Supporting Information

Martínez-García et al. 10.1073/pnas.1108137108

SI Text

Information Analysis on Medial Premotor Cortex Neurons During Vibrotactile Decision-Making with a Postponed Response. ***Single-cell information analysis.*** We are interested in knowing the information carried by neuronal responses, especially how the firing rate is related to the stimulus or to the category of the behavioral response. To estimate the information content carried by the neuron's firing rate, we performed mutual information (MI) analysis (1, 2), which quantified the average amount of common information contained in the variables R and S—in our case, firing rate and category of behavioral response. In other words, the MI reflects the uncertainty removed from one by knowing the other. The MI was calculated as

$$I(S,R) = \sum_{s,r} \mathrm{P}(s,r) \log_2 \frac{\mathrm{P}(s,r)}{\mathrm{P}(s)\mathrm{P}(r)}, \qquad [1]$$

where $P(s)$, $P(r)$ are the marginal distributions of the variables and $P(s, r)$ is the joint distribution. A zero-valued MI means that R and S are statistically independent. The MI is always ≥ 0. In the data, the calculation of firing rate was based on the mean number of spikes across trials, obtaining for each bin an n-dimensional vector of spike counts, where n is the number of frequency pairs. One frequency pair is the set of [f1:f2] stimuli presented on one specific trial. This computation was performed in sliding windows of 200 ms in steps of 100 ms for each pair of frequencies. The variable category (the behavioral response) is represented by ones with the sign of the difference between a specific frequency pair [f1:f2], i.e., $+1$ for $f_1 > f_2$, and -1 for $f_1 < f_2$. We only included "correct" trials in the analysis, i.e., only the trials on which the monkey solved the task correctly. Because the MI estimate is subject to statistical errors that can lead to an overestimate of the information, we corrected the information estimates using a first-order Monte Carlo method. In this correction procedure the mean information from many runs in which the stimuli and the responses are shuffled across trials is subtracted from the information estimate from the original unshuffled data (2, 3). We applied this Monte Carlo method basing the correction on 50 shuffled runs. This method also leads to a test of the statistical significance of the corrected MI between firing rates and category (4).

The same method was used in the simulations to measure the mutual information between the neuronal firing in different 200-ms epochs through a trial, and the decision that had been taken by the network represented by which pool had the high firing rate in the decision period when the decision cues were applied (t = 3.5–4.0 s).

We next show how we corrected for multiple comparisons in the single-cell information analysis using a Holm–Bonferroni multiple-test correction procedure (5). First, we performed a statistical significance test of the single-neuron MI on 864 neurons from the MPC (medial premotor cortex), 323 from the pre-SMA (presupplementary motor cortex), and 252 from M1 (the primary motor cortex) (6) as follows. To calculate a P value (for the null hypothesis of no information) for each single cell at a specific time window (in a 200-ms sliding window with 100-ms step), we applied a surrogate-based statistical test. The null hypothesis implemented by the surrogates corresponds to the absence of statistical dependencies between the firing rate of the neuron in a particular window and the behavioral response of the animal. To do this, we generated 200 surrogates by randomly reordering the responses and the firing rates on different trials for each cell. The P value is calculated by comparing the estimated MI value of each single cell at a specific time window with the empirical distribution of the MI of the corresponding surrogates. The Holm–Bonferroni procedure considers the fact that we have many cells and therefore multiple null hypotheses to test in a given time window. Let us assume that the significance α level is 0.05. The Holm–Bonferroni procedure consists of ordering the P values and comparing the smallest P value to α/k. The first null hypothesis is rejected if the P value is less than α/k. After this, one tests the remaining $k-1$ null hypotheses by starting again with the same α, i.e., reordering the $k-1$ remaining P values and comparing the smallest one to $\alpha/(k-1)$. This iteration is continued until the null hypothesis with the smallest P value cannot be rejected. The result of the Holm–Bonferroni correction is accepting all null hypotheses that have not been rejected at previous steps.

We are interested in finding the percentage of neurons that maintain information about the behavioral response during the postdecision delay relative to those neurons that show significant information during the decision period of f_2 presentation and also during the final behavioral response period. Therefore, we selected the neurons with an MI > 0.26 bits during these periods: from f_2 to 50 ms later and at the response time (end of the postdecisional delay period) in at least three time windows. In that way we obtained 180 neurons (18.2%) from MPC; 30 neurons (9.0%) from pre-SMA; and 34 neurons (11.1%) from M1. The results after the Holm–Bonferroni correction are shown for each of the 200-ms time windows in Fig. S1. Table S1 shows that in the MPC between 21 and two (depending on the time window) of 180 MPC neurons are able to maintain the (single-cell) information in the delay period.

