Supplementary Material for:
Critical size of neural population for reliable information transmission

Lubomir Kostal*

Institute of Physiology of the Czech Academy of Sciences, Videnska 1083, 14220 Prague 4, Czech Republic

Ryota Kobayashi†

Principles of Informatics Research Division, National Institute of Informatics, 2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo, Japan

Note: We refer to the figures and equations in the main text by their number, e.g., Eq. (1), while equations in this material are labeled as Eq. (S1), Eq. (S2), etc.

A. Information capacity

The cutting-plane algorithm [1] is applicable under general circumstances, e.g., continuously varying input and/or output, constrained optimization, and also allows to control the numerical precision of the result. The principle of the cutting-plane algorithm is the representation of a non-linear optimization problem as a sequence of converging linear programming problems. Here we follow the implementation by Huang and Meyn [2]. In the following we use indices: n as the coding dimension, r is number of support points for the numerical method (discretization), m is the number of stimulus patterns ($k = 1, \dots, m$) and j is the iteration of the numerical procedure (for both the capacity and lower bound optimization), in compatibility with the main text.

Let $\mathcal{F}$ be a set of all input stimulus distributions defined over finite stimulus range, $[x_{\min}, x_{\max}]$. The channel sensitivity function $g_p(x)$, given the input p.d.f. $p(x)$, is defined as the Kullback-Leibler distance

$$g_p(x) = \int f(y|x) \log \frac{f(y|x)}{f(y)} \, \mathrm{d}y, \tag{S1}$$

thus Eq. (1) can be expressed as

$$I(X;Y) = \int g_p(x) p(x) \, \mathrm{d}x. \tag{S2}$$

It follows that for some selected $X_0 \sim p_0(x) \in \mathcal{F}$ holds

$$I(X_0;Y) = \min_{p(x) \in \mathcal{F}} \int g_p(x) p_0(x) \, \mathrm{d}x, \tag{S3}$$

since

$$\begin{aligned} &\int g_p(x) p_0(x) \, \mathrm{d}x = \\ &= \int\int p_0(x) f(y|x) \log \left[\frac{f(y|x)}{f(y)} \frac{f_0(y)}{f_0(y)} \right] \mathrm{d}y \, \mathrm{d}x = \\ &= I(X_0;Y) + \int \log \frac{f_0(y)}{f(y)} \int p_0(x) f(y|x) \, \mathrm{d}x \, \mathrm{d}y. \end{aligned} \tag{S4}$$

* E-mail: kostal@biomed.cas.cz

† E-mail: r-koba@nii.ac.jp

where $f_0(y) = \int f(y|x)p_0(x)\,\mathrm{d}x$. The last term in Eq. (S4) is ≥ 0, since it can be written in the form of a Kullback-Leibler distance, which is known to be non-negative, and equal to zero only if $p(x) = p_0(x)$. The minimum is unique, and therefore it holds for $p(x)$: $\int g_p(x)p_0(x)\,\mathrm{d}x \geq I(X,Y)$. Eq. (S3) can be written by using the inner (scalar) product $\langle\cdot,\cdot\rangle$ as

$$I(X_0;Y) = \min_{p(x)\in\mathcal{F}} \langle g_p(x), p_0(x)\rangle. \tag{S5}$$

The general problem of finding capacity in Eq. (2) is solved as follows. The algorithm is initialized with an arbitrary distribution $p_0(x) \in \mathcal{F}$. In the j-th iteration of the algorithm we have a set of j input distributions, $\{p_0(x), p_1(x), \ldots, p_{j-1}(x)\} \subset \mathcal{F}$, and by employing Eq. (S5) we have

$$I_j(X;Y) = \min_{0\leq i\leq j-1} \langle p(x), g_i(x)\rangle, \tag{S6}$$

where $g_i(x)$ is the sensitivity function given $p_i(x)$, $g_i \equiv g_{p_i}$. The next input distribution, $p_j(x)$, is obtained as a solution to the problem

$$p_j(x) = \arg\max_{p(x)} \{I_j(X;Y) : p(x) \in \mathcal{F}\}. \tag{S7}$$

The optimization in Eq. (S7) can be expressed as a linear programming problem

$$\begin{aligned} \text{maximize} \quad & c \\ \text{subject to} \quad & \langle p(x), g_i(x)\rangle \geq c \quad \text{for } i = 0, \ldots j-1, \\ & p(x) \in \mathcal{F}. \end{aligned} \tag{S8}$$

To proceed we select r points of support of the input distribution, $\{x_1, x_2, \ldots, x_r\}$ within the allowed interval. We optimize with respect to the probabilities of these points, $\Pr(x_i)$, so that the allowed input distributions can be expressed as

$$p(x) = \sum_{i=1}^{r} \Pr(x_i)\delta(x - x_i), \tag{S9}$$

where $\delta(\cdot)$ is the Dirac delta function. We set $r = 300$ for the actual calculations, which is sufficient from the perspective of numerical precision in the analyzed neuronal models. In fact, as a consequence of Dubin's theorem [3], the optimal distributions are guaranteed to be discrete with finite points of support (even though the exact values of x_i are not known). Furthermore the number of the support points is at most the number of output states, which is limited by the maximum number of output action potentials in a time window. In addition, we alternatively also optimized r and the grid positions by an iterative algorithm, Starting with a 2-point grid and adding new point to the maximum of the channel sensitivity function after each iteration. In this way, it is possible to "build" an optimized input grid. As the number of cycles increases, some already placed grid points are no longer assigned non-zero probabilities and can be removed. In summary, however, the grid-optimization method did not yield any improvement in terms of convergence or numerical precision. Also note that while maximization of mutual information is concave in the input probabilities, it is generally neither convex nor concave in the positions of grid points.

We rewrite the problem in Eq. (S8) into standard form as

$$\begin{aligned} \text{maximize} \quad & \mathbf{d}^{\mathsf{T}}\mathbf{x} \\ \text{subject to} \quad & \mathbf{A}\mathbf{x} \leq \mathbf{b} \\ & \mathbf{x} \geq 0, \end{aligned} \tag{S10}$$

where $\mathbf{x}^\mathsf{T} = (x_1, x_2, \dots, x_r)$ and the construction of matrix $\mathbf{A}$ is described below. Let the probability of x_i in the j-th distribution, $p_j(x)$, be denoted as $p_{i,j}$, thus

$$p_j(x) = \sum_{i=1}^{r} p_{i,j}\delta(x - x_i). \tag{S11}$$

The probabilities of the points of support of the variable distribution $p(x)$ will be simply denoted as $p_1, \dots, p_r$. We form $(r+1)$-dimensional vectors $\mathbf{x}$ and $\mathbf{d}$ as

$$\mathbf{x}^\mathsf{T} = (c, p_1, p_2, \dots, p_r), \tag{S12}$$

$$\mathbf{d}^\mathsf{T} = (1, 0, 0, \dots, 0), \tag{S13}$$

so that $\mathbf{d}^\mathsf{T}\mathbf{x} = c$. The required condition $\mathbf{x} \geq 0$ is thus not in contradiction with Eq. (S12).

