

## Supplementary Note

### Autoassociative recall as probabilistic inference

We first specify the task for autoassociative recall in a normative manner<sup>1,2</sup>. This specification leads to a natural account of the dynamics of the neurons during recall, whose form is largely determined by the learning rule. Unfortunately, the full dynamics includes terms that are not purely local to the information a postsynaptic neuron has about presynaptic activity. We therefore consider approximations that are based on basic biological constraints. The simulations (**Fig. 2**, and **Supplementary Fig. 1**) verify that the approximations are not disastrous for memory.

### The posterior distribution over activity patterns

Consider a device that has stored information about  $M$  traces  $\mathbf{x}^1 \dots \mathbf{x}^M$ . In each retrieval trial, it is presented with an input  $\tilde{\mathbf{x}}$ , the recall cue, which is supposed to be a noisy version of one of the previously stored traces. A complete description of its task then is to report a distribution  $P[m|\tilde{\mathbf{x}}]$  as to the posterior probability that memory trace  $\mathbf{x}^m$  was the source of noisy input  $\tilde{\mathbf{x}}$ . Importantly, several potential candidates will have a probability greater than one to be the correct trace that should be recalled. Uncertainty arises because of input noise. Depending on a loss function, which quantifies the cost of retrieval errors, it is also possible to consider related tasks such as reporting the single most likely trace.

In the case we consider, the patterns are represented on a network of  $N$  neurons recurrently coupled with weights  $\mathbf{W}$ . Further, rather than reporting the distribution over the *indices* of the memory traces, the network should retrieve the posterior distribution  $P[\mathbf{x}|\tilde{\mathbf{x}}, \mathbf{W}]$  over the *activities* of all the neurons  $\mathbf{x}$ , also given the weights. This means that we aim ideally to compute the probability for *any* given activity pattern  $\mathbf{x}$  that it is the correct pattern to recall given that the network receives input  $\tilde{\mathbf{x}}$  and has synaptic weights  $\mathbf{W}$ , and therefore we consider methods that work in the full space of activities  $\mathbf{x}$ . This posterior has added uncertainty which arises since biological synaptic plasticity rules are data-lossy ‘compression algorithms’ (see main paper), and so  $\mathbf{W}$  specifies only imprecise information about the stored traces, in particular not being restricted to just  $\mathbf{x} \in \{\mathbf{x}^m\}$ .

The posterior distribution can be decomposed into three terms, following Bayes’ rule:

$$P[\mathbf{x}|\tilde{\mathbf{x}}, \mathbf{W}] \propto P[\mathbf{x}] P[\tilde{\mathbf{x}}|\mathbf{x}] P[\mathbf{W}|\mathbf{x}] \quad (\text{S1})$$

The terms in Equation S1 can be regarded as providing answers to the three different questions that can be asked about a particular pattern  $\mathbf{x}$  in this setting.

- ‘How probable is it in general that  $\mathbf{x}$  could be one of the patterns stored?’ The first term in Equation S1,  $P[\mathbf{x}]$ , specifies prior knowledge as to the statistical characteristics of the memories: it describes the distribution from which memory traces are drawn, and therefore gives the probability that  $\mathbf{x}$  could be one of those stored previously without considering any further information. We assume that it factorizes, *i.e.* the activity of each neuron is generated independently from the others when memory traces are constructed:  $P[\mathbf{x}] = \prod_i P_x[x_i]$ .
- ‘How likely is it that input  $\tilde{\mathbf{x}}$  is a noisy version of  $\mathbf{x}$ ?’ The second term in Equation S1,  $P[\tilde{\mathbf{x}}|\mathbf{x}]$ , describes the noise process corrupting the inputs: it describes the distribution from which input  $\tilde{\mathbf{x}}$  is drawn if memory trace  $\mathbf{x}$  is to be retrieved. For unbiased noise it will be a term in  $\mathbf{x}$  that is effectively centered on  $\tilde{\mathbf{x}}$ . We also assume that the noise corrupting each element of the patterns is independent, and independent of the original pattern, so  $P[\tilde{\mathbf{x}}|\mathbf{x}] = \prod_i P[\tilde{x}_i|\mathbf{x}] = \prod_i P[\tilde{x}_i|x_i]$ .
- ‘How likely is it that weight matrix  $\mathbf{W}$  came about by storing  $\mathbf{x}$  among other patterns?’ The third term in Equation S1,  $P[\mathbf{W}|\mathbf{x}]$ , assesses the likelihood that the weight matrix came from storing a training set of size  $M$  including pattern  $\mathbf{x}$ , together with  $M - 1$  other unspecified patterns coming from the prior distribution. (Uncertainty about  $M$  could also be incorporated into the model, but is neglected here.) This distribution is complicated in  $\mathbf{x}$ -space, but we show below that, under some simple assumptions about the nature of synaptic plasticity, it will be Gaussian in  $\mathbf{W}$ -space, to a good approximation.

Under our abstraction of synaptic plasticity, the learning updates for the synapse from neuron  $j$  to neuron  $i$  are *local* to the presynaptic ( $x_j^m$ ) and postsynaptic ( $x_i^m$ ) activities of connected neurons when pattern  $\mathbf{x}^m$  is stored:

$$\Delta w_{ij}^m = \Omega(x_i^m, x_j^m) \tag{S2}$$

Further, we assume that contributions of individual training patterns are additive,  $w_{ij} = \sum_m \Delta w_{i,j}^m$ , and that there are no autapses in the network,  $w_{ii} = 0$ .