Multiple-cell information analysis. When measuring the information about a set of stimuli S from the responses of many neurons, the response space becomes very large, as there are responses from every neuron to every stimulus. It becomes difficult to record a sufficiently large number of trials to sample this high dimensional space adequately. Rolls et al. (7) introduced a decoding procedure in which the stimulus s' (from the set S) shown on each trial is predicted from the neuronal responses. It is then possible in the low dimensional space of the number of stimuli in the set to compute the mutual information between actual stimuli s shown on a trial and the predicted stimuli s' based on the neuronal responses of the population of neurons. The mutual information between the stimuli and the predicted stimuli is then calculated as follows. Bayesian probability decoding using cross-validation was used to generate a table of conjoint probabilities P (s, s'). s' represents all possible stimuli, and hence belong to the same set S as each stimulus s. The mutual information value based on this probability decoding (I_p) was calculated as

$$<I_p> = \sum_{s\in S} \sum_{s'\in S} \mathrm{P}(s,s') \log_2 \frac{\mathrm{P}(s,s')}{\mathrm{P}(s)\mathrm{P}(s')}. \qquad [2]$$

A correction procedure for the sampling bias was used. The percentage correct was calculated using maximum-likelihood decoding in which the most likely stimulus that was shown on each trial is predicted from the neuronal response of all of the neurons on that trial. Examples and fuller descriptions of the use of these procedures are available (2, 7–13).

We performed multiple-cell information analyses to measure how the information about the decision increases with the number of MPC neurons (6) in the sample. We found, taking 18

neurons at random from those with low single-cell information content in a 600-ms window in the delay period (5.2–5.8 s), that the average information per neuron was 0.06 bits, and that with 18 neurons the information provided was 0.51 bits, and 90% correct prediction of the decision (Fig. S3). (These 18 neurons had low information even during f_2, on average 0.4 bits/neuron, and it needed 14 such neurons selected at random to reach 1 bit of information during f_2.) A further analysis showed that just seven such neurons provided 0.42 bits of information and 90% correct or better performance throughout the delay period.

If we consider 16 randomly selected neurons from those with the higher information values shown in Fig. 2, then the multiple-cell information analysis showed that the average amount of information for each cell was 0.56 bits in the same 600-ms window in the delay period, and that with subsets of cells chosen at random from the 16 cells the information reached 1 bit and 100% correct with four to six cells. The implication is that with just six of the MPC cells chosen at random from the set with higher information values in the delay period shown in Fig. 2, the animal could do the task perfectly, 100% correctly.

Model of the Postponed Response Task Using Synaptic Facilitation.

Implementation of the model. The decision-making network is illustrated in Fig. 1*C*, and operates according to the principles described elsewhere (14), with the addition of synaptic facilitation, as specified in the tabular material that follows. The model is biologically realistic and based on an attractor network (15). The network has four neuronal populations or pools: one inhibitory pool (with $N_I = 200$ neurons) and three excitatory pools or populations (with the total number of excitatory neurons $N_E = 800$), of which one pool is nonselective and the other two are selective and specific for each decision. The selective pools are involved in a competition mediated by inhibition (inhibitory interneurons), in which only one pool wins (a winner-take-all model). The nonspecific group is connected to the selective pools; likewise, all three excitatory pools are connected to the inhibitory pool. All of the pools are self-connected (recurrent connections). We used integrate-and-fire neurons with three types of receptors mediating the synaptic currents: the excitatory recurrent postsynaptic currents are mediated by AMPA (fast) and NMDA (slow) receptors, and the inhibitory postsynaptic currents to both excitatory and inhibitory neurons are mediated by GABA receptors (see refs. 2, 14, and 16 for more details). The external excitatory recurrent postsynaptic currents injected onto the network for λ_1, λ_2, and λ_{unsp} are driven only by AMPA receptors. The parameters used are shown in the following section.

