



**HAL**  
open science

## Dynamic Formation of Self-Organizing Maps

Jérémy Fix

► **To cite this version:**

Jérémy Fix. Dynamic Formation of Self-Organizing Maps. 10th International Workshop, WSOM, Jul 2014, Mittweida, Germany. pp.25-34, 10.1007/978-3-319-07695-9\_2 . hal-01060971

**HAL Id: hal-01060971**

**<https://centralesupelec.hal.science/hal-01060971v1>**

Submitted on 7 May 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Dynamic formation of self-organizing maps

Jérémy Fix\*

Suplec, IMS-MaLIS Research Group UMI-2958,  
2 rue Edouard Belin, 57070 Metz  
Jeremy.Fix@Supelec.fr

**Abstract.** In this paper, an original dynamical system derived from dynamic neural fields is studied in the context of the formation of topographic maps. This dynamical system overcomes limitations of the original Self-Organizing Map (SOM) model of Kohonen. Both competition and learning are driven by dynamical systems and performed continuously in time. The equations governing competition are shown to be able to reconsider dynamically their decision through a mechanism rendering the current decision unstable, which prevents the use of a global reset signal.

**Keywords:** dynamic neural fields, self-organizing maps

## 1 Introduction

Within the context of situated or embodied cognition[9], there is a growing interest in finding complex, non-linear dynamical systems supporting various functions such as perception, working memory or action. These dynamical systems would interact continuously with their environment being shaped by the stimuli they receive and producing continuously motor programs to be executed. In this paper, we focus on self-organizing maps. In self-organizing maps, there is a notion of topology and of similarity: neurons that are close together will encode similar aspects of those stimuli that they receive as input. This situation is observed for example in the visual cortex where, despite some non-linearity such as the pinwheels, there is a continuous orientation selectivity along the cortical tissue. In the context of the formation of topographically organized maps, there is one usual assumption having strong consequences on the model. It is indeed usually assumed that the network is reset when a new sample is presented to the network. If we completely develop this assumption, we can identify two situations. In the first one, between two successive stimuli, one may assume that the network receives no excitation and that this period is sufficient for the network to go back to a resting state. In the second, more constrained, situation, the stream of stimuli is continuous and there is no possibility to really separate one stimulus from the others. In such a continuous stream of inputs, it is hard to figure out a mechanism that would reset the network to put it in a more favorable state. In

---

\* Jérémy Fix is in the IMS-MaLIS Research Group, Suplec (France).

this article, we advocate that such a mechanism is not necessary by introducing a dynamical neural network, relying on the dynamic neural field theory, able to *reconsider its decision* whenever it is no more appropriate. Indeed, as we will see, if the network creates a decision with a bump of activity not localized where the input is locally maximal, this decision becomes unstable. A second property, in line with the previous one, is that the prototypes of the self-organizing map are continuously learned and there is no need to wait that the decision is settled before triggering learning. Competition and learning are two processes occurring at the same time although at different time scales. The article is organized as follows: in section 2 we present the standard neural field equation of [2], explain some limitations in the context of decision making and self-organization and introduce a modified neural field equation. In section 3, we illustrate the behavior of the neural field on artificial data and finally demonstrate its ability to drive a learning process within the context of self-organizing maps. The python scripts used to run the simulations of the paper are available online[5].

## 2 Methods

The Dynamic Neural Field equation (DNF), as introduced in [2, 11], reads :

$$\frac{dV}{dt}(x, t) = -V(x, t) + \int_y w(x, y) f(V(y, t)) + I(x, t) \quad (1)$$