We first consider the effects of storing  $M - 1$  unspecified patterns coming from the prior. Storing a single random pattern drawn from the prior distribution will result in a synaptic weight change with a distribution determined by the prior and the learning

rule, which, by definition, has mean

$$\mu_{\Delta w} = \langle \Omega(x_1, x_2) \rangle_{P_{x[x_1]} \cdot P_{x[x_2]}} \quad (\text{S3})$$

and variance

$$\sigma_{\Delta w}^2 = \langle \Omega^2(x_1, x_2) \rangle_{P_{x[x_1]} \cdot P_{x[x_2]}} - \mu_{\Delta w}^2 \quad (\text{S4})$$

Thus, storing (unspecified)  $M - 1$  random patterns implies adding  $M - 1$  independent, identically distributed random variables, and results, for at least moderately large  $M$ , in a synaptic weight with an approximately Gaussian distribution  $P[w_{ij}] \simeq \mathcal{G}(w_{ij}; \mu_w, \sigma_w)$ , with mean  $\mu_w = (M - 1) \mu_{\Delta w}$  and variance  $\sigma_w^2 = (M - 1) \sigma_{\Delta w}^2$ .

Adding a further *particular* pattern  $\mathbf{x}$  is equivalent to adding a random variable with a mean determined by the learning rule, and zero variance, thus:

$$P[w_{ij}|x_i, x_j] \simeq \mathcal{G}(w_{ij}; \mu_w + \Omega(x_i, x_j), \sigma_w) \quad (\text{S5})$$

We also make the approximation that elements of the synaptic weight matrix are independent, except for symmetric pairs  $w_{ij}$  and  $w_{ji}$  which are perfectly correlated for symmetric or antisymmetric learning rules (considered later), and thus write:

$$P[\mathbf{W}|\mathbf{x}] = \prod_{i,j < i} P[w_{ij}|x_i, x_j] \quad (\text{S6})$$

### The objective function

Rather than reporting the whole posterior distribution of Equation S1, we consider the decision-theoretic task of finding the single most likely trace<sup>3</sup>, *i.e.* the activities that maximize the posterior probability:

$$\hat{\mathbf{x}} = \arg \max_{\mathbf{x}} P[\mathbf{x}|\tilde{\mathbf{x}}, \mathbf{W}] \quad (\text{S7})$$

This means that we aim to find the activity pattern  $\mathbf{x}$  that is most probably the correct pattern to recall given input  $\tilde{\mathbf{x}}$  and synaptic weights  $\mathbf{W}$ . Having restricted our horizons to maximum a posteriori (MAP) inference, we can consider as an objective function the log of the posterior distribution. In the light of our factorizability assumptions, this

is

$$\begin{aligned} \mathcal{O}(\mathbf{x}) &= \log P[\mathbf{x}] + \log P[\tilde{\mathbf{x}}|\mathbf{x}] + \log P[\mathbf{W}|\mathbf{x}] \\ &= \sum_i \log P[x_i] + \sum_i \log P[\tilde{x}_i|x_i] + \sum_{i,j < i} \log P[w_{ij}|x_i, x_j] \end{aligned} \quad (\text{S8})$$

### Optimal neuronal update dynamics

Finding the global maximum of the objective function, as stated in Equation S7 is computationally extravagant and biologically questionable: in general, it would require an exhaustive scanning of the  $N$ -dimensional space of activities. In common with the vast bulk of work on associative memory<sup>2,4</sup>, we therefore specify neuronal dynamics arising from gradient ascent on the objective function:

$$\dot{\mathbf{x}} \propto \nabla_{\mathbf{x}} \mathcal{O}(\mathbf{x}) \quad (\text{S9})$$

This ensures that neurons in the network change their joint activity at every point in time such that the value of the objective function evaluated at the current activity pattern is always increased. Since the objective function is the log-posterior (Equation S8), this means that the activity of the network evolves such that it represents increasingly probable solutions to the retrieval problem. Therefore the network is guaranteed to find at least a local maximum of the posterior distribution, *i.e.* an activity pattern that is locally maximally probable to be the correct memory trace to recall.

Combining Equations S8 and S9 we obtain the optimal update dynamics describing how neuron  $i$  should change its activity  $x_i$

$$\tau_x \frac{dx_i}{dt} = \frac{\partial}{\partial x_i} \log P[\mathbf{x}] + \frac{\partial}{\partial x_i} \log P[\tilde{\mathbf{x}}|\mathbf{x}] + \frac{\partial}{\partial x_i} \log P[\mathbf{W}|\mathbf{x}] \quad (\text{S10})$$

$$\frac{\partial}{\partial x_i} \log P[\mathbf{W}|\mathbf{x}] = \frac{1}{2} \sum_{j \neq i} \frac{\partial}{\partial x_i} \log P[w_{ij}|x_i, x_j] + \frac{\partial}{\partial x_i} \log P[w_{ji}|x_j, x_i] \quad (\text{S11})$$

The first two terms in Equation S10 only depend on the activity of the neuron itself and its input. For example, for a Gaussian prior  $P_x[x_i] = \mathcal{G}(x_i; \mu_x, \sigma_x)$  and unbiased Gaussian noise on the input  $P[\tilde{x}_i|x_i] = \mathcal{G}(\tilde{x}_i; x_i, \sigma_{\tilde{x}})$ , these would be:

$$\frac{d}{dx_i} \log P[x_i] + \frac{d}{dx_i} \log P[\tilde{x}_i|x_i] = \frac{\mu_x}{\sigma_x^2} - \left( \frac{1}{\sigma_x^2} + \frac{1}{\sigma_{\tilde{x}}^2} \right) x_i + \frac{\tilde{x}_i}{\sigma_{\tilde{x}}^2} \quad (\text{S12})$$

The first term on the right-hand side of the last equality expresses a constant bias; the second involves self-decay; and the third describes the effect of the input.