In the simulations, first the network runs with a background external input of 3 Hz to each of the 800 synapses for external inputs onto every neuron, which remains on throughout the simulation. Then in a decision period corresponding to f_2 for t = 3.5–4.0 s, each selective pool is driven by a different input, λ_1 and λ_2 respectively. This time symbolizes the first part of the vibrotactile decision-making task, i.e., from the beginning of f_1 to the end of f_2 (Fig. 1). The network responds to the external inputs (λ_1 and λ_2 applied in the decision period) by starting the competition between the two selective (decision) pools, the firing rates of which grow apart during this period as one pool moves to a high firing rate attractor level. The pool that reaches the high firing attractor reflects the decision that has been made, and the other selective pool remains firing at around the spontaneous firing rate level. As a result of the calcium-mediated SF, the residual calcium levels of the neurons in the winning selective pool have grown in this decision period, and the probability of spiking has increased (16), as illustrated in Fig. S4 *Left*. After this 0.5-s decision-making period, the decision cues are removed and the postponed response delay short-term memory period lasts from 4 to 7 s. In this delay period, because the decision cues have been removed, the firing rates of the two selective pools drop to a spontaneous level of activity (as shown by the rastergram in Fig. 3 and by the firing rate in the delay period shown in Fig. 4 *Lower Left*). Note, however, that the information in the delay period is still present in the network at the synaptic level but not in the firing rates, as reflected in the information analysis shown in Fig. 2 for the subsequent recall period.

Finally, at t = 7.0–7.5 s, both selective pools receive the same extra nonspecific external input (λ_{unsp}), to reflect the moment when the subject receives the stimulus to give its response. The extra nonspecific external rate was 0.255 Hz onto each of the 800 external input synapses of each neuron in a selective pool. (With a set of N_{ext} = 800 external synapses for external inputs, this results in an additional Poisson background external input of 204 Hz to each neuron in both selective pools.) When the external nonspecific input (λ_{unsp}) is injected into the selective pools the report period starts and, because of the altered synaptic calcium levels, the firing rate of one of the selective pools increases to the attractor activity level (Fig. 3), as does its information about the response to be made (Fig. 4), whereas the firing of the other selective pool remains with low activity, although a little higher than the spontaneous firing rate. The outcome of the competition in the postponed response period (t = 7.0–7.5 s in the simulations) is the report of the task.

Analyzing the computational activity of the SF model by the MI method (Fig. 2), we can see how the information becomes significant during the discrimination period, because the MI decodes the network responses to the injected inputs (λ_1 and λ_2 to pool 1 and pool 2, respectively), one higher than the other. (In our simulations, $\lambda_1 > \lambda_2$.) During the postponed response delay period, the MI in the firing rates about the decision becomes less significant as a result of the reset of the external input, with the variability of the firing rates as well as the low and non-differential firing rates contributing to the low information at this time. Finally, the information available becomes high again during the report time owing to the λ_{unsp} injected, which acts to increase the rates, but selectively in the pool with the synaptic facilitation, as shown in Figs. 2 and 3. That is, in the SF model the rates are encoding the response only during two specific periods: after f_2, when they already have all of the sensory information, and when the response is demanded. With this criterion we confirm that the SF network remembers the discrimination and it is able to reproduce it when it is requested at the time of the postponed response without having to store the information about f_1, f_2 or the decision in short-term memory using high firing rates in an attractor. The simulation results are shown in Figs. 2–4.

In the simulations we consider a trial as correct if the activity of the more stimulated selective pool (with λ_1 = 250 Hz and λ_2= 150 Hz) is higher than the activity of the other selective pool, during all of the bins of the 100-ms period centered at the end of the λ_{unsp} period.

A	Model summary
Populations	Two: excitatory and inhibitory
Topology	—
Connectivity	Full connected
Neuron model	Leaky integrate-and-fire, fixed threshold, fixed refractory period, MNDA
Channel models	—
Synapse model	Instantaneous jump and exponential decay for AMPA and GABA and exponential jump and decay for NMDA receptors

A	Model summary
Plasticity	Synaptic facilitation
Input	Independent fixed-rate Poisson spike trains to each selective population
Measurements	Spike activity

B		Populations
Total number of neurons	n = 1,000	Neurons in each selective pool $N_{selective} = N_E$ *sparseness*
Excitatory neurons	$N_E = 0.8 \cdot N$	
Inhibitory neurons	$N_I = 0.2 \cdot N$	

