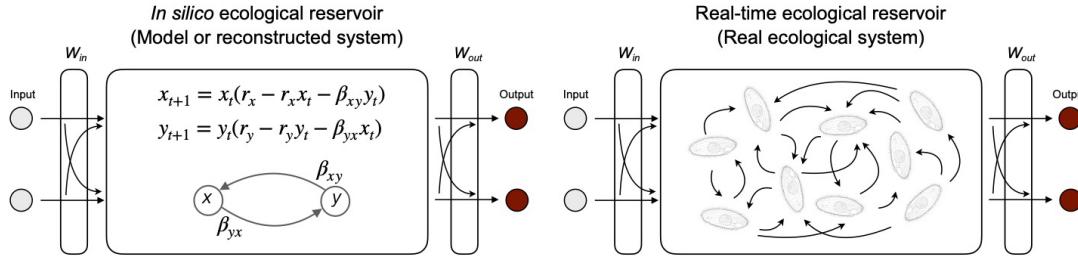
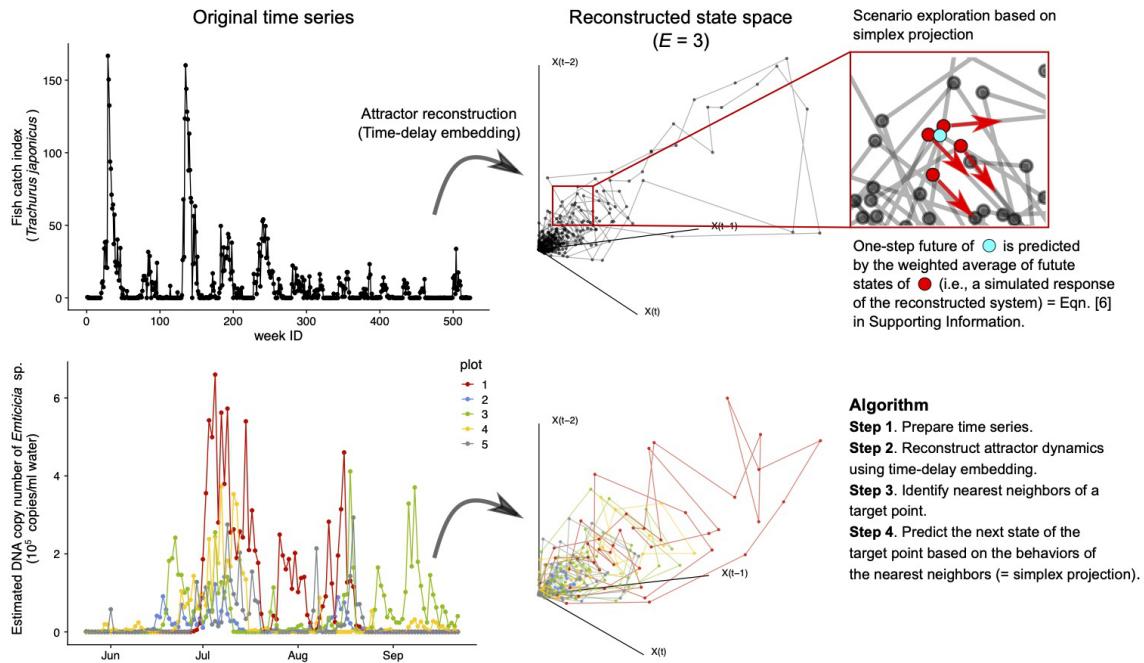


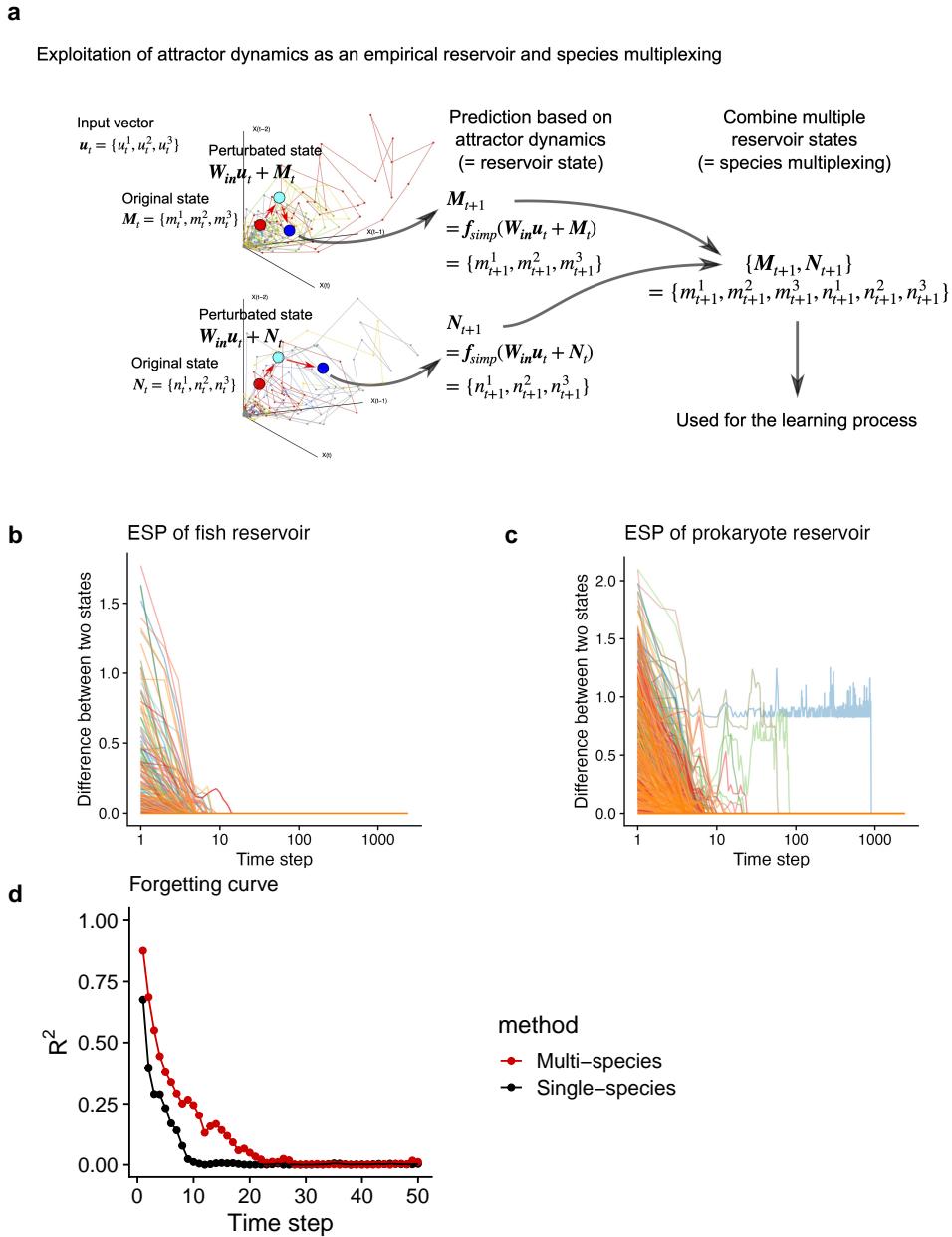
**a Conceptual illustrations of "Ecological Reservoir Computing (ERC)"**