The terms in Equation S11 indicate how a neuron should take into account the activity of other neurons based on the synaptic weights. From Equation S5, the terms are

$$\begin{aligned} \frac{\partial}{\partial x_i} \log P[w_{ij}|x_i, x_j] &= \\ &= \frac{1}{\sigma_w^2} \left[ (w_{ij} - \mu_w) \frac{\partial}{\partial x_i} \Omega(x_i, x_j) - \Omega(x_i, x_j) \frac{\partial}{\partial x_i} \Omega(x_i, x_j) \right] \end{aligned} \quad (\text{S13})$$

$$\begin{aligned} \frac{\partial}{\partial x_i} \log P[w_{ji}|x_j, x_i] &= \\ &= \frac{1}{\sigma_w^2} \left[ (w_{ji} - \mu_w) \frac{\partial}{\partial x_i} \Omega(x_j, x_i) - \Omega(x_j, x_i) \frac{\partial}{\partial x_i} \Omega(x_j, x_i) \right] \end{aligned} \quad (\text{S14})$$

Two aspects of the above formulæ are troubling. First, the last terms in each express the effects of other cells, but in a way that is independent of the synaptic weights. We therefore approximate these terms using their mean values over the prior distribution. In this case

$$\alpha_i^+ = \langle \Omega(x_i, x_j) \frac{\partial}{\partial x_i} \Omega(x_i, x_j) \rangle_{P_x[x_j]} \text{ and } \alpha_i^- = \langle \Omega(x_j, x_i) \frac{\partial}{\partial x_i} \Omega(x_j, x_i) \rangle_{P_x[x_j]} \quad (\text{S15})$$

contribute terms that only depend on the activity of the updated cell, and so can be lumped with the prior- and input-dependent terms of Equation S12. For the simulations reported in the main paper (**Fig. 2**) we used these approximate formulæ. For the supplementary simulations (**Supplementary Fig. 1**) we used the exact formulæ in order to study the effects of other approximations and factors more clearly. In our experience, the approximation was benign when the number of stored memories was not too small ( $M > 10$  in a network of  $N = 100$  neurons).

Second, Equation S14 includes synaptic weights,  $w_{ji}$ , that are *postsynaptic* with respect to the updated neuron. These terms would seem to require the neuron to change its activity depending on the weights of its postsynaptic synapses. For perfectly anti-symmetric learning rules, such as the STDP rules considered in the main paper, this is unnecessary, since  $\Omega(x_i, x_j) = -\Omega(x_j, x_i)$ , and therefore  $w_{ji} = -w_{ij}$  is known. Thus, *exact* gradient ascent is performed on the weight likelihood (Equation S5) if each neuron only considers contributions from its presynaptic weights ignoring its postsynaptic weights. The same is true for perfectly symmetric learning rules. For asymmetric learning rules, ignoring the postsynaptic weights amounts to an approxi-

mation, the quality of which we tested in supplementary simulations (**Supplementary Fig. 1c**).

Making these assumptions, we can rewrite Equation S11 in the following way:

$$\frac{\partial}{\partial x_i} \log P[\mathbf{W}|\mathbf{x}] = \frac{1}{\sigma_w^2} \left[ \sum_{j \neq i} H(x_i, x_j) - (N-1) \alpha_i \right] \quad (\text{S16})$$

showing that a postsynaptic neuron should sum the effects of its presynaptic partners, with the effect of a presynaptic neuron on a postsynaptic neuron given by the neural interaction function

$$H(x_i, x_j) = w_{ij} \cdot \frac{\partial}{\partial x_i} \Omega(x_i, x_j) \quad (\text{S17})$$

Equation S17 is one of our key results, showing that there is a simple relationship between the synaptic plasticity rule,  $\Omega(x_i, x_j)$ , and the neuronal interaction function,  $H(x_i, x_j)$ , that is approximately optimal for reading out the information that is encoded in the synaptic weight matrix by that synaptic plasticity rule. It also shows that the magnitude of this interaction should be proportional to the synaptic weight connecting the two cells,  $w_{ij}$ . For further discussion of this result, see the main paper.

## Spike timing-based networks

### The optimal coupling function

The theory we have just described allows us to treat the problem of spike timing-based autoassociative memories systematically. First, we interpret neuronal activities,  $x_i$  as firing times relative to a reference phase of the ongoing local field potential oscillation, such as the peak of theta oscillation in the hippocampus<sup>5</sup>, and will thus be circular variables. We will model them as being drawn from von Mises distributions, which are also called circular Gaussians. Next, our learning rule is an exponentially decaying Gabor-function of the phase difference between pre- and postsynaptic firing:

$$\Omega_{\text{STDP}}(x_i, x_j) = A \exp[s \cos(\Delta\phi_{ij})] \sin(\Delta\phi_{ij} - \phi_0) \quad (\text{S18})$$

with  $\Delta\phi_{ij} = 2\pi(x_i - x_j)/T_\theta$ . STDP characteristics in different brain regions are well captured by this general formula, but the parameters determining their exact shapes greatly differ among regions<sup>6</sup>. We constrain our analysis to the antisymmetric case, so that  $\phi_0 = 0$ , and set other parameters to match experimental data on hippocampal STDP<sup>7</sup> (see **Fig. 1b**, and Methods in the main paper, and also **Supplementary**

**Fig. 2).** The neuronal interaction function that satisfies our general matching rule (Equation S17) by substituting  $\Omega_{\text{STDP}}$  (Equation S18) into it is

$$H_{\text{STDP}}(x_i, x_j) = 2\pi A/T_\theta w_{ij} \exp(s \cos(\Delta\phi_{i,j})) [\cos(\Delta\phi_{ij}) - s \sin^2(\Delta\phi_{ij})] \quad (\text{S19})$$

(see **Fig. 1c**).

### The optimal phase response curve

The interaction function in Equation S19 (taken together with Equations S10 and S16) specifies how presynaptic neuron  $j$  should influence the change in postsynaptic neuron  $i$ 's firing phase relative to the ongoing LFPO ( $x_i$ ) depending on the difference of their firing phases ( $x_i - x_j$ ). However, this relationship describes a continuous interaction between the two cells, *i.e.* a presynaptic cell should be able to affect its postsynaptic partners by reporting its 'firing phase' at all times, even times during the interspike intervals between its spikes.