C1	Neuron and synapse model
Type	Leaky integrate-and-fire, conductance-based synapses
Subthreshold dynamics	$C_m \frac{dV(t)}{dt} = -g_m(V(t)-V_L) - I_{syn}(t)$ $I_{syn}(t) = I_{AMPA,\ ext}(t) + I_{AMPA,\ rec}(t) + I_{NMDA}(t) + I_{GABA}(t)$
Spiking	If $V(t) > V_\theta \wedge t > t^* + \tau_{rp}$ 1. set $t^* = t$ 2. Emit spike with time stamp t^* 3. $V(t) = V_{reset}$
Synaptic currents	$I_{AMPA,ext}(t) = g_{AMPA,ext}(V(t)-V_E)\sum_{j=1}^{N_{ext}} s_j^{AMPA,ext}(t)$ $I_{AMPA,rec}(t) = g_{AMPA,rec}(V(t)-V_E)\sum_{j=1}^{N_E} w_j s_j^{AMPA,rec}(t) u_j(t)$ $I_{NMDA}(t) = \frac{g_{NMDA}(V(t)-V_E)}{1+\gamma \exp(-\beta V(t))}\sum_{j=1}^{N_E} w_j s_j^{NMDA}(t) u_j(t)$ $I_{GABA}(t) = g_{GABA}(V(t)-V_I)\sum_{j=1}^{N_I} s_j^{GABA}(t)$

C2	Neuron and synapse model
Fraction of open channels	$\frac{ds_j^{AMPA,ext}(t)}{dt} = -s_j^{AMPA,ext}(t)/\tau_{AMPA} + \sum_k \delta(t-t_j^k-\delta)$ $\frac{ds_j^{AMPA,rec}(t)}{dt} = -s_j^{AMPA,rec}(t)/\tau_{AMPA} + \sum_k \delta(t-t_j^k)$ $\frac{ds_j^{NMDA}(t)}{dt} = -s_j^{NMDA}(t)/\tau_{NMDA,decay} + \alpha x_j(t)(1-s_j^{NMDA}(t))$ $\frac{dx_j(t)}{dt} = -x_j(t)/\tau_{NMDA,rise} + \sum_k \delta(t-t_j^k-\delta)$ $\frac{ds_j^{GABA}(t)}{dt} = -s_j^{GABA}(t)/\tau_{GABA} + \sum_k \delta(t-t_j^k-\delta)$
Synaptic facilitation	$\frac{u_j(t)}{dt} = \frac{U-u_j(t)}{\tau_F} + U(1-u_j(t))\sum_k \delta(t-t_j^k)$

D	Input
Type	Description
Poisson generators	Fixed-rate N_{ext} synapses per neuron, with each synapse driven by a Poisson process

E	Measurements
Spike activity	

Tabular description of the network following the prescription of Nordlie et al. (17).

Performance in the task by the subjects, and by the SF model, as a function of the postponed response delay period magnitude. We tested the performance of the model as a function of the duration of the delay period, because synaptic facilitation is unlikely to be able to maintain a memory trace for long time periods, and it is a prediction of the model that performance will decay to zero as the short-term memory period increases much beyond the time constant of the synaptic facilitation, 2 s. We performed four simulations with different postponed response delays. In Table S3 we compare the recorded experimental data (6) with the performance of the SF model. Our model fits the data well. Analyzing the MI of the different postponed decision delays, we found that the maximum MI value at the end of the decision period in the network is 0.967 ± 0.001 bits, and it does not depend on the delay period, whereas the maximum MI value at the end of the nonspecific stimulation is not always the same, but depends on the delay period. The longer the delay period, the lower is the MI value: 0.95, 0.91, 0.63, and 0.37 bits for 1-, 1.5-, 2-, and 3-s postponed response delays, respectively.

Magnitude of the synaptic facilitation u in the four pools as a function of time for the synaptic facilitation model, and a mean-field analysis. For the SF model, the time evolution of u, the synaptic utilization, for the four different pools is shown in Fig. S4 *Left*.

The results of a mean field analysis (18, 19) for two different scenarios for the synaptic facilitation model are shown, one with synaptic facilitation (Fig. S4, *Lower Right*) and one without synaptic facilitation (Fig. S4, *Upper Right*). We plotted the value for the firing rate difference between pools 1 and 2 for a fixed value of λ_{unsp}, $w+$ and U, for two values of U (one for each scenario, 0.15 and 0, respectively) to check that the synaptic facilitation network cannot recover the information after a delay period without synaptic facilitation. To compare both ranges of parameters λ_{unsp}, $w+$, we performed the mean field analysis by multiplying the specific connectivity weight by the two specific sets of U, one for each scenario. To obtain the set that corresponds to a nonfacilitated network, we ran the network without any stimulation, and we took the U values after the network was stable. To obtain the set that corresponds to a facilitated network, we took the U values of the simulation at the time value of 7.5 s, and show the results in Fig. S4. We show that there is selective firing at the time of recall in this model with these parameters only when SF is present.

