Upper Bounds for Robustness of CNN Templates and a Design Approach for Robust Templates

BAHRAM MIRZAI and GEORGE S. MOSCHYTZ

Signal and Information Processing Laboratory
Swiss Federal Institute of Technology
Zurich, Switzerland
mirzai@isi.ee.ethz.ch, moschytz@isi.ee.ethz.ch

Abstract — **CNNs constitute a class of spatially discrete, nonlinear dynamic systems. Once the inputs and the states are initialized, the dynamic of a CNN is determined by a set of parameters, so-called templates. We investigate issues concerning the dynamic behavior of a CNN due to variations in template values. In particular, we derive, based on the output invariance at the equlibrium, upper bounds for these variations. Furthermore, a general design approach for robust templates is proposed.**

I. Introduction

Cellular neural networks (CNNs) constitute a class of nonlinear, dynamic systems with local interaction. In contrast to other interconnected neural networks, CNNs may be more suitable for implementations in analog VLSI technology [1,2,3]. For a 2 dimensional CNN, the dynamic behavior of the ij-th unit or cell is governed by a first order differential equation, namely

$$C\frac{dx_{ij}}{dt} = -\frac{1}{R}x_{ij} + \sum_{kl} a_{ij,kl}\,\mathrm{sat}(x_{kl}) + I_{ij} \qquad (1)$$

$$\mathrm{sat}(x_{ij}) = \frac{1}{2}\{|x_{ij}+1| - |x_{ij}-1|\},$$

where $a_{ij,kl}$ are the feedback parameters, and I_{ij} is a cell-dependent bias which is usually taken to be of the following form

$$I_{ij} = \sum_{kl} b_{ij,kl}\,u_{kl} + I, \qquad (2)$$

where u_{kl} is the inputs of the ij-th cell. The output of each cell is by definition $\mathrm{sat}(x_{ij})$. Henceforth, we normalize $R = C = 1$.

Analog realizations of (1) are inevitably subject to a number of restrictions which may alter the dynamic behavior of (1) severely. Specifically, we point out two of these limitations. First, let α be any one of the CNN parameters. Assume that for (1) to perform a specific task, α has to be set to some nominal value α^*. Due to the limited accuracy of analog implementation, the actual value of α will be

$$\alpha = \alpha^* \pm \varepsilon.$$

Simulations and measurements [1,2] show that ε is typically in the region of 1–5% of the absolute value of α. Second, limitation is dynamic mismatch among cells. This is largely due to the fact that different values of parameters may cause the cells to exhibit different transient behavior; in the absence of robust templates, this renders parallel operation of identical cells invalid.

In view of these restrictions, one is faced with the question to what extent a specific task admits robust templates, and, assuming that they exist, how to obtain them. Before proceeding, we shall elaborate in some detail on the notion of robustness. Assume there exists a nominal vector $\mathbf{p}^*$ of parameters which performs some specific task. We denote $\mathbf{p}^*$ to be ε-robust ($\varepsilon \geq 0$) if the set of vectors $C_\varepsilon(\mathbf{p}^*)$ in the parameter space

$$C_\varepsilon(\mathbf{p}^*) = \{\mathbf{p};\, \|\mathbf{p}-\mathbf{p}^*\|_{max} \leq \varepsilon\}$$

leads to the same stable equilibrium output as $\mathbf{p}^*$. Note that we require only the outputs at the equilibrium to be the same. The actual states may be different, however, they have to belong to the same saturation region. Furthermore, by taking the maximum norm $\|\cdot\|_{max}$ we allow all the components of $\mathbf{p}^*$ to have the same amount of uncertainity, namely ε. In practice, due to their underlying analog realization, some parameters may be more robust than others. Therefore, by choosing $\|\cdot\|_{max}$ we take care of the worst case. In the following we derive upper bounds for ε and propose a design approach for robust templates based on the assumption that the corresponding task can be realized by monotonic state trajectories. We further incorporate the desired robustness in the design of templates. By doing this, we obtain a region of the parameter space within which any vector will solve the specific task with the desired robustness.