where  $V(x, t)$  is the membrane potential at time  $t$  and position  $x$ ,  $w(x, y)$  is the weight of the lateral connection between the units at position  $x$  and  $y$  and usually taken as an on-center/off-surround difference of Gaussians and  $f$  a transfer function mapping the membrane potential to the firing rates and usually taken as a Heaviside function or a sigmoid. Some properties of such a dynamical system have been formally described and others have been observed experimentally. In particular, essentially depending on the shape and extent of the lateral connections, one may observe competitive behaviors or working memory behaviors. While competitive behaviors mainly rely on inhibitory influences within the field, the memory property can be obtained with more local and excitatory weights. If one is interested in the ability of a neural field to form a decision, this equation has experimentally some drawbacks. In particular, if we suppose that the same neural field should be able to form a decision both when the input is randomly distributed (as we encounter initially with self-organizing maps, when the prototypes are drawn randomly) and when the input is made of several localized bumps of activities (e.g. when a self-organizing map actually gets organized, or when representing alternative motor plans), it appears quite difficult to tune the parameters. Intuitively, in order the neural field to form a localized decision in case of a randomly distributed input, one must introduce some lateral excitation within the field to recover a coherency in the shape of the decision absent from the input. When the inputs are already localized bumps, this lateral excitation is much less important relative to the lateral inhibition supporting the ability to decide. The lateral excitation can even have dramatic consequences in the

second situation since it introduces some inertia of the decision and may prevent the neural field to reconsider the decision, in particular when the input changes. We must note that a self-organizing neural field, based on the original Amari's equation, has been proposed in [4]. However, as stated by the authors, when a new sample is presented to the network, “*the activity of the field is reset to zero*”. This is a limitation we want to overcome and that motivates the new neural field equation we propose. Indeed let us consider the dynamic neural field defined by equations (2).

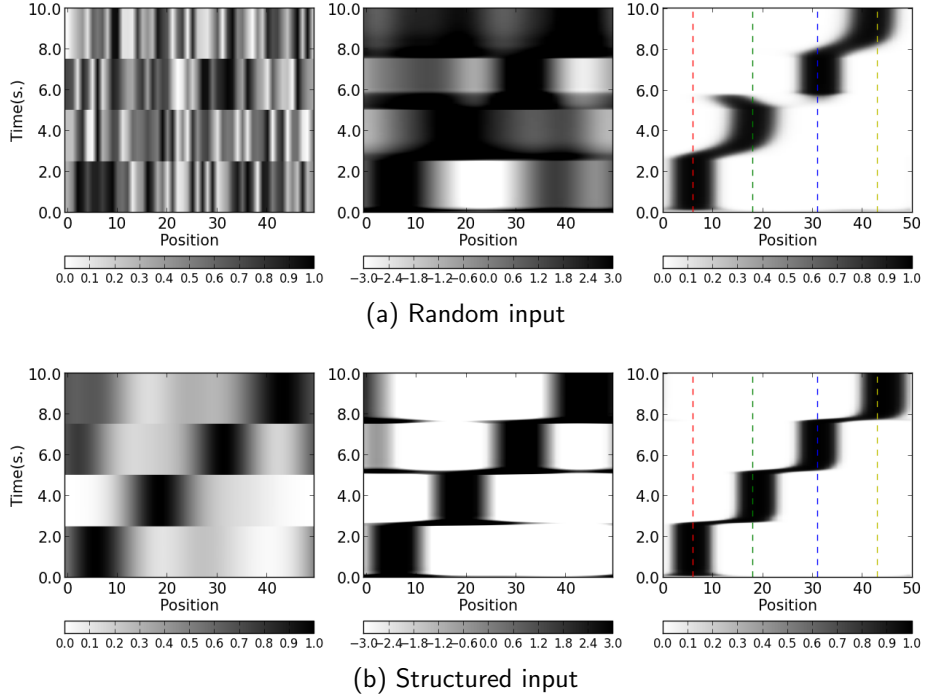
$$\begin{aligned}\tau \frac{dU}{dt}(x, t) &= -U(x, t) + \int_y g(x, y)I(y, t) - \int_y f(V(y, t))I(y, t) \\ \tau \frac{dV}{dt}(x, t) &= -V(x, t) + \int_y w(x, y)f(V(y, t)) + U(x, t)\end{aligned}\quad (2)$$

