

2006 Special Issue

Homeostatic synaptic scaling in self-organizing maps

Thomas J. Sullivan^{a,*}, Virginia R. de Sa^b

^a Department of Electrical Engineering, UC San Diego, 9500 Gilman Dr. MC 0515, 92093 La Jolla, CA, United States

^b Department of Cognitive Science, UC San Diego, La Jolla, CA, United States

Abstract

Various forms of the self-organizing map (SOM) have been proposed as models of cortical development [Choe Y., Miikkulainen R., (2004). Contour integration and segmentation with self-organized lateral connections. *Biological Cybernetics*, 90, 75–88; Kohonen T., (2001). *Self-organizing maps* (3rd ed.). Springer; Sirosh J., Miikkulainen R., (1997). Topographic receptive fields and patterned lateral interaction in a self-organizing model of the primary visual cortex. *Neural Computation*, 9(3), 577–594]. Typically, these models use weight normalization to contain the weight growth associated with Hebbian learning. A more plausible mechanism for controlling the Hebbian process has recently emerged. Turrigiano and Nelson [Turrigiano G.G., Nelson S.B., (2004). Homeostatic plasticity in the developing nervous system. *Nature Reviews Neuroscience*, 5, 97–107] have shown that neurons in the cortex actively maintain an average firing rate by scaling their incoming weights. In this work, it is shown that this type of homeostatic synaptic scaling can replace the common, but unsupported, standard weight normalization. Organized maps still form and the output neurons are able to maintain an unsaturated firing rate, even in the face of large-scale cell proliferation or die-off. In addition, it is shown that in some cases synaptic scaling leads to networks that more accurately reflect the probability distribution of the input data.

© 2006 Elsevier Ltd. All rights reserved.

Keywords: Self-organizing map; Homeostasis; Weight normalization; Synaptic scaling

1. Introduction

The self-organizing map (SOM), in its various forms, has been a useful model of cortical development (Choe & Miikkulainen, 2004; Kohonen, 2001; Obermayer, Blasdel, & Schulten, 1992; Sirosh & Miikkulainen, 1997). Sirosh and Miikkulainen (1997) showed the simultaneous development of receptive field properties and lateral interactions in a realistic model. The usefulness of the developed lateral connections was shown by Choe and Miikkulainen (2004) for contour integration and segmentation. It is this lateral connectivity that ensures the neighboring neurons come to respond to similar stimuli and form a good map.

In these models, Hebbian learning is used to strengthen associations between stimuli and winning neurons. This type of associative learning has been well documented in the experimental literature (Bi & Poo, 2001; Bliss & Lomo,

1973), but our understanding has remained incomplete. It is well known that the most straightforward implementations of Hebbian learning lead to unconstrained weight growth. To counteract this problem, typical SOM algorithms use weight normalization: after each learning iteration all the weights converging onto a neuron are divided by the sum of the incoming weights (or the square root of the sum of the squared weights). It has been argued that this type of weight normalization is biologically plausible. For example, a neuron might have a finite resource necessary for maintaining incoming synapses. This might keep an upper limit on the total summed size of the incoming synapses. While this sounds helpful in keeping Hebbian learning in check, little evidence from the experimental literature is available for support.

More plausible mechanisms for controlling the Hebbian process based on maintaining an average output firing rate have recently emerged. Two different types of these internal mechanisms have been found. One type controls the intrinsic excitability of the neuron (reviewed by Zhang and Linden (2003)). The molecular causes underlying this mechanism are

* Corresponding author.

E-mail addresses: tom@sullivan.to (T.J. Sullivan), desa@cogsci.ucsd.edu (V.R. de Sa).

still being investigated, but some of the behavior has been documented. In a typical experiment, a neuron would be excited repeatedly at high frequency. Then the output firing rate would be measured when current is injected. The intrinsic excitability (the ratio of firing rate to injected current) is higher after the stimulation. This makes the neuron even more sensitive to its inputs.

Two models have proposed that neurons modify their excitability to maintain a high rate of information transfer. In the first of these models (Stemmler & Koch, 1999), individual neurons change their ion channel conductances in order to match an input distribution. The neurons are able to maintain high information rates in response to a variety of distributions. The second model (Triesch, 2004) proposes that neurons adjust their output nonlinearities to maximize information transfer. There it is assumed that the neuron can keep track of its average firing rate and average variance of firing rate. Given this limited information, it does the best it can by adjusting the slope and offset of an output sigmoid function.

In the second type of internal neuronal mechanism it was shown (Maffei, Nelson, & Turrigiano, 2004; Turrigiano, Leslie, Desai, Rutherford, & Nelson, 1998; Turrigiano & Nelson, 2004) that neurons in the cortex actively maintain an average firing rate by scaling their incoming weights. The mechanism has been examined in cultures and in other experiments using in vivo visual deprivation. It has been shown that the incoming synapses are altered by a multiplicative factor, which presumably preserves the relative strengths of the synapses. The underlying mechanisms are not yet known, but there is ongoing research looking at intracellular chemical factors such as calcium and brain-derived neurotrophic factor (BDNF) (Turrigiano & Nelson, 2004). The levels of these factors are related to firing rates, so integrating them over time could lead to an estimate of average firing rate and produce a chemical signal for synaptic change. Another interesting finding is that a neuron with high average firing rate will decrease the strength of incoming excitatory synapses, but increase the strength of incoming inhibitory neurons (Maffei et al., 2004), thus altering the excitatory/inhibitory balance.

Homeostatic mechanisms have been implemented in two models. In one study, the molecular underpinnings of homeostasis were explored (Yeung, Shouval, Blais, & Cooper, 2004). It was suggested that LTP, LTD, and synaptic homeostatic scaling are all related through intracellular calcium levels. The homeostatic mechanism may influence the level of calcium, thereby changing under what conditions LTP and LTD are induced (since calcium levels play a major role in LTP and LTD). The other model concerns a particular class of associative memories (Chechik, Meilijson, & Ruppin, 2001). It is shown that the storage capacity can be increased if the neuron's weights are controlled with either weight normalization or homeostatic synaptic scaling.