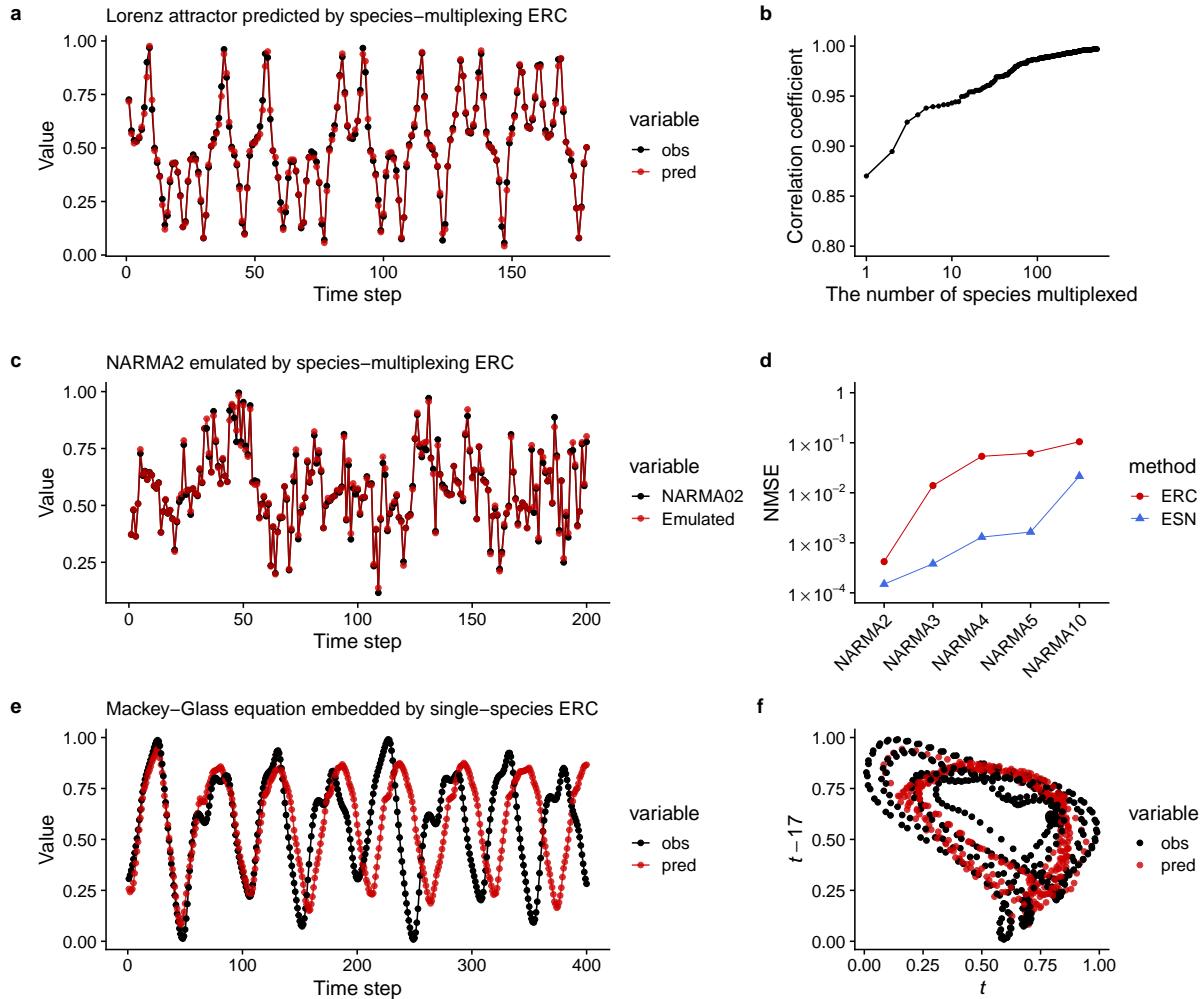
**b Examples of ecological time series, attractor reconstruction and scenario exploration for *in silico* ERC**



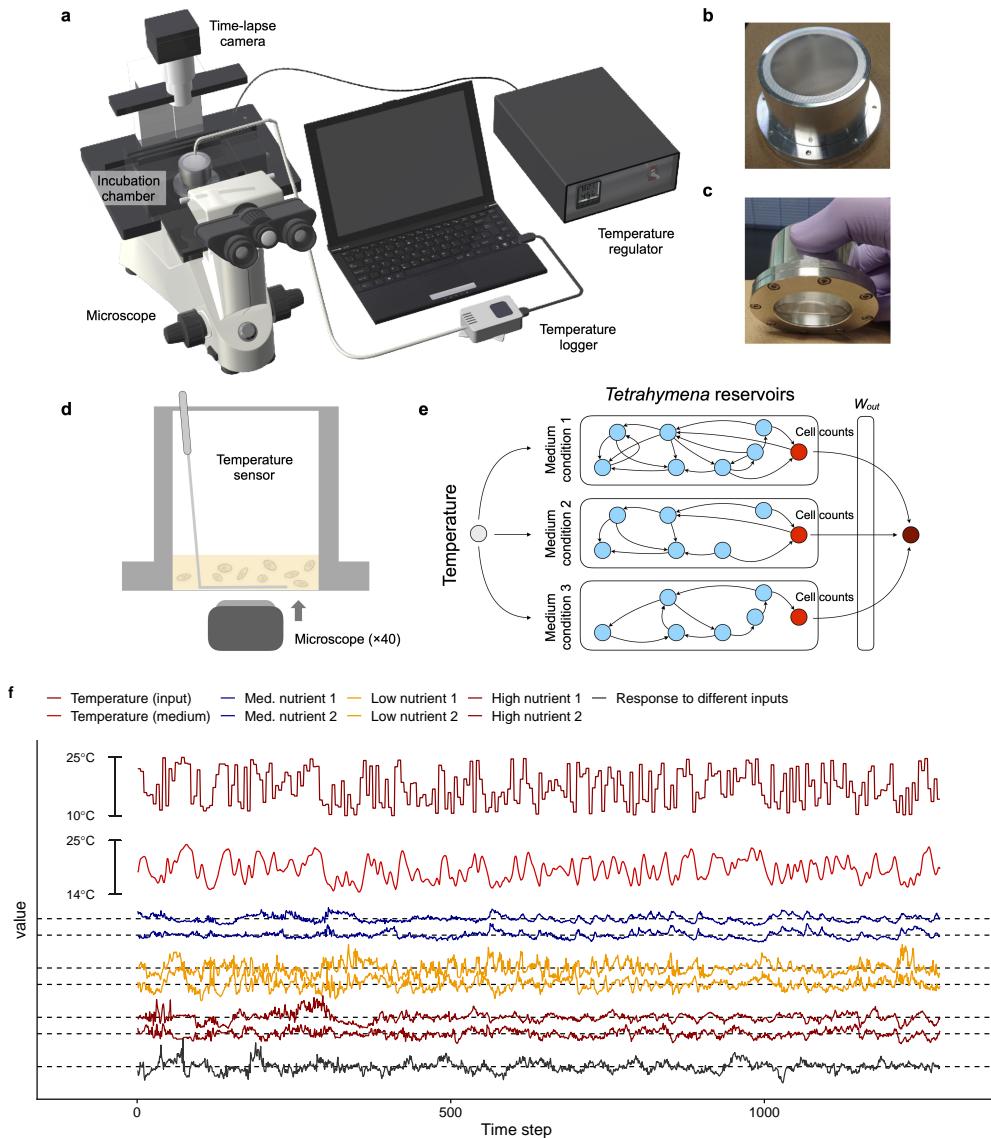
**Figure 1. Conceptual illustrations of ecological reservoir computing (ERC) and how *in silico* ERC is implemented.** (a) Conceptual illustrations of *in silico* ERC and real-time ERC. *In silico* ERC uses either equations or empirical dynamics reconstructed by empirical time series as a reservoir. Real-time ERC uses an empirical ecological interaction network as a reservoir. A node in an ecological reservoir may represent an individual, species, or abiotic variable in this study. (b) Examples of ecological time series, state space reconstruction, and scenario exploration for *in silico* ERC. Two empirical time series are shown as examples: Fish catch time series of Japanese jack mackerel (*Trachurus japonicus*) and DNA copy number time series of *Emiticicia* sp. in water samples collected from experimental rice plots (Ushio 2022). Empirical attractor dynamics can be reconstructed by time-delay embedding (Embedding dimension = 3). The red inlet indicates that the behavior of a target point (light blue) is predicted by the behaviors (red arrows) of nearest neighbors (red points).