Model of the Activity in the Delay Period Using Graded Firing Rates in an Attractor Network. The attractor network used was similar to that described for the SF network, except that no synaptic facilitation was used, and the synaptic weights in the intrapool connections for the selective pools 1 and 2 were set to an exponential-like distribution to produce an exponential-like firing rate distribution, as described below and in more detail by Webb et al. (20).

Graded weight patterns. In an attractor network, the synaptic weights of the recurrent connections are set by an associative (or Hebbian) synaptic modification rule with the form

$$\delta w_{ij} = \alpha r_i r_j, \qquad [3]$$

where δw_{ij} is the change of synaptic weight from presynaptic neuron j onto postsynaptic neuron i, α is a learning rate constant, r_j is the presynaptic firing rate, and r_i is the postsynaptic firing rate when a pattern is being trained (2, 21, 22). To achieve this for the firing rate distributions investigated, we imposed binary and graded firing rates on the network by selecting the distribution of the recurrent synaptic weights in each of the two decision pools. To achieve a binary firing pattern, used for the SF simulations described, all of the synaptic weights between the excitatory neurons within a decision pool were set uniformly to the same value w_+.

Graded firing patterns were achieved by setting the synaptic weights of the recurrent connections within each of the decision pools to be in the form of a discrete exponential-like firing rate

(r) distribution generated using methods taken from Rolls et al. (23):

$$P(r) = \begin{cases} \frac{4}{3}a\lambda e^{-2(r+r_0)} & \text{for } r > 0 \\ 1 - \sum_{r_i \in r : i > 0} \frac{4}{3}a\lambda e^{-2(r_i+r_0)} & \text{for } r = 0 \end{cases}, \quad [4]$$

where a is the sparseness of the pattern defined in Eq. 5, and r_0 is the firing rate of the lowest discretized level. The population sparseness a of a binary representation is the proportion of neurons active to represent any one stimulus or decision in the set. The sparseness can be generalized to graded representations, as shown in Eq. 5:

$$a = \frac{\left(\sum_i^{N_E} r_i\right)^2}{\sum_i^{N_E} r_i^2}, \quad [5]$$

where r_i is the firing rate measured for neuron i in the population of N_E excitatory neurons in the network (2, 23–26). We note that this is the sparseness of the representation measured for any one stimulus over the population of neurons (2, 26).

In the graded firing rate simulations, we use $a = 0.1$ to correspond to the fraction of excitatory neurons that are in a single decision pool. We chose 50 equal-spaced discretized levels to evaluate the distribution $(0, \frac{1}{3} - r_0, \frac{2}{3} - r_0, \ldots, 3 - r_0)$. r_0 and λ are chosen so that first and second moments of the firing rate distribution are equal to the sparseness, i.e., $< r > = < r^2 > = a$. A weight matrix $W = \{w_{1,1}, \ldots, w_{1,fN_E}, w_{2,1}, \ldots, w_{fN_E,fN_E}\}$ was constructed by first sampling a firing rate for each neuron, r_i, using Eq. 4 and then setting w_{ij} based on the desired firing rates of each pair of neurons, as described in more detail by Hopfield (20). The mean weight was set to a value close to 2.1, the maximum weight was ~5.0, and the minimal weight was 1.0. The final mean weight used for the simulations was 2.04, as this provided for satisfactory stability of the network in the spontaneous period, because stability is reduced by graded compared with binary firing rates (20).

Graded firing rate simulation protocol. Our focus was on the activity in the delay period after the decision had been taken, and on whether a low level of firing in the delay period could be restored to a high level, with a high information content, at the end of the delay period when an external input was applied equally to the two decision pools, 1 and 2.

In a 0.5-s period of spontaneous firing from 3.0 to 3.5 s, the external rates λ_1 and λ_2 were 3.00 Hz applied to each of the 800 external synapses on each neuron. In the decision period from 3.5 to 4.0 s, λ_1 was 3.10 Hz and λ_2 was 2.98 Hz per external synapse, values that produced a decision on almost all trials of pool 1 winning. In the delay period from 4 to 7 s, λ_1 and λ_2 were set to the lowest value that enabled firing to be maintained reliably (although at a low level) by some neurons, 2.95 Hz per synapse. During the recall period from 7.0 to 7.5 s, λ_1 and λ_2 were set to the identical value of 3.05 Hz per synapse.