There have also been attempts to change neuron thresholds within neural networks. One network that was described (Horn & Usher, 1989) was a Hopfield network with neurons that were always producing one of two possible outputs (+1 or -1). The neuronal thresholds for determining which state to be in, acting

on the total amount of input activation, was adjusted according to the recent activation. They reported that interesting periodic dynamics emerged. Another adjustable threshold network (Gorchetnikov, 2000) was a winner-take-all associative memory. In that work, the thresholds on output sigmoid functions were adjusted. It was shown that on problems like XOR classification, the network would learn the appropriate mapping faster than without an adjustable threshold. While these approaches are computationally interesting, there is not yet evidence that neurons adjust their threshold based on their average firing rate.

It has previously been suggested that homeostatic synaptic scaling might form a basis for keeping Hebbian learning in check (Miller, 1996; Turrigiano & Nelson, 2004). This possibility is explored here. It may very well be that this mechanism is one of several that constrain synapse strength, but it is examined here in isolation to get a better understanding of its capability.

2. Architecture with homeostatic synaptic scaling

The SOM model is trained with a series of episodes in which randomly selected input vectors are presented. At each step, the input vector, \vec{x} , is first multiplied by the feedforward weights, W_{FF} . In order to get the self-organizing map effect, this feedforward activity is then multiplied by a set of lateral connections, W_{lat} . The patterns of weights that the neurons send out laterally are identical to a preset prototype, so multiplying by W_{lat} is equivalent to taking a convolution with a kernel, g .

$$\vec{y}_{ff} = W_{FF}\vec{x} \quad (1)$$

$$\vec{y} = f[g * \vec{y}_{ff}]. \quad (2)$$

Here $f[a] = \max(0, a)$ and g is preset to a Mexican hat shape. These steps are similar to those proposed recently by Kohonen (2005, 2006). In that case, though, g took on a different form, as did the function f . Also, multiplying by a Mexican hat kernel was explored by Kohonen (1982), but in that work the kernel was applied iteratively until the neural activities were no longer changing. This is a time-consuming process so is not pursued here.

In this version the lateral connections are not updated with learning, but after the output activity is set the feedforward weights are updated with a Hebbian learning rule:

$$\tilde{w}_{ij}^t = w_{ij}^{t-1} + \alpha x_j^t y_i^t \quad (3)$$

α is the Hebbian learning rate, x_j is the presynaptic activity and y_i is the postsynaptic activity. \tilde{w}_{ij} is simply a weight value after Hebbian learning, but before weight normalization. The superscripts t and $t-1$ are added here to index the mathematical processing steps in this discrete-time model. The superscript t is used to indicate a current value, while $t-1$ indicates that the value is the result of calculations performed in the previous time step.

It is this type of Hebbian learning rule that would normally give problems. Since each update is positive, there is nothing to limit the growth of the weights. Normally, a weight

normalization is used that is based on the sum of the magnitudes of the weights coming into each neuron. In our case, we will normalize the weights with a value based on the recent activity of the neuron:

$$w_{ij}^t = \frac{\tilde{w}_{ij}^t}{\text{ActivityNorm}_i^t} \quad (4)$$

$$\text{ActivityNorm}_i^t = 1 + \beta_N \left(\frac{A_{\text{avg},i}^{t-1} - A_{\text{target}}}{A_{\text{target}}} \right). \quad (5)$$

Here, A_{target} is the internally defined preferred activity level for the neurons, $A_{\text{avg},i}$ is the average of recent activity for neuron i , and β_N is the homeostatic learning rate. This model assumes that some intracellular chemical, something activated by calcium for example, is integrated over the recent past and is related to the average firing rate. If the target level of this chemical is exceeded, the incoming synapse sizes will be decreased multiplicatively. If the target is not achieved, the synapses will be increased. Since the underlying relevant chemicals and their dynamics are not yet known, we instead use the average firing rate and firing rate target value directly in computing *ActivityNorm*.

In the model, each neuron keeps track of its average output firing rate, $A_{\text{avg},i}^t$, with a running average over the recent past.

$$A_{\text{avg},i}^t = \beta_C y_i^t + (1 - \beta_C) A_{\text{avg},i}^{t-1}. \quad (6)$$

Here, β_C controls the time window over which the neuron's firing rate is averaged.

This is a local computation, in the sense that each neuron keeps track of its own average firing rate. If this average, or difference from the target, is expressed as an internal level of some chemical all the synapses would conceivably have access to that information. Using the $A_{\text{avg},i}^t$ and A_{target} values directly avoids modelling the concentrations of unknown chemicals, but as more details become available through experiments, the model can become more explicit. It is interesting to note that there are different degrees to the locality of information. There is (1) information local to individual synapses, (2) information local to individual neurons, and (3) global information (information exchanged across neurons). The standard weight normalization and the new homeostatic mechanism both belong to the second class. One difference, however, is the nature of the information exchange inside the neuron. In the standard weight normalization case, a synapse would communicate its magnitude to the soma, which would in response change all the other synapse strengths. In the new method the communication is one way: the soma decides, based on the average firing rate, whether all the synapses should be increased or decreased.

3. Simulation results

Self-organizing maps were simulated using the synaptic scaling described with the previous equations. The input vectors used in these simulations are specified by a 1-D Gaussian shape (standard deviation σ of 15 units) and are presented

one per episode. For each episode, the input Gaussian is centered on one of the input units selected at random. The input locations are drawn from a uniform distribution for the experiments described in Sections 3.1 and 3.2. Other probability distributions are used in Section 3.3 as described. A typical training session lasts for 100,000 episodes. At each step Eqs. (1) and (2) are used to determine the neuron activities using the randomly selected input vector, \vec{x} . After the neuron activities are determined the Hebbian weight update of Eq. (3) is applied to every feedforward weight, as is the synaptic scaling rule of Eq. (4). Also, during each episode the *ActivityNorm* value and the A_{avg} value must be updated for each neuron using Eqs. (5) and (6), respectively. In other work, we show how to find the range of effective learning rate parameters (Sullivan & de Sa, 2006), so only values in this range are used. In all the simulations described here $\alpha = 8.3 \times 10^{-4}$, $\beta_N = 3.3 \times 10^{-4}$, $\beta_C = 3.3 \times 10^{-5}$, and $A_{\text{target}} = 0.1$ Hz.