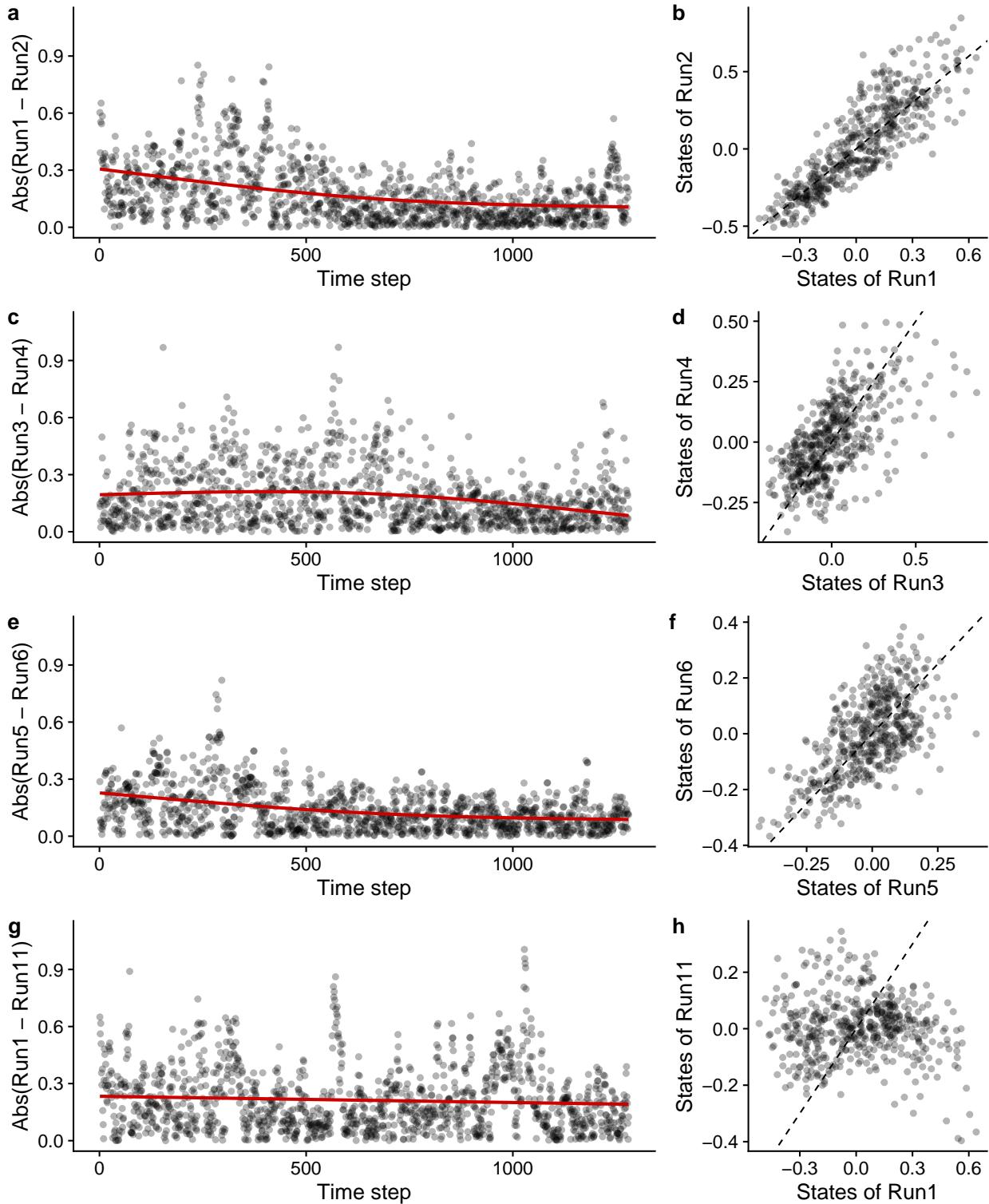
**Figure 2. Schematic illustrations of species-multiplexing and properties of *in silico* ERC.** (a) Schematic illustrations of species-multiplexing using scenario exploration. Simulated ecosystem responses are collected to generate a “multiplexed reservoir state”, which is used for the learning process. Red point in the reconstructed state space indicates the original vector, light blue point indicates perturbed vector, and blue point indicates the predicted response of the system by simplex projection (denoted by  $f_{simp}$ ). (b) Echo state property (ESP) of the reconstructed fish reservoirs (47 fish species included). For each run, the computation of *in silico* ERC started from two different initial conditions, and the dependence of the state difference on the time step was measured by the Euclidean distance between the two state. For each species, the numerical experiment repeated five times. (c) ESP of the reconstructed prokaryote reservoirs (500 prokaryote species included). *y*-axis in b and c indicates the difference between reservoir states started from different initial conditions and the difference converges to zero when the same input sequence is used. (d) An example of forgetting curves of reconstructed prokaryote reservoir. Forgotten curves of single-species prokaryotic reservoir (*Bdellovibrio* sp.) and species-multiplexed prokaryotic reservoir are shown.