1. Cover TM, Thomas JA (1991) *Elements of Information Theory* (Wiley, New York).
2. Rolls ET (2008) *Memory, Attention, and Decision-Making. A Unifying Computational Neuroscience Approach* (Oxford Univ Press, Oxford).
3. Tovée MJ, Rolls ET, Treves A, Bellis RP (1993) Information encoding and the responses of single neurons in the primate temporal visual cortex. *J Neurophysiol* 70:640–654.
4. Schreiber T, Schmitz A (2000) Surrogate time series. *Physica D* 142:346–382.
5. Holm S (1979) A simple sequentially rejective multiple test procedure. *Scand J Stat* 6: 65–70.
6. Lemus L, et al. (2007) Neural correlates of a postponed decision report. *Proc Natl Acad Sci USA* 104:17174–17179.
7. Rolls ET, Treves A, Tovee MJ (1997) The representational capacity of the distributed encoding of information provided by populations of neurons in primate temporal visual cortex. *Exp Brain Res* 114:149–162.
8. Booth MCA, Rolls ET (1998) View-invariant representations of familiar objects by neurons in the inferior temporal visual cortex. *Cereb Cortex* 8:510–523.
9. Rolls ET, Treves A, Robertson RG, Georges-François P, Panzeri S (1998) Information about spatial view in an ensemble of primate hippocampal cells. *J Neurophysiol* 79: 1797–1813.
10. Rolls ET, Aggelopoulos NC, Franco L, Treves A (2004) Information encoding in the inferior temporal visual cortex: Contributions of the firing rates and the correlations between the firing of neurons. *Biol Cybern* 90:19–32.
11. Franco L, Rolls ET, Aggelopoulos NC, Treves A (2004) The use of decoding to analyze the contribution to the information of the correlations between the firing of simultaneously recorded neurons. *Exp Brain Res* 155:370–384.
12. Rolls ET, Franco L, Aggelopoulos NC, Jerez JM (2006) Information in the first spike, the order of spikes, and the number of spikes provided by neurons in the inferior temporal visual cortex. *Vision Res* 46:4193–4205.
13. Ince RA, et al. (2010) Information-theoretic methods for studying population codes. *Neural Netw* 23:713–727.
14. Wang XJ (2002) Probabilistic decision making by slow reverberation in cortical circuits. *Neuron* 36:955–968.
15. Rolls ET, Deco G (2010) *The Noisy Brain: Stochastic Dynamics as a Principle of Brain Function* (Oxford Univ Press, Oxford).
16. Deco G, Rolls ET, Romo R (2010) Synaptic dynamics and decision making. *Proc Natl Acad Sci USA* 107:7545–7549.
17. Nordlie E, Gewaltig MO, Plesser HE (2009) Towards reproducible descriptions of neuronal network models. *PLoS Comput Biol* 5:e1000456.
18. Brunel N (2000) Dynamics of sparsely connected networks of excitatory and inhibitory spiking neurons. *J Comput Neurosci* 8:183–208.
19. Brunel N, Wang XJ (2001) Effects of neuromodulation in a cortical network model of object working memory dominated by recurrent inhibition. *J Comput Neurosci* 11: 63–85.
20. Webb TJ, Rolls ET, Deco G (2011) Noise in attractor networks in the brain produced by graded firing rate representations. *PLoS Comput Biol*, in press.
21. Hopfield JJ (1982) Neural networks and physical systems with emergent collective computational abilities. *Proc Natl Acad Sci USA* 79:2554–2558.
22. Hertz J, Krogh A, Palmer RG (1991) *Introduction to the Theory of Neural Computation* (Addison Wesley, Wokingham, U.K.).
23. Rolls ET, Treves A, Foster D, Perez-Vicente C (1997) Simulation studies of the CA3 hippocampal subfield modelled as an attractor neural network. *Neural Netw* 10: 1559–1569.
24. Rolls ET, Treves A (1990) The relative advantages of sparse versus distributed encoding for associative neuronal networks in the brain. *Network* 1:407–421.
25. Treves A, Rolls ET (1991) What determines the capacity of autoassociative memories in the brain? *Network* 2:371–397.
26. Franco L, Rolls ET, Aggelopoulos NC, Jerez JM (2007) Neuronal selectivity, population sparseness, and ergodicity in the inferior temporal visual cortex. *Biol Cybern* 96: 547–560.