3.1. Homeostasis and map formation

In order to verify proper formation of a map, a network was created with 150 inputs and 15 outputs. Both the inputs and outputs are arranged in a ring configuration to eliminate edge effects. The input vectors are specified by a 1-D Gaussian shape (standard deviation σ of 15 units). The input Gaussian is centered on one of the input units, selected uniformly at random. Plots of typical network behavior are shown in Fig. 1. In the top left plot, the average firing rate of the output neurons is shown. As the simulation progresses the average neuron firing rate approaches A_{target} . For each input, if we view the most active output neuron as the winner, then we can keep track of the neurons' winning percentages. The top right plot shows these winning percentages for all the output neurons. It can be seen that they approach a roughly equal probability of winning. The bottom plot shows an input–output map that has formed after training has ended. To obtain this plot, every possible input was presented to the network, one at a time. The winning output neuron (the one with the highest output rate) was then recorded for each input. The input number is shown on the x axis, and the corresponding winning output is plotted. This is a good map since similar inputs (inputs whose center is located on neighboring input units) correspond to nearby winning output units.

3.2. Synapse proliferation

Homeostatic mechanisms that maintain a steady output firing rate may play a particularly important role during development. As many neurons and synapses are added and pruned away, the total amount of input drive to a neuron will change dramatically. In order to avoid having a saturated firing rate, neurons must regulate themselves. Additionally, during normal functioning in a hierarchical system such as the visual cortex, one area's output is another's input. For the benefit of higher areas, it may be important for neurons to maintain a consistent firing rate level.

In order to test the ability of homeostatic synaptic scaling to withstand dramatic changes in network architecture, we created

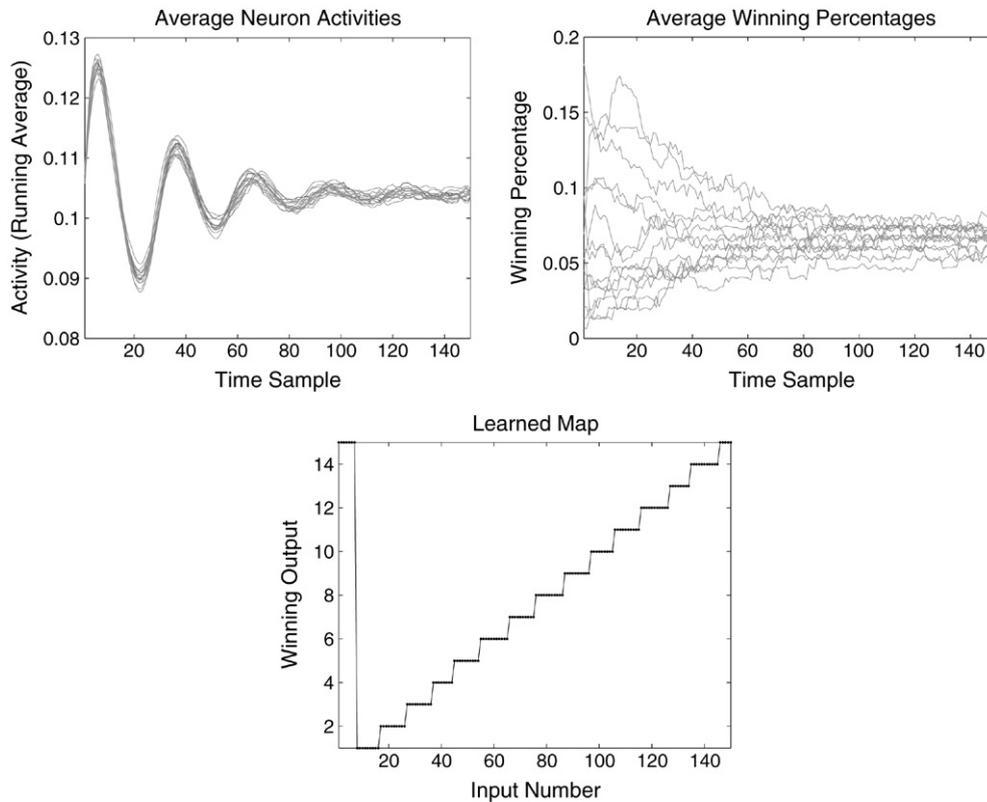


Fig. 1. Typical behavior. 150 inputs, 15 outputs, ring topology, $\alpha = 8.3 \times 10^{-4}$, $\beta_N = 3.3 \times 10^{-4}$, $\beta_C = 3.3 \times 10^{-5}$, and $A_{\text{target}} = 0.1$ Hz. (Top Left) The average neuron activities are driven to the same value. A running average of the firing rate is shown. (Top Right) The winning percentages of each of the neurons are shown. They converge to a roughly equal winning frequency. (Bottom) A smooth input–output map is formed. For each possible input, the output winner (the neuron with the maximum firing rate) is plotted. Both the inputs and outputs are arranged in a ring configuration to eliminate edge effects (so Output 1 is next to Output 15, for example).

a simulation in which the number of input neurons doubled after learning had begun. The network started with 75 inputs and 15 output neurons. After the average neuron activities settled to a constant value, 75 more inputs were added, as shown in the top panel of Fig. 2. After the neuron activities settled again, the 75 added inputs were taken away. This is a simple example meant to simulate the large scale neuron proliferation and die-off seen during cortical development.

The effect on the average output activities is shown in the bottom left panel of Fig. 2. When the number of inputs was changed, the average firing rate of the output neurons changed. The firing rates quickly returned to the target value in the network with the homeostatic mechanism. Furthermore, the continuity (smoothness) of the map was unaffected. The network using standard weight normalization has its average output firing rate permanently changed by the additional (or subtracted) neurons. To understand how the new model keeps a steady output while the standard model fails we can examine the difference in input vector and the resulting difference in neuron weights. The top of Fig. 3 shows a typical input vector during the proliferation stage and one during the die-off stage. In both cases, a Gaussian input pattern is used with the same width and a norm of 1. The individual neuron activities are smaller during proliferation because more neurons contribute to make a total activity of 1. One neuron's weight vector was chosen from each network and the size of each

component of the vector is displayed during the proliferation and die-off stages. In the standard weight normalization case during proliferation (shown bottom, left), the individual weight components become smaller because the norm of the weight vector is stuck at a constant value (the value was chosen to be 7 in this case in order for the output activities of the two networks to be similar). The result is that the neuron activities fall. The weight vector norms using homeostasis are not bound like this (as shown bottom, right) and as a result the individual components change as required to keep the average output activity at the target value.