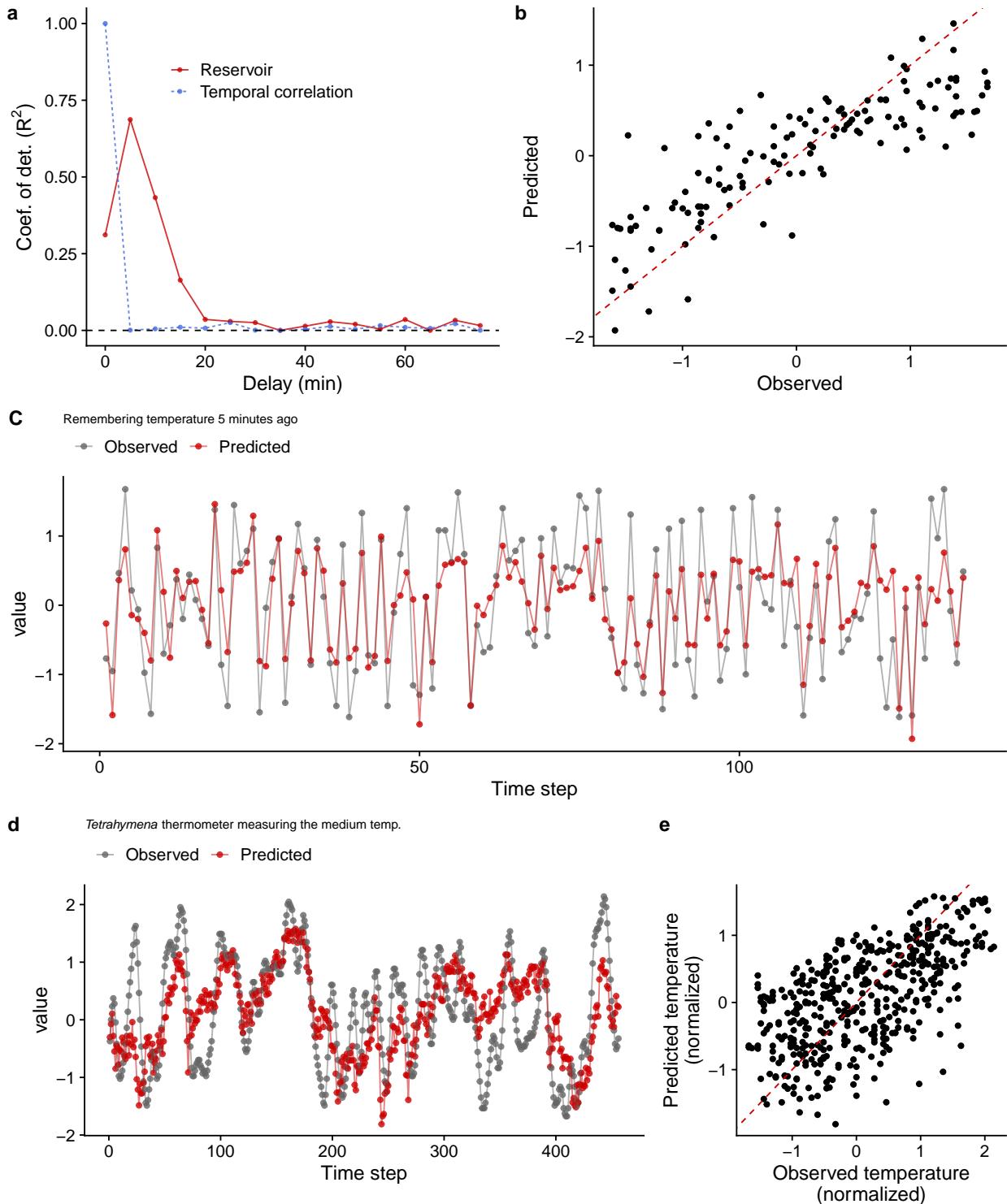
**Figure 3. The performance of species-multiplexed *in silico* ecological reservoir computing (ERC).** (a) Time series of Lorenz attractor (black points and lines) and one time-step future predictions by species-multiplexed *in silico* ERC (red points and lines). (b) Correlation coefficients of observed and predicted values of Lorenz system and the number of species multiplexed. (c) Nonlinear autoregression moving average (NARMA) time series (black points and lines) and emulation by species-multiplexed *in silico* ERC (red points and lines). NARMA is NARMA2 in this panel (d) Normalized mean square error (NMSE) of the NARMA emulations by species-multiplexed *in silico* ERC (red points and lines) and echo state network (ESN; blue points and lines). (e) The closed-loop embedding of the Mackey-Glass equations by species-multiplexed *in silico* ERC. The original attractor (black points and lines) was learned by the *in silico* ERC and autonomous dynamics was generated from time point zero (red points and lines). (f) Two-dimensional representation of the original Mackey-Glass attractor (black points) and that generated by the *in silico* ERC (red points).



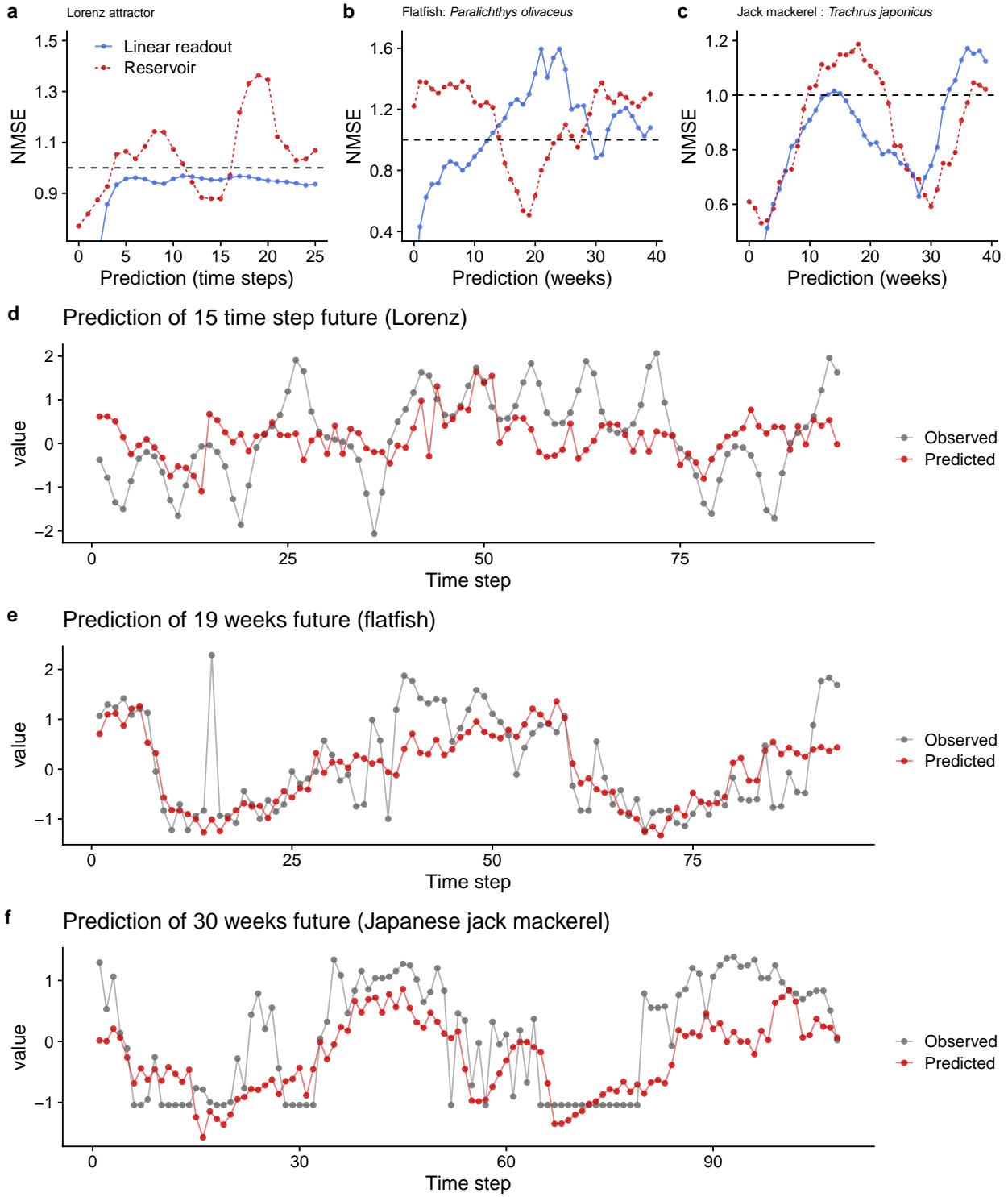
**Figure 4. The experimental set up of *Tetrahymena thermophila* reservoir.** (a) Illustrations of experimental set up. (b,c) Pre-incubated *Tetrahymena* population is maintained in an aluminum chamber. The total medium volume is five ml, and the concentration of nutrient is adjusted to change parameters of the population dynamics. (d) Cell images were taken from the bottom of the chamber. The number of cells was counted using a custom image analysis pipeline. (e) We used 1.6%, 4%, and 10% modified Neff medium in the experiments. Temperature information first transmits from the regulator to the aluminum chamber, and then propagates to several nodes in the medium and *Tetrahymena*. The temperature information is processed through complex interactions among temperature, medium, and behavior and physiology and *Tetrahymena*. The number of cells at the bottom of the chamber may contain the processed information and we use it as a reservoir state. Reservoir states of different nutrient concentrations were used to improve the performance of real-time ERC (i.e., space multiplexing) (f) Time series indicate input temperature (dense red), medium temperature (red), and a population density index (relative residual of the population density) in a 4% (blue line; Low nutrient), 1.6% (orange line; Med. nutrient), or 10% (brown line; High nutrient) modified Neff medium. Black line indicates population density index of *Tetrahymena* in a 4% modified Neff medium in response to a different temperature input sequence.