where  $f$  is a transfer function,  $w$  is the lateral weight function and  $g$  a Gaussian. In the following, we consider the firing rates  $f(V(x, t))$  as representing the decision of the neural field. This equation is similar to the standard dynamic neural field equation except that an intermediate layer with the membrane potentials  $U(x, t)$  is introduced. This intermediate layer combines the feedforward input  $I(x, t)$  with a feedback from the output layer  $V(x, t)$ . The motivation behind this intermediate layer is to modulate the input feeding the  $V$  layer in order to boost the input when a decision emerges in the neural field that does not correspond to the input region that is locally maximal. This transient boost, that will be canceled if the decision switches, helps in destabilizing the decision. In the next section, we begin by showing experimentally that such a neural field is able to dynamically form a decision where the input is locally maximal. Then, introducing adaptive prototypes for defining the input, we show experimentally that such a neural field can actually self-organize in a way similar to Kohonen's self-organizing maps [7].

### 3 Results

#### 3.1 Detecting where the input is locally maximal

When self-organizing maps were introduced by [7], the author identified several processes “*1. Formation of an activity cluster in the array around the unit at which activation is maximum. 2. Adaptive change in the input weights of those units where activity was confined*”. The neural field equation we propose aims at being a neural system for the first process. The intuition is that a neural field governed by equation 2 should converge to a localized bump of firing rates in the  $V$  layer. This bump should be centered on the input region that is *locally* maximal. For clarity, we rewrite the neural field equation (3) that we consider in this section. The parameter  $\beta$  we introduce in the equation facilitates the tuning of the neural field for the latter simulations. In addition, we discretize the space in  $N$  positions and write sums rather than integrals.



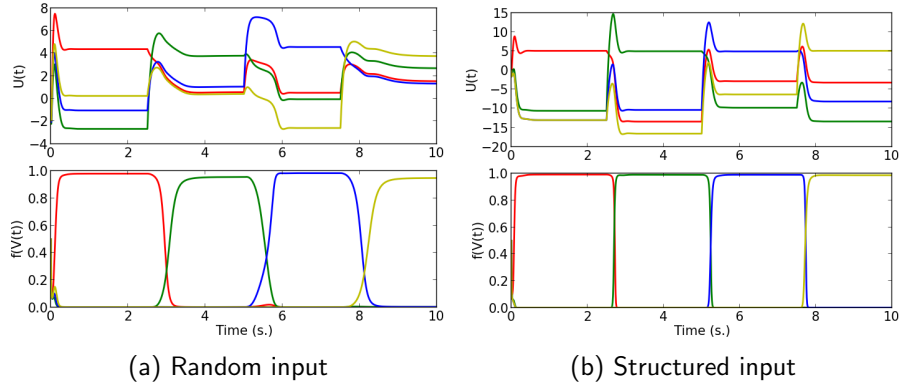
**Fig. 1.** A one-dimensional neural field with 50 positions is excited with randomly distributed inputs (a) or with more structured inputs (b). In both situations, we plot the input  $I(x, t)$  on the left, the  $U(x, t)$  membrane potential on the middle and the output firing rates  $f(V(x, t))$  on the right. The membrane potentials  $U(x, t)$  have been saturated for the illustration to lie in  $[-3, 3]$  but they typically lie in a larger domain (see Fig. 2). The dashed lines shown on  $f(V(x, t))$  indicate the slices plotted on Fig. 2. In all the conditions, the neural field is able to create a bump of activity localized where the input is locally maximal (in the sense of the input convolved with the Gaussian receptive field  $g(x, y)$ ). The bump of activity becomes unstable when the input is changed and the decision is no more appropriate.

$$\begin{aligned}
f(x) &= \frac{1}{1 + \exp(-x)} \\
\tau \frac{dU}{dt}(x, t) &= -U(x, t) + \beta \left( \sum_y g(x, y) I(y, t) - \sum_y f(V(y, t)) I(y, t) \right) \\
\tau \frac{dV}{dt}(x, t) &= -V(x, t) + \sum_y w(x, y) f(V(y, t)) + U(x, t)
\end{aligned} \tag{3}$$

