Can Self-Organization Emerge through Dynamic Neural Fields Computation?

Lucian Alecu, Hervé Frezza-Buet, Frédéric Alexandre

Can self-organization emerge through dynamic neural fields computation?

Journal:	*Connection Science*
Manuscript ID:	Draft
Manuscript Type:	Original Article
Date Submitted by the Author:	
Complete List of Authors:	Alecu, Lucian; SUPELEC, IMS Frezza-Buet, Hervé; SUPELEC, IMS Alexandre, Frédéric; LORIA, CORTEX
Keywords:	dynamic neural fields, self-organizing maps, cognitive systems

Can self-organization emerge through dynamic neural fields computation?

Lucian Alecu[a,b] [*] and Hervé Frezza-Buet[a] and Frédéric Alexandre[b]

[a]IMS, SUPELEC, 2 rue Edouard Belin, 57070 Metz, France; [b]CORTEX, LORIA - INRIA Nancy, BP 70239, 54506 Vandoeuvre-lès-Nancy, France

In this paper, dynamic neural fields are used to develop key features of a cortically-inspired computational module. Under the perspective of designing computational systems that can exhibit the flexibility of the cortical substrate, using neural field as the competition layer for self-organizing modules has to be considered. However, despite the fact that they are biologically-inspired models, using dynamic neural fields to drive self-organization is not straightforward. To address this problem, an original method for evaluating neural field equations is proposed, based on statistical measurements of the field behavior in some scenarios. Limitations of classical neural field equations are then quantified, and an original field equation is proposed to overcome these difficulties, while preserving other features. The performance of the proposed field is discussed in comparison with some previously considered fields, leading to the promotion of that field as a suitable mean for processing competition in cortex-like computation for cognitive systems.

Keywords: dynamic neural fields, self-organization, cognitive systems

1 Computationally-oriented cortical modeling

The motivation of the work presented in this paper originates in previous attempts to derive computational paradigms from the biology of the human cortex [3, 20, 32]. From a Computer Science point of view the cortex appears as an information processing system with powerful computational features. These features, like distributed and robust computation, self-organization, recruitment and reallocation of cortical tissue to adapt to some behavioral change, are familiar to neuroscientists, but are barely exploited in software engineering, whereas they could offer robust and original computational paradigms. Indeed, one important challenge in Computational Neuroscience is to identify and translate the computational logic of the highly adaptive biological structures, as the cortex, into the field of Information Technology and Communication. The work presented here intends to be a step forward in this direction.

The most attractive feature of the cortical structure to a computer scientist is certainly its plasticity. Despite the variety of information that is processed onto the cortical structure, like vision, audition, body representation, planning, language [8], the cortex exhibits an undeniable anatomical homogeneity, appearing as the tiling of a generic neural circuit called the microcolumn in the literature [5, 28, 36]. Our brain seems to handle this diversity and complexity of information processing in a very dynamic way, some cortical regions re-adapting their functional role as a response to dramatic body changes [11], sometimes in no more than a few hours [46]. This raises the question of the true nature of the cortical computation performed in such circuitry.

---

[*] Corresponding author. Email: Lucian.Alecu@Supelec.fr

This has often been described by analyzing the role played by some specific cortical areas in the processing of its corresponding modality. For example, in early studies of vision, cortical processing was depicted as Gabor wavelet image filtering [9], which is of course not the case for non visual areas. As computer scientists, our role in the understanding of the cortical functions is to recognize from the variety of such examples the expression of a common computational template that we can formalize as a model like in [2, 10]. Such a goal is also addressed in the field of biology itself, since functional similarities have been highlighted between areas dealing with different modalities [35]. It is therefore legitimate to ask what can actually this template be. Following the early proposal by Kohonen [29], we support the idea that a key feature of the cortical structure is the commonly called self-organizing property, i.e. the ability of the cortex to build dynamical topologically-preserving vector quantization mapping. Kohonen has made one attempt to describe this self-organizing paradigm in formal terms, expressing his belief that this conceptual mechanism governs indeed not only the formation of feature maps (e.g. visual orientation map, auditory map, somatosensory map), but also that of associative maps or even higher-level cognitive mechanisms. However, even if Kohonen proposes different implementations of the concept, none of these is capable of encompassing all the computational features that could be expected in such a context. These issues, as well as our expectations towards the self-organizing paradigm are detailed in paragraph 2.1. On the other hand, in a previous work [32] we designed a computational model, using the `bijama` [18] cellular programming framework, exploiting this self-organizing property to yield the unsupervised specialization of some generic modules coupled with one another. From our point of view, distributed computations and the ability of developing multimodal representations are essential features that the implemented paradigm of self-organization should enable. Considering these aspects, the self-organization process gives rise to multimodal representations when solving a rewarded arm reaching problem [32].

These encouraging results motivates us to continue using self-organization for developing higher-level behavior, but the limitations of the Kohonen models compel us to investigate the low-level implementation of this mechanism. In particular, we examine in this paper the possibility of modeling the underlying competition taking place in the self-organizing process by the means of dynamic neural fields, a theoretical formalism of cortical competition within a population of neurons. Neural field models proposed in the literature rarely address this issue, the authors being rather interested in dynamical aspects, as stability and bifurcation studies. The present paper examines the ability of two of the most commonly used neural field models to implement a self-organizing procedure. Then, the paper exhibits the limitations of the considered fields to achieve reliable self-organizing behavior, and consequently proposes a novel equation that reconciles neural fields to the role we claim they should play in the self-organization process.

The paper is organized as follows. In section 2, we propose the generic template of cortically-inspired self-organization, where the role of neural fields can be formulated. Then, we show how this template encompasses in the same framework usual vector quantization techniques, as k-means or self-organizing feature maps, and more biologically oriented computational models. Moreover, this highlights the computational properties expected from a cortically plausible mechanism, as locality of the computation, tolerance to random initial states, etc., which are key aspects not addressed by the initial Kohonen approach.

Section 3 presents the dynamic neural fields formalism that will be used as the "computational engine" of the previously introduced self-organizing template. Next, three possible candidates of neural field equations for implementing this template are introduced. First, the early model by Amari [4], since it is still considered as a reference and used in many of the current approaches [15, 40], and second, the model introduced by Pinto and Ermentrout [37, 38], are presented. This latter model highlights the coupling of inhibitory and excitatory neurons within cortical columns, and also shares common features with the model that we introduce in this paper. This last neural field model is the novel one claiming to extend the capabilities of the other classical models in respect to the self-organizing property. The dynamic behavior of these three equations will be further empirically analyzed in section 6 in order to highlight their functional features.

In section 4, a performance measurement method to evaluate the dynamic evolution of a field is introduced. This allows to qualify at every time step the behavior of a given neural field dynamics by a scalar 'quality'. Further on, from this performance criterion, a so-called 'touchstone' neural field is introduced and analyzed. This study indicates that the proposed performance criterion is suitable to the cortical modeling template and this is a sufficient condition to achieve self organization.

In the next section, 5, we present the different scenarios used to collect comparative experimental results. As the central property investigated by our work concerns self-organizing abilities of a cortical model driven by a neural field, two scenarios are examined here. Apart from this main objective, some other properties of neural fields are also investigated. These are mainly the ones related to the use of neural field in attentional mechanisms. The purpose here is to determine whether the neural field that we propose is compatible with these features as well, or is exclusively dedicated to self-organization. We support the idea that this mechanism can be involved indeed in several aspects of cognitive functions.

The last section, 6, reports actual experiments from previously described scenarios. The experiments are driven by the following rationale. We check experimentally that the touchstone field is actually able to fulfill the role expected from a neural field in developing self-organizing behavior. As a consequence, testing whether a neural field performs well, in the sense of the criterion presented in section 4, is a relevant hint concerning its capability to operate cognitive functions. This check is provided for each scenario, for the three field equations considered in this paper. This allows to discuss these models in a comparative way, under the perspective of cortically-inspired mechanisms for cognition.

2 A modular view of self-organization with neural fields

The purpose of this paper is thus to propose a self-organization mechanism that relies on neural fields, as a generic cortically-inspired computational process. In this section, we briefly review the self-organizing concept, as introduced by Kohonen in [30], presenting then a generic computational architecture inspired by this paradigm.

2.1 *Cortically-inspired self-organization*

Basically, self-organization consists in extracting, from a high dimensional data stream, prototypes that are statistically significant and arranging them on a low dimensional output representational space. The usefulness of such a process is firstly

to have at hand a set of prototypes (the codebook) that can be used to replace the diversity of events in the data stream with a reduced but representative vocabulary and secondly to display these elements of vocabulary on a topological substratum, which reveals hidden relations and can be exploited for further processing.

As one wishes to implement such a process with a distributed paradigm, such as artificial neural networks, or to better understand how cortical maps of neurons emulate this function, it must be described in terms of an architecture of units, information flows and learning processes. As sketched out in (Figure 1), the three-layered architecture consists of a layer of units for the inputs, one for the recognition (as a result of the activation of prototypes) and one for the output. The main information flow is feedforward, from data stream to output representational space, through two steps, detection and selection, each implemented by a local circuitry, corresponding to recurrent local neighborhood connectivity. Detection consists in evaluating the matching level of each stored prototype with regard to the present input (it implements a distance function in the input space) and results in the recognition layer. Selection consists in electing, from these matching levels, a limited set of these prototypes, thanks to the operation of local excitation and surround inhibition.

In parallel to these functioning rules, a learning process can take place at each step. At the detection step, prototypes are iteratively elaborated in an unsupervised way. At the selection step, the strength of competition and the topological constraints can be also adapted from some activity dependent parameters. As a consequence of these operations, prototypes corresponding to similar inputs will be actually close in the output space. It is also clear that all these processes must be seen as dynamical processes and do not share the same time constants. Another important and critical factor is therefore mutual influences between these loops, particularly depending on the time constants.

This general definition for the distributed implementation of self-organization can now be confronted to several algorithms and processes. K-means proposes an original, simple and efficient algorithm for the detection module and particularly for the learning part, but in this context reduces the selection module to a simple WTA (Winner-Take-All). The same can be said for Kohonen SOM which adds a neighborhood constraint to the selection module. It is also interesting to underline that, in this picture, dynamic neural fields can be seen as a powerful way to implement a richer selection module with dynamical units, hence the central idea in this paper to refine classical self-organization algorithms with neural fields in the output layer. Coming back to the link to cortical functions, we claim (with many others, e.g. [10]) that the self-organization is among their properties. Information flows and architectural principles of the distributed implementation of self-organization mentioned above are consistent with physiological and anatomical data of the cortex [2]. As discussed further, other processing take place in the cortex and should be also considered in further works. Similarly, feedback, generally described as an important information flow dedicated to cortico-cortical communication, is not exploited in the architecture presented here and requires further development in order to be integrated. For example, [32] exploits feedback information flow to implement multimodality. Consequently, this study is clearly an initial step of cortical modeling, wishing to explore self-organization considered as its basic and perception-oriented function. It also underlines the interest of better exploring this function with realistic characteristics of space, time and granularity.

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

2.2 *Architecture of self-organization*

Let us consider a topological space X of the positions (i.e. each position $x \in X$ has some well defined *neighboring* positions x'). Space X refers to the place where some self-organizing process stands. For example, biologically-inspired cortical modeling may consider X as a bi-dimensional space, since the cortex is known to be organized as a 2D-laminar structure [36]. Let us define a field as a distribution of scalar activations over X, i.e. a mapping from X to R. Such a distribution is the actual data that is pipelined through the data flow presented next. As the architecture describes a dynamical system, all its varying components are indexed by time t, that is considered discrete here. Therefore, a scalar field $f(x), x \in X$ at time t is noted as f_t, and the instantaneous value of that field at position x is noted as $f_t(x)$. Some other components used in this architecture are prototypes $\omega(x), x \in X$, defined at each position x. Typically, in a neural model like SOM, a prototype $\omega(x)$ models the weight vector of the neural unit at place x. From these definitions, let us sketch the data flow, illustrated on (Figure 1).
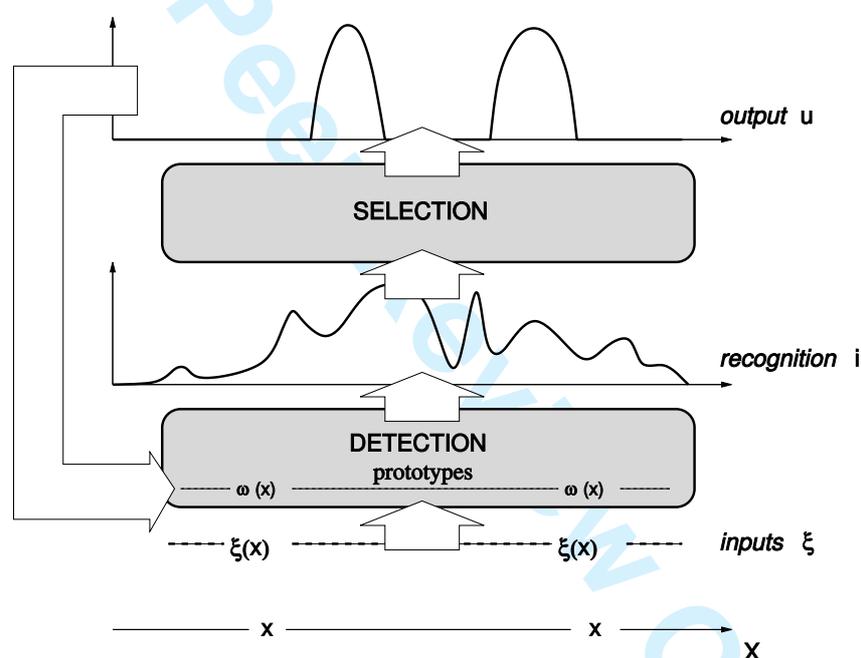

Figure 1. Information processing in a cortically-inspired self-organizing process. At each position x, an input $\xi(x)$ feeds the *detection* module. This module computes from all inputs a distribution of recognition activities $i(x)$ over X. Recognition feeds at its turn a *selection* module that has to elicit places x corresponding to local best recognition. To do so, the module produces a distribution $u(x)$ over the field X. This latter distribution modulates back the learning that occurs at the detection stage.