Theoretically, it is not difficult to see why the standard model fails. The feedforward excitation into a neuron is produced by a dot-product between the input vector, x , and the neuron's weight vector, w . We can write the dot-product in terms of the magnitudes of these vectors:

$$x \cdot w = \|x\|_2 \|w\|_2 \cos \theta \leq \|x\|_2 \|w\|_2 \leq \|x\|_1 \|w\|_1. \quad (7)$$

We can see that the dot-product is bound by the norm of the weight vector. Standard weight normalization sets the norm of the weight vector to a constant value, thus bounding the feedforward excitation. The homeostatic mechanism allows this norm to adapt up or down as necessary.

To further illustrate the flexibility of the network using homeostasis, a few more examples are given. If during the proliferation stage the input vector norm is changed to 3

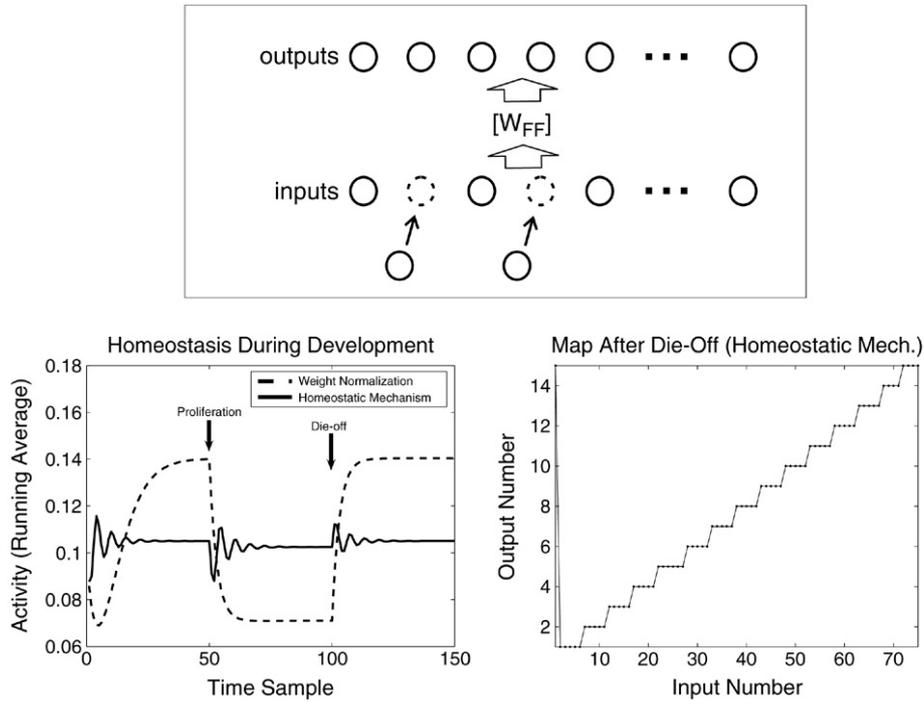


Fig. 2. Addition of new inputs. (Top) Seventy-five additional inputs were added halfway through learning. (Bottom Left) The average neuron activity is disturbed when new inputs are added or taken away, but the average quickly recovers if the homeostatic mechanism is used. (Bottom Right) For both cases, a smooth input–output map is preserved after neuron proliferation and die-off.

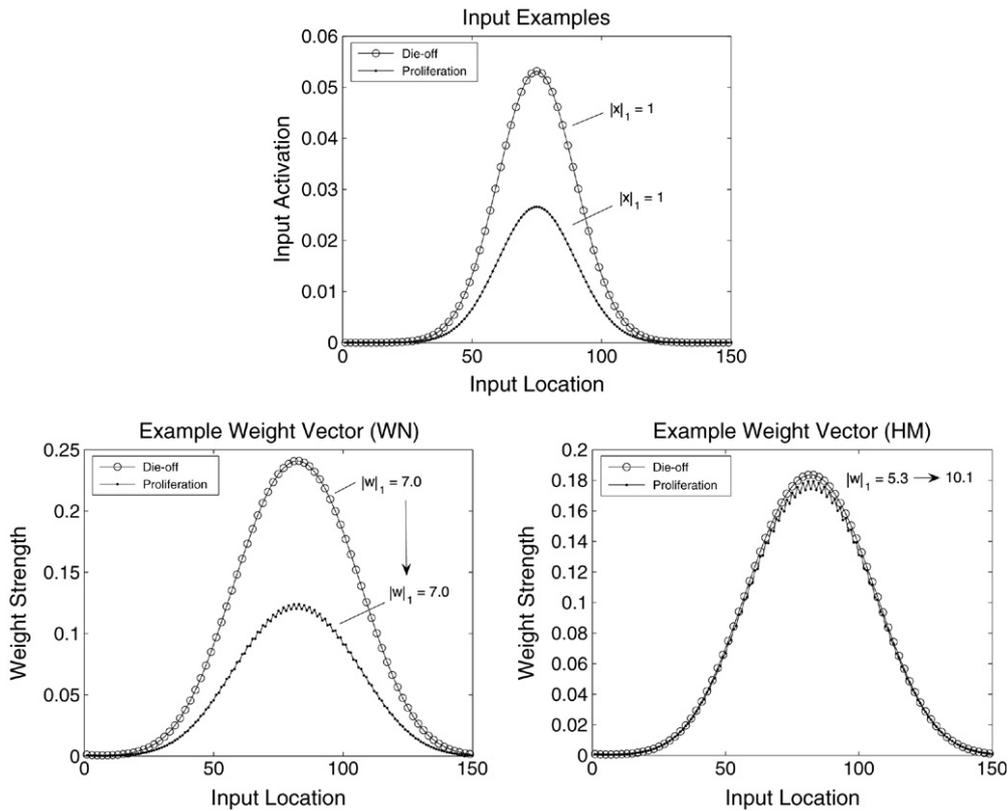


Fig. 3. Adaptation of weight vectors. (Top) The norm of the input vectors is kept constant during proliferation and die-off. With the addition or deletion of input components, the total of the norm is redistributed. (Bottom) An example weight vector into one neuron in the two networks. (Bottom Left) Standard Network. The norm of the weight vector is set to a constant value, which results in a nonconstant output activation during proliferation and die-off. (Bottom Right) Homeostatic Network. The norm of the weight vector is adjusted by the neuron to keep a constant average activation.