II. Upper Bounds

Ideally, a measure of robustness would be one that allows the CNN parameters to vary within some bounds, such that the resulting new equilibrium is still in the same saturation region as the original one. For such a measure we would, however, have to be able to keep track of the CNN transients until they start to converge in the desired saturation region. To avoid the need to determine the transient trajectories, we derive upper bounds for a given set of templates by considering only the asymptotic behavior of the CNN. The limitations that this implies will be discussed later.

In the following we consider those tasks for which the CNN admits an equilibrium point in some saturation region. Introducing matrix notation, (1) can be written as

$$\dot{\mathbf{x}}(t) = -\mathbf{x}(t) + \mathbf{A}\,\mathrm{sat}(\mathbf{x}(t)) + \mathbf{B}\mathbf{u} + \mathbf{i}\,, \qquad (3)$$

where the matrices $\mathbf{A}, \mathbf{B}$ and the vector $\mathbf{i}$ contain the corresponding parameters a_{ij}, b_{ij} and I of a spatially invariant CNN, respectively. Spatially invariant CNNs constitute a class of CNNs where the connecting weights of each cell with its neighbor cells are independent of the cell's position. We assume now that for some fixed input and initial values the templates $\mathbf{A}, \mathbf{B}, \mathbf{i}$ perform some specific task successfully. Consider deviations from these templates of the following form:

$$\mathbf{A} + \delta\mathbf{A},\, \mathbf{B} + \delta\mathbf{B},\, \mathbf{i} + \delta\mathbf{i}\,.$$

We seek to find bounds on

$$\delta\mathbf{A},\, \delta\mathbf{B},\, \delta\mathbf{i}\,,$$

such that the output $\mathbf{x}_{new}$ of the perturbed system, with the same input and initial values, for $t \gg 1$ satisfies the *output invariance* property

$$\mathrm{sat}(\mathbf{x}_{new}) = \mathrm{sat}(\mathbf{x})\,. \qquad (4)$$

For this, let $\mathbf{x}_{new} = \mathbf{x} + \boldsymbol{\eta}$, for some yet undetermined vector valued function $\boldsymbol{\eta}$. By inserting this into equation (3) we get

$$\dot{\boldsymbol{\eta}}(t) = -\boldsymbol{\eta}(t) + (\mathbf{A} + \delta\mathbf{A})\,\mathrm{sat}(\mathbf{x}_{new}(t)) - \mathbf{A}\,\mathrm{sat}(\mathbf{x}(t)) + \delta\mathbf{B}\mathbf{u} + \delta\mathbf{i}\,. \qquad (5)$$

To proceed further we assume (4) to be true and integrate (5) to obtain

$$\boldsymbol{\eta}(t) = e^{-(t-t_0)}\boldsymbol{\eta}(t_0) + \delta\mathbf{A}\,\mathrm{sat}(\mathbf{x}(t)) + \delta\mathbf{B}\mathbf{u} + \delta\mathbf{i}\,,$$

where t_0 is such that (4) holds for all $t \geq t_0$. Note that the last equation describes $\boldsymbol{\eta}(t)$ only for $t \geq t_0$. Letting $t \to \infty$ we get

$$\boldsymbol{\eta} = \delta\mathbf{A}\,\mathrm{sat}(\mathbf{x}^*) + \delta\mathbf{B}\mathbf{u} + \delta\mathbf{i}\,,$$

where

$$\mathbf{x}^* = \lim_{t \to \infty} \mathbf{x}(t)\,.$$

Inserting $\boldsymbol{\eta}$ back into (4) leads to

$$\mathrm{sat}(\mathbf{x}^* + \delta\mathbf{A}\,\mathrm{sat}(\mathbf{x}^*) + \delta\mathbf{B}\mathbf{u} + \delta\mathbf{i}) = \mathrm{sat}(\mathbf{x}^*)\,,$$

which, for $\mathbf{x}^*$ in a saturation region, is equivalent to the following conditions

i) $|\mathbf{x}^* + \delta\mathbf{A}\,\mathrm{sat}(\mathbf{x}^*) + \delta\mathbf{B}\mathbf{u} + \delta\mathbf{i}| \geq \mathbf{1}$
ii) $\mathrm{sign}(\mathbf{x}^* + \delta\mathbf{A}\,\mathrm{sat}(\mathbf{x}^*) + \delta\mathbf{B}\mathbf{u} + \delta\mathbf{i}) = \mathrm{sign}(\mathbf{x}^*)\,,$