Let us call the first module the *detection* module. It is continuously fed with input vectors $\xi_t(x) \in \Xi$ at each position x. On those positions, some internal recognition procedure matches $\xi_t(x)$ against prototype $\omega_t(x)$. This produces a

recognition rate $i_t(x)$ for each place x, whose distribution i_t over the field is the output of the *detection* module. Usually, the matching produces high values $i_t(x)$ at places x where input $\xi_t(x)$ resembles the current prototype $\omega_t(x)$. A learning process runs continuously in parallel to the recognition, leading to slight changes in the prototype values. This learning *is modulated* by the overall output u_t of the system, detailed next. More precisely, the learning of prototype $\omega_t(x)$ depends on the current input $\xi_t(x)$ and the current value of $u_t(x)$.

Let us call the second module the *selection* module. It operates from the distribution i in order to compute another distribution u over X. The latter is high around *locally* best recognitions. This locality lays on the actual topology of X, which is 1D on (Figure 1), but which is rather 2D in more realistic models of the cortical sheet.

In this framework, external control is only given by successive changes of input distribution $\xi_t, \xi_{t+1}, \cdots$. This means that no reset of u is performed when new inputs are presented, and that learning rates are kept constant.

2.3 *A comparative view of self-organizing processes*

The above presented template is a high-level abstraction of a whole class of unsupervised learning techniques. In the following paragraphs we show how the instantiation of this conceptual template with different parameters brings out some classical learning paradigms. To illustrate this aspect, let us consider a general form of the computational procedures run by each of the two modules of the template.

First, we define the matching procedure of the detection module as most classical techniques do:

$$i_t(x) = f(d(\omega_t(x), \xi_t(x))) \tag{1}$$

where d is the a distance function of the metric space Ξ of the input samples $\xi_t(x)$ and f is a monotonically decreasing function. The designer of a biologically-oriented model may rather say that function $f_x(\xi) = f(d(\omega(x), \xi))$, from Ξ to R models the *tuning curve* of the neural unit at x.

Second, we define the general learning rule implemented by the selection module as:

$$\forall t, \forall x \in X, \omega_{t+1}(x) = \omega_t(x) + \alpha u_t(x)(\omega_t(x) - \xi_t(x)) \tag{2}$$

where $\alpha > 0$ is the learning rate and is here modulated by u.

2.3.1 K-means algorithm

Let us consider that all $\{\xi_t(x)\}_{x \in X}$ are identical ($\xi_t(x) = \xi_t^{\text{const}}$ for all positions). Let us set up the selection module so that it produces a single Dirac pulse as u_t output distribution, the Dirac pulse being placed at $\text{argmax}_{x \in X} i_t(x)$. In that case, the whole architecture acts as the on-line version of the k-means algorithm [31]. The Dirac pulse implements the winner-take-all (WTA) competition, or hard competition, used in such algorithms.

2.3.2 Self-organizing maps (SOM) algorithm

This is the same as previously, except that u_t is instead the convolution of the previous Dirac pulse with a bell-shaped even function (usually a Gaussian). In that case, the learning rule is efficient in the region in the field surrounding the best recognized prototype, since in this region $u_t(x)$ is significantly high. If the width of the bell decreases through time, which is not a necessary condition for the convergence, the whole system behaves exactly as a classical Self-Organizing-Map (SOM) [30]. In that case, the bell, rather called a *bump* in the literature, induces a winner-take-most (WTM) competition, or soft competition, that is known to reorganize the prototypes $\{\omega(x)\}_{x \in X}$ according to both the topology of the field X and the inner metric of Ξ induced by d (i.e. the so-called topology preservation) [30].

2.3.3 Self-organizing maps driven by neural field computation

Let us focus now on the architectural details of our proposal, i.e. the generic computational template of self-organization driven by neural fields. This template represents the starting point of our further analysis, concerning the feasibility of deploying self-organizing mechanisms through neural field computation.

The only structural difference from the previous model in section 2.3.2 is that here u_t is computed from i_t through a neural field equation, introduced further in section 3. This means that the selection module is made of an assembly of processing units, placed at positions $x \in X$, that update their scalar state $u_t(x)$ according to some differential equation. This update, for some unit x, depends on the state $u_t(x')$ of some other units in the module, accessible through a lateral on-center (w^+ neighborhood weighting kernel) off-surround (w^- neighborhood weighting kernel) connectivity. The distribution u_t built by the neural field equation at the selection module level is known as dynamically fitting i_t by emerging bumps of activity in the field (see section 3), after some transient relaxation time. With such an approach, this time delay raises some algorithmical problem. As it takes a *relaxation* time for u_t to set up the appropriate bumps when i_t changes (i.e. when ξ_t changes, since i_t is immediately computed from ξ_t), and as learning runs continuously, driven by u_t, in the detection module, transient distribution u_t may alter prototypes $\{\omega(x)\}_{x \in X}$. To cope with this effect in self-organization experiments, ξ_t is kept constant during many time steps (actually 100 for results presented further), and a small value of α (the learning rate) is used. Consequently, u_t reaches a new equilibrium state after a few time steps from each new change of the input sample ξ_t. Then, u_t remains constant, while learning can consolidate mainly from this stable state. The value of α is chosen such that the change of i_t induced by the continuous change of prototypes is small, keeping u_t actually stable. In our experiments, $\xi_t(x) = \xi_t^{\text{const}} \ \forall x$ is used, as in the previous SOM model.

In spite of an increase in complexity, compared with the SOM algorithm, this approach has some actually important advantages. First, if one enables the field to raise several bumps, learning can perform in parallel in several places (where the

actual bumps stand). This feature is not feasible in SOM, since the use of a maximum operator to select a single bump cannot be easily generalized to one corresponding to several bumps. Second, the nature of the SOM model is dramatically different from the one presented here, since SOM cannot be implemented as a fully distributed process, because the computation of the Dirac pulse position needs some kind of centralization. This centralization can be avoided with specific inhibition rules, as in [40]. Our work sets up SOM properties in the *distributed* case, using neural fields over a discrete space X for the selection module. The architecture layers, i.e. the stack of modules implementing the distributed processing pipeline, can thus be represented at the level of each position, which is more consistent with both biological plausibility and parallel implementation (see (Figure 2)). Evaluation of neural units is performed asynchronously, similar to Hopfield networks [22]. More precisely, at each time step, all positions are evaluated sequentially, in a random order [47]. The bijama [18] programming framework developed in our team, implements parallel execution of layered *cellular* networks as the one described here and illustrated in (Figure 2).
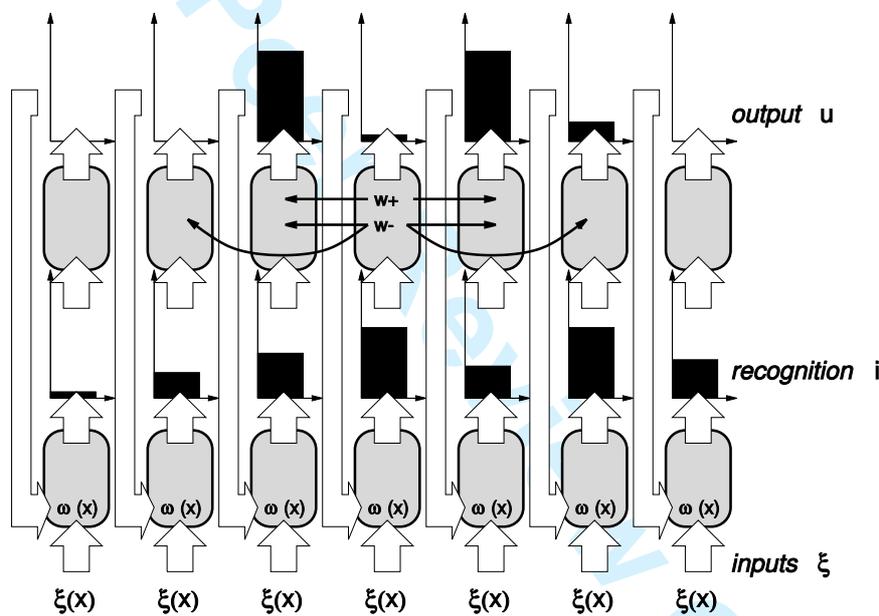

Figure 2. Cellular version of model in (Figure 1). Controlling this kind of distributed model motivates our study of the neural fields paradigm. See text for details.

2.3.4 Multimodal self-organization in bijama cellular model

In the bijama cellular model cited above, all positions do not receive the same input, since the input $\xi(x)$ is a combination of external inputs with position dependant afferences from other fields (their u values). This allows to couple self-organization between different fields to address multimodal problems. Current work on this model relies on enabling several u bumps in each field, leading within one field to several parallel learning processes. This work on the bijama model [32] is out of the scope of this paper, but it is among the motivations of the case study presented here.

2.3.5 Self-organization in RFLISSOM model

Some other significant modeling of cortical substrate can also be quoted here as fitting our proposed architecture. Let us mention here the RFLISSOM model, since it adds to the current architecture the adaptivity of the selection module, by allowing the learning of lateral connections w^+ and w^- (see (Figure 2)) [33]. RFLISSOM also implements the cellular computation paradigm. Even if the authors have to globally reset the neuronal activity after each learning step, which does not fit to our online local computational constraints, addressing learning in the selection module is challenging and original. Such investigations cannot be performed using a SOM model, which is another argument for the case study addressed by this paper.

3 Neural fields for cortical processing

3.1 *Computing with neural fields*

The dynamic neural fields (DNF) models aim to describe the dynamics of the electrical potential activities of populations of neuronal units (e.g. neurons or microcolumns) by the means of integro-differential equations. They were founded as an attempt to model the computational basis of the cortical information processing, and in particular the neural competition/cooperation in the lateral information flow.

Wilson and Cowan [54] and later Amari [4] established the foundations of dynamic neural fields as a mathematical formalism. At every time instance t, the evolution of the membrane potential, $u(x,t)$ for each neural unit x of the population (or field) X is described by an implicit equation. The neuronal units of the field are interconnected, and their lateral connections weights are fixed, typically defined as an on-center/off-surround influence (also known as Mexican-hat). This lateral connectivity for each unit x is usually defined through the form of a neighboring weighting kernel. The field conceived in this way responds to input distributions $i(x,t)$, by increasing its activity wherever the input stimuli are locally strong, emerging so-called neural "bumps". These responses may be regarded as a local *s*oft competition, an effect that may be seen as a basis for the development of more complex computational structures. One of the most commonly used neural fields equation in the domain's literature is the one proposed by Amari [4], presented in section 3.2.1 and further analyzed in the paper.

Since those early works, the DNF paradigm has been studied and extended, following two main directions. The first one, addressed next, is the use of DNF as a formal descriptive model of neural processing. The role of the model is then to bridge the gap between biological observations and mathematical analysis. Applying dynamical systems analysis to DNF then allows explaining qualitative observations of neural populations. The second direction, addressed after, is the use of DNF at another level, since their ability to achieve actual information processing is to be investigated. These two directions are complementary: while the first *analyzes* cortical computation, the second rather proposes *operational* mechanisms inspired from it.

In order to support the second approach, let us review here results from mathematical analysis, since they provide a formal grounding for some models that are very close to the original one proposed here.

Since DNF aim at modeling the dynamical behavior of the cortical tissue, mathematical analyses, that are often expressed in the mono-dimensional case for sake of simplicity, and have been extended to bi-dimensional fields, as it has been done in [48]. It shows that the actual shape of the on-center/off-surround kernel

influences the field dynamics. Conditions on lateral weight kernels have been found, for a constant input i (see (Figure 1) for notations), to have isolated bumps of u activity after the field stabilization, rather than constant u or Turing instabilities. In [48], a qualitative remark is also made about the dynamic of the reaching of these states when inputs change: the transition between two close stable states is performed by a slide of bumps from initial state to the final one. Indeed, the DNF models are able to do much more than placing stable bumps and such rich dynamics also motivate a deeper analysis of cortical neural population. Phenomena like wave front and traveling pulses, observed on the brain surface, actually appear in some richer models of DNF. First extension of DNF is the introduction of potential transmission delays between neurons [25, 51]. With a suitable formalization, bifurcation in such models to different dynamical modes can be related to parameter settings. Those modes are traveling fronts, traveling pulses, or Turing instabilities mentioned previously. The second way to extend initial DNF paradigm is to consider two types of neurons in the field. In those models [53], each excitatory neuron in the field is associated with an inhibitory neuron, so that the basic field element is an oscillator. It has been shown that a 1D field of such oscillators with only excitatory lateral connections can produce traveling activities [7], as fields with delays actually do. Bifurcations analysis and dynamics compatible with Hodgkin-Huxley-like membrane modeling has also been proposed [16], as well as the ability of such field to produce *breathers* (oscillatory activity patterns, able under some conditions to generate wave fronts periodically). More realistic on-center/off-surround lateral weight kernels also allow such dynamical behaviors [17]. It means that having a population of coupled excitatory and inhibitory neurons, as initially introduced by Wilson and Cowan [53], is a suitable extension of DNF for addressing richer dynamics, as those obtained by adding time delays. The link between the two may not be obvious. Nevertheless, in [12], a mathematical short-cut for simulating delays is proposed. It consists in transforming the equation with time delay into two coupled differential equations, which makes the delay model closer to a population of oscillators.