If neurons are only allowed to interact at times when they fire, the most straightforward approximation to the key differential Equation S9, which indicates the computation the network is performing, is to assume that the firing phases of neurons are constant during interspike intervals. This is equivalent to an event-based approach of discretizing time at the spiking events.

If the prior distribution  $P[x_i]$  of memories is a (circular Gaussian) distribution with 0 mean and  $k_x$  concentration parameter, and no other sources influence the neuron (*ie* only the prior-dependent term contributes to Equation S10), we expect its firing phase to obey the following equation:

$$\tau_x \frac{dx_i}{dt} = -k_x \sin x_i(t) \quad (\text{S20})$$

This equation has a stable solution at  $x_i = 0$  (henceforth called the 'baseline' firing phase), when the neuron phase locks to the LFPO. In other words, phase locking to an LFPO can be interpreted as an implementation of a prior distribution in our theoretical framework.

To account for the influence of a single spike fired by presynaptic neuron  $j$  at phase  $x_j$ , the postsynaptic neuron needs to implement an extended equation from the moment that spike occurred:

$$\tau_x \frac{dx_i}{dt} = -k_x \sin x_i(t) + \frac{w_{ij}}{\sigma_w^2} H_{\text{STDP}}(x_i(t), x_j) \quad (\text{S21})$$

Here, given the discretization described above,  $x_j(t) = x_j$  is constant until neuron  $j$  spikes again.

In order to determine when neuron  $i$  actually fires, Equation S21 needs to be integrated until neuron  $i$  reaches its spiking point on its own spiking cycle<sup>8</sup>. Because  $x_i$  is its firing phase *relative* to the LFPO, its phase on its own spiking cycle is

$$\psi_i(t) = x_i(t) + 2\pi f_\theta t \quad (\text{S22})$$

where  $f_\theta$  is the frequency of the LFPO. This means that Equation S21 needs to be integrated until

$$x_i(t) = -2\pi f_\theta t \pmod{2\pi} \quad (\text{S23})$$

( $\psi = 0$  defined to be the phase of spiking in the spiking cycle). We can therefore predict the relative phase of the next spike of the neuron by integrating Equation S21 starting from  $x_i(0) = 0$  until Equation S23 is satisfied, with different  $x_j$  values corresponding to different phases at which stimulation is applied. The deviation of the phase of the next spike from the baseline phase can be measured as a function of the perturbation phase, and the plot relating these two quantities is exactly the phase response curve (PRC)<sup>8</sup>.

The predicted PRC depends on a number of parameters, out of which we systematically explored the effects of changing  $k_x$  and  $w_{ij}$  (with  $\tau_x = 1$  and  $\sigma_w^2 = 1$ , as they only scale the effect of the other two parameters and thus were redundant for our purposes). We found PRCs qualitatively similar to those shown in **Figure 1d** ( $k_x = 0.6$ ,  $w_{ij} = 0.025, 0.05, 0.075, 0.01$ ) in a broad parameter regime, in which both parameters were changed by a factor of 10. We also tested the salience of various characteristics of the optimal PRC (listed in the Results section of the main paper) by starting our derivations from a different type of STDP curve (with  $s = 0$  in Equation S18) but which still retained the most important properties of experimentally described forms of STDP (**Supplementary Fig. 2a**). **Supplementary Figure 2c** shows that the matching PRC was again similar to that reported in the main paper.

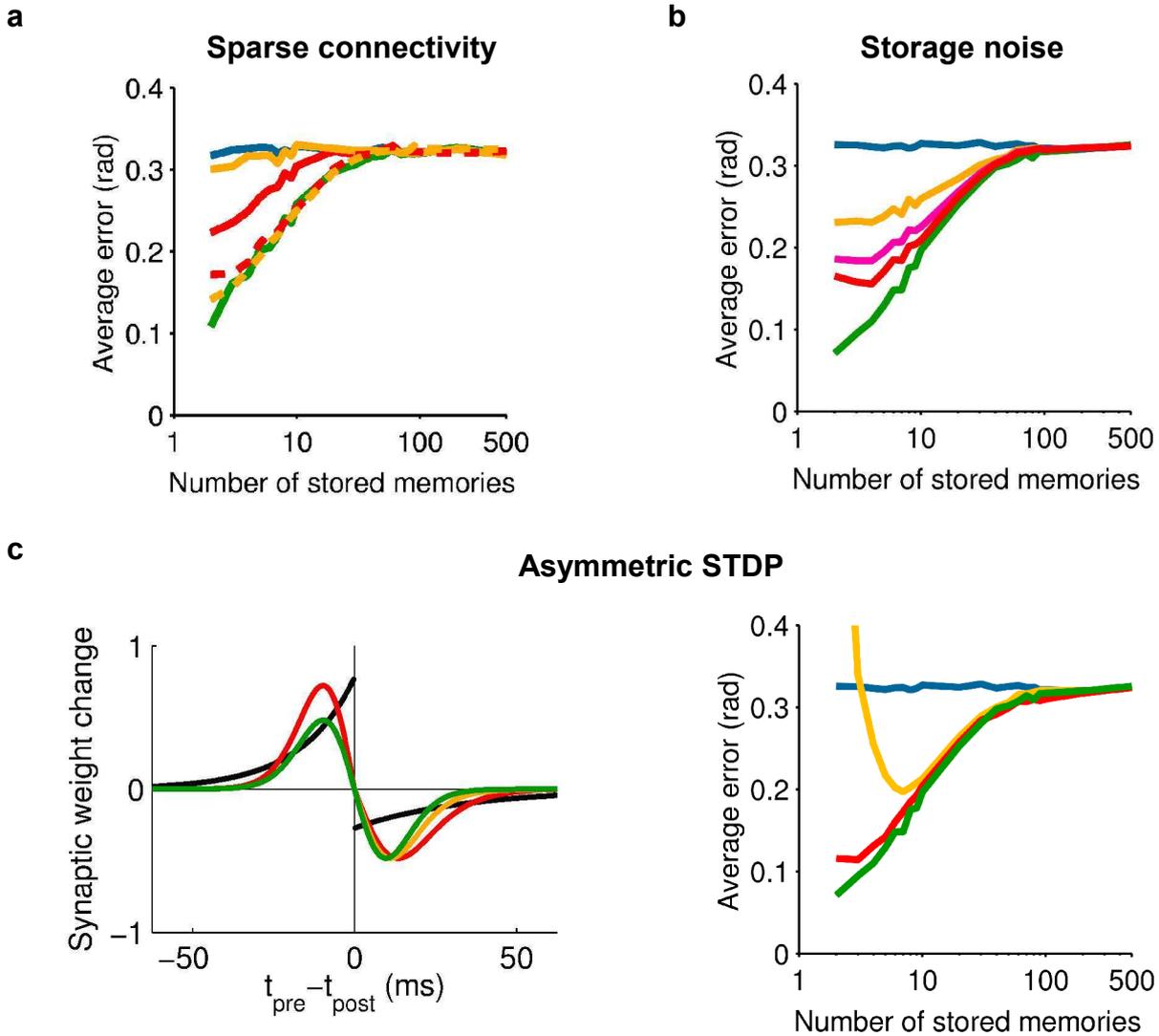
The main difference between the phase coupling functions (**Fig. 1c** and **Supplementary Fig. 2b**) and the corresponding PRCs (**Fig. 1d** and **Supplementary Fig. 2c**, respectively) is that in the PRCs there is no visible delay for presynaptic spikes arriving just before a postsynaptic spike. Crudely, each point of the PRC is obtained by integrating the value of the phase coupling function associated with the corresponding (near-constant) difference in phase between pre- and postsynaptic spikes across the postsynaptic spiking cycle until a spike. Thus, effects that happen early in a cycle (pos-