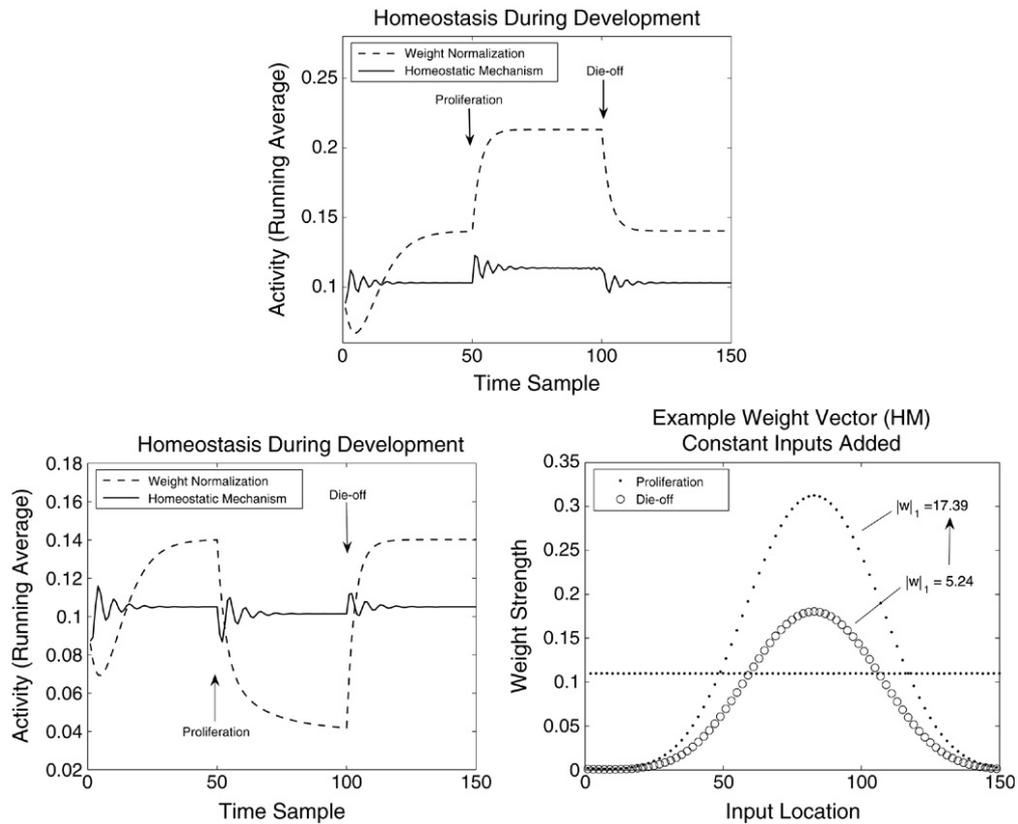


Fig. 4. Two additional examples. (Top) During proliferation, the norm of the input vector is increased to 3 (instead of 1). The average neuron activation in the standard network increases (instead of decreasing like before), but it remains fairly constant in the homeostatic network. (Bottom) During proliferation, the added neurons have a constant activation of 0.0067. (Bottom Left) As before, the neuron activities in the homeostatic network remain fairly constant, while the activities in the standard network change dramatically. (Bottom Right) As before, the norm of the weight vectors adapt to keep the average activity constant. An example weight vector is shown here during proliferation and die-off.

instead of 1, the average output activation of the neurons in the standard network increases as shown at the top of Fig. 4. The activation of the homeostatic neurons is fairly constant. A different situation in which the added inputs are useless, constantly on at a certain level, was examined as well. The value of the added inputs is a constant 0.0067 for all input vectors (which makes the 75 inputs sum to 0.5, half of the total input magnitude). A similar adaptation by the homeostatic network and failure by the standard network is shown bottom, left, in Fig. 4. An example weight vector from this homeostatic network, shown bottom, right, shows the adaptation necessary to keep the average firing rates of the neurons constant. Finally, it should be noted that the homeostatic mechanism being used is more general and can respond to more than just proliferation and die-off of neurons. Suppose there is a situation in which all the input vectors suddenly increase or decrease in magnitude (at eye-opening perhaps). Without the homeostatic mechanism, there is a potential for saturating neurons with too much or too little input. To see a case like this, a simulation was performed in which the magnitudes of the input vectors were suddenly increased from 0.5 to 1.0. They were later decreased from 1.0 back to 0.5. Two example input vectors are shown on the left of Fig. 5. Like in previous cases, the homeostatic network was able to recover and keep its neuron activations near the target

value, while the standard network failed to do this. The plot on the right of Fig. 5 illustrates this.

3.3. Probability representation

Computer simulations were run that compare networks using homeostatic scaling with networks using standard weight normalization. Networks of neurons that maintain their own average firing rate avoid the problem of dead units that do not respond to any input and overactive neurons. Intuitively, it seems that networks of this type might get more out of their neurons and thus increase their information transfer. This thought is reminiscent of networks that explicitly try to obtain maximum information, such as the works of DeSieno (1988) using a Conscience mechanism and Linsker (1989) using information maximization. In both of these examples, though, global information is needed at the synapses. Perhaps the homeostatic mechanism, or some variant, can approximate information maximization within a biologically realistic framework. This idea is tested here by comparing network performance with inputs drawn from several probability distributions.