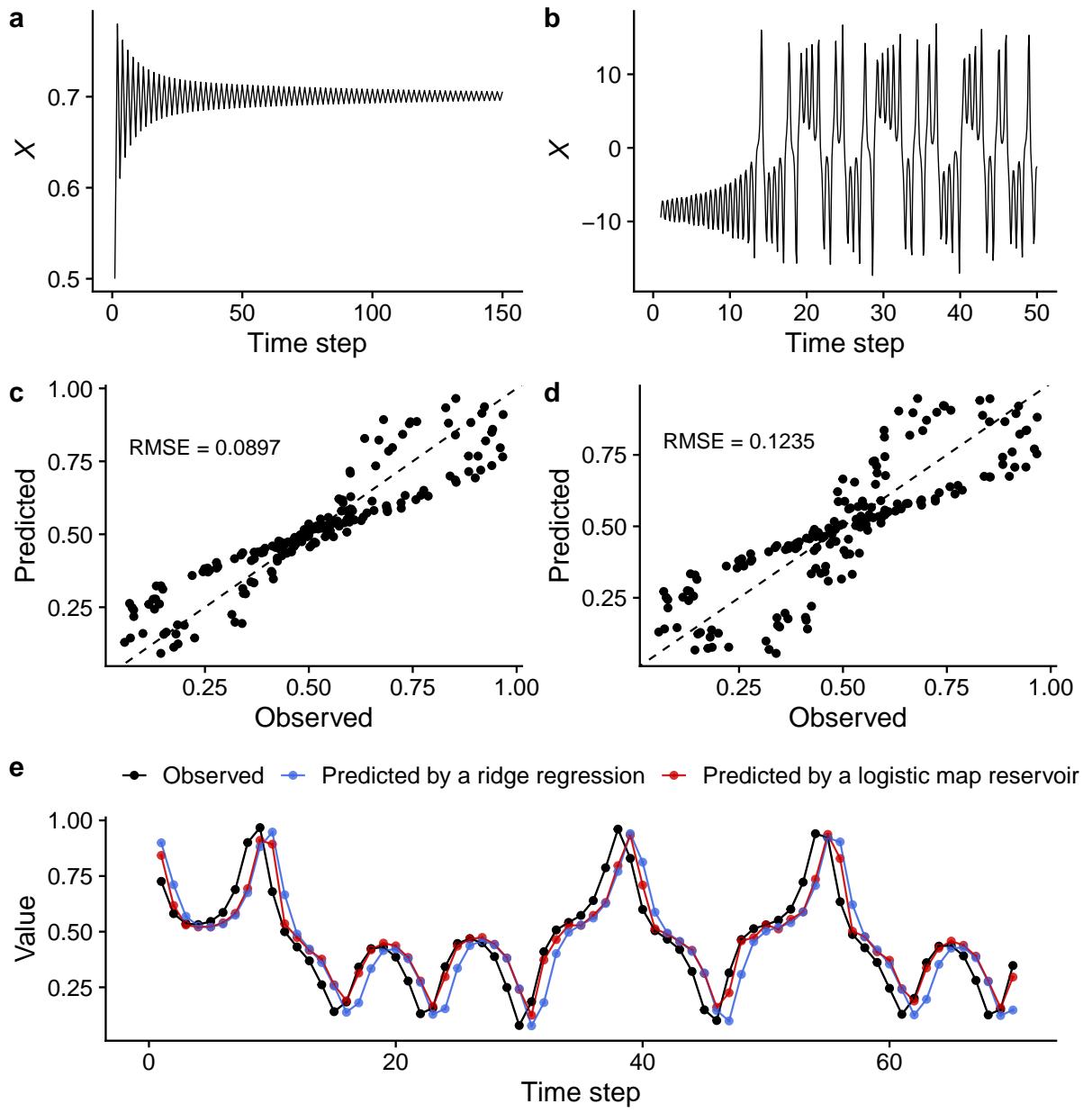
**Figure 5.** Echo state property (ESP) of real-time ecological reservoir computing (ERC). Two runs for each medium concentration were tested and the input sequence was identical for (a–f). (a, b) Comparison of two reservoir states (i.e., relative GAM residuals explained in Fig. S5; Run1 and Run2) where the same input sequence was added in 4% Neff medium. Time series plot (a) and scattered plot (b). (c, d) Comparison of two reservoir states in 1.6% Neff medium (Run3 and Run4) and (e, f) in 10% Neff medium (Run5 and Run6). (g, h) Comparison of two reservoir states in 4% Neff medium, but the input sequence was different for Run1 and Run11. Reservoir outputs converged for the identical inputs when the medium concentration was the same (a–f). On the other hand, the reservoir outputs did not converge when the input sequences were different (g, h). Red and dashed lines indicate GAM regression and 1:1 line, respectively.