where $\mathbf{1}$ is a vector with all its entries equal to 1. Equations (*i*–*ii*) are to be understood component-wise, further, they contain the desired bounds on perturbation parameters in an implicit form. This will be demonstarted below by means of a typical CNN application, namely that of horizontal line detection (HLD). Note that since we take the same initial and input values for the unperturbed as well as for the perturbed system, the bounds obtained this way depend only on the template parameters. Although for any perturbation of the parameters exceeding these bounds the CNN will fail to converge in the desired saturation region, meeting these bounds does not in general guarantee the output invariance proparty. This is due to the fact that the derivation of these bounds does not take into account the transient behavior of the cells. However, for the class of tasks that are performed by CNNs admitting monotonic state trajectories, it can be shown that these bounds are sufficient as well.

The information conveyed by these bounds can be used to compare different sets of predesigned templates, each of which solves the same specific task, in terms of their degree of robustness. If the permissable uncertainity of a set turns out to be less than about 5% (imposed by analog implementaion), they will be precluded for use in analog applications.

The following choice of templates provides us with a minimal set of parameters needed to perform HLD

$$A = \begin{pmatrix} 0 & 0 & 0 \\ a & b & a \\ 0 & 0 & 0 \end{pmatrix} \quad B = \mathbf{0} \quad i = c\,. \qquad (6)$$

In contrast to the bold face notation the matrices A, B and the scalar i denote the cloning templates of a spatially invariant CNN. Let us now assume, in agreement with our previous definition of robustness, that all parameters are subject to the same amount of maximum deviation ε, i.e.,

$$\delta\mathbf{A} = \begin{pmatrix} \pm\varepsilon & \pm\varepsilon & & & \\ \pm\varepsilon & \pm\varepsilon & \pm\varepsilon & & \\ & \ddots & \ddots & \ddots & \\ & & \pm\varepsilon & \pm\varepsilon & \pm\varepsilon \\ & & & \pm\varepsilon & \pm\varepsilon \end{pmatrix}, \quad \delta\mathbf{i} = \begin{pmatrix} \pm\varepsilon \\ \vdots \\ \pm\varepsilon \end{pmatrix}$$

and $\delta\mathbf{B} = \mathbf{0}$. Substituting these into (*i*-*ii*) leads to the following bounds

$$0 \leq \varepsilon \leq \frac{1}{4}\left(\min_{i,j}(|x^*_{ij}|) - 1\right). \qquad (7)$$

Equation (7) gives the desired upper bound for the set of templates A, B, i which lead to $\mathbf{x}^*$.

In the followng we compare 3 different sets of templates, each of which solves the HLD problem, in terms of their robustness:

a)

$$A = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 2 & 1 \\ 0 & 0 & 0 \end{pmatrix} \quad B = \mathbf{0} \quad i = -1,$$

b)

$$A = \begin{pmatrix} 0 & 0 & 0 \\ 0.5 & 1.75 & 0.5 \\ 0 & 0 & 0 \end{pmatrix} \quad B = \mathbf{0} \quad i = -0.5,$$

and c)

$$A = \begin{pmatrix} 0 & 0 & 0 \\ 3 & 7 & 3 \\ 0 & 0 & 0 \end{pmatrix} \quad B = \mathbf{0} \quad i = -4.$$

Simulations were carried out by initializing all three template sets with a given binary image, with white (represented by -1) corresponding to the background. The binary image was chosen such that it contained all possible configurations that may occur in a HLD task. This is crucial to the fact that the bounds obtained this way will then be independent of the input binary image. Further, the boundary cells were set to the fixed value of zero. After convergence was reached, the minimum over $|x^*_{ij}|$ was found for each set in order to determine ε. The following results were obtained:

$$\begin{array}{ll} \varepsilon = 0 & \text{for } a) \\ \varepsilon \le 0.0625 & \text{for } b) \\ \varepsilon \le 0.5 & \text{for } c). \end{array}$$

Clearly, set $a)$ cannot be robust. Indeed, simulations show that the correct operation is very sensitive to I, namely below 10^{-5}. Further, the set $b)$ is still rather sensitive. The tolerance required by analog implementation, i.e., 5 % of the nominal value of the parameters, cannot be guaranteed here. By contrast, set $c)$ is the most likely set of the three sets to be robust. Random variations of these values show a robustness of up to $\varepsilon = 0.5$.