Next, we express the matrix $\mathbf{A}$. We write the normalization condition $\sum_{i=1}^{r} p_{i,j} = 1$ by employing the $1 \times (r+1)$ matrix $\mathbf{A}_1$ as

$$\mathbf{A}_1 = (0, 1, 1, \dots, 1), \tag{S14}$$

$$\mathbf{A}_1\mathbf{x} = 1, \tag{S15}$$

or equivalently by respecting the inequality condition in Eq. (S10) as

$$\mathbf{A}_1\mathbf{x} \leq 1, \tag{S16}$$

$$-\mathbf{A}_1\mathbf{x} \leq -1. \tag{S17}$$

Next, we re-formulate the j conditions in the j-th iteration of the algorithm, $\langle p(x), g_\ell(x)\rangle \geq c$ for $\ell = 0, 1, \dots, j-1$. Rewriting the condition as $c - \langle p(x), g_\ell(x)\rangle \leq 0$ yields

$$\mathbf{G}_\ell = (1, -g_\ell(x_1), -g_\ell(x_2), \dots, -g_\ell(x_r)), \tag{S18}$$

$$\mathbf{G}_\ell\mathbf{x} \leq 0, \tag{S19}$$

where

$$g_\ell(x_j) = \int f(y|x_j) \log \frac{f(y|x_j)}{\sum_{i=1}^{r} p_{i,\ell} f(y|x_i)}\, \mathrm{d}y. \tag{S20}$$

The complete algorithm at j-th iteration can be summarized as follows.

1. *Initialization ($j = 0$).* Select $\{x_1, \dots, x_k\}$ and $p_{i,0} = \Pr(x_i)$, for $i = 1 \dots r$. (The initial distribution may be chosen arbitrarily without significant impact on the convergence).

2. *Iteration ($j \geq 1$).* Form the $(3+j) \times r$ matrix $\mathbf{A}$ and vector $\mathbf{b}$,

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_1 \\ -\mathbf{A}_1 \\ \mathbf{G}_0 \\ \vdots \\ \mathbf{G}_{j-1} \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} 1 \\ -1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}. \tag{S21}$$

   Solve the linear programming problem in Eq. (S10). The resulting value of c in Eq. (S12), denoted as c_j, approaches the true capacity C and the probabilities $p_{i,j}$ approach the discrete capacity-achieving distribution. The solution, $p_{i,j}$, is employed as the j-th distribution for the next iteration. More precisely [2], as j increases, it holds

$$c_0 > c_1 > \dots > c_j \to C \qquad \text{upper bound,} \tag{S22}$$

$$\langle\, g_j, p_j \rangle \to C \qquad \text{lower bound,} \tag{S23}$$

$$p_{i,j} \to p_i^* \qquad p^*(x). \tag{S24}$$

The relative precision of the solution is defined as $(c - \langle p(x), g_p(x)\rangle)/c$.

B. Achievable information rate

The value of mutual information (and capacity) scales linearly with n for ergodic memoryless channels without feedback and i.i.d. inputs, $I(\{X_1,\dots,X_n\};\{Y_1,\dots,Y_n\}) = nI(X;Y)$, as follows from the properties of the functional in Eq. (1) and the product rule in Eq. (4). However, the actually achievable information rate of the population, $R_n = (\log m)/\Delta$, subject to the probability of decoding error P_e, behaves differently as maximal possible m is generally a complicated and unknown function of n. The linear scaling, $R_n \propto n$, is possible only asymptotically in n, as follows from Gallager's proof of Shannon's channel coding theorem [4, Thm. 5.6.2 and 7.3.2]. For example, one cannot decode the equiprobable input ($x^{(1)} = 0$ or $x^{(2)} = 1$) into a binary symmetric channel with crossover probability ε so that $P_e < \varepsilon$ from a single input-output pair. However, as the coding dimension n grows and, e.g., $\mathbf{x}^{(1)} = \{0,\dots,0\}$ and $\mathbf{x}^{(2)} = \{1,\dots,1\}$, the achievable error P_e decreases (see below). The remarkable result of Shannon's theory is that it is possible to add more input vectors $\mathbf{x}^{(k)}$, $k = 1,\dots,m$, and still decrease P_e. Asymptotically it is possible to use $m = \lceil e^{n\Delta C}\rceil$ input vectors and still maintain P_e arbitrarily small. The intuitive explanation of the theorem relies on the observation that for large n the channel noise affects each vector $\mathbf{x}^{(k)}$ in a somewhat restrictive way. Each element in the vector is perturbed independently and therefore consequent perturbations are unlikely to "conspire" to produce an unexpectedly significant disturbance [5, Ch. 1].

We present an informal derivation of the bound on the achievable information rate (Eq. 9) by employing the random coding technique and the error exponent. Details can be found in Gallager [4, Chapter 5]. For the purpose of simpler notation we use discrete setting, they key arguments are valid, with few caveats, in terms of probability density functions as well [4, Chapter 7]. The codewords (input stimulus vectors) are denoted as $\mathbf{x}_k = (x_{k,1}, x_{k,2}, \dots, x_{k,n})$, where $k = 1\dots m$. The rate R of the code is given by Eq. (3). The "coding" is a mapping from $\{1\dots m\}$ integers to codewords $\mathbf{x}_1\dots\mathbf{x}_m$. If some message with index k enters the encoder, $\mathbf{x}_k$ is transmitted, and based on the output sequence $\mathbf{y}$ the index k' is estimated. If $k' \neq k$, an error is declared. Non-asymptotically, we are interested in the probability of such error, and possible values of n and R, as discussed in the manuscript.

For a discrete memoryless channel the probability of output $\mathbf{y}$ given that the k-th codeword was sent is

$$\Pr(\mathbf{y}|\mathbf{x}_k) = \prod_{j=1}^{n} \Pr(y_j|x_{k,j}). \tag{S25}$$

If the decoder decodes $\mathbf{y}$ into message k, the probability (given $\mathbf{y}$) of incorrect decoding is $1-\Pr(k|\mathbf{y})$. The maximum-likelihood decodes $\mathbf{y}$ into k' such that (*cf.* Eq. 6)

$$\Pr(\mathbf{y}|\mathbf{x}_{k'}) \geq \Pr(\mathbf{y}|\mathbf{x}_k), \quad \forall k \neq k'. \tag{S26}$$