As argued in section 2.1, we are interested in developing self-organizing behavior through neural field competition. We try to limit the complexity (but not the generality) of the DNF model used in this purpose by not considering explicitly the fine modeling of neural activity temporal propagation in the lateral flow. Nevertheless, we consider models based on a population of oscillators, adapted from [37, 38]. Consequently, we limit our analysis to the DNF models described in the next section. Development of high-level behavior relies on rich local phenomena that the field exhibits. Someone trying to elaborate computational mechanisms based on these dynamics has to exploit their features, reactivity to input, intensity of response, spatial density, frequency of occurrence or other transient and stable regime parameters. However, controlling all these aspects is far from simple. The incompleteness of the results provided by the analytical studies in this respect, and the difficulty of finding appropriate fine-tuning methods for generating a certain dynamic behavior of the field, may explain the rather few proposals of computational models based on DNF in the literature. As a consequence, the research interest of the domain shifted towards previously mentioned theoretical aspects, neglecting somehow the applicative studies of DNF.

Nevertheless, few attempts exist to use DNF as a computational tool for applications, like robotics [13, 26]. This set can be enlarged if the works aiming at exhibiting some general computational properties of DNF are also considered. Actually, models of cognition, including self-organization, attention, decision,

implicitly contribute to promote the use of neural fields to control situated devices, as autonomous robots, even if the domain today is not mature enough to derive directly such applications from DNF. Let us present here such approaches, some being closer to biology and cognition, some others rather focused on actually using the neural field, but all sharing the common goal of promoting DNF as a powerful computational paradigm, inspired from the cortical activity.

One computational feature of DNF that is highlighted by authors is their ability to take a dynamical decision from a distribution of activities. This is exactly what is expected from the selection module in (Figure 2). In that context, input i is a complex changing profile from which the "most relevant" part has to be continuously identified, by permanently updating the position of an activity bump u. As mentioned in section 2.3.3, this can be used to implement distributed self-organization. This has been done previously for oriented bars organization as in V1 [14], where input ξ_t (see notation of (Figure 1)) are restricted to varying continuously with time. The selection ability can also be used in attention, which differs from the self-organizing context. In this case, i is the response of a saliency map, and the u bump robustly emerges at the place of the appropriate focus of attention [39, 40, 43]. A similar mechanism can also be used for action selection [52]. For those purposes, a single bump may be desired, and can be ensured [34], but having several bumps in some fields may also help to keep an internal memory of recently visited positions [15]. Such feature is also involved in architecture including several fields, coupled one with the others. In this case, activity u of a field is used to bias input i of some other fields, allowing the whole architecture to place a set of bumps to solve multi-values constraints, thus accounting for cognitive properties [15, 27, 45]. We investigate such a multi-field approach for learning controllers, adding reward and self-organization to these attentional mechanisms [32]. This is what actually motivates the current paper.

Another computational feature is the use of traveling pulses, whose stability has also been studied [23]. Some authors consider such waves as produced by cortical areas in abnormal conditions, as hallucinations [6], but traveling waves can also be obtained by breaking the symmetry of the lateral coupling weight kernel. This has been used for robotic arm control [26]. In this work, each position in the field corresponds to the position of the arm extremity, and obstacles are indicated by negative i that are bump repulsive. The lateral weights are modified explicitly so that the bump moves in the direction of the goal, avoiding obstacles since the field copes permanently with bump rising and negative input avoidance. More recently, this property has been used to anticipate the consequences of action in a reinforcement learning context [49, 50].

Last, some more specific properties are addressed. Let us mention the binding problem, i.e. associating distant bumps in a field of oscillators that are locked in phase [41, 42]. This promotes once again the dynamical richness of such fields of oscillators. This richness has also been exploited for visual selection in a model of superior colliculus [44]. As opposed to previously mentioned models for visual attention that were based on the Amari equation, the selection is rather performed here by neurons behaving in two modes, thanks to a non linearity in their transfer function. The first mode is a race for accumulating activation, and the second mode performs the selection. It can be noticed from this model that populations of oscillators, when actually used with complex pattern of input, do not generate traveling pulse anymore, as opposed to the cases where mathematical results are available. This remark is also valid in this paper since two of the three neural fields used here for comparison are

actual population of oscillators, but are engaged in attentional and self-organizing tasks without exhibiting wave activity.

In this paper the two main neural field principles reported above, i.e. the usual Amari field and the one by Ermentrout involving oscillators, are considered under the perspective of self-organization presented in section 2.3.3. As results presented further will show, they hardly fulfill all the properties expected here from them. This has led us to propose an original neural field equation, which is the third field equation actually considered in experiments.

3.2 *Investigated neural fields*

We detail here the neural field models that will be further investigated in the paper, in our attempt to design a self-organizing generic mechanism based on DNF computation.
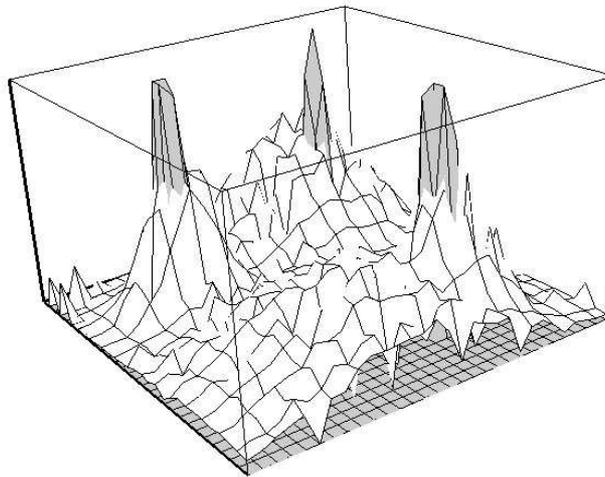

Figure 3. The stabilized response $u(x,t)$ (gray surface) of a dynamic neural field to a given input distribution $i(x,t)$ (white surface). Simulation provided by the `bijama` software library [18].

3.2.1 *Amari neural field (ANF)*

Besides being the main founder of the theoretical formulation of dynamic neural fields, the Amari field (further referred to as ANF) is still used in many neural field modeling approaches. The classical field equation [4] is given below:

$$\frac{1}{\tau}\frac{\mathrm{d}u(x,t)}{\mathrm{d}t} = -u(x,t) + \int_{x'} w(|x-x'|) f(u(x',t))\mathrm{d}x' + i(x,t) + h \tag{3}$$

where f is a non-linear function (usually a sigmoid), τ and h are real constants, and w is the lateral connections weighting kernel, usually a Mexican-hat function as below:

$$w(r) = A^{+}\mathrm{e}^{-ar^2} - A^{-}\mathrm{e}^{-br^2} \tag{4}$$

(Figure 3) plots a distribution of input stimuli, as well as the corresponding stabilized response of a two-dimensional field driven by the Amari equation.

3.2.2 Pinto and Ermentrout neural field (PENF)

The second neural field model considered in our analysis is an extension of the Amari model, proposed by Pinto and Ermentrout [37, 38] (and further referred to as PENF). The equations describing the temporal evolution of the neural units are given below:

$$\frac{1}{\tau}\frac{du(x,t)}{dt} = -u(x,t) + i(x,t) + \int_{x'} w(|x-x'|)f(u(x',t))dx' - \beta v(x,t) \tag{5}$$

$$\frac{1}{\eta}\frac{dv(x,t)}{dt} = -v(x,t) + u(x,t) \tag{6}$$

All the parameters have the same meaning as in the Amari equation 3, except the additional $\beta \in [0,1]$ and v, the latter being an auxiliary distribution over the population X. The authors suggest that v is a slow, negative feedback factor of the field's response u, that could be seen as "a spike frequency adaptation, synaptic depression or some other slow process that limits the excitation of the network" [37]. In certain conditions, the field is capable of generating sustainable traveling pulses, but this effect will not be investigated in our further analysis, as already mentioned.

3.2.3 Back inhibited neural populations (BINP)

Inspired by the Pinto and Ermentrout model, which enhances the behavior of a field through feedback contributions, we propose a new model that also introduces a feedback term to better control the dynamics of the field. As it will be shown by the further analysis, this term, suggestively called *back inhibition*, influences the field, in order to enable an enhanced bump formation and adaptability to various input stimuli. The mathematical formalism describing the new field behavior is given by the following equations:

$$1/\tau \cdot du(x,t)/dt = i(x,t) + \alpha \cdot E(x,t) - \beta \cdot I(x,t) - \gamma \cdot g(i,v) \tag{7}$$

$$dv(x,t)/dt = h(E(x,t)) \tag{8}$$

The main feature of this model is that it succeeds in enabling the formation of high saturated bumps in all input conditions. Besides, the formed bumps can easily follow the changes occurring in the input stimuli distribution.

Let us first define w^+, the on-center neighborhood weighting kernel, and w^-, the off-surround neighborhood weighting kernel:

$$w^+(r) = e^{-ar^2} \tag{9}$$

$$w^-(r) = e^{-br^2} - e^{-cr^2} \tag{10}$$

As in biological systems, that report separate excitatory and inhibitory neurons, we also consider a separation between these influences. Here we use $b < c$ in order to keep both kernels positive, as shown in (Figure 4).

Let us further define respectively the lateral excitation and lateral inhibition, experienced by each unit x in the field:

$$E(x,t) = \sigma_1\left(\int_{x'} w^+(|x-x'|)f(u(x',t))dx'\right) \tag{11}$$

$$I(x,t) = \sigma_2 \left(\int_{x'} w^-(|x-x'|) f(u(x',t)) dx' \right) \qquad (12)$$

where σ_1, σ_2 and f are sigmoid functions (cf. annex A).

A supplementary process automatically keeps u and v positive and saturates the values $u(x)$ and $v(x)$ to 1, whenever they exceed this threshold. α, β, γ and τ are real constants. g is a product of two sigmoid functions, while h is a monotone function, as represented in (Figure 5) and made explicit in annex A.
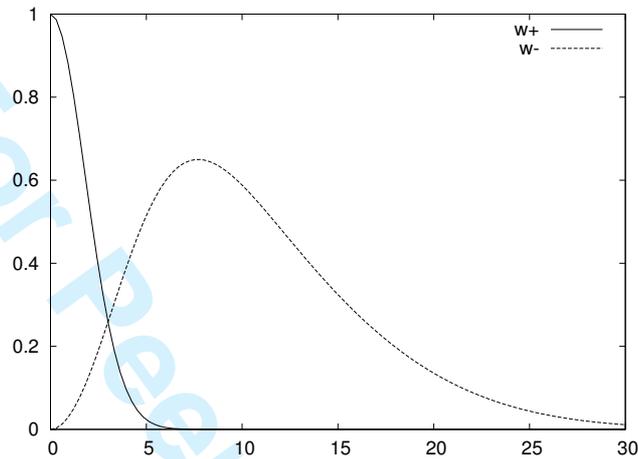

Figure 4. Excitatory and inhibitory weighting kernel functions.
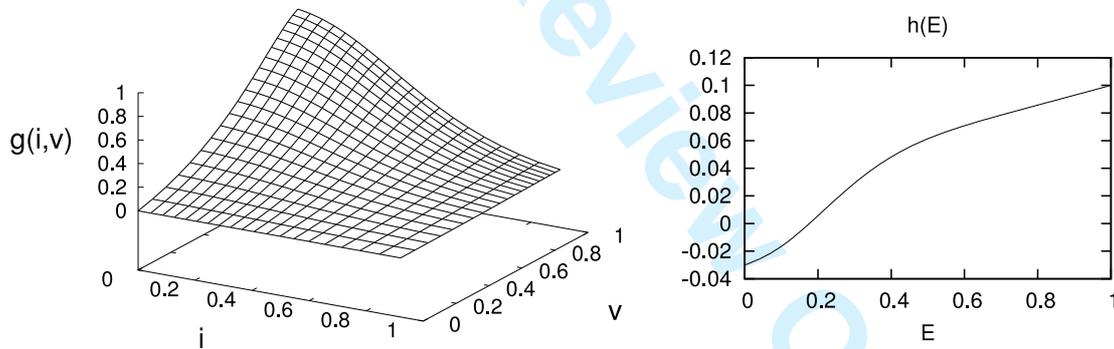See annex for the values used in our simulations.

Figure 5. The g (left) and h (right) functions used in equations
7 and 8. See annex for the values used in our simulations.

The novelty of these equations resides in the g term, our proposed mechanism for implementing an adaptive back-inhibition for all the field regions highly excited in low stimuli conditions. More specifically, wherever the field responds strongly to low inputs (i.e. $i \approx 0$, $u \approx 1$), it will be forced, within a certain delay, to decrease its response u to zero, through the inhibition induced by the increasing values of g, since v increases while u is high. This is why such a field is further referred to as a *Back Inhibited Neural Population* (BINP).

4   Qualifying the behavior of dynamic neural fields

In this section a criterion is proposed to evaluate the ability of a neural field to participate in a self-organizing process, as previously attempted in [21]. This criterion is formalized in section 4.1, legitimated in section 4.2 and applied in section 6 as a mean to compare neural field models.

The criterion presented here aims to measure two properties that we are expecting from a stabilized field response. First, locally strongest stimuli should be represented by bumps with a high amplitude, independent of the actual intensity of the stimuli. Second, at any given time instance, these bumps should be sparsely distributed along the field.

In particular, if the sparseness principle is formulated as to allow for the formation of a single bump along the entire field, the resulting field response can be interpreted as a distributed computation of a global maximum. As a consequence, the template described in section 2.3.3 that would use such a field to implement its selection module, would intuitively behave as a classical SOM algorithm. An empirical investigation of this assertion is presented in section 4.2, its conclusions making us consider such principles legitimate for supporting the development of self-organizing properties.

Inspired by [34], we propose here some illustrations of fields satisfying these principles. Figure 6 describes visually the expected responses of such a neural field performing in various key scenarios. For visual simplicity, we consider here only the one-dimensional X space, but the principles presented here can be extrapolated to a multidimensional X. Additionally, for the same practical reasons, we limit our discussion to the vicinity of a single bump, as we are interested in highlighting the local selective behavior of the field. All the cases may be immediately generalized to fields able to raise multiple bumps simultaneously. As we will see, the definition of the quality criterion that we will further formulate is independent of the dimension of X or number of bumps.