itive phases on the x-axis) will be integrated over a longer time than the effects due to perturbations in the late part of the spiking cycle (negative phases on the x-axis). This asymmetry in integration times explains the asymmetry of the PRCs, even if the underlying phase coupling functions are symmetric.

It is these predicted PRCs that allow a direct experimental test of our theory. We achieve this by injecting a controlled sinusoidal current into CA3 pyramidal cells. This induces a cell to phase lock to the oscillation at the baseline phase (**Fig. 3c, 4a**). The cell can then be subjected to an additional stimulus at a given phase of its oscillation. The empirical PRC is the deviation of the phase of the next spike from the baseline phase, as a function of the perturbation phase. These PRCs can be compared with those derived from the STDP rule.

## References

1. MacKay, D.J.C. Maximum entropy connections: neural networks. in *Maximum Entropy and Bayesian Methods, Laramie, 1990* (eds. Grandy, Jr, W.T. & Schick, L.H.) 237–244 (Kluwer, Dordrecht, The Netherlands, 1991).
2. Sommer, F.T. & Dayan, P. Bayesian retrieval in associative memories with storage errors. *IEEE Trans. Neural Netw.* **9**, 705–713 (1998).
3. Jaynes, E.T. *Probability Theory: the Logic of Science* (Cambridge University Press, Cambridge, UK, 2003).
4. Hopfield, J.J. Neurons with graded response have collective computational properties like those of two-state neurons. *Proc. Natl. Acad. Sci. USA* **81**, 3088–3092 (1984).
5. O’Keefe, J. & Recce, M.L. Phase relationship between hippocampal place units and the EEG theta rhythm. *Hippocampus* **3**, 317–330 (1993).
6. Abbott, L.F. & Nelson, S.B. Synaptic plasticity: taming the beast. *Nat. Neurosci.* **3**, 1178–1183 (2000).
7. Bi, G.Q. & Poo, M.M. Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type. *J. Neurosci.* **18**, 10464–10472 (1998).
8. Rinzel, J. & Ermentrout, B. Analysis of neural excitability and oscillations. in *Methods in Neuronal Modeling 2<sup>nd</sup> edn.* (eds. Koch, C. & Segev, I.) 251–291 (MIT Press, Cambridge, MA, 1998).

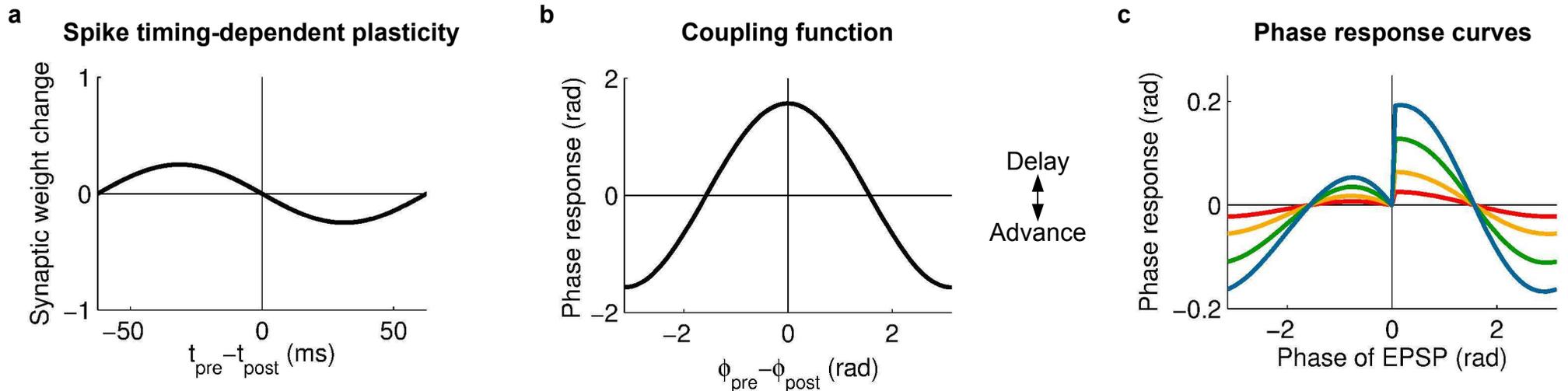


### Recall performance in adversarial settings.

**(a)** For a better match with hippocampal anatomy the connectivity of the network was reduced from full (green line, 100%) to increasing levels of sparsity (red lines, 50%; yellow lines, 20%). The direct determinant of recall performance was the number of synapses per neuron: increasing the number of neurons proportionately ameliorated the effects of reduced connectivity (red dashed line, doubled network size; yellow dashed line, quintupled network size). As a baseline, the input-only network (blue line) is also shown.