The lateral weights  $w(x, y)$  are defined as an excitatory Gaussian with an inhibitory offset. The afferent weights  $g$  are defined as a Gaussian chosen to match approximately the size of the bump-like firing rates  $f(V)$  the neural field is producing. Before embedding the neural field within a self-organizing process (with prototypes associated with each location and used to compute the input distribution  $I$ ), we first simulate the competitive process governed by Eq. (3) with randomly distributed inputs. For clarity of the illustrations, we restrict ourselves to one dimensional neural fields. On Fig. 1 we simulate a 1D neural field with  $N = 50$  spatial positions,  $\tau = 0.05$ ,  $g(x, y) = \exp(-\frac{d(x, y)^2}{2\sigma_i^2})$  with  $\sigma_i = 4.7$ , lateral connections  $w(x, y) = A_+ \exp(-\frac{d(x, y)^2}{2\sigma_+^2}) - A_-$  with  $A_+ = 1.2$ ,  $\sigma_+ = 4.6$ ,  $A_- = 0.9A_+$ ,  $\beta = 2.6$ . The parameters of the neural field have been obtained following the method given in [6] with the cost function introduced in [1]. The neural field is integrated synchronously with Euler and  $\Delta t = 0.01$ . To avoid boundary effects, the distance within the field  $d(x, y)$  is cyclic. We consider two situations that will be relevant for the latter application of the neural field to self-organizing maps. In the context of self-organizing maps, we typically observe inputs that are initially random (because the prototypes are randomly sampled) and become more structured, bell-shaped, as learning goes on (because the prototypes actually get organized). These are the scenarios we simulate on Fig. 1a,b. On these figures, we plot the input  $I(x, t)$  function of time on the left, the membrane potentials  $U(x, t)$  on the middle and the output firing rates  $f(V(x, t))$  on the right which we define as the decision of the neural field<sup>1</sup>. In both situations, the neural field is able to create a bell-shaped decision centered where the input is locally maximal. In case the input is changed (which we do every few iterations), and the decision is no more localized appropriately, it becomes unstable and a new decision is created after a few steps. This is this property of dynamically reconsidering its decision, what one might call automatic reset, when the input is both random or more structured, that is hard to obtain with the original Amari equation.

In order to better appreciate the behavior of the neural field, we show on Fig. 2, the membrane potentials  $U(x, t)$  and output firing rates  $f(V(x, t))$  at the positions where the input is locally maximal and where the decision will

<sup>1</sup> One must note that the randomly generated inputs have been shifted in space to ensure that the field will create a bump at, successively, the positions  $x = 6, 18, 31, 43$ , which makes the illustrations easier to understand.



**Fig. 2.** Slices of the membrane potentials  $U(x, t)$  and firing rates  $f(V(x, t))$  at the positions  $x = 6, 18, 31, 43$  for the scenarios we consider on Fig. 1a,b. The spatial positions for the slices are indicated by dashed lines over the plots of  $f(V(x, t))$  on Fig. 1. When a decision is not appropriately localized, the membrane potentials  $U(x, t)$  get increased until the current decision is destabilized, allowing a new one to appear.

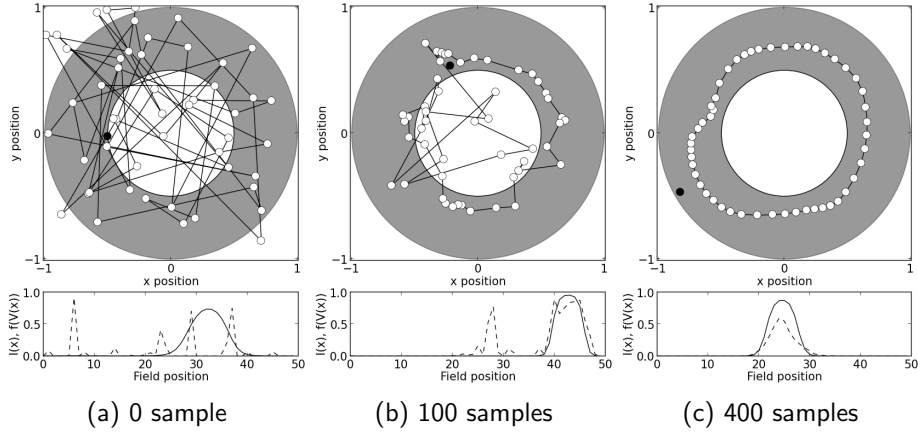
emerge. The transition around time  $2.5s.$ , both for random and structured inputs, is typical from the rest of the simulation. Until that time, the positions around  $x = 6$  won the competition. However, as the input gets changed, these locations are no more the most excited in the input. The potentials  $U(x, t)$  then begins to grow around the location  $x = 18$ . They grow until the current decision is destabilized and suppressed in layer  $V(x, t)$ . Following the suppression of the decision, a new decision appears around  $x = 18$  and the  $U(x, t)$  activities then come back to a resting level. The  $U(x, t)$  layer really brings a boosting signal to favor the correct decision. We repeatedly mention that the decision is created where the input is locally maximal. We do not bring a clear definition of locally maximal but observed experimentally that it corresponds to the position where the input convoluted by the gaussian-like decision is maximal.