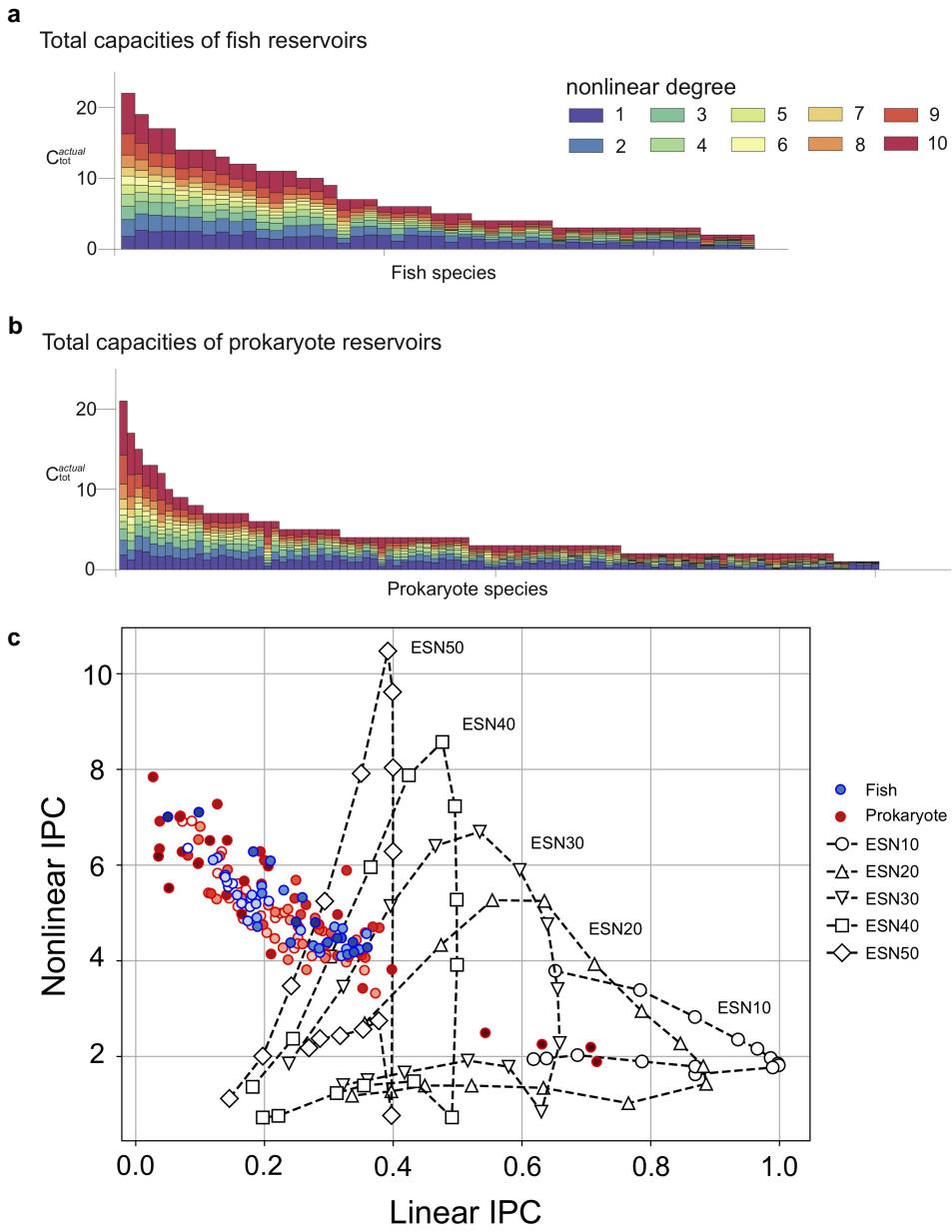
**Figure 6. Memory capacity of the *Tetrahymena* reservoirs.** (a) Memory capacity of the *Tetrahymena* reservoir measured using three time series of population density index (from three medium concentrations) as a training set and the other three time series as a test set (total six time series were used). Red points and lines indicate how well the *Tetrahymena* dynamics remembers the uniform random inputs. Blue points and lines indicate temporal correlations. (b) Correlations between observed and predicted values of uniform random inputs. Red dashed line indicates 1:1 line. (c) Observed (gray points and lines) and predicted (red points and lines) time series of uniform random inputs. In the panel c, temperatures five minutes ago were predicted by the states of the *Tetrahymena* reservoir. (d) Measurements of the medium temperatures five minutes ago by the *Tetrahymena* reservoir, suggesting that the *Tetrahymena* reservoir may work as a thermometer (*Tetrahymena* thermometer). (e) Correlations between observed and predicted temperature by the *Tetrahymena* thermometer.



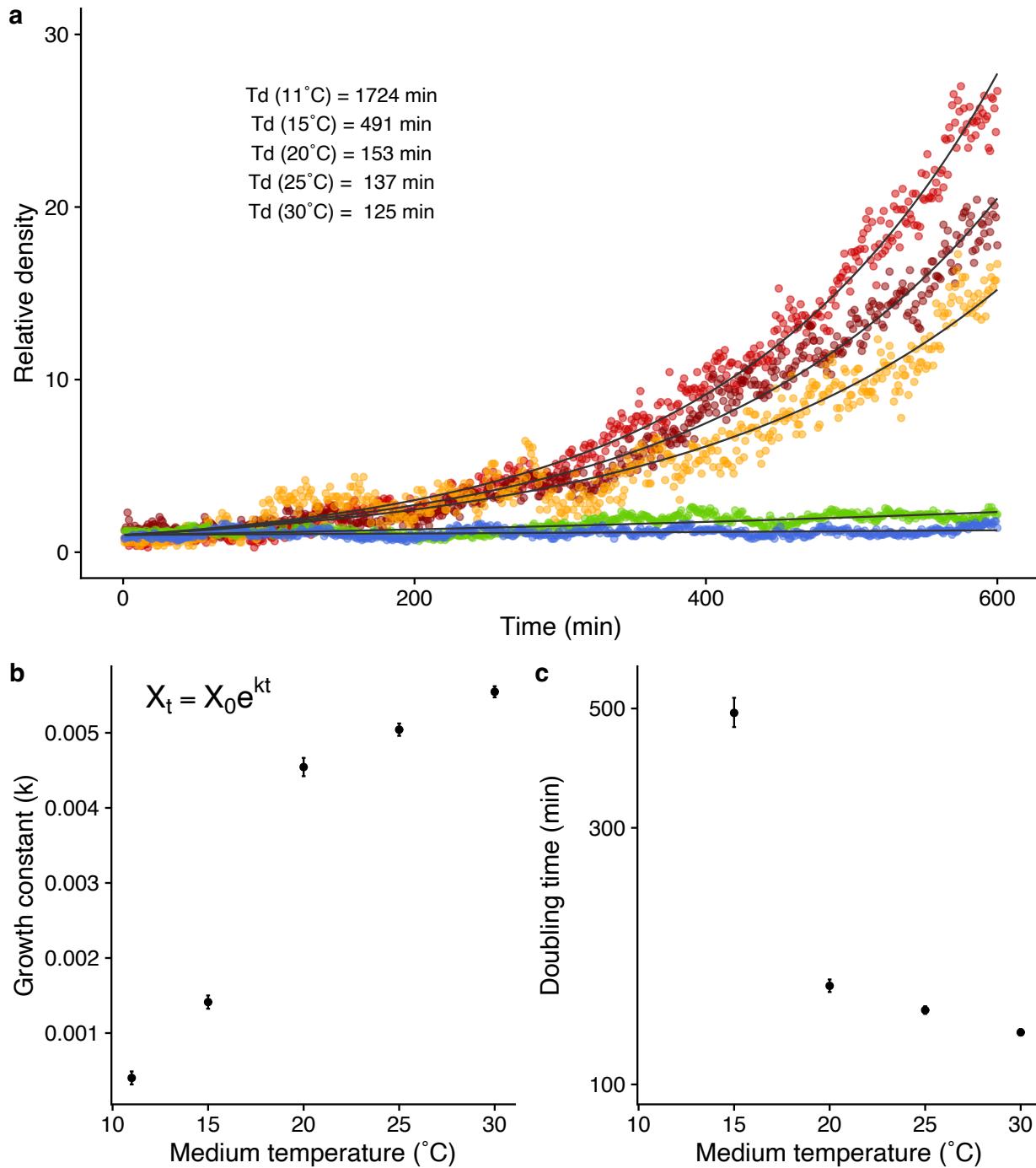
**Figure 7. The relationship between prediction accuracy (Normalized Mean Square Error; NMSE) and prediction time step. (f) Lorenz attractor, (g) flatfish time series, and (h) Japanese jack mackerel time series.** Red points and lines indicate predictions by the *Tetrahymena* reservoir, and blue points and lines indicate predictions by ridge regressions. Time series of observed (gray points and lines) and predicted (red points and values) of (i) Lorenz attractor with 15 time-step future prediction, (f) flatfish time series with 19 weeks future prediction, and (k) Japanese jack mackerel time series with 30 weeks future prediction.