**(b)** Increasing storage noise degrades performance gradually (variance of noise: green line, 0; red line, 0.1; pink line, 0.2; yellow line, 0.5; variance of the contribution of a single memory is  $\sim 0.04$ ). As a baseline, the input-only network (blue line) is also shown.

**(c)** Left: the original STDP curve used in the main text (green line) was modified to better approximate STDP in cultured hippocampal neurons (black line, Bi & Poo, 1998) by making it weakly (yellow line, time constant of LTD part increased by 33%) or strongly (red line, time constant of LTD part doubled, amplitude of LTP increased by 50%) asymmetric.  $t_{pre}$ ,  $t_{post}$  times of pre- and postsynaptic firing. Right: recall performance of the network when phase-coupling function was (red line) or was not (yellow line) matched to the asymmetric STDP. For comparison, the network with symmetric STDP (green line) and the input-only network (blue line) is also shown. Even though the derivations of the matching rule (Eq.3, main text) assumed a symmetric STDP, a network with strongly asymmetric STDP performs almost identically to the one with symmetric STDP if the phase coupling-function is matched to it (red line), but even mild perturbations to the STDP result in poorer performance if the phase-coupling function is not adjusted appropriately (yellow line).

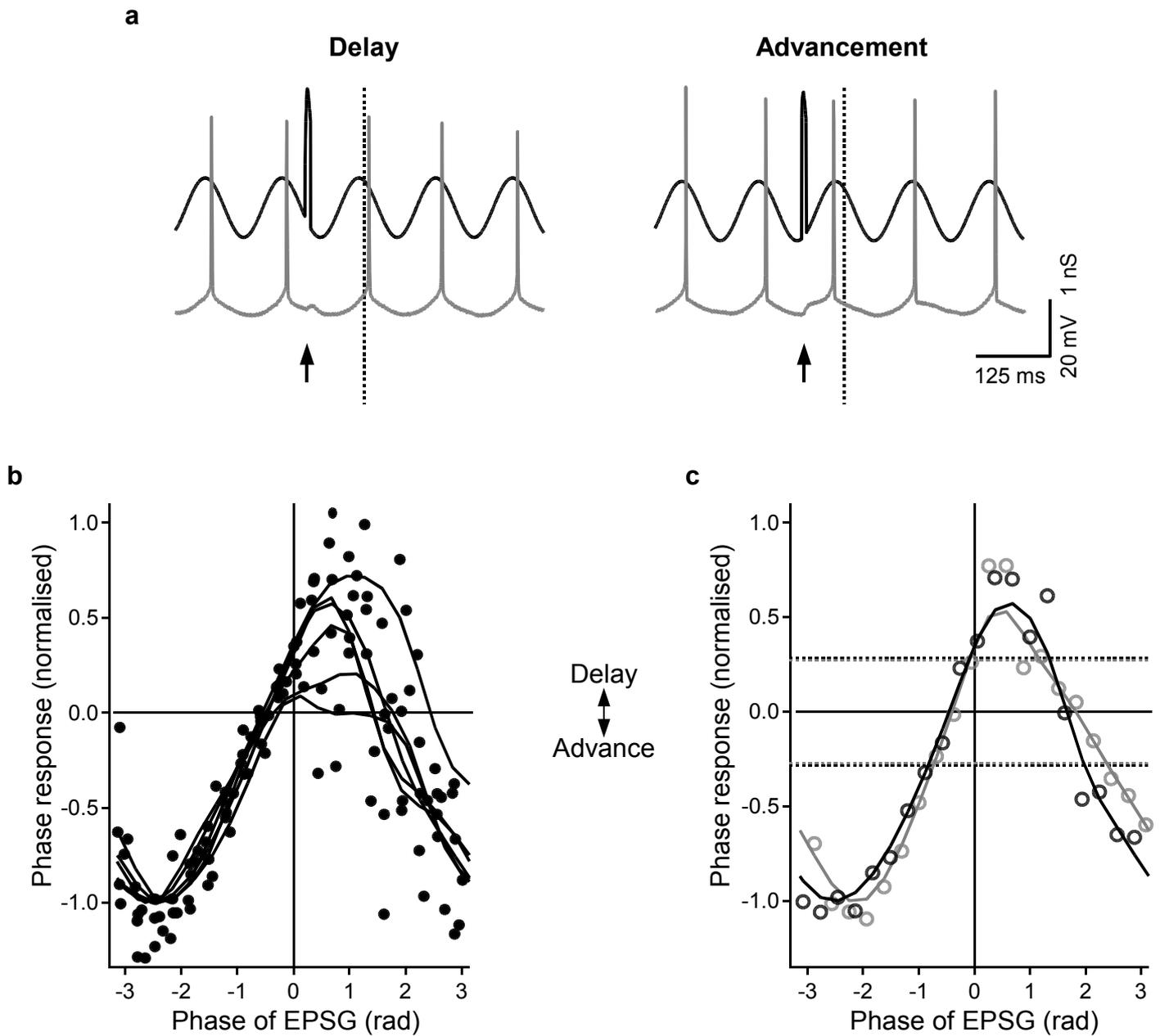


### Consequences of a broad, smoothly varying STDP curve on the optimal coupling function and phase response curves.

**(a)** Memories are stored by a STDP rule that is broader and more smoothly varying than most experimentally described forms of STDP (but see C.D. Meliza, N. Caporale & Y. Dan. Spike timing-dependent plasticity of visually evoked synaptic responses, *Soc. Neurosci. Abstr.* 57.10, 2004). Nevertheless, main characteristics of STDP are still reflected: a synapse is strengthened if the presynaptic neuron fires before the postsynaptic neuron, and is weakened if the order of firings is reversed, and time differences beyond an ideal value result in decreasing synaptic weight change. The parameters used for this STDP curve were  $s=0$ ,  $A=0.25$ , and  $T_\theta=125$  ms (see Methods and Supplementary Note).  $t_{pre}$ ,  $t_{post}$ , times of pre- and postsynaptic firing.