### 3.2 A self-organizing dynamic neural field

We now embed the neural field equation within a learning architecture similar to Kohonen Self-Organizing Maps. Indeed, we consider the dynamical system given by Eq. (4). A prototype  $p(x, t) \in \mathbb{R}^2$  is associated with each position  $x \in [1..N]$  in the neural field. At regular time intervals, a sample  $\zeta$ , drawn from an unknown distribution  $D \subseteq \mathbb{R}^2$ , is presented to the field. The input  $I(x, t)$  at each position is computed as a Gaussian, with standard deviation  $\sigma$ , of  $\zeta$  centered on  $p(x, t)$ . Importantly, the prototypes are updated at every iteration and not just when the field has converged. Also, it is important to note that, even if the samples are presented at regular time intervals, there is no specific signal sent to the network to indicate it that a new sample is presented; the network will automatically update its decision in case it is no more appropriate.

We choose  $\sigma = 0.2$ ,  $\tau_p = 100$  and the equations are integrated synchronously with Euler and  $\Delta t = 0.01$ . The learning rate is set to  $\alpha = 0.001$ . The other parameters are the same as in the previous simulation.

$$\begin{aligned}
 I(x, t) &= \exp\left(-\frac{|p(x, t) - \zeta|^2}{\sigma^2}\right) \\
 \tau_p \frac{dp}{dt}(x, t) &= f(V(x, t))(\zeta - p(x, t)) \\
 \tau \frac{dU}{dt}(x, t) &= -U(x, t) + \beta \left( \sum_y g(x, y) I(y, t) - \sum_y f(V(y, t)) I(y, t) \right) \quad (4) \\
 \tau \frac{dV}{dt}(x, t) &= -V(x, t) + \sum_y w(x, y) f(V(y, t)) + U(x, t)
 \end{aligned}$$