The probability of decoding error for message k is

$$P_{e,k} = \sum_{\mathbf{y}\in Y_k^c} \Pr(\mathbf{y}|\mathbf{x}_k), \tag{S27}$$

where Y_k^c describes the set of all such output sequences, that when decoded they do not yield the message k. The average error probability is

$$P_e = \sum_{k=1}^{m} \Pr(k) P_{e,k}. \tag{S28}$$

Assume first only two codewords $\mathbf{x}_1, \mathbf{x}_2$. The error probabilities for each codeword can be bounded as

$$P_{e,1} = \sum_{\mathbf{y}\in Y_1^c} \Pr(\mathbf{y}|\mathbf{x}_1)^{1-s} \Pr(\mathbf{y}|\mathbf{x}_1)^s \leq \sum_{\mathbf{y}} \Pr(\mathbf{y}|\mathbf{x}_1)^{1-s} \Pr(\mathbf{y}|\mathbf{x}_2)^s, \tag{S29}$$

$$P_{e,2} \leq \sum_{\mathbf{y}} \Pr(\mathbf{y}|\mathbf{x}_2)^{1-r} \Pr(\mathbf{y}|\mathbf{x}_1)^r, \tag{S30}$$

for any $0 < s < 1$ and $0 < r < 1$. By setting $r = 1 - s$ obtain the same bound, independent of $k = 1, 2$, in particular, for memoryless channel we have due to Eq. (S25)

$$\begin{aligned} P_{e,k} &\leq \sum_{\mathbf{y}} \prod_{j=1}^{n} \Pr(y_j|x_{1,j})^{1-s} \Pr(y_j|x_{2,j})^s = \\ &= \prod_{j=1}^{n} \sum_{y_j} \Pr(y_j|x_{1,j})^{1-s} \Pr(y_j|x_{2,j})^s = \prod_{j=1}^{n} g_j(s), \end{aligned} \tag{S31}$$

where $g_j(s) = \sum_{y_j} \Pr(y_j|x_{1,j})^{1-s} \Pr(y_j|x_{2,j})^s \leq 1$. The best error bound is then found by minimizing Eq. (S31) with respect to s. The importance of Eq. (S31) lies in the fact, that for two codewords the probability of decoding error goes to zero exponentially with increasing blocklength.

The random ensemble of codes (codebooks) is defined by the probability of some particular (n, m)-code $\mathbf{x}_1, \ldots, \mathbf{x}_m$ is (Eq. 8),

$$\Pr(\text{"}(n,m)\text{-code"}) = \prod_{k=1}^{m} p_n(\mathbf{x}_k). \tag{S32}$$

Each code in the ensemble has its own probability of decoding error and in the following an upper bound on the expectation (over the ensemble) of this error probability is derived. At least one code in the ensemble must satisfy the bound. As follows from Eqns. (S29) and (S30), the error probability for either codeword, $P_{e,k} = P_{e,1} = P_{e,2}$, is a function of $\mathbf{x}_1$ and $\mathbf{x}_2$. Thus, the average error probability over the 2-codeword ensemble from Eq. (S32) is given by

$$\bar{P}_{e,k} = \sum_{\mathbf{x}_1} \sum_{\mathbf{x}_2} p_n(\mathbf{x}_1) p_n(\mathbf{x}_2) P_{e,k}. \tag{S33}$$

Substituting the bound on $P_{e,k}$ from Eq. (S29) gives for any $0 < s < 1$ and $k = 1, 2$,

$$\begin{aligned} \bar{P}_{e,k} &\leq \sum_{\mathbf{x}_1,\mathbf{x}_2,\mathbf{y}} p_n(\mathbf{x}_1) p_n(\mathbf{x}_2) \Pr(\mathbf{y}|\mathbf{x}_1)^{1-s} \Pr(\mathbf{y}|\mathbf{x}_2)^s = \\ &= \sum_{\mathbf{y}} \Big[\sum_{\mathbf{x}_1} p_n(\mathbf{x}_1) \Pr(\mathbf{y}|\mathbf{x}_1)^{1-s} \Big] \Big[\sum_{\mathbf{x}_2} p_n(\mathbf{x}_2) \Pr(\mathbf{y}|\mathbf{x}_2)^s \Big]. \end{aligned} \tag{S34}$$

The minimum of the RHS of Eq. (S34) occurs at $s = 1/2$, and thus

$$\bar{P}_{e,k} \leq \sum_{\mathbf{y}} \Big[\sum_{\mathbf{x}} p_n(\mathbf{x}) \sqrt{\Pr(\mathbf{y}|\mathbf{x})} \Big]^2. \tag{S35}$$

For the memoryless channel holds Eq. (S25), and furthermore we choose $p_n(\mathbf{x}) = \prod_{j=1}^{n} p(x_j)$, so that Eq. (S35) can be manipulated into

$$\bar{P}_{e,k} \leq \left\{ \sum_{y\in\mathcal{Y}} \Big[\sum_{x\in\mathcal{X}} p(x) \sqrt{\Pr(y|x)} \Big]^2 \right\}^n. \tag{S36}$$

For $m > 2$ the average error of the k-th codeword can be expressed as

$$\bar{P}_{e,k} = \sum_{\mathbf{x}_k} \sum_{\mathbf{y}} p_n(\mathbf{x}_k) \Pr(\mathbf{y}|\mathbf{x}_k) \Pr(\text{error}|k, \mathbf{x}_k, \mathbf{y}), \tag{S37}$$

where $\Pr(\text{error}|k, \mathbf{x}_k, \mathbf{y})$ is the probability of decoding error, conditioned on the message k entering the encoder, codeword $\mathbf{x}_k$ being formed and sequence $\mathbf{y}$ being received. For each decoding error, $k \neq k'$ (for given $k, \mathbf{x}_k, \mathbf{y}$), define the event $A_{k'}$ that the codeword $\mathbf{x}_{k'}$ is decoded wrongly, $\Pr(\mathbf{y}|\mathbf{x}_{k'}) \geq \Pr(\mathbf{y}|\mathbf{x}_k)$. The probability can be bounded as

$$\Pr(\text{error}|k, \mathbf{x}_k, \mathbf{y}) \leq \Pr\left(\bigcup_{k' \neq k} A_{k'} \right) \leq \left[\sum_{k' \neq k} \Pr(A_{k'}) \right]^{\varrho}, \tag{S38}$$

where $0 < \varrho \leq 1$ and the first inequality follows from the fact that if for some k' holds $\Pr(\mathbf{y}|\mathbf{x}_{k'}) = \Pr(\mathbf{y}|\mathbf{x}_k)$, the decoder may not make an error. From the definition of $A_{k'}$ follows

$$\Pr(A_{k'}) = \sum_{\mathbf{x}_{k'}} p_n(\mathbf{x}_{k'}) \leq \sum_{\mathbf{x}_{k'}} p_n(\mathbf{x}_{k'}) \frac{\Pr(\mathbf{y}|\mathbf{x}_{k'})^s}{\Pr(\mathbf{y}|\mathbf{x}_k)^s}, \tag{S39}$$

for any $s > 0$, where the first summation is over $\mathbf{x}_{k'} : \Pr(\mathbf{y}|\mathbf{x}_{k'}) \geq \Pr(\mathbf{y}|\mathbf{x}_k)$, the second summation is over the dummy variable $\mathbf{x}_{k'}$. Substitution of Eq. (S39) into Eq. (S38) gives

$$\Pr(\text{error}|k, \mathbf{x}_k, \mathbf{y}) \leq \left[(m-1) \sum_{\mathbf{x}} p_n(\mathbf{x}) \frac{\Pr(\mathbf{y}|\mathbf{x}_{k'})^s}{\Pr(\mathbf{y}|\mathbf{x}_k)^s} \right]^{\varrho}. \tag{S40}$$

Substituting Eq. (S40) into Eq. (S37), and setting $s = 1/(1+\varrho)$, finally gives

$$\bar{P}_{e,k} \leq (m-1)^{\varrho} \sum_{\mathbf{y}} \left[\sum_{\mathbf{x}} p_n(\mathbf{x}) \Pr(\mathbf{y}|\mathbf{x})^{1/(1+\varrho)} \right]^{1+\varrho}. \tag{S41}$$