**(b)** Optimal coupling function for retrieving memories stored by STDP shown in **a**.  $\phi_{pre}$ ,  $\phi_{post}$ , firing phases of pre- and postsynaptic cells relative to a local field potential oscillation. (For further explanation see **Figure 1c** of the main paper.)

**(c)** Optimal phase response curves derived from the optimal coupling function (shown in **b**). Different curves correspond to linearly increasing synaptic weights ( $w_{ij}$ ) (0.025, red; 0.05, yellow; 0.075, green; 0.1, blue),  $k_x = 0.6$  in all cases. (For further details and explanations, see Supplementary Note online, and **Figure 1d** of the main paper.)

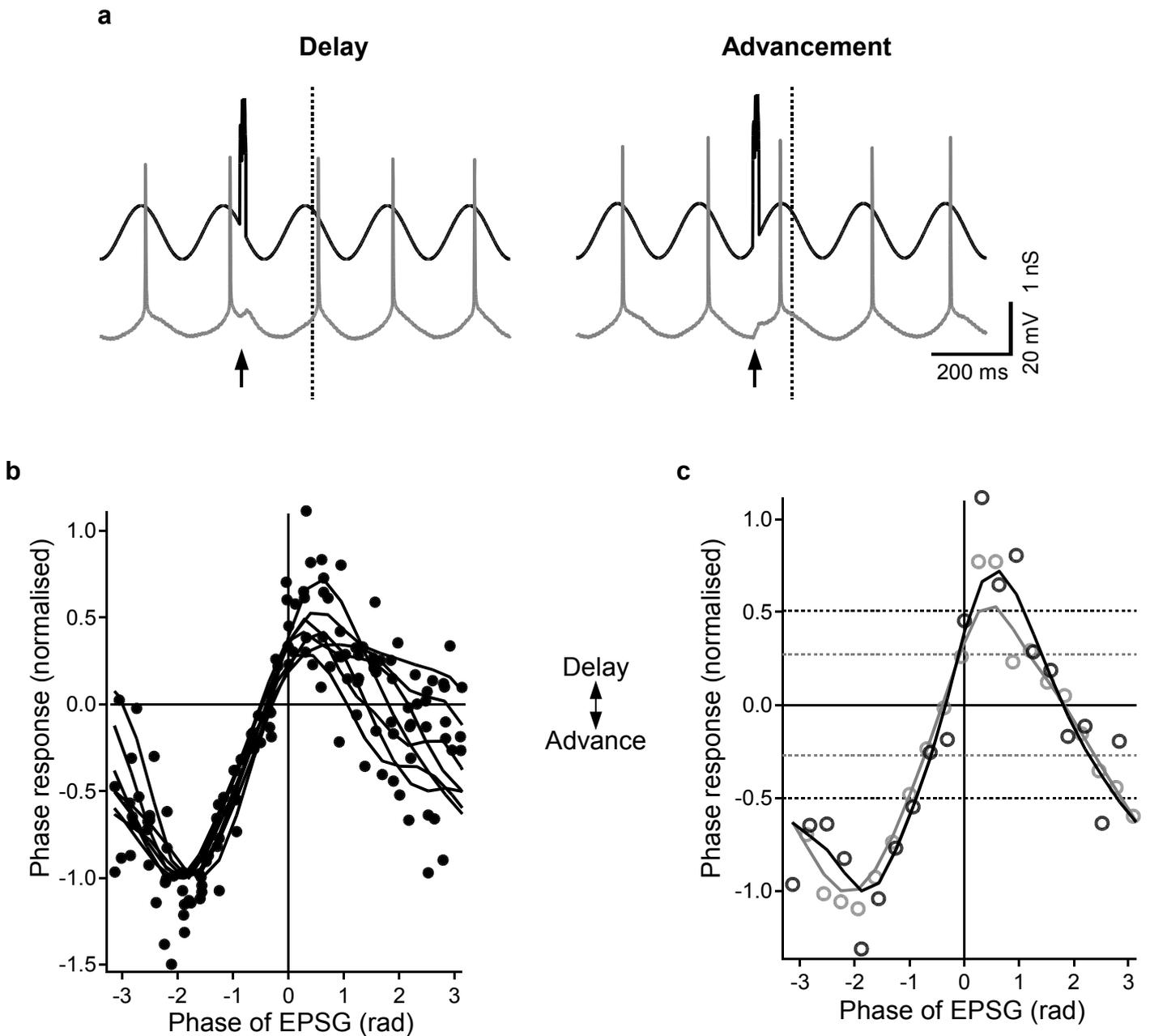


### Effect of increased oscillatory frequency on the PRC.

**(a)** Sample of current-clamp recordings showing the response of a CA3 neuron (gray trace) to the EPSG (arrows: times of stimulation) during 8 Hz oscillation (black trace). The vertical dashed line represents the average phase at which spikes occurred during 8 Hz oscillation without EPSG. As with 5 Hz oscillation, EPSG could delay or advance the spike phase following stimulation during 8 Hz oscillation.