As a general remark, the fields respond, or adapt their responses so as to match locally strongest stimuli (the distribution i_t computed by the detection module of the architecture described in (Figure 1)). Their behavior should be robust, the fields not being distracted by isolated peaks of input stimuli (see (Figure 6. d, e)).

Let us highlight here the case of null input scenario presented in (Figure 6. h). The fact that the input is absent should not suppress the selectivity character of the field. Non-intuitively, the responses of the field in the (c) and (h) cases from (Figure 6) are qualitatively equivalent: the field does not cease selecting (i.e. form high bumps) even in conditions of uniform level of input stimuli. In this respect, this field does not make a distinction between scenarios of different levels of input stimuli. Its responses are selective and well-saturated in any input conditions.

Let us now express our insight on how fields with such a behavior can be used to participate in self-organization. We recall that the field response, as assumed in section 2.3.3, will be used to modulate the learning rule controlling the adjustments of the prototypes (see equation 2). The prototypes match usually very poorly at the beginning of the learning process, and only after the procedure converges, they increase their matching response. Consequently, the input stimuli received by the neural field from the selection module of the self-organizing template will be very low at the beginning, and only afterwards will increase. Therefore, in order to correctly modulate learning, the field has to enable the emergence of high responses regardless of the level of input stimuli. In this sense, the particular selectivity of the

field, defined by the first principle stated here, is suitable for self-organization. The illustration of this argument is presented in section 4.2, where we show how the template from section 2.3.3, using as the selection module a neural field satisfying the two principles described here, can indeed achieve self-organization.
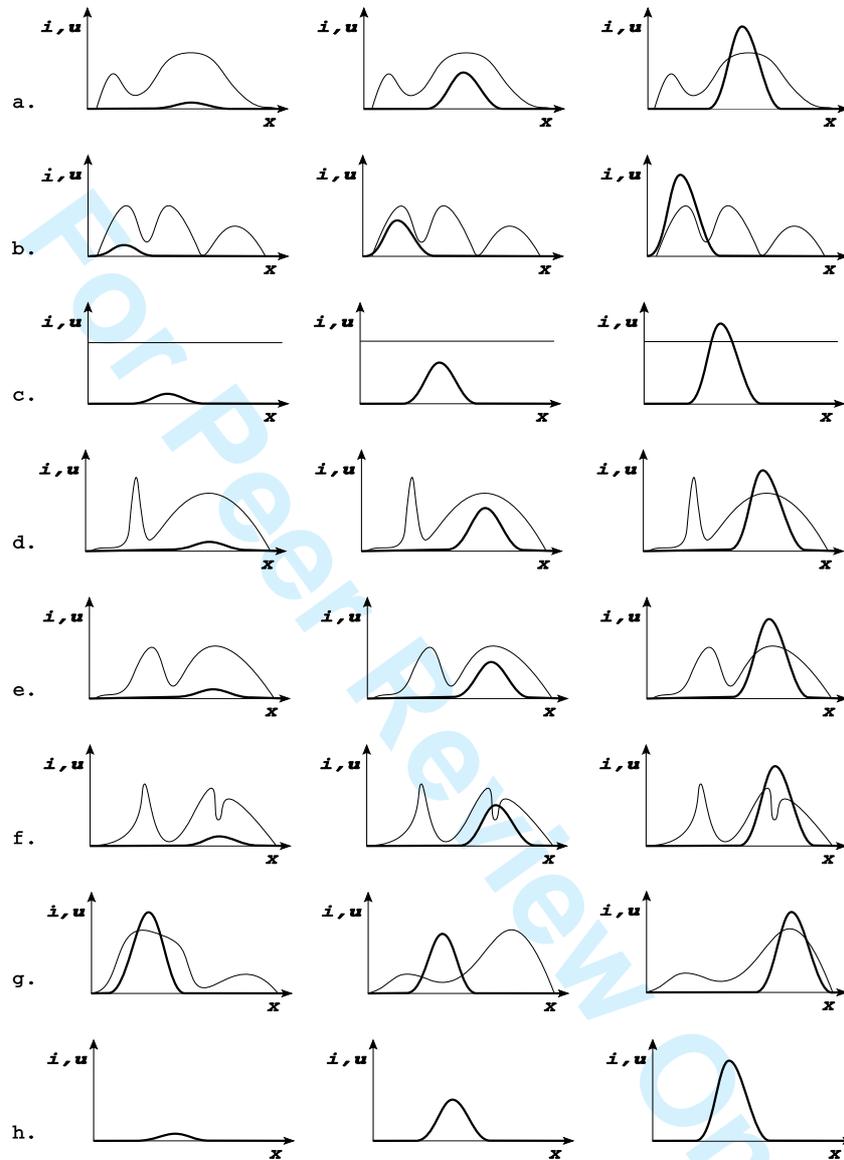

Figure 6. Some expected effects (an extension from [34]) of a one-dimensional neural field activity (the thick plot), in response to various inputs (the thin plot). For each row, the temporal evolution of the field is shown from left to right. The field is assumed to generate a single bump of activity. a. selectivity, b. competition, c. exploration (high plateau), d. competition (strong distractor), e. competition (weak distractor), f. collaboration, g. adaptation, h. exploration (low plateau).

Following the above considerations, let us formulate the properties (P) and (Q) corresponding to the two principles. (P) stands for the field selectivity principle, and (Q) for the bump density constraint.

Property P *Bumps of a stabilized field response emerge in regions where the input stimuli are locally the strongest and their amplitudes do not depend on the amplitudes of the input stimuli.*

Property Q *The distance between the center positions of any two bumps of a stabilized field response must be kept between two bounding limits (b_{min} and b_{max}, with $b_{min} < b_{max}$, as in (Figure 7)), such that the bumps are neither too sparse, nor too dense in the field.*

From now we will be guided by the desiderata of finding neural fields that fulfill the two properties (P $\wedge$ Q). A specific numerical instrument to capture and measure how precisely a neural field satisfies these properties along its evolution is described in the next section.
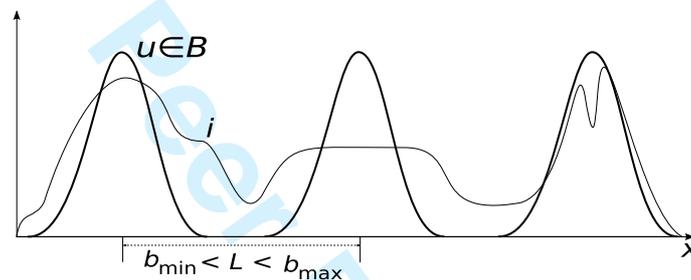

Figure 7. An example of a one-dimensional stabilized response
of a field satisfying (P $\wedge$ Q). See text for details.

4.1 *Evaluation of neural field computational capabilities*
In this section, we express (P $\wedge$ Q) into equations, describing also a method to qualify the dynamics of a field with respect to these principles. The conceived instrument will then be used as a tool to investigate whether a specific field satisfies the (P $\wedge$ Q) properties in a particular benchmarking scenario.

While this approach does not provide a direct solution to the parameter choice problem, one can quantitatively evaluate any field dynamics with respect to (P $\wedge$ Q) and thus validate a specific parameter choice [1]. Additionally, based on the actual measuring instrument, an optimization procedure may be implemented for studying the impact of the variation of one or more parameter values upon the field behavior. However, this procedure has a very high computational cost.

4.1.1 Measure of performance
The following analysis is performed for any time instance t. However, for the sake of simplicity, the notations used here do not carry the t index, as time does not appear explicitly in any of the equations below.

Let us consider an input distribution $\{i(x)\}_{x \in X}$ over the field X, and the field response distribution $\{u(x)\}_{x \in X}$. Let us also note R^X the set of functions from X to R. Both $u(x)$ and $i(x)$ belong to R^X. The Euclidian distance d on this set is given by equation 13.

$$a \in \mathsf{R}^X, b \in \mathsf{R}^X, d(a,b) = \sqrt{\int_{x \in X}(a(x)-b(x))^2} \tag{13}$$

The (Q) property states that the distance between any two bumps of the field has to lay in the interval $(b_{\min}, b_{\max})$. Thereby, let us define $B \subset \mathsf{R}^X$ as the set of all possible field distributions obtained by placing bumps throughout the field as to satisfy the (Q) condition. For example, (Figure 7) shows one element of the B set in such a case. In particular, if b_{min} is very large ($b_{min} \to \infty$), B is the set of distributions formed by a single bump placed at position x, for whichever $x \in X$.

In order that a given distribution u satisfies the two properties, (P $\wedge$ Q), by definition it should be an element of B. Therefore, the distance from u to B (i.e. $d(u,B) = \min_{u' \in B} d(u,u')$) should be ideally zero, or practically as small as possible. On the other hand, the (P) property of the fields states that $u(x)$ should be high when $i(x)$ is locally the strongest, thus the two distributions should be correlated. This implies that the distance $d(u,i)$ should also be as small as possible. It may be impossible to have $d(i,u) = 0$, since u should belong to B and i may not, thus $d(i,u)$ should be reduced to $d(i,B)$. Unless i itself is an element of B (i.e. $d(i,B) = 0$), fulfilling the two conditions (P $\wedge$ Q) implies satisfying two opposite constraints, i.e. minimizing two different interrelated distances at the same time.

Let us define $\Delta_i^B(u)$ as the residual error of u minimizing the two above distances, a measure of performance of u with regard to i and B:

$$\Delta_i^B(u) = \sqrt{(d(i,u)-d(i,B))^2 + d^2(u,B)} \tag{14}$$

In addition, we define a normalized version of $\Delta_i^B(u)$ as below:

$$\overline{\Delta_i^B}(u) = \Delta_i^B(u)/d(0,B) \tag{15}$$

From these definitions follows immediately that the touchstone fields always have $\overline{\Delta_i^B}(u) = \Delta_i^B(u) = 0$.

To illustrate the intuition behind the introduction of $\Delta_i^B(u)$, let us outline here a geometrical interpretation of this performance measuring instrument. In (Figure 8), we represent R^X (the set of functions from X to R) as the bi-dimensional Euclidian plane. In this plane, each point stands for a particular distribution of activity over the field, and the Euclidian distance between two points represents the actual distance d defined in equation 13.

Let us consider $B_i = \{i' \in B, d(i,i') = \min_{u' \in B} d(i,u') = d(i,B)\}$ the projection of i onto B, the set of all the field responses satisfying (P $\wedge$ Q) in respect to i. The proximity of u to one of the elements of B_i is upper bounded by the value of $\Delta_i^B(u)$. Indeed, in order for u to approach a good field behavior, both quadratic terms in equation 14 (also depicted in (Figure 8)) have to be small. However, minimizing one of the terms of $\Delta_i^B(u)$ may trigger the increase of the other one and vice versa, as both depend on i. Therefore, the minimization of $\Delta_i^B(u)$ is not a straightforward procedure. Finally, we can state that near zero values of $\Delta_i^B(u)$ indicate a good field behavior, well satisfying the (P $\wedge$ Q) properties.
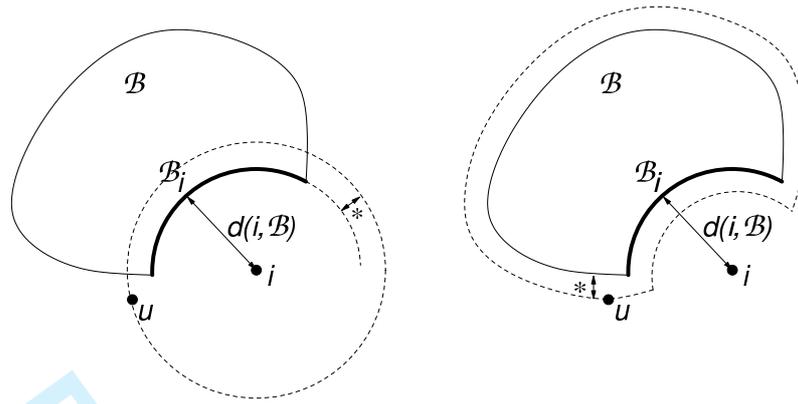
Figure 8. Geometrical interpretation of $\Delta_i^B(u)$. We remind that i is the input and u is the resulting field, while B is the set of fields satisfying the Q condition. The thick line represents the set B_i of field responses satisfying (P $\wedge$ Q) in respect to i, and it indicates a quality reference. u is nearly as qualitative if both following distances marked with the symbol * in the figure are small: $(d(i,u)-d(i,B))$ (see the left side of the figure) and $d(u,B)$ (see the right side of the figure). (P $\wedge$ Q) is the intersection of the two dotted lined regions in the figure. The proximity of u to B_i increases while this intersection shrinks around B_i, i.e. while $\Delta_i^B(u)$ reduces towards zero.

4.1.2 Performance computation and touchstone fields

The introduced performance criterion of the field selectivity is based on distances from a distribution (i.e. u or i) to a set of distributions (i.e. B). Computing such distances implies finding among the elements of B a distribution that minimizes the distance to u or i. This searching procedure is described in the current section, since it represents the core mechanism for computing the $\Delta_i^B(u)$ value.

Whereas B is a continuous set, computing a projection to it (e.g. $d(u,B)$) can be performed in a finite time when this set is defined as a parametrized functional set. The density condition (Q) asserted in the previous section allows such parameterization. In addition, if we consider a fixed shape for an elementary bump of any touchstone field response (e.g. we may choose for this a particular bell shape), we can search for any element $b \in B$ in a finite time. In such case, the set of parameters completely defining B is given by the coordinates of the centers of such bumps. Let us therefore note $x \in X$ a position in the field, and $\theta = \{x_1, \cdots, x_p\}$ a set of p bump center positions. Let Θ be the set of all possible θ. Some $\theta \in \Theta$ defines an element $b_\theta \in B$ only when the positions x_i satisfy the density condition (Q) of B. The set Θ can then be discretized, by considering discretized positions, and then explored in a combinatorial way. The constraints imposed by the condition of bumps density within X prune dramatically the research in this discrete set of parameters. Namely, it restricts the parameters values to fewer discrete valid positions. The set of these discrete positions which satisfy property (P) is noted $\hat{\Theta}$ in the following.