The (n, R) code is defined as a code of length n with $\lceil e^{nR} \rceil$ codewords. Considering the ensemble of codes to be the ensemble of (n, R) block codes with $(m-1) \leq e^{nR} \leq m$, we rewrite Eq. (S41) for the discrete memoryless channel as

$$\bar{P}_{e,k} \leq e^{nR\varrho} \left\{ \sum_{y \in \mathcal{Y}} \left[\sum_{x \in \mathcal{X}} p(x) \Pr(y|x)^{1/(1+\varrho)} \right]^{1+\varrho} \right\}^{n}, \tag{S42}$$

which can be furthermore rearranged into

$$\bar{P}_{e,k} \leq \exp\{-n[E_0(\varrho, p) - \varrho R]\}, \tag{S43}$$

$$E_0(\varrho, p) = -\log \sum_{y \in \mathcal{Y}} \left[\sum_{x \in \mathcal{X}} p(x) \Pr(y|x)^{1/(1+\varrho)} \right]^{1+\varrho}. \tag{S44}$$

Since Eq. (S43) is valid for all messages in the codebook, the average error probability over the messages $P_e = \sum_{k=1}^{m} \Pr(k) \bar{P}_{e,k}$

$$P_e \leq \exp\{-n[E_0(\varrho, p) - \varrho R]\}, \tag{S45}$$

Furthermore, tighter bound is obtained by choosing ϱ and $p(x)$ to maximize $[E_0(\varrho, p) - \varrho R]$. By inverting Eq. (S45) and by generally using probability density functions $p(x), f(y|x)$ instead of discrete probabilities $p(x), \Pr(y|x)$ we obtain Eqs. (9–11) in the manuscript.

In order to evaluate Eq. (10) we follow the cutting-plane method as described in [6]. Let $\mathcal{F}$ be the set of all possible input distributions $p(x)$. We define the functional $G^\varrho(p)$, convex $\cup$ in $p(x)$, in accordance with Gallager [4, p.144] and Huang *et al.* [6] as

$$G^\varrho(p) = \exp[-E_0(\varrho, p)], \tag{S46}$$

so that $E_0(\varrho, p) = -\log G^\varrho(p)$. The maximization problem yielding $E_r(R)$ in Eq. (10) is then resolved as follows. For each ϱ we define

$$G^{\varrho,*} = \min_{p(x)} G^\varrho(p), \tag{S47}$$

where the minimization is over all admissible $p(x)$. To each $G^{\varrho,*}$ and ϱ there corresponds a line in the "(E_r, R)-plane", given by

$$L^\varrho(Y) = -\varrho R - \log G^{\varrho,*}. \tag{S48}$$

Note that the information rate, R, is hereafter given in the units of nats for notational convenience, i.e., $R/\log 2$ is in bits. From Eq. (10) follows that $E_r(R)$ is the maximum over all these lines at given R [4, p. 144].

The error exponent sensitivity function $g_p^\varrho(x)$ is defined so that

$$G^\varrho(p) = \int g_p^\varrho(x) p(x)\,\mathrm{d}x, \tag{S49}$$

and thus by comparison with Eq. (10) we obtain

$$g_p^\varrho(x) = \int \left[\int f(y|u)^{1/(1+\varrho)} p(u)\,\mathrm{d}u\right]^\varrho f(y|x)^{1/(1+\varrho)}\,\mathrm{d}y. \tag{S50}$$

The next step is to express $G^\varrho(p_0)$ at some arbitrary $p_0(x) \in \mathcal{F}$ in terms of linear functionals (in p_0), similarly to the case of capacity-cost function. From the convexity of the functional $G^\varrho(p)$ follows that for some admissible $p_0(x)$ and $p(x)$ holds

$$G^\varrho(p_0) \geq G^\varrho(p) + \langle \partial_p G^\varrho(p), p_0 - p\rangle, \tag{S51}$$

where $\langle f, g\rangle = \int f(x)g(x)\,\mathrm{d}x$. In other words,

$$\begin{aligned} G^\varrho(p_0) \geq \int g_p^\varrho(x)p(x)\,\mathrm{d}x + \\ + (1+\varrho)\int g_p^\varrho(x)[p_0(x) - p(x)]\,\mathrm{d}x = \\ =(1+\varrho)\left\langle g_p^\varrho, p_0\right\rangle - \varrho\left\langle g_p^\varrho, p\right\rangle, \end{aligned} \tag{S52}$$

with equality if and only if $p(x) = p_0(x)$. Hence, $G^\varrho(p_0)$ can be expressed as the maximization problem,

$$G^\varrho(p_0) = \max_{p(x)}[(1+\varrho)\left\langle g_p^\varrho, p_0\right\rangle - \varrho G^\varrho(p)]. \tag{S53}$$

The main idea here is that the maximization above is linear in p_0.

The evaluation of $G^\varrho(p)$ is then performed iteratively. We start with an arbitrary distribution $p_0(x) \in \mathcal{F}$. In the j-th iteration we have a set of j admissible input distributions, $\{p_0(x), p_1(x), \ldots, p_{j-1}(x)\} \subset \mathcal{F}$, and from Eq. (S53) follows

$$G_j^\varrho(p) = \max_{0\leq i\leq j-1}[(1+\varrho)\langle g_i^\varrho, p\rangle - \varrho\langle g_i^\varrho, p_i\rangle]. \tag{S54}$$

where $g_i^\varrho(x) \equiv g_{p_i}^\varrho(x)$. The next input distribution, $p_j(x)$, is obtained from

$$p_j(x) = \arg\min_{p(x)} \{G_j^\varrho(p) : p(x) \in \mathcal{F}\}. \tag{S55}$$

Consequently, we arrive to the following sequence of mini-max problems

$$\begin{aligned} \text{minimize} \quad & \varepsilon \\ \text{subject to} \quad & (1+\varrho)\langle g_i^\varrho, p\rangle - \varrho\langle g_i^\varrho, p_i\rangle \le \varepsilon, \\ & \text{for } i = 0, \dots j-1, \quad p(x) \in \mathcal{F}, \end{aligned} \tag{S56}$$

which is converted into the sequence of linear programming problems of the form,

$$\begin{aligned} \text{maximize} \quad & \mathbf{d}^\mathsf{T}\mathbf{z} \\ \text{subject to} \quad & \mathbf{A}\mathbf{z} \le \mathbf{b} \\ & \mathbf{z} \ge 0. \end{aligned} \tag{S57}$$

Again, we fix the grid of r points of support $(x_1, x_2, \dots, x_r)$, let

$$p_j(x) = \sum_{i=1}^{r} p_j(x_i)\delta(x - x_i), \tag{S58}$$

where $p_j(x_i)$ are the components of the input distribution in the j-th iteration and we form

$$\mathbf{z}^\mathsf{T} = (\varepsilon, p(x_1), p(x_2), \dots, p(x_r)), \tag{S59}$$

$$\mathbf{d}^\mathsf{T} = (-1, 0, 0, \dots, 0). \tag{S60}$$

From $(1+\varrho)\langle g_i^\varrho, p\rangle - \varepsilon \le \varrho\langle g_i^\varrho, p_i\rangle$ we have in the j-th iteration,

$$\mathbf{\Gamma}_j = (-1, \gamma_j^\varrho(x_1), \gamma_j^\varrho(x_q), \dots, \gamma_j^\varrho(x_k)), \tag{S61}$$

$$\mathbf{\Gamma}_j\mathbf{z} \le \beta_j, \tag{S62}$$

where

$$\gamma_j^\varrho(x_i) = (1+\varrho)g_j^\varrho(x_i), \tag{S63}$$

$$\beta_j = \varrho \sum_{\ell=1}^{r} g_j^\varrho(x_\ell)p_j(x_\ell), \tag{S64}$$

$$g_j^\varrho(x_i) = \int \left[\sum_{\ell=1}^{r} f(y|x_\ell)^{1/(1+\varrho)} p_j(x_\ell)\right]^\varrho f(y|x_i)^{1/(1+\varrho)}\,\mathrm{d}y. \tag{S65}$$

The algorithm is evaluated for each $\varrho \in (0,1)$ as follows.