III. Robust Templates

In [4,5] a design approach to robust templates based on inscribing a maximal norm–body in a polytope, given by the design constraints, was introduced. We explore here the monotony property to obtain robust templates. Our approach is demonstrated in detail in the example of HLD. The proposed approach can be applied to other tasks of interest as well, insofar as they permit the design of robust templates by the monotony assumption about the states [6]. In what follows, we restrict ourselves to binary input and output data. Note that once we have robust template parameters the restriction of binary inputs can be relaxed to allow noisy inputs with the noise variance being dependent on the degree of the robustness.

From (1) and (6) we obtain the following differential equation for each cell

$$\frac{dx_{ij}}{dt} = -x_{ij} + a\,\mathrm{sat}(x_{ij-1}) + b\,\mathrm{sat}(x_{ij}) + a\,\mathrm{sat}(x_{ij+1}) + c\,.$$

Depending on the initial state of a cell and its neighboring cells, the desired performance can be obtained if we can impose one of the following equations simultaneously for all cells

$$a\,\mathrm{sat}(x_{ij-1}) + b\,\mathrm{sat}(x_{ij}) + a\,\mathrm{sat}(x_{ij+1}) + c < x_{ij} \qquad (8)$$

$$a\,\mathrm{sat}(x_{ij-1}) + b\,\mathrm{sat}(x_{ij}) + a\,\mathrm{sat}(x_{ij+1}) + c > x_{ij}\,. \qquad (9)$$

To guarantee binary outputs, i.e., $|x^*_{ij}| \ge 1$, instead of (8–9) we require

$$a\,\mathrm{sat}(x_{ij-1}) + b\,\mathrm{sat}(x_{ij}) + a\,\mathrm{sat}(x_{ij+1}) + c \le -1 \qquad (10)$$

$$a\,\mathrm{sat}(x_{ij-1}) + b\,\mathrm{sat}(x_{ij}) + a\,\mathrm{sat}(x_{ij+1}) + c \ge 1\,. \qquad (11)$$

Clearly, (10–11) preserve the monotony of (8–9), respectively. Further, if, for example, a decreasing state settles at some value $x^*_{ij} > -1$, from the equilibrium condition, we will have

$$a\,\mathrm{sat}(x^*_{ij-1}) + b x^*_{ij} + a\,\mathrm{sat}(x^*_{ij+1}) + c = x^*_{ij} > -1\,,$$

which contradicts (10). In the case of HLD the following initial configurations may occur

○○○ ○●● ●●● ○●○ ●○● ●○○

From (10-11) we obtain the following *necessary* inequalities for the depicted combinations of cells at the equilibrium:

$$\begin{array}{rr} -2a - b + c \le -1 & 2a + b + c \ge 1 \\ 2a - b + c \le -1 & b + c \ge 1\,. \\ -b + c \le -1 & \end{array} \qquad (12)$$

Assuming $a \ge 0$, it can be shown that (12) extended by

$$-2a + b + c \le 1 \qquad (13)$$

provides us with a set of *sufficient* conditions to perform HLD. To show this, we consider only the following case in some detail

○●○

Since the state of the black cell shall decrease from 1 to below -1, solving the differential equation we have the following inequality in the linear region

$$x_{ij}(t) \le \left(1 - \frac{2a-c}{b-1}\right)e^{(b-1)t} + \frac{2a-c}{b-1}. \qquad (14)$$

From (12) follows that $b \geq 1$, this with the equations (13–14) imply that the state of the black cell will settle at some point below -1.

Finally, we would like to mention that (12–13) applies to both -1 or 0 boundary values. In the latter case, however, we have additional constraints which can be shown to be already implied by (12). Including the desired robustness ε into the design constraints (12), we obtain, after some calculation, the following bounds for the parameters:

$$\begin{aligned} a \geq & \ \ \varepsilon \\ b \geq & \ \ a+1+4\varepsilon \\ -b+1+4\varepsilon \leq c \leq & -2a+b-1-4\varepsilon . \end{aligned} \tag{15}$$

These values are to be understood as nominal, but for the sake of simplicity we have left out the * designation. Note that the values given in the template sets *(a–c)* above are within these bounds. Moreover, the calculated upper bounds for ε coincide with the actual ε that would be used to obtain ε-robust templates from (15).