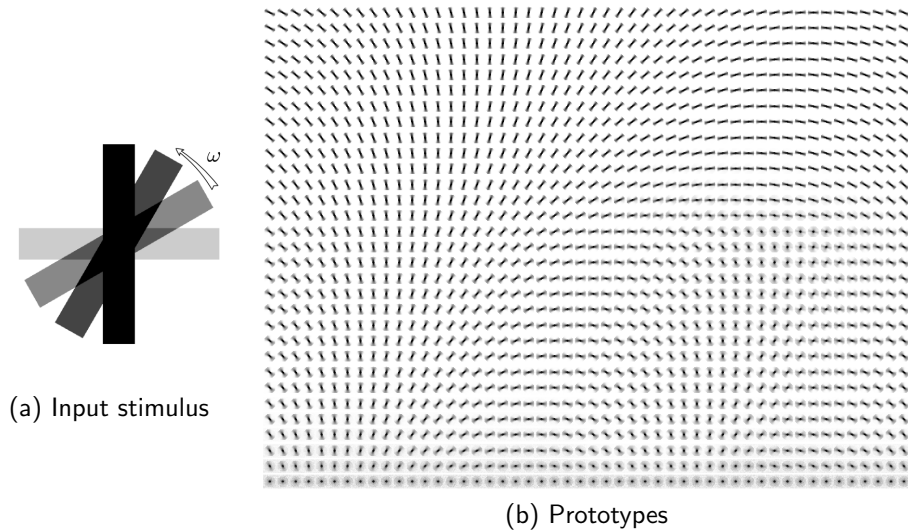


**Fig. 3.** Simulation of the self-organizing neural fields given by Eq. 4. The samples presented to the network are drawn from a ring distribution shown in gray. The white dots represent the prototypes and are interconnected following the 1D circular topology of the neural field. The current input  $I$  (dashed line) and output firing rates  $f(V)$  (solid line) are shown below each plot of the prototypes. The figures show the state of the neural field at initialization (a), after presenting 100 samples (b) and 400 samples (c). The current sample is shown as the black dot. A new sample is presented every  $2s$ .

For the simulation shown on Fig. 3, the samples presented to the network are drawn from a ring distribution shown as the shaded area on the figure. This distribution allows to check two important properties. In case the neural field is not able to create a decision, the prototypes will remain unchanged as the output firing rates  $f(V)$  gate learning of the prototypes. In case the neural field is not able to reconsider its decision, the prototypes will converge to the center of mass of the distribution  $(0, 0)$ . Indeed, the field would create a bump of activity and,



if we suppose the decision is stuck to that position, it will always be the same prototypes that will learn whatever the input sample. These prototypes would then move to the center of mass of the distribution until they become so far away from the distribution that the associated input becomes weak enough to allow an other position in the field to win the competition. Then the same process would apply and, step by step, all the prototypes would converge to the center of mass of the distribution. In the simulation, we observe that the prototypes correctly self-organize.



**Fig. 4.** (a) Experimental setup: a bar rotating at a constant angular speed  $\omega = 10^\circ\text{s}^{-1}$  is drawn on a patch of  $100 \times 100$  pixels. Such a patch is the input feeding all the positions of the neural field. (b) The  $N = 50$  prototypes at one given time are displayed on a row. We display the prototypes after each of the 30 full turns from bottom (initial prototypes) to top (final prototypes).

As a second example, we now consider a continuous stream of inputs. The point of this simulation is to demonstrate the ability of the neural field to self-organize from a continuous stream of inputs without requiring an explicit reset signal. Indeed, we present to the network a small patch of  $100 \times 100$  pixels on which a rotating bar is drawn. The bar is rotating at a constant angular speed  $\omega = 10^\circ\text{s}^{-1}$ . The bar is performing 30 full turns. Examples of input shown to the neural field are displayed on Fig. 4a. The bar is always with the same contrast. On the figure, the gray levels have been adjusted for representing four consecutive stimuli. We use exactly the same neural field equations (4) than for the previous simulation. The prototypes  $p(x, t)$  and input  $\zeta$  are now patches of  $100 \times 100$

pixels. All the parameters (time constants, learning rates, weights) are the same except the variance of the input set to  $\sigma = 50$ . The prototypes are initialized to the mean of 50 stimuli spanning the range  $0^\circ$  to  $180^\circ$  to which is added a uniform random noise sampled from  $[-0.1, 0.1]$ . We come back to the issue of initializing the prototypes in the discussion. As the neural field is evaluated synchronously, the random noise to the initial prototypes helps in breaking the symmetry which would lead to oscillating activities within the field. On Fig 4b we show the prototypes at the 50 spatial positions after each full turn of the bar. The initial prototypes are shown on the bottom row and the final prototypes on the top row. Initially all the prototypes are almost identical. As learning goes on, the prototypes finally span uniformly the range of orientations from  $0^\circ$  to  $180^\circ$ .