1. *Initialization ($j = 0$).* Set ϱ, $\{x_1, \dots, x_r\}$ and some $p_0(x_i)$ for $i = 1 \dots r$. Evaluate $\mathbf{\Gamma}_0$ and β_0.

2. *Iteration ($j \ge 1$).* Form the matrix $\mathbf{A}$ and vector $\mathbf{b}$,

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_1 \\ -\mathbf{A}_1 \\ \mathbf{\Gamma}_0 \\ \vdots \\ \mathbf{\Gamma}_{j-1} \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} 1 \\ -1 \\ \beta_0 \\ \vdots \\ \beta_{j-1} \end{pmatrix}, \tag{S66}$$

where $\mathbf{A}_1 = (0, 1, 1, \ldots, 1)$, $\mathbf{\Gamma}_\ell$ and β_ℓ are given by Eqns. (S61) and (S64). Solve the linear programming problem in Eq. (S57), denote the resulting value of ε as ε_j, use the values $\{p(x_1), \ldots, p(x_k)\}$ as the j-th distribution for the next iteration. The following holds as $j \to \infty$ [6, Theorem 3.1]

$$\left\langle g_j^{\varrho}, p_j \right\rangle \to G^{\varrho}(p^*), \tag{S67}$$

$$\varepsilon_1 \leq \varepsilon_2 \leq \varepsilon_3 \cdots \to G^{\varrho}(p^*), \tag{S68}$$

$$p_j(x_i) \to p^*(x_i), \tag{S69}$$

so that ε_j provides a lower bound on the true $G^{\varrho}(p^*)$ and $p_j(x_i)$ converges to the optimizer $p^*(x_i)$.

C. Gaussian approximation

The Eqs. (9) and (14) can rarely be solved in a closed-form. Even simple and frequently employed models of $f(y|x)$, such as Poisson distribution with intensity x, or Gaussian with x-dependent variance, require the numerical approach. We provide a potentially useful closed-form approximation to Eqs. (9) and (12), by employing several results valid for the discrete-time additive white Gaussian noise (AWGN) channel subject to variance constraint on the input and average variance constraint on the (non-asymptotic) codeword ensemble [4, Ch. 7] and [5, 7].

The AWGN channel is described by the conditional probability density function $f(y|x)$ such that $f(y|x) = f_N(y - x)$, where $f_N(z)$ is Gaussian with zero mean and variance σ^2. Furthermore, the support of the input distribution $p(x)$ is the whole real line, but X is constrained in variance, $\mathrm{Var}(X) \leq P$. The AWGN information capacity C_G is then known to be

$$C_G = \frac{1}{\Delta} \log\left(1 + \frac{P}{\sigma^2}\right), \tag{S70}$$

where P/σ^2 is the signal-to-noise ratio (SNR). We use Eq. (S70) to define the effective SNR, S, of an arbitrary information channel with capacity C given by Eq. (2) as

$$S = e^{2\Delta C} - 1. \tag{S71}$$

Efficient algorithms for the calculation of capacity exist, e.g., the cutting-plane approach employed here or the classical alternatives [8, 9].

The maximization over $p(x)$ in Eq. (10) is not carried out explicitly for AWGN, instead it is assumed that the ensemble-generating distribution is Gaussian with zero mean and variance P, which permits the indicated integration and maximization over ρ [4, pp. 338–343] but also can be shown to be essentially optimal [7]. Since we only need the whole ensemble to satisfy the variance constraint, but not the individual input vectors, the term $[2e^{r\delta}/\mu]^2$ in [4, Thm. 7.3.2, Eq. 7.3.45] can be omitted [5, Ch. 4] and the results may be summarized as follows.

For all rates $\tilde{R}_c < R \leq C$ the Eq. (10) gives

$$E_r(\Delta R) = \frac{S}{4\beta}\left\{\beta + 1 - (\beta - 1)\sqrt{1 + \frac{4\beta}{(\beta-1)S}} + \frac{1}{2}\log\left[\beta - \frac{(\beta-1)S}{2}\sqrt{1 + \frac{4\beta}{(\beta-1)S}} - 1\right]\right\}, \tag{S72}$$

where $\beta = e^{2\Delta R}$ and $\tilde{R}_c$ is the AWGN critical rate (Eq. 15)

$$\tilde{R}_c = \frac{1}{2\Delta}\log\left(\frac{1}{2} + \frac{S}{4} + \frac{1}{2}\sqrt{1 + \frac{S^2}{4}}\right). \tag{S73}$$

Below the critical rate, for $R_e < R \le \tilde{R}_c$ it holds

$$E_r(\Delta R) = 1 - \beta + \frac{S}{2} + \frac{1}{2}\log\left(\beta - \frac{S}{2}\right) + \frac{\log\beta}{2} - \Delta R, \tag{S74}$$

$$\beta = \frac{1}{2}\left[1 + \frac{S}{2} + \frac{S^2}{4}\right], \tag{S75}$$

$$R_e = \frac{1}{2\Delta}\log\left[\frac{1}{2} + \frac{1}{2}\sqrt{1 + \frac{S^2}{4}}\right], \tag{S76}$$

from which the critical population size $\tilde{n}_c = \lceil -(\log P_e)/E_r(\Delta\tilde{R}_c)\rceil$ follows as

$$\tilde{n}_c = \left\lceil -4\left[2 + S - \sqrt{4+S^2} - 4\log 2 + 2\log\left(2 - S + \sqrt{4+S^2}\right)\right]^{-1}\log P_e\right\rceil. \tag{S77}$$

Finally, for $0 < R \le R_e$,

$$E_r(\Delta R) = \frac{S}{4}\left(1 - \sqrt{1 - e^{2\Delta R}}\right). \tag{S78}$$

The Gaussian lower (achievable) bound in Fig. 2 is obtained by combining Eqs. (S72), (S76) and (S78) and substituting into Eq. (9).

The case of AWGN channel with variance constraint and P_e defined by Eq. (7) is exceptional in the sense that the normal approximation [10] given by Eq. (12) cannot be directly employed [5, pp. 136, Thm. 77]. Instead, the upper bound in Fig. 2 is approximated by the sphere-packing bound as derived by Shannon [7], which is given by Eq. (S72) for $0 < R \le C$ and $R_n \le nE_r^{-1}(-\log P_e/n)/\Delta$, i.e., the upper and lower bounds coincide above $\tilde{R}_c$.