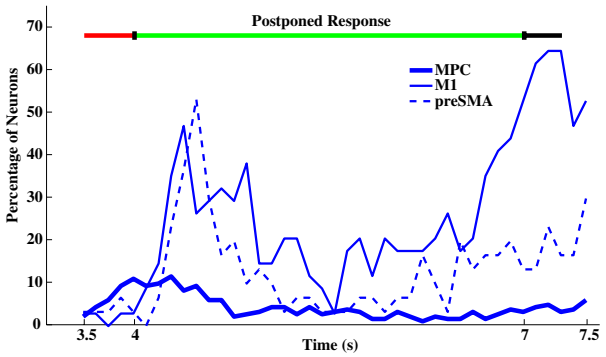

Fig. S1. Results of the Holm–Bonferroni correction for multiple tests for the single-cell information measure in the following areas: MPC, M1, and pre-SMA. The *y* axis represents the percentage of neurons for each 200-ms time window that have significant information after the correction has been applied during the postponed response delay period from 4 to 7 s. Time course of the task is shown above (Fig. 1).

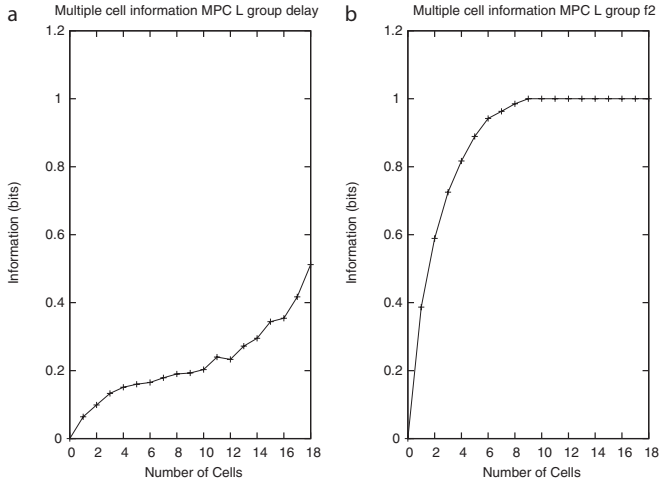

Fig. S2. Multiple-cell information for the group of MPC neurons with low single-cell information values. (*A*) The values for the average information available in the responses of different numbers of MPC neurons on each trial taken in a 600-ms period toward the end of the delay period about which decision had been made. (*B*) The values for the average information available in the responses of different numbers of MPC neurons on each trial taken in a 600-ms period during f_2 about which decision had been made. The decoding method was Bayesian probability estimation.

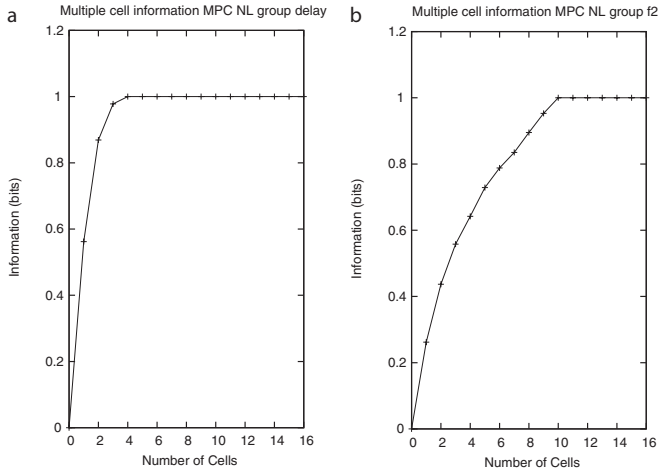

Fig. S3. Multiple-cell information for the group of MPC neurons with high single-cell information values. (*A*) The values for the average information available in the responses of different numbers of MPC neurons on each trial taken in a 600-ms period toward the end of the delay period about which decision had been made. (*B*) The values for the average information available in the responses of different numbers of MPC neurons on each trial taken in a 600-ms period during f_2 about which decision had been made. The decoding method was Bayesian probability estimation.