Since our networks do not have a winner-take-all output, there is no obvious winner for each input. For the sake of comparison, we will define the winner as the neuron with the

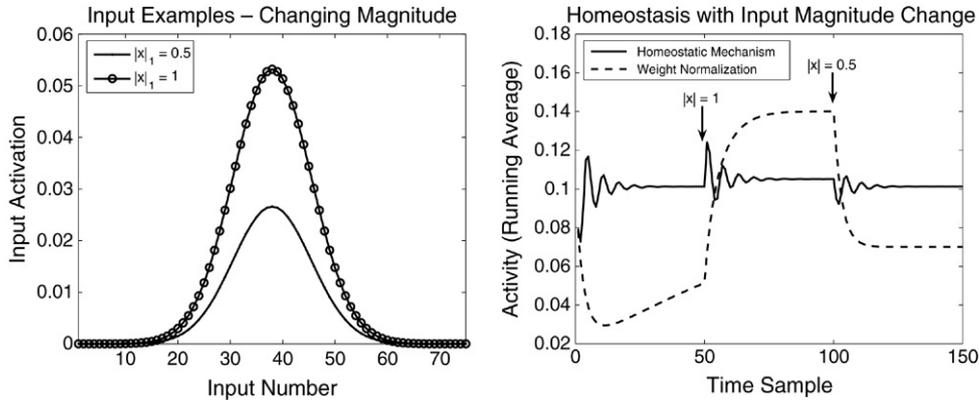


Fig. 5. Changing input magnitude. To simulate an event like eye-opening, the magnitude of the input vector was changed without any proliferation or die-off. (Left) Example input vectors. The norm of the input vectors is changed from 0.5 to 1.0 and back again. (Right) As before, the average activations of the neurons in the homeostatic network remain fairly constant, while the activations in the standard network do not.

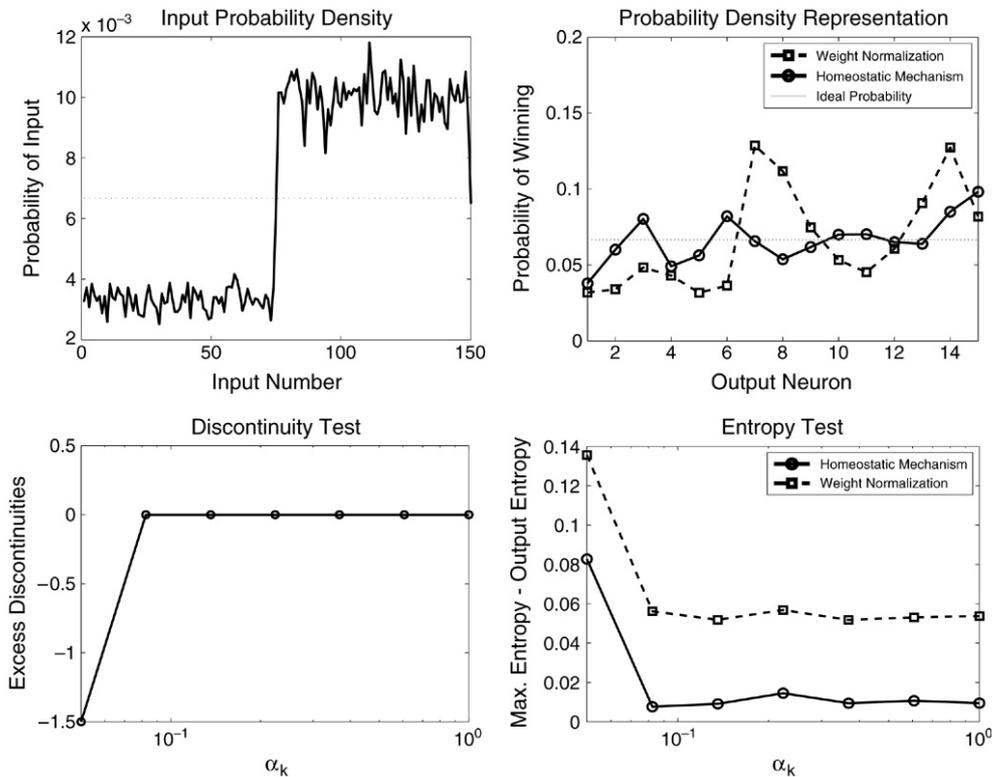


Fig. 6. Performance Comparison on a 150×15 Map with Step Input Distribution. (Top Left) The centers of the input patterns were drawn from this step distribution in which half of the potential inputs were three times more likely than the others. The input patterns were Gaussian shapes with a standard deviation width equal to 15 input units. (Top Right) For one simulation for each case (standard weight normalization (WN) and the homeostatic mechanism (HM)), the actual output winning probabilities are shown. The weight normalization case has large errors at the steps in the input probability densities. (Bottom) Two measures of map quality are used to compare performance of WN and HM. The Hebbian learning rate, α_k , was varied over a wide range. Each point is the average of five separate computer simulations. (Bottom Left) The number of discontinuities in a given input–output map (as in the bottom panels) were counted and subtracted from the number of output units. A smooth map that utilizes all the outputs will have a Discontinuity Test score of 0. WN and HM have the same perfect performance. (Bottom Right) For a large number of randomly chosen inputs, the output unit with the highest activation was called the winner. The entropy of the output unit winning probabilities was computed and subtracted from the highest possible entropy (all winning an equal number of times). The best value is zero. The homeostatic mechanism had entropy that was closer to maximum (and thus higher information content) in all cases.

highest output rate. In this way we can find the probability of winning for each output neuron, and then calculate the entropy of this vector of probabilities. For our network with population-coded outputs, it may not be completely fair to

measure information in this way, but for a first approximation it might be helpful.

The first example, whose results are shown in Fig. 6, uses networks with 150 inputs and 15 outputs. The learning rates

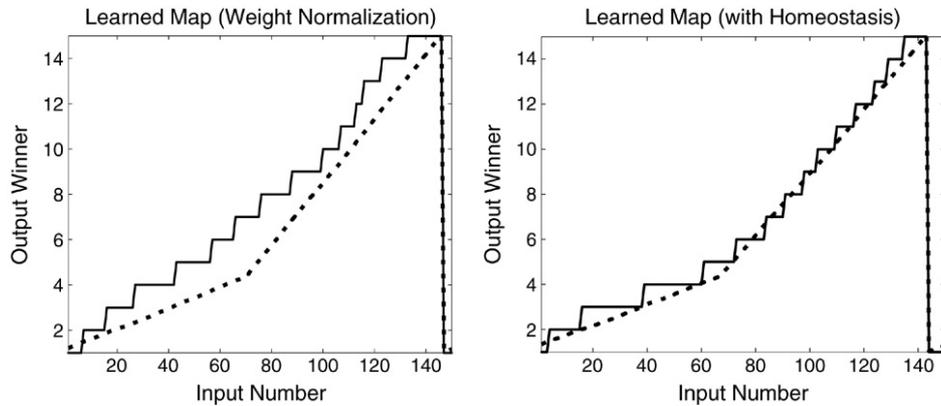


Fig. 7. Map Formation Comparison on a 150×15 Map with Step Input Distribution. For the simulation whose results are depicted, the final input–output maps are shown with (Left) WN and (Right) HM. For reference, the cumulative distribution of the input is plotted with a dashed line. The network with the homeostatic mechanism has learned a mapping that better matches this distribution, and thus increases the output entropy.