The Gaussian approximation above would be exact for a given neuronal model $f(y|x)$ and stimulus distribution $p(x)$, if there existed bijective mappings $\tilde{x} = \varphi(x)$ and $\tilde{y} = \xi(y)$ such that $\tilde{X} \sim N(0, P)$ and $\tilde{Y}|\tilde{x} \sim N(\tilde{x}, \sigma^2)$, as follows from the transformation invariance of $I(X;Y)$. (E.g., for a "neuronal" model with exponential tuning curve and lognormal distribution of firing rates). Nonetheless, the Gaussian approximation is useful and works rather well for a range of actual neuronal models (Figs. 2, and S1–S3), where $f(y|x)$ is sufficiently regular and varies continuously with x.

D. Neuronal models

We simulated two kinds of model neurons, the conductance based model and the spike response model. Both neurons are driven by synaptic input, which consists of excitatory and inhibitory inputs.

1. *Conductance based model*

The neuron is modeled by the single-compartment conductance based model [11, 12] driven by synaptic input I_{syn},

$$C_m \frac{\mathrm{d}V}{\mathrm{d}t} = -g_L(V - E_L) - I_{\mathrm{Na}} - I_{\mathrm{Kd}} - I_{\mathrm{syn}}, \tag{S79}$$

where $C_m = 1\,\mu\mathrm{F/cm}^2$ is the membrane capacitance, V is the membrane voltage, $g_L = 0.1\,\mathrm{mS/cm}^2$ is the leak conductance and $E_L = -67\,\mathrm{mV}$ is the leak reversal potential of the neuron.

The sodium current I_{Na} is given by

$$
\begin{aligned}
I_{\text{Na}} &= g_{\text{Na}} m^3 h (V - E_{\text{Na}}), \\
\frac{\mathrm{d}m}{\mathrm{d}t} &= \alpha_m(V)(1-m) - \beta_m(V)m, \\
\alpha_m(V) &= \frac{-0.32(V+54)}{\exp[-(V+54)/4] - 1}, \\
\beta_m(V) &= \frac{0.28(V+27)}{\exp[(V+27)/5] - 1}, \\
\frac{\mathrm{d}h}{\mathrm{d}t} &= \alpha_h(V)(1-h) - \beta_h(V)h, \\
\alpha_h(V) &= 0.128 e^{-(V+50)/18}, \\
\beta_h(V) &= \frac{4}{1+\exp[-(V+27)/5]},
\end{aligned}
\tag{S80}
$$

where $g_{\text{Na}} = 100\,\text{mS/cm}^2$, $E_{\text{Na}} = 50\,\text{mV}$.

The delayed rectifier potassium current I_{Kd} is given by

$$
\begin{aligned}
I_{\text{Kd}} &= g_{\text{Kd}} n^4 (V - E_{\text{K}}), \\
\frac{\mathrm{d}n}{\mathrm{d}t} &= \alpha_n(V)(1-n) - \beta_n(V)n, \\
\alpha_n(V) &= \frac{-0.032(V+52)}{\exp[-(V+52)/5] - 1}, \\
\beta_n(V) &= 0.5 e^{-(V+57)/40},
\end{aligned}
\tag{S81}
$$

where $g_{\text{Kd}} = 80\,\text{mS/cm}^2$ and $E_K = -100\,\text{mV}$. Eqs. (S79−S81) was solved numerically using by using Euler-Maruyama integration scheme with a time step of 0.01 ms.

2. *Spike Response Model (SRM)*

The neuron is modeled by the spike response model (SRM) [13–17] driven by synaptic input I_{syn},

$$
V(t) = V_{\text{res}} - \int_0^\infty \kappa(s) I_{\text{syn}}(t-s) ds + \sum_{t_i < t} \eta(t - t_i). \tag{S82}
$$

where $V(t)$ is the membrane voltage, $V_{\text{res}} = -70\,\text{mV}$ is the resting potential, t_i is the i-th spike time, and the kernels $\kappa(s)$ and $\eta(s)$ describe the integrative and refractory property of the neuron. The model neuron generates a spike, if the voltage $V(t)$ reaches the threshold $\theta_V = -59\,\text{mV}$. We used exponential kernels [15], $\kappa(s) = \kappa_0 e^{-s/\tau_m}$ and $\eta(s) = \eta_0 e^{-s/\tau_m}$, where $\tau_m = 10\,\text{ms}$ is the membrane time constant. This model (S82) can be written in a form of a differential equation [18]:

$$
C_m \frac{dV(t)}{dt} = -g_{\text{tot}}(V(t) - V_{\text{res}}) - I_{\text{syn}}(t) + q_\eta \sum_{t_i < t} \delta(t - t_i) \tag{S83}
$$

where $C_m (= \kappa_0^{-1})$ is the capacitance, $g_{\text{tot}} (= C_m/\tau_m)$ is the total conductance, and $\delta(t)$ is the Dirac's δ function. Eq. (S83) was solved numerically using by using Euler-Maruyama integration scheme with a time step of 0.01 ms. The parameters are $C_m = 1\,\mu\text{F/cm}^2$, $g_{\text{tot}} = 0.1\,\text{mS/cm}^2$, and $q_\eta (= \eta_0/\kappa_0) = -10\,\text{nC/cm}^2$.

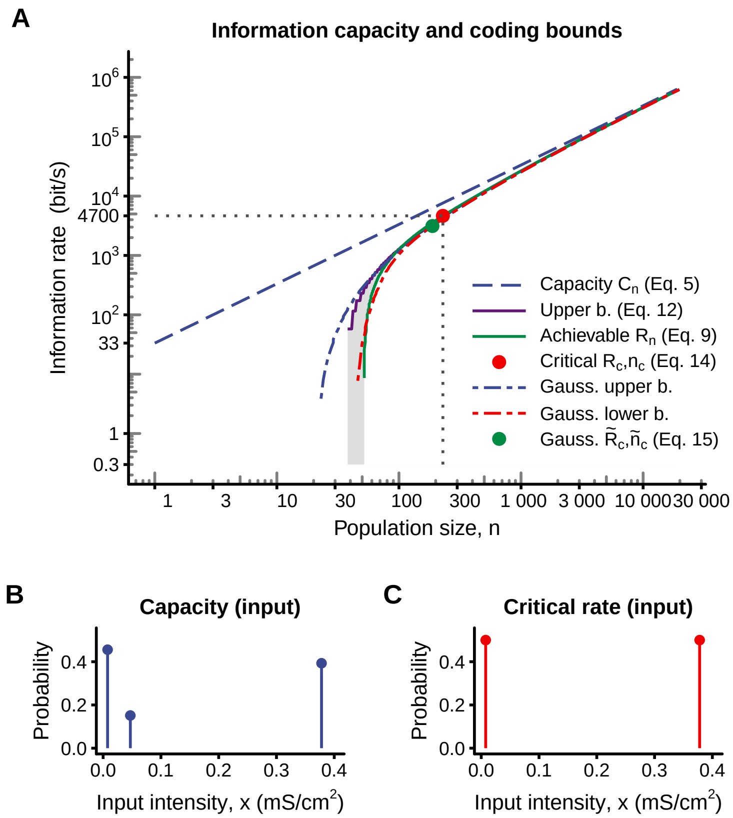

FIG. S1. Non-asymptotic behavior of achievable information transmission rate in a homogeneous neuronal population discretized to non-overlapping time intervals $\Delta = 25\,\mathrm{ms}$. The visualisation is analogous to Fig. 2 in the manuscript, other parameters of the conductance based model were not changed. The information capacity ($C \doteq 33\,\mathrm{bit/s}$ per neuron) is almost twice as large compared to the $\Delta = 50\,\mathrm{ms}$ case ($C \doteq 17.3\,\mathrm{bit/s}$ per neuron). Qualitatively, the situation looks similar to Fig. 2. Note that the Gaussian approximation works well in this case.