With the previous notations, let us show how $\hat{\Theta}$ is used to compute the touchstone field response, i.e. to find $b \in B$ that minimizes $d(i,b)$. This can actually

be done by considering all $d(i, b_{\hat{\theta}}), \hat{\theta} \in \hat{\Theta}$. Indeed, for each $b_{\hat{\theta}}$, an exploration is started, driven by a gradient descent according to $\overleftarrow{\nabla}_{\theta}(d(i, b_{\theta}))$ from $\hat{\theta}$. This is done in a stochastic way, by modifying randomly some components of current θ, and applying only modification that leads to an improvement of the objective (here a smaller $d(i, b_{\theta})$). In this manner the search runs from $\hat{\theta}$ until θ is stable, i.e. θ is a local optimum.

At the end of such a process started for all $\hat{\theta} \in \hat{\Theta}$, the element of B that minimizes an expression is available, as well as the minimal value of this expression. This element b^* is the touchstone field response, for which $\Delta_i^B(b^*) = 0$. Such a mechanism is used only for computing the performance criterion, and is not a valid neural field process since it is not distributed. Nevertheless, in the next section, the optimal function found by this procedure, considered as a "touchstone field response", is used as the output of the selection module in (Figure 1), in order to justify the relevance of the $\Delta_i^B(u)$ criterion.

4.2 *The relevance of (P $\wedge$ Q) in a self-organizing context*

We present here the pragmatic arguments that entitle us to consider (P $\wedge$ Q) actually relevant for the goals that motivated our research. Consequently, we investigate here whether fields satisfying these properties may implement the selection module of the architecture described in section 2.3.3, in order to successfully solve a self-organizing task. Lacking formal instruments to analyze this aspect, we examine it through experimental means, comparing the outputs of classical SOM algorithm (see section 2.3.2) with those provided by the neural field implementation.

Let us consider the simple two-dimensional input domain $\Xi = \{\xi = (\xi_1, \xi_2) \mid 0.5 \leq \sqrt{\xi_1^2 + \xi_2^2} < 1\}$, plotted as the gray coronal shape shown in the two sub-figures of (Figure 9). Let also X be a discrete field with a ring topology. For implementation details, see further section 5.1.1 where the same scenario is used.

We are interested here in finding whether feeding with randomly distributed values $\xi \in \Xi$ the two architectures (a self-organizing architecture using the touchstone field as the selection module cf. 2.3.3, and a classical SOM cf. 2.3.2), may lead to similar results.
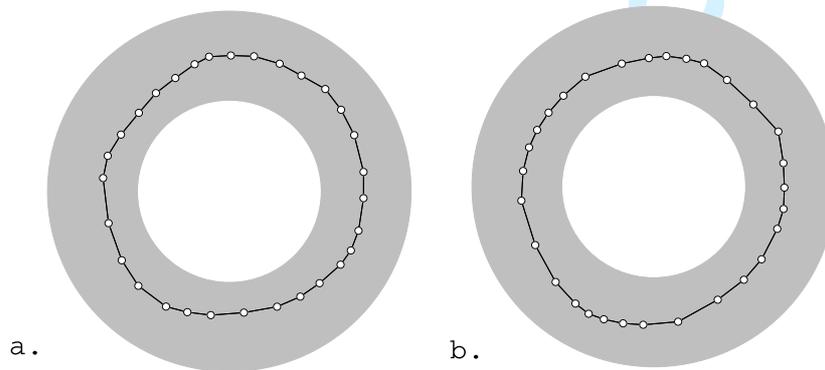

Figure 9. Using the self-organizing template from section 2 to solve a one-dimensional task. Results obtained when using as selection module the one described in 2.3.3 (figure a) and that described in 2.3.2, in which neural fields are ideally satisfying the (P $\wedge$ Q) properties (figure b).

(Figure 9) gathers results of the two solutions. In both cases, the prototypes $\{\omega(x)\}_{x \in X}$ of the detection module, randomly initialized, succeed in reorganizing well and finally describe a discrete one-dimensional representation of the coronal shape.

As observed, there is no qualitative difference between the results of the two implementations. Nevertheless, we remind that the distributions u of the touchstone neural fields are computed by the gradient descent procedure described in paragraph 4.1.2. They are, therefore, not the result of some local algorithm computation, but the results of the explicit combinatorial research procedure of the elements of B that minimize $d(i, B)$. Following these considerations, the touchstone fields cannot be used in practical applications, as this procedure is not distributed, and besides, is extremely costly in terms of computational complexity. The essential conclusion of these results is nevertheless that *the satisfaction of the (P $\wedge$ Q) properties is a sufficient condition for a field to achieve self-organization*. This argument shows that (P $\wedge$ Q) may be considered as desirable properties for neural fields used as selection module of the template described in section 2. Motivated by these remarks, we examine in the next sections, with the aid of the presented performance measurement $\Delta_i^B(u)$, the ability of the neural fields models considered in section 3.2 to satisfy these assumed properties.

5 Measuring experimental neural field capacities from scenarios

Driven by the same practical arguments expressed throughout the paper, we describe here an empirical methodology to study the behavior of a field in specific scenarios. The procedure consists in evaluating the dynamics of the field by computing the value of $\overline{\Delta}_i^B(u)$ for a sequence of time steps. We call a plot *dynamic quality curve* of $\overline{\Delta}_i^B(u)$ as a function of time, a of the field for that specific scenario i.

While any empirical analysis is incomplete in regard to the variety of the dynamics a field may experience, it can nevertheless provide relevant results that can be extrapolated to a much broader context. Consequently, we analyze the behavior of the field in multiple simple scenarios, each of them exhibiting one or few key aspects.

In order to test the capacity of the fields to satisfy the (P $\wedge$ Q) properties, we plot the dynamic quality curves of the field in scenarios exhibiting one or more aspects depicted in (Figure 6). In the following paragraphs, we describe a short taxonomy of benchmark scenarios, based on their relevance in modeling applications (in particular biologically-inspired applications modeling cognitive features).

5.1 *Self-organization*
We describe here two scenarios that will be used to analyze the capacity of neural fields to perform self-organization as indicated in section 2.3.3.

5.1.1 *1D scenario*
The first scenario is identical to the one described in section 4.2. The input domain is the two-dimensional set $\Xi = \{\xi = (\xi_1, \xi_2) \mid 0.5 \leq \sqrt{\xi_1^2 + \xi_2^2} < 1\}$. X is a one-dimensional set with a ring topology. A new sample is drawn from Ξ every 100 steps, and is maintained for the entire epoch of 100 steps. The tuning curve function of the prototypes is given by $i(x) = e^{-5\|\omega(x) - \xi\|^2}$. The learning rule is given by equation 2, with $\alpha = 2 \cdot 10^{-4}$.

5.1.2 2D scenario

The second scenario is inspired from the approaches of modeling visual cortical processing. Bi-dimensional neural fields are of primary interest when cortical modeling is addressed. As many authors in such a context address the modeling of orientation selectivity in V1 [33, 48], the scenario proposed here aims at feeding the neural field with inputs standing for orientations.

Here, a bi-dimensional field X is considered for quantifying data drawn according to a uni-dimensional uniform distribution from the input space $\Xi = [-\pi/2, \pi/2]$. Even if such projection, from uni- to bi-dimensional discrete space may appear unusual, since the $\{\omega(x)\}_{x \in X}$ are randomly initialized, self-organization in such context consists in unfolding a 2D map, which is not straightforward when considered under the perspective of the field dynamics. Input is actually the orientation angle $\xi \in [-\pi/2, \pi/2]$ of an oriented bar stimulus and the prototypes are also angles $\omega(x) \in [-\pi/2, \pi/2]$. This allows to represent the prototypes values $\omega(x)$ at each position x graphically with an oriented bar.

The frequency of drawing samples from the input domain, as well as the learning rate are the same as in the first scenario. However, in this case we will examine two different tuning curves. First, $i(x) = cos((\omega(x) - \xi) \mathrm{mod} \pi)$. Second, $i(x) = e^{|\omega(x) - \xi| \mathrm{mod} \pi}$. Even if the tuning curve is not a parameter of the field, we will show how it can influence dramatically the global output of the self-organizing process.

For both scenarios, all the prototypes are randomly initialized. The learning is permanent (i.e. the prototypes are adjusted at every time step), and no reset of the field potentials is performed during the simulations. The input $\xi(x), x \in X$ is the same ξ for all the positions x. In order not to experience any border effects, X is considered having a toroidal topology in every case.

5.2 *Auxiliary scenarios*

In addition to the self-organization, we investigate here some other scenarios exhibiting features that may be exploited in practical applications of neural fields. Furthermore, we are interested to observe whether features shown by classical neural fields are also conserved by the newly introduced neural field model.

5.2.1 Attention

The attentional scenarios are typically characterized by inputs following continuous trajectories that are sometimes occluded or suppressed, or patches of stimuli randomly appearing and disappearing in the input distribution.

The expected behavior of a field in such cases is to focus its activity (raise bumps) at some salient place and to track it, undisturbed by noise, local distractors or the other input patterns.

Therefore, in these scenarios, there are two main features that have to be investigated: the fact that the field raises high bumps somewhere in the field (focus), and the fact that it adapts its response so as to follow the focused patches of input (tracking). For example, the cases (a, d, g) from the (Figure 6) are in this sense relevant as basic attentional scenarios.

5.2.2 Exploration

Exploration scenarios refer to an input characterized by almost constant level of stimuli (the so-called plateaus). In these cases, as illustrated in (Figure 6. c, h), the field is expected to select only some regions where to raise high bumps, the input locations being all equally probable to elicit the formation of bumps.

6 Experiments

In the current section, the three models introduced in section 3.2 (i.e. ANF, PENF, BINP) are empirically evaluated according to the methodology described in the previous section. The values of the free parameters of the neural field models used in each scenario simulation are reported in annex A.

6.1 *Self-organization*
6.1.1 1D scenario
The three neural fields are used here to implement the detection module of the template described in 2.3.3, in order to solve the uni-dimensional self-organization scenario presented previously in section 5.1.1. The results obtained by using each of the three considered models are presented in (Figure 10). For each step of the self-organization scenario, the quality indicator $\overline{\Delta}_i^B(u)$ was measured. The histograms of these data corresponding to each neural field implementation are plotted in (Figure 11).
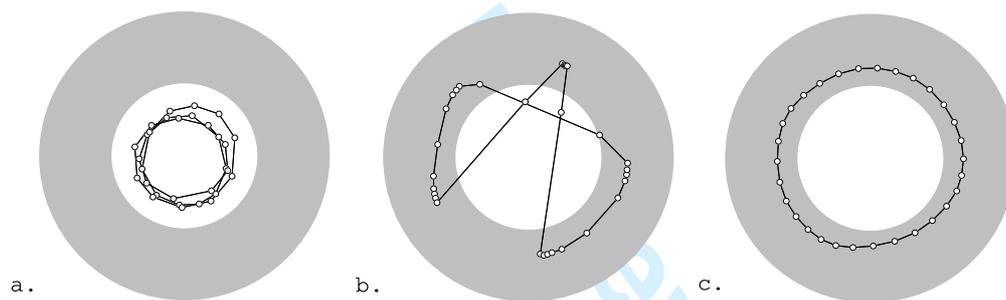

Figure 10. Solving a one-dimensional self-organizing task using neural fields as described in section 2.3.3. Results provided by the usage of the following field models: a. ANF, b. PNF, c. BINP. The input distribution is shown as the gray ring shape seen in every sub-figure. New samples are presented every 100 steps and a total of 16000 samples are drawn throughout the experiment.
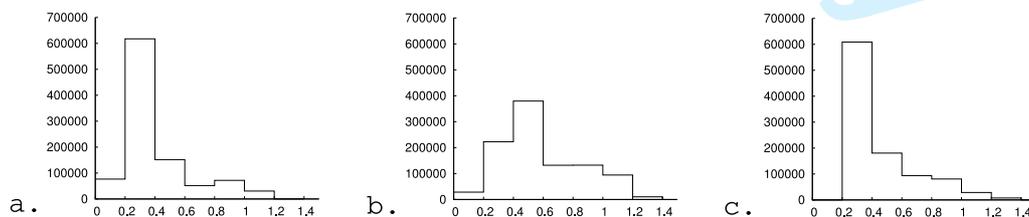

Figure 11. Histograms of the $\overline{\Delta}_i^B(u)$ values measured for each step of the uni-dimensional self-organizing scenario. The corresponding plots for each of the three neural fields implementations: a. ANF, b. PENF, c. BINP.

As it can be observed, the two classical models (ANF and PENF) fail to achieve the expected results, while the new model (BINP) succeeds in correctly reorganizing the prototypes. In paragraph 6.3 the plots from the two figures (10 and 11) are analyzed in detail and then the explanation for such results is provided.

6.1.2 2D scenario
As described previously in section 5.1.2, for the bi-dimensional case, we also study the impact of the tuning curve on the global behavior of the self-organization process.