**(b)** Smoothed PRCs (black traces) and raw data points (dots) obtained from CA3 neurons during 8 Hz oscillation ( $n = 6$ ). PRCs were smoothed and normalized as described in **Figure 3d-e** of the main paper. All PRCs show prominent phase advancement and delay of spikes.

**(c)** One of the PRCs from **b** (black trace and circles) superimposed onto a PRC during 5 Hz oscillation (gray trace and circles). Both PRCs were obtained with same amplitude EPSG stimulation (1.5 nS). Black and gray horizontal dashed lines show  $\pm 2$  s.d. of the average spike phase without EPSG for the correspondingly colored PRCs.



### Burst-based PRCs.

**(a)** Sample of current-clamp recordings showing the response of a CA3 neuron (gray trace) to a burst of three EPSPs (arrows: times of stimulation) at 200 Hz during 5 Hz oscillation (black trace). The vertical dashed line represents the average phase at which spikes occurred. As with single EPSP, burst of EPSPs could delay and advance the spike phase following stimulation.

**(b)** Smoothed PRCs (black traces) and raw data points (dots) obtained from CA3 neurons with bursts of EPSPs ( $n = 6$ ). PRCs were smoothed and normalized as described in **Figure 3d-e**. Phase of EPSP in the PRC represents the phase of the first EPSP in a burst. All PRCs show prominent phase advancement and delay of spikes.

**(c)** One of the PRCs from figure **b** (black trace and circles) superimposed onto a PRC with single EPSP (gray trace and circles). Both PRCs were obtained with same EPSP amplitude stimulation (1.5 nS). Black and gray horizontal dashed lines show  $\pm 2$  s.d. of the average spike phase without EPSP for the correspondingly colored PRCs.

## Supplementary Methods

### Detailed experimental procedures

**Slice preparation.** Horizontal hippocampal slices (350  $\mu\text{m}$ ) were prepared from young Wistar rats (postnatal day 13-19) of both sexes after decapitation under deep isoflurane-induced anesthesia, in accordance with British Home Office regulations. Slices were maintained at room temperature in a submerged-style holding chamber with artificial cerebrospinal fluid (ACSF) containing (mM): NaCl 126; KCl 3;  $\text{NaH}_2\text{PO}_4$  1.25;  $\text{MgSO}_4$  2;  $\text{CaCl}_2$  2;  $\text{NaHCO}_3$  25; glucose 10; pH 7.2 – 7.4; bubbled with carbogen gas (95%  $\text{O}_2$ , 5%  $\text{CO}_2$ ) and transferred one by one to the recording chamber.

**Recording conditions.** Patch-clamp recordings of CA3 pyramidal neurons in hippocampal slices were made under visual guidance by infrared differential interference contrast video microscopy<sup>1</sup>. Patch pipettes were pulled from standard-walled borosilicate tubing. The electrode solution contained (mM): Potassium gluconate 110; Hepes 40; NaCl 4; ATP – Mg 4; GTP 0.3 (pH 7.2 – 7.3; osmolarity 280 – 300 mosmol/l). 1  $\mu\text{M}$  gabazine (SR95531) was added to the extracellular solution in experiments where excitatory postsynaptic potentials (EPSPs) were evoked by extracellular stimulation. Whole-cell current-clamp recordings were made with an Axoclamp-2B amplifier in bridge mode. Capacitance compensation was maximal and bridge balance adjusted (15 – 50  $\text{M}\Omega$ ) during recording. CA3 pyramidal cells were identified by their location, shape and orientation as seen by video microscopy, and by their characteristic responses to step current pulses. All recordings were made at temperatures 29 – 31  $^\circ\text{C}$ . Igor Pro software (WaveMetrics, Lake Oswego, OR, USA) was used to generate command signals, and to acquire data online, and subsequently to analyze it.

**Recording protocols and dynamic clamp.** In order to simulate theta oscillation in the neuron, a sinusoidal inhibitory oscillatory conductance of 1 – 2 nS peak amplitude at 5 Hz (**Fig. 3** and **4**, **Supplementary Fig. 4**) or 8 Hz (**Supplementary Fig. 3**) was injected using dynamic clamp. A positive tonic current was superimposed on the oscillatory input so that the membrane potential was depolarized just enough to reliably evoke one action potential near the positive peak of each cycle of the oscillation. EPSPs were evoked with extracellular stimulation (50  $\mu\text{s}$ , 5 V) using a monopolar stimulation electrode placed in the *stratum oriens*, within 100  $\mu\text{m}$  from the neuron being recorded from. In order to generate artificial EPSP, dynamic clamp-simulated excitatory postsynaptic conductance (EPSPG) of 0.5 – 4.5 nS peak amplitude was injected through the

patch pipette. The shape of EPSPG was modeled using an alpha function<sup>2</sup> with time to peak at 3.85 ms. Both EPSPs and EPSPGs were elicited at 20 different phases of the oscillatory inhibitory conductance and repeated 10 times for each EPSP (EPSPG) phase, and the resulting responses averaged. Dynamic clamp was implemented using ITC-18 A-D board (Instrutech, Port Washington, NY) and custom made macros written in Igor Pro software. The current was calculated as:

$$I(t) = g(t) \cdot (V_m(t) - E_{\text{rev}})$$

where  $I$  is the current to be injected,  $g$  is the conductance waveform,  $V_m$  is the membrane potential, and  $E_{\text{rev}}$  is the reversal potential, which was 0 mV for EPSPG and  $-70$  mV for inhibitory conductance.

## References

1. Sakmann, B. & Stuart, G. Patch-pipette recordings from the soma, dendrites, and axon of neurons in brain slices. in *Single-Channel Recording* 2<sup>nd</sup> edn. (eds. Sakmann, B. & Neher, E.) 199–211 (Plenum Press, New York, NY, 1995).
2. Koch, C. & Segev, I. *Methods in Neuronal Modeling: From Ions to Networks* (MIT Press, Cambridge, MA, 1998).