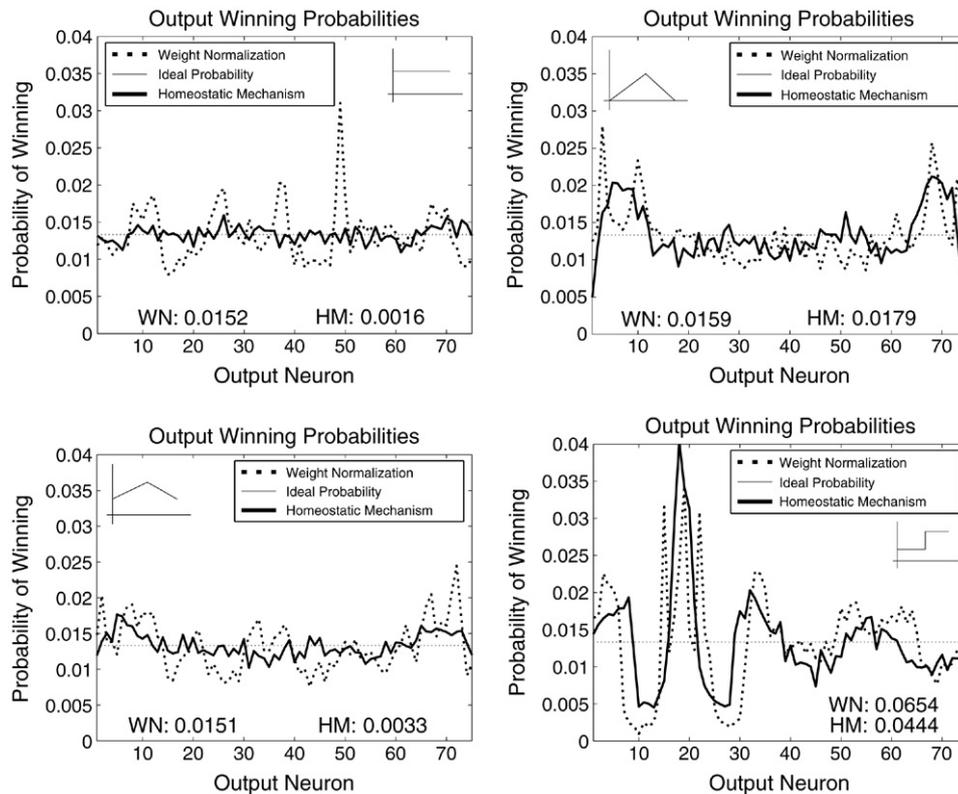


Fig. 8. Examples with Various Input Density Distributions. Each panel shows the results of simulations with the two networks (Standard Weight Normalization (WN) and the Homeostatic Mechanism (HM)) using a different input distribution. Within each panel, a diagram shows the input distribution in the corner. The probability of winning for each output neuron is plotted. The dark line represents HM, and the dashed line shows WN. A faint line gives the ideal output probability, which is 1 divided by the number of output units. The two numbers presented at the bottom are the resulting entropy measures. These numbers are the difference between the entropy of the output probabilities and the maximum entropy for this network, making zero the best possible value. In three of the four examples, HM performs better than WN.

and target output rates were set the same as above. The inputs are drawn from a step function with half the inputs being unlikely, while the other half are more likely (as seen in the top left panel). In this case, the network with homeostatic scaling has consistently better performance as measured by output entropy (see the bottom right panel). The top right panel shows the probability of winning for each output neuron for one simulation. This was obtained by testing the trained network on

a representative set of inputs. The difference in performance between the two network types is due to the shape of the input–output map that forms (Fig. 7).

Several more networks were tested using different input distributions. For these simulations, networks of 750 inputs and 75 outputs were used. Typical results for each distribution are shown in Fig. 8. In each plot, the input distribution is shown in one corner, next to the legend. The plot shows the probability

of winning for each output neuron. The dark line represents the neurons in the network with the homeostatic mechanism, and the dashed line gives the network with standard weight normalization. A faint line gives the ideal output winning probability, which is 1 divided by the number of output units. At the bottom of each plot are the resulting entropy measures for the two networks, standard weight normalization (WN) and the homeostatic mechanism (HM). These numbers are the difference between the entropy of the output probabilities and the maximum entropy for this network, making zero the best possible value.

In three out of four cases, the network with the homeostatic mechanism had better performance. The network with standard weight normalization was slightly better in the second case. Interestingly, this ramp-like distribution caused the homeostatic mechanism to converge to a state in which some neurons rarely won (had the most activation). These output neurons received enough activation from neighboring winners that their target activity goal was achieved. In other words, all neurons had similar average activities, but some neurons rarely ‘won’. Also interesting were the results of the last distribution. This was the same input step distribution as was used in the example above. As the network size increased, performance gets worse for both networks. This is again due to the algorithm optimizing for average firing rate, not average winning percentage. The discrepancy between these measures, especially in the regions of low probability, should be interesting grounds for future investigation.

We do not fully understand why the homeostatic network performs better in these simulations. Our intuition leads us to believe that in this network, neurons that are not responding much can make themselves more useful by increasing their average activations. This is not necessarily the same as increasing a winning percentage, but it may be some sort of approximation. We believe that this can be an interesting area for further investigation and we would benefit by understanding some theoretical aspects behind this performance.