Let us first use the tuning curve of the prototypes as defined by $i(x) = cos((\omega(x) - \xi) \mathrm{mod} \pi)$. Unlike the 1D scenario, both classical fields (ANF and PENF) succeed in reorganizing very accurately the prototypes (see (Figure 12)).
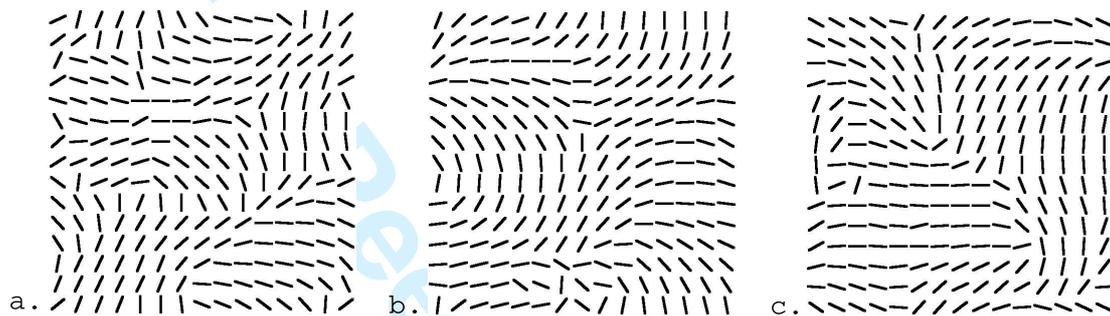

Figure 12. Solving a two-dimensional self-organizing task with different neural field implementations. The task consists in organizing a 2D orientation map, similar to that found in the V1 cortical area. The input is given by a uniform distribution of orientations in $[-\Pi/2, \Pi/2]$ interval. Here, the response of the prototypes is defined by the tuning curve function $i(x) = cos((\omega(x) - \xi) \mathrm{mod} \pi)$. and a graphical representation of the different field solutions is given: a. ANF, b. PENF, c. BINP. New samples are drawn every 100 steps and the scenario consists of a total of 8000 samples.
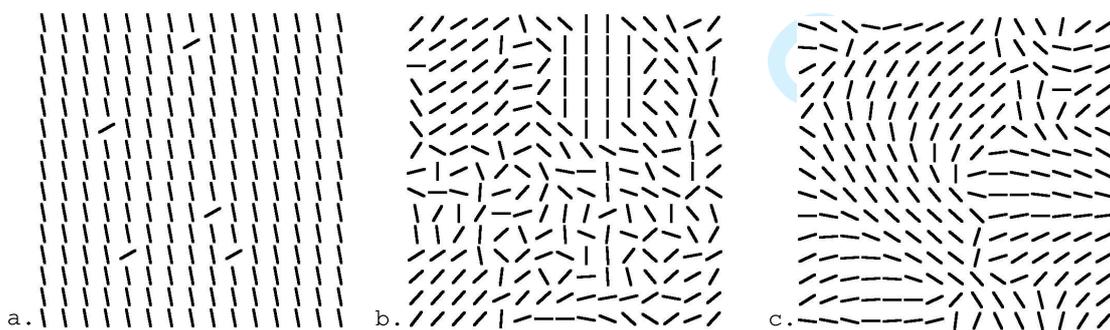

Figure 13. Solving the same self-organizing task as reported in (Figure 2), but here with a different tuning curve for the prototypes, defined as $i(x) = e^{-|\omega(x) - \xi| \mathrm{mod} \pi}$. Final results produced by neural fields: a. ANF, b. PENF, c. BINP.

In the second bi-dimensional experiment, the tuning curve of the prototypes is set to $i(x) = e^{-|\omega(x) - \xi| \mathrm{mod} \pi}$, that is much sharper than the one used previously. All the

other parameters or conditions of the self-organizing scenario remain unchanged. Unlike the previous context, the results obtained in this case by the classical fields (ANF and PENF) are completely unsatisfactory, while the newly introduced field model, BINP continues to achieve very good results (see (Figure 13)).

The results presented here empirically suggest that the newly proposed BINP model is indeed a good candidate for developing self-organization mechanisms driven by dynamic neural field computation. In addition, the two scenarios presented here highlight the influence of the tuning curve function (and thus the sensitivity of the prototypes) on the global behavior of the classical fields. The results from both scenarios are further analyzed and explained in paragraph 6.3.

6.2 *Auxiliary scenarios*
6.2.1 Attention
Dynamic neural fields (mainly ANF) have been successfully used in many applications to simulate basic attentional mechanisms. Here we examine the evolution of the three fields dynamics in some simple scenarios that could be relevant for attention modeling.
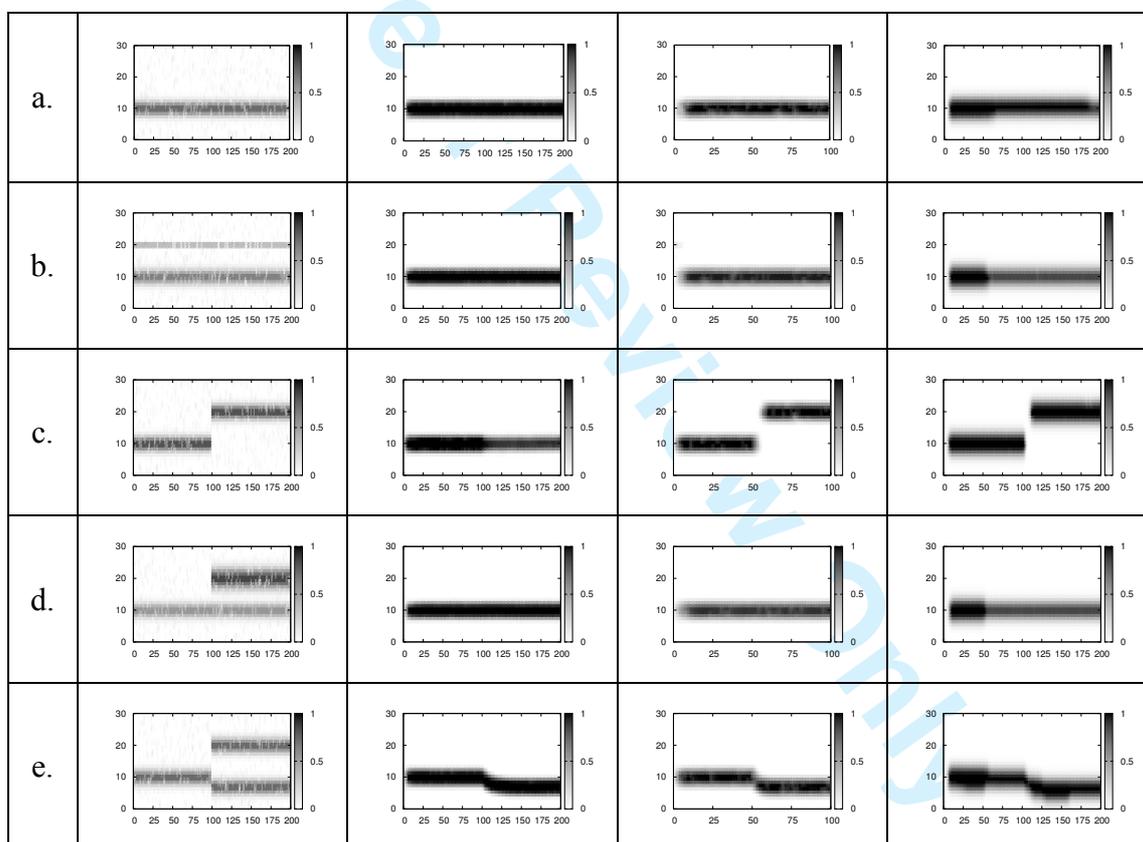

Figure 14. Basic attentional scenarios simulations. The parameters of the fields equations are chosen as to allow the emergence of only one activity bump. For each row, from left to right: the input stimulus (first column) and the response of the three neural fields: ANF (second column), PENF (third column) and BINP (fourth column). For each individual plot, the x-axis represents the time step of the simulation and the y-axis, the intensity (shown in levels of gray) of the neuronal activity of each neuronal unit of the field.
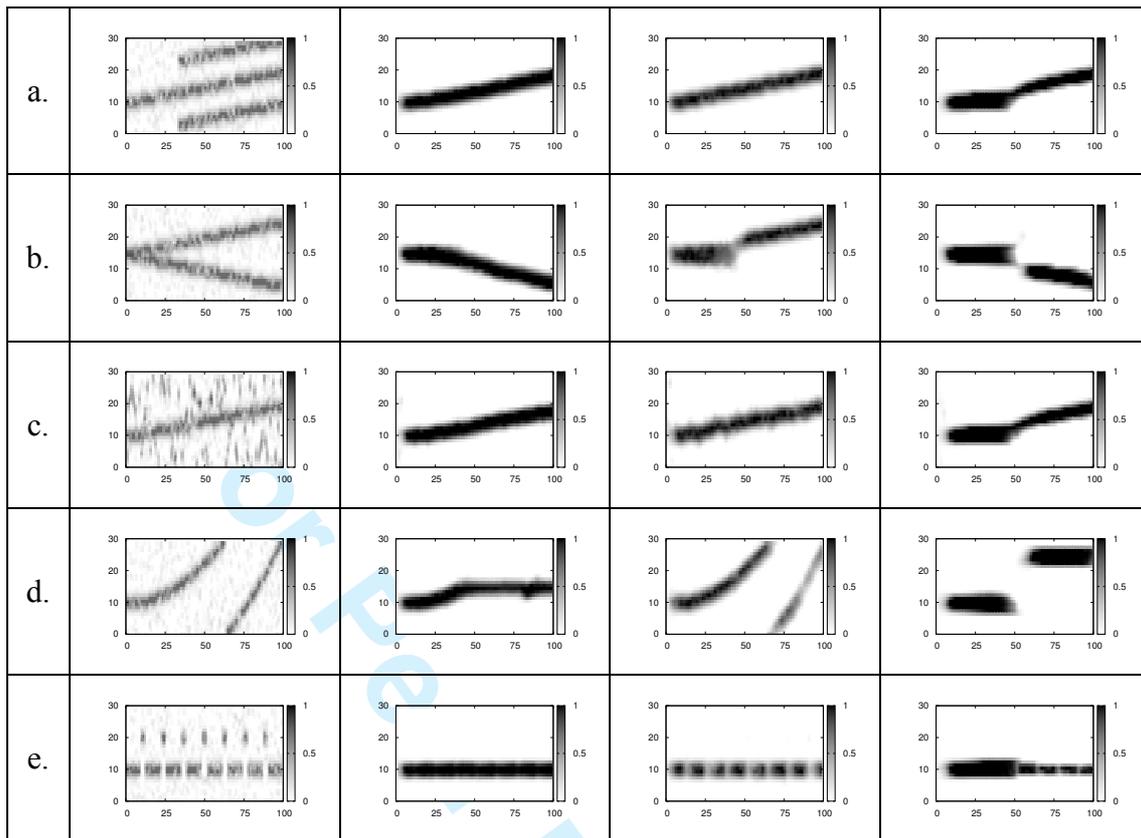
Figure 15. Basic attentional scenarios simulations. Continuation
from (Figure 14). Same legend as in (Figure 14).

For visual simplicity, the experiments presented here concern only uni-dimensional scenarios. In (Figures 14 and 15) each line plots the following sequence of sub-figures: the temporal evolution of input i, and three columns with the corresponding response of each of the three fields considered in our analysis (ANF, PENF and BINP). In all the sub-figures, the x-axis represents the temporal axis. For each step of simulation, the amplitude of the distribution (input i or field response u) for each position $x \in X$ is shown in levels of gray along the y-axis.

The parameters of the field equations considered in all the following experiments are chosen as to guarantee the emergence of a single neural bump in the field. Results where the density constraint allows multiple bumps in the field can be easily extrapolated from these data. The formation of multiple simultaneous bumps sparsely distributed along the field will be addressed in the near future of our research, as our long-term goal is to model multimodal self-organization through neural field computation.

(Figures 14 and 15) present the simulated data from several attentional scenarios. As a general remark, all the fields are behaving as expected in most of the cases, i.e. first focusing their activity on a patch of input stimuli (thus raising a bump) and then following this patch throughout the simulation. Consequently, the corresponding quality curves of the fields approach zero in such cases, as seen in (Figures 16 and 17). These empirical results entitle us to consider them suitable for implementing attentional mechanisms. Moreover, the newly proposed neural field is also appropriate to be used in such attentional modeling, therefore being completely compatible with the current cognitive modeling approaches.
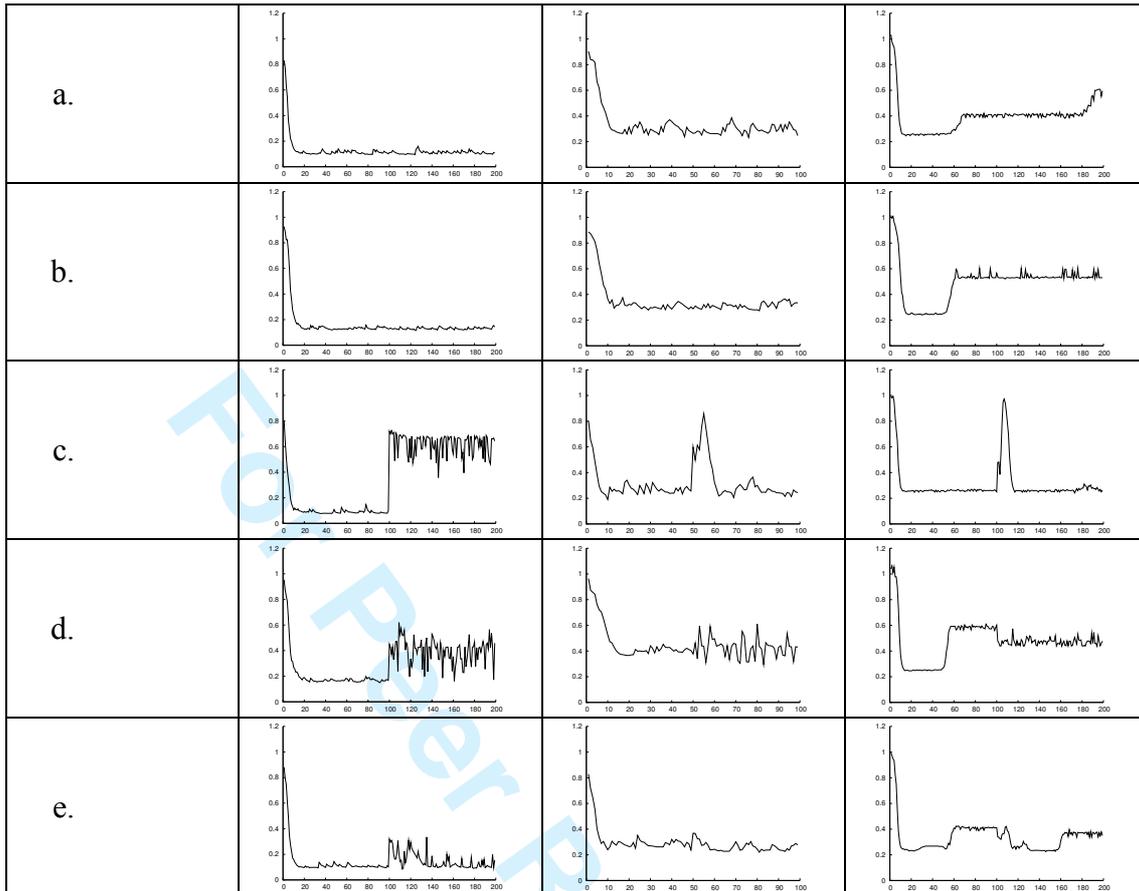
Figure 16. Quality curves of the three fields from the scenarios
presented in (Figure 14). For each row, from left to right: ANF
(first column), PENF (second column), BINP (third column).
For each individual plot, the x-axis represents the time step of
the simulation and the y-axis, the quality indicator $\overline{\Delta}_i^B(u)$.