We conclude our synthesis with a class of applications for which the correct operation of the network critically depends on the accurate ratios of the nominal template values. Differential operators constitute examples of such a class. To be specific, we investigate the heat equation

$$\partial_t u(t,\mathbf{x}) = \Delta u(t,\mathbf{x}) . \tag{16}$$

on a compact domain $\Omega \in \mathbb{R}^2$ with zero boundary conditions. By discretizing (16) in spatial components we obtain

$$\frac{du_{ij}}{dt} = u_{i+1j} + u_{i-1j} - 4u_{ij} + u_{ij+1} + u_{ij-1} .$$

This corresponds to a CNN with the following templates:

$$A = \begin{pmatrix} 0 & 1 & 0 \\ 1 & -3 & 1 \\ 0 & 1 & 0 \end{pmatrix} \ B = \mathbf{0} \ \ i = 0 . \tag{17}$$

Intuitively, it is clear that disturbing *any* entry of the matrix A will result in a different dynamic behavior, e.g., if we change -3 by an additive constant, say α, (16) will become

$$\partial_t v(t,\mathbf{x}) = \Delta v(t,\mathbf{x}) + \alpha v(t,\mathbf{x}) . \tag{18}$$

Using the trial solution

$$v(t,\mathbf{x}) = e^{\alpha t} u(t,\mathbf{x}) ,$$

with $u(t,\mathbf{x})$ an arbitrary function, it is easily shown that $u(t,\mathbf{x})$ has to satisfies (16). In other words, by changing -3 to $-3+\alpha$, the solution of (16) becomes modified by an exponentially increasing or decreasing factor, depending on the sign of α. Further, if we change one of the 1's in A to say $1+\alpha$, no matter how small α, we will not only destroy the rotational symmetry of the Laplace operator but also end up with unstable solutions. This is best seen by calculating the eigenvalues of the modified Laplacian in the Fourier transformed domain. The eigenvalues corresponding to the modified direction will have a positive real part leading to divergence.

IV. Conclusion

We derived upper bounds for parameter perturbations by imposing the output invariance at the equlibrium. This in turn enabled us to compare different sets of templates with respect to their robustness. Applications were presented for horizontal line detector. We presented further a general approach to the design of robust templates for binary input-output CNNs. Our approach resides on the assumption that the corresponding task can be performed by CNNs whose states evolve monotonically. Our method of design was demonstrated in some detail on the example of horizontal line detection. We showed that templates used to implement some partial differential equations are inherently highly sensitive with respect to template entries.

V. Acknowledgement

The first author would like to thank Franz Merkel for discussions.

References

[1] D. Lím and G. S. Moschytz: A Programmable, Modular CNN Cell. *Proceeding of the Third IEEE international Workshop on CNNs and their Applications (Rome), pp. 79–84, Dec. 1994.*

[2] R. Domínguez-Castro, S. Espejo, A. Rodríguez-Vázquez and R. Carmona: A CNN Universal Chip in CMOS Technology. *Proceeding of the Third IEEE international Workshop on CNNs and their Applications (Rome), pp. 91–96, Dec. 1994.*

[3] D. Lím and G. S. Moschytz: A CNN-based signal processor. Submitted to *CNNA–96, Sevilla*

[4] G. Seiler, A. J. Schuler and J. A. Nossek: Design of Robust Cellular Neural Networks. *IEEE Trans. Circuits Syst., CAS-I, vol. 40, pp. 358–364, 1993.*

[5] I. Fajfar and F. Brotkovič: Statistical Design Using Variable Parameter Variances and Application to CNNs. *Proceeding of the Third IEEE international Workshop on CNNs and their Applications (Rome), pp. 147–152, Dec. 1994.*

[6] B. Mirzai, D. Lím and G. S. Moschytz: Robust CNN Templates for VLSI Applictions: Theory and Simulations. Submitted to *CNNA–96, Sevilla*