## Discussion

The dynamic neural field equation we proposed for self-organizing maps in section 3.1 suffers from some limitations. One appears when the prototypes are initialized far from the unknown distribution from which the samples are drawn. This issue is linked to the fact that the neural field requires to be driven by some activities that should not be too small. Otherwise, no decision will emerge in the field. Also, in case some of the prototypes would be able to come closer to the samples, one region of the neural field would learn but not the others. The learning region would learn a mean prototype and the others would remain unchanged. A second limitation is a boundary effect when the neural field is no more laterally connected with cyclic connections. If the connections are no more cyclic, the positions on the border of the *map* are less excited and more inhibited than the positions in the middle of the map. This brings a penalty for those units on the border preventing them from adapting their prototypes. A common mechanism might solve these two issues. A constant baseline added to one of the layers might be sufficient for solving the first issue but not the second. Indeed, the boundary effect appears because of a lack of excitation or excess of inhibition on the units on the border of the map. The bonus must not be uniform in space. An alternative relies in homeostatic plasticity[10], for example by adjusting the baseline of the units depending on the difference between the mean of recent firing rates and a target firing rate. Units that never win the competition (because their prototypes are far from the samples or because these are locations on the border of the map) would receive a transient self-excitation. This transient excitation would push them to fire for everything. At some point they would win the competition and come into play for adapting their prototypes. Such an idea has already been explored in [8] where the authors used a homeostatic learning rule to modulate the lateral connections of a neural field allowing to recover activities at sites where the input has been impaired, and in [3] where the firing threshold of the units is adjusted to bring it closer to a target firing rate.

The main point for the simulations of the self-organizing dynamic neural field is that it can learn continuously; there is no separate phase for evaluating the field until convergence and for learning the prototypes. The second important property is that the neural field is able to automatically reconsider its decision and there is no need for indicating it that a new sample is presented for, for example, resetting the activities. These properties allow the neural field to learn from a continuous stream of stimuli. The dynamic neural field equation we propose can be modified to remove any dependency on space in the interactions. If the gaussian  $g$  is replaced by a Kronecker, the weight function replaced by a self-excitation and uniform inhibition from the other locations, the network is able to perform a Winner-Takes-All. One can actually demonstrate that such a network has a single fixed point corresponding to the output of a Winner-Takes-All if the transfer function is taken to be a Heaviside step function. In that respect, instead of a dynamic self-organizing map, we can obtain a dynamic k-means.

## References

1. Alecu, L., Frezza-Buet, H., Alexandre, F.: Can self-organization emerge through dynamic neural fields computation? . *Connection Science* **23**(1) (2011) 1–31
2. Amari, S.: Dynamics of Pattern Formation in Lateral-Inhibition Type Neural Fields. *Biological Cybernetics* **27** (1977) 77–87
3. Bednar, J.A.: Building a mechanistic model of the development and function of the primary visual cortex. *Journal of Physiology-Paris* **106**(56) (2012) 194 – 211
4. Detorakis, G., Rougier, N.: A neural field model of the somatosensory cortex: formation, maintenance and reorganization of ordered topographic maps. *PLoS One* **7**(7) (2012) e40257
5. Fix, J.: Python source scripts for generating the illustrations. [http://jeremy.fix.free.fr/Simulations/dynamic\\_som.html](http://jeremy.fix.free.fr/Simulations/dynamic_som.html) (2013) [Online; accessed 5-November-2013].
6. Fix, J.: Template based black-box optimization of dynamic neural fields. *Neural Networks* **46**(0) (2013) 40 – 49
7. Kohonen, T.: Self-organized formation of topologically correct feature maps. *Biological Cybernetics* **43**(1) (1982) 59–69
8. Moldakarimov, S.B., McClelland, J.L., Ermentrout, G.B.: A homeostatic rule for inhibitory synapses promotes temporal sharpening and cortical reorganization. *Proceedings of the National Academy of Sciences* **103**(44) (2006) 16526–16531
9. Pfeifer, R., Bongard, J.C.: *How the Body Shapes the Way We Think: A New View of Intelligence* (Bradford Books). The MIT Press (2006)
10. Turrigiano, G.G.: Homeostatic plasticity in neuronal networks: the more things change, the more they stay the same. *Trends in Neurosciences* **22**(5) (1999) 221 – 227
11. Wilson, H.R., Cowan, J.D.: A Mathematical Theory of the Functional Dynamics of Cortical and Thalamic Nervous Tissue. *Kybernetik* **13** (1973) 55–80