The only significant difference between the fields behavior is the smaller degree of
"inertia" that the dynamics of Pinto and Ermentrout field manifests. Indeed, in
scenario (j), the ANF and BINP are not capable of following the accelerated shifting
of the input stimuli. Conversely, the Pinto and Ermentrout field is more sensitive to
the input. Definitely, the additional negative feedback term $v(x,t)$ used in the PENF
and in the BINP models facilitates the dynamics of the bumps, allowing therefore a
quicker adaptation to input stimuli changes.

6.2.2 Exploration
These exploration scenarios concern a key aspect to our analysis. As observed (see
(Figure 18)), the classical fields are incapable of forming a significant response in low
input conditions. Only the new field is able to do so, showing also a very good
dynamics of the bumps. In such case, the classical fields are non-selective, and this is
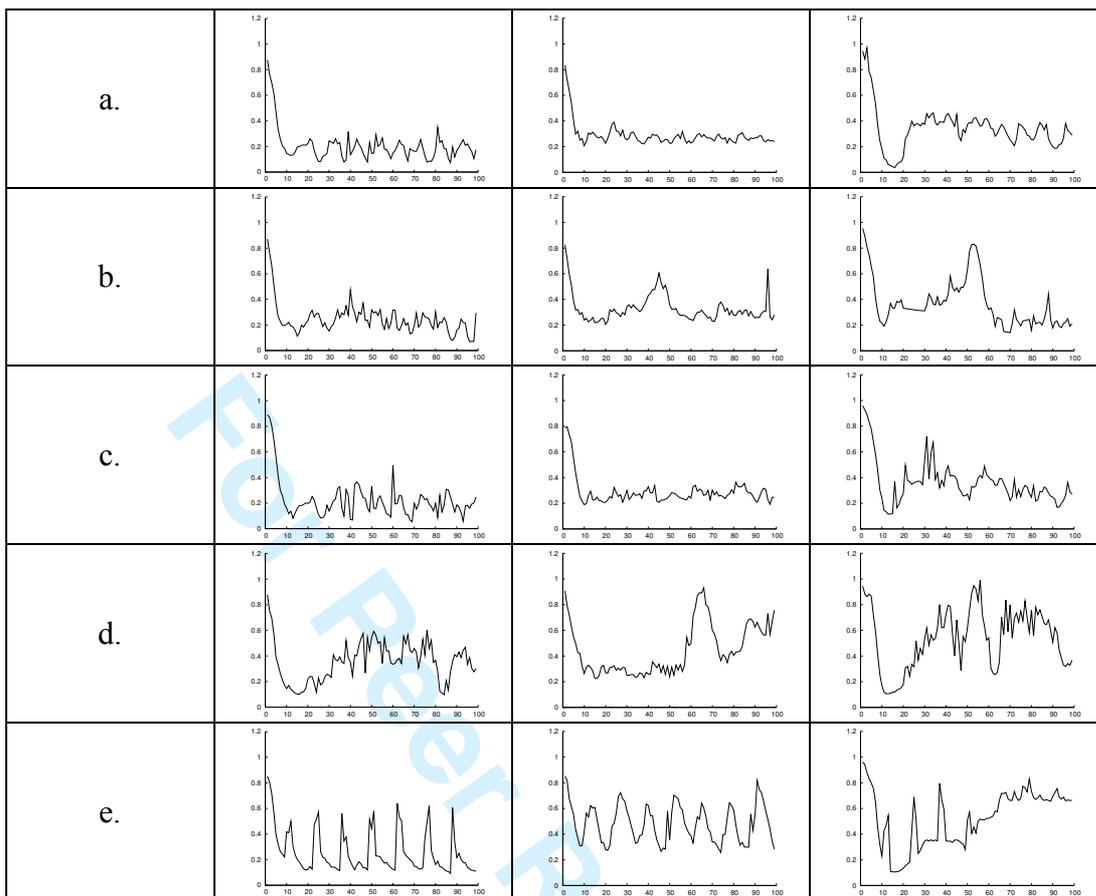immediately reflected in the quality curves plots from the (Figure 19).

Figure 17. Quality curves of the three fields from the scenarios
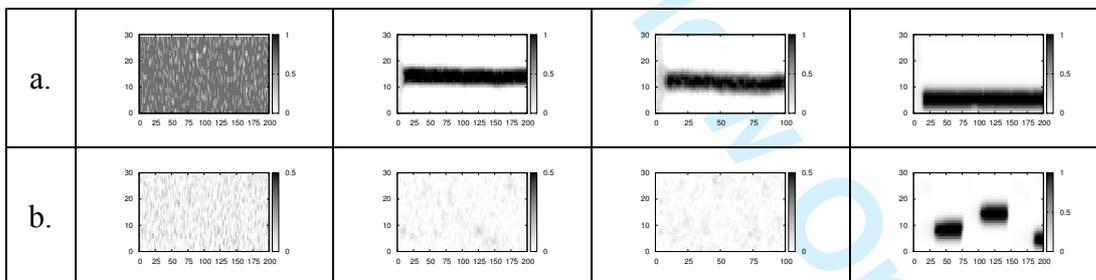presented in (Figure 15). Same legend as in (Figure 16).

Figure 18. Basic exploration scenarios simulations. The
parameters of the fields equations are chosen as to allow the
emergence of only one activity bump. For each row, from left
to right: the input stimulus (first column) and the response of a
neural field: ANF (second column), PENF (third column) and
BINP (fourth column). a. high level plateau input; b. low level
plateau input. Same legend as in (Figure 14).

6.3 *Discussion*

In this section 6, the three neural field models (ANF, PENF and BINP) were
examined in multiple scenarios in order to empirically highlight particular features of
their dynamics.

As a whole, the above qualitative analysis of the three fields dynamics entitle us to consider the newly proposed BINP model appropriate for supporting the implementation of reliable and robust self-organization mechanisms (see (Figure 13. c)). In addition, other dynamical features exhibited by the classical fields are globally conserved by the new model (see sections 5.2 and 6.2), and thus the new field may be regarded as an enhancement of the former models in our claimed attempt to extend the applicative area of dynamic neural fields.
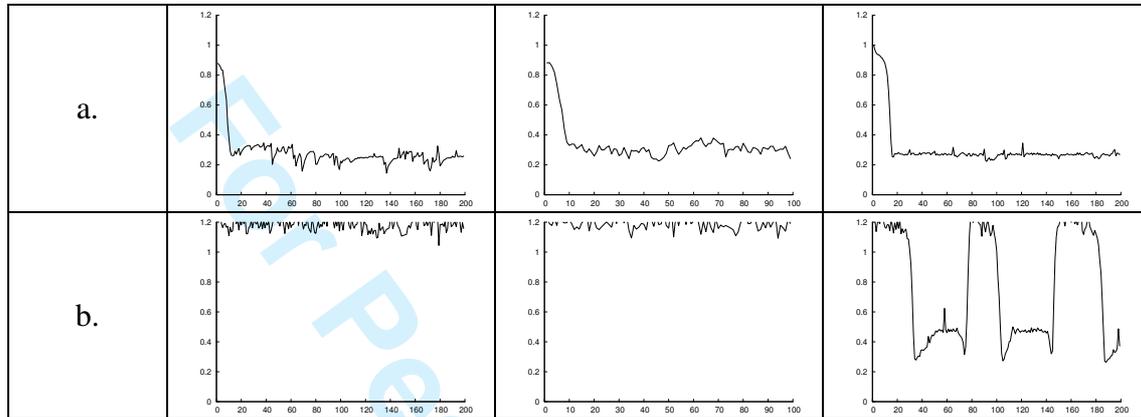

Figure 19. Quality curves of the three fields from the scenarios
presented in (Figure 18). Same legend as in (Figure 16).

As the actual goal of the paper is to conceive self-organizing mechanisms by the means of neural field computation, we focus now on scenarios mentioned in sections 5.1 and 6.1. We remind that the role of neural fields in these scenarios is to implement (in a totally distributed way) the WTM policy that modulates the learning rule of the self-organizing algorithm (according to equation 2).

Let us start the analysis of results on the 1D ring shape distribution illustrated on (Figure 10. a), where the classical ANF is used. At the beginning of the learning process, prototypes are set to random values taken in the enclosing box of the ring. When some input is tossed from the coronal shape Ξ (drawn in light gray in (Figure 10)), it may be close to only few prototypes, that are the only ones for which the matching i is not null. Moreover, those prototypes may be scattered over X, the ring of units, since no organization stands a priori. The i distribution (see (Figure. 2)) is thus made of few sparse peaks. In such condition, we have observed that ANF field is able to set up a bump of activity around the highest of those peaks. After stabilization, as mentioned in paragraph 2.3.3, the next input is tossed from the ring, without any reset of the current bump. The new sparse peak resulting from the next input are not able to force the field to reconsider the bump position, especially if there are still some units in the current bump that match the new input. Through the presentation of many input $\xi \in \Xi$, the effect of this constant bump on the units placed where the bump stands is a continuous learning. Due to the update rule (equation 2), this makes the prototypes in the bump smoothly change in order to reach the average of all the presented inputs. In the case of the coronal shape, this average is out of the shape, in the central hole. The result is that the longer the prototypes in the bump learn, the less they match. After enough time steps, matching is so weak for the units in the bump, that the distribution of peaks for the next input outside the bump is significantly stronger than each poor matching of the units inside the bump. Then only, the bump can move elsewhere, be stable until the units in that new place in the field do not

match in turn, etc. After many inputs, the result displayed on (Figure 10. a) shows that all the prototypes have been ``pushed away'' the coronal Ξ, near the average central point.

When the same experiment is made using the PENF field, things start the same way, pushing prototypes in the central hole of the ring distribution of inputs. Then, we have observed that the higher mobility of the bumps (that can be seen on (Figure. 15)) allows the prototypes to be extracted from the hole. Nevertheless, the resulting field needs to be unfolded to achieve perfect topological matching, as shown by the stable state reached in this case (Figure 10.b). Last, the BINP field proposed in this paper takes benefits from the back inhibition to avoid having constant bumps on a weakly matching area of units (see (Figure 18. b)). The story at the beginning of learning is thus dramatically different, and is much closer to what an touchstone field, in the sense of paragraph 4.1.2, should have done, as result in (Figure 10. c) shows. For those first set of experiments, it can be said that the dynamics of self-organization is sometimes complex, even depending on the topological features of the input distribution. Here, the presence of a hole at the center of mass of the coronal shape has tricky effects. These effects are difficult to capture with our quality measurement tool, since (Figure 11) shows similar histograms in that experiment for the three fields. Even if keeping $\overline{\Delta}_i^B(u)$ null, as the touchstone field does by definition, is a sufficient condition for self-organizing (see (Figure 9)), a slight drift from this condition, as the ones by ANF and PENF in (Figure 11), may have dramatic effects. The observation of good (but not perfect) quality measurements may thus be used as a hint for the designer of a field equation, rather than a guaranty for the capability of self-organization. Nevertheless, it is more relevant as a tool to quantify the behavior on attentional scenarios, as results in section 5.2 show, and it is also helpful for parameter tuning [1].

The experiments inspired from V1 (Figures 12 and 13) show the role of this feature in the self-organization dynamics. In both figures, the inputs are an angle (1D), that is mapped on a bi-dimensional array of units. Quite surprisingly, in (Figure 12), all the three fields manage to organize prototypes, whereas performances in (Figure 13) are coherent with what has been observed on in the previous coronal input distribution case, using a 1D field. The difference between the two is a sensitivity of the matching. Each unit acts as a band-pass filter centered at the prototype value, but the width of this band is wide on (Figure 12) and quite sharp in (Figure 13). At the beginning of learning, the former leads to peaks in the distribution of i that are *not* sparse, thus allowing to reconsider bump position even for the ANF equation, whereas the latter produces sparse distribution of peaks, as for (Figure 10). The sparseness of i distribution when learning starts is crucial for self-organization, and this parameter depends on the band-pass width, regards to the number of units and the size of the region where the inputs are taken from.

To sum up, a deep analysis of the self-organization dynamics with WTM procedure carried out by a neural field reveals that this process is complex and depends on hidden parameters, as the topology of the input distribution (e.g. presence of holes) or the width of the band-pass filters in comparison with the input variability. These crucial aspects of the learning task seem to be quite difficult to handle with formal tools, and the one proposed here helps only partially. Nevertheless, the BINP field introduced in this paper appears to be robust to those aspects, due to the active back inhibition mechanism that ensures an extreme bump mobility when the self-organization is not achieved (i.e. some units in the bump still poorly match). This is why we consider the BINP network as a good candidate for addressing more complex

self-organizing task, as the one of organizing behavior in cortically inspired architectures, that motivates the present study.