3. Synaptic input

The synaptic input I_{syn} is given by sum of excitatory and inhibitory input,

$$I_{\mathrm{syn}}(t) = g_e(t)(V - E_e) + g_i(t)(V - E_i), \tag{S84}$$

where $g_{e(i)}$ and $E_{e(i)}$ are the excitatory (inhibitory) synaptic conductances and reversal potentials, $E_e = 0\,\mathrm{mV}$, $E_i = -70\,\mathrm{mV}$. The synaptic conductances are described by the point-conductance model [19],

$$\frac{\mathrm{d}g_{e(i)}(t)}{\mathrm{d}t} = -\frac{g_{e(i)}(t) - \langle g_{e(i)}\rangle}{\tau_{e(i)}} + \sqrt{\frac{2\sigma^2_{e(i)}}{\tau_{e(i)}}}\eta_{e(i)}(t), \tag{S85}$$

where $\tau_{e(i)}$ is the time constant of the excitatory (inhibitory) synaptic conductance, and $\eta_{e(i)}(t)$ are mutually independent Gaussian white noises with zero means and unit variances. The asymptotic mean and variance of the synaptic conductances are $\langle g_{e(i)}\rangle$ and $\sigma^2_{e(i)}$, respectively. The standard deviations of the synaptic conductances are assumed to be proportional to the corresponding mean, $\sigma_e = \langle g_e\rangle/2$ and $\sigma_i = \langle g_i\rangle/4$. The parameters are $E_e = 0\,\mathrm{mV}$, $E_i = -70\,\mathrm{mV}$, $\tau_e = 2.7\,\mathrm{ms}$, and $\tau_i = 10.5\,\mathrm{ms}$. The effective potential V_r is defined by the voltage at which the mean synaptic current

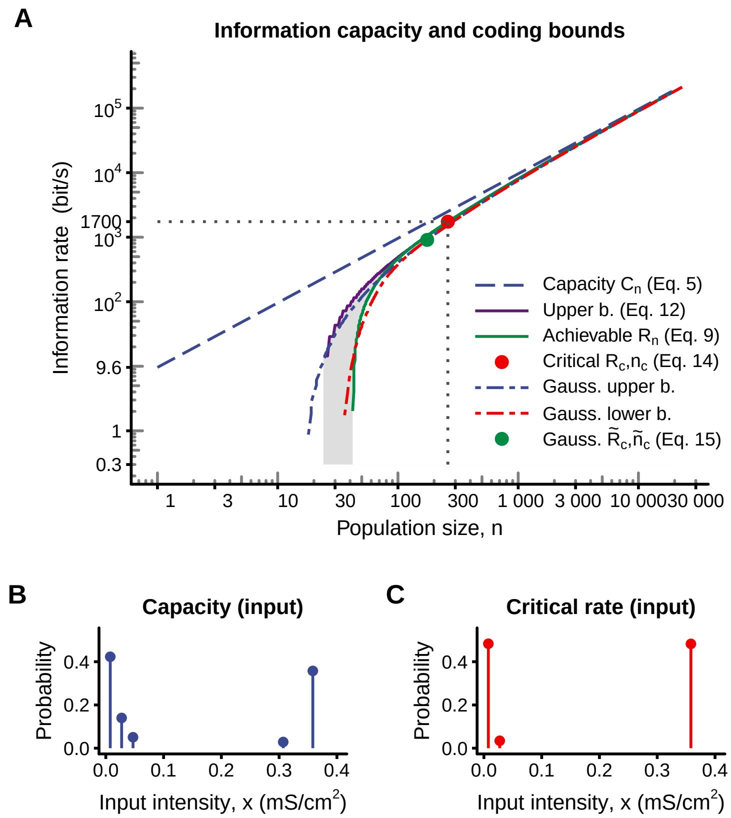

FIG. S2. For completeness we present a coarsely-grained situation with time intervals set to $\Delta = 100\,\text{ms}$, other model parameters unchanged (conductance-based model). The upper and lower bounds for the exact and Gaussian cases are in good agreement, the critical rate is approximated less accurately.

I_{syn} is equals to zero, and determines the proportionality between $\langle g_e \rangle$ and $\langle g_i \rangle$ (the balanced input [20–22])

$$\langle g_e \rangle (E_e - V_r) + \langle g_i \rangle (E_i - V_r) = 0. \tag{S86}$$

where $V_r = -65\,\text{mV}$ is the effective reversal potential. We confirmed that the result did not change qualitatively when the effective potential was changed.

E. Additional results

We assume that the neuron encodes the stimulus by the rate-code. The stimulus (mean excitatory synaptic conductance $\langle g_e \rangle$) and response (number of action potentials) are determined within non-overlapping time intervals of length Δ. The timescale $\Delta = 50\,\text{ms}$ is investigated in the manuscript (Fig. 2) was chosen to minimize the effect of stimulus history on the current response (so that the discrete-time memoryless channel assumption is not violated) while being biologically relevant. For completeness we show two other timescales ($\Delta = 25\,\text{ms}$ and $\Delta = 100\,\text{ms}$) while keeping other model parameters (conductance model) unchanged. In addition we show results for the SRM model ($\Delta = 50\,\text{ms}$).

– Fig. S1: $\Delta = 25\,\text{ms}$, conductance-based model. The shorter timescale is more interesting from the biological perspective, however, we believe that memory effects might start to play a non-negligible