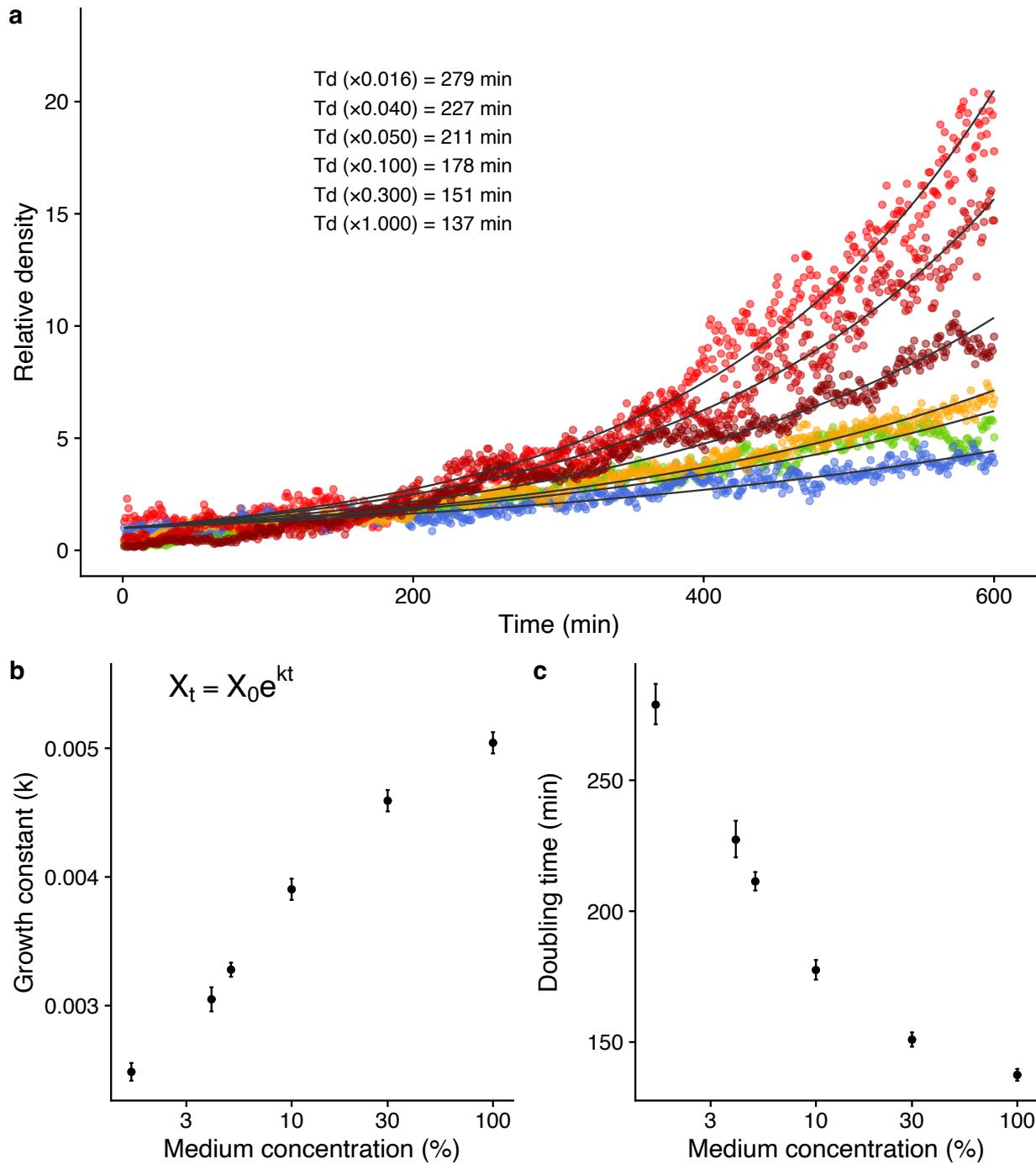
**Figure S1.** (a) Time series of a variable  $x$  of a logistic map. (b) Time series of a variable  $x$  in Lorenz system. The parameter values to generate the dynamics are described in the Supplementary Text (c-e) Results of reservoir computing using the logistic system as a reservoir. Input vectors are a variable  $x$  of the Lorenz system, and the one-step prediction accuracy was evaluated by Root Mean Square Error (RMSE). (c) The correlation between observed and predicted values by a logistic map reservoir. (d) As a comparison, the correlation between observed and predicted values by a ridge regression is shown (i.e., predictions without a reservoir). The prediction accuracy of the logistic reservoir is slightly better than that of the ridge regression. (e) Time series plot of the observed and predicted values by the logistic reservoir and ridge regression. The predictions made by a logistic reservoir delayed by the observed values, suggesting that a relatively low prediction capacity of the reservoir due to the small reservoir size ( $N = 2$ ).



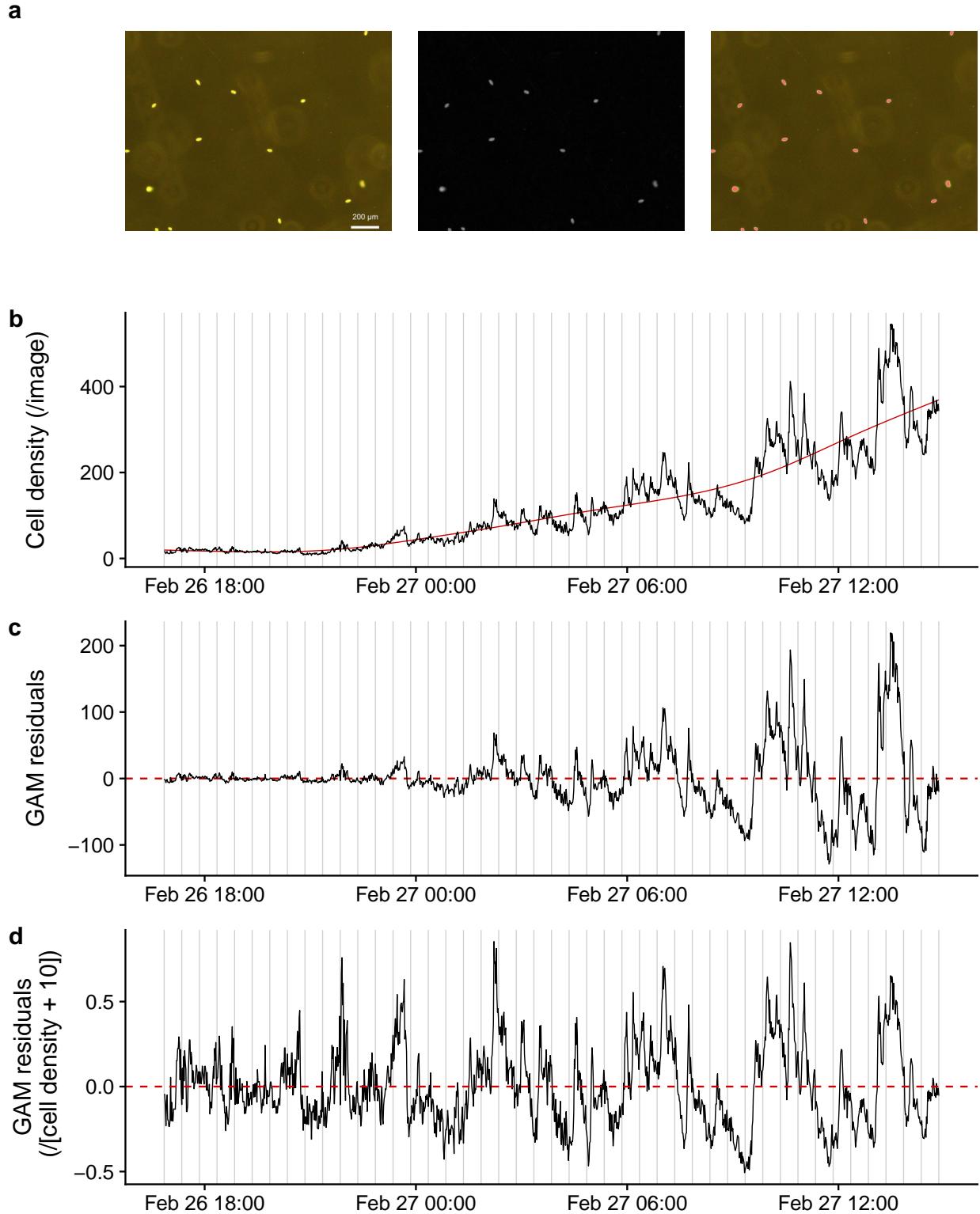
**Figure S2.** Measurements of information processing capacity of *in silico* ecological reservoirs. **a** and **b** show the actual total capacities and the degree of nonlinearity consisting of each capacity for each species in fish reservoirs (**a**; top 47 species are shown) and prokaryote reservoirs (**b**; top 100 species are shown). Horizontal axis shows the sorted number of species in descending order of their capacity size. Colors indicate the degree of nonlinearity. **c** shows the linearity–nonlinearity balance of the actual total IPC. Horizontal axis shows  $L(C_{tot}^{actual})$ , and vertical axis shows  $NL(C_{tot}^{actual})$ . Results from all the species were overlaid together with the results of the echo state networks (ESNs) for comparison. Different symbols indicate different reservoir settings (*in silico* ERC or ESN), and different colors of the filled circles indicate different species. For the setting of ESNs, we used a nonlinear activation function ( $f(x) = \tanh x$ ), the input weight was randomly assigned in the scale of  $[-0.01, 0.01]$ , and the spectral radius of the internal weight matrix of the ESN was altered from 0.1 to 1.5 for ESNs. The number of nodes for the ESNs are varied from 10 to 50 for comparison, and all the points and lines show the averaged value for 10 trials obtained by randomly choosing the weights of ESNs with respect to the given condition.