7 Conclusion and future work

Even if the use of distributed competition mechanisms in a population of adaptive band-pass filters as the core mechanism of cortical processing is not a recent idea, since it was formulated by Kohonen in the eighties, the research in cortically-inspired computational systems allowing situated and flexible computation is still not mature. The amazing computational properties of the cortical substrate is a strong motivation to persevere in this direction, but this research has to cope with the complexity of non linear dynamical systems, since even toy simulations require to set up recurrent network with emerging properties that are not fully understood by mathematical approaches.

In this paper, an empirical tool is proposed to tackle those design difficulties. It consists of a procedure that gives a scalar evaluation, at each time step, to the state of a neural field, according to the input it receives. This evaluation has been shown to have the right meaning, since it actually determines whether the field succeeds in behaving as a cortical competition should do, for endowing a system with attentional selectivity and self-organization. These latter two properties are indeed the ones expected for designing cognitive systems based on populations of units.

From suitable experiments, it has been shown that the classical neural field equation by Amari needs to be improved in such a context, as well as the oscillator-based neural field by Ermentrout and Pinto. An improvement is proposed, that is shown to fit the requirements for the design of self-organizing systems, as well as for subsequent attentional properties. It allows to get rid of the common computational tricks that are necessary to other fields. First of them is the need for specific weight initialization, that is not required with the BINP equation. Second, there is no need to reset the network when a new input occurs. This implies that the system is free from being informed or synchronized with some input presentation scheme, it runs on-line and only copes with the intrinsic time steps for evaluating its computational units, just receiving input as a continuous stream, as any situated system necessarily does. Third, successive inputs do not need to be correlated, since the field is able to reconsider a bump position when it becomes irrelevant, allowing a completely new input to be considered without the influence of the previous one. This is what the back inhibition in the BINP model actually does. For example, this allows to consider visual inputs in more realistic cases. Indeed, when saccades make frequent and sudden changes in the gaze direction, sensing the visual scene, the input flow coming from the retina to the brain is undoubtedly discontinuous.

Future work consists in investigating the perspective offered by a reliable distributed self-organizing scheme. First, using the mechanism for joint self-organization between multi-modal fields is the next step, since this work [32] was the motivation of the present study, as mentioned in the text. Second, the robustness of such dynamical systems when used with large fields (several millions of units) is also considered, since we are currently working on implementing the model sketched in (Figure 2) on a cluster of PCs [19]. Last, another direction is to investigate how this mechanism can behave in a self-organizing field where several bumps are allowed. This is, for example, mandatory for vision, since the primary visual cortex is organized as a gradient from central vision to peripheral one [24], and learning

processes have to operate in parallel, within different bumps, at different levels of eccentricity. This requirement is mandatory for any other realistic cortical map.

To sum up, being able to quantify, and then control, the inner mechanisms of an artificial cortical system makes the study of the outer properties emerging from population of units feasible, allowing then to address cognitive systems by a cortical paradigm.

8 Acknowledgements

References

[1] L. Alecu and H. Frezza-Buet, *Application-driven parameter tuning methodology for dynamic neural field equations.* In Proceedings of the 16th International Conference on Neural Information Processing (ICONIP), Bankok, Thailand. (2009).

[2] F. Alexandre, *Cortical basis of communication: Local computation, coordination, attention.* Neural Networks, (2009), 22(2):126–133.

[3] F. Alexandre, F. Guyot, J.-P. Haton and Y. Burnod, *The cortical column: A new processing unit for multilayered networks.* Neural Networks, (1991), 4(1):15–25.

[4] S.-I. Amari, *Dynamics of pattern formation in lateral-inhibition type neural fields.* Biological Cybernetics, (1977), 27:77–87.

[5] T. Binzegger, R. Douglas and K. Martin, *Cortical architecture.* In De Gregorio, M., Di Maio, V., Frucci, M., and Musio, C., editors, Brain, Vision, and Artifical Intelligence, (2005), LNCS 3704, pages 15–28. Springer-Verlag.

[6] P. Bressloff, J. Cowan, M. Golubitsky, P. Thomas and M. Wiener, *What geometric visual hallucinations tell us about the visual cortex.* Neural Computation, (2002), 14:473–491.

[7] P. Bressloff and S. Folias, *Front bifurcations in an excitatory neural network.* SIAM Journal of Applied Mathematics, (2004), 65(1):131–151.

[8] Y. Burnod, *An adaptive Neural Network: The Cerebral Cortex.* Masson. (1989).

[9] J. Daugmann, *Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters.* Journal of Optical Society of America A, (1985), 2(7):1160–1169.

[10] K. Doya, *What are the computations of the cerebellum, the basal ganglia and the cerebral cortex?* Neural Networks, (1999), 12(7-8):961–974.

[11] T. Elbert and B. Rockstroh, *Reorganization of human cerebral cortex: The range of changes following use and injury.* The Neuroscientist, (2004), 10(2):129–141.

[12] M. Enculescu and M. Bestehorn, *Mechanisms for activity spread in a neural field model.* Neurocomputing, (2007), 70:2996–3000.

[13] W. Erlhagen and E. Bicho, *The dynamic neural field approach to cognitive robotics.* Journal of Neural Engineering, (2006), 3:R36–R54.

[14] W. Fellenz and J. Taylor, *Establishing retinotopy by lateral-inhibition type homogeneous neural fields.* Neurocomputing, (2002), 48:313–322.

[15] J. Fix, N. Rougier and F. Alexandre, *From physiological principles to computational models of the cortex.* Journal of Physiology-Paris, (2007), 101(1-3):32–39.

[16] S.E. Folias and P.C. Bressloff, *Breathing pulses in an excitatory neural network.* SIAM Journal of Applied Dynamical Systems, (2004), 3(3):378–407.

[17] S.E. Folias and P.C. Bressloff, *Breathers in two-dimensional neural media.* Physical Review Letters, (2005), 95.

[18] H. Frezza-Buet, *Bijama software library.* Website (retrieved December, 2009a). http://ims.metz.supelec.fr/spip.php?rubrique28.

[19] H. Frezza-Buet, *Intercell project.* Website (retrieved December, 2009b). http://intercell.metz.supelec.fr.

[20] H. Frezza-Buet and F. Alexandre, *From a biological to a computational model for the autonomous behavior of an animat.* Information Sciences, (2002), 144(1-4):1–43.

[21] H. Frezza-Buet and O. Ménard, *Making competition in neural fields suitable for computational architectures.* In Duch, W., Kacprzyk, J., and Oja, E., editors, International Conference on Artificial Neural Networks (ICANN), Lecture Notes in Computer Science (LNCS) 3696, part I, (2005), pages 217–222. Springer-Verlag.

[22] J. Hertz, A. Krogh, and R. Palmer, *Introduction to the Theory of Neural Computation.* (1991), Addison-Wesley.

[23] C. Horta and W. Erlhagen, *Robust persistent activity in neural fields with asymmetric connectivity.* Neurocomputing, (2006), 69:1141–1145.

[24] D. Hubel and T. Wiesel, *Receptive fields, binocular interaction, and functional architecture of cat striate cortex.* Journal of Physiology. (1962).

[25] A. Hutt, M. Bestehorn and T. Wennekers, *Pattern formation in intracortical neuronal fields.* Network: Computation in Neural Systems, (2003), 14:351–368.

[26] I. Iossifidis and A. Steinhage, *Controlling an 8 dof manipulator by means of neural fields.* In Int. Conf. on Field and Service Robotics, (2001).

[27] J. Johnson, J. Spencer, and G. Schöner, *Moving to higher ground: The dynamic field theory and the dynamics of visual cognition.* New Ideas in Psychology, (2008), 26(2):227 – 251.

[28] E. Jones, *Microcolumns in the cerebral cortex.* (2000), PNAS, 97(10):5019–5021.

[29] T. Kohonen, *Self-organized formation of topologically correct feature maps.* Biological Cybernetics, (1982), 43:59–69.

[30] T. Kohonen, *Self-Organization and Associative Memory*, volume 8 of Springer Series in Information Sciences. Springer-Verlag, (1989).

[31] J.B. MacQueen, *Some methods for classification and analysis of multivariate observations.* In Cam, L. M. L. and Neyman, J., editors, Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability, (1967), volume 1, pages 281–297. University of California Press.

[32] O. Ménard and H. Frezza-Buet, *Model of multi-modal cortical processing: Coherent learning in self-organizing modules.* Neural Networks, (2005), 18(5-6):646–655.

[33] R. Miikkulainen, J.A. Bednar, Y. Choe, and J. Sirosh, *Computational Maps in the Visual Cortex.* (2005), Springer.

[34] I. Mikhailova and C. Goerick, *Conditions of activity bubble uniqueness in dynamic neural fields.* Biological Cybernetics, (2005), 92(2):82–91.

[35] K.D. Miller, D.J. Simons and D.J. Pinto, *Processing in layer 4 of the neocortical circuit: New insights from visual and somatosensory cortex.* Current Opinion in Neurobiology, (2001), 11:488–497.

[36] V. Mountcastle, *The columnar organization of the neocortex.* Brain, (1997), 120:701–722.

[37] D. Pinto and G. Ermentrout, *Spatially structured activity in synaptically coupled neuronal networks: I. traveling fronts and pulses.* SIAM J. Appl. Math, (2001a), 62:206–225.

[38] D. Pinto and G. Ermentrout, *Spatially structured activity in synaptically coupled neuronal networks: II. Lateral inhibition and standing pulses.* SIAM J. Appl. Math, (2001b), 62:226–243.

[39] N. Rougier, *Implicit and explicit representations.* Neural Networks, (2009), 22(2):155–160.

[40] N. Rougier and J. Vitay, *Emergence of attention within a neural population.* Neural Networks, (2006), 5(19):573–581.

[41] S. Sabatini and L. Secchi, *A continuum-field model of visual cortex stimulus-driven behaviour: emergent oscillations and coherence fields.* Neurocomputing, (2004), 57:411–433.

[42] S. Sabatini, and L. Secchi, *Emergence of oscillations and spatiotemporal coherence states in a continuum-model of excitatory and inhibitory neurons.* BioSystems, (2005), 79:101–108.

[43] J. Salmon and T. Trappenberg, *Modeling the integration of expectations in visual search with centre-surround neural fields.* Neural Networks, (2008), 21(10):1476–1492.

[44] S. Schneider and W. Erlhagen, *A neural field model for saccade planning in the superior colliculus: speed-accuracy tradeoff in the double-target paradigm.* Neurocomputing, (2002), 44-46:623–628.

[45] V. Simmering, A. Schutte and J. Spencer, *Generalizing the dynamic field theory of spatial cognition across real and developmental time scales.* Brain Research, (2007).

[46] M. Stavrinou, S. Penna, V. Pizzella, K. Torquati, F. Cianflone, R. Franciotti, A. Bezerianos, G. Romani and P. Rossini, *Temporal dynamics of plastic changes in human primary somatosensory cortex after finger webbing.* Cerebral Cortex, (2007), 17(9):2134–2142.

[47] W. Taouali, F. Alexandre, A. Hutt and N. Rougier, *Asynchronous evaluation as an efficient and natural way to compute neural networks.* In Proceedings of ICNAAM, (2009).

[48] J.G. Taylor, *Neural networks for consciousness.* Neural Netowrks, (1997), 10(7):1207–1225.

[49] M. Toussaint, *Learning a world model and planning with a self-organizing, dynamic neural system.* In NIPS (2003).

[50] M. Toussaint, *A sensorimotor map: modulating lateral interactions for anticipation and planning.* Neural Computation, (2006), 18(5):1132–1155.

[51] N.A. Venkov, S. Coombes and P. Matthews, *Dynamic instabilities in scalar neural field equations with space-dependent delays.* Physica D, (2007), 232:1–15.

[52] C. Wilimzig and G. Schöner, *The emergence of stimulus-response assoctiations from neural activation fields: dynamic neural theory.* In The 27th Annual Conference of the Cognitive Science Society, (2004), pages 2359–2364.

[53] H. Wilson and J. Cowan, *Excitatory and inhibitory interactions in localized populations of model neurons.* Biophysical Journal, (1972), 12(1):1–24.

[54] H. Wilson and J. Cowan, *A mathematical theory of the functional dynamics of cortical and thalamic nervous tissue.* Kybernetik, (1973), 13:55–80.

Appendix A

Here we present the numeric values used in our simulations. For all the scenarios considered in the paper, we use the following parameter values:

• for the ANF equations (3 and 4):

$A^+ = 0.8, A^- = 0.5, a = 0.05, b = 0.001, \tau = 0.3, h = 0$;

• for the PENF equations (4, 5 and 6):

$A^+ = 0.67, A^- = 0.35, a = 0.05, b = 0.001, \beta = 0.85, \tau = 0.3, \eta = 0.5$.

• for the BINP equations (4, 7 and 8):

$A^+ = 0.67, A^- = 0.35, a = 0.15, b = 0.005, c = 0.04, \alpha = 2.4, \beta = 3, \gamma = 3, \tau = 0.3$.

$g = g_1(i)g_2(v)$ and h functions are the discretized forms (in order to speed up the simulations computations) of the sigmoid functions plotted in figure 5, as shown below.

$$
g_1(i) = \begin{cases}
1 & i = 0 \\
-0.33i + 1 & 0 < i \le 0.30 \\
-5.33i + 2.50 & 0.30 < i \le 0.45 \\
-0.18i + 0.18 & 0.45 < i \le 1
\end{cases}
$$

$$
g_2(v) = \begin{cases}
0 & v = 0 \\
0.33v & 0 < v \le 0.60 \\
3.50i - 1.90 & 0.60 < i \le 0.80 \\
0.50i + 0.50 & 0.80 < i \le 1
\end{cases}
$$

$$
h(v) = \begin{cases}
0.215v - 0.095 & 0 \le v \le 0.20 \\
0.620v - 0.175 & 0.20 < v \le 0.30 \\
0.014v + 0.005 & 0.30 < v \le 1
\end{cases}
$$