4. Conclusions

In this work, we have proposed a way to go beyond the standard weight normalization. This long-used measure has worked to counteract the unconstrained growth of Hebbian learning, but there is little evidence from experiments that justifies its use. Homeostatic synaptic scaling, on the other hand, has been seen recently in biological experiments. It has been shown here that using homeostatic synaptic scaling in place of standard weight normalization still leads to proper organized map formation. In addition, the neurons are able to maintain their average firing rates at a set point. This helps them from becoming saturated as neurons are added or taken away, as happens during development. Finally, it was shown that synaptic scaling in some cases leads to a better representation of the input probability distribution compared with weight normalization. This observation suggests the intriguing possibility that this homeostatic mechanism helps drive the network to a state of increasing information transfer.

The output entropy was measured using the probability of each output neuron having the highest activation. This may not be the most natural way to measure information transfer in this network, since a population code is used as the output. Indeed, since the neurons’ goal is to maintain a useful average firing rate, information transfer may not be the most important measure of performance. These issues will be addressed in future work. The algorithm will also be tested more extensively with two-dimensional input and output spaces. An interesting challenge is how to integrate this mechanism into existing models of cortical development like the LISSOM (Sirosh & Miikkulainen, 1997) and whether it will lead to increased performance in practical applications (Choe, Sirosh, & Miikkulainen, 1996). Additionally, a more thorough investigation into various forms of homeostatic equations could be interesting. For example, in other work a threshold on an output sigmoid was changed according to a neuron’s output activity (Gorchetchnikov, 2000; Horn & Usher, 1989). Without further analysis, it is hard to compare, since their networks were a Hopfield net and a winner-take-all associative memory, respectively. It is possible, though, that a functionally similar effect will result.

Acknowledgements

We would like to thank Jochen Triesch for helpful comments on this work. This material is based upon work supported by the National Science Foundation under NSF Career Grant No. 0133996 and was also supported by NSF IGERT Grant #DGE-0333451 to GW Cottrell. This work was also supported by an NIH Cognitive Neuroscience Predoctoral Training Grant.

References

- Bi, G. -q., & Poo, M. -m. (2001). Synaptic modification by correlated activity: Hebb’s postulate revisited. *Annual Review of Neuroscience*, 24, 139–166.
- Bliss, T. V. P., & Lomo, T. (1973). Long-lasting potentiation of synaptic transmission in the dentate area of the anesthetized rabbit following stimulation of the perforant path. *Nature Neuroscience*, 232, 331.
- Chechik, G., Meilijson, I., & Ruppin, E. (2001). Effective neuronal learning with ineffective hebbian learning rules. *Neural Computation*, 13, 817–840.
- Choe, Y., & Miikkulainen, R. (2004). Contour integration and segmentation with self-organized lateral connections. *Biological Cybernetics*, 90, 75–88.
- Choe, Y., Sirosh, J., & Miikkulainen, R. (1996). Laterally-interconnected self-organizing maps in hand-written digit recognition. In D. Touretzky, M. Mozer, & M. Hasselmo (Eds.), *Advances in neural information processing systems 8: Vol. 8*. Cambridge, MA: MIT Press.
- DeSieno, D. (1988). Adding a conscience to competitive learning. In *Proceedings of the IEEE international conference on neural networks, vol. 1* (pp. 117–124).
- Gorchetchnikov, A. (2000). Introduction of threshold self-adjustment improves the convergence in feature-detective neural nets. *Neurocomputing*, 32–33, 385–390.
- Horn, D., & Usher, M. (1989). Neural networks with dynamical thresholds. *Physical Review A*, 40(2), 1036–1044.
- Kohonen, T. (1982). Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43, 59–69.
- Kohonen, T. (2001). *Self-organizing maps* (3rd ed.). Springer.
- Kohonen, T. (2005). Pointwise organizing projections. In M. Cottrell (Ed.), *Proceedings of the 5th workshop on self-organizing maps* (pp. 1–8).
- Kohonen, T. (2006). Self-organizing neural projections. doi:10.1016/j.neunet.2006.05.001.

- Linsker, R. (1989). How to generate ordered maps by maximizing the mutual information between input and output signals. *Neural Computation*, 1(3), 402–411.
- Maffei, A., Nelson, S. B., & Turrigiano, G. G. (2004). Selective reconfiguration of layer 4 visual cortical circuitry by visual deprivation. *Nature Neuroscience*, 7(12), 1353–1359.
- Miller, K. D. (1996). Synaptic economics: Competition and cooperation in correlation-based synaptic plasticity. *Neuron*, 17, 371–374.
- Obermayer, K., Blasdel, G., & Schulten, K. (1992). Statistical-mechanical analysis of self-organization and pattern formation during the development of visual maps. *Physical Review A*, 45(10), 7568.
- Sirosh, J., & Miikkulainen, R. (1997). Topographic receptive fields and patterned lateral interaction in a self-organizing model of the primary visual cortex. *Neural Computation*, 9(3), 577–594.
- Stemmler, M., & Koch, C. (1999). How voltage-dependent conductances can adapt to maximize the information encoded by neuronal firing rate. *Nature Neuroscience*, 2(6), 521–527.
- Sullivan, T. J., & de Sa, V. R. (2006). A self-organizing map with homeostatic synaptic scaling. *Neurocomputing*, 69, 1183–1186.
- Triesch, J. (2004). Synergies between intrinsic and synaptic plasticity in individual model neurons. In *Advances in neural information processing systems: Vol. 17*.
- Turrigiano, G. G., Leslie, K. R., Desai, N. S., Rutherford, L. C., & Nelson, S. B. (1998). Activity-dependent scaling of quantal amplitude in neocortical neurons. *Nature*, 391, 892–896.
- Turrigiano, G. G., & Nelson, S. B. (2004). Homeostatic plasticity in the developing nervous system. *Nature Reviews Neuroscience*, 5, 97–107.
- Yeung, L. C., Shouval, H. Z., Blais, B. S., & Cooper, L. N. (2004). Synaptic homeostasis and input selectivity follow from a calcium-dependent plasticity model. *Proceedings of the National Academy of Sciences*, 101(41), 14943–14948.
- Zhang, W., & Linden, D. J. (2003). The other side of the engram: experience-driven changes in neuronal intrinsic excitability. *Nature Reviews Neuroscience*, 4, 885–900.