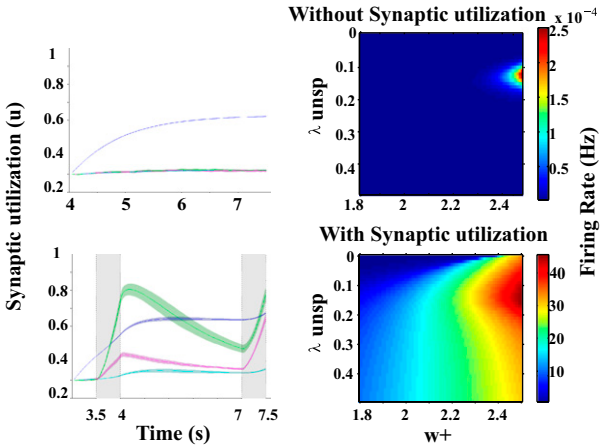

Fig. S4. (*Left*) The time evolution of the variable *u*, the synaptic utilization, for the four different pools. (*Upper Left*) The network without stimulation. (*Lower Left*) the network with stimulation. The shaded gray rectangles show the period of stimulation; blue, the inhibitory pool; green, selective pool 1; pink, the selective pool 2; cyan, the nonselective pool. The plots are an average of >50 trials. The dark lines are the mean, and the colored shadows are 1 SD. (*Right*) Results of the mean field analysis (19, 20). Each point represents the firing rate difference between pool 1 and pool 2 for a fixed value of (λ_{unsp}, w_+) and *U*. We used the values of the mean synaptic facilitation (one for each pool) at the time instant of 7,500 ms extracted from the simulation shown in the above plots.

Table S1. The proportions of neurons in different areas that retain single cell information in the postponed response delay period when tested with a Holm–Bonferroni correction for multiple comparisons

Area	Responsive	Significant MI in f_2 and recall period	Holm–Bonferroni multiple test	
			k maximum	*k* minimum
MPC	867	180 (18.2%)	21 (11.7%)	2 (1.1%)
pre-SMA	323	30 (9.0%)	5 (53.3%)	1 (3.3%)
M1	252	34 (11.1%)	18 (52.9%)	1 (2.9%)

Column 2 shows the number of neurons responsive in the task. Column 3 shows the number and proportion of neurons that show significant information in both f_2 and in the response period. Columns 4 and 5 show the maximum and the minimum across each of the 200-ms time windows in the delay period of the number of neurons *k* with significant information after the Holm–Bonferroni test has been applied. The number and proportion of neurons in an area is shown (see Fig. S1 for more details). f_2, second stimulus; MPC, medial premotor cortex; pre-SMA, presupplementary motor area; M1, primary motor cortex.

Table S2. Parameters used in the integrate-and-fire simulations, and the network connection parameters

Parameters			
Integrate and fire			
C_m (excitatory)	0.5 nf	C_m (inhibitory)	0.2 nf
g_m (excitatory)	25 ns	g_m (inhibitory)	20 ns
V_L	−70 mV	V_{thr}	−50 mV
V_{reset}	−55 mV	V_E	0 mV
V_I	−70 mV		
$g_{AMPA,\ ext}$ (excitatory)	2.08 ns	$g_{AMPA,\ rec}$ (excitatory)	0.104 ns
g_{NMDA} (excitatory)	0.327 ns	g_{GABA} (excitatory)	1.25 ns
$g_{AMPA,\ ext}$ (inhibitory)	1.62 ns	$g_{AMPA,\ rec}$ (inhibitory)	0.081 ns
g_{NMDA} (inhibitory)	0.258 ns	g_{GABA} (inhibitory)	0.973 ns
$\tau_{NMDA,\ decay}$	100 ms	$\tau_{NMDA,\ rise}$	2 ms
τ_{AMPA}	2 ms	τ_{GABA}	10 ms
τ_{rp} (excitatory)	2 ms	τ_{rp}(inhibitory)	1 ms
α	0.5 ms^{-1}	γ	$[Mg^{2+}]/(3.57\ mM) = 0.280$
β	0.062 mV^{-1}	Sparseness	0.10
N_{ext}	800		
U	0.15	τ_F	2,000 ms
Connection			
w_+	2.17	w_i	0.97

Table S3. Performance of the task (6) measured in percent correct vs. performance of the SF network with different postponed decision delays: 1, 1.5, 2.5, and 3 s

	Postponed response delay, %			
	1 s	1.5 s	2.5 s	3 s
Task performance	100	98.6	92.6	85.5
Model performance	100	99	92	83