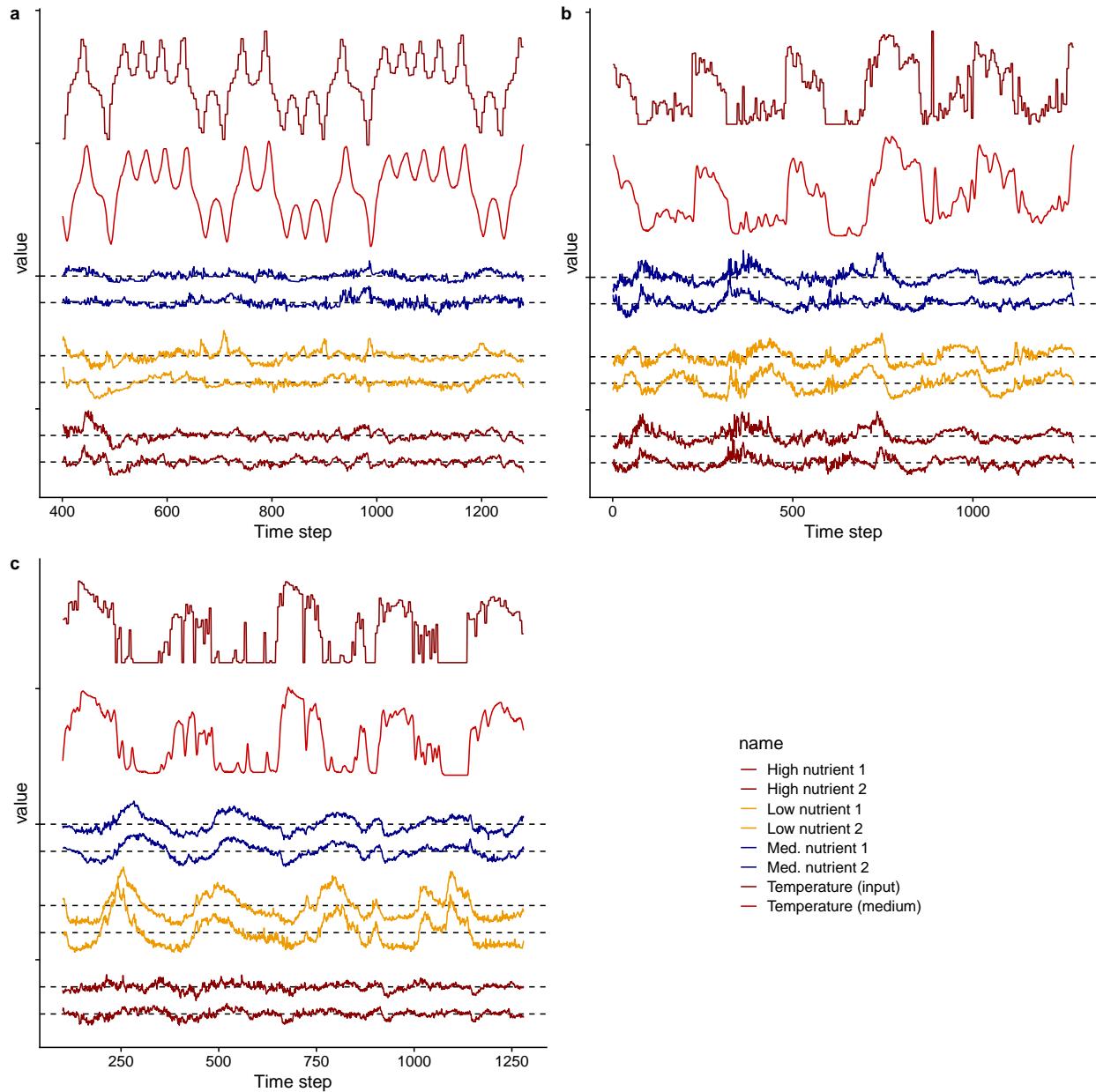
**Figure S3.** Dependence of *Tetrahymena* population growth rate on the medium temperature. (a) Changes in population density under different temperature conditions. The initial population density is scaled to one (“Relative density”) to facilitate the comparison among different temperatures. Detailed experimental conditions are described in the text. (b) Growth constant and medium temperature. Growth constants were estimated by fitting the data to the exponential growth model. (c) Doubling time ( $T_d$ ) and medium temperature ( $T_d = \ln(2)/k$ ).



**Figure S4.** Dependence of *Tetrahymena* population growth rate on the medium concentrations. (a) Changes in population density under different medium concentrations. The initial population density is scaled to one (“Relative density”) to facilitate the comparison among different temperatures. Detailed experimental conditions are described in the text. (b) Growth constant and medium concentration. Growth constants were estimated by fitting the data to the exponential growth model. (c) Doubling time ( $T_d$ ) and medium concentration ( $T_d = \ln(2)/k$ ).



**Figure S5.** Calculation of reservoir state of real-time ecological reservoir computing (ERC). **(a)** Processing of *Tetrahymena* cell images. Images ( $\times 40$ ) were obtained using an optical inverted microscope (the image covers ca. 1.8 mm width  $\times$  1.35 mm height). The background of the image was subtracted, converted to the gray-scale, and cells were counted by the watershed algorithm. **(b)** Change in the *Tetrahymena* cell density. Solid red line indicates a trend estimated by general additive model (GAM regression). **(c)** Residuals of GAM regression. Dashed red line indicates zero. **(d)** Relative GAM residuals. GAM residuals divided by [cell density + 10] to stabilize the fluctuations, which is used as reservoir state of real-time ERC.



**Figure S6.** Input sequences, medium temperature, and six reservoir states of ecological reservoir. **(a)** Input sequence that represents Lorenz system and the responses of *Tetrahymena* reservoir. **(b)** Input sequence that represents *Paralichthys olivaceus* time series and the reservoir responses. **(c)** Input sequence that represents *Trachurus japonicus* time series and the reservoir responses. The first top and second top time series indicate input temperature and medium temperature, respectively. The dark blue (third and fourth top), orange, and dark red lines indicate reservoir states of middle (4%), low (1.6%) and high (10%) medium conditions, respectively.