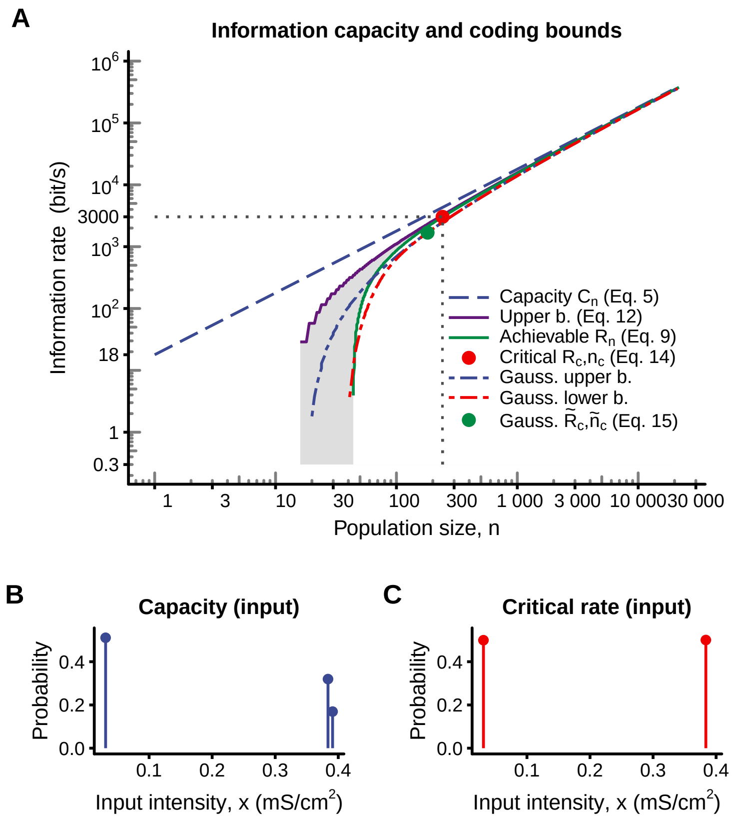

FIG. S3. Spike response model, $\Delta = 50\,\mathrm{ms}$. The capacity ($C \doteq 17.9\,\mathrm{bit/s}$ per neuron) is comparable to the conductance based model (Fig. 2), the upper bound in the Gaussian approximation is less accurate.

role here. Qualitatively, the situation looks similar to Fig. 2 in the manuscript. The quantitative indicators are changed, i.e., the capacity $C \doteq 33\,\mathrm{bit/s}$ per neuron and the critical parameters $R_c \doteq 20\,\mathrm{bit/s}$, $R_c n_c \doteq 4650\,\mathrm{bit/s}$ and the critical population size $n_c = 229$. The Gaussian approximation works well in this case and $\tilde{R}_c \doteq 17\,\mathrm{bit/s}$, $\tilde{R}_c \tilde{n}_c \doteq 3196\,\mathrm{bit/s}$ and $\tilde{n}_c = 188$.

– Fig. S2: $\Delta = 100$ conductance-based model. The timescale is perhaps too coarse-grained and is shown for completeness only. The capacity $C \doteq 9.6\,\mathrm{bit/s}$ per neuron and the critical parameters $R_c \doteq 6.6\,\mathrm{bit/s}$, $R_c n_c \doteq 1741\,\mathrm{bit/s}$, $n_c = 260$. The Gaussian approximation gives: $\tilde{R}_c \doteq 5.2\,\mathrm{bit/s}$, $\tilde{R}_c \tilde{n}_c \doteq 909\,\mathrm{bit/s}$ and $\tilde{n}_c = 175$.

– Fig. S3: $\Delta = 50\,\mathrm{ms}$, spike response model. The capacity $C \doteq 17.9\,\mathrm{bit/s}$ per neuron and the critical parameters $R_c \doteq 12.6\,\mathrm{bit/s}$, $R_c n_c \doteq 3039\,\mathrm{bit/s}$, $n_c = 241$. The Gaussian approximation, especially of the upper bound, is less accurate, however, the other parameters are comparable within the order of the magnitude: $\tilde{R}_c \doteq 9.3\,\mathrm{bit/s}$, $\tilde{R}_c \tilde{n}_c \doteq 1685\,\mathrm{bit/s}$ and $\tilde{n}_c = 181$.

[1] J. E. Kelley, "The cutting-plane method for solving convex programs," J. Soc. Industrial App. Math. **8**, 703–712 (1960).
[2] J. Huang and S. P. Meyn, "Characterization and computation of optimal distributions for channel coding," IEEE Trans. Inf. Theory **51**, 2336–2351 (2005).
[3] H. S. Witsenhausen, "Some aspects of convexity useful in information theory," IEEE Trans. Inf. Theory **26**, 265–271 (1980).

[4] R. G. Gallager, *Information Theory and Reliable Communication* (John Wiley and Sons, Inc., New York, USA, 1968).
[5] Y. Polyanskiy, *Channel coding: non-asymptotic fundamental limits* (PhD thesis, Princeton University, New Jersey, 2010).
[6] J. Huang, S. Meyn, and M. Medard, "Error exponents for channel coding with application to signal constellation design," IEEE J. Select. Areas Commun. **24**, 1647–1661 (2006).
[7] C. E. Shannon, "Probability of Error for Optimal Codes in a Gaussian Channel," Bell Syst. Tech. J. **38**, 611–656 (1959).
[8] R. Blahut, "Computation of channel capacity and rate-distortion functions," IEEE Trans. Inf. Theory **18**, 460–473 (1972).
[9] C. Chang and L. D. Davissson, "On calculating the capacity of an infinite-input finite (infinite)-output channel," IEEE Trans. Inf. Theory **34**, 1004–1010 (1988).
[10] V. Strassen, "Asymptotische Abschätzungen in Shannons Informationstheorie," in *Trans. Third Prague Conf. Inf. Theory., Prague* (1962) pp. 689–723.
[11] A. L. Hodgkin and A. F. Huxley, "Propagation of electrical signals along giant nerve fibres," Proc. Roy. Soc. B **140**, 177–183 (1952).
[12] J. Benda and A. V. M. Herz, "A universal model for spike-frequency adaptation," Neural Comput. **15**, 2523–2564 (2003).
[13] W. Gerstner and W. M. Kistler, *Spiking Neuron Models: Single Neurons, Populations, Plasticity* (Cambridge University Press, Cambridge, 2002).
[14] T. Toyoizumi, J-P. Pfister, K. Aihara, and W. Gerstner, "Generalized Bienenstock–Cooper–Munro rule for spiking neurons that maximizes information transmission," Proc. Natl. Acad. Sci. USA **102**, 5239–5244 (2005).
[15] J-P. Pfister, T. Toyoizumi, D. Barber, and W. Gerstner, "Optimal spike-timing-dependent plasticity for precise action potential firing in supervised learning," Neural Comput. **18**, 1318–1348 (2006).
[16] R. Kobayashi and S. Shinomoto, "State space method for predicting the spike times of a neuron," Phys. Rev. E **75**, 011925 (2007).
[17] R. Jolivet, R. Kobayashi, R. Rauch, R. Naud, S. Shinomoto, and W. Gerstner, "A benchmark test for a quantitative assessment of simple neuron models," J. Neurosci. Methods **169**, 417–424 (2008).
[18] R. Kobayashi and K. Kitano, "Impact of slow K^+ currents on spike generation can be described by an adaptive threshold model," J. Comput. Neurosci. **40**, 347–362 (2016).
[19] A. Destexhe, M. Rudolph, J. M. Fellous, and T. J. Sejnowski, "Fluctuating synaptic conductances recreate in vivo-like activity in neocortical neurons," Neuroscience **107**, 13–24 (2001).
[20] M. Wehr and A. M. Zador, "Balanced inhibition underlies tuning and sharpens spike timing in auditory cortex," Nature **426**, 442–446 (2003).
[21] R. W. Berg, A. Alaburda, and J. Hounsgaard, "Balanced inhibition and excitation drive spike activity in spinal half-centers," Science **315**, 390–393 (2007).
[22] R. Kobayashi, J. He, and P. Lansky, "Estimation of the synaptic input firing rates and characterization of the stimulation effects in an auditory neuron," Front. Comput. Neurosci. **9**, 59 (2015).